الوحدة الأولى

مقدمة إلى أنظمة المعلومات

Introduction to Information Systems (IS)

محتويات الوحدة

الموضوع	الصفحة
1. المقدمـــة	5
1.1 تمهيد	5
2.1 أهداف الوحدة	5
3.1 أقسام الوحدة	5
4.1 القراءات المساعدة	6
5.1 ما تحتاج إليه لدراسة الوحدة	6
2. أنظمة المعلومات وخصائصها Information Systems and their Characteristics	7
3. أنواع أنظمة المعلومات Types of Information Systems	9
4. دورة حياة نظام المعلومات An Information System Life Cycle	12
5. محلل الأنظمة Systems Analyst	14
6. الخلاصة	15
7. لمحة عن الوحدة الدراسية الثانية	15
8. مسرد المصطلحات	16
9. المراجع	18

1. المقدمة

1.1 تمهيد

أهلاً بك – عزيزي الدارس– إلى الوحدة الأولى من مقرر" 1380 تحليل الأنظمة وتصميمها"، وهي بعنوان" مقدمة إلى أنظمة المعلومات".

إن أنظمة المعلومات كغيرها من الأنظمة الأخرى لها عناصر ومكونات ولها خصائص تميزها عن غيرها، وهنالك العديد من الأنواع لأنظمة المعلومات ولكل منها خصائص تميزه وأهداف محددة لاستخدامه.

ويُشبّهُ العلماء {انظر المرجع رقم 1} نظام المعلومات بالنظام الهندسي حيث إنه عند القيام ببنائه، بحاجة في البداية إلى تخطيط وتحليل حيث تتم دراسة الجدوى من بنائه ومن ثم يتم تصميمه وبرمجته. وتسمى هذه الخطوات بدورة حياة النظام.

إن هذه الوحدة هي مقدمة أساسية لبقية المقرر وسيتم فيها تقديم شرح مفصل لكل هذه الأفكار بطريقة تساعدك عزيزي الدارس على فهم تحليل الأنظمة فهماً جيداً ومفصلاً.

2.1 أهداف الوحدة

يتوقع منك عزيزي الدارس بعد الانتهاء من دراسة هذه الوحدة أن تكون قادراً على أن:

1. تعّرف نظام المعلومات.
2. تحدد خصائص وعناصر نظام المعلومات.
3. تفرق بين أنواع أنظمة المعلومات.
4. تشرح دورة حياة أنظمة المعلومات.
5. تذكر خصائص محلل الأنظمة.

3.1 أقسام الوحدة

تعتبر هذه الوحدة مقدمة مهمة لبقية وحدات المقرر، وهي مكونة من أقسام أربعة رئيسة: القسم الأول: يعنى بشرح نظام المعلومات من حيث تعريفه وخصائصه وعناصره وهذا يغطي الهدفين الأول والثاني. أما القسم الثاني فيعنى ببيان أنواع أنظمة المعلومات وهذا يغطي الهدف الثالث. وأما القسم الثالث: فيتطرق إلى شرح مراحل بناء أنظمة المعلومات، وهذا يحقق الهدف الرابع. وأما القسم الرابع والأخير: فيقدم شرحاً وافياً عن

شخصية المحلل وخصائصه وصفاته والقدرات التي يجب أن تتوفر فيه، وهذا يحقق الهدف الخامس.

4.1 القراءات المساعدة

عزيزي الدارس: حاول أن تراجع القراءات التالية لما لها من أهمية واتصال مباشر ووثيق بموضوع هذه الوحدة، وهذا سيساعدك على فهمها واستيعابها بعمق، وهذه القراءات هي:

1-Aktas, Z., Structured Analysis and Design of Information Systems, Prentice-Hall, 1987.
2- Edwards, P., Systems Analysis, Design, and Development with Structured Concepts, Rinehart and -Winston, 1999.
3- Hoffer, J. et al., Modern Systems Analysis & Design, Second Edition, Addison-Wesley, 1999.
4- Kendal, K., and Kendal, J., Systems Analysis and Design, 5th Ed.,2002.
5- Lejk, M. and Deeks., An Introduction to Systems Analysis Techniques, 2nd Ed., Pearson Education Limited, 2002.
6- Satzinger, J., Systems Analysis and Design, 2nd Ed., Thomson Learing, 2002.
7- Silver, G. and Silver, M., Systems Analysis and Design, Addison-Wesley, 1996.
8- http://www.psut.edu.jo / computer science/ staff/saleh abu-soud

5.1 ما تحتاج إليه لدراسة الوحدة

إن كل ما تحتاج إليه–عزيزي الدارس– لدراسة هذه الوحدة واستيعابها هو أن تكون أولاً مستعداً لذلك وقد قمت بتهيئة المكان المناسب للدراسة. حاول الاطلاع على المراجع المخصصة لهذه الوحدة وقراءة ما تستطيع منها لما لها من أثر في إثراء معلوماتك بموضوع هذه الوحدة.

2. أنظمة المعلومات وخصائصها
Information Systems and their Characteristics

"النظام" عبارة عن مصطلح يستخدم في العديد من فروع الدراسة والمعرفة. وقد تطور استخدامه بشكل ملحوظ إثر التطور الكبير الذي حصل في أنظمة الحاسوب وانتشار أنظمة المعالجة الإلكترونية (Electronic Data Processing EDP). إن أنظمة التعليم، والأنظمة الاجتماعية، والأنظمة الهندسية والصناعية وأنظمة الأعمال، وكثيراً من الأنظمة وغيرها تعتبر من الأمثلة على الأنظمة المستخدمة في عالمنا اليوم.

وللنظام أكثر من تعريف، وذلك لأن الأنظمة، سواءً أكانت أنظمة حاسوبية أم غير حاسوبية، تختلف بعضها عن بعض باختلاف التطبيق، ولكن لها صفات مميزة تشترك فيها كل الأنظمة على اختلاف أنواعها؛ فكل الأنظمة لها **عناصر** (Elements) مثل المستخدمين، والوثائق، والحواسيب، والقوانين والإجراءات المتبعة، الخ ... ويجب توفر **بيئة مناسبة** (Environment) للنظام فبيئة البنك تكون في المدينة قريبة من التجمعات السكنية، أما بيئة المصنع الكيميائي فتكون أكثر مناسبةً إذا كانت في الصحراء مثلاً وبعيدة عن السكان، وتمتاز كل الأنظمة بوجود **تفاعل** (Interaction) بين العناصر بعضها مع بعض من جهة وبين العناصر والبيئة من جهة أخرى. وأخيراً يجب أن يكون للنظام **أهداف** (Goals) محددة مسبقاً بحيث يعمل النظام على تحقيقها.

وبذلك يمكن – عزيزي الدارس– تعريف **النظام** بأنه عبارة عن: **مجموعة منظمة من العناصر الموجودة في بيئة مناسبة بحيث يوجد تفاعل بين هذه العناصر والبيئة الخارجية لتحقيق مجموعة من الأهداف المعرفة والمتفق عليها مسبقاً.**

نشاط (1)

ابحث عن تعريفات أخرى للنظام، وقم بتلخيصها ومقارنتها بعضها ببعض، ثم اكتشف أوجه التشابه والاختلاف في هذه التعريفات.

لتعريف نظام المعلومات، يجب أن نعرف أولاً معنى كلمة معلومات. وفي الواقع هناك ثلاثة مصطلحات تستخدم كثيراً وبالتناوب خاصة في الأنظمة الحاسوبية وهي:

- بيانات (Data): وهي مجموعة الحروف والرموز والأرقام الخام التي ليس لها معنى أو بمعنى أكثر دقة، لا تفيد باتخاذ القرارات.

- معلومات :(Information) وهي البيانات المعالجة والمصنفة والمبوبة بحيث تصبح مفيدة لاتخاذ القرارات الحالية.
- معرفة (Knowledge): وهي المعلومات ذات المعنى الأكثر تجرداً والتي تكون مفيدة للاستنتاج واتخاذ القرارات المستقبلية.

أسئلة التقويم الذاتي (1)

بين الفرق بين البيانات، والمعلومات، والمعرفة.

بعد هذه التعريفات المهمة، أصبح بإمكاننا الآن تعريف نظام المعلومات، حيث يمكن تعريفه بأنه ذلك النظام الذي يستخدم لتزويد المستخدمين كافة في جميع المستويات الإدارية بالمعلومات. وبذلك يمكننا القول- عزيزي الدارس- بأن النظام الجيد هو الذي تكون مخرجاته معلومات مفيدة كحد أدنى، ولذلك تسمى بأنظمة المعلومات، ولتحقيق ذلك يجب أن يتصف نظام المعلومات بالصفات الآتية:

- يجب أن يتصف بالمعالجة الفعالة للمعلومات من حيث التأكد من صحة مدخلاته، والاستغلال الأمثل للأجهزة والبرمجيات.
- يجب أن يتصف كذلك بالإدارة الفعالة للمعلومات من حيث العناية بعمليات إدارة الملفات والاهتمام بالحفاظ على سرية البيانات وسلامتها.
- يجب أن يتصف نظام المعلومات بالمرونة التامة بحيث يمتاز بسهولة الإضافة والتعديل والشطب عليه، وكذلك أن يكون مرناً من ناحية الزيادة في حجم العمل وعدد العمليات في المستقبل.
- يمتاز النظام الجيد برضاء المستخدم عن أدائه وبالطبع إذا لم يكن النظام جيداً فمن الصعب أن يرضى مستخدموه عن أدائه.

أسئلة التقويم الذاتي (2)

ما الفرق بين المعالجة الفعالة والإدارة الفعالة للأنظمة؟

3. أنواع أنظمة المعلومات
Types of Information Systems

إن نظام المعلومات، كما تم تعريفه سابقاً في هذه الوحدة، يزود المستخدمين في المستويات الإدارية كافة بالمعلومات، إذ إن أي تنظيم يتكون عادة من مستويين إداريين هما:

- مستوى الأعمال اليومية الروتينية.
- المستوى الإداري الذي بدوره يُقسّم إلى عدة مستويات، يمكن إجمالها كالآتي:
  - مستوى الإدارة العليا ذات البعد الاستراتيجي ومهمتها:
    - تحديد أهداف التنظيم (Objectives)
    - تحديد الموارد (Resources) اللازمة لتحقيق هذه الأهداف
    - تحديد السياسات (Policies) التي تحكم توفير هذه الموارد واستخدامها والتخلص منها.

    المدى الزمني لنشاطات هذا المستوى من سنة إلى عشر سنوات.
  - مستوى الإدارة الوسطى ومهمتها التأكد من أن الموارد قد تم توفيرها واستخدامها بشكل فعال لتحقيق الأهداف المتفق عليها. وتكون نشاطات هذا المستوى شهرية أو سنوية.
  - مستوى الإدارة الدنيا أو الإدارة الإشرافية ومهمتها التأكد من تنفيذ المهام الصغيرة بشكل صحيح وفعال. وتكون نشاطات هذا المستوى على مستوى يومي أو أسبوعي.

أسئلة التقويم الذاتي (3)

ما هي المستويات الإدارية في التنظيم؟ وما هي مهمات كل مستوى؟

نلاحظ، عزيزي الدارس، أن لكل مستوى من هذه المستويات نشاطات تختلف عن المستويات الأخرى مما يؤدي حتماً إلى اختلاف متطلبات كل مستوى من المعلومات. فنرى التركيز على التخطيط في أعلى مستوياته في الإدارة العليا وفي أدناه في الإدارة الدنيا، ونلاحظ التركيز على الرقابة في الإدارة الوسطى وبدرجة متوسطة في الإدارة العليا.

فلذلك نستنتج بسهولة أن لكل مستوى من هذه المستويات نظام معلومات ذا مواصفات معينة بحيث يخدم نشاطات هذا المستوى بطريقة فعالة. ويمكن بيان أنواع الأنظمة المختلفة والمستوى الذي تخدمه كالآتي:

1) نظام معالجة الأعمال الروتينية اليومية (Transaction Processing System)

وينفذ هذا النوع من الأنظمة المهام الموجودة ضمن مستوى الأعمال اليومية، ويمتاز بالخصائص التالية:

- يركز هذا النوع من الأنظمة على عمليات إدخال البيانات وتخزينها ومعالجتها بشكل فعال.
- يخدم الإدارة الدنيا، أي أن الذي يستفيد من مخرجاته هم أفراد الإدارة الدنيا.
- يتعامل مع المهام الروتينية الواضحة الخطوات.
- مخرجات هذا النوع من الأنظمة عادة ما تكون رسمية ومحددة مسبقاً من قبل الإدارة ولا يجوز التلاعب بها، ككشف العلامات في جامعة ما.

ومن الأمثلة على هذا النوع من الأنظمة: نظام التسجيل في جامعة، ونظام السحب والإيداع في مصرف، ونظام شؤون الموظفين والرواتب، وغيرها الكثير من الأنظمة.

2) أنظمة المعلومات الإدارية (Management Information Systems)

يساعد هذا النوع من الأنظمة الإدارة على القيام بعمليات مراقبة فعالية تنفيذ الأعمال في التنظيم حالياً، والتنبؤ بها في المستقبل. وتعتمد أنظمة المعلومات الإدارية في معلوماتها على المعلومات الداخلية للتنظيم الواردة من أنظمة معالجة الأعمال الروتينية العاملة بالذات. وتمتاز هذه الأنظمة بالخصائص التالية:

- تخدم الإدارة الإشرافية بشكل خاص، والإدارة الوسطى بشكل عام.
- تصدر هذه الأنظمة تقاريرها بشكل دوري.
- تكون التقارير على شكل ملخصات من بيانات أنظمة معالجة الأعمال الروتينية.
- تكون التقارير الناتجة عن هذه الأنظمة معرفة مسبقاً.

3) أنظمة دعم القرار (Decision Support Systems)

يعرف نظام دعم القرار بأنه ذلك النظام الذي يساعد المدير، أو كل من يستطيع اتخاذ قرار على اتخاذ قراراته، ولكن هذا النوع من الأنظمة يوجه عادة إلى الإدارة الوسطى أكثر من غيرها. وتمتاز هذه الأنظمة بالخصائص التالية:

- أنظمة تفاعلية تعتمد على طرح الأسئلة وأخذ الإجابات مباشرة من المستخدم.

- سهلة الاستخدام من قبل مستخدميها، وذلك لأن معظم مستخدمي أنظمة دعم القرار ليسوا من المتخصصين في استخدام الحاسوب.
- تحتوي على قدرات تحليلية كبيرة.
- لا تتخذ قرارات بل تساعد على اتخاذ القرار.
- تعتمد هذه الأنظمة بمعلوماتها بشكل كبير على المصادر الداخلية، وجزئياً على المصادر الخارجية.
- تعتبر أنظمة المعلومات الإدارية، وأنظمة معالجة الأعمال الروتينية، المدخلات الرئيسة للمعلومات لهذه الأنظمة.

4) أنظمة دعم القرارات الإستراتيجية (Executive Support Systems)

صممت هذه الأنظمة لاستخدام الإدارة العليا، ومساعدتها في اتخاذ قراراتها الإستراتيجية بعيدة المدى. وهذه الأنظمة لها الخصائص التالية:

- تمتاز بالقدرات التحليلية الكبيرة، والقدرة على الاستنتاج والتنبؤ بالمستقبل فيما يتعلق بالتنظيم
- تعتمد بشكل كبير على المعلومات الخارجية؛ كمعلومات المنافسين والمزودين والشركات الحليفة وغيرها من المعلومات التي يعتمد عليها التخطيط الاستراتيجي.
- تمتاز بأنظمة اتصالات قوية وإمكانيات كبيرة لعرض المعلومات.
- تعتبر أنظمة المعلومات الإدارية وأنظمة اتخاذ القرار المصادر الرئيسة للمعلومات لهذه الأنظمة.

إن معرفة نوع النظام يساعد محلل الأنظمة على بناء نظام ناجح وفعال للتنظيم يخدم المستخدمين في المستوى الإداري الصحيح.

أسئلة التقويم الذاتي (4)

ما هي أنواع أنظمة المعلومات؟ وما هي خصائص كل نوع؟

4. دورة حياة نظام المعلومات
An Information System Life Cycle

حدثت إخفاقات كبيرة في مجال أنظمة المعلومات؛ في الستينيات من القرن الماضي، إثر فشل العديد من الأنظمة الضخمة آنذاك. لقد ولّد هذا الفشل، القناعة بوجوب وجود طرق فعالة لبناء أنظمة المعلومات، ومنذ ذلك الحين، ظهر العديد من منهجيات بناء الأنظمة وتم تجريبها على أنظمة مختلفة وحقق العديد منها نجاحات مهمة في هذا المجال. إن اللافت للنظر أن مصممي كل هذه المنهجيات بلا استثناء قد أجمعوا على أن عملية بناء نظم المعلومات شبيهة بعملية بناء الأنظمة الهندسية بأنواعها كافة، كالقيام ببناء وتشغيل المباني.

ويمكن- عزيزي الدارس- تلخيص المراحل الرئيسة لعملية البناء هذه، التي تسمى بالطريقة أو المنهجية التقليدية، كما يلي:

1) مرحلة التخطيط (Planning Phase)

يقوم المحلل في هذه المرحلة:

- بجمع المعلومات اللازمة عن المشكلة الموجودة في النظام الحالي.
- بجمع متطلبات المستخدمين للنظام الجديد المراد بناؤه.
- يضع معايير حل للمشكلة موضوع الدراسة.
- يحدد القيود والضوابط على النظام الجديد كالميزانية المقدرة للمشروع ووقت التنفيذ.
- يقترح حلاً أو عدة حلول للمشكلة.

2) مرحلة التحليل(Analysis Phase)

تعتبر هذه المرحلة من أهم مراحل بناء الأنظمة، حيث يعتمد النظام الجديد على ما تم فيها. ويلعب المحلل هنا دوراً محورياً حيث يقوم المحلل بدراسة كل حل من الحلول المقترحة حسب المعايير والقيود والضوابط التي تم تحديدها في المرحلة السابقة، ومن ثم يختار أفضل الحلول.

3) مرحلة التصميم (Design Phase)

تعتمد هذه المرحلة كلياً على ما تم عمله خلال مرحلة التحليل. حيث يقوم فريق التصميم ببناء النظام على الورق من شاشات إدخال، وتقارير، وقاعدة البيانات، وخوارزميات النظام،...

4) مرحلة البناء (Development Phase)

تعتبر هذه المرحلة مرحلة البناء الفعلي للنظام، حيث تتم في هذه المرحلة برمجة النظام واختباره وتشغيله.

5) مرحلة الصيانة (Maintenance Phase)

يحتاج النظام للصيانة إما لتصليح خطأ ظهر فيه، أو لإجراء تحسينات على النظام إما بإضافة وظائف جديدة له، أو بتعديله، أو بشطب جزء منه. وتعتبر الصيانة من أكثر المراحل استنفاذاً للجهد والوقت والمال، خاصة إذا كانت طريقة بناء النظام المتبعة غير جيدة.

أسئلة التقويم الذاتي (5)

ما هي مراحل بناء نظام المعلومات؟ وما هي نشاطات كل مرحلة؟

إن مصطلح "دورة حياة النظام " يعني هنا مراحل بناء النظام، والواقع أن الطريقة المتبعة في البناء تؤثر في النهاية على صفة النظام الناتج، فإما أن يكون نظاماً ناجحاً أو نظاماً فاشلاً. فهناك طرق بناء ناجحة تؤدي إلى إنتاج نظام جيد وطرق سيئة تؤدي إلى بناء أنظمة فاشلة. وسنسمي مجموعة الطرق الجيدة بالطرق الهيكلية لبناء الأنظمة (Structured Approach) والطرق السيئة بالطرق التقليدية لبناء الأنظمة (Classical Approach). وسيتم- عزيزي الدارس- التطرق لهذه الطرق بالتفصيل في الوحدة القادمة إن شاء الله.

5. محلل الأنظمة Systems Analyst

إن عملية التحليل تعتبر من مراحل بناء النظام المهمة جداً وذلك لأن كل النظـــام يعتمد على نتائج هذه المرحلة، كما تعتبر هذه المرحلة من أصعب المراحل وأعقدها إذ إن المحلل في هذه المرحلة يتعامل مع مستخدمين معظمهم لا يجيدون اســـتخدام الحاســـوب، وفي كثير من الأحيان لا يعرفون عن الأنظمة الحاسوبية شيئاً. فيضطر محلل الأنظمـــة، حين يقوم بجمع متطلبات النظام من المستخدمين، إلى أن يستخدم اللغة الطبيعية وهذا يزيد من صعوبة هذه المرحلة وذلك لأن اللغة الطبيعية مطاطة ومفرداتها تحمـــل أكثـــر مـــن معنى. فمن مسؤولية المحلل محاورة كل من له علاقة بالنظام واستخلاص المتطلبات منهم وفهمها وتحليلها، ثم بعد ذلك تحويلها إلى شكل مفهوم لمجموعة المصممين.

لذلك يجب أن تتوافر عدة صفات في محلل الأنظمة حتى يكون قادراً على القيام بعمله بطريقة فعالة، ومن هذه الصفات:

أ- أن تكون لديه المقدرة على فهم المشكلة وإدراك عناصرها، إذ إنه لا يمكن اقتراح حل لمشكلة ما دون فهم هذه المشكلة.

ب- أن يكون لديه منطق سليم يستطيع من خلاله تحليل المشكلة.

ت- أن يكون قادراً على العمل ضمن فريق متكامل، وذلك لأن بناء الأنظمة يتطلب وجود فريق متكامل يتكون من العديد من المتخصصين.

ث- أن يتمتع بشخصية اجتماعية تمكنه من الاتصال مع الأفراد بكل المستويات الإدارية، حيث إن عمله يتطلب التعامل مع العديد من الأشخاص في التنظيم وأخذ المعلومـــات منهم ومناقشتهم والاجتماع معهم.

ج- أن يكون حاضر البديهة دائماً وجاداً في تحقيق أهدافه.

ح- أن يكون قادراً على التنبؤ بمستجدات ومتطلبات العمل المستقبلية، حيـــث إن النظـــام الناجح هو النظام المرن الذي يستوعب التغيرات المستقبلية في العمل.

خ- أن يكون قادراً على فهم واستيعاب كل من يتعامل معهم وأن يكون قادراً على التعامل مع بعض المشاكل النفسية حتى يتسنى له أخذ المعلومات اللازمة بسهولة ويسر.

أسئلة التقويم الذاتي (6)

ما هي الصعوبات التي تواجه المحلل في تحليل الأنظمة؟ وما هي الصفات التي يجب أن يتحلى بها؟

6. الخلاصة

اتضح لنا، عزيزي الدارس، من خلال دراسة هذه الوحدة أن للأنظمة أنواعاً عديدة، كل نوع منها يخدم مستوى إدارياً محدداً وله أهداف ومخرجات تختلف عن بقية الأنواع، فحين يُطلَب من المحلل بناء نظام ما لتنظيم معين، عليه أولاً دراسة المستويات الإدارية التي يتكون التنظيم منها ويحدد نوع النظام المطلوب.

ورأينا أن عملية بناء أنظمة المعلومات تشبه تماماً بناء الأنظمة الهندسية من حيث إنها تعتمد إلى حد كبير على الإبداع الذاتي للمحلل أو المهندس لأنه لا توجد طريقة علمية بحتة يمكن إتباعها فتعطينا نظاماً جاهزاً للاستخدام، وإنما كل ما لدينا طريقة نظامية تتكون من عدة مراحل لبناء الأنظمة هي: التخطيط، والتحليل، والتصميم، والبناء، والصيانة، ولا بد من تدخل المحلل في كثير من خطواتها ليقرر ويختار الأفضل من بين عدة بدائل متاحة.

كما استفدنا أيضا أن مرحلة التحليل هي من أصعب المراحل، حيث إن المحلل يتعامل مع أناس غير متخصصين في الحاسوب فيضطر إلى استخدام اللغة الطبيعية للتخاطب معهم واستخلاص متطلباتهم، وذلك عن طريق مقابلتهم ومناقشتهم. فلذلك يجب على المحلل أن يتحلى ببعض الصفات التي تجعله قادراً على القيام بعمله على أكمل وجه.

7. لمحة عن الوحدة الدراسية الثانية

تتطرق الوحدة الدراسية الثانية إلى مناقشة موضوع من أهم مواضيع تحليل الأنظمة، ألا وهو منهجيات بناء أنظمة المعلومات. والواقع أنه توجد عشرات المنهجيات المستخدمة لهذا الغرض، ولكننا سنقوم في هذه الوحدة بتصنيفها إلى منهجيتين رئيستين: المنهجية التقليدية (ذات المشاكل العديدة)، والمنهجية المهيكلة (الجيدة)، بالإضافة إلى منهجية بناء الأنظمة بالكينونات الموجهة (الحديثة). وسنقوم بدراسة هذه المنهجيات بشكل مفصل، والفرق بينها، وخصائص كل منها، بالإضافة إلى لغة النمذجة (UML) شائعة الاستخدام بين المحللين ومصممي الأنظمة.

8. مسرد المصطلحات

- **أنظمة المعالجة الإلكترونية Electronic Data Processing EDP:** هي الأنظمة التي تعتمد على الحاسبات في تنفيذ مهامها.
- **بيانات Data:** وهي مجموعة الحروف والرموز والأرقام الخام التي ليس لها معنى أو بمعنى أكثر دقة، لا تفيد باتخاذ القرارات.
- **مرحلة البناء Development Phase:** هي المرحلة الرابعة من مراحل بناء أنظمة المعلومات. وتعتبر هذه المرحلة مرحلة البناء الفعلي للنظام، حيث تتم في هذه المرحلة برمجة النظام واختباره وتشغيله.
- **مرحلة التحليل Analysis Phase:** هي المرحلة الثانية من مراحل بناء أنظمة المعلومات. وتعتبر هذه المرحلة من أهم مراحل بناء الأنظمة، حيث يعتمد النظام الجديد على ما تم فيها. ويلعب المحلل هنا دوراً محورياً حيث ويقوم المحلل بدراسة كل حل من الحلول المقترحة حسب المعايير والقيود والضوابط التي تم تحديدها في المرحلة السابقة، ومن ثم يختار أفضل الحلول.
- **مرحلة التخطيط Planning Phase:** هي المرحلة الأولى من مراحل بناء أنظمة المعلومات. ويقوم المحلل في هذه المرحلة بجمع المعلومات اللازمة عن المشكلة الموجودة في النظام الحالي ويقوم بجمع متطلبات المستخدمين للنظام الجديد ويضع معايير حل للمشكلة موضوع الدراسة ثم يحدد القيود والضوابط على النظام الجديد كالميزانية المقدرة للمشروع ووقت التنفيذ، وفي النهاية يقترح عدة حلول للمشكلة.
- **مرحلة التصميم Design Phase:** هي المرحلة الثالثة من مراحل بناء أنظمة المعلومات، وتعتمد هذه المرحلة كلياً على ما تم عمله خلال مرحلة التحليل. حيث يقوم فريق التصميم ببناء النظام على الورق من شاشات الإدخال والتقارير وقاعدة البيانات، وخوارزميات النظام.
- **مرحلة الصيانة Maintenance Phase:** هي المرحلة الخامسة والأخيرة من مراحل بناء أنظمة المعلومات. تعتبر هذه المرحلة من أكثر المراحل استنفاداً للجهد والوقت والمال، خاصة إذا كانت طريقة بناء النظام المتبعة غير جيدة.
- **معرفة Knowledge:** وهي المعلومات ذات المعنى الأعلى والتي تكون مفيدة للاستنتاج واتخاذ القرارات المستقبلية.
- **معلومات Information:** وهي البيانات المعالجة والمصنفة والمبوبة بحيث تصبح مفيدة لاتخاذ القرارات الحالية.

- **نظام دعم القرار Decision Support System:** أحد أنواع أنظمة المعلومات، ويعرف بأنه ذلك النظام الذي يساعد المدير على اتخاذ قراراته، والواقع أن كل من يستطيع اتخاذ قرار يمكن أن يستخدم أنظمة دعم القرار، ولكن هذا النوع من الأنظمة يوجه عادة إلى الإدارة الوسطى أكثر من غيرها.
- **نظام دعم القرارات الإستراتيجية Executive Support System:** أحد أنواع أنظمة المعلومات حيث صممت هذه الأنظمة لاستخدام الإدارة العليا، ومساعدتها في اتخاذ قراراتها الإستراتيجية بعيدة المدى.
- **نظام معالجة الأعمال الروتينية اليومية Transaction Processing System:** أحد أنواع أنظمة المعلومات حيث ينفذ هذا النوع من الأنظمة المهام الموجودة ضمن مستوى الأعمال اليومية.
- **نظام المعلومات الإدارية Management Information System:** أحد أنواع أنظمة المعلومات حيث يساعد هذا النوع من الأنظمة الإدارة على القيام بعمليات مراقبة فعالية تنفيذ الأعمال في التنظيم حالياً، والتنبؤ بها في المستقبل.

9. المراجع

1. Aktas, Z., Structured Analysis and Design of Information Systems, Prentice-Hall, 1987.
2. Baudoin, C. and Hollowell, G., Realizing The Object-Oriented Life Cycle, Prentice -Hall, 1996.
3. Booch, G., Rumbaugh, J., and Jacobson, I., The Unified Modeling Language User Guide, Addison- Wesley, 1999.
4. Dennis, A. et al, Systems Analysis & Design: An Object-Oriented Approach with UML. John Wiley & Sons Inc., 2002.
5. Dewitz, S., Systems Analysis and Design and the Transition to Objects, McGraw-Hill, 1996.
6. Edwards, P., Systems Analysis, Design, and Development with Structured Concepts, Rinehart and Winston, 1999.
7. Henderson, P., Object-Oriented Specification and Design with C++, McGraw Hill, 1993.
8. Hoffer, J. et al., Modern Systems Analysis & Design, 2nd Ed., Addison-Wesley, 1999.
9. Kendal, K., and Kendal, J., Systems Analysis and Design, 5th Ed., 2002.
10. Kruchten, P., The Rational Unified Process: An Introduction. 2nd Ed., Addison- Wesley, 2000.
11. Lejk, M. and Deeks. An Introduction to Systems Analysis Techniques, 2nd Ed., Pearson Education Limited, 2002.
12. Maciaszek, A., Requirements Analysis and System Design, Developing Information Systems with UML, Addison- Wesley, 2001.
13. Quatrani, T., Visual Modeling with Rational Rose 2000 and UML, Addison- Wesley, 2000.
14. Satzinger, J., Systems Analysis and Design, 2nd Ed., Thomson Learing, 2002.
15. Silver, G. and Silver, M., Systems Analysis and Design, Addison-Wesley, 1996.
16. http://www.psut.edu.jo / computer science/ staff/saleh abu-soud
17. www.rational.com
18. www.omg.com

الوحدة الثانية

منهجيات بناء أنظمة المعلومات

Information Systems Development Methodologies

محتويات الوحدة

الموضوع	الصفحة
1. المقدمـــة	23
1.1 تمهيد	23
2.1 أهداف الوحدة	23
3.1 أقسام الوحدة	23
4.1 القراءات المساعدة	24
5.1 ما تحتاج إليه لدراسة الوحدة	24
2. منهجيات بناء أنظمة المعلومات	25
1.2 المنهجية التقليدية Classical Approach	25
2.2 المنهجية المهيكلة Structured Approach	26
3.2 منهجية بناء الأنظمة بالكينونات الموجهة Object Oriented Approach ..	27
3. مقدمة إلى لغة النمذجة الموحدة Unified Modeling Language UML	29
4. الخلاصة	36
5. لمحة عن الوحدة الدراسية الثالثة	36
6. مسرد المصطلحات	37
7. المراجع	39

1. المقدمة

1.1 تمهيد

أهلاً بك – عزيزي الدارس– إلى الوحدة الثانية من مقرر "1380 تحليل الأنظمة وتصميمها"، وهي بعنوان " منهجيات بناء أنظمة المعلومات"

لبناء أنظمة المعلومات منهجيات عديدة جداً، منها المنهجيات الرديئة ومنها الجيدة. وينتج عن المنهجيات الرديئة أنظمة ضعيفة، وعادة ما تفشل عند أول تطبيق فعلي لها، بينما ينتج عن المنهجيات الجيدة أنظمة ناجحة تتصف بالكثير من الخصائص والميزات الجيدة.

وفي هذه الوحدة سيتم تمثيل المنهجيات الرديئة بالمنهجية التقليدية، والمنهجيات الجيدة بالمنهجية المهيكلة ومنهجية الكينونات الموجهة الحديثة، وسيتم بيان خصائص كل منها ونقاط القوة والضعف فيها.

وفي القسم الأخير من هذه الوحدة، سيتم بيان عناصر لغة النمذجة الموحدة UML والتي تستخدم في منهجية الكينونات الموجهة.

2.1 أهداف الوحدة

يتوقع منك، عزيزي الدارس– بعد الانتهاء من دراسة هذه الوحدة أن تكون قادراً على أن:

1. تفرق بين منهجيات بناء أنظمة المعلومات.
2. تبين ضعف المنهجية التقليدية في بناء أنظمة المعلومات.
3. توضح أهمية المنهجية المهيكلة في بناء أنظمة المعلومات.
4. تحدد أهمية منهجية بناء الأنظمة بالكينونات الموجهة.
5. تذكر عناصر لغة النمذجة الموحدة (UML).

3.1 أقسام الوحدة

تقسم هذه الوحدة إلى قسمين رئيسين. القسم الأول: يبين منهجيات بناء أنظمة المعلومات وهي المنهجية التقليدية، والمنهجية المهيكلة، ومنهجية بناء الأنظمة بالكينونات الموجهة، وهذا يغطي الأهداف من الأول حتى الهدف الرابع. أما القسم الثاني: فيبين عناصر لغة النمذجة الموحدة وأهميتها، وهذا يحقق الهدف الخامس.

4.1 القراءات المساعدة

أنصحك- عزيزي الدارس- بالإطلاع على المراجع التالية لما تحويه من معلومات مفيدة تتعلق بموضوع هذه الوحدة:

1. Aktas, Z., Structured Analysis and Design of Information Systems, Prentice-Hall, 1987.
2. Baudoin, C. and Hollowell, G., Realizing The Object-Oriented Life Cycle, Prentice- Hall, 1996.
3. Booch, G., Rumbaugh, J., and Jacobson, I., The Unified Modeling Language User Guide, Addison- Wesley, 1999.
4. Dennis, A. et al, Systems Analysis & Design: An Object-Oriented Approach with UML. John Wiley & Sons Inc., 2002.
5. Dewitz, S., Systems Analysis and Design and the Transition to Objects, McGraw-Hill, 1996.
6. Edwards, P., Systems Analysis, Design, And Development With Structured Concepts, Rinehart and Winston, 1999.
7. Henderson, P., Object-Oriented Specification and Design with C++, McGraw- Hill, 1993.
8. Hoffer, J. et al., Modern Systems Analysis & Design, 2nd Ed., Addison-Wesley, 1999.
9. Kendal, K., and Kendal, J., Systems Analysis and Design, 5th Ed. 2002.
10. Kruchten, P., The Rational Unified Process: An Introduction. 2nd Ed., Addison- Wesley, 2000.
11. Lejk, M. and Deeks. An Introduction to Systems Analysis Techniques, 2nd Ed., Pearson Education Limited, 2002.
12. Quatrani, T., Visual Modeling with Rational Rose 2000 and UML, Addison Wesley, 2000.
13. Satzinger, J., Systems Analysis and Design, 2nd Ed., Thomson Learing, 2002.
14. Silver, G. and Silver, M., Systems Analysis and Design, Addison-Wesley, 1996.
15. http://www.psut.edu.jo / computer science/ staff/saleh abu-soud
16. www.rational.com
17. www.omg.com

5.1 ما تحتاج إليه لدراسة الوحدة

عزيزي الدارس: قبل أن تبدأ بدراسة هذه الوحدة تأكد من أنك هيأت المكان الهادئ المناسب للدراسة مما يساعدك على التركيز والتفكير العميق. وتأكد من وجود قلم وأوراق لديك، وأنصحك بالاستعانة بمشرفك الأكاديمي للحصول على نسخة من برمجية Rational وحاول الإطلاع ورسم كل أنواع المخططات الواردة في هذه الوحدة. وفي أثناء الدراسة حاول الإجابة على أسئلة التقويم الذاتي مما يعطيك الفرصة لفهم أعمق لكل المفاهيم. ولا تتردد في الاتصال مع مشرفك الأكاديمي إن واجهتك أية مشكلة.

2. منهجيات بناء أنظمة المعلومات
Information Systems Development Methodologies

1.2 المنهجية التقليدية Classical Approach

إن المنهجية التقليدية لبناء أنظمة المعلومات، كما تم بيانه في الوحدة الأولى، تتكون- عزيزي الدارس- من الخطوات التالية: التخطيط، والتحليل، والتصميم، والبناء، والصيانة. لقد أثبتت التجارب أن إتباع هذه الخطوات وحدها غير كاف لإنتاج نظام ناجح يلبي متطلبات المستخدم.

إن لهذه المنهجية العديد من المشكلات التي يكمن معظمها في عدم وجود آلية أو تكنيك واضح للتوقف بعد كل خطوة رئيسة ومراجعة وفحص ما قد تم إنجازه في الخطوة السابقة، حيث إنه في حالة وجود خطأ أو خلل فيها، يتم تصحيح هذا الخلل والانتقال إلى الخطوة اللاحقة.

وفي هذه المنهجية يتم بذل جهد أقل في التحليل والتصميم وجهداً أكبر في البرمجة مما يزيد من تعقيد هذه المنهجية وصعوبة السيطرة على البرامج وعملية بنائها، حيث تسمى هذه المنهجية في كثير من الأحيان بالهرولة نحو البرمجة (Rush to Code) مما يعني أن همّ متبعي هذه المنهجية الأكبر هو البرمجة وليس التحليل أو التصميم، مما يضطرهم إلى كتابة البرامج بطريقة تفتقر إلى المعيارية أو الالتزام بالحد الأدنى من معايير الجودة. وهذا بدون أدنى شك يجعل صيانة النظام مكلفة وفي غاية الصعوبة، بل ربما يؤدي ذلك إلى عدم القدرة على ربط النظام (Integration) بعد الانتهاء من بنائه مع الأنظمة الأخرى في التنظيم، ويفسَّر ذلك بعدم وجود الوقت للتحليل وبالتالي عدم وجود الوقت الكافي للتفكير في الأنظمة الأخرى وكيفية ربطها مع النظام.

وتتصف هذه المنهجية بأنها تعتمد على الإدارة، وليس على الفنيين المختصين، في تحديد بنود المشروع وتقدير تكلفته والوقت اللازم لتنفيذه، مما يجعل تقدير هذه الأمور غير دقيق وبعيد عن الواقع.

وفي الواقع إن المشكلة الكبرى في المنهجية التقليدية تكمن في افتراض متبعيها أنه بانتهاء مرحلة التحليل تكون متطلبات مستخدمي النظام كاملة وصحيحة بشكل تام، وأن عليهم الاستمرار في مرحلتي التصميم والبرمجة بدون إشراك المستخدمين وأخذ رأيهم. وهذا بالطبع أمر خاطئ وغير صحيح على الإطلاق، حيث إن متطلبات المستخدمين لا تكون كاملة وصحيحة أثناء مرحلة التحليل، بل هي دائمة التغيير ليس خلال مرحلة

التحليل فقط وإنما طوال فترة العمل على المشروع بما يتضمن مرحلتي التصميم والبرمجة. وبالطبع هذا يؤدي إلى بناء أنظمة مليئة بالأخطاء لا يراها المستخدم إلا بعد الانتهاء من بناء النظام مما يجعل تصحيح هذه الأخطاء، كما تم ذكره مسبقاً، أمراً صعباً ومكلفاً حيث يتطلب ذلك في كثير من الأحيان إعادة التحليل والتصميم والبرمجة.

أسئلة التقويم الذاتي (1)

عدد خمس مساوئ للمنهجية التقليدية لبناء الأنظمة.

2.2 المنهجية المهيكلة Structured Approach

إن المشاكل الموجودة في المنهجية التقليدية أدت إلى ظهور الحاجة إلى منهجية جديدة مختلفة. وهذه المنهجية التي ظهرت في بداية السبعينيات من القرن الماضي تسمى بالمنهجية المهيكلة. وإن هذه المنهجية جاءت لتقدم حلولاً لكثير من المشكلات المنهجية التقليدية، فهي تركز على مرحلة التحليل والتصميم أكثر من مرحلة البرمجة وتعطيهما حقهما من الوقت والجهد.

وتعتمد هذه المنهجية على تجزئة النظام إلى أجزاء برمجية صغيرة (Modules) بحيث تكون العلاقة ودرجة الاعتمادية بينها أقل ما يمكن (Low Coupling)، وينفذ الجزء الواحد مَهمَّة واحدة (High Cohesion). وهذا يؤدي إلى بناء نظام سهل (Simple) خالٍ من التعقيد، ومرن (Flexible) بحيث تسهل الإضافة إليه والتعديل عليه والحذف منه بدون أن تتأثر بقية الأجزاء، كما يؤدي إلى سهولة اختبار النظام (Testable) حيث إنه من السهل اختبار أجزاء برمجية صغيرة من اختبار البرامج الضخمة. وإن نظاماً بهذه المواصفات من الطبيعي أن يكون قليل الحاجة إلى الصيانة وإذا احتاج إلى ذلك تكون صيانته سهلة وميسرة (Maintainable). وستتم مناقشة هذه النقاط– عزيزي الدارس – بشيء من التفصيل في الوحدة التاسعة من هذا المقرر.

وفي هذه المنهجية يتم إشراك المستخدمين في عملية بناء النظام وأخذ رأيهم في كل خطوة من خطوات البناء، وبذلك يتم ضمان رضاهم عن النظام وقبولهم إياه، مما ينتج عن ذلك نظاماً سهل الاستخدام من قبَل المستخدمين (User Friendly) يحقق رغباتهم واحتياجاتهم.

والواقع أن الميزة الرئيسة في المنهجية المهيكلة هي وجود إمكانية- بعد الانتهاء من كل خطوة رئيسة- للتوقف والتأكد من صحة الخطوة قبل الاستمرار على الخطأ بحيث يصبح تصحيحه بحاجة إلى جهد أكبر.

أسئلة التقويم الذاتي (2)

عدد خمس فوائد للمنهجية المهيكلة لبناء الأنظمة.

3.2 منهجية بناء الأنظمة بالكينونات الموجهة
Object- Oriented Approach

تعتبر تكنولوجيا الكينونات الموجهة من العبارات الطنانة الحديثة في ميدان صناعة البرمجيات، حيث تتسابق كبريات المؤسسات والشركات لتبني هذه التكنولوجيا الجديدة ودمجها في تطبيقاتها البرمجية. في الواقع، إن الغالبية العظمى من الأنظمة التي يتم تطويرها هذه الأيام تطبق تكنولوجيا الكينونات الموجهة.

إن منهجية بناء الأنظمة بالكينونات الموجهة تعتبر طريقة مختلفة لبناء الأنظمة تختلف عن المنهجيات الأخرى في كثير من الأمور، فباستخدام هذه المنهجية يتم تقسيم النظام إلى أجزاء صغيرة أو كينونات (Objects) منفصلة إلى حد ما بعضها عن بعض.

فكر -عزيزي الدارس- في هذه المنهجية كمن أراد بناء بيت من الطوب. أولاً، يجهز الطوب، ثم بعدها يضع بعضها مع بعض ليُكوّن البيت. وكذلك الأمر، عند بناء نظام ما، يتم تقسيمه إلى أجزاء برمجية بسيطة منفصلة بعضها عن بعض قدر الإمكان، وبعدها يتم جمع بعضها مع بعض لتكوّن النظام.

ومن الخصائص الهامة لمنهجية بناء الأنظمة بالكينونات هي خاصية إعادة الاستخدام (Reusability) وتعني أنه يتم بناء الأجزاء البرمجية الصغيرة بطريقة يمكن استخدامها مرات ومرات سواءً في نفس النظام الذي يتم بناؤه أو حتى في الأنظمة الأخرى. وتجدر الإشارة هنا إلى أن هذه الخاصية توفر علينا الوقت والجهد المبذولين في كتابة الأجزاء البرمجية المكررة.

وتختلف هذه المنهجية عن المنهجيات التقليدية (Traditional Approaches) بأن المنهجيات التقليدية تركز على المعلومات التي تدخل إلى النظام وكيفية تخزينها والحفاظ عليها، ثم نقوم ببناء شاشات الإدخال ونصمم التقارير التي عن طريقها نحصل على النتائج، أي أن التركيز الرئيس في هذه المنهجيات يكون على المعلومات نفسها وليس على

العمليات التي ستطبق عليها أو على سلوك النظام نفسه، لذلك تسمى مثل هذه المنهجيات المنهجيات مركزية البيانات (Data-Centric).

إن المنهجيات مركزية البيانات جيدة في حالة بناء قاعدة بيانات حيث يهمنا هنا إدخال البيانات وتخزينها واسترجاعها بسهولة ويسر. ولكن لوحظ ظهور بعض المشكلات عند استخدام هذه المنهجيات لبناء تطبيقات تجارية كاملة، ومن أهم هذه المشكلات هي أن متطلبات المستخدم دائمة التغيير مع مرور الوقت أثناء وبعد الانتهاء من بناء النظام، حيث إن إجراء تعديلات على النظام يعتبر من الأمور الصعبة.

أما منهجية الكينونات الموجهة فقد أخذت بعين الاعتبار هذه التغييرات المحتملة، حيث يتم بناء النظام بطريقة لو طُلب فيها أي تغيير وفي أي وقت، يتم إجراء هذه التغييرات بسهولة ويسر. وإن التركيز في هذه المنهجية يكون ليس فقط على المعلومات بل على سلوك النظام أيضاً.

أسئلة التقويم الذاتي (3)

بيّن خصائص منهجية الكينونات الموجهة، وبين الفرق بينها وبين المنهجيات الأخرى.

3. مقدمة إلى لغة النمذجة الموحدة
Unified Modeling Language UML

إن لغة النمذجة الموحدة عبارة عن لغة معيارية لتخصيص (Specifying)، تصور (Visualizing)، وبناء (Constructing)، وتوثيق (Documenting) عناصر النظام، يستخدمها المحللون ومصممو النظام. وسنحاول في هذا القسم عزيزي الدارس، إعطاء نبذة بسيطة جداً عن هذه اللغة واستخداماتها ومكوناتها وأنواع المخططات فيها، حيث سيتم شرح معظم هذه المخططات تباعاً من خلال وحدات هذا المقرر، وذلك من وجهة نظر محللي ومصممي الأنظمة.

لقد بُدِئ ببناء هذه اللغة في أواخر سنة 1994، عن طريق عالمين من علماء هندسة البرمجيات هما غريدي بووخ (Grady Booch) وجيم رامبو (Jim Rambaugh) اللذان كانا يعملان في شركة راشونال للبرمجيات، حين قاما بدمج طريقتي (Booch) و (OMT) بعضهما مع بعض. في خريف سنة 1995 وإنضم إليهما العالم جاكوبسون (Jacobson) الذي كان يمتلك شركة أوبجكتوري (Objectory) بطريقته المسماة (OOSE)، حيث كون هؤلاء العلماء لغة النمذجة الموحدة (UML).

وللغة النمذجة الموحدة فوائد واستخدامات عدة، أهمها:

أ- تزود المستخدمين وفريق البناء بلغة نمذجة مرئية معبرة وجاهزة للاستخدام، بحيث تمكنهم من بناء وتبادل نماذج مفيدة.

ب- تزود مستخدميها بإمكانية التوسع (Extensibility) والتعديل على النماذج بكل سهولة ويسر.

ج- تعتبر لغة النمذجة الموحدة لغةً مجردة ولا تعتمد على أية لغة برمجة بحيث يستطيع مستخدموها استخدام اللغة التي يرغبون فيها لاستكمال بناء النظام.

د- ساعدت في انتشار منهجية بناء الأنظمة بالكينونة.

هـ- لا يقتصر استخدامها على مرحلة التحليل بل يمتد إلى مراحل متأخرة من التصميم.

وتستخدم هذه اللغة لعدة أغراض أهمها:

أ- تبين حدود النظام (System Boundary) ووظائف النظام (System Functions) الرئيسة عن طريق استخدام حالات الاستخدام (Use Cases) والمتفاعلين (Actors).

ب- توضح إدراك (Realization) حالات الاستخدام عن طريق المخططات التفاعلية (Interaction Diagrams).

ج- توضح البنية التركيبية الجامدة للنظام (Static Structure) باستخدام مخططات الكينونات (Class Diagrams).

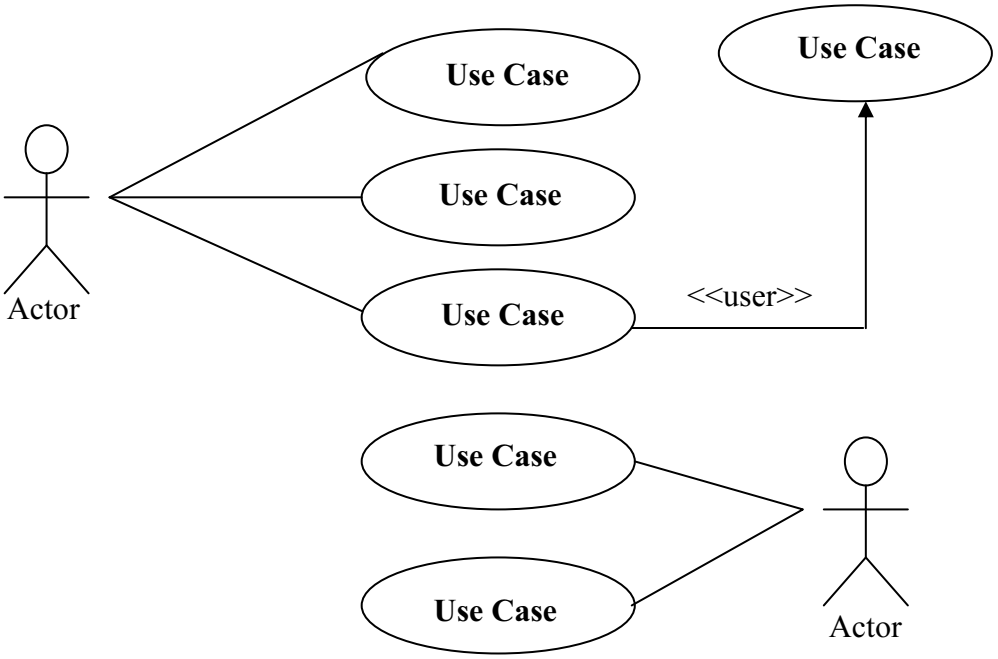
د- توضح سلوك الكينونات (Objects Behavior) باستخدام مخططات تحول الحالة (State Transition Diagrams).

هـ- تظهر التطبيق المادي الملموس للنظام (Physical Implementation).

وتتكون لغة النمذجة الموحدة من ثمانية مخططات، ولكل منها هدف معين ويستطيع المستخدمون وفريق التطوير عن طريقها أن يروا النظام من جهة معينة. ويمكن تلخيص هذه المخططات، عزيزي الدارس، كالآتي:

1) مخطط حالات الاستخدام (Use Case Diagram)

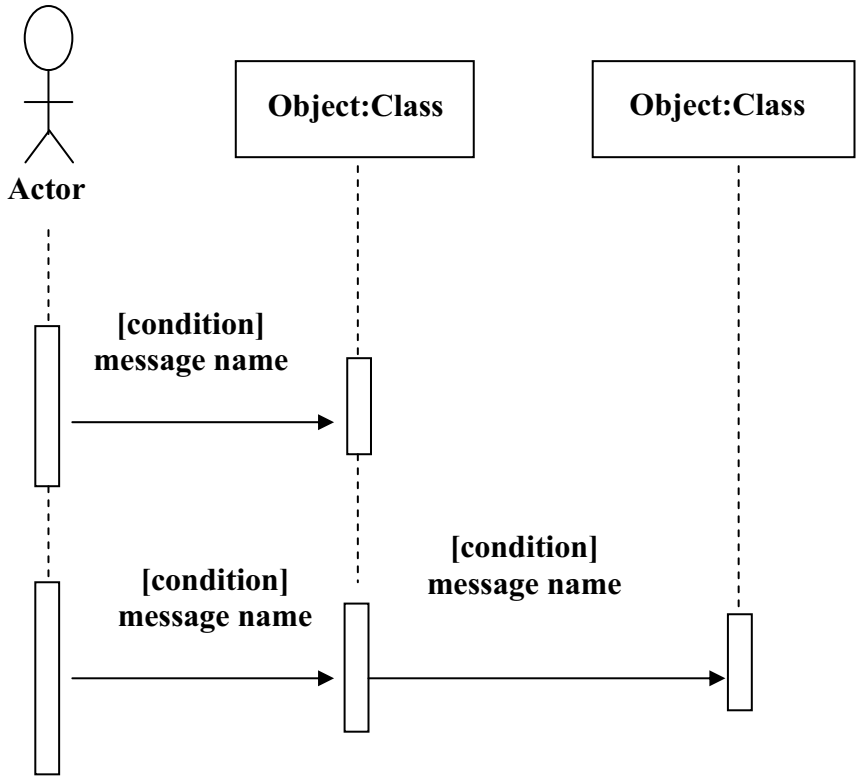
يبين هذا المخطط العلاقة بين المتفاعلين وحالات الاستخدام، حيث يمثل المتفاعلون حواف النظام وهي إما أن تكون مصدراً لبيانات النظام أو مستلماً لها. وتمثل حالات الاستخدام وظائف النظام. والشكل (1) يمثل الشكل العام لمخطط حالات الاستخدام، حيث يبين من يقوم بماذا.



الشكل (1) الشكل العام لمخطط حالات الاستخدام

2) مخططات التسلسل أو التتابع (Sequence Diagrams)

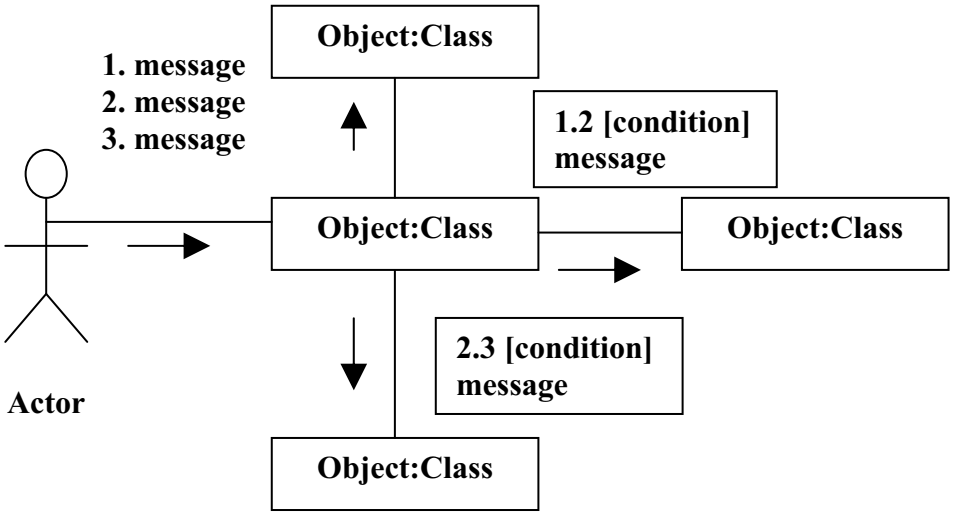
يبين هذا النوع من المخططات التسلسل الزمني للكينونات التي تشارك في التفاعل مع كينونات أخرى، ويعتبر من المخططات التفاعلية (Interaction Diagrams). ويتكون مخطط التسلسل، وكما هو مبين في الشكل (2)، من بعدين أولهما أفقي ويمثل الكينونات المشتركة في التفاعل، والبعد العامودي ويمثل الوقت.



الشكل (2): الشكل العام لمخطط التسلسل

3) مخططات التعاون (Collaboration Diagrams)

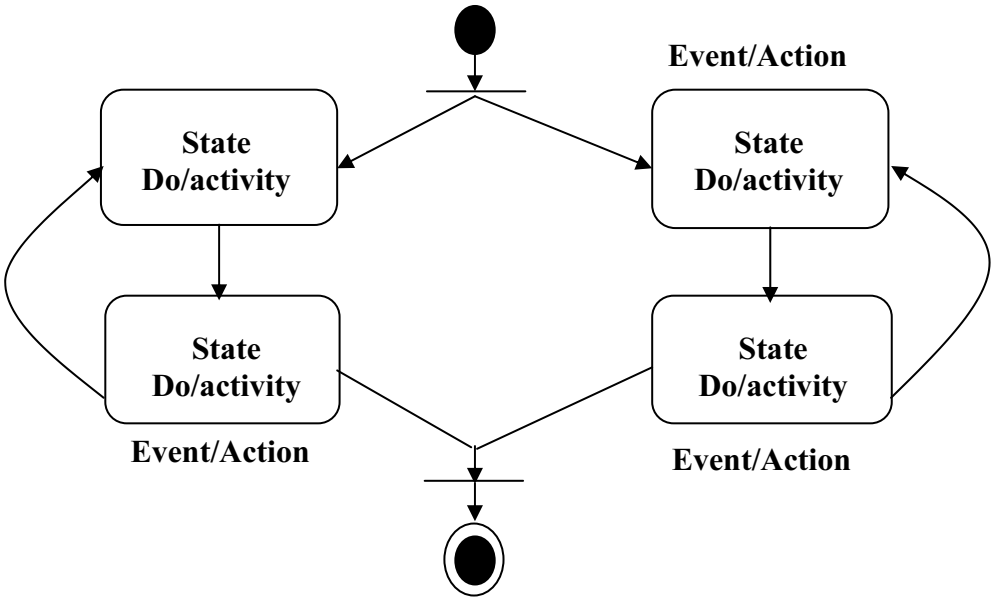
يعتبر هذا النوع من المخططات من المخططات التفاعلية (Interaction Diagrams) كذلك، حيث يبين التفاعل بين الكينونات، ويكون هذا التفاعل منظماً بطريقة تبين علاقة الكائن مع غيره من الكينونات كما تبين الرسائل التي تنتقل بين الكينونات. ويمكن ترقيم الرسائل لبيان تسلسلها الزمني. الشكل (3) يبين الشكل العام لمخطط تعاون.



الشكل (3) الشكل العام لمخطط التعاون

4) مخططات الحالة (State Diagrams)

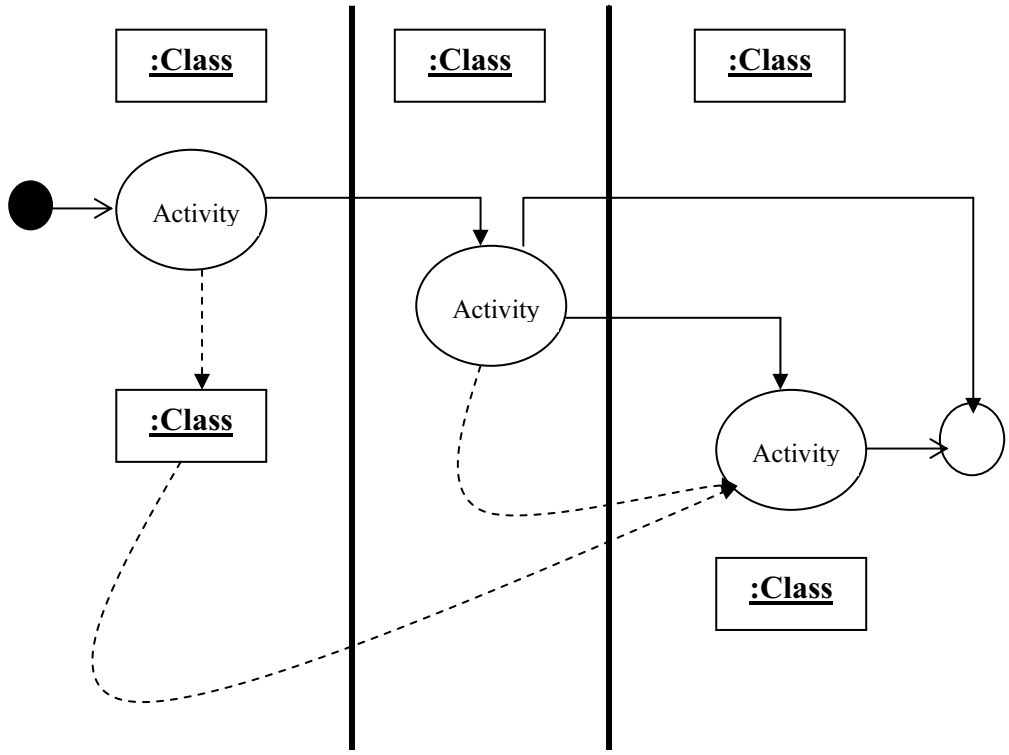
يبين مخطط الحالة تسلسل حالات الكينونات أثناء تفاعلها بعضها ببعض. ويتم الانتقال من حالة إلى أخرى نتيجة استلامها لحافز (Stimuli) بالإضافة إلى استجاباتها (Responses) وأفعالها (Actions). والشكل (4) يبين مثالاً لمخطط حالات.



الشكل (4) الشكل العام لمخطط الحالة

5) مخططات الفعالية (Activity Diagrams)

يعتبر مخطط الفعالية نوعاً خاصاً من مخططات الحالة، حيث يتم الإنتقال من حالة إلى أخرى بعد انتهاء الحالة المصدر من أعمالها. ويركز هذا المخطط على التدفق (Flow) الناتج عن المعالجات الداخلية للحالات. الشكل (5) يبين مثالاً لمخطط الفعالية.

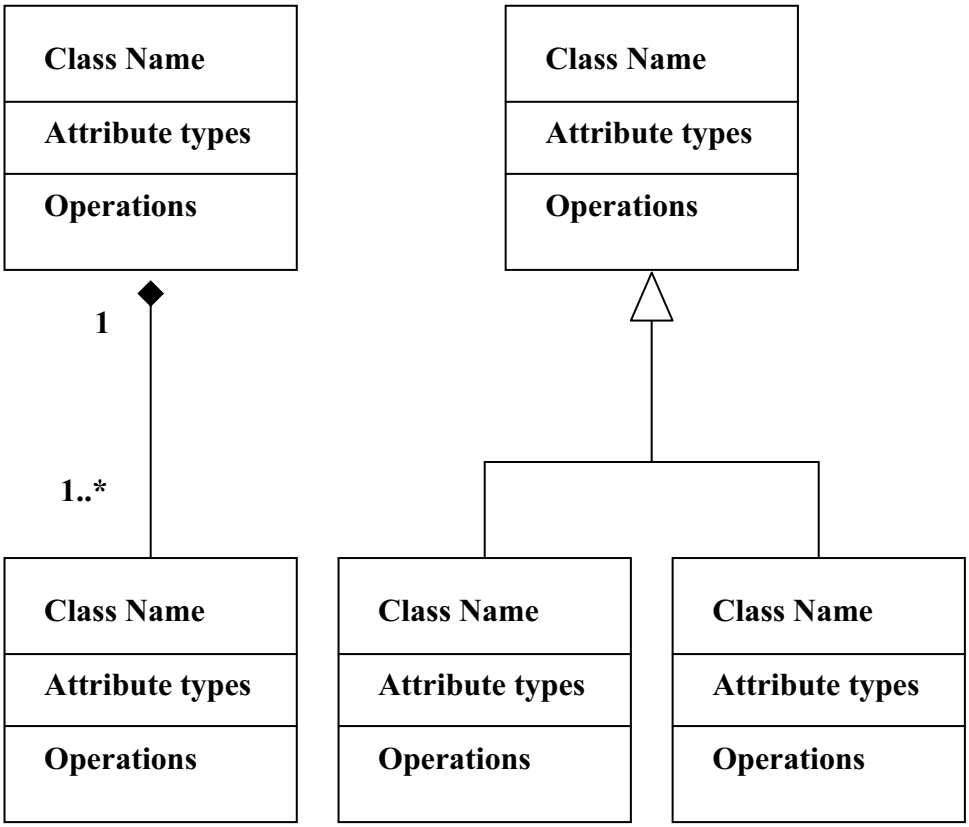


الشكل (5) الشكل العام لمخطط الفعالية.

6) مخططات الكينونات (Class Diagrams)

يبين هذا النوع من المخططات، البنية التركيبية للأصناف بالإضافة إلى محتوياتها. كما تبين العلاقات بينها مثل التوريث (Inheritance) والتشارك (Association) وغيرها من العلاقات.

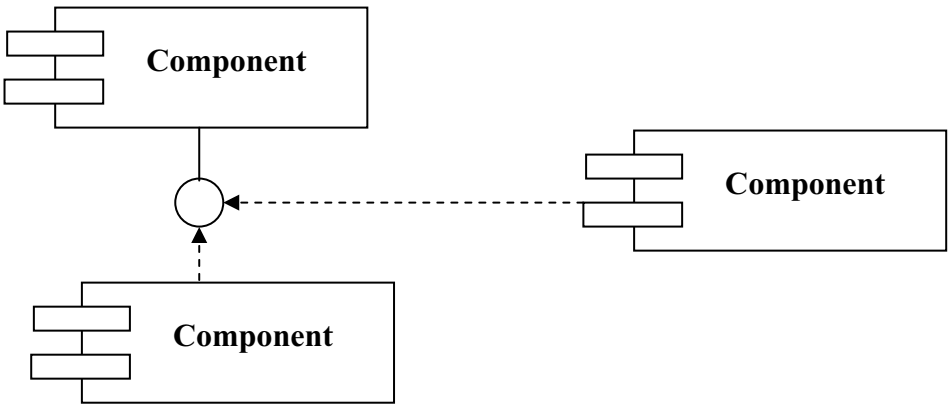
وتعتبر مخططات الكينونات العمود الفقري لكل طرق بناء الأنظمة بالكينونات الموجهة بما فيها لغة النمذجة الموحدة. الشكل (6) يبين مثالاً لمخطط الكينونات.



الشكل (6) الشكل العام لمخطط الكينونات

7) مخططات المكونات (Component Diagrams)

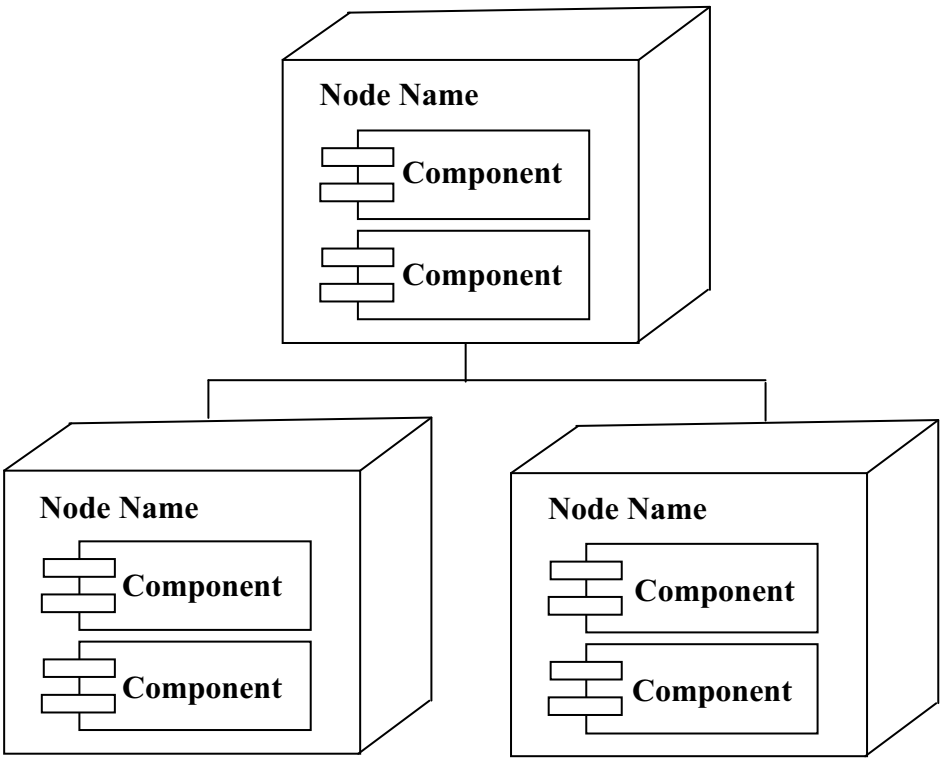
تعتبر مخططات المكونات من المخططات المادية (Physical Diagrams)، حيث تظهر طريقة ترتيب وتجميع البرنامج نفسه إلى مكونات. وتظهر هذه المخططات العلاقات بين المكونات وكيفية اعتماد بعضها على بعض. الشكل (7) يبين الشكل العـــام لمخطـــط المكونات.



الشكل (7) الشكل العام لمخطط المكونات

8) مخططات الانتشار (Deployment Diagrams)

تبين هذه المخططات الهيئة العامة (Configuration) لعناصر معالجة وقت التنفيذ بالإضافة إلى المكونات البرمجية (Software Components)، والعمليات (Processes)، والكينونات التي تعيش عليها. والشكل (8) يبين الشكل العام لمخططات الانتشار.



الشكل (8) الشكل العام لمخططات الإنتشار.

نشاط (1)

استخدم برمجية راشونال لرسم جميع أنواع المخططات الموجودة في لغة النمذجة الموحدة (UML).

4. الخلاصة

لقد لاحظت- عزيزي الدارس- من خلال دراسة هذه الوحدة أن المنهجية التقليدية تعتبر منهجية رديئة ويجب تجنبها وعدم استخدامها لبناء الأنظمة الحاسوبية وذلك لما تتصف به من صفات غير جيدة تؤثر على النظام الناتج سلباً.

بينما تعتبر المنهجية المهيكلة جيدة وينصح كل من أراد بناء نظام معلومات باتباعها وذلك لأن النظام الناتج عن استخدام هذه المنهجية يكون عادة نظاماً ناجحاً يلبي متطلبات المستخدم بشكل كبير.

ولاحظت أيضاً أن منهجية بناء الأنظمة بالكينونات الموجهة هي من المنهجيات القوية والحديثة في نفس الوقت، حيث تسهل على مستخدمها عملية بناء النظام بالإضافة إلى أن النظام الناتج عنها يتصف بصفات جيدة.

ومن الأمور التي نستطيع استخلاصها من هذه الوحدة أن لغة النمذجة الموحدة تعتبر لغة تحليل ونمذجة عالمية وشائعة الاستخدام بشكل كبير بين المحللين ومصممي الأنظمة خاصة متبعي منهجية بناء الأنظمة بالكينونات الموجهة.

5. لمحة عن الوحدة الدراسية الثالثة

سوف نتناول الحديث في الوحدة الثالثة عن ثلاثة محاور أساسية: المحور الأول: يبين مفاهيم أساسية في التحليل المبدئي ويشرح خطواته الرئيسة. المحور الثاني: يدور حول الخطوات الرئيسة للتحليل التفصيلي، ويشرح كيفية إجراء المقابلات وكتابة الاستبانات وكيفية كتابة تقرير الجدوى. أما المحور الثالث فيتطرق لبعض أدوات التحليل مثل قاموس البيانات، وأشجار القرار، وجداول القرار، والترميز.

6. مسرد المصطلحات

- **لغة النمذجة الموحدة Unified Modeling Language UML:** إن لغة النمذجة الموحدة عبارة عن لغة معيارية لتخصيص (Specifying)، وتصور (Visualizing)، وبناء (Constructing)، وتوثيق (Documenting) عناصر النظام، يستخدمها المحللون ومصممو النظام.
- **مخطط حالات الاستخدام Use Case Diagram:** يبين هذا المخطط العلاقة بين المتفاعلين وحالات الاستخدام، حيث يمثل المتفاعلون حواف النظام وهي إما أن تكون مصدراً لبيانات النظام أو مستلماً لها. وتمثل حالات الاستخدام وظائف النظام.
- **مخططات الإنتشار Deployment Diagrams:** تبين هذه المخططات الهيئة العامة (Configuration) لعناصر معالجة وقت التنفيذ بالإضافة إلى المكونات البرمجية (Software Components)، العمليات (Processes)، والكينونات التي تعيش عليها.
- **مخططات التسلسل Sequence Diagrams:** يبين هذا النوع من المخططات التسلسل الزمني للكينونات التي تشارك في تفاعل مع كينونات أخرى، ويعتبر من المخططات التفاعلية (Interaction Diagrams) . ويتكون مخطط التسلسل من بعدين أولهما أفقي ويمثل الكينونات المشتركة في التفاعل، والبعد العامودي ويمثل الوقت.
- **مخططات التعاون Collaboration Diagrams:** يعتبر هذا النوع من المخططات من المخططات التفاعلية (Interaction Diagrams) كذلك، حيث يبين التفاعل بين الكينونات، ويكون هذا التفاعل منظماً بطريقة تبين علاقة الكائن مع غيره من الكينونات كما تبين الرسائل التي تنتقل بين الكينونات. يمكن ترقيم الرسائل لبيان تسلسلها الزمني.
- **مخططات الفعالية Activity Diagrams:** يعتبر مخطط الفعالية نوعاً خاصاً من مخططات الحالة، حيث يتم الانتقال من حالة إلى أخرى بعد انتهاء الحالة المصدر من أعمالها. ويركز هذا المخطط على التدفق (Flow) الناتج عن المعالجات الداخلية للحالات.
- **مخططات الكينونات Class Diagrams:** يبين هذا النوع من المخططات، البنية التركيبية للأصناف بالإضافة إلى محتوياتها. كما تبين العلاقات بينها مثل التوريث (Inheritance) والتشارك (Association) وغيرهما من العلاقات. وتعتبر مخططات الكينونات العمود الفقري لكل طرق بناء الأنظمة بالكينونات الموجهة

بما فيها لغة النمذجة الموحدة.

- **مخططات المكونات Component Diagrams:** تعتبر مخططات المكونات من المخططات المادية (Physical Diagrams)، حيث تظهر طريقة ترتيب وتجميع البرنامج نفسه إلى مكونات. وتظهر هذه المخططات العلاقات بين المكونات وكيفية إعتماد بعضها على بعض.
- **منهجية بناء الأنظمة بالكينونات الموجهة Object- Oriented Approach:** منهجية جديدة من منهجيات بناء الأنظمة وتعتبر تكنولوجيا الكينونات الموجهة من العبارات الطنانة الحديثة في ميدان صناعة البرمجيات، حيث تتسابق كبريات المؤسسات والشركات لتبني هذه التكنولوجيا الجديدة ودمجها في تطبيقاتها البرمجية. وإن الغالبية العظمى من الأنظمة التي يتم تطويرها هذه الأيام تطبق تكنولوجيا الكينونات الموجهة وإن منهجية بناء الأنظمة بالكينونات الموجهة تعتبر طريقة مختلفة لبناء الأنظمة عن المنهجيات الأخرى في كثير من الأمور، وباستخدام هذه المنهجية يتم تقسيم النظام إلى أجزاء صغيرة أو كينونات منفصلة إلى حد ما بعضها عن بعض.
- **المنهجية التقليدية Classical Approach:** منهجية من منهجيات بناء أنظمة المعلومات، تتكون من الخطوات التالية: التخطيط، والتحليل، والتصميم، والبناء، والصيانة. ولها العديد من الخصائص السيئة التي تجعلها غير جيدة.
- **المنهجية المهيكلة Structured Approach:** منهجية من منهجيات بناء أنظمة المعلومات. ظهرت في بداية السبعينيات من القرن الماضي، وقد جاءت لتقدم حلولاً لكثير من مشاكل المنهجية التقليدية، فهي تركز على مرحلتي التحليل والتصميم أكثر من مرحلة البرمجة وتعطيهما حقهما من الوقت والجهد.

7. المراجع

1. Aktas, Z., Structured Analysis and Design of Information Systems, Prentice-Hall, 1987.
2. Baudoin, C. and Hollowell, G., Realizing The Object-Oriented Life Cycle, Prentice- Hall, 1996.
3. Booch, G., Rumbaugh, J., and Jacobson, I., The Unified Modeling Language User Guide, Addison -Wesley, 1999.
4. Dennis, A. et al, Systems Analysis & Design: An Object-Oriented Approach with UML. John Wiley & Sons Inc., 2002.
5. Dewitz, S., Systems Analysis and Design and the Transition to Objects, McGraw-Hill, 1996.
6. Edwards, P., Systems Analysis, Design, and Development with Structured Concepts, Rinehart and Winston, 1999.
7. Henderson, P., Object-Oriented Specification and Design with C++, McGraw- Hill, 1993.
8. Hoffer, J. et al., Modern Systems Analysis & Design, 2nd Ed., Addison-Wesley, 1999.
9. Kendal, K., and Kendal, J., Systems Analysis and Design, 5th Ed. 2002.
10. Kruchten, P., The Rational Unified Process: An Introduction. Second Edition, Addison Wesley, 2000.
11. Lejk, M. and Deeks. An Introduction to Systems Analysis Techniques, 2nd Ed., Pearson Education Limited, 2002.
12. Maciaszek, A., Requirements Analysis and System Design, Developing Information Systems with UML, Addison- Wesley, 2001.
13. Quatrani, T., Visual Modeling with Rational Rose 2000 and UML, Addison -Wesley, 2000.
14. Satzinger, J., "Systems Analysis and Design", 2nd Ed., Thomson Learing, 2002.
15. Silver, G. and Silver, M., Systems Analysis and Design, Addison-Wesley, 1996.
16. http://www.psut.edu.jo / computer science/ staff/saleh abu-soud
17. www.rational.com
18. www.omg.com

الوحدة الثالثة

جمع وتحليل متطلبات النظام

System Requirements Collection and Analysis

محتويات الوحدة

الموضوع	الصفحة
1. المقدمـــة	45
1.1 تمهيد	45
2.1 أهداف الوحدة	45
3.1 أقسام الوحدة	46
4.1 القراءات المساعدة	46
5.1 ما تحتاج إليه لدراسة الوحدة	47
2. التحليل المبدئي Preliminary Analysis	48
1.2 طلب التحليل User Request	48
2.2 تقويم طلب التحليل Evaluation of User request	48
3.2 تحديد المشكلة والتأكد منها Problem Statement	49
4.2 التقرير المبدئي Preliminary Report	49
3. التحليل التفصيلي Detailed Analysis	51
1.3 جمع المتطلبات وإيجاد الحقيقة Requirements Gathering and Fact Finding	51
1.1.3 المقابلات Interviews	52
2.1.3 الإستبانات Questionnaires	53
3.1.3 المشاهدة والملاحظة Observation	55
2.3 التقرير التفصيلي ودراسة الجدوى Detailed Report and Feasibility Study	55
4. أدوات تحليل الأنظمة Analysis Tools	57
1.4 قاموس البيانات Data Dictionary	57
2.4 جداول القرار Decision Tables	57
3.4 أشجار القرار Decision Trees	60
4.4 الترميز Coding	60
5. الخلاصة	64
6. لمحة عن الوحدة الدراسية الرابعة	65
7. إجابات التدريبات	65

8. مسرد المصطلحات 74
9. المراجع 75

1. المقدمة

1.1 تمهيد

أهلاً بك- عزيزي الدارس- إلى الوحدة الثالثة من مقرر "1380 تحليل الأنظمة وتصميمها"، وهي بعنوان " جمع وتحليل متطلبات النظام".

إن مرحلة التحليل تعد من أهم مراحل بناء النظام، إذ إن بقية المراحل تعتمد على نتائجها، وتعتبر جمع المتطلبات وتحليلها من أصعب المهام في مرحلة التحليل.

وتقسم مرحلة التحليل إلى قسمين رئيسين، هما:

القسم الأول: التحليل المبدئي، ويتم فيه استلام طلب من المستخدم يبين فيه أن هناك مشكلة في النظام، فيقوم المحلل بالتأكد من هذه المشكلة، وإذا كانت صحيحة ينتقل إلى القسم الثاني.

القسم الثاني: التحليل التفصيلي، ويتم فيه فهم المشكلة بشكل عميق، وجمع متطلبات المستخدمين للنظام الجديد، عن طريق إجراء المقابلات وكتابة الاستبانات، ومراقبة النظام الحالي، ثم يقوم المحلل باقتراح عدة حلول للمشكلة ويوثقها في تقرير دراسة الجدوى، وفي النهاية يقوم بعرض ومناقشة تقرير دراسة الجدوى مع الإدارة ليحصل على موافقة عليه.

في هذه الوحدة ستدرس- عزيزي الدارس- خطوات ومهام مرحلة التحليل بالتفصيل.

2.1 أهداف الوحدة

يتوقع منك- عزيزي الدارس- بعد الانتهاء من دراسة هذه الوحدة أن تكون قادراً على أن:

1. تقّوم طلب التحليل.
2. تكتب التقرير المبدئي.
3. تجمع متطلبات المستخدمين.
4. تُجري المقابلات.
5. تبني استبانات.
6. تنفذ دراسة الجدوى.
7. تكتب التقرير التفصيلي.
8. تستخدم أدوات التحليل.

3.1 أقسام الوحدة

تقسم هذه الوحدة إلى ثلاثة أقسام رئيسة: القسم الأول يبين المفاهيم الأساسية في التحليل المبدئي ويشرح خطواته الرئيسة؛ وهذا يغطي الهدفين الأول والثاني: والقسم الثاني يبين الخطوات الرئيسة للتحليل التفصيلي، ويتم في هذا القسم شرح كيفية إجراء المقابلات وكتابة الاستبانات وينتهي ببيان كيفية كتابة تقرير الجدوى ؛ وهذا يغطي الأهداف من الثالث حتى السابع. أما القسم الثالث والأخير، فيبين بعض أدوات التحليل مثل قاموس البيانات، وجداول القرار، وأشجار القرار، والترميز؛ وهذا يغطي الهدف الثامن.

4.1 القراءات المساعدة

عزيزي الدارس: أنصحك بمراجعة القراءات التالية لما لها من أهمية واتصال مباشر ووثيق بموضوع هذه الوحدة، وهذا سيساعدك على فهمها واستيعابها بعمق، وهذه القراءات هي:

1. Aktas, Z., Structured Analysis and Design of Information Systems, Prentice-Hall, 1987.
2. Dennis, A. et al, Systems Analysis & Design: An Object-Oriented Approach with UML. John Wiley & Sons Inc., 2002.
3. Dewitz, S., Systems Analysis and Design and the Transition to Objects, McGraw-Hill, 1996.
4. Edwards, P., Systems Analysis, Design, and Development with Structured Concepts, Rinehart And Winston, 1999.
5. Hoffer, J. et al., Modern Systems Analysis & Design, 2nd Ed., Addison-wesley, 1999.
6. Kendal, K., and Kendal, J., Systems Analysis and Design, 5th Ed., 2002.
7. Lejk, M. and Deeks., An Introduction to Systems Analysis Techniques, 2nd Ed., Pearson Education Limited, 2002.
8. Maciaszek, A., Requirements Analysis and System Design, Developing Information Systems with UML, Addison -Wesley, 2001.
9. Satzinger, J., Systems Analysis and Design, 2nd Ed., Thomson Learing, 2002.
10. Silver, G. and Silver, M., Systems Analysis and Design, Addison-Wesley, 1996.
11. http://www.psut.edu.jo / computer science/ staff/saleh abu-soud

5.1 ما تحتاج إليه لدراسة الوحدة

عزيزي الدارس: قبل أن تبدأ بدراسة هذه الوحدة تأكد من أنك هيأت المكان الهادئ المناسب للدراسة مما يساعدك على التركيز والتفكير العميق. وتأكد من وجود قلم وأوراق لديك. وفي أثناء الدراسة حاول الإجابة على التدريبات وأسئلة التقويم الذاتي. ويفضل زيارة مؤسسة أو شركة وتطبيق ما تدرسه في هذه الوحدة، إذ إن ما ستدرسه في هذه الوحدة هو عبارة عن مفاهيم تحليلية عملية يجب تنفيذها على أرض الواقع.

2. التحليل المبدئي Preliminary Analysis

يعتبر التحليل المبدئي أول خطوات مرحلة التحليل، ويتم فيه استلام طلب التحليل من أحد المعنيين بالنظام وعادة ما يكون أحد المستخدمين، ثم يقوم مدير المعلومات بتقويم هذا الطلب للتأكد من أنه جاد وأنه يحتوي على مشكلة تستحق العناء. فإذا كان كذلك، يعطيه لأحد المحللين للتأكد من المشكلة على أرض الواقع، ويكون ذلك عادة بإجراء مقابلات مع المستخدم الذي رفع طلب التحليل، ثم يقوم بكتابة تقرير بما جمع من معلومات عن المشكلة يسمى بالتقرير المبدئي ويسلمه لمديره.

سنقوم– عزيزي الدارس– خلال النقاط التالية بإعطاء شرح مفصل لخطوات التحليل المبدئي.

1.2 طلب التحليل User Request

حين يلاحظ المستخدم خللاً أو مشكلة معينة في النظام، يقوم بتقديم طلب مكتوب لتصحيح هذا الخلل إلى مديره الذي يقوم بدوره بتسليمه إلى الشخص المعني الذي عادة ما يكون مدير دائرة المعلومات. ويسمى هذا الطلب بطلب التحليل أو طلب المستخدم User Request ويسميه بعضهم الآخر طلب النظام (System Request).

ويمكن للمستخدم أن يكتب هذا الطلب على شكل مذكرة داخلية أو أن يستخدم نموذجاً خاصاً لهذا الأمر.

نشاط (1)

قم بزيارة لإحدى الشركات الموجودة في منطقتك واحصل على طلب تحليل دراسة ولاحظ محتوياته.

2.2 تقييم طلب التحليل Evaluation of User Request

يتسلم مدير دائرة المعلومات الطلب ويقوم بدراسته وتقييمه بشكل مبدئي وذلك قبل أن يقوم بتسليمه إلى أحد المحللين. وعند تقييم الطلب، يقوم المدير بطرح عدة أسئلة بينه وبين نفسه للتأكد من أن المشكلة الواردة في الطلب هي مشكلة جدية وتستحق العناء. ويمكن أن تكون الأسئلة على نمط الأسئلة الآتية:

- هل المشكلة مُلحّة؟

- هل المشكلة حرجة أو خطيرة؟
- هل حل المشكلة يصب في أهداف الشركة؟
- هل يسهم حل المشكلة في زيادة الأرباح أو تقليل التكاليف؟
- هل هناك أعمال متراكمة نتيجة عدم حل المشكلة؟
- هل يستطيع التقنيون في الشركة التعامل مع هذه المشكلة؟

إلى غير ذلك من الأسئلة. فإذا كانت الإجابات بمجملها إيجابية، يسلم المدير الطلب إلى أحد المحللين ويطلب منه التأكد من المشكلة وهل هي حقيقية كما وردت في الطلب أم لا؟

3.2 تحديد المشكلة والتأكد منها Problem Statement

المهمة الأساسية للمحلل هنا، هي التأكد من المشكلة والتحقق من أن المستخدم قد ذكر المشكلة الحقيقية في الطلب أم أنه أشكل عليه وذكر مشكلة أخرى وظنها المشكلة الحقيقية. حتى يتم ذلك، يقوم المحلل في هذه المرحلة بمقابلة من لهم علاقة بالأمر، وإذا لزم الأمر يقوم بمراقبة النظام ليتأكد بنفسه من المشكلة. طبعاً المقابلات لها أصولها وخطواتها حتى تكون ناجحة ومفيدة، لذلك سيتم شرح المقابلات بالتفصيل لاحقاً خلال هذه الوحدة.

ويقوم المحلل في هذه الخطوة بتقدير تكلفة ووقت التحليل التفصيلي، أي يقوم بتقدير التكلفة والوقت اللازمين لإيجاد الحل (ليس الحل نفسه) لهذه المشكلة.

4.2 التقرير المبدئي Preliminary Report

يقوم المحلل، بعد أن يتأكد من حقيقة المشكلة نتيجة مشاهداته للنظام ومقابلاته للمعنيين يقوم بكتابة نتيجة بحثه وما استخلصه واستنتجه من معلومات في تقرير يسمى بالتقرير المبدئي (Preliminary Report). ويحتوي التقرير على الأقسام التالية:

- وصف المشكلة: حيث يقوم المحلل هنا بكتابة وصف للمشكلة التي رفعها المستخدم، وعادة ما يكون الوصف هنا من وجهة نظر المستخدم وحسب ما ورد في الطلب.
- النتائج: يكتب المحلل في هذا القسم نتائج بحثه ومقابلاته مع المستخدمين ومشاهداته للنظام. تحتوي النتائج على طبيعة المشكلة من وجهة نظره هو وليس من وجهة نظر المستخدم.

- التوصيات: يقوم المحلل هنا بكتابة توصياته للمدير. وتشمل هذه التوصيات ما إذا كانت المشكلة حقيقية أم لا، وهل هي مُلحّة، أم يمكن الإنتظار وتأجيل حلها؟ وهل تسهم برفع سوية النظام؟ وبمعنى آخر يقوم المحلل بالإجابة عن الأسئلة التي سألها المدير حين قام بتقييم المشكلة.
- تقدير تكلفة ووقت التحليل التفصيلي: المقصود هنا تقدير الوقت والتكلفة اللازمين حتى نتعرف على طريقة حل المشكلة. وسيتم شرح التحليل التفصيلي- عزيزي الدارس- في النقطة اللاحقة.

أسئلة التقويم الذاتي (1)

اذكر باختصار خطوات التحليل المبدئي

3. التحليل التفصيلي Detailed Analysis

يكوّن التحليل التفصيلي الشق الثاني لمرحلة التحليل، فالشق الأول كان التحليل المبدئي حيث يُعنى بالتأكد من المشكلة الموجودة بالنظام، أما التحليل التفصيلي فيعنى بإيجاد حلول للمشكلة. وحتى نقترح حلولاً للمشكلة، لا بد من فهم المشكلة نفسها فهماً عميقاً ولا بد من إدراك أبعادها وأسبابها الحقيقية ودوافعها ومدى تأثيرها على النظام. هذا من ناحية المشكلة، أما من ناحية حلول المشكلة، فلا بد من أخذ آراء ومتطلبات المستخدمين (User Requirements) وكل من له علاقة بالنظام (Stakeholders) في الحلول المقترحة، حيث تسمى هذه العملية بعملية جمع المتطلبات وإيجاد الحقيقة.

لا ينصح بالبدء في التحليل التفصيلي إلا بعد الحصول عل تفويض من الإدارة للقيام بذلك، وعلى الإدارة إعلام الجميع بهذا الأمر. وتسمى هذه العملية بالتفويض والإعلام (أنظر حل التدريب (1)).

وفي الأجزاء التالية من هذا القسم سنقوم- عزيزي الدارس- بشرح التحليل التفصيلي شرحاً مفصلاً، وسنبين خطواته والمهمات التي تُنفَّذ فيه.

1.3 جمع المتطلبات وإيجاد الحقيقة
Requirements Gathering and Fact Finding

تعتبر هذه الخطوة من أهم خطوات التحليل، بل يمكن اعتبارها مرحلة التحليل برمتها، وذلك لأن فيها تتم دراسة النظام الحالي وفهمه جيداً واكتشاف الأخطاء والمشاكل الموجودة فيه، وفيها يتم اقتراح النظام الجديد الذي يحل مشاكل النظام القديم. ويقوم المحلل هنا بتحديد النقاط الآتية:

- البيانات التي يقوم النظام الحالي بجمعها.
- الكيفية التي يقوم بها النظام الحالي بمعالجة هذه البيانات.
- مخرجات النظام الحالي.
- الطرق التي يمكن إتباعها لتحسين أداء النظام الحالي.

ولكنها تعتبر من أصعب الخطوات، وذلك لأن مهمة المحلل هنا معقدة بعض الشيء، إذ إنه يتعامل مع مستخدمين عديمي الخبرة بالحاسوب بجمع متطلباتهم للنظام الجديد، وعليه أن يحللها ويصنفها ويحل التضارب الموجود بينها ويتأكد من خلوها من الأخطاء، وكل ذلك بالتنسيق مع المستخدمين.

وتعتبر صعبة كذلك لأنه على المحلل تحويل متطلبات المستخدمين إلى شكل مفهوم من قبل فريق التصميم المتخصص، ومما يزيد الأمر صعوبة أن هذا الشكل يجب أن يكون كذلك مفهوماً إلى حد ما من قبل المستخدمين.

وإن لجمع المتطلبات طرقاً ووسائل عديدة أهمها إجراء مقابلات مع المستخدمين ومن له علاقة بالنظام، وتصميم وتوزيع الاستبانات، ومراقبة النظام وملاحظة كيفية إجراء العمليات فيه، وزيارة مؤسسات مشابهة، وزيارة شركات الحاسوب. وسيتم شرح هذه الطرق في الأجزاء التالية.

1.1.3 المقابلات Interviews

تعتبر المقابلات من أفضل طرق جمع المعلومات، ولكنها بحاجة إلى ترتيب مسبق وتخطيط شامل قبل البدء بإجراء المقابلة، وذلك حتى يتم استغلال وقت المقابلة بشكل جيد والاستفادة منها بجمع معلومات وافية وصحيحة وعدم التخمين غير الدقيق، فيجب تحديد الأسئلة المراد سؤالها للمستخدم وترتيبها والتفكير بكل احتمالات الإجابة على هذه الأسئلة ووضع أسئلة لكل الإجابات المحتملة.

وحتى يتسنى لمجري المقابلة اختيار الشخص المناسب، يفضل إلقاء نظرة فاحصة على الهيكل التنظيمي للمؤسسة والوصف الوظيفي للموظفين فيها، أو يمكن تتبع بعض العمليات ذات العلاقة في المؤسسة، كتتبع عملية صرف شيك مثلاً.

وينصح بإتباع البروتوكول عند طلب إجراء مقابلة، كالاتصال مع الشخص المراد إجراء المقابلة معه، والاتفاق على موعد يناسب الطرفين. وللحصول على أفضل النتائج، يمكن إرسال نسخة عن الأسئلة له وذلك حتى تتسنى له دراستها والتحضير لها وتجهيز الإجابات الصحيحة لها.

وعند إجراء المقابلة، يفضل اتباع المراحل الآتية:

1) مرحلة الإحماء (Warm-Up)

إن الهدف الأساسي للمقابلة هو الحصول على معلومات صادقة وصحيحة، وللحصول على ذلك لا ينصح بالبدء بطرح الأسئلة حالما تبدأ المقابلة، إذ إن المستخدم يمكن أن يكون متوتراً وربما خائفاً من التغيير. فلذلك اعمل على خلق جو من الألفة مع الشخص الذي تجري المقابلة معه، واعمل على إزالة جو التوتر إن وجد، واجعله شريكاً لك في العملية لا نداً لك فيها.

2) مرحلة طرح الأسئلة Questioning

عندما تشعر بأن الوضع أصبح مهيأً ومناسباً، ابدأ بعرض سبب المقابلة ودور المستخدم في الإسهام بتحسين النظام وحل المشاكل فيه. ثم ابدأ بطرح الأسئلة، ولكن يجب إتباع الأسئلة التي تم تحضيرها مسبقاً مع وجود مرونة حسب ردود المستخدم. وينصح بإتباع الإرشادات الآتية:

- كن مستمعاً جيداً.
- لا تسأل السؤال وتجيب عليه بنفسك.
- أعط المستخدم الفرصة لإعطاء أجوبة كاملة.
- اهتم جيدا بكل ما يقال ولا تقاطع المستخدم بدون ضرورة.
- لا تحرضه على إعطاء إجابات معينة ولا تجبره على إعطاء الإجابة.
- قلّل من استخدام المصطلحات والعبارات الطنانة واستخدم اللغة المحكية.
- اكتب الإجابات ولا تعتمد على ذاكرتك.
- بإمكانك استخدام أدوات مساعدة كجهاز تسجيل أو غيره، ولكن بعد أخذ إذن المستخدم.

3) مرحلة الإنهاء (Closure)

عندما تتم الإجابة عن جميع الأسئلة، قم بإنهاء المناقشة، واسأل المستخدم إن كان لديه أي استفسار أو اقتراحات، ثم أشكره واستأذنه إن كان من الممكن أن تقابله مرة ثانيةً إن لزم الأمر.

وبعد الانتهاء من المقابلة، قم بتلخيص الإجابات بطريقة مرتبة، وراجع هذا التلخيص مع المستخدم وتأكد من أن الإجابات صحيحة. وفي النهاية يفضل إرسال رسالة شكر رسمية للمستخدم تشكره فيها على حسن تعاونه، ولا بأس بإرسال نسخة منها لمديره، وذلك للحفاظ على علاقات جيدة مع المستخدمين.

أسئلة التقويم الذاتي (2)

بين أهمية المقابلات، واذكر مراحلها.

2.1.3 الاستبانات Questionnaires

يتم اللجوء إلى استخدام الإستبانات لجمع المعلومات في الحالات التي يكون فيها عدد الأشخاص المراد جمع المعلومات منهم كبيراً، بحيث يصعب إجراء مقابلات معهم

لأن ذلك يتطلب وقتاً وجهداً كبيرين. وتستخدم الإستبانات لجمع البيانات من أعداد كبيرة من الأشخاص بطريقة اقتصادية وسريعة، ويتم تبويب وتفريغ النتائج بكل سهولة ويسر.

إن بناء الاستبانات يتطلب تخطيطاً كبيراً وتحضيراً جيداً أكثر بكثير من إجراء المقابلات، وذلك لسببين: أولهما أنه إذا حصل سوء فهم لسؤال معين في المقابلة فيمكن توضيحه للمستخدم ومناقشته، وهذا غير ممكن في حالة الاستبانات. والسبب الثاني: يكمن في أعداد المشاركين الكبيرة في الاستبانة حيث إن الخطأ سيتم انتشاره سريعاً ويصعب تصحيحه بسهولة.

وهنالك ثلاثة أمور لا بد- عزيزي الدارس- من أخذها بعين الإعتبار عند التخطيط لبناء إستبانة وتوزيعها، حيث يجب أن نضمنها:

- توزيع الاستبانة على أكبر قدر من الأشخاص المعنيين، حيث يمكن هنا استخدام الإنترنت أو البريد الإلكتروني.
- تعبئة الإستبانة بشكل صحيح من قبل الأشخاص المشمولين في العينة، وهنا لا بد من أن تكون الأسئلة قصيرة ومحددة، وسهلة الفهم، وغير منحازة وليس فيها تهديد. والواقع أن للأسئلة أشكالاً عديدة منها:

أ- أسئلة الاختيار من متعدد (Multiple Choice) حيث يُستخدَم هذا النوع من الأسئلة عندما يكون للسؤال عدة إجابات محددة لا تتعدى الأربعة أو الخمسة خيارات.

ب- أسئلة التقدير (Rating)، تستخدم هذا النوع من الأسئلة للحالات التي يكون لها خيارات كثيرة ويراد للإجابة أن تكون أكثر دقة، كأن تسأل المستخدم عن رأيه في الخدمة في كفتيريا الجامعة مثلاً، فيكون شكل الخيارات كالآتي:

جيدة (1 2 3 4 5 6 7 (8) 9 10) سيئة.

ج- أسئلة الرتبة (Ranking)، وتستخدم في الحالات التي لا نرغب فيها للمستخدم بإعطاء إجابة دقيقة لأن الذي يهمنا هنا هي رتبة الإجابة وليس الإجابة نفسها، كأن تسأل طالباً مثلاً عن رتبته بين أقرانه وليس عن علامته بعينها، فتكون الخيارات كالآتي:

- من أفضل 10% من الطلاب.
- من أقل 20% من الطلاب ... إلى آخره

د- الأسئلة المفتوحة (Open Ended)، ويستخدم هذا النوع من الأسئلة للحالات التي لا يوجد لها خيارات، كسؤال المستخدم عما إذا كان لديه اقتراحات أخرى.

أسئلة التقويم الذاتي (3)

اذكر أنواع الأسئلة المستخدمة في الاستبانات، وأعط مثالاً عن كل منها.

- إعادة الإستبيانات كاملة، ولضمان هذه النقطة، لا بد من تشجيع الشخص على إرجاع الاستبانة، فنعطيه مثلاً مغلفاً بريدياً مدفوعاً مسبقاً، أو نتفق معه على جمع الاستبانة بطريقة شخصية، أو تسليمها إلى سكرتيرة قسمه، وما إلى ذلك من الطرق...

3.1.3 المشاهدة والملاحظة Observation

بعد الحصول على المعلومات اللازمة عن طريق المقابلات والإستبانات، ينصح بمراقبة النظام وذلك للتأكد من صحة المعلومات التي تم جمعها وتأكيد فهمها بالشكل الصحيح. والمراقبة نوعان: النوع الأول: يتعلق بالعمليات والإجراءات التي يجريها النظام، وهذه تكون علنية ويمكن تنفيذها باتباع هذه الإجراءات خطوة بخطوة، ويمكن جمع النماذج الفارغة المستخدمة لدراستها والتعديل عليها إن احتاج الأمر لذلك. النوع الثاني: الذي يتعلق بمراقبة أداء الأفراد، وهذه تتم بطريقة سرية غير معلنة حتى لا يغير المستخدم من طريقة أدائه إذا شعر أن شخصاً يراقبه. ويمكن الاستعانة بكاميرات المراقبة الموجودة بالمؤسسات لهذه الغاية.

ويمكن زيارة مؤسسات مشابهة للاستفادة من طريقة عملها أو أي أفكار جديدة، كما ينصح بزيارة شركات الحاسوب وأنظمة المعلومات للاستفادة من آخر المستجدات في عالم البرمجيات والأجهزة.

2.3 التقرير التفصيلي ودراسة الجدوى
Detailed Report and Feasibility Study

في الواقع أن المحلل وعند هذه النقطة يكون قد فهم النظام الحالي فهماً مفصلاً بإجراءاته وعملياته ومشاكله، ويكون قد اقترح عدة حلول للمشاكل الموجودة في النظام الحالي، وهو جاهز ليوثق كل هذه النقاط في تقرير يسمى بتقرير دراسة الجدوى أو التقرير التفصيلي.

ويحتوي هذا التقرير على الأجزاء التالية:

- عرض للمشكلة وأهميتها.
- أهداف الدراسة، وهنا تتضمن هذه النقطة الأسباب التي دعت إلى إجراء هذه الدراسة، والغاية المرجوة من النظام الجديد.

- ملخص للنتائج التي توصل إليها المحلل، وذلك بناءً على المعلومات التي جمعها من المستخدمين وغيرهم، والمشاهدات والمراقبة التي قام بها للنظام.
- الحلول والبدائل المقترحة، ويتضمن هذا الجزء تقديراً لتكاليف كل حل والوقت اللازم لبنائه، بالإضافة إلى سيئات وحسنات كل حل. وهنا لا بد للمحلل أن يكون محايداً وصادقاً عند عرضه للبدائل وأن لا ينحاز لأحدها.
- التوصيات، هنا يحق للمحلل أن ينحاز لأحد البدائل ويوصي بالأخذ به مع إبداء الأسباب لهذا الانحياز.

عند الانتهاء من كتابة هذا التقرير يقوم المحلل بتسليمه إلى مدير دائرة المعلومات الذي بدوره يدعو إلى اجتماع لعرضه على الإدارة واختيار حل من الحلول المقترحة. ويلعب المحلل في هذا الاجتماع دوراً رئيساً ويجب أن يجهز نفسه بشكل جيد لهذا الحدث، ويُنصح المحلل هنا بعدم قراءة التقرير بل تلخيصه ومناقشته أمام الإدارة. ويمكن للمحلل استخدام أدوات مساعدة للعرض، كما يمكنه عرض بعض الأجهزة أو البرمجيات الجاهزة اللازمة للنظام الجديد. وبعد الإنتهاء من الإجتماع تعطي الإدارة قرارها الذي يتراوح بين قبول الحل المقترح أو رفضه كلياً.

وإذا تم الاتفاق على النظام المقترح، يقوم المحلل بنمذجته (Modeling) باستخدام إحدى وسائل النمذجة. ووحدات هذا المقرر اللاحقة تعنى- عزيزي الدارس- بشرح نمذجة النظام بطريقة الكينونات الموجهة.

تدريب (1)

قم بتنفيذ ما درسته في مرحلة التحليل على مشكلة في نظام معين من اختيارك.

4. أدوات تحليل الأنظمة Analysis Tools

تستخدم أدوات تحليل الأنظمة لتوثيق وتوصيف بعض جوانب التحليل كل يغطي جانباً معيناً. وفي هذا القسم ستدرس- عزيزي الدارس- أربعاً من هذه الأدوات المهمة والشائعة الاستخدام.

1.4 قاموس البيانات Data Dictionary

يستخدم قاموس البيانات لوصف البيانات الداخلة للنظام من خلال: الاسم، والتصنيف، والشكل، والاستخدام،... ويعتبر خطوة أولية فعالة لبناء قاعدة البيانات، كما أنه يحسّن الاتصال والتفاهم بين المحللين والمستخدمين، ويساعد في عمليات صيانة النظام. الشكل (1) يبين مثالاً لقاموس بيانات.

الإسم	الشكل	الموقع	الاستخدام
رقم العميل	9(5)	ملف العملاء، ملف الطلبيات	رقم يعطى لعملاء الشركة بشكل تسلسلي بحيث لا يتكرر لأكثر من عميل
اسم العميل	X(30)	ملف العملاء، شاشة الطلبيات، تقرير العملاء	اسم عميل من عملاء الشركة الحاليين

الشكل (1) مثال على قاموس البيانات

أسئلة التقويم الذاتي (4)

بين أهمية قاموس البيانات لمرحلة التحليل.

2.4 جداول القرار Decision Tables

تعتبر جداول القرار وسيلة تساعد محلل الأنظمة على تعريف المشكلة ودراسة أبعادها والتعرف على الحلول البديلة المتاحة ومن ثم الوصول إلى الحل الأمثل، كما أنها تعتبر أداة تساعد محلل الأنظمة على فهم وتحليل النظام القائم والنظام المقترح وإيضاح المنطق من وراء صناعة كل قرار. وإن استخدام جداول القرار يساعد المحلل في توضيح

القرارات والشروط في البرامج التي يصممها للمبرمج. ويقسم جدول القرار، كما هو موضح في الشكل (2) إلى أربعة أقسام: الشروط، وإجابات الشروط، والأفعال، واتخاذ القرار.

الشروط ؟ ؟Conditions	إجابات الشروط Conditions Entries
الأفعال Actions	اتخاذ القرار Action Entries

الشكل (2) الشكل العام لجدول القرار

ولتصميم جدول القرار خطوات يمكن إجمالها كالآتي:

1. حدد جميع الشروط الممكن توافرها في عملية اتخاذ القرار.
2. حدد الأفعال المتوقعة المرتبطة بالشروط.
3. أوجد عدد البدائل المتاحة ($= 2^n$ حيث n هي عدد الشروط).
4. إملأ إجابات الشروط حسب كل حالة محتملة.
5. خذ الإجابات حسب مدى توافق الشروط في كل حالة احتمالية.
6. افحص الجدول لإزالة الزيادات غير الضرورية في الحالات المحتملة

مثال (1)

مؤسسة تمنح العلاوة الاجتماعية وفقا للسياسة التالية:

1) إذا كان الموظف مواطناً ذكراً، و:
   - متزوجاً: يمنح علاوة اجتماعية مقدارها 200 دينار
   - أعزباً: يمنح علاوة اجتماعية مقدارها 100 دينار
2) إذا كان الموظف أنثى فلا تمنح علاوة اجتماعية بغض النظر عن الجنسية.

كوّن جدول القرار لهذه الحالة.

الحل:

الشروط:

الشرط الأول: مواطن أو غير مواطن.

الشرط الثاني: ذكر أو أنثى.

الشرط الثالث: متزوج أو أعزب.

الأفعال:

الفعل الأول: يمنح علاوة اجتماعية مقدارها 200 دينار.

الفعل الثاني: يمنح علاوة اجتماعية مقدارها 100 دينار.

الفعل الثالث: لا تمنح علاوة اجتماعية بغض النظر عن الجنسية.

البدائل:

عدد البدائل = 2 للقوة عدد الشروط

= 8 بدائل

الشروط/الحالات	1	2	3	4	5	6	7	8
الموظف مواطن	نعم	نعم	نعم	نعم	لا	لا	لا	لا
الجنس ذكر	نعم	نعم	لا	لا	نعم	نعم	لا	لا
الموظف متزوج	نعم	لا	نعم	لا	نعم	لا	نعم	لا
يمنح علاوة 200 دينار	√	--	--	--	--	--	--	--
يمنح علاوة 100 دينار	--	√	--	--	√	--	--	--
لا يمنح علاوة	--	--	√	√	--	√	√	√

نلاحظ أن الحالات 3، 4، 7 و 8 مكررة حيث نستطيع اختصارها إلى شرط واحد وهو عندما يكون الجنس ليس ذكراً فلا يمنح علاوة، فيصبح جدول القرار كالآتي:

الشروط/الحالات	1	2	3	4	5
الموظف مواطن	نعم	نعم	--	لا	لا
الجنس ذكر	نعم	نعم	لا	نعم	نعم
الموظف متزوج	نعم	لا	--	نعم	لا
يمنح علاوة 200 دينار	√	--	--	--	--
يمنح علاوة 100 دينار	--	√	--	√	--
لا يمنح علاوة	--	--	√	--	√

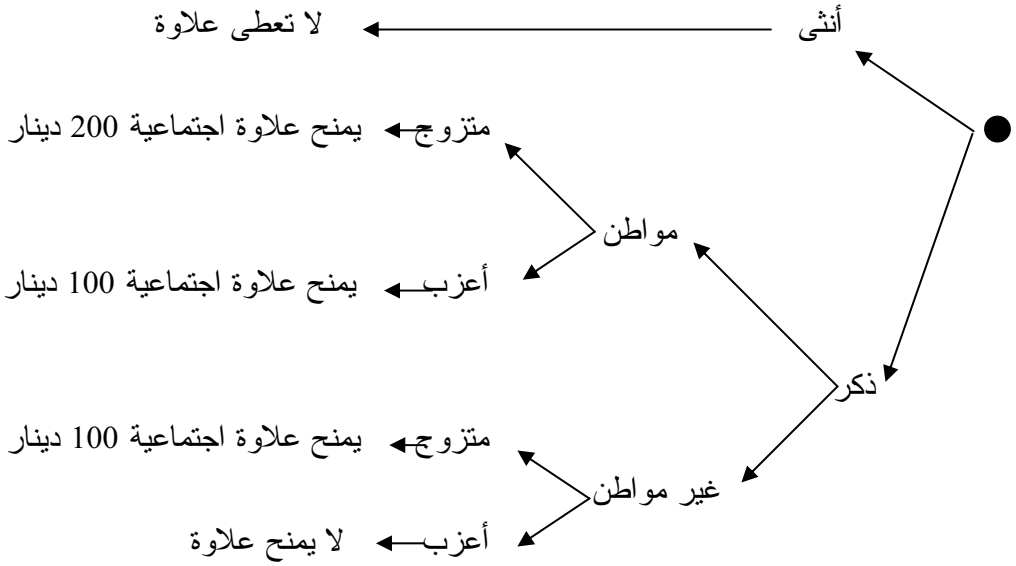
3.4 أشجار القرار Decision Trees

إن شجرة القرارات هي تمثيل تخطيطي لعملية اتخاذ القرار وتعتبر بـــديلاً عـــن جدول القرارات، حيث تمتاز بالبساطة والوضوح مقارنة بجدول القرارات.

مثال (2)

ارسم شجرة القرار للحالة الواردة في المثال (1).

الحل:



أسئلة التقويم الذاتي (5)

ما الفرق بين جداول القرار وأشجار القرار، وهل يجب استخدام النوعين معاً، أم يتم الاكتفاء باستخدام واحدة منها أثناء التحليل.

4.4 الترميز Coding

يقصد بالترميز هو تمثيل البيانات الكبيرة برموز صغيرة بحيث يسهل التعامل مع هذه البيانات ذات الحجم الكبير، وللترميز فوائد كثيرة أهمها:

- ييسر عمليات إدخال البيانات.

• يفيد بتسريع المعالجة.
• يساعد في تقليل مساحة الذاكرة المستخدمة.

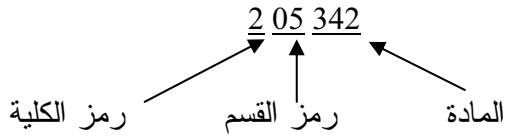
للترميز نوعان:

1. ترميز رقمي: كأن نستخدم 1 للأعزب و 2 للمتزوج.
2. ترميز حرفي: كأن نستخدم S للأعزب و M للمتزوج.

مثال (3)

ترميز المواد الدراسية في جامعة

لو فرضنا بأن رقم المادة مكون من ثلاثة أجزاء كما يلي:



فإن الحجم الأقصى في هذا الترميز: 999 مادة، 99 قسم، و9 كليات

فمثلا هذه المادة تمثل:

أ- مادة من كلية الهندسة (على اعتبار أن 2 تمثل كلية الهندسة)
ب- مادة من قسم هندسة الحاسوب (على اعتبار أن 05 تمثل هندسة الحاسوب)
ج- مادة من مستوى السنة الثالثة (على اعتبار أن 3 تمثل السنة الثالثة)
د- مادة من فئة معدات الحاسوب (على اعتبار أن 4 تمثل معدات الحاسوب)
هـ- مادة معمارية الحاسوب (على اعتبار أن 2 تمثل معمارية الحاسوب)

يمتاز الترميز ببعض الخصائص التي تجعله ترميزاً جيداً، أهمها:

أ- الدقة: تميز بين مفردات البيانات.
ب- عدم التكرار.
ج- وضوح المعنى.
د- الإختصار.
ه- التوسع والمرونة.
و- قابلية التشغيل.

ومن المزايا التي يحققها الترميز الجيد:

أ- توفير الجهد المبذول في عملية الإدخال.

ب- يقلل من احتمال الوقوع في الخطأ.

ج- يوفر في مساحة وسط التخزين المستخدم.

د- يزيد من سرعة استجابة النظام.

ه- يحافظ على سلامة البيانات وسريتها.

أنواع أنظمة الترميز:

1) الترميز المتسلسل البسيط:

يقصد به إعطاء أرقام متسلسلة لعناصر البيانات بغض النظر عن طبيعة العلاقة التي تربط بين هذه العناصر. ويناسب هذا النوع القوائم ذات العدد القليل من العناصر، كما أنه يناسب القوائم ذات العناصر التي لا تحتاج إلى تصنيف.

وتمتاز هذه الطريقة بترميز عدد غير محدد من عناصر النظام بأقل عدد من الخانات، ولكن محدودية المعلومات التي يمكن استنباطها من الرمز المتسلسل يعتبر من أهم عيوبها.

مثال (4)

عملاء شركة يقدر عددهم بـ 850 عميلاً، هؤلاء العملاء لا يهمنا تصنيفهم حسب الجنس أو المنطقة أو أي تصنيف آخر. لذلك إن أفضل طريقة ترميز لهم هي إعطاؤهم أرقام متسلسلة كالتالي:

الرقم	العميل
0001	خليل محمود
0002	عادل سالم

2) الترميز المتسلسل بالتقسيم إلى كتل:

يعتمد هذا النوع على مبدأ تقسيم المفردات إلى مجموعات تسمى كل منها كتلة، حيث يتم تخصيص أرقام متسلسلة لكل كتلة، كأن نعطي الإدارة العليا من 001 إلى 025، الوسطى من 026 إلى 250، والتنفيذية من 251 إلى 999.

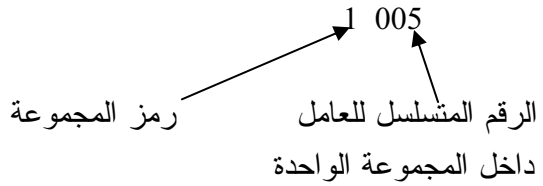
وتمتاز هذه الطريقة بأن الترميز ذو معنى، وبالمعلومات ذات المعنى التي يمكن استنباطها من الرمز.

3) الترميز التصنيفي بالمجموعات:

يعتمد هذا النوع على مبدأ تصنيف المقررات إلى مجموعات رئيسة قد تتفرع عنها مجموعات وسيطة قد يتفرع عنها أيضا مجموعات صغرى ....

مثال (5)

يصنف العاملون في الجامعة إلى أعضاء هيئة تدريسية وإداريين ولا يتجاوز العاملون في كل مجموعة 80 موظفاً. اقترح الترميز المناسب.

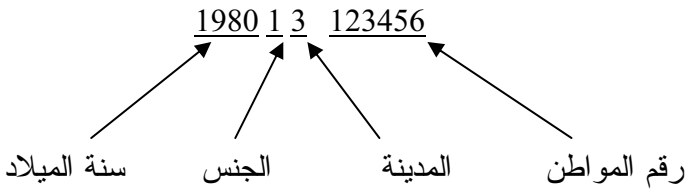


4) الترميز بالرقم المعنوي:

المقصود بالرقم المعنوي هو رقم ذو معنى يوضع بين خانات الرمز ليدل على صفة مادية يمكن قياسها مثل: الطول، والوزن،...وذلك بهدف توضيح المعنى.

مثال (6)

لترميز الرقم الوطني يمكن استخدام الترميز الآتي:



5) الترميز باستخدام رقم التحقق:

رقم التحقق هو عبارة عن خانة رقمية تضاف إلى الرقم وتوضع في الغالب إلى يمين أو يسار الرمز المكون للعنصر. يستخدم للتأكد من صحة المدخلات ذاتياً بعد إجراء

مجموعة من العمليات الحسابية عليه. وهناك خوارزميات مختلفة لاحتساب رقم التحقق.

مثال (7)

إذا كان رقم الطالب مكوناً من خمس خانات. وللتأكد من الرقم يضاف لليسار الرقم 1 إذا كان رقم الطالب زوجياً، ويضاف 2 إذا كان فردياً. فالرقم 38275 فردي فيصبح 238275، والرقم 78634 زوجي فيصبح 178634 . الآن لو تم إدخال الرقم 256438 فسنعرف فوراً بأن هناك خطأ ما في هذا الرقم.

أسئلة التقويم الذاتي (6)

اذكر أنواع الترميز، وبين ميزات كل منها.

5. الخلاصة

تعتبر مرحلة التحليل من أهم وأصعب مراحل بناء النظام. وتتكون مرحلة التحليل من خطوتين رئيستين: التحليل المبدئي: حيث يتم فيها التأكد من المشكلة، وتقدير تكاليف إيجاد الحل، والتحليل التفصيلي: حيث يتم فيها جمع متطلبات المستخدم للنظام الجديد، وبناء عليها يتم اقتراح حل للمشكلة الموجودة في النظام. ويتم جمع المتطلبات بعدة طرق أهمها: إجراء المقابلات، وتوزيع الاستبانات، ومراقبة النظام، وزيارة مؤسسات شبيهة، وزيارة شركات الحاسوب. ويتم في النهاية توثيق هذه المرحلة بكتابة التقرير التفصيلي، أو تقرير دراسة الجدوى.

وللتحليل أدوات عدة، حيث تستخدم لتوثيق التحليل بطريقة معيارية تيسر العمل لفرق التصميم، كما تساعد على فهم النظام وتيسر صيانته، وقد تم في هذه الوحدة شرح أربع أدوات واسعة الانتشار بين المحللين وهي قاموس البيانات، وجداول القرار، وأشجار القرار والترميز.

6. لمحة عن الوحدة الدراسية الرابعة

تعنى الوحدة الرابعة بتعريف حالات الاستخدام والمتفاعلين وبيان كيفية بنائها وتوثيقها. كما تعنى ببيان كيفية تحديد العلاقات التي تربط حالات الاستخدام بعضها ببعض من جهة، وبالمتفاعلين من جهة أخرى. حيث يقوم محلل النظام، خلال مرحلة التحليل، وبناء على المعلومات التي جمعها أثناء التحليل ببناء هذا المخطط ليبين كيفية تصرف النظام. ويندرج موضوع نمذجة حالات الاستخدام في إطار وصف تصرف النظام حيث يعتبر نموذج حالات الاستخدام توثيقاً لوظائف النظام المرجوة والمتفاعلين مع هذه الحالات والعلاقات بينهم.

وتنبع أهمية نموذج حالات الاستخدام من الدور الذي يقوم به لتسهيل الاتصال بين المستخدم ومحلل النظام حيث يوفر هذا النموذج الوسائط المناسبة لكل منهما لمناقشة وظائف النظام وتصرفه. ويتم تصميم نموذج حالات الاستخدام في المراحل الأولى من تصميم النظام. حيث تتم بداية تحديد حالات استخدام ومتفاعلين أساسيين ومن ثم في المراحل المتقدمة من النظام تتم إضافة وتعديل بعض حالات الاستخدام.

7. إجابات التدريبات

تدريب (1)

1. التحليل المبدئي/ طلب مستخدم

مذكرة داخلية

التاريخ: 2003/8/3

إلى: راضي السيد، مدير قسم الحاسوب

من: عصام محمد، مسؤول قسم المبيعات

الموضوع: الديون المستحقة

خلال الأشهر القليلة الماضية، راقبت نظام الديون المستحقة علينا ولاحظت أننا لا نستفيد من الخصومات المعروضة من قبل المزودين الذين نتعامل معهم. وتحدثت مع كاتب الحسابات وأكد لي بأن المشكلة ليست مشكلة سيولة نقدية. وأعتقد بأننا نستطيع توفير مبالغ كبيرة إذاً استفدنا من هذه الخصومات. وحساباتي تدل على أننا كنا نستطيع توفير مبلغ 1259$ فقط في الشهرين الماضيين.

أرجو التكرم بالإيعاز لمن يلزم للقيام بدراسة نظـــام الـــديون المـــستحقة علينـــا للاستفادة من هذه المبالغ الضائعة.

وتقبلوا فائق الاحترام،،،

هذا الجزء للاستخدام الرسمي

استلمت بتاريخ: 2003/8/4 عهدت إلى: المحلل جميل علي.

الإجراء المتخذ:

تمت الموافقة على الطلب. لتبدأ الدراسة في 8/10 وتنتهي في 8/20.

رقم الملف: 03/15 موقعة من: المحلل جميل علي

2. حالة دراسية/ التحليل المبدئي/ التفويض

مذكرة داخلية

التاريخ: 2003/8/5

إلى: السادة موظفي قسم المبيعات

من: مدير قسم المبيعات

الموضوع: تفويض بدراسة نظام الديون المستحقة.

نفوض السيد جميل علي بالقيام بدراسة نظام الديون المستحقة لتحديد مـــا يمكـــن القيام به للاستفادة من الخصومات المقدمة من المزودين عند دفع الفواتير.

نأمل منكم تقديم المساعدة الممكنة للسيد جميل. وبما انه ســـيقوم بالتحـــدث مـــع بعضكم خلال الفترة القادمة، أرجو إعطاءه الوقت الكافي لذلك.

وتقبلوا فائق الاحترام،،،

3. حالة دراسية/التحليل المبدئي/ الوصف الوظيفي

وصف وظيفي

الوظيفة: كاتب الحسابات

تاريخ إقرار الوظيفة: 1995/4/4

يخاطب: المدير المالي

المهام والمسؤوليات:

1. ترحيل الحسابات.
2. تجهيز قائمة المزودين.
3. تدريب الموظفين الجدد في الدائرة.
4. يرأس مكتباً فيه كاتبين.

الخبرات السابقة: 3- 4 سنوات في مؤسسة خاصة ككاتب حسابات.

المؤهلات: دبلوم كلية جامعية متوسطة دراسة سنتين على الأقل.

الراتب: راتب مستوى 4.

4. حالة دراسية/ شركة فلسطين/التحليل المبدئي

مذكرة داخلية

التاريخ: 2003/8/6

إلى: السيد سليم الأحمد / كاتب الحسابات.

من: المحلل جميل علي.

الموضوع: دراسة نظام الديون المستحقة.

هذا لتأكيد موعد المقابلة التي سأجريها معك في 8/10 الساعة الثالثة مساءً في قاعة اجتماعاتكم. وكما تعلم أقوم بدراسة نظام الديون المستحقة لتحديد ما يمكن القيام به للاستفادة من الخصومات المقدمة من المزودين عند دفع الفواتير. وسأقوم بتغطية المواضيع التالية في هذه المقابلة:

1. بالمعدل، كم فاتورة تقوم بدفعها شهرياً؟
2. هل يوجد فترات محددة أثناءها يحصل نقص في السيولة النقدية للوفاء بالإلتزامات؟
3. كم تستغرق الفاتورة من المخزن حتى تصل إليك؟
4. في حالة استلامها، كم تستغرق من الوقت للإنتهاء من معالجتها؟
5. هل تستلم الفواتير ناقصة؟ رجاءً أعط بعض الأمثلة.
6. هل تعاني من أية مشاكل أخرى في نظام الديون المستحقة؟

إذا احتجت إلى معلومات إضافية، رجاءً لا تتردد في الاتصال بي.

مذكرة داخلية

التاريخ: 2003/8/11

إلى: السيد سليم الأحمد / كاتب الحسابات

من: المحلل جميل علي

الموضوع: ملخص لمقابلة 2003/8/10

أقدر لك الوقت والجهد اللذين بذلتهما بالأمس، حيث أعلم بأنك مشغول جداً. وقد كانت معلوماتك قيمة فاستفدت منها كثيراً في فهمي للنظام. ولقد قمت بتلخيص المقابلة حسب ما فهمته منك كما يلي:

1. عدد الفواتير التي نقوم بدفعها شهرياً 85 فاتورة تقريباً بمعدل قيمته 520$.
2. يحصل عادة نقص في السيولة النقدية مقداره (3465$) في أول أسبوع من الشهر لدفع أجرة المبنى والرواتب.
3. تستغرق الفاتورة 6 أيام من المخزن حتى تصل إليك.
4. في حالة استلام الفاتورة، تستغرق 7 أيام للانتهاء من معالجتها.
5. تحتوي الفواتير على كل المعلومات اللازمة عند استلامها، ولكن تستلمها أحياناً ممزقة نتيجة الإهمال أثناء حملها.
6. لا تعاني من مشاكل أخرى في نظام الديون المستحقة.

وإذا احتجت إلى إضافة أو تغيير المعلومات الواردة، رجاء لا تتردد في الاتصال بي. وشكراً لك مرة أخرى.

5. حالة دراسية / التحليل المبدئي / التقرير المبدئي

مذكرة داخلية

التاريخ: 2003/8/13

إلى: راضي السيد، مدير قسم الحاسوب.

من: المحلل جميل علي.

الموضوع: التقرير المبدئي على نظام الديون المستحقة.

ملخص المشكلة:

لقد انتهيت من الدراسة الأولية لنظام الديون المستحقة لتحديد ما يمكن القيام به للاستفادة من الخصومات المقدمة من المزودين عند دفع الفواتير. لقد قابلت الأشخاص

الأربعة الرئيسين في النظام، وقد قمت بتوزيع ملخصات المقابلات على من يهمه الأمـــر. وقد قمت بمقابلة اثنين من مزودينا الكبار: شركة رام الله، ومؤسسة العودة للتأكد من نظام الخصومات لديهم.

النتائج:

من خلال المقابلات والبحث الذي أجريناه، اتضح لنا ما يلي:

1. معدل ما تستغرقه الفاتورة 16 يوماً حتى يتم دفعها.
2. معدل عدد الفواتير التي نقوم بدفعها شهرياً 85 فاتورة.
3. معدل قيمة الفاتورة 520$.
4. النظام يعمل بشكل جيد وبدون مشاكل، وسمعتنا عند المزودين جيدة.
5. حجم العمل على قسم كتابة الحسابات كبير ولم يتم تزويدهم بمعدات جديـــدة منـــذ 4 سنوات.
6. معظم المزودين يعرض 2% خصماً على الفواتير إذا تم دفعها خلال 10 أيام، وعلى الأقل 2 منهم يعرضان 3% خصماً على الفواتير إذا تم دفعها خلال 5 أيام.

التوصيات:

اعتماداً على ما استخلصناه أعلاه، نقترح ما يلي:

1. البدء في تحليل تفصيلي حالاً لتحديد جدوى حوسبة نظام الديون المـــستحقة اليـــدوي خاصة وإنه اعتماداً على 85 فاتورة شهرياً بمعدل 520$ للفاتورة وخصم2% خلال أول 10 أيام، فيكون ما قد يتم توفيره 884$ في الشهر أو 10,608$ في السنة.
2. اعطاء السيد كاتب الحسابات صلاحية توقيع الشيكات (بعد اخذ الضمانات الكافية) مما يؤدي الى توفير 3 ايام من دفع الفاتورة.
3. تقديم شكر خاص لقسم كتابة الحسابات لما يقومون به من جهد كبير.

الوقت والتكاليف المتوقعة:

موظفو قسم الحاسوب جاهزون ومستعدون للبدء بالدراسة التفصيلية خلال 4 أيام من موافقة الإدارة على ذلك. المحلل جميل علي مستعد لإكمال التحليل ويتوقع أن يستغرق ذلك 5 أسابيع. والتكاليف المتوقعة للتحليل التفصيلي ممكن أن تكون كما يلي:

1. راتب مقطوع للمحلل لـــ 5 أشهر	3192$
2. سكرتاريا	210$
المجموع	3402$

6. حالة دراسية / التحليل التفصيلي/التفويض والإعلام

مذكرة داخلية

التاريخ: 2003/8/21

إلى: كل الموظفين ومساعدي المدير العام

من: رؤوف محسن/ المدير العام

الموضوع: التحليل التفصيلي لنظام الديون المستحقة.

بالأمس قررنا الاستمرار بالتحليل المبدئي وعمل التحليل التفصيلي. المحلل جميل علي سيقوم بهذه الدراسة ومن المتوقع أن يبدأ هذه المهمة أول الأسبوع القادم.

ويرغب المحلل بالتحدث مع العديد منكم مرة أخرى وأتوقع منكم، كما تعودت دائماً، تقديم المساعدة الممكنة له. وخلال هذه المهمة سيقوم جميل بجمع النماذج الفارغة، وإجراء مقابلات، ونقاشات عامة، ومراقبة النظام.

وقد أظهر التحليل المبدئي أن عبْء العمل كبير على موظفي قسم كتابة الحسابات، ونتوقع بأن يخف هذا العبء عند تحويل النظام إلى نظام محوسب. وأظهرت الدراسة كذلك أن كاتب الحسابات يجب أن يخول بالتوقيع على الشيكات، لذا سيظهر توقيعه على كل الشيكات من الآن فصاعداً.

7. حالة دراسية / التحليل التفصيلي/الاستبانة

نقوم الآن بعملية تقييم نظام الديون المستحقة لدينا من أجل تسريع عملية دفع الفواتير لمزودينا الكرام. والرجاء تعبئة النموذج المرفق وإعادته في المغلف المرفق.

1. اسم مؤسستكم:
2. اسم الشخص الذي عبأ هذه الاستبانة:
3. وظيفته:

أ) مدير ب) كاتب حسابات ج) مراقب د) مندوب مبيعات هـ) غيرها:

4. هل لاحظت أي تأخير في السداد من قبل شركة فلسطين؟ إذا كانت الإجابة بنعم، متى لاحظت ذلك وكم كان مقدار التأخير؟
5. عندما تتصل بشركة فلسطين، هل تكون المعاملة سيئة عادة؟

نعم 1 2 3 4 5 لا

6. بالمقارنة مع زبائنك الآخرين، تعتبر شركة فلسطين:

أ) نفس العملاء الآخرين ج) أسوأ من معظمهم

ب) أفضل من معظمهم د) لا أعرف

7. هل من الممكن أن تقدموا لنا خصماً خاصاً إذا قمنا بالسداد خلال 5 أيام؟ إذا كانت الإجابة بنعم، فما هي شروطكم؟
8. مع من تتصل عادة في شركتنا؟
9. هل لديك أي ملاحظات أو اقتراحات أخرى؟

8. حالة دراسية / التحليل التفصيلي/دراسة الجدوى

مذكرة داخلية

التاريخ: 2003/8/29

إلى: رؤوف محسن/ المدير العام

من: المحلل جميل علي

الموضوع: التقرير النهائي للتحليل التفصيلي لنظام الديون المستحقة

المشكلة:

لقد انتهينا من التحليل التفصيلي لنظام الديون المستحقة. ويلخص هذا التقرير الأهداف، ونتائج البحث، والتوصيات. وبالمختصر، نوصي ببناء نظام جديد للديون المستحقة يحقق في النهاية أكثر من $16,000 توفيراً في السنة.

أهداف النظام:

1. تسريع عملية تسديد الفواتير إلى 5 أيام.
2. تقليل عبء العمل عن موظفي قسم كتابة الحسابات.

نتائج البحث:

1. عملية معالجة الفواتير انخفضت بمقدار يومين حيث أن السيد كاتب الحسابات أصبح مفوضاً بالتوقيع على الشيكات، ولكن للأسف هذا الانخفاض ليس كافياً للاستفادة من الخصم.
2. الخسارة من عدم الاستفادة من الخصم كبيرة ($700 في الشهر).
3. النظام اليدوي الحالي بطيء.
4. يعتمد كاتب الحسابات على ذاكرته في معرفة الفاتورة التي ستسدد.
5. تكاليف النظام الحالي عالية، وسنحتاج إلى موظفين أكثر.
6. النظام الحالي فاشل في تنظيم المزودين، مما يؤدي إلى إرسال الشيكات إلى المزود الخطأ، مما يؤدي إلى دفع غرامات تأخير.

الحلول المقترحة:

تم اقتراح 3 حلول لمشاكل النظام الحالي، كل حل منها يسهم بحل المشاكل الموجودة بالنظام الحالي بطريقة ما:

الخيار الأول:

ابقِ على النظام الحالي مع توظيف موظفين جديدين، واحد الآن والآخر خلال السنة القادمة.

التكاليف: الخسارة من عدم الحصول على الخصم	$12000/السنة
الموظف الأول	$12,600
الموظف الثاني	$33,600

التوفير: لا شيء

التأثير: سيتم المحافظة على ثبات النظام الحالي ولكن بعد بضعة سنوات ستعود المشكلة كما كانت

الخيار الثاني:

اشتر نظاماً جاهزاً. والنظام التجاري الشامل موجود ويلبي احتياجاتنا. هذا النظام يعمل فعلياً في أكثر من مؤسسة.

التكاليف: ثمن النظام التجاري الشامل	$7,500
التعديلات اللازمة على النظام	$2,600
تكاليف التدريب	$1,200
التوفير: الخصومات	$12,000/سنة
بدل عمل إضافي	$4,200/سنة

التأثير: سيتم التسديد خلال 5 إلى 10 أيام. وستتم بعض التعديلات التي تلزم في المكاتب والمخزن. وهذا النظام سيكون جاهزاً للعمل خلال 3 أشهر. ويحتوي على توثيق جيد.

الخيار الثالث:

ابن نظاماً خاصاً بمؤسستنا. هذا الخيار يعطينا المرونة القصوى بحيث نستطيع بناء ما نريد وبالطريقة التي نرغب.

التكاليف: التحليل والتصميم والبرمجة	$13,400
تكاليف التدريب	$1,200
الصيانة السنوية	$900/سنة
التوفير: الخصومات	$12,000/سنة

بدل عمل إضافي $4,200/سنة

التأثير: سيتم التسديد خلال 5 إلى 10 أيام. وسيتم بعض التعديلات التي تلزم في المكاتب والمخزن. وهذا النظام سيكون جاهزاً للعمل خلال 6 أشهر.

التوصيات:

أوصي بتبني الخيار الثالث للأسباب التالية:

1. تكاليف بناء نظام للمؤسسة ليست أكثر بكثير من تكاليف نظام جاهز ($15,500 – $11,300)
2. سيحل كل المشاكل الموجودة في النظام الحالي.
3. النظام الجديد سيكون متطابقاً كلياً مع بقية الأنظمة في المؤسسة.
4. سنسترجع تكاليف النظام خلال سنتين.
5. تكاليف النظام الجاهز تكون عادة أكثر من التكاليف المعروضة.

الفوائد الأخرى:

الخيار الثالث يعطي المؤسسة الفوائد التالية أيضاً:

1. خدمة أفضل لزبائننا.
2. سيعطي التقارير اللازمة للإدارة بالشكل المطلوب وعلى أساس دوري.
3. ستحافظ المؤسسة على موقعها بين المنافسين.
4. سيشعر الموظفين براحة أكبر مما يؤدي إلى إنتاجية أكبر
5. سيؤدي إلى استغلال أكبر للأجهزة والحواسيب الموجودة في المؤسسة.

مذكرة داخلية

التاريخ: 2003/8/30

إلى: كل الموظفين ومساعدي المدير العام

من: رؤوف محسن/ المدير العام

الموضوع: القرار حول النظام الجديد.

كما تعلمون، فإن نّظام تسديد الديون المستحقة الخاص بالمؤسسة خضع لدراسة شاملة خلال الفترة السابقة. وبالأمس، أظهرت الدراسة أن النظام الحالي بحاجة إلى تغيير. فقررنا بأن الأفضل لنا على المدى البعيد بأن نقوم ببناء نظام جديد خاص بنا ويعمل على الحاسوب الموجود. وعند أخذ هذا القرار، أخذنا بعين الاعتبار: السرعة، وعبء العمل الزائد على الموظفين، والرغبة في أن نبقى في مقدمة المؤسسات المنافسة. والعمل على بناء النظام الجديد سيبدأ فوراً كما هو متوقع وسيكون جاهزاً في الربيع القادم.

8. مسرد المصطلحات

- **أشجار القرار Decision Trees:** إن شجرة القرارات هي تمثيل تخطيطي لعملية اتخاذ القرار وتعتبر بديلاً عن جدول القرارات، حيث تمتاز بالبساطة والوضوح مقارنة بجدول القرارات.
- **التحليل المبدئي Preliminary Analysis:** يعتبر التحليل المبدئي أول خطوات مرحلة التحليل، ويتم فيه استلام طلب التحليل من أحد المعنيين بالنظام وعادة ما يكون أحد المستخدمين، ثم يقوم مدير المعلومات بتقييم هذا الطلب للتأكد من أنه جاد وأنه يحتوي على مشكلة تستحق العناء. فإذا كان كذلك، يعطيه لأحد المحللين للتأكد من المشكلة على أرض الواقع، ويكون ذلك عادة بإجراء مقابلات مع المستخدم الذي رفع طلب التحليل، ثم يقوم بكتابة تقرير بما جمع من معلومات عن المشكلة يسمى بالتقرير المبدئي ويسلمه لمديره.
- **الترميز Coding:** يقصد بالترميز تمثيل البيانات الكبيرة برموز صغيرة بحيث يسهل التعامل مع هذه البيانات ذات الحجم الكبير.
- **جداول القرار Decision Tables:** أداة من أدوات تحليل الأنظمة تساعد محلل الأنظمة على تعريف المشكلة ودراسة أبعادها والتعرف على الحلول البديلة المتاحة ومن ثم الوصول إلى الحل الأمثل، كما أنها تعتبر أداة تساعد محلل الأنظمة على فهم وتحليل النظام القائم والنظام المقترح وإيضاح المنطق من وراء صناعة كل قرار.
- **طلب التحليل User Request:** طلب مكتوب يقدمه أحد المستخدمين إلى مديره عند ملاحظة خلل ما في النظام وذلك لتصحيح هذا الخلل والذي يقوم بدوره بتسليمه إلى الشخص المعني والذي يكون عادة مدير دائرة المعلومات. ويسمى هذا الطلب طلب التحليل أو طلب المستخدم (User Request) وبعضهم الآخر يسميه طلب النظام (System Request).
- **قاموس البيانات Data Dictionary:** أداة من أدوات تحليل الأنظمة تستخدم لوصف البيانات الداخلة للنظام من خلال: الاسم، والتصنيف، والشكل، والاستخدام،... ويعتبر خطوة أولية فعّالة لبناء قاعدة البيانات، كما أنه يحسَن الاتصال والتفاهم بين المحللين والمستخدمين، ويساعد في عمليات صيانة النظام.
- **متطلبات المستخدمين User Requirements:** رغبات المستخدمين ووصفهم لما سيحويه النظام الجديد من وظائف وصفات.
- **النمذجة Modeling:** عملية يقوم بها المحللون ومصممو النظام لوصفه من جميع جوانبه بطريقة رسمية باستخدام إحدى طرق أدوات النمذجة المعيارية.

9. المراجع

1. Aktas, Z., Structured Analysis and Design of Information Systems, Prentice-Hall, 1987.
2. Baudoin, C. and Hollowell, G., Realizing The Object-Oriented Life Cycle, Prentice- Hall, 1996.
3. Booch, G., Rumbaugh, J., and Jacobson, I., The Unified Modeling Language User Guide, Addison Wesley, 1999.
4. Dennis, A. et al, Systems Analysis & Design: An Object-Oriented Approach with UML. John Wiley & Sons Inc., 2002.
5. Dewitz, S., Systems Analysis and Design and the Transition to Objects, McGraw-Hill, 1996.
6. Edwards, P., Systems Analysis, Design, And Development With Structured Concepts, Rinehart and Winston, 1999.
7. Henderson, P., Object-Oriented Specification and Design with C++, McGraw- Hill, 1993.
8. Hoffer, J. et al., Modern Systems Analysis & Design, 2nd Ed., Addison-Wesley, 1999.
9. Kendal, K., and Kendal, J., Systems Analysis and Design, 5th Ed. 2002.
10. Kruchten, P., The Rational Unified Process: An Introduction. 2nd Ed., Addison- Wesley, 2000.
11. Lejk, M. and Deeks. An Introduction to Systems Analysis Techniques, 2nd Ed., Pearson Education Limited, 2002.
12. Maciaszek, A., Requirements Analysis and System Design, Developing Information Systems with UML, Addison- Wesley, 2001.
13. Quatrani, T., Visual Modeling with Rational Rose 2000 and UML, Addison- Wesley, 2000.
14. Satzinger, J., Systems Analysis and Design, 2nd Ed., Thomson Learing, 2002.
15. Silver, G. and Silver, M., Systems Analysis and Design, Addison-Wesley, 1996.
16. http://www.psut.edu.jo / computer science/ staff/saleh abu-soud
17. www.rational.com
18. www.omg.com

الوحدة الرابعة

نمذجة حالات الاستخدام

Use–Cases Modeling

محتويات الوحدة

الموضوع	الصفحة
1. المقدمـــة	81
1.1 تمهيد	81
2.1 أهداف الوحدة	81
3.1 أقسام الوحدة	82
4.1 القراءات المساعدة	82
5.1 ما تحتاج إليه لدراسة الوحدة	82
2. المتفاعلون Actors	83
1.2 بناء المتفاعلين Development of Actors	83
2.2 توثيق المتفاعلين Documentation of Actors	84
3. حالات الاستخدام Use- Cases	87
1.3 أنواع حالات الاستخدام Use- Cases Types	87
2.3 بناء حالات الاستخدام Use- Cases Development	88
3.3 توثيق حالات الاستخدام Use- Cases Documentation	90
4.3 علاقات حالات الاستخدام Use- Cases Relationships	93
4. مخططات حالات الاستخدام Use-Cases Charts	98
5. الخلاصة.....	100
6. لمحة عن الوحدة الدراسية الخامسة.....	100
7. إجابات التدريبات.....	101
8. مسرد المصطلحات.....	104
9. المراجع.....	105

1. المقدمة

1.1 تمهيد

أهلاً بك، عزيزي الدارس، إلى الوحدة الرابعة من مقرر "1380 تحليل الأنظمة وتصميمها"، وهي بعنوان " نمذجة حالات الاستخدام".

تعرف حالات الاستخدام بأنها وسيلة اتصال جيدة بين المستخدم ومطور النظام حيث توفر الوسائط المناسبة لكل منهما لمناقشة وظائف النظام وتصرفاته، بينما يعرف المتفاعلون بأنهم أطراف خارجية تتعامل مع النظام بمقتضى الحاجة والوظائف المتوفرة في النظام.

ويندرج موضوع نمذجة حالات الاستخدام في إطار وصف تصرف النظام حيث يعتبر نموذج حالات الاستخدام توثيقاً لوظائف النظام المرجوة والمتفاعلين مع هذه الحالات والعلاقات بينهم.

وتنبع أهمية نموذج حالات الاستخدام من الدور الذي يقوم به لتسهيل الاتصال بين المستخدم ومحلل النظام حيث يوفر هذا النموذج الوسائط المناسبة لكل منهما لمناقشة وظائف النظام وتصرفه. ويتم تصميم نموذج حالات الاستخدام في المراحل الأولى من تصميم النظام. حيث تتم بداية تحديد حالات استخدام ومتفاعلين أساسيين ومن ثم في المراحل المتقدمة من النظام تتم إضافة وتعديل بعض حالات الاستخدام.

2.1 أهداف الوحدة

يتوقع منك–عزيزي الدارس– بعد الانتهاء من دراسة هذه الوحدة أن تكون قادراً على أن:

1. تبين أهمية المتفاعلين.
2. تتمكن من بناء وتصميم المتفاعلين.
3. توضح أهمية حالات الاستخدام.
4. تحدد عناصر وكيفية بناء حالات الاستخدام.
5. ترسم مخططات حالات الاستخدام.

3.1 أقسام الوحدة

تقسم هذه الوحدة إلى ثلاثة أقسام رئيسة:

القسم الأول: يناقش المتفاعلين وأهميتهم وكيفية بنائهم وبذلك يغطي الهدفين الأول والثاني.

أما القسم الثاني فيبين حالات الاستخدام وأنواعها، وأهمية كل نوع، ويغطي الهدفين الثالث والرابع. وأمّا القسم الأخير فيغطي عملية بناء نموذج حالات الاستخدام، وهذا يحقق الهدف الخامس والأخير.

4.1 القراءات المساعدة

أنصحك، عزيزي الدارس، بالاطلاع على المراجع التالية لما تحويه من معلومات مفيدة تتعلق بموضوع هذه الوحدة:

1. Baudoin, C. and Hollowell, G., Realizing The Object-Oriented Life Cycle, Prentice -Hall, 1996.
2. Booch, G., Rumbaugh, J., and Jacobson, I., The Unified Modeling Language User Guide, Addison- Wesley, 1999.
3. Dennis, A. et al, Systems Analysis & Design: An Object-Oriented Approach with UML. John Wiley & Sons Inc., 2002.
4. Dewitz, S., Systems Analysis and Design and the Transition to Objects, McGraw-Hill, 1996.
5. Henderson, P., Object-Oriented Specification and Design with C++, McGraw- Hill, 1993.

5.1 ما تحتاج إليه لدراسة الوحدة

عزيزي الدارس: قبل أن تبدأ بدراسة هذه الوحدة تأكد من أنك قمت بتهيئة المكان الهادئ المناسب للدراسة مما يساعدك على التركيز والتفكير العميق. ثم احرص على تتبع الإرشادات التي توجهك لدراسة الوحدة، وحاول بجدية الإجابة عن أسئلة التقويم الذاتي، وحل التدريبات، لأنها تساعدك على مراجعة موضوعات الوحدة، وتعمق معلوماتك حولها.

2. المتفاعلون Actors

لا يعتبر المتفاعلون جزءاً من النظام بل هم أطراف خارجية تتعامل مع النظام بمقتضى الحاجة والوظائف المتوفرة في النظام. ويمكن أن يكون المتفاعل شخصاً أو نظاماً آخر أو آلة كالطابعة مثلاً. وقد يكون المتفاعل عبارة عن مصدر من مصادر معلومات النظام، أو طرف يتلقى مخرجات من النظام، أو قد يكون طرف إدخال وإخراج في نفس الوقت.

ونستطيع عادة استنباط المتفاعلين من وصف المشكلة أو بمحادثة الزبائن المتخصصين في مجال النظام. ويمكن ذلك بالإجابة على أسئلة شبيهة بالأسئلة التالية:

1- من هو المهتم في إحدى متطلبات النظام ؟
2- في أي مكان في المؤسسة سيستخدم النظام؟
3- من سينتفع من استخدام النظام ؟
4- من الذي سيزود النظام بالمعلومات ويستخدمها ويعدلها ويقوم بمسحها؟
5- من الذي سيدعم صيانة النظام؟
6- هل يستخدم النظام مصادر خارجية؟
7- هل يوجد شخص واحد يقوم بعدة أدوار في النظام ؟
8- هل هناك عدة أشخاص يقومون بدور واحد في النظام؟
9- هل يقوم النظام بالتواصل مع النظام القديم؟

والجدير بالذكر هنا- عزيزي الدارس- أن هذه الأسئلة ليست محصورة بهذه المجموعة بل يمكن إضافة أسئلة أخرى حسب متطلبات النظام.

1.2 بناء المتفاعلين Development of Actors

يجب على المصمم أن يوفر الكثير من العناية عند تحديد المتفاعلين. وتتم عملية تحديد المتفاعلين بطريقة تفاعلية تكرارية حيث يتم حصر المتفاعل من مجموعة مبدئية مقترحة وكبيرة من المتفاعلين، وتستمر هذه العملية حتى يتم التوصل إلى العدد المناسب من المتفاعلين. وعلى سبيل المثال هل يوجد فرق بين عامل جديد وعامل ذي خبرة في نظام شؤون الموظفين من ناحية القدوم إلى العمل والمغادرة منه؟ مبدئياً نستطيع القول أنهما مختلفان لكن بعد دراسة وظيفة النظام بشكل أعمق نجد أنهما يتعاملان مع النظام بنفس الطريقة حيث لا يوجد فرق بينهما، وبذلك يفضل في هذه الحالة اعتبارهما متفاعلاً واحداً. لكن في حال وجود فرق كبير في استخدامهما للنظام نقوم بفصلهما إلى متفاعلين اثنين.

أسئلة التقويم الذاتي (1)

بين كيف أن طريقة إيجاد المتفاعلين تكون طريقه تفاعلية؟

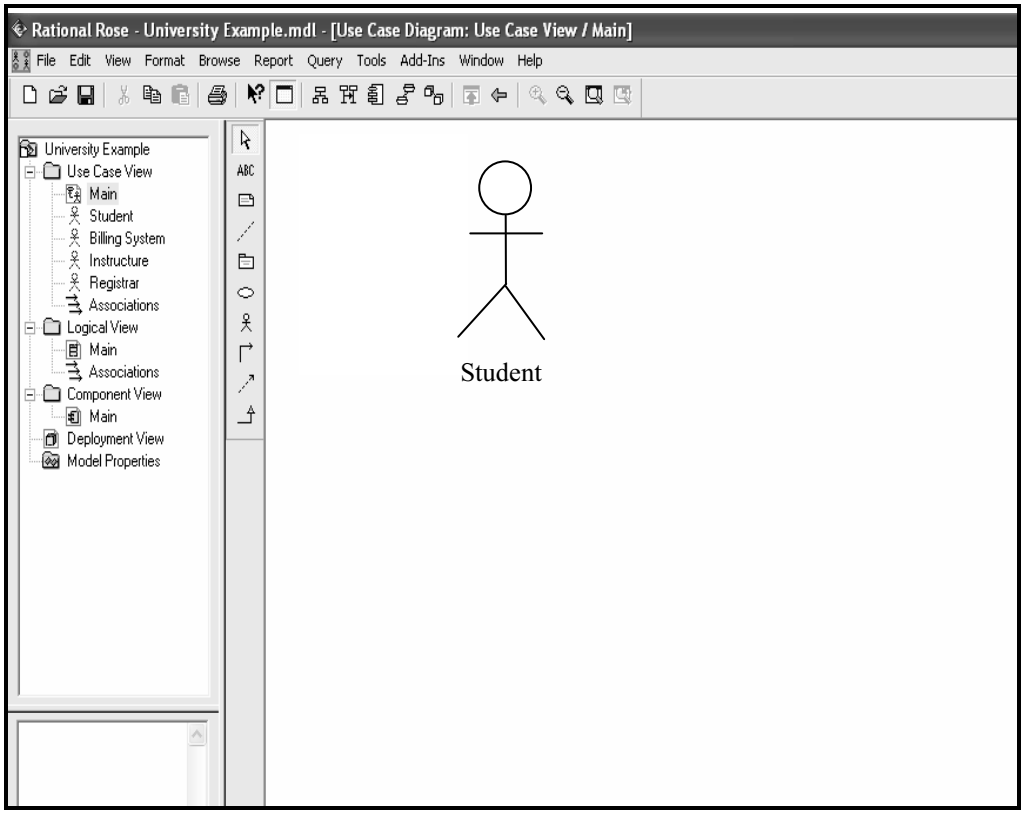
2.2 توثيق المتفاعلين Documentation of Actors

يقوم محلل النظام بتوثيق كل متفاعل في النظام ويضيف شرحاً مختصراً لكل منهم حيث يضاف إلى نموذج حالات الاستخدام الذي سيتم شرحه لاحقاً في هذه الوحدة. ويتم توثيق المتفاعلين باستخدام الرمز الموضح في الشكل (1) حيث يرمز له بشكل شخص وتتم كتابة اسم المتفاعل أسفل هذا الرمز.

الشكل (1): الرمز المستخدم لتوثيق المتفاعل

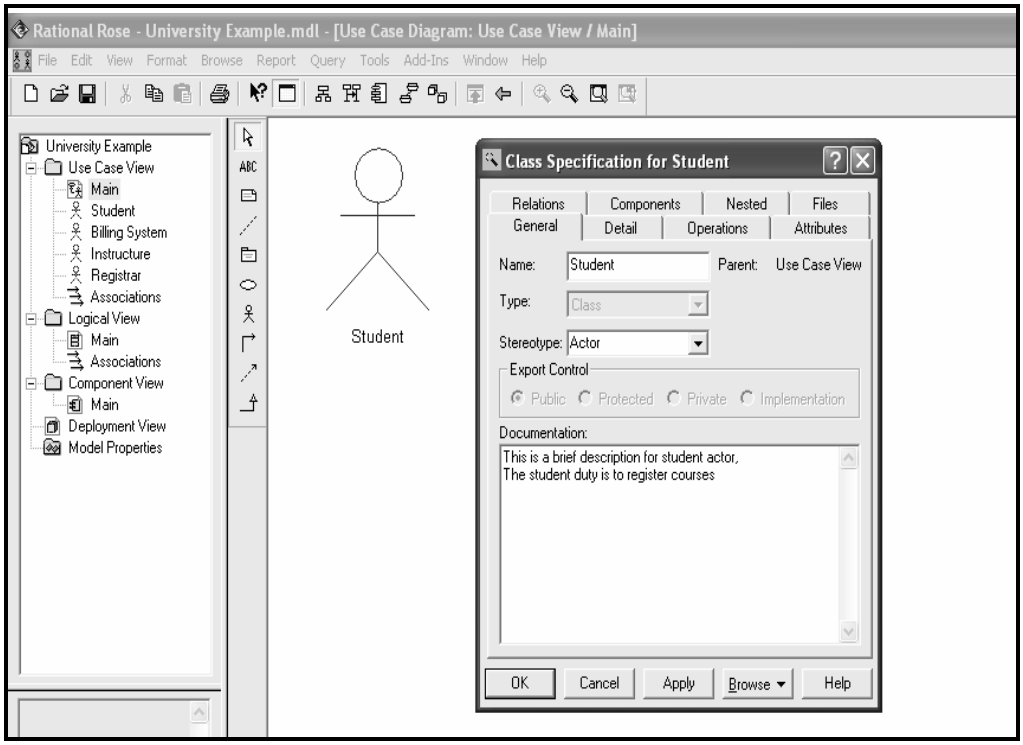
وسنقوم خلال هذه الوحدة والوحدات اللاحقة- عزيزي الدارس- بإتباع الخطوات اللازمة لبناء نظام تسجيل جامعي واستخدامه كحالة دراسية لبيان المفاهيم الواردة خلال هذا المقرر. وسيتم استخدام برمجية Rational لهندسة البرمجيات كأداة لتطبيق هذه الحالة.

يعتبر الطالب من أهم المتفاعلين في نظام التسجيل. ولتوثيق "المتفاعل" الطالب نستعمل الرمز المبين في الشكل (1) مع كتابة (Student) كاسم لهذا المتفاعل كما هو موضح في الشكل (2).



الشكل (2) إضافة المتفاعل الطالب

ويفضل عادة إضافة وصف مختصر لكل متفاعل، حيث يبين هذا الوصف وظيفة المتفاعل ويوضح طريقه تفاعله مع النظام كما هو موضح بالشكل (3).



الشكل (3) إضافة وصف مختصر للمتفاعل الطالب

وبنفس الطريقة تمكن إضافة بقية المتفاعلين في نظام التسجيل. وتجدر الإشـــارة هنا إلى أن مجموعة المتفاعلين المستخلصة في هذه المرحلـــة هـــي مجموعـــة تقريبيـــة للمتفاعلين حيث يمكن تعديل هذه المجموعة في خطوات لاحقة من مرحلة التحليل.

تدريب (1)

استخرج بقية المتفاعلين من نظام التسجيل الجامعي واتبع نفس الخطوات السابقة لرسمها.

3. حالات الاستخدام Use-Cases

تعتبر حالات الاستخدام وسيلة اتصال جيدة بين المستخدم ومطور النظام حيث توفر الوسائط المناسبة لكل منهما لمناقشة وظائف النظام وسلوكه.

وتمثل حالة الاستخدام الوظيفة التي يستطيع النظام توفيرها حيث تكون هذه الوظيفة عبارة عن قدرات النظام؛ أي أن وجود حالة الاستخدام يوفر طريقة فعّالة لوصف كل الطرق التي يمكن فيها استخدام النظام.

ويمكن تعريف حالة الاستخدام بأنها عبارة عن تتابع من العمليات يمكّن النظام من الحصول على نتائج قابلة للقياس لصالح متفاعل معين.

1.3 أنواع حالات الاستخدام Use-Cases Types

نستطيع القول بشكل عام- عزيزي الدارس- أن هناك العديد من أنواع حالات الاستخدام، لكن اعتاد محللو النظم على تصنيف حالات الاستخدام حسب بعدين حيث يكون أحد هذين البعدين هو تصنيف حالة الاستخدام بناءً على الهدف منها، وتسمى حالات الاستخدام في هذا البعد بحالات الاستخدام ذوات الشكل العام (General Use-Cases). والبعد الآخر يكون حسب إمكانية استخدام حالة الاستخدام من قبل أكبر قدر ممكن من حالات الاستخدام الأخرى، وتسمى حالات الاستخدام هنا بحالات الاستخدام ذوات الشكل الأساسي (Essential Use-Cases).

إن حالات الاستخدام ذوات الشكل العام تمكن محللي النظام ومستخدميه من التوصل إلى اتفاقية حول متطلبات النظام بشكل مبدئي، وعادة ما تنشأ حالات الاستخدام من هذا النوع في مرحلة متقدمة من تحليل متطلبات النظام، وتكون عبارة عن توثيق للمعلومات الأساسية لحالة الاستخدام مثل: الاسم، والرقم التسلسلي، والمتفاعل الرئيس، والنوع، والوصف المختصر. وبمجرد موافقة كلا الطرفين على التفاصيل العامة لكل حالات الاستخدام ذات الشكل العام، يمكن إضافة العديد من التفاصيل الأخرى إليها، مثل اسم المستخدم الذي قام باقتراحها وتاريخ آخر تعديل وهكذا...

أما حالات الاستخدام ذات الشكل الأساسي فهي عبارة عن حالات استخدام تحتوي على وظيفة واحدة محددة يمكن استخدامها من قبل حالات الاستخدام الأخرى. مثل حالة الاستخدام التي تمثل الولوج إلى النظام (Login) حيث يمكن استدعاؤها من قبل العديد من حالات الاستخدام الأخرى، ويمكن استخدامها من قبل عدة متفاعلين مثل الطالب، والمسجل، والمدرس. لذا تعتبر حالة الاستخدام (Login) حالة استخدام أساسية مجردة.

أسئلة التقويم الذاتي (2)

عدد أنواع حالات الاستخدام.

2.3 بناء حالات الاستخدام Use- Cases Development

لبناء حالات الاستخدام قد يحتاج محلل النظام لطرح التساؤلات التالية بينه وبين نفسه وإيجاد إجابات لها بالتعاون مع المستخدمين:

1- ما هي واجبات كل متفاعل؟

2- هل يوجد متفاعلون يلزمهم تخزين، وحذف، وتعديل أو قراءة معلومات من النظام؟

3- ما هي حالات الاستخدام التي ستخزن وتغير وتحذف وتستدعي المعلومات آنفة الذكر؟

4- هل سيقوم المتفاعل بإجراء تغييرات طارئة عن خط سير النظام الطبيعي، كإعطاء صلاحيات أو سحب صلاحيات؟

5- أي من حالات الاستخدام سيقوم بصيانة ودعم النظام؟

6- هل ستغطي حالات الاستخدام كل الوظائف المطلوبة في النظام؟

هذه الأسئلة- عزيزي الدارس- ليست حصرية حيث يمكن للمحلل إضافة أية أسئلة أخرى يرى أنها تساعده على حصر حالات الاستخدام قدر الإمكان.

أسئلة التقويم الذاتي (3)

اذكر الأسئلة التي من الممكن للمحلل أن يضيفها لحصر حالات الاستخدام.

بعد تحديد حالات الاستخدام يجب على المحلل تقييم هذه الحالات واستثناء غير المفيد منها وغير الفعالة. وعلى سبيل المثال هناك مشكلة تتعلق بكمية التفاصيل المتعلقة لكل حالة استخدام، بمعنى هل هناك داع للكثير من التفاصيل أو من الأفضل اختصارها؟ والجدير بالذكر هنا أنه لا يمكن تحديد ما إذا كانت حالة الاستخدام مفيدة أم لا بشكل قطعي، وإنما الأمر نسبي، ويمكن لمحلل النظام الاهتداء بالقاعدتين التاليتين لهذه الغاية:

أولاً: حالة الاستخدام في داخلها عبارة عن خطوات متسلسلة لها بداية ونهاية.

ثانياً: يجب على كل حالة استخدام خدمة متفاعل واحد على الأقل من متفاعلي النظام.

وفي بعض الحالات ربما يلجأ المحلل إلى دمج عدة حالات استخدام بحالة واحدة

بناء على طبيعة الوظائف التي تقدمها هذه الحالات. وفي أحيان أخرى قد يلجأ المحلل إلى تجزئة حالة إلى حالات أخرى.

أسئلة التقويم الذاتي (4)

اذكر القاعدتين التي من الممكن استخدامها لمعرفة جودة حالات الاستخدام.

مثال (1)

سنقوم في هذا المثال- عزيزي الدارس- ببناء بعض حالات الاستخدام في نظام التسجيل التي لها علاقة بالمتفاعل "الطالب" وذلك عن طريق طرح الأسئلة والإجابة عليها، ومن ثم تحييد غير المفيد منها:

1- ما هي واجبات المتفاعل "الطالب"؟
   - تسجيل المواد.
   - الدراسة.
   - دفع رسوم التسجيل.
   - استخدام المكتبة.

2- هل سيقوم المتفاعل "الطالب" بتخزين، أو حذف، أو تعديل أو قراءة معلومات من النظام؟
   - نعم

3- إذا كانت إجابة السؤال الثاني بنعم، فما هي طبيعة عمليات التخزين، أو الحذف، أو التعديل أو القراءة ؟

العمليات هي:
   - تعديل المواد المختارة (Update_Selected_Courses)
   - الانسحاب من فصل (Withdraw_from_Semester)
   - تعديل معلومات الطالب (Update_Student_Info)

4- هل سيقوم المتفاعل "الطالب" بإعلام النظام بأي تغيير طارئ عن خط سير النظام الطبيعي كتوقيف تنفيذ النظام أو منح أو تغيير صلاحيات المستخدمين؟
   - نعم، حيث يمكن للطالب أن يعطي صلاحيات تسجيل مواده لطالب آخر في حالة سفره مثلاً.

الآن وبعد الإجابة عن بعض الأسئلة التي رأى المحلل بأنها ضرورية، يقوم المحلل بتحديد المجموعة الابتدائية من حالات الاستخدام التي يمكن للطالب القيام بها،

والتي يمكن إجمالها كالآتي:

1- تسجيل المواد.
2- الدراسة.
3- دفع رسوم التسجيل.
4- استخدام المكتبة.
5- تعديل المواد المختارة.
6- الانسحاب من الفصل الدراسي.
7- تعديل معلوماته.
8- إعطاء صلاحيات لطالب آخر للتسجيل نيابة عنه.

عند التمعن في الحالات السابقة، نلاحظ أن هناك بعض الحالات غير المفيدة في موضوع نظام التسجيل، لذا من الواجب تحييدها وعدم اعتبارها من ضمن النظام كالدراسة واستخدام المكتبة فهي من واجبات الطالب الأساسية ولكنها ليست مهمة لنظام التسجيل.

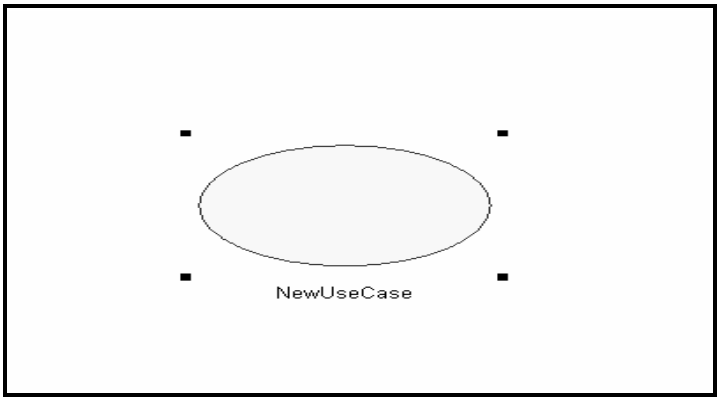
وعليه تصبح القائمة كما يلي:

1- تسجيل المواد.
2- دفع رسوم التسجيل.
3- تعديل المواد المختارة.
4- الانسحاب من فصل.
5- تسجيل الطالب.
6- تعديل ملف الطالب.

وتجدر الملاحظة أن كل الحالات المذكورة هي من نوع الشكل العام.

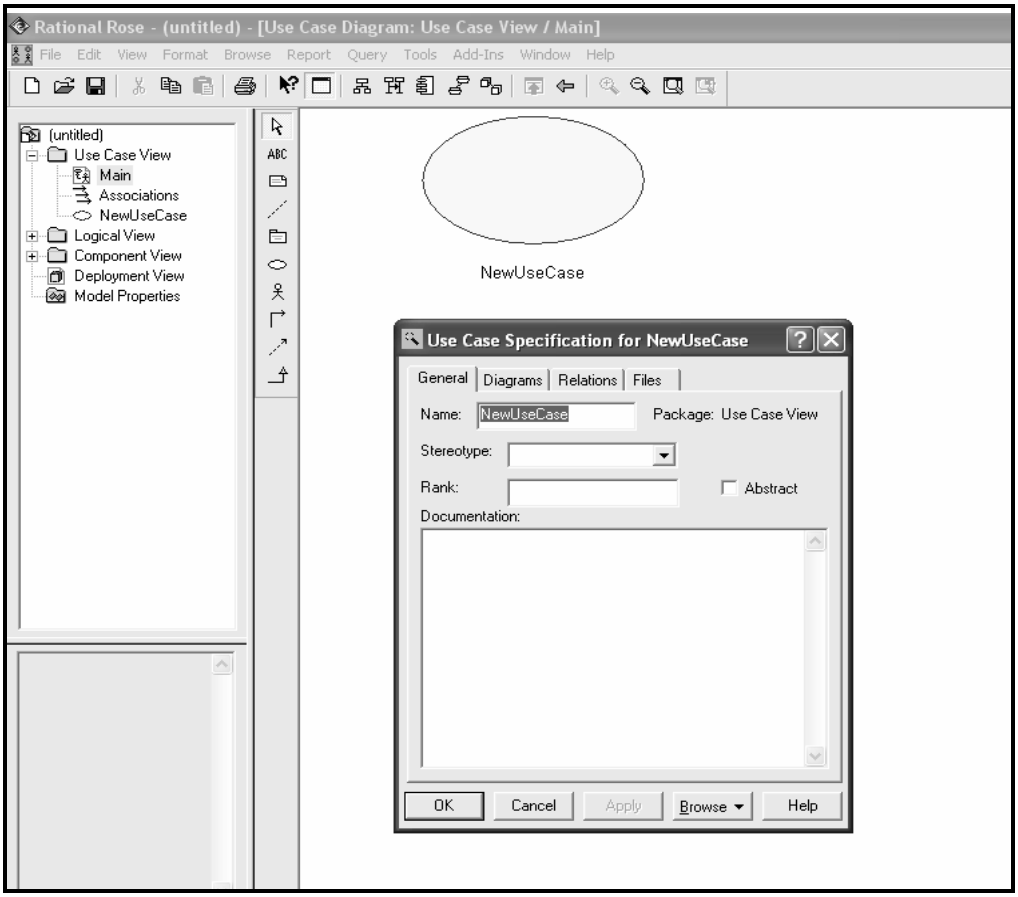
3.3 توثيق حالات الاستخدام Use-Cases Documentation

يستخدم عادة الشكل البيضاوي لتوثيق حالات الاستخدام حيث يتم ترميزها ويوضع اسمها أسفل الشكل كما هو موضح في الشكل (4).



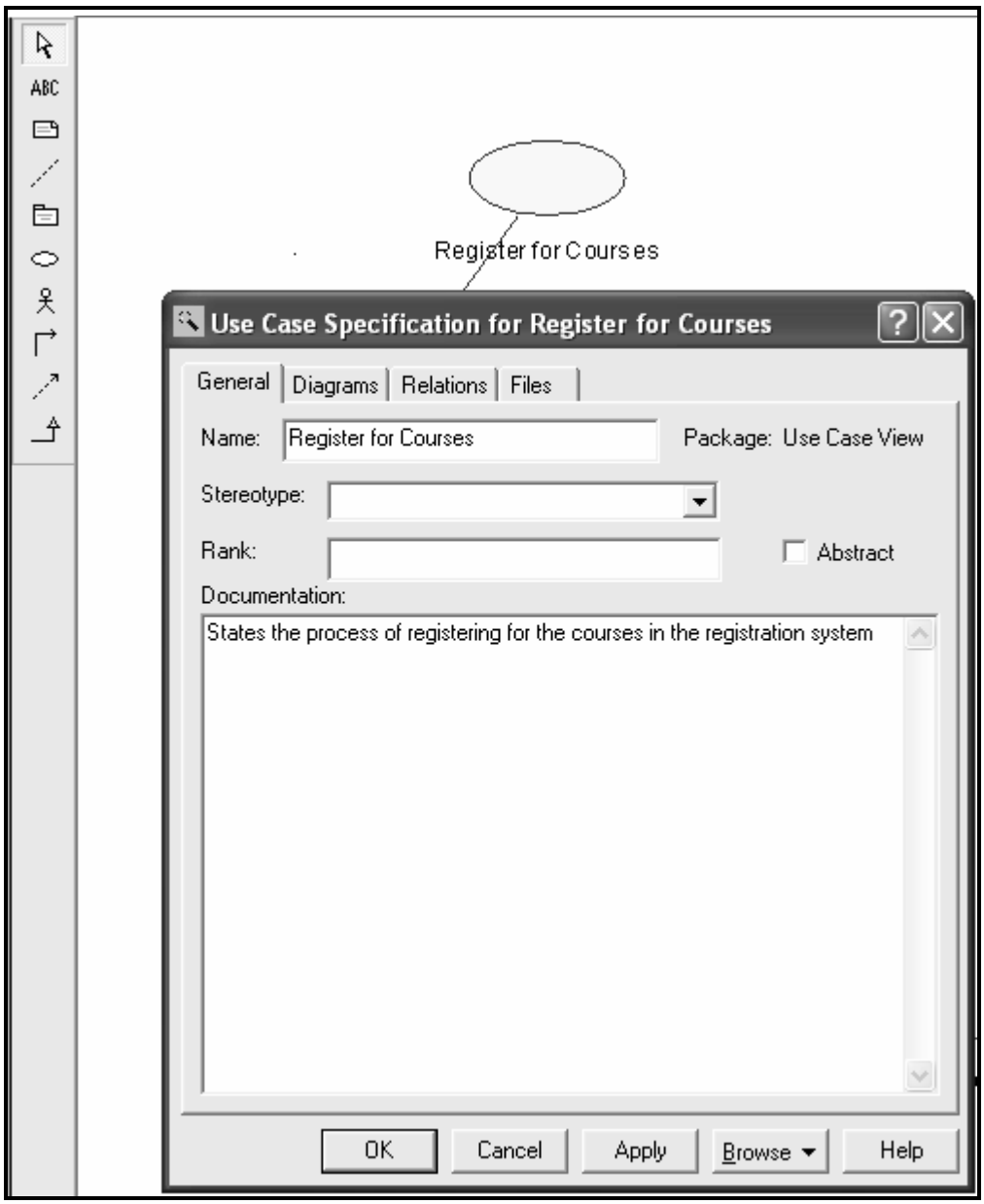
الشكل (4): الرمز المستخدم لتوثيق حالة الاستخدام

ويحتاج المحلل لإضافة وصف مختصر لكل حالة استخدام، حيث يفيد هذا الوصف لبيان الهدف من حالة الاستخدام ويوفر تعريفاً لوظيفتها كما هو موضح في الشكل (5).



الشكل (5): إضافة وصف لحالة الاستخدام

يُعتبر "تسجيل المواد" إحدى حالات الاستخدام التي من الممكن استخراجها مـــن نظام التسجيل، ولتوثيق هذه الحالة وإضافة وصف مختصر لها، نستخدم الرمز البيضاوي الذي تم بيانه في الشكل (4) مع كتابة اسمها أسفل هذا الرمز كما هو موضح في الـــشكل (6) حيث يوضح ترميز حالة الاستخدام "تسجيل المواد" بالإضافة إلى وصفها.



الشكل (6): حالة الاستخدام "تسجيل المواد"

تدريب (2)

استخرج بقية حالات الاستخدام من نظام التسجيل الجامعي وارسمها وأضف وصفاً مختصراً لكل منها.

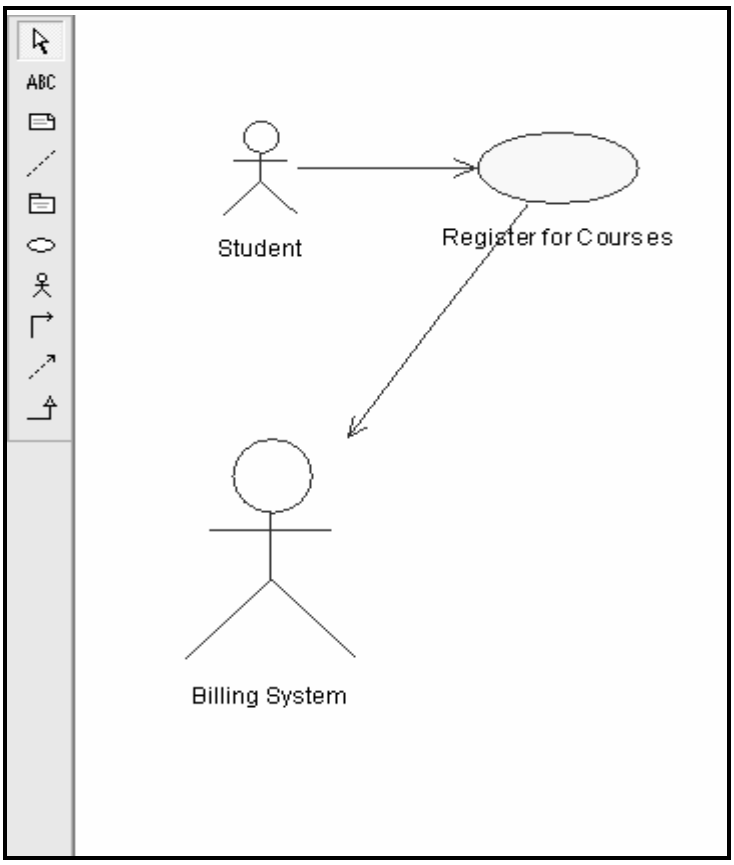
4.3 علاقات حالات الاستخدام

Use – Cases Relationships

بعد تحديد مجموعة المتفاعلين ومجموعة حالات الاستخدام، لا بد من إيجاد العلاقات بين المجموعتين، حيث إنه لا بد لكل متفاعل من استخدام حالة استخدام واحدة أو أكثر. وإذا بقي متفاعل ليس له حالة استخدام أو حالة استخدام بدون متفاعل، تجب إعادة النظر في المجموعتين وتعديلهما.

تسمى العلاقة بين المتفاعل، وحالة الاستخدام بعلاقة التشارك (Associate Relationship)، وتحديد طريقة الاتصال تحدد من الذي يعطي إشارة البدء لتنفيذ خطوات تتابع حالة الاستخدام التي تمكنها من إنجاز الوظيفة المرجوة منها، بمعنى هل حالة الاستخدام هي التي تبدأ بتنفيذ خطواتها أم المتفاعل هو الذي قام ببدء تنفيذ الخطوات؟

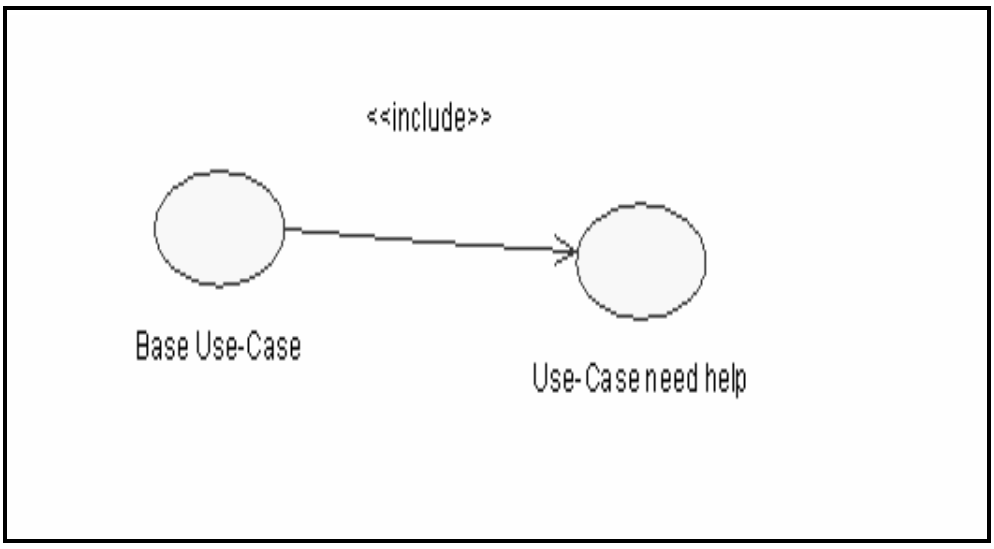
يستخدم عادة خط يربط ما بين الأطراف المشاركة لترميز علاقة التشارك ويمكن تحديد اتجاه التشارك بواسطة الأسهم، حيث يبين اتجاه السهم بداية التنفيذ. والشكل (7) يبين علاقتي التشارك الأولى بين المتفاعل (Student) وحالة الاستخدام (Register for Courses) وتعني أن الطالب هو الذي يعطي إشارة البدء لتنفيذ حالة الاستخدام (Register for Courses) والعلاقة الثانية بين حالة الاستخدام (Register for Courses) والمتفاعل (Billing System)، حيث توضح أن حالة الاستخدام (Register for Courses) هي التي تعطي نتيجة للمتفاعل (Billing System) وهذه النتيجة هي التي تستخدم من قبل هذا المتفاعل للبدء في إنجاز مهامه.



الشكل (7): اتجاه علاقة التشارك بين المتفاعلين وحالات الاستخدام

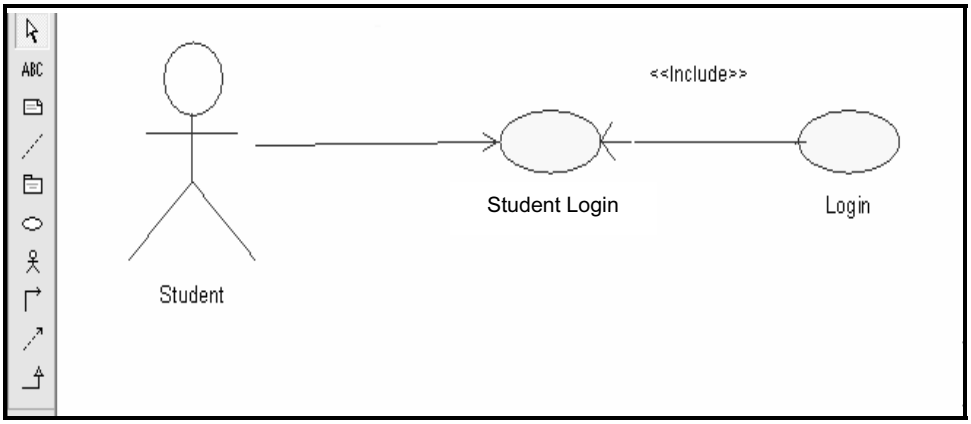
يمكن- عزيزي الدارس- تكوين علاقة بين حالات الاستخدام نفسها، حيث يوجد نوعان من العلاقات:

1. علاقة التضمّن (Include): قد تتشارك العديد من حالات الاستخدام في جزء من وظيفة ما، ففي هذه الحالة يمكن تضمين الجزء المشترك في حالة استخدام واحدة، بدلاً من تكرارها في العديد من حالات الاستخدام هذه، وإنشاء علاقة بينها وبين حالات الاستخدام هذه التي تحتاج هذه الوظيفة. ويساعد هذا النوع من العلاقات في تقليل التكرار غير الضروري مما يساعد على فهم أفضل للنظام وصيانته. والشكل (8) يوضح ترميز هذا النوع من العلاقات، حيث أن حالة الاستخدام (Use-Case need help) تستدعي حالة الاستخدام (Base Use-Case).



الشكل (8): ترميز علاقة التضمّن Include

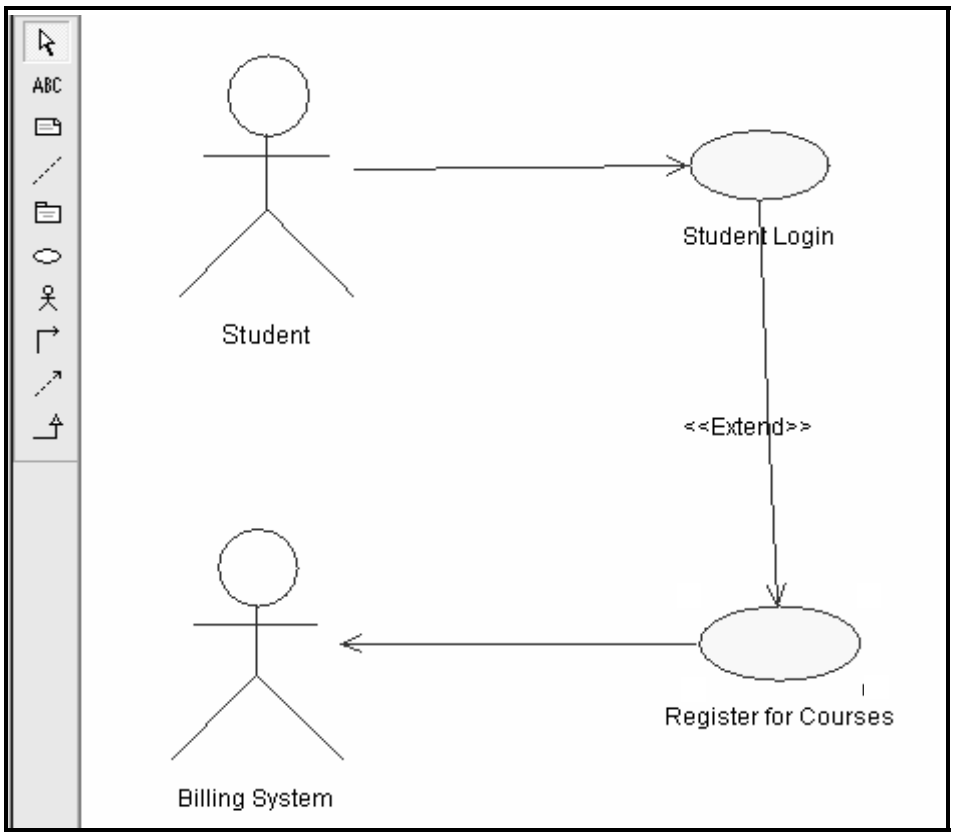
ويبين الشكل (9) مثالاً حياً على علاقة التضمُّن حيث أن حالة الاستخدام (Student_Login) تستخدم علاقة التضمُّن لتربطها مع حالة الاستخدام (Login)، حيث أن محتويات حالة الاستخدام (Student_Login) هي نفس محتويات حالة الاستخدام الأساسية Login بالإضافة إلى وظائف خاصة بها.



الشكل (9): مثال على علاقة التضمّن (Include)

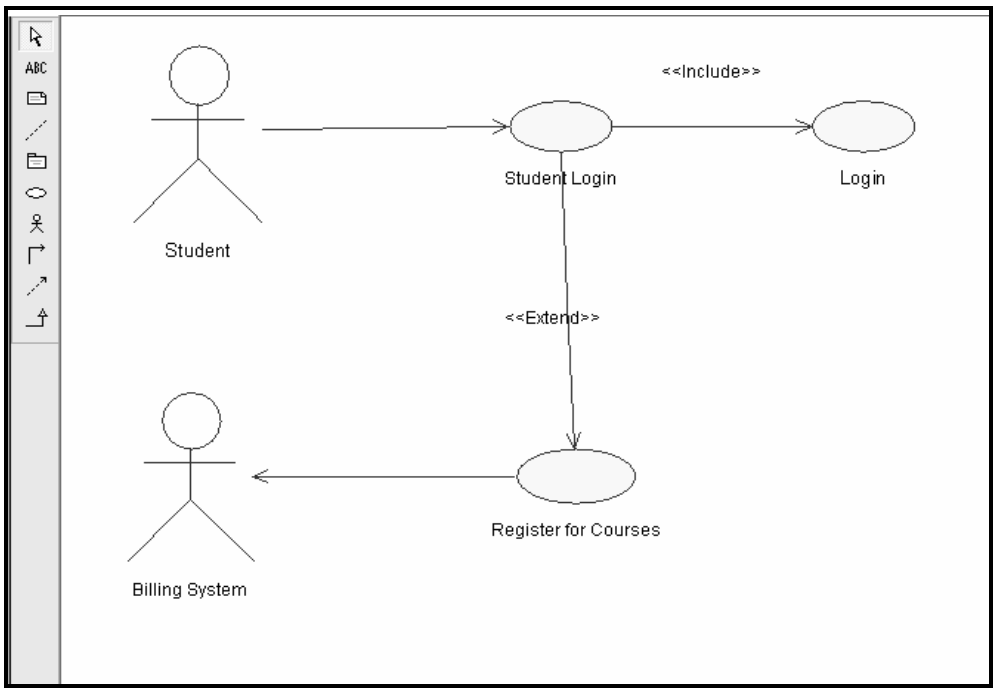
2.علاقة الامتداد (Extend): يستخدم هذا النوع من العلاقات بين حالة استخدام معينة وحالات أخرى حسب تحقق شرط معين. الشكل (10) يوضح مثالاً على استخدام علاقة الامتداد حيث إن على الطالب المرور بحالة (Student_Login) كشرط مسبق

قبل الدخول إلى حالة الاستخدام (Register_for_Courses).



الشكل (10): مثال على علاقة الامتداد (Extend)

يمكنك – عزيزي الدارس– كما هو موضح في الشكل (11)، تضمين كلا النوعين من العلاقات في مخطط واحد مع ملاحظة أن لكل منها وظيفتها المحددة.

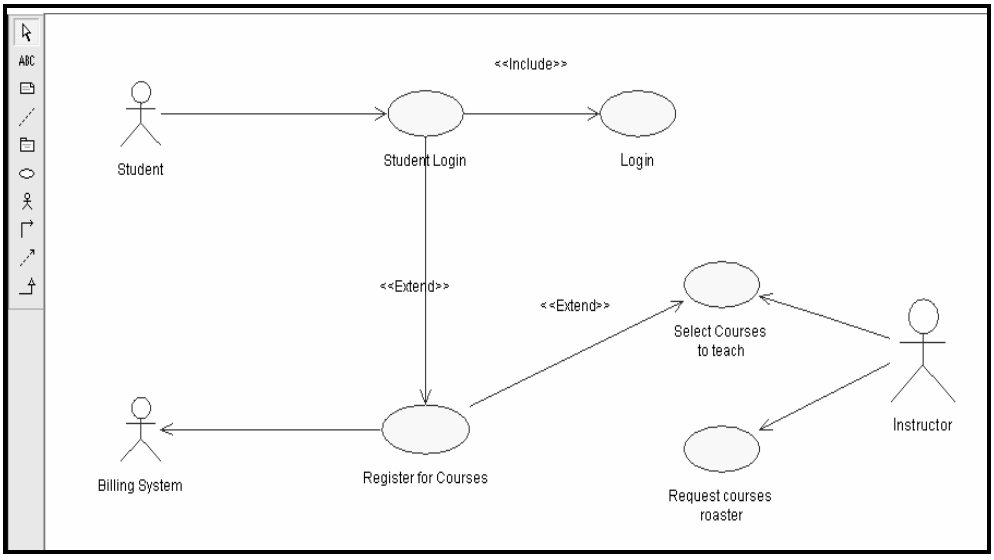


الشكل (11): علاقة التضُّمن Include وعلاقة الامتداد Extend

4. مخططات حالات الاستخدام Use – Cases Charts

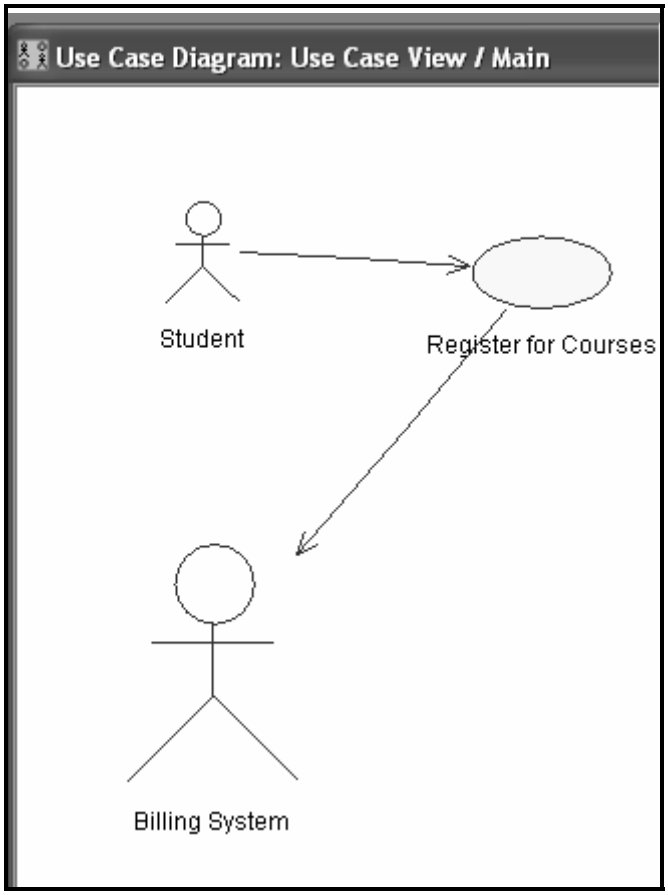
مخطط حالات الاستخدام هو عبارة عن شكل يعبر عن مجموعة المتفاعلين وحالات الاستخدام والعلاقات بينهما، وبما أن النظام يفضل أن يكون مجزأ (Modular)، يمكن تجزئة المخطط الرئيس إلى عدد من المخططات حسب عدد الأجزاء. وتسمى هذه الأجزاء بحدود النظام (System Boundary).

وكل نظام يحتوي عادة على مخطط حالات استخدام رئيس يصف حدود النظام ويبين الوظائف الرئيسة في النظام، وبالإضافة إلى ذلك يستطيع محلل النظم استنباط بعض المخططات الأخرى من المخطط الرئيس وذلك لتسهيل عملية بناء النظام. الشكل (12) يمثل مخطط حالات استخدام في نظام التسجيل الجامعي ويحتوي على ربط لأغلب المتفاعلين وحالات الاستخدام التي تم تحديدها مسبقاً.



الشكل (12): مثال على مخطط حالات الاستخدام

وكمثال من نظام التسجيل، يبين الشكل (13) جزءاً من مخطط يوضح علاقة بعض المتفاعلين بحالة استخدام "تسجيل المواد". حيث نرى، عزيزي الدارس، من خلال الشكل كيف يتعامل كل من المتفاعل "الطالب" والمتفاعل "نظام المحاسبة" مع حالة الاستخدام "تسجيل المواد"، إذ يقوم "الطالب" ببدء التنفيذ وإصدار الأوامر لحالة الاستخدام "تسجيل المواد"، أما المتفاعل "نظام المحاسبة"، فيتلقى النتائج منها.



الشكل (13) جزء من مخطط يحتوي على علاقات بين عدة متفاعلين مع حالة استخدام واحدة

تدريب (3)

ارسم مخطط حالات الاستخدام لنظام التسجيل الجامعي.

5. الخلاصة

لقد تم–عزيزي الدارس– في هذه الوحدة تعريف حالات الاستخدام والمتفاعلين وبيان كيفية بنائها وتوثيقها. كما تم بيان كيفية تحديد العلاقات التي تربط حالات الاستخدام بعضها ببعض من جهة، وبالمتفاعلين من جهة أخرى.

وقد تم بيان أهمية استخدام مخطط حالات الاستخدام وقد رأيت– عزيزي الدارس– بأنه ما هو إلا تجميع للمفاهيم التي تمت مناقشتها في هذه الوحدة، حيث يقوم محلل النظام، خلال مرحلة التحليل، وبناءً على المعلومات التي جمعها أثناء التحليل ببناء هذا المخطط ليبين كيفية تصرف النظام.

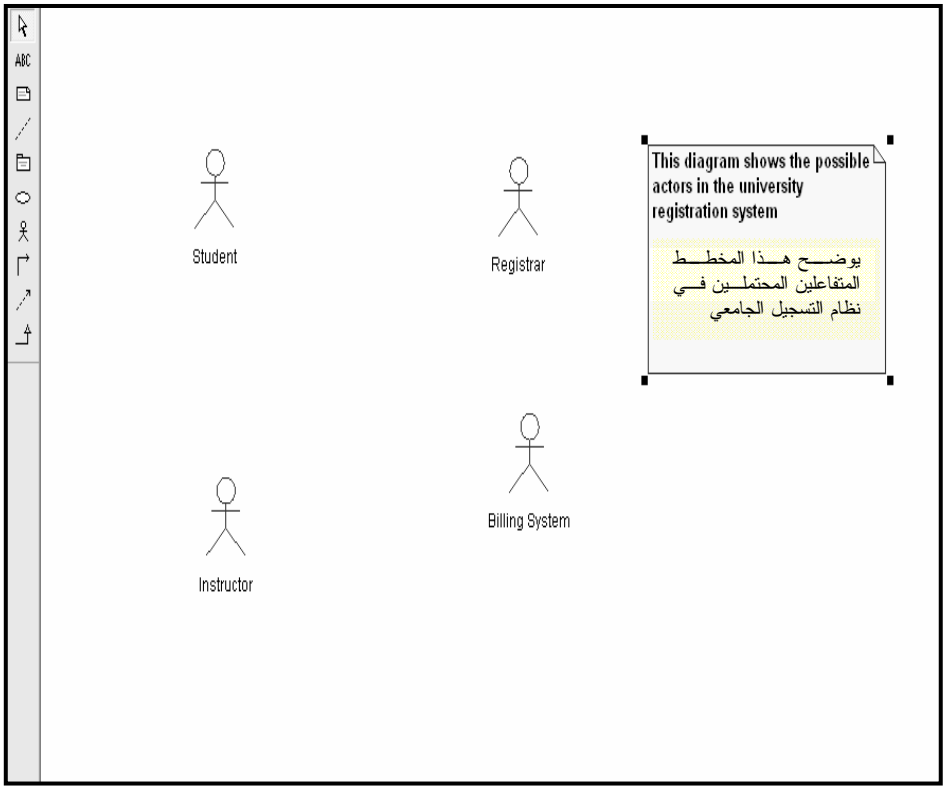
6. لمحة عن الوحدة الدراسية الخامسة

سيتم في الوحدة الخامسة بيان كيفية نمذجة المعلومات المتداخلة والمصنعة في النظام مثل معلومات الطالب أو معلومات المواد المسجلة لكي يتم استخدامها من قبل حالات الاستخدام المعروفة مسبقاً. كما وسيتم مناقشة الأصناف والكينونات التي ستحتوي هذه المعلومات. كما تبين هذه الوحدة أيضاً طريقة ترابط هذه المعلومات بعضها ببعض.

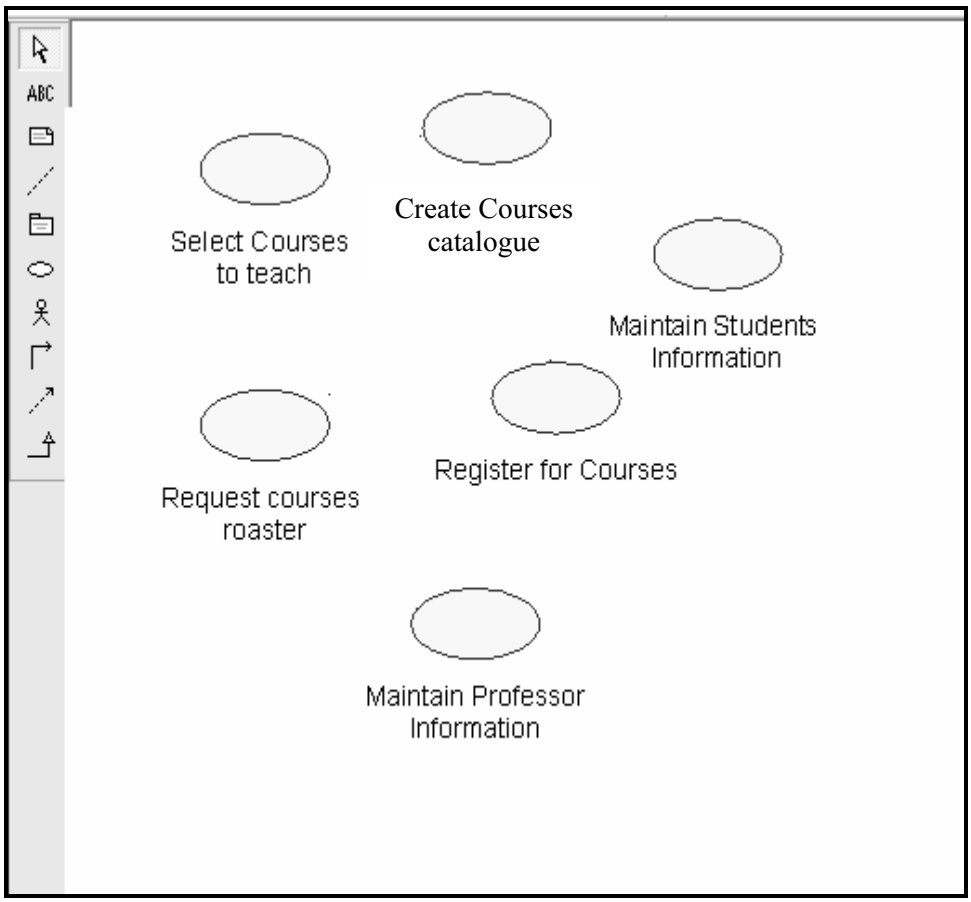
كما تركز هذه الوحدة على بيان واستخدام بطاقة الصنف–المسؤولية–التعاون (CRC Card) والتي تعتبر وسيلة لتحصيل البيانات من أطراف النظام، بالإضافة إلى مخططات الأصناف حيث إنه وباستخدام هذه الأدوات نستطيع أن نتعرف على عناصر البيانات المستخدمة في النظام.

7. إجابات التدريبات

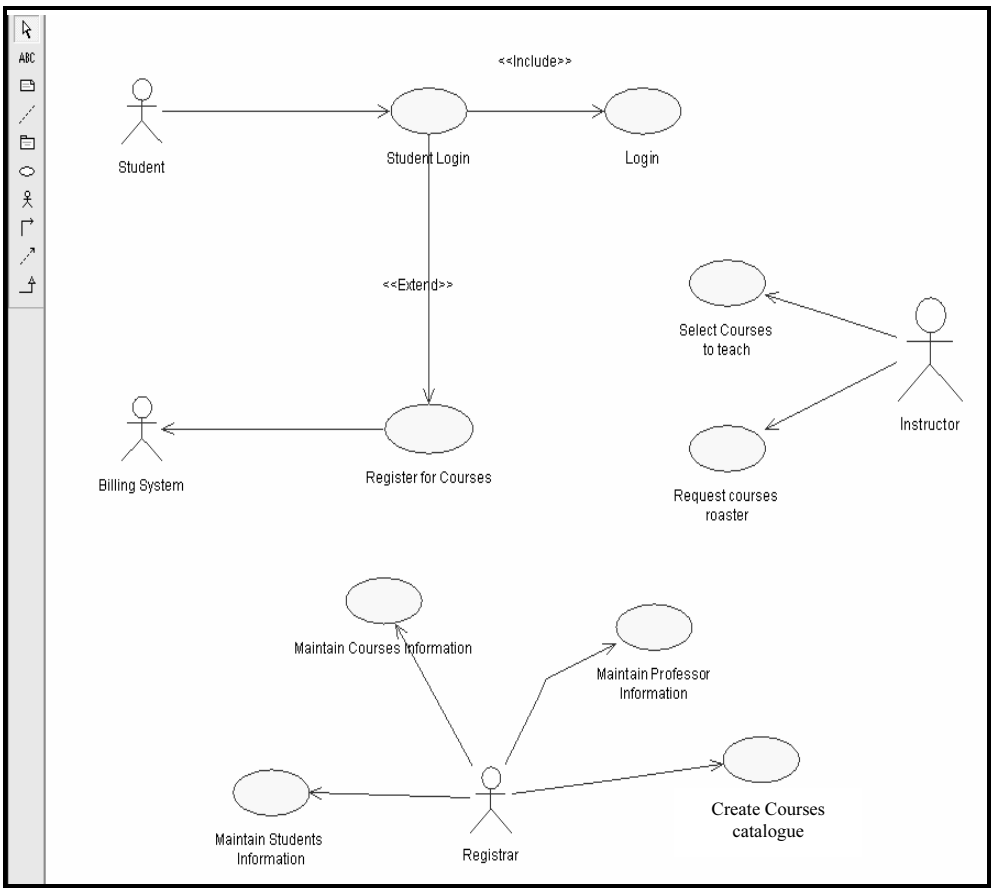
تدريب (1)



تدريب (2)


ABC
Select Courses to teach
Create Courses catalogue
Maintain Students Information
Register for Courses
Request courses roaster
Maintain Professor Information

تدريب (3)


<<Include>>
Student
Student Login
Login
<<Extend>>
Select Courses
to teach
Instructor
Billing System
Register for Courses
Request courses
roaster
Maintain Courses Information
Maintain Professor
Information
Maintain Students
Information
Registrar
Create Courses
catalogue

8. مسرد المصطلحات

- **حالة الاستخدام Use-Case:** تمثل حالة الاستخدام الوظيفة التي يستطيع النظام توفيرها حيث تعبّر هذه الوظيفة عن قدرات النظام؛ أي أن وجود حالة الاستخدام يوفر طريقة فعّالة لوصف كل الطرق التي يمكن فيها استخدام النظام.
- **حدود النظام System Boundary:** هي عملية تجزئة المخطط الرئيس إلى عدد من المخططات حسب الحاجة، ويستخدم لتسهيل بناء النظام وفهمه.
- **علاقة التضمُّن Include:** قد تتشارك العديد من حالات الاستخدام في جزء من وظيفة ما، ففي هذه الحالة يمكن تضمين الجزء المشترك في حالة استخدام واحدة، بدلاً من تكرارها في العديد من حالات الاستخدام هذه، وإنشاء علاقة بينها وبين حالات الاستخدام التي تحتاج هذه الوظيفة.
- **علاقة الامتداد Extend:** يستخدم هذا النوع من العلاقات بين حالة استخدام معينة وحالات أخرى حسب تحقق شرط معين.
- **المتفاعلون Actors:** المتفاعلون هم أطراف خارجية تتعامل مع النظام بمقتضى الحاجة والوظائف المتوفرة في النظام.

9. المراجع

1. Baudoin, C. and Hollowell, G., Realizing The Object-Oriented Life Cycle, Prentice- Hall, 1996.
2. Booch, G., Rumbaugh, J., and Jacobson, I., The Unified Modeling Language User Guide, Addison- Wesley, 1999.
3. Dennis, A. et al, Systems Analysis & Design: An Object-Oriented Approach with UML. John Wiley & Sons Inc., 2002.
4. Dewitz, S., Systems Analysis and Design and the Transition to Objects, McGraw-Hill, 1996.
5. Henderson, P., Object-Oriented Specification and Design with C++, McGraw- Hill, 1993.
6. Kruchten, P., The Rational Unified Process: An Introduction. 2nd Ed., Addison- Wesley, 2000.
7. Lejk, M. and Deeks. An Introduction to Systems Analysis Techniques, 2nd Ed., Pearson Education Limited, 2002.
8. Maciaszek, A., Requirements Analysis and System Design, Developing Information Systems with UML, Addison- Wesley, 2001.
9. Quatrani, T., Visual Modeling with Rational Rose 2000 and UML, Addison- Wesley, 2000.
10. Satzinger, J., Systems Analysis and Design, 2nd Ed., Thomson Learing, 2002.
11. www.rational.com
12. www.omg.com

الوحدة الخامسة

نمذجة بنية النظام

System Structure Modeling

محتويات الوحدة

الموضوع	الصفحة
1. المقدمـــة	111
1.1 تمهيد	111
2.1 أهداف الوحدة	111
3.1 أقسام الوحدة	112
4.1 القراءات المساعدة	112
5.1 ما تحتاج إليه لدراسة الوحدة	112
2. نماذج بنية النظام System Structure Models	113
1.2 الأصناف Classes	113
2.2 الصفات Attributes	114
3.2 الطرق Methods	114
4.2 العلاقات Relationships	114
1.4.2 علاقة التعميم Generalization	115
2.4.2 علاقة التجميع Aggregation	117
3.4.2 علاقة التشارك Association	118
5.2 توثيق الأصناف Classes Documentation	118
3. بطاقات الصنف – المسؤولية–التعاون Classes-Responsibility-Collaboration Cards (CRC Cards)	120
4. مخططات الأصناف Classes Diagrams.....	124
5. الخلاصة.....	126
6. لمحة عن الوحدة الدراسية السادسة	127
7. إجابات التدريبات	127
8. مسرد المصطلحات	133
9. المراجع	134

1. المقدمة

1.1 تمهيد

أهلاً بك، عزيزي الدارس- إلى الوحدة الخامسة من مقرر" 1380 تحليل الأنظمة وتصميمها"، وهي بعنوان " نمذجة بنية النظام".

يقوم محلل النظام، خلال مرحلة التحليل، ببناء مخطط حالات الاستخدام ليبين كيفية تصرف النظام، وفي نفس الوقت يحتاج التحليل لفهم هذه المعلومات المتداخلة والمصنعة في النظام مثل معلومات الطالب أو معلومات المواد المسجلة.

وسيتم في هذه الوحدة بيان كيفية نمذجة هذه المعلومات لكي يتم استخدامها من قبل حالات الاستخدام المعروفة مسبقاً.

نمذجة بنية النظام هي طريقة رسمية لتمثيل البيانات المتداخلة والمصنعة في النظام، حيث توضح الأصناف والكينونات التي ستحتوي المعلومات في النظام، كما تبين أيضاً طريقة ترابط هذه المعلومات بعضها ببعض. إن طريقة بناء نموذج بنية النظام تتم بطريقة تكرارية حيث يتم تعديله خلال مرحلة التحليل ليصبح مفصلاً بشكل كبير ويكون جاهزاً للاستخدام خلال مرحلة التصميم كنموذج مفاهيمي (Conceptual Model).

أما في مرحلة تصميم النظام، فيقوم المحلل بتطوير نموذج بنية النظام المفاهيمي إلى نموذج تصميم يعكس ويوضح كيفية تنظيم البيانات في قواعد البيانات والملفات وفي المراحل المتقدمة في النظام يقوم المحلل بتطوير النموذج إلى صورته النهائية القابلة للاستخدام الفعلي.

تركز هذه الوحدة أيضاً- عزيزي الدارس- على بيان طريقة إنشاء نموذج مفاهيمي لبنية النظام للبيانات حتى يتسنى استخدام بطاقة الصنف-المسؤولية-التعاون (CRC Card) والتي تعتبر وسيلة لتحصيل البيانات من أطراف النظام، بالإضافة إلى مخططات الأصناف حيث إنه وباستخدام هذه الأدوات نستطيع أن نتعرف على عناصر البيانات المستخدمة في النظام.

2.1 أهداف الوحدة

يتوقع منك، عزيزي الدارس، بعد الانتهاء من دراسة هذه الوحدة أن تكون قادراً على أن:

1. توضح أهمية نماذج بنية النظام.

2. تحدد عناصر بنية النظام.
3. تبين العلاقات بين الأصناف
4. تستخدم بطاقة الصنف– المسؤولية– التعاون
5. ترسم مخططات الأصناف

3.1 أقسام الوحدة

تقسم هذه الوحدة إلى ثلاثة أقسام رئيسة:

القسم الأول يبين عناصر بنية النظام، وهذا يغطي الأهداف الثلاثة الأولى. أما القسم الثاني: فيناقش استخدام بطاقة الصنف– المسؤولية– التعاون وكيفية بنائها، ويغطي الهدف الرابع، أما القسم الأخير فيغطي الهدف الخامس؛ ويبين طريقة رسم مخططات الأصناف.

4.1 القراءات المساعدة

أنصحك–عزيزي الدارس– بالإطلاع على المراجع التالية لما تحويه من معلومات مفيدة تتعلق بموضوع هذه الوحدة:

1. Baudoin, C. and Hollowell, G., Realizing The Object-Oriented Life Cycle, Prentice- Hall, 1996.
2. Booch, G., Rumbaugh, J., and Jacobson, I., The Unified Modeling Language User Guide, Addison- Wesley, 1999.
3. Dennis, A. et al, Systems Analysis & Design: An Object-Oriented Approach with UML. John Wiley & Sons Inc., 2002.
4. Dewitz, S., Systems Analysis and Design and the Transition to Objects, McGraw-Hill, 1996.

5.1 ما تحتاج إليه لدراسة الوحدة

عزيزي الدارس؛ قبل أن تبدأ بدراسة هذه الوحدة تأكد من أنك قمت بتهيئة المكان الهادئ المناسب للدراسة مما يساعدك على التركيز والتفكير العميق. ثم احرص على تتبع الإرشادات التي توجهك لدراسة الوحدة، وحاول بجدية الإجابة عن أسئلة التقويم الذاتي، وحل التدريبات، لأنها تساعدك على مراجعة موضوعات الوحدة، وتعمق معلوماتك حولها.

2. نماذج بنية النظام System Structure Models

على المحلل – عزيزي الدارس– حين القيام بتحليل نظام ما، أن يكـون مطلعـاً على مجال النظام المنوي بناؤه، فإذا أراد بناء نظام محاسبة على سبيل المثال، عليـه أن يكون عارفاً بأساسيات الطرق المحاسبية، وذلك حتى يتسنى له بناء نظام صحيح بطريقـة سلسة ومعرفة البيانات الرئيسة المحتواة في مجال النظام، والتغلب على الصعوبات التـي قد تواجهه في معرفة كيفية تحويل البيانات المجردة إلى بيانات عملية تخزن على جهـاز الحاسوب.

ومن نتائج بناء بنية النظام كذلك إنشاء معجم للبيانات (Data Dictionary) بحيث يكون ذا استخدام مشترك ما بين المحلل والمستخدم ويوفر قناة اتـصال فعالـة بينهمـا. وتستخدم عادة بطاقات الصنف –المسؤولية– التعاون ومخططات الأصناف في بناء نموذج بنية النظام. ويتكون نموذج بنية النظام من عدة عناصـر هـي: الأصـناف (Classes)، والصفات (Attributes)، والطرق (Methods)، والعلاقات (Relationships).

1.2 الأصناف Classes

الأصناف عبارة عن قالب عام نستخدمه لتصنيع الكينونـات (Objects). وكـل الكينونات المشتقة من صنف واحد متشابهة في التركيب والتعريف ومختلفة في البيانـات المحتواة في خصائصها. هناك ثلاثة أنواع عامة ومختلفة من الأصناف تهم محلل النظم:

- النوع الأول: الأصناف الصلبة (Solid Classes): عندما تكون الأصـناف ذات علاقـة مباشرة بالوصف الفني لمجال النظام (Technical Domain) نقول بأنهـا أصـناف صلبة ويمكن اعتبارها أصنافاً ملموسة (Tangible Classes). كصنف الراتـب فـي مؤسسة أو الكتاب في نظام المكتبة.
- النوع الثاني: الأصناف المفاهيمية (Conceptual Classes): عندما تشير الأصناف إلـى كينونات معنوية أو مفاهيمية غالباً ما تكون غير ملموسة، نسميها أصنافاً مفاهيميـة. فعلى سبيل المثال، يعتبر صنف الموظفين وصنف الزبائن اللـذين ينـدرجان تحـت صنف الأشخاص أصنافاً مفاهيمية.
- النوع الثالث: الأصناف الحقيقية (Real World Classes) كأصناف واجهات الاسـتعمال وأصناف تركيب البيانات وأصناف أنواع قواعد البيانات والملفات أو أي صنف مـن الأصناف التي يحتاجها النظام في الناحية العملية التقنية أثناء بناء النظـام وتكـون خاصة بالمبرمجين، كصنف أزرار الشاشة (Screen Buttons) أو صـنف الطـابور (Queue).

2.2 الصفات Attributes

إن صفة الصنف هي عبارة عن قطعة معلومات مفيدة ترتبط بوصف جزء معين من هذا الصنف. والصفات تحتوي عادة على المعلومات التي يعتقد المحلل أو المستخدم بأن على النظام الاحتفاظ بها. فعلى سبيل المثال، إن صفات الموظف كثيرة مثل الاسم، والطول، ولون الشعر، والعنوان، ولكن من وجهة نظر المحلل؛ إن صفتَي الاسم والعنوان هما الصفتان المفيدتان اللتان يجب الاحتفاظ بهما في النظام، أما الطول، ولون الشعر، فغير مفيدتين في هذا النظام فلذلك ليس علينا الاحتفاظ بهما.

أسئلة التقويم الذاتي (1)

عرف صفة الصنف.

3.2 الطرق Methods

الطريقة هي ذلك الجزء من الصنف الذي يعرف فيه تصرف الصنف، وتكون الطريقة عادة متعلقة بشكل كبير بطور التطبيق. ويمكن تعريف الطريقة بأنها الأفعال التي تنفذها كينونات مشتقة من صنف معين. كما هو الحال في صفات الأصناف، ويجب أن تكون الطرق مرتبطة بالصنف وأن تكون الطريقة ذات أهمية في النظام، كما يجب على الكينونة في صنف معين أن تكون قادرة على تعديل أي من صفات تلك الكينونة، فعلى سبيل المثال يستطيع صنف الطالب أن يحتوي على طريقة تسجيل مادة لكن لا يستطيع أن يحتوي على طريقة أكل الطعام على الرغم من أن أكل الطعام هو إحدى الطرق التي يقوم بها الطالب لكن هذه الطريقة غير مهمة للمستخدمين وليست ذات علاقة بمحتوى النظام.

أسئلة التقويم الذاتي (2)

ما هي الشروط الواجب توفرها في الطريقة؟

4.2 العلاقات Relationships

العلاقات هي عبارة عن روابط بين الأصناف، وهي ثلاثة أنواع: علاقات التعميم، وعلاقات التجميع وعلاقات التشارك. وتجدر الإشارة هنا – عزيزي الدارس–

إلى أنه يجب على المحلل أن يستخدم العلاقات ذات الصلة بالنظام. وسيتم في الفقرة التالية بيان هذه الأنواع:

1.4.2 علاقة التعميم Generalization

علاقة التعميم تمكن المحلل من إنشاء أصناف تستطيع توارث صفات أصناف أخرى والعمليات التي تحويها. وتسمى الأصناف التي تورث الصفات والعمليات لأصناف أخرى بالأصناف الرئيسة (Super Class)، أما الأصناف التي ترث من غيرها بالأصناف الفرعية(Sub Class). ويمكن للصنف أن يكون رئيساً وفرعاً في نفس الوقت، إذا كان يورث ويرث في نفس الوقت.

وتسمى العلاقة من الصنف الرئيس إلى الصنف الفرعي بالتخصيص Specialization، والعلاقة من الصنف الفرعي إلى الرئيس بالتعميم Generalization.

مثال (1)

لو فرضنا أن المحلل جمع المعلومات التالية عن عملاء شركة ما:

1. عملاء الشركة هم الذين يتعاملون مع الشركة حيث تبيع لهم البضاعة التي تتعامل بها.
2. تصنف الشركة عملاءها كعملاء خارجيين أي من خارج فلسطين وعملاء داخليين من داخل فلسطين.
3. يقسم العملاء الداخليين إلى عملاء دائمين وعملاء غير دائمين.
4. الصفات اللازمة عن العملاء هي:

أ) العملاء الخارجيّون: الرقم، والاسم، والعنوان، ورقم الهاتف، ونوع العملة، وطريقة التصدير، والبلد.

ب) العملاء الداخليوّن:

1- العملاء الدائمون: الرقم، والاسم، والعنوان، ورقم الهاتف، وصندوق البريد، والخصم.

2- العملاء غير الدائمين: الرقم، والاسم، والعنوان، ورقم الهاتف، وصندوق البريد، والمحافظة.

الآن، عزيزي الدارس: يمكن تحديد صنف واحد للعملاء ونضع كل هذه الصفات فيه، ولكن لو أمعنا النظر في هذا الحل لوجدنا أنه يحتوي على قيم صفات مكررة في بعض الأحيان وصفات فارغة أحياناً أخرى، وذلك لأن العملاء مصنّفون إلى عدة

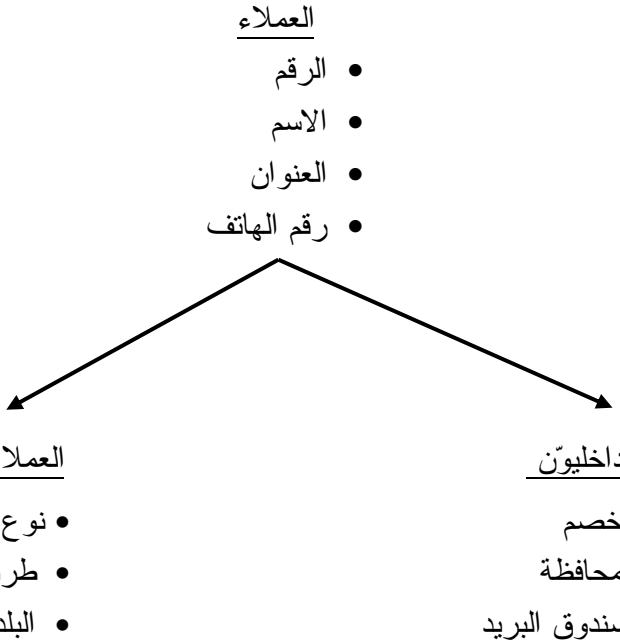
تصنيفات وكل تصنيف يحتوي على صفات معينة، يتشابه بعضها مع التصنيفات الأخرى ويختلف مع بعضها الآخر. إذاً ما هو الحل؟؟

نلاحظ بأن بعض الصفات يشترك فيها كل أصناف العملاء وهي: الاسم، والعنوان، ورقم الهاتف، فنحدد صنفاً رئيسا نسميه مثلاً العملاء ونضع فيه هذه الصفات المشتركة:

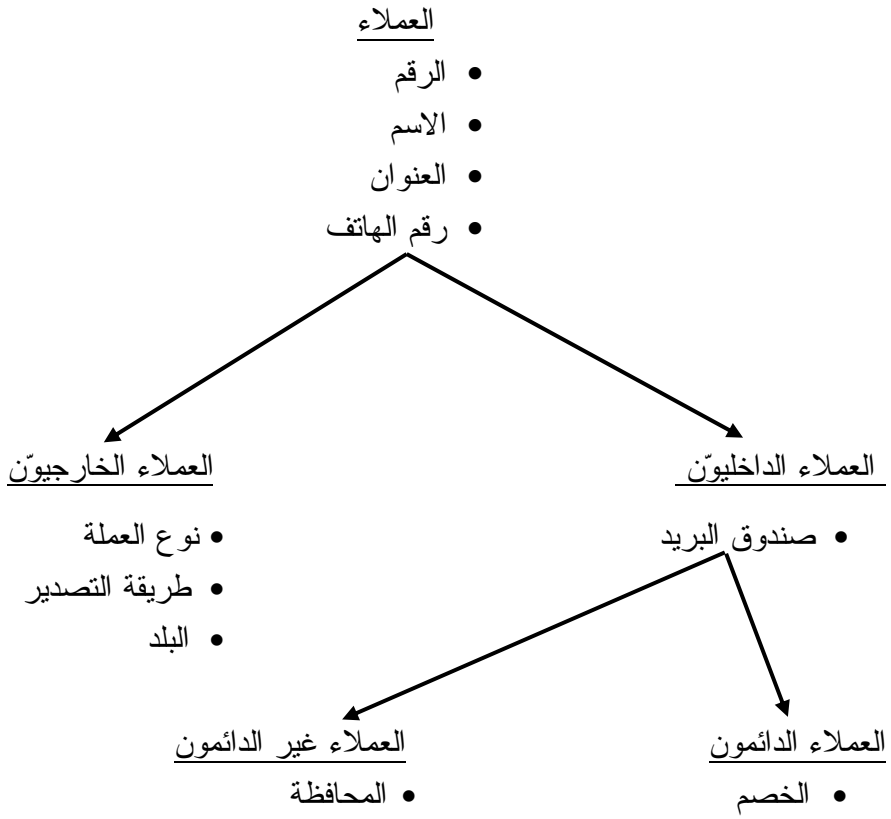
العملاء

- الاسم
- العنوان
- رقم الهاتف

ونلاحظ أن العملاء يصنفون إلى تصنيفين هما: الداخليّن والخارجيّون، فنحدد صنفاً لكل منهما ونضع في كل صنف الصفات المتبقية لكل منها، بعد استثناء صفات صنف العملاء، ونربطهما مع صنف العملاء وذلك كالآتي (لاحظ- عزيزي الدارس- بأن علاقة التعميم يرمز لها بخط على رأسه مثلث من جهة الصنف الفرعي):



ونكرر العملية على صنف العملاء الداخليين، حيث أنه يقسم إلى عملاء دائمين وعملاء غير دائمين، فيصبح الوضع كالآتي:



وتكون صفات الصنف مكونة من صفاته بالإضافة إلى صفات الأصناف الرئيسة بالنسبة له، فمثلاً صفات الصنف للعملاء الداخليين تكون كالآتي: الرقم، والاسم، والعنوان، ورقم الهاتف، وصندوق البريد، وهكذا...

2.4.2 علاقة التجميع Aggregation

يستخدم هذا النوع من العلاقات في نمذجة البيانات وتمثيل المعلومات بتجميع صفات أصناف عديدة لتكوين صنف واحد. ولعلاقة التجميع عدة أشكال، أهمها:

أ) "علاقة جزء من" (Part-of): كعلاقة باب السيارة بالسيارة، نقول: إن باب السيارة جزء من السيارة.

ب) "محتواة في" (Contained-in): عبارة عن مجموعة جزئية محتواة في مجموعة كلية، كصنف مجموعة طلاب السنة الأولى محتواة في صنف مجموعة الطلاب.

ت) "مشارك بـ" (Participate-in): عبارة عن عدة أصناف مختلفة الصفات تساهم جميعها في تكوين صنف جديد كل حسب دوره، كأعضاء فريق كرة القدم كلهم

يشكلون صنف الفريق الكلي ولكل منهم دور معين.

أسئلة التقويم الذاتي (3)

عدد أشكال علاقة التجميع.

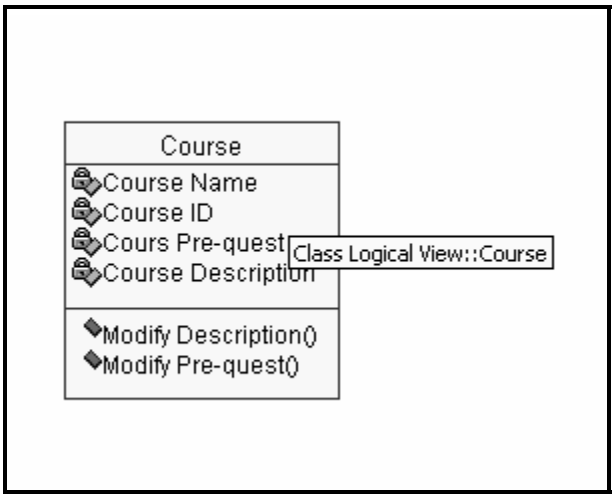
والجدير بالذكر-عزيزي الدارس- أنه ليس بالضرورة أن تكون الأصناف المرتبطة بعلاقة تجميعية متشابهة من حيث التكوين، كأجزاء السيارة مثل الباب، والشباك، والزجاج، والدواليب، والهيكل كلها بمجموعها تشكل صنف السيارة مع العلم أنها لا تتشابه بعضها مع بعض من الناحية التكوينية. وتسمى علاقة أجزاء السيارة بالسيارة بعلاقة التجميع (Aggregation)، في حين تسمى علاقة السيارة بأجزائها بعلاقة التفصيل (De-Aggregation).

3.4.2 علاقة التشارك Association

هناك أنواع أخرى من العلاقات لا تنطوي بشكل مباشر تحت لوائي علاقات التجميع والتعميم، وتكون هذه العلاقات- عادة- عبارة عن شكل معين من علاقات التجميع ولكن بطبيعة مختلفة. فعندما يحدد للمريض- على سبيل المثال- موعد من الطبيب، يمكن القول هنا: أن المريض جزء من الموعد، ولكن لا يصح الموعد إلا به وهذا شرط غير موجود في علاقة التجميع.

5.2 توثيق الأصناف Classes Documentation

يجب على المحلل القيام بتوثيق كل صنف في النظام، حيث يتم توثيق الأصناف باستخدام مربع مقسم إلى ثلاثة أقسام، كما هو مبين في الشكل (1)، حيث يتم كتابة اسم الصنف في القسم الأول، والصفات في القسم الثاني، والعمليات في القسم الثالث. ويوضح الشكل (1) صنف (Course) في نظام التسجيل، حيث يحتوي هذا الصنف على صفات عديدة مثل اسم المادة (Course Name)، ورقم المادة (Course ID)، والمتطلب السابق (Course Prerequisite)، ووصف المادة (Course Description). والعمليات التي يقوم بها هذا الصنف أو التي تطبق عليه وهي: تعديل محتوى المادة وتعديل المتطلب السابق للمادة.

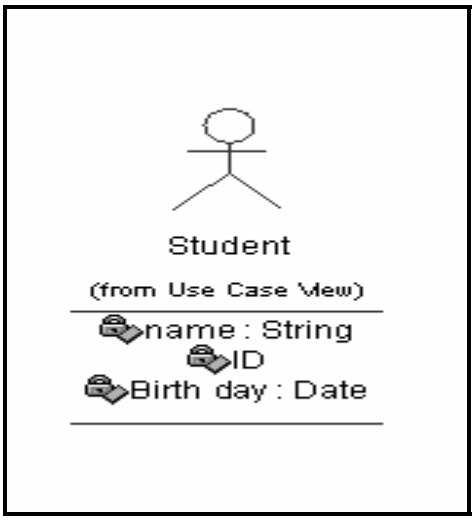


الشكل (1): توثيق الصنف Course

أسئلة التقويم الذاتي (4)

اذكر أقسام مربع توثيق الصنف ومحتويات كل منها.

ويمكن- عزيزي الدارس-أن يكون الصنف هو أحد المتفاعلين كما هي الحالة في نظام التسجيل الجامعي، فالمتفاعل الطالب هو أحد الأصناف، وفي هذه الحالة يرمز له بشكله الأصلي مع إضافة معلومات الصنف، كما هو موضح بالشكل (2).



الشكل (2): توثيق الصنف عندما يكون متفاعلاً

3. بطاقات الصنف–المسؤولية–التعاون
Class-Responsibility- Collaboration (CRC)

تستعمل بطاقات الصنف–المسؤولية–التعاون لتوثيق المسؤوليات والتعاونيات المناطة بكل صنف من أصناف النظام. وتقسم مسؤوليات أي صنف إلى نوعين: المعرفة (Knowing) والفعل (Doing). ومسؤوليات المعرفة هي تلك الأصناف التي تكون كينوناتها قادرة على معرفة قدرات وأفعال الكينونات الأخرى. أما مسؤوليات الفعل فتكون في الأصناف القادرة على تنفيذ العمليات. أما التعاونيات فهي متعلقة بالأصناف التي تساعد في تطبيق هياكل البيانات الخاصة بالنظام.

تصنف الكينونات حسب مسؤولياتها وتعاونياتها إلى ثلاثة أنواع: كينونة العميل (Agent Object)، وكينونة الخادم (Server Object)، وكينونة العقد (Contract Object). وكينونة العميل هي عبارة عن كينونة مشتقة من صنف ما وتقوم بإرسال طلب إلى كينونة مشتقة من صنف آخر لتقوم بتنفيذ عملية ما لحساب الكينونة الأولى، وتسمى الكينونة الثانية بكينونة الخادم. أما كينونة العقد فتقوم بتأمين التواصل بين العميل والخادم. فعلى سبيل المثال– عزيزي الدارس– عندما يقوم المريض بحجز موعد مع الطبيب، ينشأ التزام بين الطبيب والمريض للتواجد في الموعد المحدد، ويمكن التعامل مع تبعات هذا الموعد، مثل المحاسبة وتوثيق الموعد، عن طريق كينونة العقد التي تقوم بترتيب طريقة للتواصل ما بين العميل (المريض) والخادم (الطبيب). وهكذا تكون مسؤوليات العقد من نوع مسؤوليات المعرفة، وذلك لأنه يتوجب على مسؤوليات العقد معرفة قدرات الكينونات الأخرى.

ويستطيع المحلل استخدام فكرة مسؤوليات الأصناف وفكرة العميل والخادم والعقد لتحديد الأصناف وصفاتها والعمليات والعلاقات بينها. وإن إحدى أسهل الطرق لتكوين بنية النظام هي استخدام بطاقات الصنف–المسؤولية–التعاون، وذلك بتخيل الصنف المفترض كأنه إنسان يحاوره المحلل بمجموعة الأسئلة مثل: من أنت ؟، ماذا تعرف ؟، ما هو فعلك؟... حيث تستخدم إجابات هذه الأسئلة كمرشد لتطوير بطاقات الصنف – المسؤولية – التعاون.

وتحتوي بطاقات الصنف–المسؤولية–التعاون على كل المعلومات الضرورية لبناء نموذج بنية النظام بشكل منطقي، ولكل بطاقة وجهان، يستعمل الوجه الأمامي لوصف العناصر الأساسية لكل صنف، كاسم الصنف ورقمه، ونوعه، وقائمة بأسماء حالات الاستخدام المشتركة والمسؤوليات والتعاونيات المناطة بها. حيث يجب أن يكون

الصنف اسماً، لكن ليس اسم علم أو شيئاً (Object)، أما وصف الصنف فيمكن أن يكــون جملة مختصرة قابلة لأن تكون تعريفاً نصّياً للصنف. ومسؤوليات الصنف تشبه إلى حــد ما العمليات التي على الصنف أن يقوم بها.

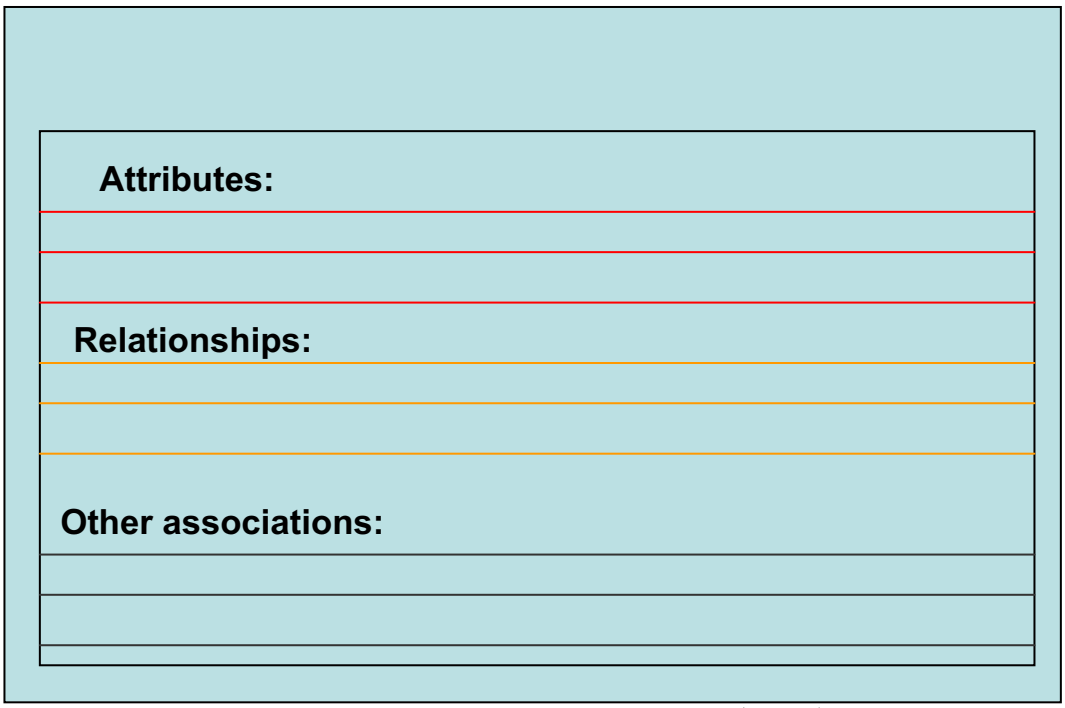
أما الوجه الخلفي للبطاقة فيحتوي على الصفات والعلاقات الخاصة بها، ويمثــل مسؤوليات المعرفة التي تحتاج إليها كل كينونة من الصنف عدا نوع البيانات، وكل صفة تكون مرتبطة إما بصنف معين ومتغير من متغيرات اللغة، فعلى ســبيل المثــال، صــفة الراتب تمثل بمجموعة الأرقام الحقيقية وأما صفة اسم المشترك بالتأمين فيمكن أن تمثــل بنص.

وتجدر الإشارة هنا – عزيزي الدارس– أننا نستطيع اشتقاق العلاقات الثلاث التي ذكرت سابقاً : علاقة التعميم، وعلاقة التجميع، وعلاقة التشارك، في هذه المرحلــة مــن التحليل خصوصاً أن البطاقة تستخدم عادة لتوثيق الخصائص الأساسية للأصناف.

الشكل (3) يبين مثالاً للشكل العام لبطاقة الصنف–المسؤولية–التعاون، حيث يبين الــشكل (3–أ) الوجه الأمامي للبطاقة والشكل (3– ب) يبين الوجه الخلفي لها.

Class Name:	**ID:**	**Type**
Description: here you can insert the class description		
Responsibilities	**Collaborations**	

الشكل (3–أ): الوجه الأمامي لبطاقة الصنف–المسؤولية–التعاون



الشكل (3– ب): الوجه الخلفي لبطاقة الصنف–المسؤولية–التعاون

مثال (2)

يبين الشكل (4–أ) الوجه الأول لبطاقة CRC الخاصة بصنف Student، حيث تبين مسؤوليات وتعاونيات الطالب. بينما يبين الشكل (4– ب) الوجه الخلفي للبطاقة حيث يبين صفات الطالب وعلاقاته، بالإضافة إلى الأصناف التي يتشارك معها.

Class Name: Student	ID: 01	Type: General
Description: this class represents a general student class		
Responsibilities	Collaborations	
Login to registration system	String	
Register for Courses	Network and Connection	
Know the Offered Courses		

الشكل (4–أ): الوجه الأمامي لبطاقة (CRC) الخاصة بالصنف (Student)

Attributes:
Number, Name, Year, Tel-No
Relationships:
Courses
Instructors
Other associations: PartTime-Student

الشكل (4–ب): الوجه الخلفي لبطاقة (CRC) الخاصة بالصنف (Student)

4. مخططات الأصناف Classes Diagrams

إن مخطط الأصناف عبارة عن مخطط ثابت (Static) يحتوي على الأصناف ويوضح العلاقات بينها. ويحتوي مخطط الأصناف عادة على عدة عناصر أهمها:

أولاً: الأصناف التي ترسم على شكل مربع يحتوي عدة خلايا كما تم بيانه سابقا.

ثانياً: العلاقات التعميمية والتشاركية، حيث أن أحد أهم أهداف مخطط الأصناف هو توضيح المعلومات والعلاقات ما بين الأصناف التي ترسم على شكل خطوط. وكل هذا يرسم ويعنون باسم العلاقة، على سبيل المثال- علاقة تجميع أو علاقة تعميم.

ثالثاً: التعددية حيث تعني عدد الكينونات من صنف ما المرتبطة بصنف آخر، حيث إن الأرقام الموضوعة على خط التشابك توضح أقل أو أكثر عدد ممكن من الكينونات. الجدول (1)، يبين بعض أنواع التعددية المستخدمة بشكل واسع.

الجدول (1): أنواع التعددية

Quantity Notation	Meaning
1	Exactly one
0..*	Zero or more
1..*	One or more
0..1	Zero or one
2..4	Specified range
1..3,5	Multiple, disjoint

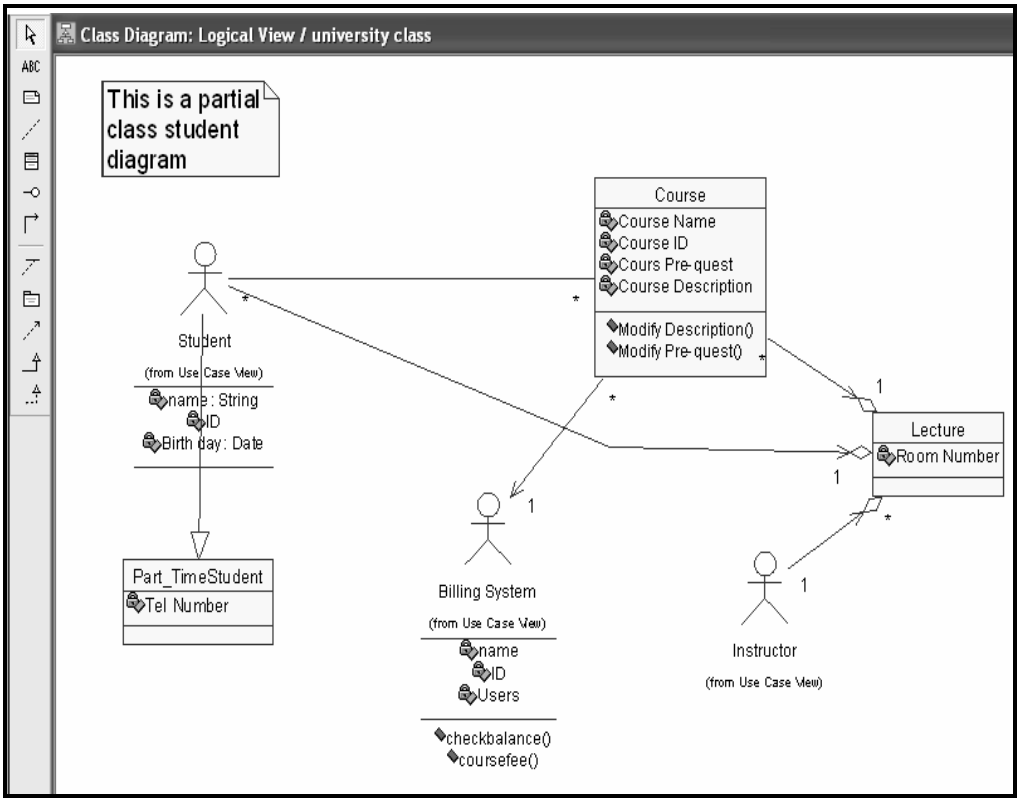
وسيتم- عزيزي الدارس- شرح الأصناف ومخططاتها بشكل مفصل في الوحدة التاسعة من هذا المقرر.

مثال (3)

الشكل (5) يبين جزءاً من مخطط أصناف نظام التسجيل، حيث تظهر ستة أصناف هي: (Part_TimeStudent, Lecture, Instructor, Student, Course, Billing System). لاحظ – عزيزي الدارس – أن صنف Student هو صنف وحالة استخدام في نفس الوقت، وكذلك الأمر بالنسبة للصنفين (Billing System) و(Instructor). ويبين المخطط كذلك أن الصنف الفرعي (Part_TimeStudent) هو حالة خاصة من الصنف الرئيس (Student)، والذي بدوره يرتبط مع الصنف Course بعلاقة تعددية من الجهتين وتعني أن الطالب يستطيع أن يسجل أكثر من مادة والمادة يستطيع أن يسجلها أكثر من طالب، ويرتبط مع الصنف (Lecture) بعلاقة تعددية تعني أن الطالب لا يستطيع أن يحضر أكثر من محاضرة أما المحاضرة فيستطيع أن يحضرها أكثر من طالب. والعلاقة بين المدرس والمحاضرة علاقة تعددية أيضاً، وتعني أن المدرس يعطي أكثر من محاضرة، أما المحاضرة فيعطيها مدرس واحد.

لاحظ – عزيزي الدارس– أن صنف الطالب وصنف المدرس وصنف المادة كلها ترتبط بعلاقة تجميعية مع المحاضرة، وهذا يعني أن المحاضرة تتكون من مدرس ومادة وطالب.

وأخيراً، يبين الشكل علاقة تشاركية تعددية بين الصنف Course والصنف (Billing System)، حيث تعني هذه العلاقة أن الفاتورة يمكن أن تحتوي على أكثر من مادة والمادة توجد في فاتورة واحدة، والعلاقة التشاركية تعني أنه لا يمكن إصدار فاتورة إلا بوجود مواد.



الشكل (5): جزء من مخطط أصناف نظام التسجيل

تدريب (1)

قم، عزيزي الدارس، برسم بطاقات الصنف–المسؤولية–التعاون للأصناف التي حددتها في التدريبات السابقة لنظام التسجيل، ثم ارسم مخطط الأصناف وبّين عليه أنواع العلاقات والتعددية.

5. الخلاصة

لقد تم، خلال هذه الوحدة، بيان مفهوم بنية النظام وأهميته وعناصره: الأصناف (Classes)، والصفات (Attributes)، والطرق (Methods)، والعلاقات (Relationships). وتم التطرق إلى تعريف الأصناف واستخدام بطاقات الصنف–المسؤولية – التعاون بالإضافة إلى شرح عملية رسم مخططات الأصناف.

6. لمحة عن الوحدة الدراسية السادسة

تعنى الوحدة السادسة بشرح نموذج سلوك النظـــام وبيـــان أهميتـــه واســـتخدامه وطريقة بنائه، حيث يستخدم المحلل نموذج السلوك ليستطيع فهم الجوانب الديناميكية مـــن النظام. وفي هذه الوحدة سيتم التركيز على إنشاء نماذج الـــسلوك باســـتخدام مخططـــات التفاعل (مخططات التتابع والتعاون) ومخططات الحالة. وسنرى أن هناك نـــوعين مـــن نموذج السلوك: النوع الأول: يستخدم لوصف تفاصيل العمليات الموجودة في النظام التي يتضمنها نموذج حالات الاستخدام، والمخططات التفاعلية بنوعيها المتسلسلة والمتعاونـــة المستخدمة لتصميم نموذج سلوك النظام. أما النوع الثاني: فيستخدم لوصف التغيرات التي تحدث على البيانات.

7. إجابات التدريبات

تدريب (1)

Class Name: Student	**ID: 01**	Type: **General**
Description: this class represents a general student class		
Responsibilities	**Collaborations**	
Login to registration system	String	
Register for Courses	Network and Connection	
Know the Offered Courses		

Attributes:
Number, Name, Year, Tel-No
Relationships:
Courses
Instructors
Other associations: PartTime-Student

Class Name: Course	**ID: 02**	**Type: General**
Description: this class represents a general Course class		
Responsibilities	**Collaborations**	
Classes Descriptions	String	
Modify Prerequisite	Network and Connection	

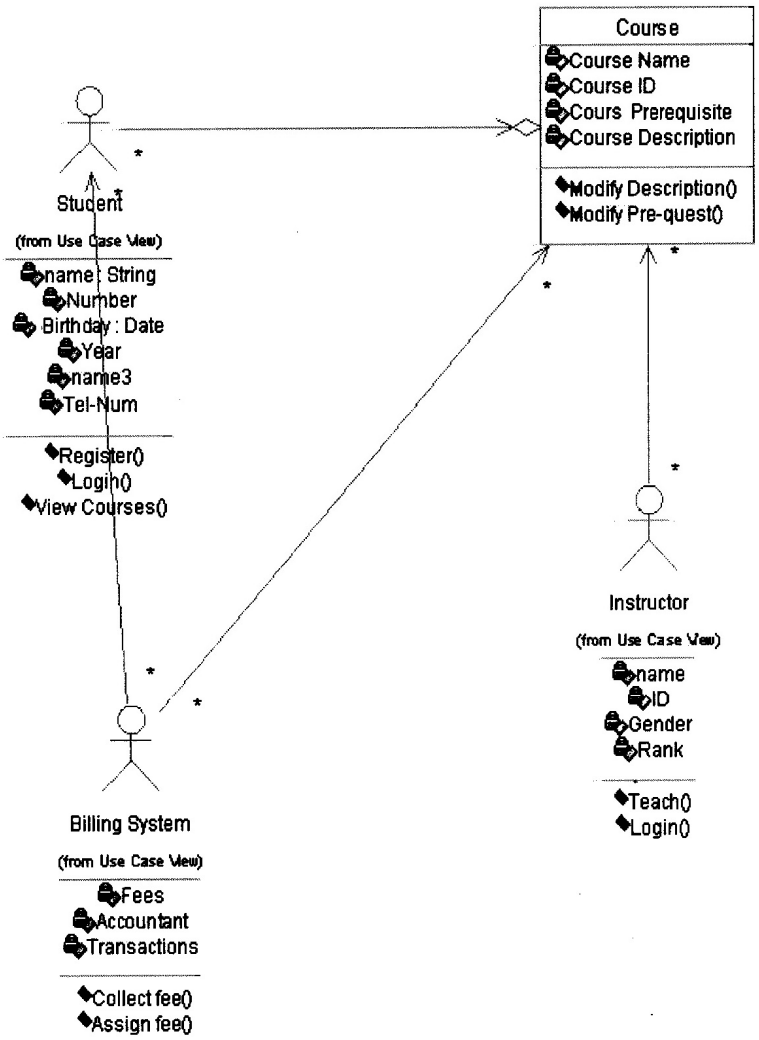
Attributes:
Course Name, Course ID, Course Prerequisite, Course Description
Relationships:
Students
Instructors
Other associations:

Class Name: Instructor	**ID: 03**	**Type: General**
Description: this class represents a general Instructor class		
Responsibilities	**Collaborations**	
Teach Courses	String	
Login to the system	Network and Connection	

Attributes:
Name, Gender ,ID, Rank
Relationships:
Billing System Course
Other associations:

Class Name: Billing System	**ID: 04**	**Type: General**
Description: this class represents a general Billing System class		
Responsibilities	**Collaborations**	
Collect fees	String	
Assign fees for courses	Network and Connection	

Attributes:
Fees, Accountant, Transactions
Relationships:
Student
Course
Other associations:


Course
Course Name
Course ID
Cours Prerequisite
Course Description
Modify Description()
Modify Pre-quest()
Student
(from Use Case View)
name : String
Number
Birthday : Date
Year
name3
Tel-Num
Register()
Login()
View Courses()
Instructor
(from Use Case View)
name
ID
Gender
Rank
Teach()
Login()
Billing System
(from Use Case View)
Fees
Accountant
Transactions
Collect fee()
Assign fee()

8. مسرد المصطلحات

- **الأصناف Classes:** عبارة عن قالب عام نستخدمه لتصنيع الكينونات (Objects).
- **الأصناف الرئيسة Super Class:** هي الأصناف التي تورث الصفات والعمليات لأصناف أخرى.
- **الأصناف الفرعية Sub Class:** هي الأصناف التي ترث من غيرها.
- **بطاقات الصنف- المسؤولية- التعاون Class-Responsibility-Cooperation:** عبارة عن وسيلة لتحصيل البيانات من أطراف النظام وتستعمل كذلك لتوثيق المسؤوليات والتعاونيات المناطة بكل صنف من أصناف النظام.
- **التخصيص Specialization:** هي العلاقة من الصنف الرئيس إلى الصنف الفرعي.
- **التعميم Generalization:** هي العلاقة من الصنف الفرعي إلى الرئيس.
- **الصفات Attributes:** إن صفة الصنف هي عبارة عن قطعة معلومات مفيدة ترتبط بوصف جزء معين من هذا الصنف. والصفات تحتوي عادة على المعلومات التي يعتقد المحلل أو المستخدم بأن على النظام الاحتفاظ بها.
- **العلاقات Relationships:** العلاقات هي عبارة عن روابط بين الأصناف، وهي ثلاثة أنواع: علاقات التعميم، وعلاقات التجميع، وعلاقات التشارك.
- **الطرق Methods:** الطريقة هي ذلك الجزء من الصنف الذي يعرف فيه تصرف الصنف، وتكون عادة متعلقة بشكل كبير بطور التطبيق. ويمكن تعريف الطريقة بأنها الأفعال التي تنفذها كينونات مشتقة من صنف معين.

10. المراجع

1. Aktas, Z., Structured Analysis and Design of Information Systems, Prentice-Hall, 1987.
2. Baudoin, C. and Hollowell, G., Realizing The Object-Oriented Life Cycle, Prentice- Hall, 1996.
3. Booch, G., Rumbaugh, J., and Jacobson, I., The Unified Modeling Language User Guide, Addison- Wesley, 1999.
4. Dennis, A. et al, Systems Analysis & Design: An Object-Oriented Approach with UML. John Wiley & Sons Inc., 2002.
5. Dewitz, S., Systems Analysis and Design and the Transition to Objects, McGraw-Hill, 1996.
6. Edwards, P., Systems Analysis, Design, and Development with Structured Concepts, Rinehart and Winston, 1999.
7. Henderson, P., Object-Oriented Specification and Design with C++, McGraw- Hill, 1993.
8. Hoffer, J. et al., Modern Systems Analysis & Design, 2nd Ed., Addison-Wesley, 1999.
9. Kendal, K., and Kendal, J., Systems Analysis and Design, 5th Ed. 2002.
10. Kruchten, P., The Rational Unified Process: An Introduction. 2nd Ed., Addison- Wesley, 2000.
11. Lejk, M. and Deeks. An Introduction to Systems Analysis Techniques, 2nd Ed., Pearson Education Limited, 2002.
12. Maciaszek, A., Requirements Analysis and System Design, Developing Information Systems with UML, Addison- Wesley, 2001.
13. Quatrani, T., Visual Modeling with Rational Rose 2000 and UML, Addison- Wesley, 2000.
14. Satzinger, J., Systems Analysis and Design, 2nd Ed., Thomson Learing, 2002.
15. Silver, G. and Silver, M., Systems Analysis and Design, Addison-Wesley, 1996.
16. http://www.psut.edu.jo / computer science/ staff/saleh abu-soud
17. www.rational.com
18. www.omg.com

الوحدة السادسة

نمذجة سلوك النظام

System Behavior Modeling

محتويات الوحدة

الموضوع	الصفحة
1. المقدمـــة	139
1.1 تمهيد	139
2.1 أهداف الوحدة	139
3.1 أقسام الوحدة	140
4.1 القراءات المساعدة	140
5.1 ما تحتاج إليه لدراسة الوحدة	140
2. نماذج السلوك Behavioral Models	141
3. مخططات التفاعل Interaction Diagrams	142
1.3 الكينونات Objects، والعمليات Operations، والرسائل Messages	142
2.3 مخططات التسلسل أو التتابع Sequence Diagrams	143
3.3 مخططات التعاون Collaboration Diagrams	146
4. مخططات الحالة State Chart Diagrams	148
5. الخلاصة	150
6. لمحة عن الوحدة الدراسية السابعة	150
7. إجابات التدريبات	151
8. مسرد المصطلحات	154
9. المراجع	155

1. المقدمة

1.1 تمهيد

أهلاً بك – عزيزي الدارس– إلى الوحدة السادسة من مقرر "1380 تحليل الأنظمة وتصميمها"، وهي بعنوان "نمذجة سلوك النظام".

تستخدم نماذج حالات الاستخدام، كما تم بيانه خلال الوحدات الدراسية السابقة، لوصف التصرف الخارجي للنظام ولتوضيح وظيفته، بينما يستخدم النموذج الهيكلي لرسم واجهة ثابتة عن معلومات النظام.

في هذه الوحدة سنقوم ببيان كيفية وصف السلوك الديناميكي للنظام من الداخل، حيث يتمكن المحلل من فهم ووصف التصرف الداخلي للنظام والصورة المتحركة لمعلوماته، وذلك عن طريق مخططات تسمى بمخططات التفاعل (Interaction Diagrams) التي تحتوي على العديد من الأنواع مثل مخططات التسلسل أو التتابع (Sequence Diagrams) التي تستخدم لوصف تفاصيل العمليات الموجودة في النظام التي كان قد تم توضيحها باستخدام نموذج حالات الاستخدام، ومخططات التعاون (Collaboration Diagrams) التي تستخدم لوصف التغيرات التي تحدث على البيانات، بالإضافة إلى مخططات الحالة (State Chart Diagrams) التي تستخدم في حالة وصف أنظمة الوقت الحقيقي. ويستخدم هذا النوع من النمذجة خلال مرحلة التحليل.

في هذه الوحدة سيتم كذلك– عزيزي الدارس– التركيز على إنشاء نماذج السلوك باستخدام مخططات التفاعل بنوعيها: مخططات التسلسل، ومخططات التعاون، بالإضافة إلى مخططات الحالة.

2.1 أهداف الوحدة

يتوقع منك، عزيزي الدارس، بعد الانتهاء من دراسة هذه الوحدة أن تكون قادراً على أن:

1. تبين أهمية نمذجة تصرف النظام.
2. تحدد عناصر وكيفية بناء مخطط التتابع.
3. تبين أهمية مخطط التعاون.
4. تتمكن من بناء وتصميم مخطط الحالة.

3.1 أقسام الوحدة

تقسم الوحدة إلى ثلاثة أقسام رئيسة:

القسم الأول: يبين نمذجة سلوك النظام وهذا يغطي الهدف الأول، أما القسم الثاني: فيناقش مخططات التفاعل التي تشمل مخططات التتابع ومخططات التعاون وكيفية بنائها، وهذا يغطي الهدفين الثاني والثالث، والقسم الثالث وهو الأخير يغطي استخدام وبناء مخططات الحالة، وهذا يغطي الهدف الرابع.

4.1 القراءات المساعدة

أنصحك- عزيزي الدارس- بالاطلاع على المراجع التالية لما تحويه من معلومات مفيدة تتعلق بموضوع هذه الوحدة:

1. Baudoin, C. and Hollowell, G., Realizing The Object-Oriented Life Cycle, Prentice- Hall, 1996.
2. Booch, G., Rumbaugh, J., and Jacobson, I., The Unified Modeling Language User Guide, Addison- Wesley. 1999.
3. Dennis, A. et al, Systems Analysis & Design: An Object-Oriented Approach with UML. John Wiley & Sons Inc.، 2002.
4. Dewitz,، S., Systems Analysis and Design and the Transition to Objects, McGraw-Hill, 1996.

5.1 ما تحتاج إليه لدراسة الوحدة

عزيزي الدارس: قبل أن تبدأ بدراسة هذه الوحدة تأكد من أنك قمت بتهيئة المكان الهادئ المناسب للدراسة مما يساعدك على التركيز والتفكير العميق. ثم احرص على تتبع الإرشادات التي توجهك لدراسة الوحدة، وحاول بجدية الإجابة عن أسئلة التقويم الذاتي، وحل التدريبات، لأنها تساعدك على مراجعة موضوعات الوحدة، وتعمق معلوماتك حولها. أنصحك- عزيزي الدارس- كذلك بمحاولة تطبيق المخططات الواردة خلال هذه الوحدة وغيرها من المخططات على برمجية Rational، لما لها من أثر على تعميق فهمك لمادة هذه الوحدة.

2. نماذج السلوك Behavioral Models

عند تحليل النظام، ولفهم النظام بتفاصيله كافة، ينبغي على المحلل أن يأخذ بعين الاعتبار النواحي الديناميكية المتمثلة بسلوك النظام، بالإضافة إلى النواحي غير الديناميكية المتمثلة بالكينونات. إن تحليل النظام المعتمد على الكينونة يعتمد على مجموعة من حالات الاستخدام المدعومة بمجموعة من الكينونات المتعاونة، وبذلك يقوم محلل النظم بتقليص الفجوة المعنوية بين مجموعة الكينونات في العالم الحقيقي ومجموعة الكينونات المنمذجة.

إن أحد أهم الأهداف لنمذجة السلوك هو إظهار الكينونات التي تتعاون بعضها مع بعض لتدعيم حالات الاستخدام، بينما يقوم نموذج بناء النظام بتمثيل الكينونات والعلاقات بينها. إن نموذج السلوك يرسم المنظور الداخلي لمنطق العمل الذي تبينه حالات الاستخدام، حيث يتم تمثيل هذه العملية على شكل تفاعل بين الكينونات التي تتعاون بعضها مع بعض لتدعم حالات الاستخدام وذلك باستخدام المخططات التفاعلية (التتابعية والتعاونية)، كما يمكن عرض تأثير مجموعة من حالات الاستخدام على الكينونات في النظام من خلال استعمال مخططات الحالة State Chart Diagrams .

3. مخططات التفاعل Interaction Diagrams

إن أحد أهم الفروقات بين مخطط الأصناف ومخططات التفاعل، بالإضافة إلى الفرق الواضح الذي يتمثل في أن الأول يستخدم لوصف البنية والآخر يصف السلوك، هو أن مخطط الأصناف يركز على الصنف بينما الآخر يركز على الكينونة نفسها.

وسنقوم في الفقرات التالية- عزيزي الدارس- بشرح الكينونات والعمليات والرسائل بالإضافة إلى مخططات التتابع والتعاون.

1.3 الكينونات Objects، والعمليات Operations، والرسائل Messages

من المعروف أن الكينونة ما هي إلا مشتقه لحظية من صنف معين، وهي عادة ما تمثل شخصاً أو مكاناً أو حدثاً أو شيئاً حقيقياً نريد الاحتفاظ بمعلومات عنه. إن مجموعة الأصناف في نظام التسجيل في الجامعة- على سبيل المثال- تحتوي على صنف الطالب وصنف المحاضر وصنف المواد....إلخ. ومن ناحية أخرى فإن الطالب الحقيقي مثل: أحمد وسعيد وخليل ما هم إلا مشتقات لحظية من صنف طالب تسمى بالكينونات. وكل كينونة تحتوي على صفات تصف معلوماتها كاحتواء كينونة الطالب للصفات: اسم الطالب، ورقم هاتفه، ورقمه الجامعي.

وكل كينونة تحتوي كذلك على مجموعة من التصرفات التي تُعرَّف بالعمليات التي تعبر عن الفعل الذي تستطيع الكينونة القيام به، كأن يقوم الطالب بإجراء عملية التسجيل وعملية السحب.

وتستطيع كل كينونة إرسال الرسائل واستقبالها، والرسالة هي عبارة عن معلومات ترسلها كينونة إلى أخرى لإخبارها بالقيام بتصرف معين. كما يمكن كذلك تعريف الرسالة بطريقة أخرى بأنها عبارة عن استدعاء أو أمر يوجه من كينونة ما إلى كينونة أخرى. ويستطيع الطالب- على سبيل المثال- الانضمام إلى نظام التسجيل بإرسال رسالة طلب تسجيل للمسجل ومن ثم يتم التسجيل بإعطاء رقم جديد للطالب.

أسئلة التقويم الذاتي (1)

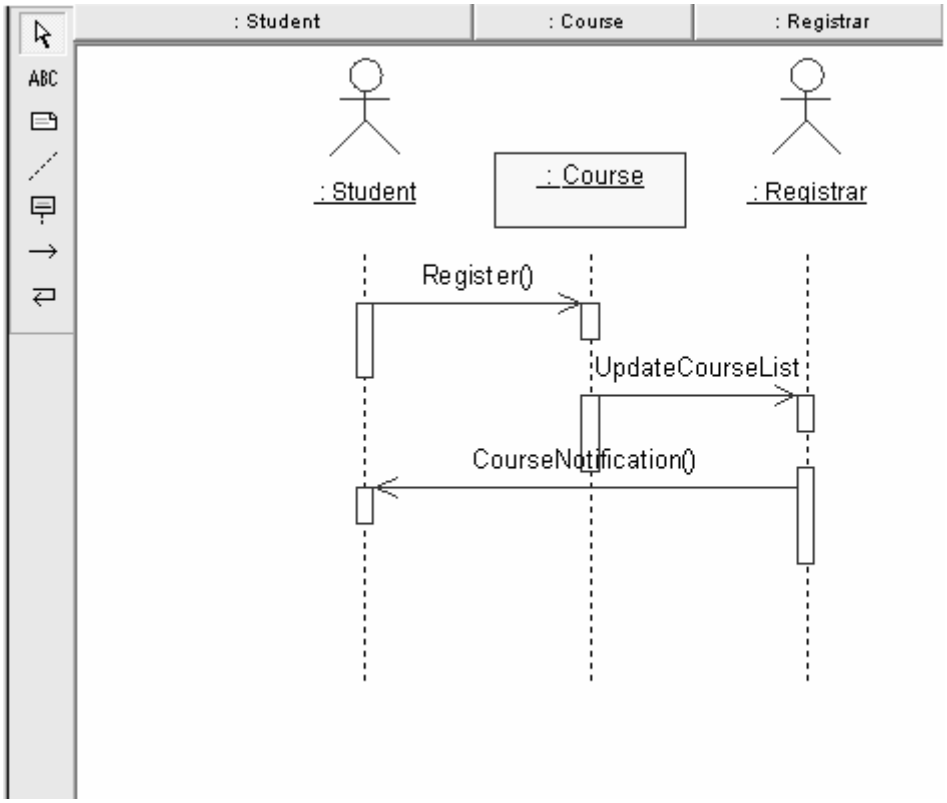
وضح المقصود بكل من الكينونات، والعمليات، والرسائل.

2.3 مخططات التسلسل أو التتابع Sequence Diagrams

يعتبر مخطط التسلسل أو التتابع أحد أنواع مخططات التفاعل، حيث إنه يوضح الكينونات التي تشترك في حالات الاستخدام، كما يوضح مجرى الرسائل على فترة من الزمن من كينونة إلى أخرى ومن كينونة إلى حالة استخدام. وإن مخطط التتابع هو عبارة عن مخطط ديناميكي يوضح تتابع الرسائل التي يتم تمريرها بين الكينونات بشكل تفاعلي كونه مخطط يعتمد على الترتيب أكثر من التفاعلات التي تحدث بين الكينونات. ويعتبر هذا النوع من المخططات مفيداً جداً لفهم حقيقي وبشكل واقعي للطلبات المتاحة في أعقد حالات الاستخدام. وإن مخططات التتابع هي مخططات ذات شكل عام يتعرض لجميع سيناريوهات حالات الاستخدام، لكن وصف جميع السيناريوهات لحالات الاستخدام عمل غير واقعي، لذلك على المحلل أن يقوم باختيار مجموعة سيناريوهات ذات علاقة بعضها مع بعض توضح فكرة معينة على امتداد فترة من الزمن، لذلك تتحقق فكرة التتابع على الزمن بشكل واقعي.

وتستخدم مخططات التتابع في مرحلتي التحليل والتصميم حيث إنها تحتوي في مرحلة التصميم على مفاهيم كثيرة مثل قواعد البيانات وواجهات الاستخدام (User Interfaces) وجميع المواصفات التقنية.

والشكل (1) يبين مخططاً تتابعياً من نظام التسجيل الجامعي حيث يوضح الخطوات والرسائل اللازمة لتسجيل مادة وحذفها وتعديلها. ويتضح لنا- عزيزي الدارس- من الشكل (1) أن المتفاعلين والكينونات التي تشارك في التتابع توضع على أعلى قائمة المخطط باستخدام رمز المتفاعلين. وتجدر الإشارة هنا إلى أن ترتيب المتفاعلين ليس بالضرورة أن يكون ذا معنى، لكن يستحسن أن ترتب بشكل منطقي كأن يكون الترتيب حسب ترتيبهم في المشاركة في الحدث.



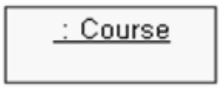
الشكل (1) مخطط تتابع لتسجيل مادة وتعديل قائمة المواد

لاحظ – عزيزي الدارس– أن كل كينونة في المخطط تستخدم اسم الصنف الذي اشتقت منه، ولاحظ أيضاً أن هناك خطاً متقطعاً بشكل عامودي تحت كل متفاعل أو كينونة يرمز إلى خط الحياة لكل منها. ويمكن استخدام مربع رفيع يسمى تركيز التحكم "وقت التنفيذ" وهو يرسم فوق خط الحياة ويرمز إلى الفترة التي يكون فيها المتفاعل أو الكينونة في حالة استقبال إذا أرسل بعض الرسائل، حيث تُعبر الرسالة عن التواصل بين الأصناف المتحاورة ويتم التعبير عنها باستخدام سهم أفقي ذي متجه يصل بين مراحل الاتصال ويحدد اتجاه الرسالة. ويمكن أيضاً إرسال بعض المعاملات (Parameters) عند استدعاء الرسالة حيث يتم وضع هذه المعاملات بين أقواس تلي اسم الرسالة. وترتب الرسائل من أعلى إلى أسفل المخطط، أي أن الرسالة التي تحل أعلى تحدث قبل الرسالة التي تحل أسفل منها.

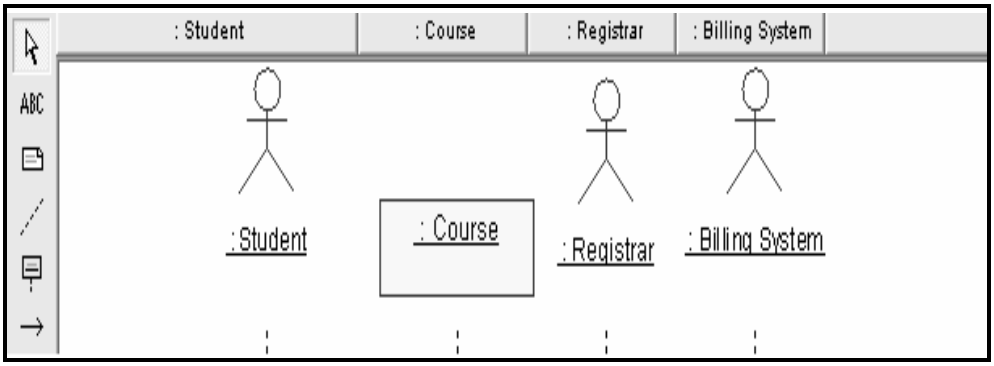
أسئلة التقويم الذاتي (2)

اذكر الفرق بين الخط العامودي المتقطع المستخدم في مخطط التتـــابع والمربـــع الرفيع الذي يرسم فوقه.

الجدول (1) الرموز المستخدمة في مخطط التسلسل والهدف من استخدامها.

الرمز	التعريف
	المتفاعل
	كينونة من صنف معين
	خط الحياة: الذي يعرف وجود الكينونة
	وقت التنفيذ: يحدد أن الكينونة تنفذ عمل ما
	رسالة من كينونة إلى أخرى ومعاملاتها

الشكل (2) يبين مجموعة المتفاعلين والكينونات التي نحتاجها في نظام التسجيل الجامعي.



الشكل (2) مجموعة من المتفاعلين والكينونات التي نحتاجها في نظام التسجيل الجامعي.

تدريب (1)

ارسم مثالاً لمخطط تتابع من نظام التسجيل الجامعي لتسجيل مادة ودفع رسومها الجامعية.

3.3 مخططات التعاون Collaboration Diagrams

المخططات التعاونية تشبه المخططات التتابعية من حيث إن النوعين يبينان الناحية الديناميكية للنظام. وهي عبارة عن مخططات كينونة تظهر تراسل الرسائل بدلاً من إظهار العلاقات التجميعية والتعميمية والتشاركية، وتعتبر هذه المخططات مفيدة لإظهار نشاط العمليات، وهي تعتبر مكافئة لمخططات التتابع ولكن بفارق جوهري وهو أن لها القدرة على إظهار تدفق تفاعل مجموعة من الكينونات بينما تركز مخططات التتابع على الترتيب الزمني لهذا التدفق. لذلك إذا كان الاهتمام منصباً على فهم التفاعل ما بين الكينونات التعاونية يجب استخدام مخططات التعاون، أما إذا كان الاهتمام للتوقيت الزمني فيجب استخدام مخططات التتابع. تجدر الإشارة هنا – عزيزي الدارس – إلى أنه في بعض الحالات نحتاج إلى استخدام كلا المخططين للحصول على فهم عميق لديناميكية النظام.

ومن أوجه التشابه بينهما أن النوعين يحتويان على متفاعلين وكينونات، حيث تتعاون فيما بينها لتنفيذ حالات الاستخدام.

الاختلاف

أما وجه الاختلاف بين مخططات التعاون والتتابع فهو أن الأولى لا توضح تبرير إرسال الرسائل بينما تستطيع مخططات التتابع توضيح ذلك بشكل اختياري. ويتم توضيح التشارك بين المتفاعل والكينونة عن طريق خطوط تصل فيما بينها، وتكون الرسائل موضحة فوق هذه الخطوط جنباً إلى جنب مع استخدام أسهم لتوضيح اتجاه هذه الرسائل، أما بالنسبة إلى ترتيب وتتابع الرسائل فيتم توضيحه باستخدام رقم يلحق باسم الرسالة.

تدريب (2)

ارسم مخططاً تعاونياً لبيان التفاعل بين الطالب، والمادة، ونظام الدفع في نظام التسجيل

4. مخططات الحالة State Chart Diagrams

بعض الأصناف في مخططات الأصناف تمثل مجموعة من الكينونات الديناميكية التي تمثل مجموعة معقدة من الحالات، كأن يكون الطالب مسجلاً، أو غير مسجل، أو منسحب، أو منتظم، وبدوام جزئي، ودافع للرسوم، أو غير دافع، حيث يمكن تمثيل هذه الحالات باستخدام مخطط يسمى بمخطط الحالة.

إن مخطط الحالة عبارة عن نموذج ديناميكي يظهر مجموعة الحالات التي يمكن أن تحويه كينونة معينة خلال فترة حياتها. وتجدر الإشارة هنا – عزيزي الدارس– إلى أن استعمال مخططات الحالة لجميع الكينونات حل غير عملي، ويفضل استخدامها للكينونات المعقدة من ناحية التصميم، حيث يستخدم هذا النوع من المخططات لتسهيل التعامل مع هذه الكينونات المعقدة.

إن مخطط الحالة يظهر عادة حالات الكينونة المختلفة، كما يبين السبب الذي أدى بالكينونة للتغير من حالة إلى أخرى، حيث يسمى هذا السبب بالحدث، وحيث يعرف بأنه شيء ما قد يحدث في وقت محدد ويغير في قيم صفات الكينونة مما يؤدي إلى تغير حالة الكينونة، فعندما ينسحب طالب معين من مادة فإنه يغير من حالة الطالب "مسجل" فتصبح حالته الجديدة "منسحب".

وبالمقارنة مع المخططات التفاعلية، فإن مخططات الحالة تستخدم إذا كان محلل النظم مهتماً بفهم نواحٍ ديناميكية لصنف واحد وكيفية تطوير الكينونات لهذا الصنف على مر الزمن وليس مهتما بحالة استخدام معينة.

وتعرف حالة الكينونة على أنها قيمة إحدى الصفات في هذه الكينونة في لحظة معينة من الزمن، وتسمى هذه الصفة بالصفة المؤثرة حسب أهميتها، فحالة الطالب الأكاديمية في مادة معينة، على سبيل المثال، إما أن تكون "ناجح" أو "راسب" أو "منسحب" بناء على علامته في المادة. وتجدر الإشارة هنا إلى أن هناك صفات لا يؤثر تغييرها في تغيير حالة الكينونة، كرقم هاتف الطالب.

أسئلة التقويم الذاتي (3)

عرف الصفات المؤثرة والصفات غير المؤثرة وأذكر الفرق بينها.

يبين الجدول (2) المصطلحات المستخدمة في مخططات الحالة. حيث يرمز للحالة، كما هو موضح في الجدول، بمربع ذي زوايا دائرية مع عنوان وصفي داخل هذا المربع يوضح طبيعة الحالة. وهناك استثناءان لهذه القاعدة، الأول: في حالة البداية حيث يرمز لها بدائرة صغيرة مظللة من الداخل، والثاني: في حالة النهاية حيث يرمز لها بدائرة مظللة محاطة بدائرة أخرى. وتستخدم الأسهم للربط بين الحالات ولتمثيل الانتقال بين تلك الحالات. وكل سهم يتميز باسم حدث مناسب ولديه معامل أو شرط يستخدم لمزيد من الإيضاح.

الجدول (2): المصطلحات المستخدمة في مخطط الحالة

الرمز	التعريف
	حالة
	انتقال
	حالة نهاية
	حالة بداية

تدريب (3)

ارسم مخطط الحالة لعملية التسجيل لطالب جديد في الجامعة.

5. الخلاصة

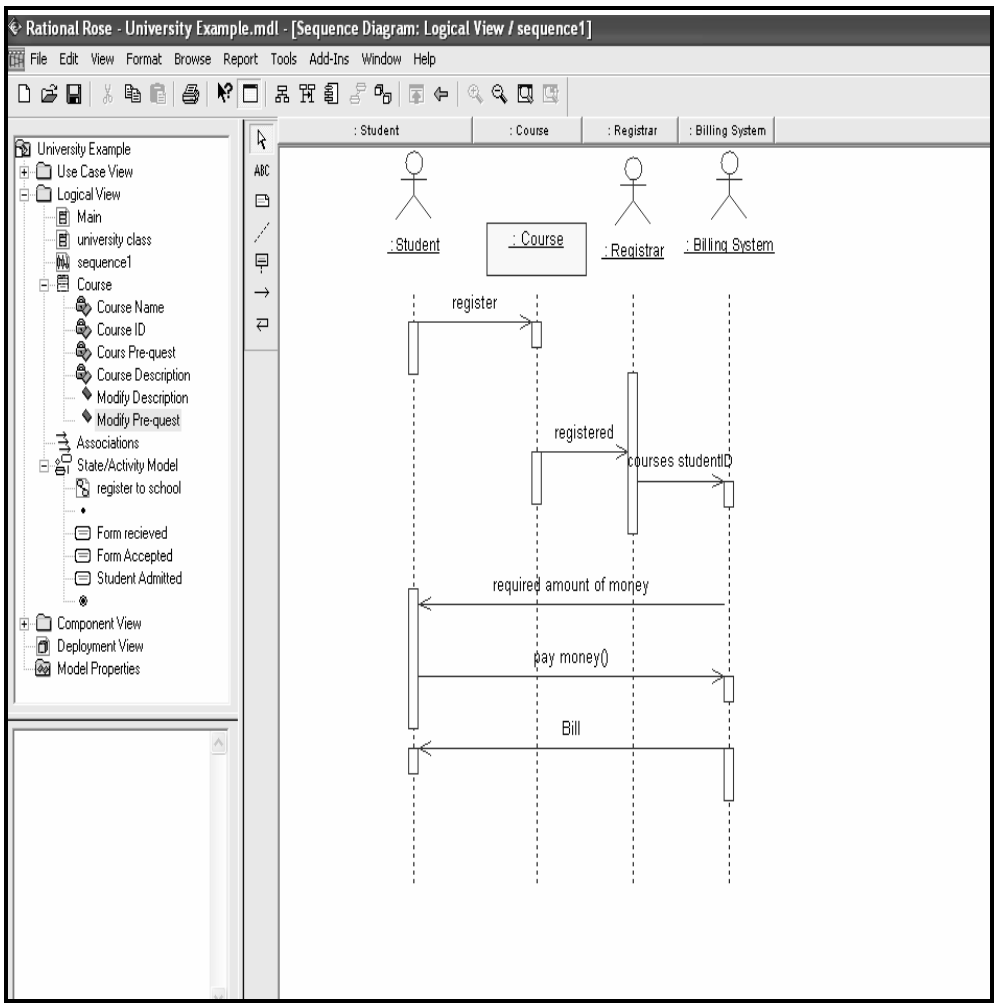
لقد تعلمت– عزيزي الدارس– في هذه الوحدة كيفية بناء نموذج لوصف سلوك النظام للحصول على فهم عميق لكيفية تفاعل وديناميكية النظام وذلك باستخدام التفاعلية بنوعيها مخططات التتابع ومخططات التعاون إضافة إلى مخططات الحالة.

6. لمحة عن الوحدة الدراسية السابعة

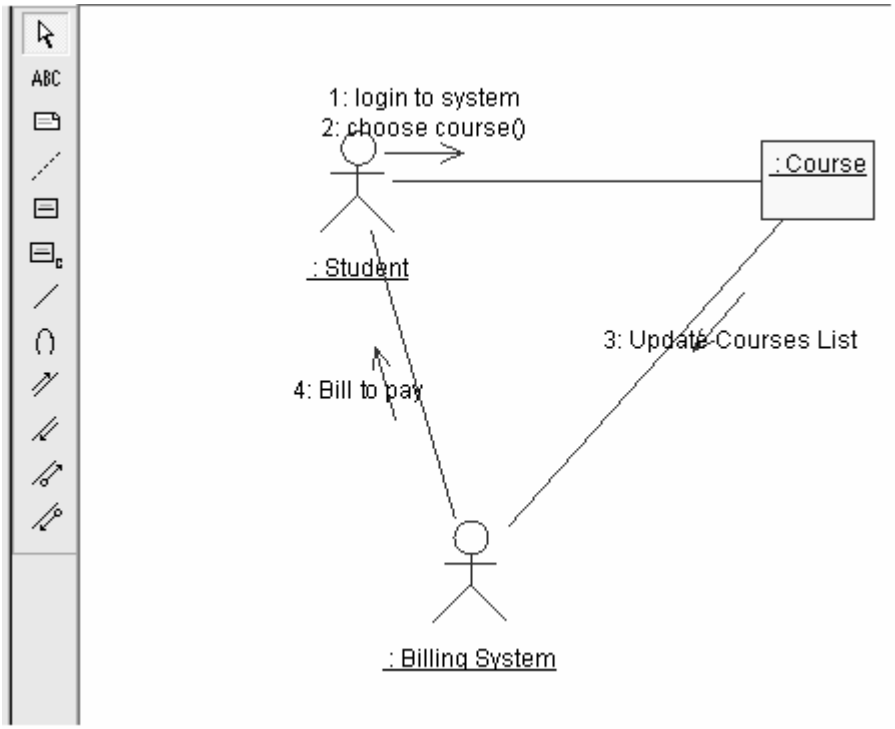
تقسم الوحدة السابعة إلى ثلاثة أقسام، يبين القسم الأول: عملية التحول من التحليل إلى التصميم، بينما يناقش القسم الثاني: معماريات النظام المختلفة مثل: المعمارية المعتمدة على الخادم، والمعمارية المعتمدة على المخدوم، والمعمارية المعتمدة على الخادم/ المخدوم، وطبقات الخادم/ المخدوم، وأخيراً معمارية الكينونات الموزعة ويبين الفرق بينها وميزات كل منها. أما القسم الثالث والأخير: فيشرح تصميم البنية التحتية للنظام، وطرق تحديد مواصفات المعدات والبرمجيات المطلوبة للنظام.

7. إجابات التدريبات

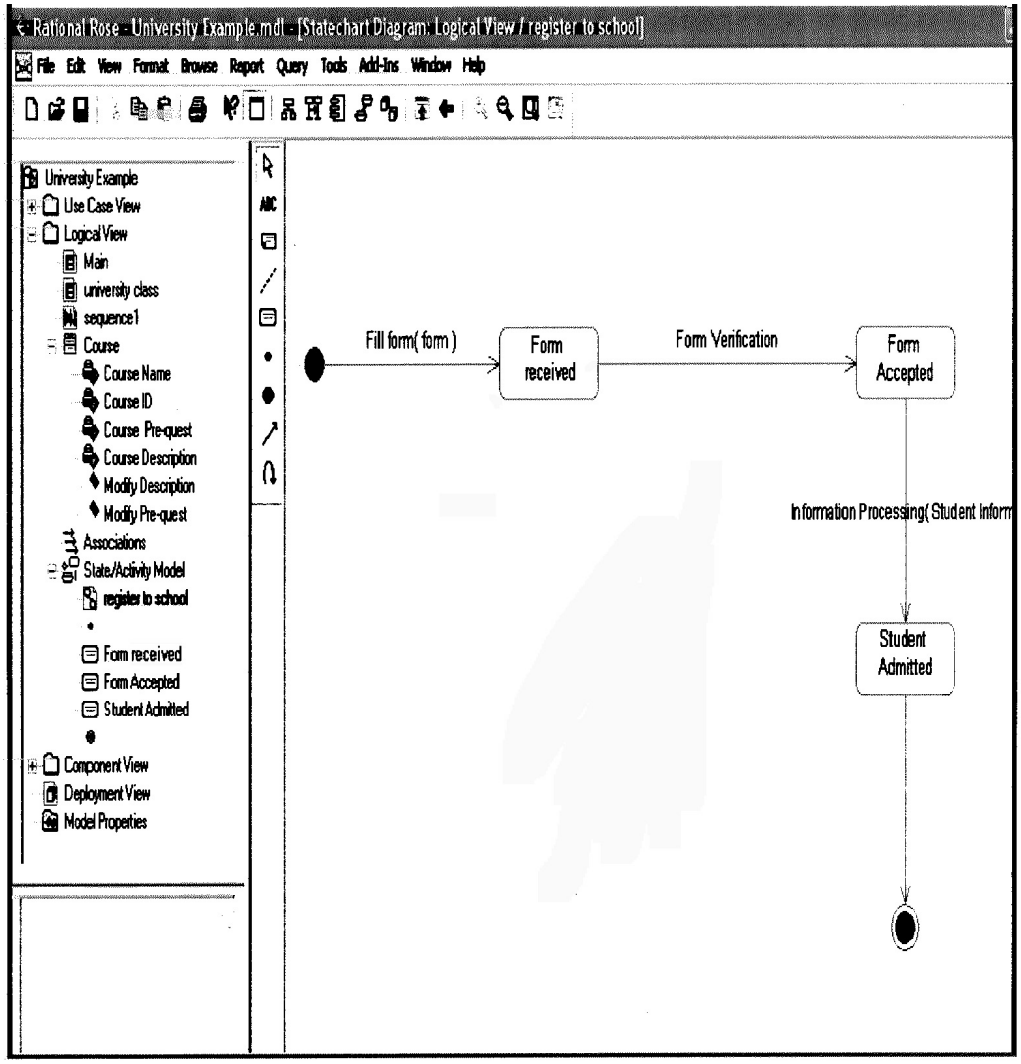
تدريب (1)



تدريب (2)


ABC
1: login to system
2: choose course()
: Course
: Student
3: Update Courses List
4: Bill to pay
: Billing System

تدريب (3)


Rational Rose - University Example.mdl - [Statechart Diagram: Logical View / register to school]
File Edit View Format Browse Report Query Tools Add-Ins Window Help
University Example
Use Case View
Logical View
Main
university class
sequence1
Course
Course Name
Course ID
Course Pre-quest
Course Description
Modify Description
Modify Pre-quest
Associations
State/Activity Model
register to school
Form received
Form Accepted
Student Admitted
Component View
Deployment View
Model Propeties
Fill form(form)
Form received
Form Verification
Form Accepted
Information Processing(Student Inform
Student Admitted

8. مسرد المصطلحات

- **مخطط التسلسل أو التتابع Sequence Diagram:** يعتبر مخطط التسلسل أو التتابع أحد أنواع مخططات التفاعل، حيث إنه يوضح الكينونات التي تشترك في حالات الاستخدام، كما يوضح مجرى الرسائل على فترة من الزمن من كينونة إلى أخرى ومن كينونة إلى حالة استخدام.
- **مخطط التعاون Collaboration Diagram:** تشبه المخططات التتابعية من حيث إن النوعين يبينان الناحية الديناميكية للنظام. وهي عبارة عن مخططات كينونة تظهر تراسل الرسائل بدلاً من إظهار العلاقات التجميعية والتعميمية والتشاركية، وتعتبر هذه المخططات مفيدة لإظهار نشاط العمليات، وهي تعتبر مكافئة لمخططات التتابع ولكن بفارق جوهري وهو أن لها القدرة على إظهار تدفق تفاعل مجموعة من الكينونات بينما تركز مخططات التتابع على الترتيب الزمني لهذا التدفق.
- **مخطط الحالة State Chart Diagram:** إن مخطط الحالة عبارة عن نموذج ديناميكي يظهر مجموعة الحالات التي يمكن أن تحويه كينونة معينة خلال فترة حياتها. وإن استعمال مخططات الحالة لجميع الكينونات حل غير عملي ويفضل استخدامها للكينونات المعقدة من ناحية التصميم، حيث يستخدم هذا النوع من المخططات لتسهيل التعامل مع هذه الكينونات المعقدة.
- **نمذجة السلوك Behavioral Modeling:** هو إظهار الكينونات التي تتعاون بعضها مع بعض لتدعيم حالات الاستخدام، حيث يقوم نموذج السلوك برسم المنظور الداخلي لمنطق العمل الذي تبينه حالات الاستخدام، حيث يتم تمثيل هذه العملية على شكل تفاعل بين الكينونات التي تتعاون بعضها مع بعض لتدعم حالات الاستخدام وذلك باستخدام المخططات التفاعلية (التتابعية والتعاونية).

9. المراجع

1. Aktas, Z., Structured Analysis and Design of Information Systems, Prentice-Hall, 1987.
2. Baudoin, C. and Hollowell, G., Realizing The Object-Oriented Life Cycle, Prentice- Hall, 1996.
3. Booch, G., Rumbaugh, J., and Jacobson, I., The Unified Modeling Language User Guide, Addison- Wesley, 1999.
4. Dennis, A. et al, Systems Analysis & Design: An Object-Oriented Approach with UML. John Wiley & Sons Inc., 2002.
5. Dewitz, S., Systems Analysis and Design and the Transition to Objects, McGraw-Hill, 1996.
6. Edwards, P., Systems Analysis, Design, and Development with Structured Concepts, Rinehart and Winston, 1999.
7. Henderson, P., Object-Oriented Specification and Design with C++, McGraw- Hill, 1993.
8. Hoffer, J. et al., Modern Systems Analysis & Design, 2nd Ed., Addison-Wesley, 1999.
9. Kendal, K., and Kendal, J., Systems Analysis and Design, 5th Ed., 2002.
10. Kruchten, P., The Rational Unified Process: An Introduction. 2nd Ed., Addison Wesley, 2000.
11. Lejk, M. and Deeks. An Introduction to Systems Analysis Techniques, 2nd Ed., Pearson Education Limited, 2002.
12. Maciaszek, A., Requirements Analysis and System Design, Developing Information Systems with UML, Addison Wesley, 2001.
13. Quatrani, T., Visual Modeling with Rational Rose 2000 and UML, Addison- Wesley, 2000.
14. Satzinger, J., Systems Analysis and Design, 2nd Ed., Thomson Learing, 2002.
15. Silver, G. and Silver, M., Systems Analysis and Design, Addison-Wesley, 1996.
16. http://www.psut.edu.jo / computer science/ staff/saleh abu-soud
17. www.rational.com
18. www.omg.com

الوحدة السابعة

تصميم معمارية النظام

System Architecture Design

محتويات الوحدة

الموضوع	الصفحة
1. المقدمـــة	161
1.1 تمهيد	161
2.1 أهداف الوحدة	161
3.1 أقسام الوحدة	162
4.1 القراءات المساعدة	162
5.1 ما تحتاج إليه لدراسة الوحدة	162
2. التحول من التحليل إلى التصميم Moving from Analysis to Design	163
3. معماريات النظام System Architecters	164
1.3 المعمارية المعتمدة على الخادم Server-based Architecture	164
2.3 المعمارية المعتمدة على المخدوم Client-based Architecture	165
3.3 معمارية الخادم/ المخدوم Client-Server Architecture	167
4.3 طبقات الخادم/ المخدوم Client-Server Tiers	168
5.3 معمارية الكينونات الموزعة Distributed Objects Architecture	171
4. متطلبات بناء النظام	173
1.4 تصميم البنية التحتية للنظام Infrastructure Design	173
2.4 تحديد مواصفات البرمجيات والمعدات Software and Hardware Specifications	173
5. الخلاصة.....	175
6. لمحة عن الوحدة الدراسية الثامنة.....	175
7. إجابات التدريبات.....	176
8. مسرد المصطلحات.....	177
9. المراجع.....	178

1. المقدمة

1.1 تمهيد

أهلاً بك- عزيزي الدارس- إلى الوحدة السابعة من مقرر" 1380 تحليل الأنظمة وتصميمها"، وهي بعنوان" تصميم معمارية النظام".

يعتبر تصميم معمارية النظام من أهم خطوات مرحلة التصميم، إذ إنه يعتبر التخطيط للبرمجيات والمعدات والبنية التحتية للاتصال والأمن والحماية والدعم الخارجي للنظام الجديد.

يتكون النظام عادة من ثلاثة أجزاء: البرامج التطبيقية (Application Programs) التي تقوم بمهام معالجة منطق النظام، وقاعدة البيانات (Database) لتخزين البيانات، وواجهات المستخدم (User Interface) التي تسمح للمستخدم بالتخاطب والتفاعل مع النظام. وفي الواقع إن توزيع هذه الأجزاء يحدد طبيعة معمارية النظام، فإذا كانت جميعها على جهاز المستخدم، سميت المعمارية معمارية المخدوم (Client Architecture)، أما إذا كانت على جهاز مركزي موصول مع أجهزة المستخدمين، تسمى المعمارية معمارية الخادم (Server Architecture)، وأما إذا تم توزيعها على أجهزة المخدوم والخادم، فتسمى عندها بمعمارية الخادم/المخدوم (Client/Server Architecture).

إن أول خطوة في تصميم معمارية النظام هو تقرير ما إذا كانت معمارية النظام تعتمد على الخادم (Server) أو المخدوم (Client) أو كليهما (Client/Server) وذلك حسب ما تقتضيه مصلحة النظام. إن المعمارية المعتمدة على الخادم تتضمن استخدام أجهزة ضخمة بينما المعمارية المعتمدة على المخدوم تعتمد على الأجهزة الشخصية وتدعم عمليات التطبيقات الجديدة. وإن هذا القرار يؤثر على العناصر الأخرى لمعمارية النظام، لذا يتم اتخاذ هذا القرار في بدايات مرحلة التصميم.

وستتعرف- عزيزي الدارس- خلال هذه الوحدة على موضوعات مهمة مثل نظام الشبكات، وتطبيقات الخادم والمخدوم، ومعمارية الكينونات الموزعة، وكيفية تحديد مواصفات البرمجيات والمعدات والأجهزة.

2.1 أهداف الوحدة

يتوقع منك- عزيزي الدارس- بعد الانتهاء من دراسة هذه الوحدة أن تكون قادراً على أن:

1. تبين أهمية عملية تصميم معمارية النظام.
2. تميز بين المعماريات المتنوعة للنظام.

3. تحدد طبقات النظام.
4. تصمم البنية التحتية للنظام.
5. تحدد مواصفات البرمجيات والمعدات المطلوبة للنظام.

3.1 أقسام الوحدة

تقسم هذه الوحدة إلى ثلاثة أقسام رئيسة:

يبين القسم الأول: عملية التحول من التحليل إلى التصميم، وهذا يغطي الهدف الأول. بينما يناقش القسم الثاني: معماريات النظام المختلفة، مثل: المعمارية المعتمدة على الخادم، المعمارية المعتمدة على المخدوم، والمعمارية المعتمدة على الخادم/ المخدوم، وطبقات الخادم/ المخدوم، وأخيراً معمارية الكينونات الموزعة ويبين الفرق بينها وميزات كل منها، وهذا يغطي الهدفين الثاني والثالث. أما القسم الثالث والأخير: فيشرح تصميم البنية التحتية للنظام، وطرق تحديد مواصفات المعدات والبرمجيات المطلوبة للنظام؛ وهذا يغطي الهدفين الرابع والخامس.

4.1 القراءات المساعدة

أنصحك- عزيزي الدارس- بالإطلاع على المراجع التالية لما تحويه من معلومات مفيدة تتعلق بموضوع هذه الوحدة:

1. Edwards, P., Systems Analysis, Design, and Development with Structured Concepts, Rinehart and Winston, 1999.
2. Kendal, K., and Kendal, J., Systems Analysis and Design, 5th Ed., 2002.
3. Silver, G. and Silver, M., Systems Analysis and Design, Addison-Wesley, 1996.
4. Aktas, Z., Structured Analysis and Design of Information Systems, Prentice-Hall, 1987.
5. http://www.psut.edu.jo / computer science/ staff/saleh abu-soud

5.1 ما تحتاج إليه لدراسة الوحدة

عزيزي الدارس. قبل أن تبدأ بدراسة هذه الوحدة تأكد من أنك قمت بتهيئة المكان الهادئ المناسب للدراسة مما يساعدك على التركيز والتفكير العميق. ثم احرص على تتبع الإرشادات التي توجهك لدراسة الوحدة، وحاول بجدية الإجابة عن أسئلة التقويم الذاتي، وحل التدريبات، لأنها تساعدك على مراجعة موضوعات الوحدة، وتعمق معلوماتك حولها.

2. التحول من التحليل إلى التصميم
Moving From Analysis to Design

إن الهدف الأساسي من عملية التحليل هو معرفة احتياجات النظام، حيث تسمى هذه المرحلة بمرحلة الـ "ماذا" (What)، أما الهدف من التصميم هو تقرير كيفية بناء النظام، فلذلك تسمى مرحلة الـ "كيف" (How)، حيث يتم فيها تطوير مجموع نتاجات مرحلة التحليل إلى تصميم.

في الواقع- عزيزي الدارس- لا يوجد حد فاصل وواضح بين مرحلتي التحليل والتصميم بل تُوجد منطقة مشتركة بينهما تحتوي على مهمات مشتركة بين المرحلتين، بالإضافة للمهمات التي تنفرد كل مرحلة بها، وهذه المهمات يمكن تفصيلها كالآتي:

1) مرحلة التحليل تحتوي على المهام التالية:
   أ- تعريف متطلبات المستخدمين (User Requirements): وفيها يتم جمع متطلبات المستخدمين للنظام الجديد.
   ب- تعريف متطلبات النظام (System Requirements): وفيها يتم تحديد اللازم عمله في النظام الجديد لتحقيق متطلبات المستخدمين.
2) المنطقة المشتركة بين مرحلتي التحليل والتصميم وتحتوي على المهام التالية:
   أ- تصميم معمارية النظام (System Architectural Design): وفيها يتم بيان شكل النظام وأجزائه وطريقة توزيع النظام على هذه الأجزاء.
   ب- توصيف النظام (System Specification): ويتم فيها استخدام إحدى طرق التوصيف أو إحدى لغات النمذجة مثل UML، لتوصيف النظام من جميع جوانبه كتحديد حدود النظام (System Boundary)، وحالات الاستخدام (Use Cases)، وتفاعل النظام (System Interaction) إلى آخره...
3) مرحلة التصميم: ويتم فيها التصميم ذو المستوى العالي (High-Level Design) مثل تصميم واجهات المستخدم من شاشات وتقارير، وتصميم البرامج باستخدام أدوات تصميم البرامج مثل الخوارزميات، وتصميم قاعدة البيانات، إلى آخره...

من هنا نرى- عزيزي الدارس- أن تصميم معمارية النظام الذي هو موضوع هذه الوحدة، يلعب دوراً محورياً في عملية التحول من مرحلة التحليل إلى مرحلة التصميم.

أسئلة التقويم الذاتي (1)

ما هي مهام مرحلتي التحليل والتصميم؟ وما هي المهام المشتركة بينهما؟

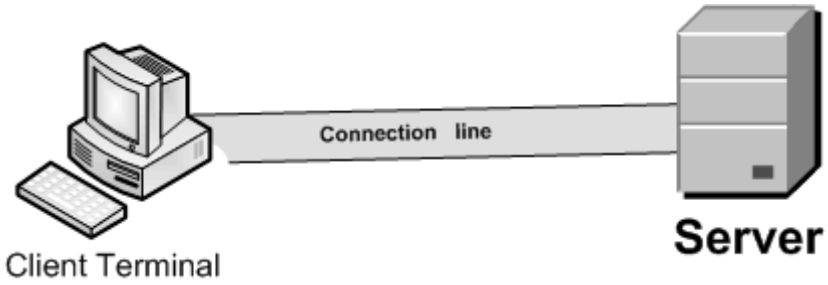
3. معماريات النظام System Architectures

1.3 المعمارية المعتمدة على الخادم

Server-based Architecture

إن المعمارية المعتمدة على الخادم تعتبر من معماريات المعالجة المركزية (Centralized Processing)، حيث يكون الخادم عبارة عن جهاز حاسوب عملاق ذي قدرات كبيرة من حيث حجم الذاكرة والسرعة ليكون قادراً على تنفيذ العمليات بشكل سريع، ويكون جهاز المخدوم عبارة عن وحدة طرفية غير ذكية (Dump Terminal) بمعنى أنها لا تتمتع بأي قدرات ذاتية، بل يقتصر عملها على استقبال وإرسال البيانات من وإلى جهاز الخادم.

ويكون المستخدم في هذه المعمارية قادراً فقط على إدخال المعلومات والطلبات عن طريق الوحدة الطرفية ومن ثم إرسالها إلى جهاز الخادم للمعالجة حيث يقوم الخادم بكل العمل ثم يرسل النتائج إلى المستخدم. الشكل (1) يبين الشكل الرئيس لهذه المعمارية.



الشكل (1) الشكل الرئيس للمعمارية المعتمدة على الخادم

في هذه المعمارية يتم تخزين أجزاء النظام الثلاثة: البرامج التطبيقية، وقاعدة البيانات، وواجهات المستخدم على جهاز الخادم، مما يؤدي إلى تقليل حركة مرور البيانات بين الوحدة الطرفية والخادم إذ إن معظم حركة المرور تتم بين البرامج التطبيقية وقاعدة البيانات وكلاهما مخزن على الخادم، وهذا يؤدي بدوره إلى زيادة كفاءة النظام.

لكن المشكلة الأساسية مع هذه المعمارية أن الخادم هو الذي يقوم بكل المعالجة لوحده، وهذه بحد ذاتها لا تعتبر مشكلة، ولكن مع النمو المطرد في تطبيقات الحاسوب أصبحت أجهزة الخوادم مشغولة جداً ومحملة أكثر من قدراتها وغير قادرة على الإجابة السريعة على طلبات المستخدم، بمعنى أن وقت الإجابة أصبح كبيراً، مما أدى إلى التأثير سلباً على أداء النظام وفعاليته. لذا وجب على مديري الشبكات أن ينفقوا الكثير من

الأموال لتطوير أجهزة الخادم، ولكن لسوء الحظ هذا التطوير لا يحل كل المشاكل كما سيتم بيانه- عزيزي الدارس- في الفقرات الآتية من هذه الوحدة.

أسئلة التقويم الذاتي (2)

ما هي المشكلة الأساسية التي تعاني منها المعمارية المعتمدة على الخادم ؟

2.3 المعمارية المعتمدة على المخدوم
Client-based Architecture

مع الانتشار الكبير لأجهزة الحاسوب الشخصي (Personal Computers)، وجد المستخدمون أنهم قادرون على تنفيذ برامجهم مثل معالج النصوص، وبرامج قواعد البيانات، وغيرها من البرامج على حواسيبهم الشخصية دون الحاجة إلى جهاز خادم.

وتسمى هذه المعمارية معمارية المعالجة المنفردة (Stand-Alone Computing) حيث تتم المعالجة بالكامل على الحاسوب الشخصي، وهذا يؤدي إلى أنه يتوجب على كل مستخدم تخزين كل البرمجيات اللازمة على حاسوبه الشخصي، مما ينتج عنه وجود برمجيات في الشركة الواحدة بعدد الحواسيب الشخصية الموجودة فيها. ومن الطبيعي أن يؤدي هذا إلى عدة مشاكل جوهرية منها:

أ- مشكلة صيانة البرمجيات (Maintenance): حيث يتوجب علينا تكرار عمليات الصيانة نفسها على كل الحواسيب الشخصية.

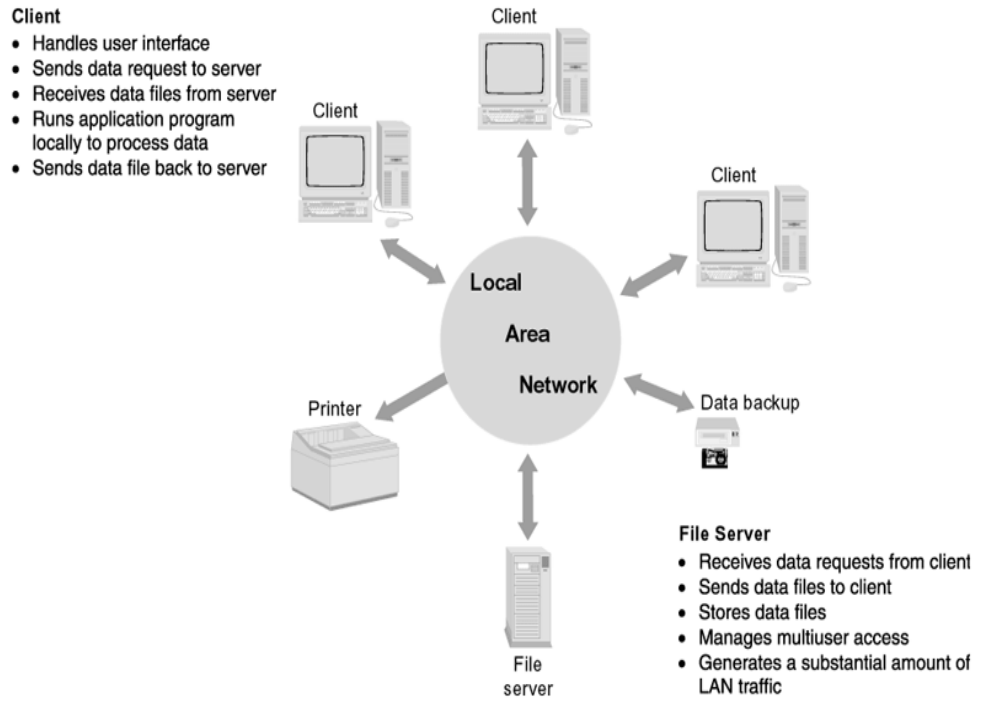
ب- مشكلة حماية البيانات والبرمجيات (Data Security): وذلك لعدم وجودها في مكان مركزي واحد حيث تسهل حمايتها وعمل نسخ احتياطية لها بسهولة.

ج- مشكلة تكامل البيانات واتساقها (Consistency and Integrity): وذلك لوجود أكثر من نسخة من البيانات موجودة على أكثر من موقع، مما يؤدي إلى إمكانية وجود أكثر من قيمة للبيانة الواحدة.

د- مشكلة زيادة التكلفة (Cost): حيث تتطلب هذه المعمارية وجود حواسيب ذات قدرات عالية قادرة على تخزين كل البرمجيات اللازمة، على كل جهاز. بالإضافة لشراء أكثر من نسخة من نفس البرمجية.

في الواقع- عزيزي الدارس- يمكن التقليل من بعض تأثيرات هذه المشاكل وذلك بربط أجهزة الحاسوب الشخصية داخل الموقع على شكل شبكة حواسيب محلية (Local

(Area Network، واعتبار أحد الأجهزة على الشبكة كجهاز خاص لتخزين الملفات (File Server)، مع بقاء واجهات الاستخدام وتطبيقات المعالجة على الحواسيب الشخصية، وذلك كما هو مبين في الشكل (2).



الشكل (2) ربط أجهزة الحاسوب الشخصية وجهاز تخزين الملفات بشبكة محلية

أسئلة التقويم الذاتي (3)

ما الفرق الأساسي بين المعمارية المعتمدة على الخادم والمعمارية المعتمدة على المخدوم؟

3.3 معمارية الخادم/المخدوم
Client/Server Architecture

إن أغلب المؤسسات في يومنا الحالي تتحول إلى معمارية الخادم/ المخدوم التي تحاول الموازنة في توزيع الحمل ما بين الخادم والمخدوم لجعل كليهما يشارك في تنفيذ أجزاء من العمل المطلوب في هذه المعمارية ليكون المخدوم مسؤولاً عن واجهات المستخدم، بينما يكون الخادم مسؤولاً عن تخزين البيانات والوصول إليها وتخزينها، وبينما يبقى منطق التطبيق إما من مسؤولية الخادم أو المخدوم أو يتم فصله عن كليهما بطبقة (Tier) منفصلة وهذا ما سيتم بيانه- عزيزي الدارس- في الفقرات الآتية من هذه الوحدة. فإذا كان منطق التطبيق من مسؤوليات المخدوم، فيسمى عندها المخدوم المخدوم السمين (Fat Client) كناية عن كثرة ما يخزنه، وأما إذا لم يكن من ضمن مسؤوليات المخدوم، فيسمى المخدوم في هذه الحالة بالمخدوم الضعيف (Thin Client).

إن خبراء تكنولوجيا المعلومات يُقرّون بأن التصميم المعتمد على المخدوم الضعيف يعطينا أداء أفضل للنظام وذلك لوجود البرامج التطبيقية بالقرب من قاعدة البيانات على الخادم، بينما يكون الأمر على عكس ذلك في التصميم المعتمد على المخدوم السمين، وذلك لأن جهاز المخدوم يأخذ على عاتقه عمليات المعالجة التي تحتاج إلى قراءة البيانات وتحديثها والوصول إليها بشكل كبير، ولكن هذه المعالجة بعيدة عن قاعدة البيانات الموجودة على جهاز الخادم، مما يؤدي إلى زيادة حركة مرور البيانات بين الخادم والمخدوم مما ينتج عنه إضعاف لكفاءة النظام.

إن التصميم المعتمد على المخدوم السمين أكثر تكلفة من التصميم المعتمد على المخدوم الضعيف، وذلك لأن جهاز المخدوم السمين أضخم وأكثر قوة من جهاز المخدوم الضعيف، وكذلك الأمر بالنسبة لتكاليف الصيانة، فهي أكثر في حالة جهاز المخدوم السمين وذلك لوجود البرامج التطبيقية عليه.

إن معمارية الخادم/المخدوم تحتوي على العديد من المنافع، أهمها:

أولاً: هذه المعمارية قابلة للتوسع والتقلص من ناحية القدرة على التخزين والوصول إلى المعلومات حيث إذا امتلأ الخادم بالبيانات، فباستطاعتنا إضافة خادم جديد وتحويل بعض العمليات أو البيانات إليه بكل سهولة ويسر.

ثانياً: إن معمارية الخادم/المخدوم تستطيع دعم العديد من أنواع أجهزة الخادم وأجهزة المخدوم، فلسنا ملزمين بنوع معين كما هي الحال في المعمارية المعتمدة على

الخادم فقط، وكذلك الأمر بالنسبة لأنظمة التشغيل.

ثالثاً: بهذه المعمارية، أصبح من الممكن وصل مجموعة الحواسيب باستخدام أنظمة تشغيل مختلفة بسبب عدم وجود خادم واحد يدعم جميع التطبيقات، فبذلك تكون الشبكة موثوقة أكثر ولا يوجد هناك ما يسمى بنقطة التحكم المركزية التي تسبب وقوف الشبكات عن العمل كما هو الحال في معمارية الخادم، ففي معمارية الخادم/المخدوم إذا تعطل أحد الخوادم فلا داعي لتوقيف الشبكة، فقط ما علينا إلا تغيير الخادم المتعطل بخادم آخر.

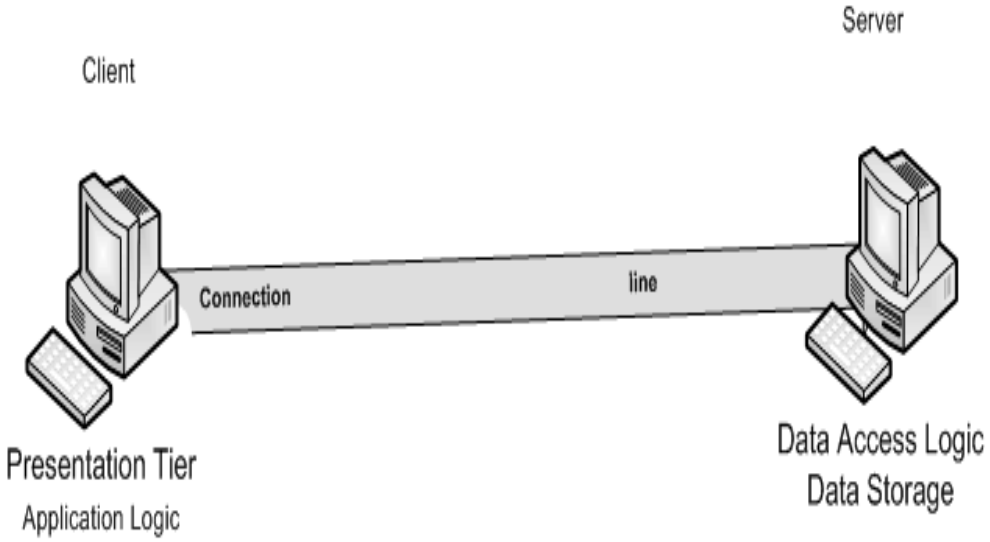
لكن معمارية الخادم/المخدوم تعاني من بعض المحددات، أهمها هو التعقيد (Complexity) حيث إن كل التطبيقات في هذه المعمارية تحتوي على جزأين من البرمجيات جزء في الخادم وجزء في المخدوم وتطوير هذا النوع من البرمجيات يحتاج لجهود مضنية تختلف عن البرمجيات العادية. وعلى المبرمجين في هذا النوع من المعماريات تعلم الكثير من طرق البرمجة الحديثة والمعقدة مما يتطلب الكثير من إعادة التدريب والتعديل على البرمجيات في كل من الخادم والمخدوم.

أسئلة التقويم الذاتي (4)

اذكر الأسباب الرئيسة التي جعلت معظم المؤسسات تعتمد معمارية الخادم/المخدوم.

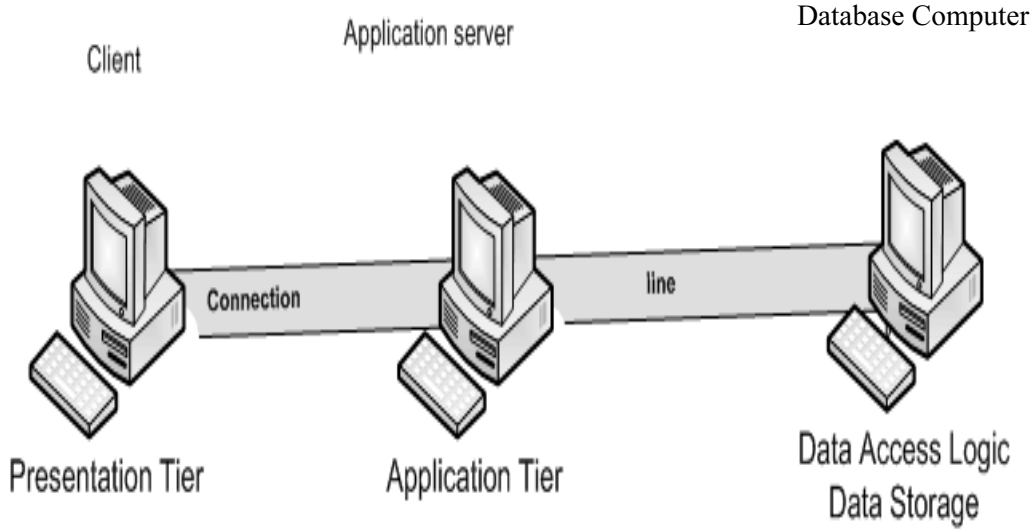
4.3 طبقات الخادم/المخدوم Client-Server Tiers

إن معمارية الخادم/المخدوم التي قد تم بيانها سابقاً- عزيزي الدارس- تسمى معمارية الطبقتين (Two Tiers Architecture)، ويكوّن الخادم، وكما هو موضح في الشكل (3)، الطبقة الأولى وتحتوي على قاعدة البيانات، بينما يكوّن المخدوم الطبقة الأخرى وتحتوي على واجهات الاستخدام والبرامج التطبيقية.



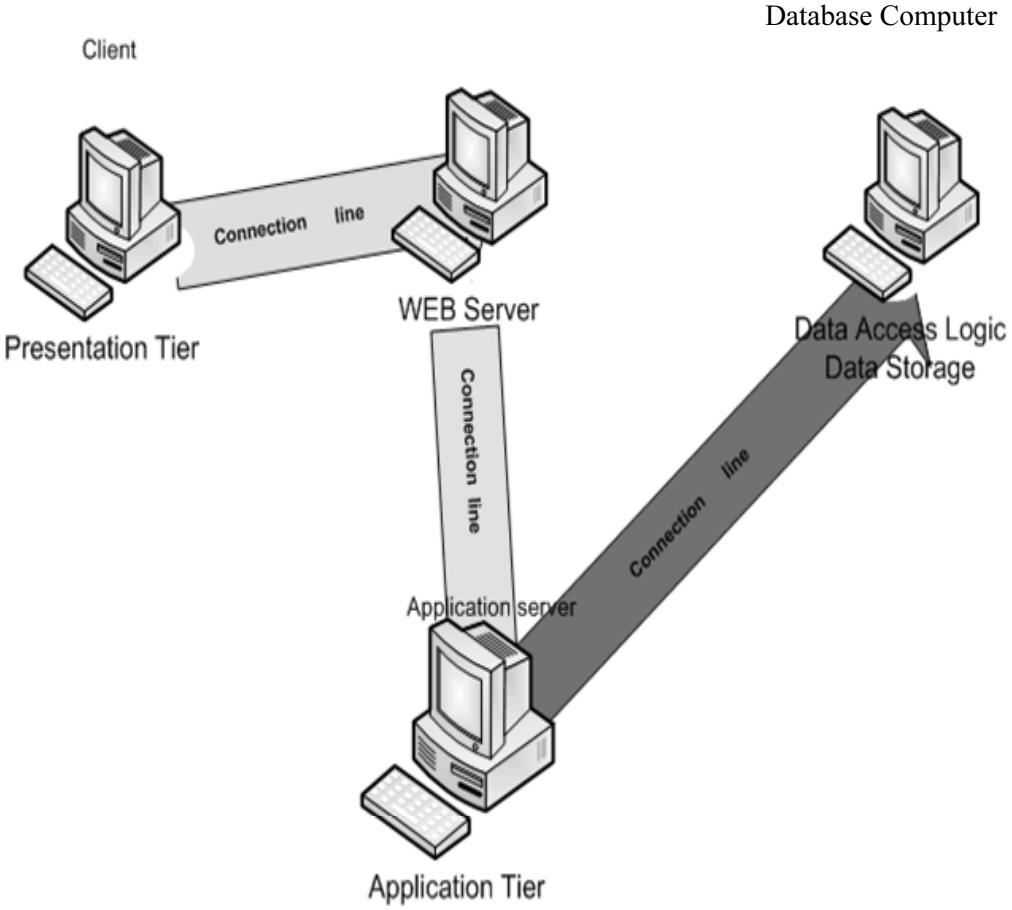
الشكل (3) معمارية الطبقتين

وفي الأنظمة الضخمة ربما يتطلب الأمر لوضع البرامج التطبيقية علـــى جهـــاز حاسوب ثالث غير الخادم أو المخدوم ويقع بينهما ويسمى بخادم التطبيقات (Application Server). وتسمى هذه المعمارية معمارية الثلاث طبقات (Three Tiers Architecture)، والشكل (4) يبين الشكل العام لها.



الشكل (4) معمارية الثلاث طبقات

هناك بعض الحالات– عزيزي الدارس– التي يتطلب فيها النظام توزيع البـــرامج التطبيقية على عدة خوادم في الشبكة، حيث تسمى هـــذه المعماريـــة معماريـــة الطبقـــات المتعددة (Multiple Tiers). الشكل (5) يوضح هذه المعمارية.



الشكل (5) معمارية الطبقات المتعددة

إن الهدف الرئيس من المعماريات المتعددة الطبقات مقارنـــة مـــع المعماريـــات السابقة هو أنه يتم فصل العمليات بشكل جيد مما يحدث ما يعرف بتوازن الحمل (Load Balance) وبالتالي تتمتع هذه المعمارية بالمرونة الكافية والكفاءة المطلوبة لأي نظام.

ولكن هناك بعض السلبيات للمعمارية المتعددة الطبقات بالمقارنة مع المعماريات الأخرى، أهمها:

أولاً: إنها تضع الكثير من الحمل على الشبكات حيث إنها تتطلب الكثير من الاتـــصالات

وتمرير البيانات وبذلك تُولّد الكثير من الأزمات في الشبكة.

ثانياً: إنها أصعب من ناحية البرمجة والفحص وأكثر تعقيداً لوجود العديد من الطبقات ولوجود العديد من الأشياء المتداخلة قبل إتمام طلب الزبون.

أسئلة التقويم الذاتي (5)

كيف تمكنت فكرة تعدد الطبقات من تحقيق مفهوم توازن الحمل (Load Balance)؟

5.3 معمارية الكينونات الموزعة

Distributed Objects Architecture

من وجهة نظر تحليل النظم بالكينونة تعتبر– عزيزي الدارس– الحوسبة باستخدام الكينونة الموزعة نسخة أخرى من الحوسبة باستخدام الخادم/المخدوم، ولكن يختلف المخدومون عن الخوادم. فأجهزة المخدوم تطلب خدمات (Services) من الخوادم وليس من أجهزة مخدوم أخرى، بينما تعمل بعض الخوادم كأجهزة مخدوم بطلبها خدمات من أجهزة خوادم أخرى. وأجهزة المخدوم يجب أن تعرف الخدمات التي تقدمها أجهزة الخوادم الأخرى وكيفية مخاطبتها.

تناسب هذه المعمارية العديد من أنواع الأنظمة، ولكنها قد تقيد المصممين من ناحية المرونة (Flexibility)، إذ إنهم يجب أن يحددوا مواقع تخزين الخدمات. كما أنهم يجب أن يخططوا جيداً لإمكانية قابلية النظام للتوسع والتغير (Scalability) عند إضافة مخدومين جدد، من حيث توزيع الأحمال على أجهزة الخوادم.

ولحل هذه الإشكاليات، يمكننا إلغاء الاختلافات بين أجهزة الخادم وأجهزة المخدوم وتصميم معمارية النظام كمعمارية الكينونات الموزعة. وفي هذه المعمارية، يمكن توزيع الكينونات على كل الأجهزة على الشبكة، من دون أن نفرق بين أجهزة الخادم وأجهزة المخدوم، وهذه الكينونات تكون مزودة بواجهة (Interface) بالخدمات التي تقدمها، وتتفاعل فيما بينها عن طريق طبقة من البرمجيات يكون موقعها ما بين الخادم والمخدوم وتدعى أحياناً ببرمجيات الطبقة الوسطى (Middleware Programs).

ومن أوجه القوة في هذه المعمارية:

أ- يمكن تخزين الكينونات في أي جهاز على الشبكة بغض النظر إن كان خادماً أو مخدوماً.

ب- عند برمجة برامج أجهزة المخدوم، من غير الضروري معرفة كينونة المخدوم لموقع كينونة الخادم، إذ إنها عند استدعائها وبكل بساطة تطلب من الشبكة تحديد موقعها، مما يعطي الفرصة لإضافة وحذف الخوادم من الشبكة دون الحاجة لتعديل برمجيات المخدوم، فقط البرمجيات الوسطية يجب أن تعرف عن هذه التغيرات وهذا يؤدي إلى تقليص تكلفة الصيانة في برمجية الخادم/المخدوم، مما يزيد من مرونة النظام ومن قابليته للتوسع.

أسئلة التقويم الذاتي (6)

ما هي الفوائد التي جناها محللو النظم من استخدام البرمجيات الوسطية (Middleware Programs)؟ وما هو دورها في معمارية الكينونات الموزعة؟

4. متطلبات بناء النظام

1.4 تصميم البنية التحتية للنظام Infrastructure Design

في أغلب الأحيان، وعند بناء نظام لمؤسسة ما، تحتوي هذه المؤسسة على معدات وبرمجيات وبنية تحتية للاتصالات. لذلك على فريق العمل أن يأخذ بعين الاعتبار طريقة استغلال وتغيير هذه البنية التحتية الموجودة وإعادة استغلالها بما يتناسب مع متطلبات النظام الجديد. ومن الجدير بالذكر- عزيزي الدارس- أن التعامل مع عناصر البنية التحتية أمر معقد ويتطلب أشخاصاً فنيين ومحترفين.

وكأي فريق عمل من الأفضل أن يكون الشخص المسؤول عن تحليل البنية التحتية هو نفسه صاحب اتخاذ القرارات بخصوص تعديل البنية التحتية ولذلك سنتعلم في البند التالي كيفية الحصول والتعامل مع مواصفات المعدات والبرمجيات في النظم.

2.4 تحديد مواصفات البرمجيات والمعدات Software and Hardware Specifications

يتم في مرحلة التصميم اختيار البرمجيات والمعدات اللازم استخدامها في النظام المستقبلي. ويتم تشغيل النظام الجديد عادة على المعدات الموجودة في المؤسسة، لكن في أحيان أخرى يتم شراء معدات وبرمجيات جديدة والتي يجب تحديد مواصفاتها بدقة.

إن عملية تحديد المواصفات هذه هي لتوثيق المعدات والبرمجيات التي نحتاجها لدعم النظام الجديد وهي من مهمات مرحلة التصميم، بينما يجب أن تترك طلبات شراء البرمجيات والمعدات الحقيقية والنهائية لقسم المشتريات في تلك المؤسسة ولكن يبقى فريق العمل مسؤولا عن التواصل مع قسم المشتريات المتخصصة باستخدام مواصفات الأجهزة والبرمجيات التي تم تحديدها أثناء مرحلة التصميم.

هنالك العديد من الخطوات التي يجب اتباعها لصياغة وثيقة مواصفات المعدات والبرمجيات، ويمكن إجمالها – عزيزي الدارس- في النقاط الآتية:

أولاً: علينا إنشاء لائحة بالمعدات المطلوبة لدعم النظام الجديد، حيث تحتوي هذه اللائحة بشكل عام على أجهزة مثل خوادم قواعد البيانات، وخوادم شبكات الاتصال ومعدات المخرجات والمدخلات، مثل الشاشات والطابعات والماسحات الضوئية والمعدات الاحتياطية ومعدات التخزين وأي معدات أخرى نحتاجها لدعم تطبيقات النظام.

ثانياً: وضع مواصفات مفصلة لكل بند من بنود القائمة. ويتم تحديد المواصفات من قبل

فريق العمل عادةً بطريقة من طريقتين:

1. بتحديد المواصفات الدقيقة (Request For Proposal RFP) حيث يتم في هذه الطريقة تحديد مواصفات الأجهزة والبرمجيات المطلوبة بكل دقة. ومن ميزات هذه الطريقة سهولة متابعة العروض واختيار الأجهزة المناسبة، ولكن لا ينصح باستخدام هذه الطريقة إذا كان فريق العمل يعرف ما يريد من أجهزة بكل دقة، وإذا كان مطلعاً على ما هو موجود في السوق.
2. بتحديد الاحتياجات (Request For Quotation RFQ) ويتم في هذه الطريقة تحديد الحد الأدنى من مواصفات المعدات والبرمجيات المطلوبة، ويترك المجال للشركات بتقديم مواصفات أفضل ما يتوفر عندها من أجهزة. ومن ميزات هذه الطريقة أنها تعطي الفرصة لفريق العمل لمعرفة أحدث ما يتوفر في السوق من معدات وبرمجيات دون جهد أو عناء، حيث تقدم الشركات أفضل وأحدث ما لديها عند تقديم العروض.

تدريب (1)

اكتب RFP لجهاز حاسوب شخصي.

تدريب (2)

اكتب (RFQ) لجهاز حاسوب شخصي.

أسئلة التقويم الذاتي (7)

ما هو الفرق بين (RFP) و(RFQ)؟ واذكر ميزات كل منها.

5. الخلاصة

إن تحديد معمارية النظام يعتبر من الخطوات المهمة في مرحلة التحليل. لقد رأينا خلال هذه الوحدة – عزيزي الدارس– أن النظام يتكون من ثلاثة أجزاء: البرامج التطبيقية (Application Programs) التي تقوم بمهام معالجة منطق النظام، وقاعدة البيانات (Database) لتخزين البيانات، وواجهات المستخدم (User Interface) التي تسمح للمستخدم بالتخاطب والتفاعل مع النظام. وقد رأينا أن توزيع هذه الأجزاء يحدد طبيعة معمارية النظام، فإذا كانت جميعها على جهاز المستخدم، سميت المعمارية معمارية المخدوم (Client Architecture) ، وأما إذا كانت على جهاز مركزي موصول مع أجهزة المستخدمين، تسمى المعمارية بمعمارية الخادم (Server Architecture)، وأما إذا تم توزيعها على أجهزة المخدوم والخادم، فتسمى عندها معمارية الخادم/المخدوم (Client/Server Architecture).

وفي الأنظمة الموزعة، يمكن بناء معمارية النظام ككينونات موزعة يتم التخاطب فيما بينها عن طريق برمجيات الطبقة الوسطى (Middleware Programs). وهذه المعمارية مرنة أكثر من الأنواع الأخرى، وبرمجتها أسهل وتكلفة صيانتها أقل.

6. لمحة عن الوحدة الدراسية الثامنة

تعنى الوحدة الثامنة بدراسة خصائص واجهات الاستخدام، وأنواعها وعناصرها، بالإضافة إلى طرق تصميمها، ووسائل حمايتها والحفاظ عليها. وتتضمن الوحدة إرشادات عامة لتصميم واجهات الاستخدام.

وتعرض الوحدة الثامنة بيان وشرح مبادئ تصميم شاشات الإدخال في النظام، ويتضمن ذلك أنواع المدخلات وطرق التحقق من صحتها، بالإضافة إلى إرشادات لعملية التصميم. كما تتطرق الوحدة لبيان أنواع التقارير، ثم تعرج على بيان مبادئ وطرق تصميم التقارير، وتنتهي بإرشادات لعملية التصميم.

7. إجابات التدريبات

تدريب (1)

- Pentium 4
- 4 Mhz Processor
- 4 Gbyte Hard Disk
- 17 inch flat screen
- 512 Mbyte RAM

تدريب (2)

- جهاز كمبيوتر آخر موديل وسريع
- (RAM) تكفي لتنفيذ برمجية (Oracle)
- (Hard Disk) يكفي لتخزين برمجية (Oracle) ومايكروسوفت أوفيس، بالإضافة إلى 1000 برنامج
- شاشة كبيرة نسبياً

8. مسرد المصطلحات

- **برمجيات الطبقة الوسطى Middleware Programs:** تمثل طبقة من البرمجيات يكون موقعها ما بين الخادم والمخدوم وتكمن وظيفتها في دعم التفاعل بين الكينونات في بيئة حوسبة موزعة وتساعد مبرمجي النظام على تطوير البرمجيات وتساعد أيضاً المبرمجين على التركيز على مستعملي النظام والكينونات والطرق المستخدمة في التطبيق دون الحاجة لتعديل الخادم الذي يحتوي هذه الكينونات.
- **تحديد الاحتياجات Request For Quotation RFQ:** طريقة من طرق تحديد مواصفات الأجهزة والبرمجيات المطلوبة، حيث يتم فيها تحديد الحد الأدنى من المواصفات المطلوبة، ويترك المجال للشركات لتقديم مواصفات أفضل ما يتوفر عندها من أجهزة.
- **تحديد المواصفات الدقيقة Request For Proposal RFP:** طريقة من طرق تحديد مواصفات الأجهزة والبرمجيات المطلوبة بشكل دقيق.
- **طبقات الخادم/ المخدوم Client/Server Tiers:** عبارة عن أجهزة حاسوب تقع بين الخادم والمخدوم يمكن توزيع بعض أجزاء النظام عليها.
- **المخدوم السمين Fat Client:** عبارة عن جهاز المخدوم الذي يحتوي على واجهات الاستخدام ومنطق التطبيق.
- **المخدوم الضعيف Thin Client:** عبارة عن جهاز المخدوم الذي يحتوي على واجهات الاستخدام فقط.
- **المعمارية المعتمدة على الخادم Server-based Architecture:** إحدى معماريات النظام ويخزن النظام بأجزائه الثلاثة على جهاز الخادم الذي هو عبارة عن جهاز حاسوب عملاق ذي قدرات كبيرة من حيث حجم الذاكرة والسرعة إذ يكون قادراً على تنفيذ العمليات بشكل سريع، ويكون جهاز المخدوم عبارة عن وحدة طرفية غير ذكية (Dump Terminal) بمعنى أنها لا تتمتع بأي قدرات ذاتية، بل يقتصر عملها على استقبال وإرسال البيانات من وإلى جهاز الخادم.
- **المعمارية المعتمدة على الخادم/المخدوم Client/Server-based Architecture:** في هذه المعمارية يكون المخدوم مسؤولاً عن واجهات المستخدم، بينما يكون الخادم مسؤولاً عن تخزين البيانات والوصول إليها وتخزينها، ويبقى منطق التطبيق إما من مسؤولية الخادم أو المخدوم أو يتم فصله عن كليهما بطبقة (Tier) منفصلة وإذا كان منطق التطبيق من مسؤوليات المخدوم، فيسمى عندها المخدوم المخدوم السمين (Fat Client) كناية عن كثرة ما يخزنه، أما إذا لم يكن من ضمن مسؤوليات

المخدوم، فيسمى المخدوم في هذه الحالة المخدوم الضعيف (Thin Client).

- **المعمارية المعتمدة على المخدوم Client-based Architecture:** تسمى هذه المعمارية معمارية المعالجة المنفردة (Stand-Alone Computing) حيث تتم المعالجة بالكامل على الحاسوب الشخصي، وهذا يؤدي إلى أنه يتوجب على كل مستخدم تخزين كل البرمجيات اللازمة على حاسوبه الشخصي.

9. المراجع

1. Aktas, Z., Structured Analysis and Design of Information Systems, Prentice-Hall, 1987.
2. Baudoin, C. and Hollowell, G., Realizing The Object-Oriented Life Cycle, Prentice- Hall, 1996.
3. Booch, G., Rumbaugh, J., and Jacobson, I., The Unified Modeling Language User Guide, Addison-Wesley, 1999.
4. Dennis, A. et al, Systems Analysis & Design: An Object-Oriented Approach with UML. John Wiley & Sons Inc., 2002.
5. Dewitz, S., Systems Analysis and Design and the Transition to Objects, McGraw-Hill, 1996.
6. Edwards, P., Systems Analysis, Design, and Development with Structured Concepts, Rinehart and Winston, 1999.
7. Henderson, P., Object-Oriented Specification and Design with C++, McGraw- Hill, 1993.
8. Hoffer, J. et al., Modern Systems Analysis & Design, 2nd Ed., Addison-Wesley, 1999.
9. Kendal, K., and Kendal, J., Systems Analysis and Design, 5th Ed. 2002.
10. Kruchten, P., The Rational Unified Process: An Introduction. 2nd Ed., Addison- Wesley, 2000.
11. Lejk, M. and Deeks. "An Introduction to Systems Analysis Techniques", 2nd Ed., Pearson Education Limited, 2002.
12. Maciaszek, A., Requirements Analysis and System Design, Developing Information Systems with UML, Addison-Wesley, 2001.
13. Quatrani, T., Visual Modeling with Rational Rose 2000 and UML, Addison- Wesley, 2000.
14. Satzinger, J., Systems Analysis and Design, 2nd Ed., Thomson Learing, 2002.
15. Silver, G. and Silver, M., Systems Analysis and Design, Addison-Wesley, 1996.
16. http://www.psut.edu.jo / computer science/ staff/saleh abu-soud
17. www.rational.com
18. www.omg.com

الوحدة الثامنة

تصميم واجهات المستخدم

User-Interface Design

محتويات الوحدة

الموضوع	الصفحة
1. المقدمـــة	183
1.1 تمهيد	183
2.1 أهداف الوحدة	183
3.1 أقسام الوحدة	183
4.1 القراءات المساعدة	184
5.1 ما تحتاج إليه لدراسة الوحدة	184
2. النمذجة والتصميم الموجه للمستخدم Prototyping and User-Centered Design	185
3. تصميم شاشات الإدخال Input Screen Design	186
1.3 عناصر شاشة الإدخال Elements of Input Screen	186
2.3 التحقق والتيقن من المدخلات Validation and Verification of Inputs	187
3.3 عملية تصميم الشاشات Input Screen Design Process	189
4.3 إرشادات لتصميم الشاشات Input Screens Design Guidelines	193
4. تصميم التقارير Output Reports Design.....	194
1.4 أنواع التقارير Types of Reports	194
2.4 عملية تصميم التقارير Reports Design Process	195
3.4 إرشادات لتصميم التقارير Reports Design Guidelines	200
5. الخلاصة.....	201
6. لمحة عن الوحدة الدراسية التاسعة.....	202
7. إجابات التدريبات.....	202
8. مسرد المصطلحات.....	203
9. المراجع.....	205

1. المقدمة

1.1 تمهيد

أهلاً بك- عزيزي الدارس- إلى الوحدة الثامنة من مقرر "1380 تحليل الأنظمة وتصميمها"، وهي بعنوان " تصميم واجهات المستخدم".

يعتبر تصميم واجهات الاستخدام؛ وشاشات الإدخال والتقارير، من أهم الأمـــور، وذلك لأنها تكاد تكون الوحيدة من أجزاء النظام التي يراها المستخدم ويتعامل معها كـــل لحظة، وفي معظم الأحيان يحكم المستخدم على النظام من خلال واجهات الاستخدام. لهذه الأسباب، يجب إيلاء تصميم واجهات الاستخدام الاهتمام الأكبر وأخذ رأي المستخدم فيها قبل برمجتها بشكل نهائي.

في هذه الوحدة، ستقوم- عزيزي الدارس- بدراسة خصائص واجهات الاستخدام، وأنواعها وعناصرها، وطرق تصميمها، بالإضافة إلى وسائل حمايتها والحفـــاظ عليهـــا. وتتضمن الوحدة إرشادات عامة لتصميم واجهات الاستخدام.

2.1 أهداف الوحدة

يتوقع منك-عزيزي الدارس- بعد الانتهاء من دراسة هذه الوحدة أن تكون قادراً على أن:

1. توضح المقصود بالتصميم الموجه للمستخدم.
2. تصمم شاشات النظام.
3. تصمم تقارير النظام

3.1 أقسام الوحدة

تقسم هذه الوحدة إلى ثلاثة أقسام رئيسة: حيث يوضـــح القـــسم الأول:المقـــصود بالنمذجة والتصميم الموجه للمستخدم، وهذا يغطي الهدف الأول. أما القسم الثاني: فيعنـــى ببيان وشرح مبادئ تصميم شاشات الإدخال في النظام، ويتضمن ذلك أنـــواع المـــدخلات وطرق التحقق من صحتها، بالإضافة إلى إرشادات لعملية التـــصميم. ويغطـــي الهـــدف الثاني. وأما القسم الثالث: فيبدأ ببيان أنواع التقارير، ثم يعرج على بيان مبـــادئ وطـــرق تصميم التقارير، وينتهي بإرشادات لعملية التصميم ويغطي الهدف الثالث.

4.1 القراءات المساعدة

أنصحك- عزيزي الدارس، بالاطلاع على المراجع التالية لما تحويه من معلومات قيمة تساعدك على فهم موضوع هذه الوحدة:

1. Dennis, A. et al, Systems Analysis & Design: An Object-Oriented Approach with UML. John Wiley & Sons Inc., 2002.
2. Dewitz, S., Systems Analysis and Design and the Transition to Objects, McGraw-Hill, 1996.
3. Edwards, P., Systems Analysis, Design, and Development with Structured Concepts, Rinehart And Winston, 1999.
4. Hoffer, J. et al., Modern Systems Analysis & Design, 2nd Ed., Addison-Wesley, 1999.
5. Kendal, K., and Kendal, J., Systems Analysis and Design, 5th Ed.,.2002.
6. Satzinger, J., Systems Analysis and Design, 2nd Ed., Thomson Learing, 2002.
7. Silver, G. and Silver, M., Systems Analysis and Design, Addison-Wesley, 1996.
8. http://www.psut.edu.jo / computer science/ staff/saleh abu-soud

5.1 ما تحتاج إليه لدراسة الوحدة

عزيزي الدارس- إن كل ما تحتاج إليه لدراسة هذه الوحدة وفهمها هو أن تكون مستعداً للتعلم، موفراً المكان المناسب للدراسة، ثم احرص على تتبع الإرشادات التي توجهك لدراسة الوحدة، وحاول بجدية الإجابة عن أسئلة التقويم الذاتي، وحل التدريبات، لأنها تساعدك على مراجعة موضوعات الوحدة، وتعمق معلوماتك حولها. وإذا شعرت بحاجة إلى مناقشة بعض الموضوعات أو طرح بعض الاستفسارات التي تثير اهتمامك، فارجع إلى مشرفك الأكاديمي دون تردد، وستجد منه العون المناسب بإذن الله.

2. النمذجة والتصميم الموجه للمستخدم
Prototyping and User-Centered Design

التصميم الموجه للمستخدم يعني إشراك المستخدم في عملية التصميم، ويكون هذا الإشراك- عزيزي الدارس- على وجهين:

- إما أن يشترك المستخدم مع المصممين في عملية التصميم ويكون جزءاً من فريق التصميم.
- أو أن يقوم المستخدم بعملية تقييم التصميم بعد الانتهاء منه.

يبدأ التصميم الموجه للمستخدمين بفهم طبيعة المستخدمين، وعلى المصمم أن يأخذ بعين الاعتبار النقاط التالية:

- من هم أهم المستخدمين؟
- ما هو هدفهم من استخدام النظام؟
- ما هو معدل عدد مرات استخدام النظام ؟
- ما هي خبراتهم؟
- ما هي جنسياتهم واللغة التي يستطيعون التعامل بها؟
- ما هي طبيعة المعلومات التي تهمهم؟
- كيف سيستخدمون المعلومات التي يحصلون عليها من النظام، هل يريدونها مطبوعة أم مقروءة فقط على الشاشة؟

يمكن للمصمم أن يرسم الشاشة أو التقرير على الورق أو باستخدام أحد البرمجيات السهلة الاستخدام مثل فيجوال بيسك أو أوراكل وأن يريها للمستخدم من أجل تقييمها وأخذ رأيه فيها، عندها يمكن للمستخدم أن يغير ما يشاء بدون جهد أو وقت أو تكاليف، وعند الاتفاق على الشكل النهائي للشاشة أو التقرير، يقوم المصمم بتصميم الشكل النهائي للشاشة، وهذا يسمى نموذج الشاشة أو التقرير (User Interface Prototype) وللإسراع في العملية ولزيادة دقة تصميم النموذج يمكن للمصمم أن يستخدم برامج خاصة تساعد على عمل النموذج بكل سهولة ويسر.

أسئلة التقويم الذاتي (1)

ما المقصود بالتصميم الموجه للمستخدم؟

3. تصميم شاشات الإدخال Input Screen Design

1.3 عناصر شاشة الإدخال Elements of Input Screen

تتكون شاشة الإدخال من عدة عناصر، أهمها:

1) الترويسة (Header)، وتحتوي على العناصر الآتية:
- اسم النظام وعنوان الشاشة.
- شعار المؤسسة.
- التاريخ.
- الوقت.
- تاريخ ووقت آخر مرة تم استخدام الشاشة فيه، وهذا جيد لحماية النظام.

2) جسم الشاشة (Screen Body):

ويحتوي على الحقول (Data Fields) التي يتم إدخال البيانات للنظام عن طريقها، ولهذه الحقول عدة أنواع أهمها:

- (Radio Button): يستخدم لاختيار خيار واحد من عدة خيارات، وتكون هذه الخيارات عادةً قليلة ومحدودة، وظاهرة جميعها على الشاشة. من ميزات هذا النوع أنه سهل الاستخدام، ولكنه جامد Static أي أنه لتعديل الخيارات يتوجب علينا تعديل البرمجة الخاصة به.
- (Check List): يستخدم لاختيار عدة خيارات تكون ظاهرة على الشاشة. ومن ميزات هذا النوع أنه سهل الاستخدام أيضاً، ولكنه جامد (Static) أي انه لتعديل الخيارات يتوجب علينا تعديل البرمجة الخاصة به.
- (Combo Box): يستخدم لإختيار خيار واحد من عدة خيارات، وتكون هذه الخيارات عادة كثيرة، وغير ظاهرة على الشاشة عدا الخيار الأول. ومن ميزات هذا النوع أنه سهل الاستخدام، وغير جامد (Dynamic) أي انه لتعديل الخيارات لا يتوجب علينا تعديل البرمجة الخاصة به، ولكننا نستطيع ذلك من خلال التعديل على قاعدة البيانات.
- (Text Field): يستخدم هذا النوع لإدخال البيانات عن طريق لوحة المفاتيح. ومن سيئاته أنه يصعب التحكم بصحة البيانات المدخلة، لذلك عادة ما تكون الأولوية في اختيار نوع الحقل للأنواع الأخرى، ولا يستخدم هذا النوع إلا إذا اضطررنا لذلك.

3) الأزرار (Buttons):

يحتوي على الأزرار التي يتم عن طريقها اختيار عملية معينة، مثل:

أسئلة التقويم الذاتي (2)

ما هي أجزاء الشاشة؟ وما هي محتويات كل منها؟

2.3 التحقق والتيقن من المدخلات

Validation and Verification of Inputs

إن الأخطاء موجودة في كل أنواع وسائل إدخال البيانات بدون استثناء، لذلك يجب إيجاد طريقة لاكتشاف الأخطاء سريعاً قبل دخولها إلى النظام، وذلك لأن الأخطاء يكون تأثيرها عادة متراكماً وفي بعض الأحيان يصعب بل يستحيل تصحيح الخطأ بعد دخوله للنظام، كإدخال سعر الدولار في نظام بنكي بشكل خاطئ، فمن الصعب إلغاء تأثير هذا الخطأ بعد إجراء صفقات بناءً على السعر الخاطئ. لذلك فإنه من الحكمة قضاء وقت كبير في تصميم طرق لاكتشاف الأخطاء قبل دخولها للنظام.

وهنالك العديد من وسائل اكتشاف الأخطاء – عزيزي الدارس– هذه أهمها:

1. **التحقق (Verification)**

يعتبر التحقق طريقة يدوية يتبعها مدخل البيانات للتأكد من هذه البيانات عند إدخالها، فإما أن تكون بإعادة الإدخال مرتين أو أكثر لنفس البيانات، ومقارنة بعضها ببعض، أو بالنظر كقيام المستخدم بالتأكد من هوية الزبون بالنظر في هويته الشخصية.

2. **التثبت (Validation)**

عبارة عن طريقة آلية برمجية، تحتوي على العديد من الأنواع، مثل:

أ- فحص النوع (Class Test): يستخدم للتأكد من نوع البيانات المدخلة، مثل البيانات العددية (Numeric Data) أو البيانات النصية (Text Data) أو البيانات العددية النصية المختلطة (Alphanumeric Data).

ب- فحص الإشارة (Sign Test): يناسب هذا الفحص البيانات التي تكون دائماً موجبة أو سالبة ولا يمكن أن تأخذ الحالتين، مثل المسافة والوزن.

ج- فحص المدى (Range Test): يناسب هذا الفحص البيانات التي يكون لها مدى معلوم بغض النظر عن الإشارة، مثل العلامة عندما يكون الحد الأدنى لها صفر والحد الأعلى مائة.

د- فحص المعقولية (Reasonableness Test): هنالك حالات لا يكون للبيانات فيها مدى معلوم، سواءً من طرف واحد مثل العمر فهو معلوم الحد الأدنى وغير معلوم الحد الأعلى، أو من طرفين كدرجة الحرارة. وفي هذه الحالة لا نستطيع تطبيق أي من أنواع التثبت الأخرى، فنلجأ إلى فحص المعقولية ويكون الخطأ على شكل تنبيه (Warning). ومن الأمثلة على هذا النوع العمر إذ إنه لا يوجد حد أعلى واضح فإذا فرضنا أن الحد الأعلى مائة وعشرون مثلاً، يمكن لعمر شخص آخر أن يكون عمره مائة وخمسة وعشرين سنة، وفي هذه الحالة يعطي النظام تنبيهاً للمستخدم بأن هذا العمر غير معقول، فإذا كان صحيحاً يتغاضى عنه المستخدم أما إذا كان خطأً يقوم عندها المستخدم بتصحيحه.

ه- فحص التسلسل (Sequence Test): يستخدم للبيانات المركبة التي تحتوي على أكثر من جزء ويكون الترتيب مهماً، مثل التاريخ عندما يحتوي على اليوم ثم الشهر ثم السنة، فلا يجوز للمستخدم أن يدخل الشهر قبل اليوم.

و- فحص الموجودية (Presence Test): يستخدم هذا الفحص لمفاتيح البيانات عادة، مثل رقم الطالب أو الرقم الوطني، فعند إدخال رقم الطالب لإدخال علاماته مثلاً، يجب التأكد من أن الطالب موجود فعلاً في قاعدة البيانات قبل القيام بإدخال علاماته.

ز- فحص الترميز (Code Test): نضطر أحياناً- وكما بيّنا في الوحدة الثالثة من هذا المقرر- لاستعمال رمز معين بدلاً من البيانة نفسها، كاستخدام 11 بدلاً من قسم الحاسوب مثلاً و34 بدلاً من قسم الهندسة الكهربائية. وفي هذه الحالة يسمى فحص الرمز المدخل إن كان من ضمن الرموز المعروفة في النظام أم لا بفحص الترميز.

ح- الفحص المركب (Combination Test): يستخدم هذا النوع من الفحص في حالة وجود ضرورة لربط أكثر من حقل بعضها مع بعض. كعلاقة اسم الزبون وفرع البنك الذي فتح حساباً فيه، في هذه الحالة وعند إدخال اسم العميل نتأكد من أنه أحد عملاء البنك أم لا وكذلك الأمر بالنسبة لاسم فرع البنك، وهنا لا بد لنا- عزيزي الدارس- من فحص ما إذا كان هذا العميل هو من ضمن عملاء هذا الفرع.

3. فحص الخانات (Check Digits)

عبارة عن عملية حسابية تنفذ على البيانات المدخلة ومقارنة نتائجها بنتيجة معلومة مسبقاً، فإذا كانت النتيجتان غير متطابقتين، نتأكد عندها من وجود خطأ في البيانات المدخلة. وهذه الطريقة تساعد على اكتشاف الأخطاء في الحقل المُدخَل.

ولتوضيح هذا الفحص، دعنا نأخذ مثالاً وهو رقم الكتاب العالمي (ISBN)، فهذا الرقم يتكون من عشر خانات، لو ضربنا كل خانة من الخانات التسع الأولى برتبتها وجمعنا هذه الأرقام وقسمنا الناتج على 11 فلن يبقى باقٍ. فمثلاً لو طبقنا المعادلة على هذا الرقم: 0-534-00615-9 كالآتي:

$$9X5+8X3+7X4+6X0+5X0+4X6+3X1+2X5+1X9 = 143$$

ثم قسمنا الناتج على 11 فسنحصل على الرقم 13 بدون باقٍ، عندها نتأكد من أن هذا الرقم صحيح حسب هذه المعادلة.

أسئلة التقويم الذاتي (3)

عدد وسائل التحقق، وبين ميزات واستخدامات كل منها.

3.3 عملية تصميم الشاشات

Input Screen Design Process

تتكون عملية تصميم شاشات الإدخال من عدة خطوات نجملها في الآتي:

1. ارسم الشاشة

المقصود برسم الشاشة، هو تصميم شكلها بدون برمجة. وهناك عدة طرق لهذا الغرض، وبإمكانك- عزيزي الدارس- اختيار واحدة منها. وهذه الطرق نجملها في النقاط التالية:

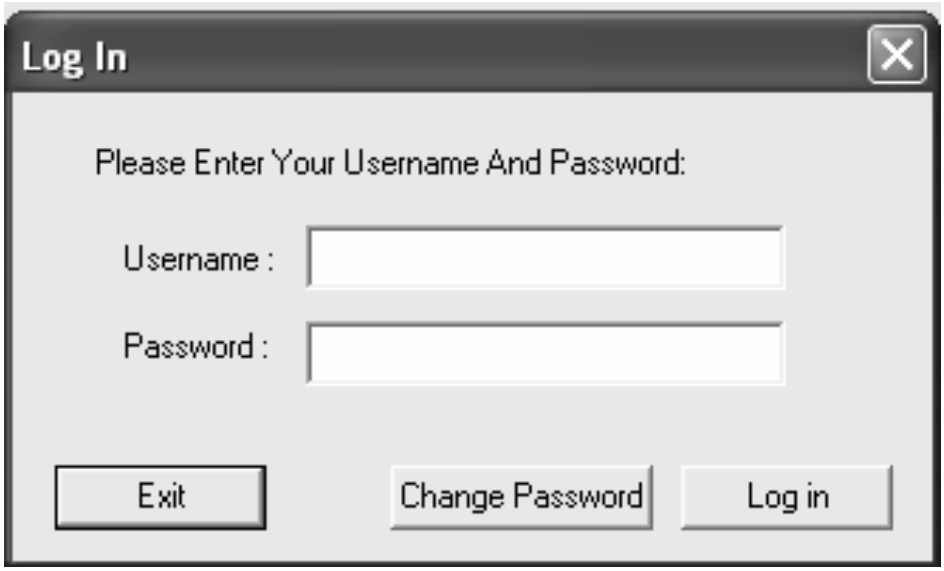
أ- استعمل ورقة الشاشة (Screen Sheet): وهي عبارة عن ورقة مربعات تحتوي على 24 سطراً و80 عاموداً. إذا اخترت هذه الطريقة، ارسم حرفاً واحداً في كل مربع وارسم كل شيء في موضعه. وتجدر الإشارة هنا بأن هذه الطريقة قديمة وبدائية.

ب- استعمل الحاسوب الشخصي: بإمكانك هنا استخدام برمجيات (Excel) أو (Word) لرسم الشاشة. وتعتبر هذه الطريقة أفضل من ورقة الشاشة وذلك لأنها تبين أشكال الحروف وأحجامها ومن الممكن تضمين رسومات عليها كشعار المؤسسة، هذا بالإضافة إلى كونها سهلة التعديل. ولكن من سيئاتها أن الشاشة المصممة بهذه

الطريقة لا تشبه الشاشة الحقيقية إلى درجة كبيرة، مما يؤدي إلى مشاكل مع المستخدم عند تصميم الشاشة الأصلية.

ج- استعمل الأدوات المساعدة (CASE TOOLS): تعتبر هذه الطريقة من أفضل الطرق لتصميم الشاشات، حيث من الممكن استخدام إحدى اللغات المرئية مثل (VB أو ORACLE). ومن ميزات هذه الطريقة أنه من الممكن بناء شاشة حقيقية تشبه تماماً الشاشة الأصلية، حيث من الممكن استخدامها كنموذج (Prototype) للمستخدم.

الشكل (1) يبين تصميماً مقترحاً لشاشة دخول للنظام تحتوي على معظم العناصر سالفة الذكر. والشكل (2) يبين شاشة بطاقة مادة لنظام المخازن من أحد الأنظمة التجارية.



الشكل (1): تصميم شاشة دخول للنظام

الشكل (2): تصميم لشاشة بطاقة مادة

2. صمم وثيقة وصف الشاشة (Screen Source Document)

تعتبر وثيقة وصف الشاشة من أفضل الطرق لوصف الشاشة وما تقوم بـــه مـــن وظائف للمبرمج، حيث أثبتت التجارب أنه وباستخدام هذه الوثيقة تصبح برمجة الـــشاشة أمراً ميسراً، قليل الأخطاء وكما يريده المحلل والمستخدم. وتتكون هذه الوثيقة من ترويسة (Header) تحتوي على اسم وعنوان الشاشة، واسم المحلل، والهدف من الشاشة وجدول يحتوي على وصف للحقول الموجودة على الشاشة.

الشكل (3) يبين وثيقة وصف شاشة دخول النظام المبينة في الشكل (1)، حيث:

- (Item): تعني اسم العنصر الموجود على الشاشة.
- (Item Type): نوع العنصر.
- (Size): حجم العنصر.

- (Format): نوع البيانات التي يحويها العنصر.
- (Default): القيمة الأولية أو الحالة التي يكون عليها العنصر عند فتح الشاشة.
- (User Action): طريقة تعامل المستخدم مع العنصر.
- (System Reaction): طريقة تصرف النظام كرد فعل على حركة المستخدم.
- (Tab Sequence): ترتيب العناصر عند الضغط على زر (Tab) على لوحة المفاتيح، حيث يستخدم زر (Tab) للتنقل من عنصر إلى آخر على الشاشة
- (Navigatable?): هل يسمح لمؤشر الشاشة بالوقوف على العنصر؟

Item	Item Type	Size	Format	Default	User Action	System Reaction	Tab Sequence	Navigatable? Y/N
User Name	Text Field	20 Chars	A(20)	Empty with cursor	Enter Data	Move Cursor to Password	1	Y
Password	Text Field	20 Chars	X(20)	Empty	Enter Data	--	2	Y
Exit	Button	0.5X2.5 Cm	--	Active	Press Button	Exit from System	4	N
Change Password	Button	0.5X3.5 Cm	--	Inactive	Press Button	Show Chng_psw form	5	N
Login	Button	0.5X2.5 Cm	--	Inactive	Press Button	Show Main form and hide login form	3	N

الشكل (3): وثيقة وصف شاشة دخول النظام المبينة في الشكل (1)

لو حاولنا قراءة محتويات الشكل (3)، لوجدنا أن الشاشة تحتوي على خمسة عناصر هي:
(Login, Change Password, Exit, Password, User Name).

فمثلاً (User Name): عبارة عن حقل نصي يتسع لعشرين حرفاً نصياً، تكون حالته الابتدائية عند فتح الشاشة فارغاً من البيانات ويظهر عليه مؤشر الفأرة. ويتعامل معه المستخدم بإدخال البيانات عليه عن طريق لوحة المفاتيح. وعند الانتهاء من تعبئته بالبيانات ينتقل مؤشر الفأرة إلى حقل (Password). وترتيب هذا العنصر هو الأول عند الضغط على زر (Tab). ويمكن أن يظهر عليه مؤشر الفأرة. وهكذا لبقية العناصر...

تخيل- عزيزي الدارس- لو أنك ستصف الشاشة إنشائياً، كيف سيكون الوضع وكمية المعلومات التي ستقوم بكتابتها. من هنا يتضح كم من الوقت والجهد نوفره عند استخدام وثيقة وصف الشاشة، وكيف سيكون الوضع سهلاً للمبرمج عند برمجة الشاشة.

4.3 إرشادات لتصميم الشاشات

Input Screens Design Guidelines

ينصح بإتباع النصائح والإرشادات التالية عند القيام بتصميم شاشة إدخال بيانات:

- يجب أن تكون شاشة الإدخال سهلة الاستخدام (User Friendly).
- يجب أن تكون عملية الإدخال سهلة وتفسر نفسها بنفسها.
- يجب أن تحتوي على رسائل الخطأ (Error Messages) ورسائل التنبيه (Warning Messages).
- يجب أن تحتوي على وسائل لاكتشاف الخطأ.
- يجب أن تحتوي على ترويسة للتعريف بالهدف من الشاشة.
- يجب أن تكون الشاشة بسيطة، أي أن تحتوي على عملية واحدة.
- يجب أن تكون الشاشة سهلة القراءة. أي أن تكون العناصر في نفس المكان في كل الشاشات.
- عند فتح الشاشة للبدء في استخدامها، احرص على وضع المؤشر على الحقل الذي سيبدأ المستخدم إدخال البيانات فيه.
- عند حدوث خطأ في أحد الحقول، احرص على تحريك المؤشر إلى بداية ذلك الحقل.
- قلل من استخدام المستخدم للوحة المفاتيح قدر الإمكان، أي اجعله يعتمد على الفأرة. إن ذلك يسهل على المستخدم استخدام النظام ويساعد على تقليل الأخطاء.
- حاول أن لا تسأل المستخدم عن نفس المعلومة أكثر من مرة.
- ميز بين الحقول الإجبارية والحقول الاختيارية على الشاشة، وذلك باستخدام ألوان مختلفة للحقول الإجبارية أو بوضع إشارة كالنجمة مثلاً بجانب هذه الحقول.
- يجب أن يفسر كل حقل نفسه: كإظهار شكل حقل التاريخ مثلاً عند وضع مؤشر الفأرة أو مؤشر الشاشة عليه تظهر رسالة مساعدة (Hint) تحتوي على شكل هذا الحقل كالآتي: (dd/mm/yyyy).
- استخدم طريقة متجانسة لرسائل الخطأ والتنبيه.
- استخدم شاشات المساعدة (Help) لمساعدة المستخدمين، واحرص على أن يكون استدعاؤها سهلاً بكبسة واحدة أو كبستين على الأكثر.

4. تصميم التقارير Output Reports Design

إن تصميم التقارير لا يقل أهمية عن تصميم الشاشات، وذلك لكثرة استخدامها وكثير من المستخدمين يقومون بالحكم على نجاح النظام أو فشله من خلال التقارير التي ينتجها النظام. لذلك لا بد من إعطاء العناية اللازمة لعملية تصميم التقارير.

ستقوم-عزيزي الدارس- من خلال هذا القسم من الوحدة بالتعرف على أنواع التقارير ودراسة خطوات عملية التصميم، بالإضافة إلى إرشادات عامة تساعد على تصميم تقارير جيدة.

1.4 أنواع التقارير Types of Reports

للتقارير عدة أنواع، ولكل نوع خصائص مختلفة ووظيفة معينة. ويمكن تلخيص هذه الأنواع كما يلي:

- التقارير الاستفسارية (Query Reports): التقرير الاستفساري عبارة عن تقرير ينتج عن جملة استفسارية بسيطة (Query Statement).
- التقارير الاستفسارية التفصيلية (Detail Reports): عبارة عن تقرير ينتج عن جملة استفسارية مركبة تربط أكثر من ملف من قاعدة البيانات.
- التقارير التلخيصية (Summary Reports): عبارة عن تقارير تحتوي على ملخصات لنتائج العمليات اليومية في الشركة التي يتم طلبها من الإدارة، لذا تعتبر هذه التقارير من التقارير الإدارية.
- التقارير الاستثنائية (Exception Reports): وهي التقارير التي تنتج عن وضع استثنائي غير مكرر، ولا يمكن التكهن بوقت حدوثه.
- التقارير الدورية (Periodic Reports): هي التقارير التي تصدر بشكل دوري، أي أن وقت طلبها يكون معلوماً بشكل مسبق. وكشف علامات الطالب يعتبر مثالاً على هذا النوع من التقارير، إذ إنه تتم طباعته في نهاية كل فصل دراسي.

أسئلة التقويم الذاتي (4)

عدد أنواع التقارير وبين استخدامات كل منها.

2.4 عملية تصميم التقارير Reports Design Process

هناك بنية عامة (Format) لكل تقرير لا نستطيع تجاوزها، فتصميم أي تقرير لا بد أن يحتوي على بعض أو على كل النقاط التالية، وهذا ما يسمى ببنية التقرير (Report Formatting):

- ترويسة التقرير (Report Heading): وهي ما يكتب في بداية التقرير ولمرة واحدة، كجامعة القدس المفتوحة في الشكل (4).
- ترويسة الصفحة (Page Heading): وهي ما يكتب على رأس كل صفحة من التقرير، كقائمة الطلاب والتاريخ في الشكل (4).
- الترويسة الفاصلة(Control Heading): وهي ما يفصل قائمة بيانات عن أخرى بحيث تكون هذه الترويسة مكتوبة في بداية هذه البيانات، كطلاب السنة الأولى وطلاب السنة الثانية في الشكل (4).
- اسم العامود (Column Heading): إذا كانت البيانات مرتبة على شكل جدول فلا بد من وجود اسم لكل عامود من أعمدة الجدول، كالاسم والعلامة في الشكل (4).
- السطر المفصل (A Detail Line): لا يجوز وجود معلومة منفصلة بدون اسم لهذه المعلومة، مثل "يبلغ عدد الطلاب الكلي: " في الشكل (4).
- الحاشية الفاصلة (Control Footing): هذه تشبه الترويسة الفاصلة ولكنها تُكتَب في نهاية البيانات، كنهاية قائمة أسماء طلاب السنة أولى ونهاية قائمة أسماء طلاب السنة الثانية.
- حاشية الصفحة (Page Footing): وهي ما يكتب ويكرر في نهاية كل صفحة من صفحات التقرير، مثل رقم الصفحة في الشكل (4).
- حاشية التقرير (Report Footing): وهي ما يكتب في نهاية التقرير ولمرة واحدة فقط، مثل نهاية القائمة في الشكل (4).

أسئلة التقويم الذاتي (5)

عدد عناصر بنية التقرير، وبين تعريف واستخدامات كل منها.

لا يشترط- عزيزي الدارس- أن يحتوي التقرير على كل هذه النقاط وإنما لا يجوز للتقرير أن يحتوي على غيرها. والشكل (4) يعتبر مثالاً على تقرير يحتوي على كل هذه النقاط وذلك بهدف التوضيح.

جامعة القدس المفتوحة قائمة الطلاب 2/6/2003 الإسم العلامة محمد علي 90 رغد صالح 85 نهاية طلاب السنة الأولى طلاب السنة الثانية: 1 of 2	قائمة الطلاب 2/6/2003 الإسم العلامة سمير راسم 70 داليا صالح 93 نهاية طلاب السنة الثانية يبلغ عدد الطلاب الكلي:40000 نهاية القائمة 2 of 2

الشكل (4): تقرير يحتوي على بنية كاملة

تتكون عملية تصميم التقارير من ثلاث خطوات رئيسة، ويمكن بيان تفاصيلها في النقاط الآتية:

الخطوة الأولى: ارسم كل صفحة من التقرير

يمكن رسم التقرير باستخدام القلم والورقة ولكن في هذه الطريقة يصعب التعديل على التصميم، ويمكن استخدام برمجية (Excel) أو ما شابهها حيث يكون تعديل التصميم سهلاً، كما يمكن استخدام إحدى البرمجيات المساعدة (CASE Tool) لهذا الغرض، كبرمجية فيجوال بيسك أو أوراكل.

عند الرسم يستحسن استخدام الرموز التالية:

- الخط العريض للدلالة على إطار التقرير.
- Xs للدلالة على الحروف والأرقام.
- As للدلالة على الحروف فقط.
- 9s للدلالة على الأرقام فقط.
- دلالة إلى المعلومات المطبوعة مسبقاً على ورقة التقرير، كشعار المؤسسة واسمها.
- الشَرطة (-) للدلالة على أماكن القطع.

الشكل (5) يبين مثالاً لتصميم تقرير استخدمت فيه الرموز آنفة الذكر.

XXXXXXXXXXXXXXXXXXXXXX
COURSE SCHEDULE

DATE:dd/mm/yyyy
TIME: hh:mm

Student Id: XXXXXXXX
Student Name: AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
Semester: XXXXXXXXX
Year: 9999
Degree: Major: AAAAAAAAAAAAAAA
AAAAAAAA

Course No.	Course Name	Section	Day	Time From	Time To	Credit Hours
9999	XXXXXXXXXXXXXX	9	AAA	99:99	99:99	9

End of Report

Page: 99/99

For Official Use Only

Name:AAAAAAAAAAAAAAAAAAA
Number: 9999999999

الشكل (5): مثال لتصميم تقرير

لاحظ –عزيزي الدارس– أنه يمكن استخدام صفحة منفصلة للدلالة على المعلومات المطبوعة مسبقاً على التقرير لتمييزها عن المعلومات التي سيقوم النظام بطباعتها. الشكل (6) يبين المعلومات المطبوعة مسبقاً على التقرير الوارد في الشكل (5).

COURSE SCHEDULE

DATE:
TIME:

_____

_____

For Official Use Only

الشكل (6): المعلومات المطبوعة مسبقاً على التقرير الوارد في الشكل (5)

الخطوة الثانية: معلومات التقرير

إن عمل محلل الأنظمة ليس فقط بناء برنامج يعمل على الحاسوب، بل بناء نظام كامل متكامل ببرمجياته وأجهزته وإجراءاته وقوانينه. وإن تصميم التقرير وحده غير كاف لبناء هذه العناصر، فلذلك لا بد من كتابة بعض المعلومات التي تصف التقرير وليست متضمنة في التصميم. ويستفيد من هذه المعلومات المبرمجون، وموظفو التشغيل

(Operators)، والمطبعة التي تطبع أوراق التقرير الفارغة. وتحتوي هذه المعلومات- عزيزي الدارس- على النقاط التالية:

- الترتيب: يتم ذكر اسم العامود الذي يتم ترتيب معلومات التقرير بناءً عليه إما تنازلياً أو تصاعدياً، ويستخدم في حالة وجود بيانات على التقرير على شكل جدول. تفيد هذه المعلومة المبرمجين.
- معدل عدد صفحات التقرير.
- معدل تكرار طباعته في السنة.
- عدد النسخ المطلوبة للتقرير الواحد. حيث ينتج من هذه النقطة والنقطتين السابقتين عدد الأوراق الواجب تجهيزها في بداية العام. وتفيد هذه المعلومات المطبعة، وموظف التشغيل، والمبرمج.
- طريقة النسخ (Duplication Method): المقصود هنا بيان الطريقة المراد عمل أكثر من نسخة فيها للتقرير. طريقة النسخ إما أن تكون باستخدام ورق كربون أو أن يكون ورق التقرير نفسه مكربناً أو باستخدام آلة النسخ.
- نوع الورق: يحدد بحجمه ووزنه ونسبة مادة الفايبر فيه.
- طريقة التوزيع: يطبع التقرير في بعض الأحيان على الطابعة الرئيسة الموجودة عند مشغل النظام، وإذا كانت هذه هي الحالة، فيجب معرفة طريقة توزيع التقرير بعد طباعته من قبل المشغل.
- درجة السرية: تستخدم هذه المعلومة للحفاظ على التقرير من العبث أو الوصول إليه بطريقة غير شرعية.
- طريقة التخلص منه (Disposal Method): تصف هذه النقطة كيفية التصرف بالتقرير بعد قراءته، كأن يحفظ، أو يُمزَّق أو يعلق على لوحات الإعلان مثلاً.

تدريب (1)

اقترح معلومات للتقرير الوارد في الشكل (5).

الخطوة الثالثة: ابن قاموس البيانات الخاص بالتقارير

يفيد قاموس البيانات بأنه يعطي تفسيراً لما يرد في التقرير من حقول وأسماء بيانات، وهو يفيد في عمليات البرمجة وصيانة النظام. وقاموس البيانات الخاص بالتقارير يشبه إلى حد كبير قاموس البيانات الخاص بقاعدة البيانات. والشكل (7) يبين قاموس بيانات للتقرير الوارد في الشكل (5).

Data Dictionary for Course Schedule

Item	Type	Length	Format	Notes
Date	Date	8	dd/mm/yyyy	Report issuing date
Time	Time	4	hh:mm	Report issuing time
Major	Alphabetic	15	A(15)	Student Major
Degree	Alphabetic	8	A(8)	Student Degree
Year	Numeric	4	9(4)	Academic Year
Course Name	Alphanumeric	15	X(15)	
Semester	Alphanumeric	9	X(9)	Current Semester
Student Id	Alphanumeric	8	X(8)	
Student Name	Alphabetic	35	X(35)	
Course No.	Numeric	4	9(4)	
Section	Numeric	1	9	
Day	Alphabetic	3	A(3)	Day Abbreviation: SUN, MON, TUE, WED, THU, FRI, SAT
Time From	Numeric	4	9(2):9(2)	Course Starting Time
Time To	Numeric	4	9(2):9(2)	Course Ending Time
Credit Hours	Numeric	1	9	Course Credit Hours
Page	Numeric	4	99/99	Page of Report with format: current page/total pages
Name	Alphabetic	21	A(21)	Registrar Name
Number	Numeric	10	9(10)	Registrar Number

الشكل (7): قاموس بيانات خاص بالتقارير

3.4 إرشادات لتصميم التقارير

Reports Design Guidelines

للحصول على تقارير جيدة ومفيدة ترضي المستخدم، ننصحك – عزيزي الدارس– بإتباع الإرشادات التالية عند تصميم التقارير:

- يجب أن يكون التقرير واضحاً ودقيقاً.
- يجب أن يكون التقرير سهل الفهم.
- يجب أن يكون التقرير مرتباً ترتيباً منطقياً، ككتابة اسم النظام أولاً في كشف العلامات ثم اسم التقرير يتبعه اسم الطالب وذلك قبل إدراج العلامات.
- يجب استخدام وسيلة الإخراج المناسبة، مثل الشاشة أو الطابعة أو غيرها من وسائل الإخراج
- يجب بناء وسائل لحماية التقرير من السرقة أو العبث به بطريقة غير شرعية، وتسمى هذه الوسائل بأنظمة الضبط والسيطرة وهي عبارة عن مجموعة من الإجراءات

لـــحماية الشيء من كل ما نخاف عليه منها، حيث من الممكن أن تكـــون يدويـــة أو آلية.

إن نظام الضبط والسيطرة الخاص بالتقارير عبارة عن مجموعة من الإجراءات تهدف إلى منع ظهور الأخطاء في التقرير ومنع الاحتيال وسرقة التقرير معنوياً ومادياً، ويحتوي على النقاط التالية:

- يجب أن يحتوي التقرير على أرقام الصفحات، ويفضل أن تكون على شكل page of Total، فمثلاً 3 of 20 تعني الصفحة الثالثة من أصل عشرين صفحة.
- يجب أن يحتوي التقرير على التاريخ والوقت.
- يجب أن يظهر اسم النظام على التقرير.
- يجب أن تحتوي بعض أنواع التقارير على الأرقام المتسلسلة بدلاً من أرقام الصفحات كدفاتر الشيكات مثلاً.
- يجب أن نبين طريقة توزيع كل تقرير.
- يجب أن نبين طريقة التخلص من التقرير بعد استعماله.
- يجب الاحتفاظ بالتقارير المهمة في أماكن أمينة.
- نسخ التقارير يجب أن تختلف عن الأصل.

5. الخلاصة

إن تصميم شاشات الإدخال والتقارير من أهم خطوات التصميم لأنهـــا تـــستخدم بشكل كبير من قبل المستخدمين ولا يكاد يرى غيرها من النظام، فهي غالباً مـــا تكـــون مصدر حكمه الوحيد على النظام، أي أنه من الممكن أن تفشل النظام أو تنجحه.

لذلك، لا بد من إتباع طرق تصميم جيدة للشاشات والتقارير، كما يجب أخذ رأي المستخدم في كل خطوة من خطوات تصميمها، لذلك يسمى تصميم واجهـــات الاســـتخدام بالتصميم الموجه للمستخدم.

لقد تم في هذه الوحدة– عزيزي الدارس– شرح طرق جيدة لتـــصميم واجهـــات الاستخدام، وقد تم بيان بعض الإرشادات الهامة للتصميم. وأخيراً، رأينا أنه لا بـــد مـــن التفكير بوسائل لحماية الشاشات من الاستخدام غير المشروع أو البيانات الخاطئة، وحماية التقارير من الوصول غير المشروع أو أن تحتوي على معلومات خاطئة.

6. لمحة عن الوحدة الدراسية التاسعة

تعنى الوحدة الدراسية التاسعة بشرح عملية تـــصميم الأصـــناف، وتـــساعدك- عزيزي الدارس- على فهم الطرق والتقنيات المتبعة لتصميم الأصناف بشكل جيد، حيـــث سنقوم بإعادة التطرق إلى مفاهيم الأصناف والكينونات والطرق التي تـــم شـــرحها فـــي الوحدات السابقة من هذا المقرر ومن ثم سنقوم بتقديم مجموعة من الطرق والمفاهيم التي تساعد على تصميم الأصناف ومعرفة جودة التصميم بالإضافة إلى تصميم الطرق الخاصة بتلكَ الأصناف.

7. إجابات التدريبات

تدريب (1)

- الترتيب: يرتب التقرير تصاعدياً حسب اسم الطالب.
- معدل عدد الصفحات: 3 صفحات X 1000.
- معدل التكرار: يطبع 3 مرات في السنة (1/1 و6/1 و 9/1).
- عدد النسخ: نسختين (نسخة للطالب ونسخة للقسم المعني).
- طريقة النسخ: ورق مكربن.
- نوع الورق: A4 80 Grams 70% Fiber.
- طريقة التوزيع: يسلم باليد لسكرتيرة القسم المعني.
- درجة السرية: عالية، لا يمكن الاطلاع عليه إلا من قبل الطالب المعني.
- طريقة التخلص منه: يمزق بعد الاستخدام.

8. مسرد المصطلحات

- **اسم العامود Column Heading:** إذا كانت البيانات مرتبة على شكل جدول فلا بد من وجود اسم لكل عامود من أعمدة الجدول.
- **التثبت Validation:** عبارة عن طريقة آلية برمجية للتأكد من صحة البيانات المدخلة، وتحتوي على العديد من الأنواع.
- **التحقق Verification:** التحقق عبارة عن طريقة يدوية يتبعها مدخل البيانات للتأكد من هذه البيانات عند إدخالها، فإما أن تكون بإعادة الإدخال مرتين أو أكثر لنفس البيانات، ومقارنتها بعضها ببعض، أو بالنظر كقيام المستخدم بالتأكد من هوية الزبون بالنظر في هويته الشخصية.
- **ترويسة التقرير Report Heading:** وهي ما يكتب في بداية التقرير ولمرة واحدة.
- **ترويسة الصفحة Page Heading:** وهي ما يكتب على رأس كل صفحة من التقرير.
- **الترويسة الفاصلة Control Heading:** وهي ما يفصل قائمة بيانات عن أخرى بحيث تكون هذه الترويسة مكتوبة في بداية هذه البيانات.
- **التصميم الموجه للمستخدم User-Centered Design:** التصميم الموجه للمستخدم يعني إشراك المستخدم في عملية التصميم بوجهين، فإما أن يشترك المستخدم مع المصممين في عملية التصميم ويكون جزءاً من فريق التصميم، وإمّا أن يقوم المستخدم بعملية تقييم التصميم بعد الانتهاء منه.
- **التقارير الاستثنائية Exception Reports:** وهي التقارير التي تنتج عن وضع استثنائي غير مكرر، ولا يمكن التكهن بوقت حدوثه.
- **التقارير الاستفسارية Query Reports:** التقرير الاستفساري عبارة عن تقرير ينتج عن جملة استفسارية بسيطة (Query Statement).
- **التقارير الاستفسارية التفصيلية Detail Reports:** عبارة عن تقرير ينتج عن جملة استفسارية مركبة تربط أكثر من ملف من قاعدة البيانات.
- **التقارير التلخيصية Summary Reports:** عبارة عن تقارير تحتوي على ملخصات لنتائج العمليات اليومية في الشركة التي يتم طلبها من الإدارة، لذا تعتبر هذه التقارير من التقارير الإدارية.
- **التقارير الدورية Periodic Reports:** هي التقارير التي تصدر بشكل دوري، أي أن وقت طلبها يكون معلوماً بشكل مسبق. وكشف علامات الطالب يعتبر مثالاً على هذا النوع من التقارير، إذ إنه تتم طباعته في نهاية كل فصل دراسي.

- **حاشية التقرير Report Footing:** وهي ما يكتب في نهاية التقرير ولمرة واحدة فقط.
- **حاشية الصفحة Page Footing:** وهي ما يكتب ويكرر في نهاية كل صفحة من صفحات التقرير.
- **الحاشية الفاصلة Control Footing:** هذه تشبه الترويسة الفاصلة ولكنها تُكتَب في نهاية البيانات.
- **حقل الخيار الأوحد الظاهر Radio Button:** نوع من أنواع حقول إدخال البيانات على الشاشة، ويستخدم لاختيار خيار واحد من عدة خيارات، وتكون هذه الخيارات عادة قليلة ومحدودة، وظاهرة جميعها على الشاشة. ومن ميزات هذا النوع أنه سهل الاستخدام، ولكنه جامد (Static) أي أنه لتعديل الخيارات يتوجب علينا تعديل البرمجة الخاصة به.
- **حقل الخيار الأوحد المخفي Combo Box:** نوع من أنواع حقول إدخال البيانات على الشاشة، ويستخدم لاختيار خيار واحد من عدة خيارات، وتكون هذه الخيارات عادة كثيرة، وغير ظاهرة على الشاشة عدا الخيار الأول. ومن ميزات هذا النوع أنه سهل الاستخدام، وغير جامد (Dynamic) أي انه لتعديل الخيارات لا يتوجب علينا تعديل البرمجة الخاصة به، ولكننا نستطيع ذلك من خلال التعديل على قاعدة البيانات.
- **حقل الخيارات المتعددة Check List:** نوع من أنواع حقول إدخال البيانات على الشاشة، ويستخدم لاختيار عدة خيارات تكون ظاهرة على الشاشة. ومن ميزات هذا النوع أنه سهل الاستخدام أيضاً، ولكنه جامد (Static) أي أنه لتعديل الخيارات يتوجب علينا تعديل البرمجة الخاصة به.
- **الحقل النصي Text Field:** نوع من أنواع حقول إدخال البيانات على الشاشة، ويستخدم هذا النوع لإدخال البيانات عن طريق لوحة المفاتيح. ومن سيئاته أنه يصعب التحكم بصحة البيانات المدخلة، لذلك تكون الأولوية عادة في اختيار نوع الحقل للأنواع الأخرى، ولا يستخدم هذا النوع إلا إذا اضطررنا لذلك.
- **السطر المفصل A Detail Line:** لا يجوز وجود معلومة منفصلة بدون اسم لهذه المعلومة.
- **فحص الخانات Check Digits:** عبارة عن عملية حسابية تنفذ على البيانات المدخلة ومقارنة نتائجها بنتيجة معلومة مسبقاً، فإذا كانت النتيجتان غير متطابقتين، نتأكد عندها من وجود خطأ في البيانات المدخلة. وهذه الطريقة تساعد على اكتشاف الأخطاء في الحقل المُدخَل.

- **نموذج الشاشة أو التقرير User Interface Prototype:** هو عبارة عن تصميم غير مبرمج للشاشة أو التقرير على الورق أو باستخدام أحد البرمجيات سهلة الاستخدام مثل فيجوال بيسك أو أوراكل، يراها المستخدم من أجل تقييمها وأخذ رأيه فيها، وعندها يمكن للمستخدم أن يغير ما يشاء بدون جهد أو وقت أو تكاليف، وعند الاتفاق على الشكل النهائي للشاشة أو التقرير، يقوم المصمم ببرمجتها.

9. المراجع

1. Aktas, Z., Structured Analysis and Design of Information Systems, Prentice-Hall, 1987.
2. Baudoin, C. and Hollowell, G., Realizing The Object-Oriented Life Cycle, Prentice-Hall, 1996.
3. Booch, G., Rumbaugh, J., and Jacobson, I., The Unified Modeling Language User Guide, Addison- Wesley, 1999.
4. Dennis, A. et al, Systems Analysis & Design: An Object-Oriented Approach with UML. John Wiley & Sons Inc., 2002.
5. Dewitz, S., Systems Analysis and Design and the Transition to Objects, McGraw-Hill, 1996.
6. Edwards, P., Systems Analysis, Design, and Development with Structured Concepts, Rinehart And Winston, 1999.
7. Henderson, P., Object-Oriented Specification and Design with C++, McGraw -Hill, 1993.
8. Hoffer, J. et al., Modern Systems Analysis & Design, 2nd Ed., Addison-Wesley, 1999.
9. Kendal, K., and Kendal, J., Systems Analysis and Design, 5th Ed. 2002.
10. Kruchten, P., The Rational Unified Process: An Introduction. 2nd Ed., Addison - Wesley, 2000.
11. Lejk, M. and Deeks. An Introduction to Systems Analysis Techniques, 2nd Ed., Pearson Education Limited, 2002.
12. Maciaszek, A., Requirements Analysis and System Design, Developing Information Systems with UML, Addison- Wesley, 2001.
13. Quatrani, T., Visual Modeling with Rational Rose 2000 and UML, Addison- Wesley, 2000.
14. Satzinger, J., Systems Analysis and Design, 2nd Ed., Thomson Learing, 2002.
15. Silver, G. and Silver, M., Systems Analysis and Design, Addison-Wesley, 1996.
16. http://www.psut.edu.jo / computer science/ staff/saleh abu-soud
17. www.rational.com
18. www.omg.com

الوحدة التاسعة

تصميم الأصناف

Classes Design

محتويات الوحدة

الموضوع	الصفحة
1. المقدمـــة	211
1.1 تمهيد	211
2.1 أهداف الوحدة	211
3.1 أقسام الوحدة	212
4.1 القراءات المساعدة	212
5.1 ما تحتاج إليه لدراسة الوحدة	212
2. الأصناف: الصفات والطرق Classes: Attributes and Methods	213
3. التوريث Inheritance	214
4. تعدد الأوجه Polymorphism	215
5. الارتباط الديناميكي Dynamic Binding	216
6. معايير جودة التصميم	217
1.6 التغليف وإخفاء المعلومات Encapsulation and Information Hiding	217
2.6 الترابط بين الأجزاء Coupling	218
3.6 معيار التماسك Cohesion	222
7. توصيف الطرق Methods Specification	223
8. الخلاصة	226
9. لمحة عن الوحدة الدراسية العاشرة	227
10. إجابات التدريبات	227
11. مسرد المصطلحات	229
12. المراجع	230

1. المقدمة

1.1 تمهيد

أهلاً بك – عزيزي الدارس– إلى الوحدة التاسعة من مقرر "1380 تحليل الأنظمة وتصميمها"، وهي بعنوان "تصميم الأصناف".

يعد تصميم الأصناف من أهم خطوات مرحلة التصميم، حيث إن النظم المعتمدة على الكينونة من الممكن أن تكون معقدة كثيراً ، ولكن بوجود تصميم جيد للأصناف يصبح عمل المحلل والمصمم سهلاً جداً وميسراً، علاوة على أننا سنحصل على نظام فعال سهل الفهم والبناء وتكون صيانته سهلة، بالإضافة إلى كونه مرناً من السهل التعديل عليه.

تساعدك هذه الوحدة – عزيزي الدارس– على فهم الطرق والتقنيات التي تساعدك على تصميم الأصناف بشكل جيد، حيث سنقوم بإعادة التطرق إلى مفاهيم الأصناف والكينونات والطرق التي تم شرحها في الوحدات السابقة من هذا المقرر ومن ثم سنقوم بتقديم مجموعة من الطرق والمفاهيم التي تساعد على تصميم الأصناف ومعرفة جودة التصميم بالإضافة إلى تصميم الطرق الخاصة بتلكَ الأصناف.

2.1 أهداف الوحدة

يتوقع منك – عزيزي الدارس – بعد الانتهاء من دراسة هذه الوحدة أن تكون قادراً على أن:

1) توضح أهمية تصميم الأصناف.
2) تشرح مفاهيم التوريث وتعدد الأوجه والارتباط الديناميكي.
3) تستعمل معايير جودة التصميم.
4) تذكر طرق توصيف الأصناف.

3.1 أقسام الوحدة

تقسم هذه الوحدة إلى ستة أقسام رئيسة:

القسم الأول: يبين أهمية تصميم الأصناف وطرقه وهذا يغطي الهدف الأول. أما القسم الثاني: فيبين آلية التوريث وكيفيتها ويغطي الهدف الثاني. في حين يوضح القسم الثالث: مفهوم تعدد الأوجه، وهذا يغطي الهدف الثاني. أما القسم الرابع: فيوضح مفهوم الارتباط الديناميكي ويغطي الهدف الثاني أيضاً. ثم يأتي القسم الخامس: ليبيّن معايير جودة التصميم وهو بذلك يغطي الهدف الثالث. وأخيراً يأتي القسم السادس: ليناقش كيفية توصيف الطرق محققاً بذلك الهدف الرابع.

4.1 القراءات المساعدة

أنصحك، عزيزي الدارس، بالإطلاع على المراجع التالية لما تحويه من معلومات مفيدة تتعلق بموضوع هذه الوحدة:

1. Edwards, P., Systems Analysis, Design, and Development with Structured Concepts, Rinehart and Winston, 1999.
2. Kendal, K., and Kendal, J., Systems Analysis and Design, 5th Ed., 2002.
3. Silver, G. and Silver, M., Systems Analysis and Design, Addison-Wesley, 1996.
4. Aktas, Z., Structured Analysis and Design of Information Systems, Prentice-Hall, 1987.
5. Baudoin, C. and Hollowell, G., Realizing The Object-Oriented Life Cycle, Prentice- Hall, 1996.
6. Dewitz, S., Systems Analysis and Design and the Transition to Objects, McGraw-Hill, 1996.

5.1 ما تحتاج إليه لدراسة الوحدة

قبل أن تبدأ – عزيزي الدارس– بدراسة هذه الوحدة تأكد من أنك قمت بتهيئة المكان الهادئ المناسب للدراسة مما يساعدك على التركيز والتفكير العميق، ثم احرص على تتبع الإرشادات التي توجهك لدراسة الوحدة، وحاول بجدية الإجابة عن أسئلة التقويم الذاتي، وحل التدريبات، لأنها تساعدك على مراجعة موضوعات الوحدة، وتعمق معلوماتك حولها.

2. الأصناف: الصفات والطرق
Classes: Attributes and Methods

إن البنية الأساسية- عزيزي الدارس- لأي نظام هي الكينونة (Object)، حيث إن الكينونة عبارة عن مَثَل (Instance) يتم اشتقاقه من الصنف الذي بدوره يوضح طريقة تصنيع هذه الكينونات. وإن الأصناف هي التي تُعرف البيانات والعمليات التي تحتويها كل كينونة، وكل كينونة تحتوي على صفات توضح الحالات التي تمر فيها الكينونة، حيث إن كل كينونة لها حالة (State) وتحدد هذه الحالة حسب قيمة البيانات المخزنة في صفاتها في لحظة معينة، وكل كينونة في النظام تحتوي على عدة طرق والتي بدورها تحدد العمليات التي تستطيع أن تقوم بها هذه الكينونة.

مثال (1)

لو فرضنا أن لدينا صنفاً للطالب اسمه(Student) كما هو مبين في الشكل (1)، حيث يحتوي الصفات الآتية:

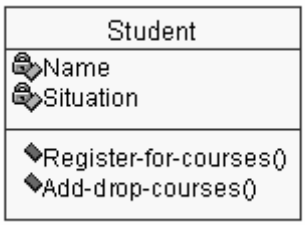
(Name): ويمثل اسم الطالب

(Situation): ويمثل وضع الطالب الأكاديمي (مسجل، منسحب، مفصول)

ويحتوي الطرق الآتية:

(Register-for-courses): وتعني تسجيل المواد.

(Add-drop-courses): وتعني سحب وإضافة مواد.



الشكل (1): الصنف Student

الآن، وعلى سبيل المثال، لو كانت قيمة الصفة (Name) لطالب معين هي محمد وقيمة الصفة (Situation) مسجل، عندها نقول: إنه قد تم اشتقاق كينونة من هذا الصنف للطالب محمد، ويستطيع هذا الطالب أن يقوم بتنفيذ الطرق الموجودة في الصنف الذي تم

اشتقاقه منه؛ أي يستطيع أن يسجل مواد ويسحب ويضيف مواد. لاحظ – عزيزي الدارس– أن حالته في هذه اللحظة هي قيم صفاته أي محمد ومسجل، ولكن لو تغيرت قيمة الصفة (Situation) مثلاً إلى مفصول، عندها تصبح حالته الجديدة: محمد ومفصول وهكذا ...

3. التوريث Inheritance

على الرغم من أننا نستطيع تعريف الأصناف بشكل منفصل، بحيث يحتوي كل صنف على صفاته وطرقه الخاصة به، وإذا لزمت هذه الصفات أو الطرق لأصناف أخرى نقوم بالتعريف مجدداً لكل الأصناف التي تحتاج إليها، إلا أن هذه الطريقة تزيد من صعوبة بناء النظام وصيانته وتعديله، لأنها تزيد من الصفات والطرق المكررة رغم أنها تعود إلى نفس المصدر.

الحل لهذه المشكلة يعود إلى إعادة استخدام الأصناف المعرفة مسبقاً كأساسات للأصناف الحالية، بحيث نستطيع استخدام صفات وطرق هذه الأصناف المعرفة مسبقاً للأصناف الجديدة دون اللجوء إلى إعادة تعريفها مرة ثانية. وتسمى هذه العملية بالتوريث، حيث تعني أن الأصناف المعرفة مسبقاً قد ورّثت صفاتها وطرقها للصفات الجديدة.

تتم عملية التوريث بتعريف صنف رئيس يحتوي على الصفات والطرق التي تحتاجها الأصناف الفرعية، ومن ثم نجعل هذه الأصناف تقوم بوراثة هذه الصفات والعمليات من الصنف الرئيس.

هناك العديد من آليات التوريث، لكن أكثر الطرق شهرةً إما الوراثةً بشكل فردي (Single Inheritance) أو الوراثة بشكل متعدد (Multiple Inheritance). ويتيح نظام التوريث بشكل فردي للصنف الفرعي بالوراثة من صنف رئيس واحد فقط، أما الوراثة بشكل متعدد فيتيح للصنف الفرعي بالوراثة من أكثر من صنف رئيس، كالآلية البرمائية التي ترث بعض صفاتها وطرقها من السيارة وترث بعضها الآخر من السفينة، فهي لها دواليب وتسير على اليابسة وفي نفس الوقت تطفو على سطح الماء. ومن الجدير بالذكر أن معظم الطرق البرمجية ونظم قواعد البيانات تتيح استخدام نظام التوريث الفردي فقط.

أسئلة التقويم الذاتي (1)

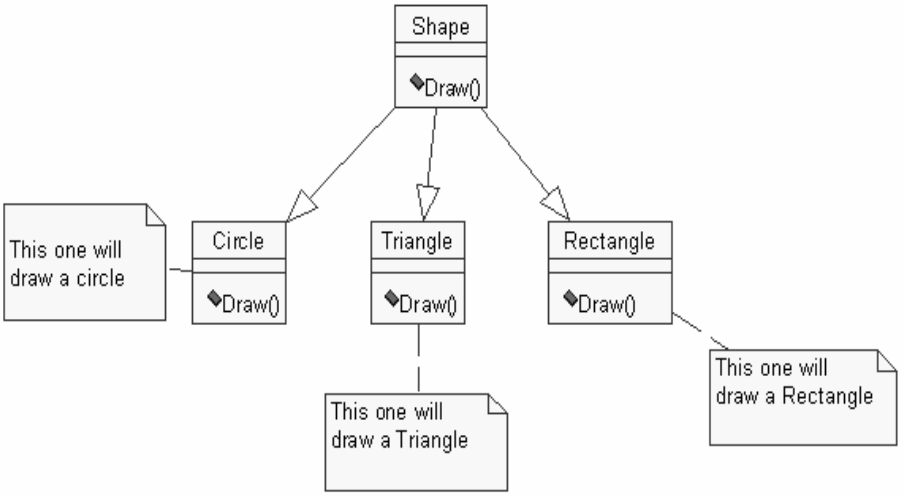
ما الفرق بين التوريث الفردي والتوريث المتعدد؟

4. تعدد الأوجه Polymorphism

يحدث تعدد الأوجه في حالة وجود كينونة رئيسة لها كينونة فرعية أو أكثر، بحيث تحتوي الكينونة الرئيسة على طريقة ما، وتحتوي بعض الكينونات الفرعية على نفس هذه الطريقة ولكن بتطبيق مختلف حسب متطلبات الكينونة الموجودة فيها. وتكون الطريقة الموجودة في الكينونة الرئيسة مجردة (Abstract) أي أنها غير قابلة للتطبيق لأنها عبارة عن واجهة اتصال فقط (Interface)، بينما تعتبر الطرق الموجودة في الكينونات الفرعية طرقاً مطبقة بالكامل ويمكن مناداتها مباشرة.

ويساعد هذا المفهوم على توحيد معايير التحليل والتصميم والبرمجة، كمصنع يصنع أجهزة كهربائية مثل الراديو، والتلفاز، وجهاز الخلوي. ومن المعروف أن كل الأجهزة الكهربائية لها طريقة تشغيل، ولكن تختلف هذه الطريقة من جهاز إلى آخر. ولتطبيق مفهوم تعدد الأوجه، يقوم المحلل بتعريف كينونة رئيسة ويسميها مثلاً الأجهزة الكهربائية ويضع فيها طريقة مجردة تدعى على سبيل المثال، تشغيل، ويعرّف كينونة فرعية لكل جهاز ويضع في كل واحدة منها طريقة لتشغيل الجهاز المعني.

يبين الشكل (2) مثالاً على تعدد الأوجه، حيث يحتوي الشكل على كينونة رئيسة: Shape وثلاثة كينونات فرعية: (Circle, Triangle, Rectangle)، وكل من هذه الكينونات تحتوي على الطريقة (() Draw)، ولكن عند استدعاء هذه الطريقة فإنها ترسم الشكل المحدد في الكينونة الموجودة فيها؛ فعند استدعاء هذه الطريقة الموجودة في الكينونة (Circle) فإنها تقوم برسم دائرة، وهكذا ...



الشكل (2) مثال على تعدد الأوجه

5. الارتباط الديناميكي Dynamic Binding

الارتباط الديناميكي يتضمن قدرة النظام المبني على الكينونة بالرجوع إلى البيانات ومعرفة نوعها في وقت التنفيذ. افترض- على سبيل المثال عزيزي الدارس- أن لديك مصفوفة من كينونات مشتقة من صنف طالب، ويمكن أن تتضمن هذه المصفوفة طلاباً منسحبين وطلاباً بدوام جزئي وطلاباً بدوام كامل، وافترض أنه ولجميع أنواع الطلاب توجد طريقة أو وظيفة متماثلة يمكن للطالب أن ينفذها من خلال النظام مثل تسجيل المواد الجامعية.

يجب على النظام أن يكون قادراً على اختيار الطريقة المناسبة لتنفيذ هذه الوظيفة بحسب نوع الطالب خلال وقت التنفيذ لأن الطريقة التي تستخدم لتسجيل مواد لطالب بدوام جزئي تختلف عن تسجيل مواد لطالب بدوام كلي أو منسحب؛ فالطالب المنسحب، على سبيل المثال، لا يستطيع تسجيل مادة خلال فترة الانسحاب، وكذلك الطالب ذو الدوام الجزئي لا يستطيع اختيار كل المواد المتاحة.

تدعم أغلب لغات البرمجة الحديثة مفهوم الارتباط الديناميكي، لكنّها تختلف من لغة برمجة إلى أخرى في كيفية تطبيق هذا المفهوم. لذا يجب اختيار لغة البرمجة التي تدعم الارتباط الديناميكي في حال أن مصمم النظام اعتمد على فكرة تعدد الأوجه.

وقد يكون مفهوم تعدد الأوجه سلاحاً ذا حدين لأن الطريقة أو الوظيفة التي سيتم استدعاؤها غير معروفة إلا في وقت التنفيذ، مما يجعل من الصعب التكهن بالطرق المستدعاة ومما قد يحدث أخطاء في وقت التنفيذ وذلك حين يتم استدعاء وظائف غير مبرمجة في النظام أو غير معدة مسبقاً.

أسئلة التقويم الذاتي (2)

بين الأسباب والظروف التي تدعو إلى وجود الارتباط الديناميكي في اللغات المعتمدة على مفهوم تعدد الأوجه.

6. معايير جودة التصميم

إن لطريقة تصميم النظام أثراً كبيراً على أمور كثيرة في النظام، سواءً الآثار الداخلية مثل الفعالية والسرعة والخارجية مثل سرعة البناء ودرجة فهم النظام من قبل المبرمجين وفريق الصيانة.

ووضع العلماء نقاطاً إرشادية لتصميم نظام جيد يمكن تلخيصها بأربع نقاط كالآتي:

1. يجب أن يكون النظام مجزأ (Modular) وهذا يعني عدم تصميم النظام ككتلة واحدة بل يفضل تقسيمه إلى برامج أصغر، حتى تسهل السيطرة عليه.
2. يجب أن تكون هذه الأجزاء البرمجية في عدة مستويات (Hierarchical)، أي يستدعي بعضها الآخر.
3. يجب أن يكون هذا الاستدعاء منطقياً (Logical)، أي أن لا يستدعي الجزء جزءاً آخر إلا عند الحاجة.
4. يجب أن تكون العلاقة بين الأجزاء أقل ما يمكن (Modularly Independent).

والحقيقة، إن طريقة الكينونة تساعد جداً على بناء الأنظمة بهذه المواصفات، ويعود السبب في ذلك – عزيزي الدارس– إلى كون هذه الطريقة تجبر مستخدمها على استخدام الأصناف، والكينونات، والطرق، حيث تعتبر هذه هي أجزاء النظام. والواقع أن هناك العديد من معايير جودة التصميم سنتطرق في البنود الآتية إلى أهمها.

1.6 التغليف وإخفاء المعلومات

Encapsulation and Information Hiding

التغليف عبارة عن آلية تستخدم لوضع الصفات والطرق في كينونة واحدة. بينما يعرف إخفاء المعلومات بأنه عملية الاحتفاظ بالصفات والطرق التي تتطلب استخدامها في كينونةً معينة ولا يجوز للكينونات الأخرى الوصول إليها واستعمالها إلا إذا لزم الأمر لذلك، وفي هذه الحالة يكون الاستخدام باستدعاء هذه الطرق ولا ضرورة لتعريفها وكتابتها مرة أخرى في هذه الكينونات، بل يجب أن تكون أسماء هذه الطرق معروفة لها حتى تستطيع استدعاءها.

وبذلك يكون التواصل بين الكينونات فقط من خلال استدعاء الطـرق الموجـودة فيها. وهذا يؤدي إلى دعم فكرة إعادة استعمال الأصناف (Reusability) ودعـم حمايـة العمل الداخلي للطرق في الكينونات من التغيير من قبل أي جهة خارجية.

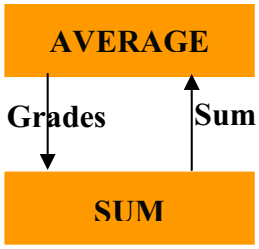
2.6 الترابط بين الأجزاء Coupling

يحدد هذا المعيار درجة اعتمادية أجزاء النظام؛ والأصناف والكينونات والطرق؛ على بعضها البعض، حيث إنه كلما زاد الترابط بين الطرق كلما زاد تعقيد النظام وأضحى التعامل معه صعباً، وإذا كان الوضع كذلك، سيصبح النظام صعب الفهم والتتبع، كمـا أن صيانته تصبح صعبة بل غير ممكنة في كثير من الأحيان، وسيغدو النظام غيـر مـرن للتغييرات؛ إذ سيصبح من الصعب إضافة أجزاء جديدة للنظام أو إلغاء أجزاء من النظـام أو تعديل أجزاء موجودة؛ وذلك لأن أي تعديل على جزء يتطلب تعديلاً على كثيـر مـن الطرق الأخرى.

وللترابط نوعان: الترابط التفاعلي، والترابط الناتج عـن التوريـث. وبالنـسبة للترابط التفاعلي ينتج عن تبادل البيانات والأوامر بين الطرق. وهناك ستة أنواع للتـرابط التفاعلي تعتمد في تصنيفها على كمية وطبيعة البيانات المتبادلة بين الطرق. وهذه الأنواع هي كالآتي:

1. اعتمادية البيانات الضرورية (Data Coupling)

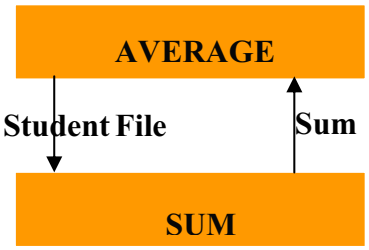
تسمى الاعتمادية بين جزأين برمجيين اعتمادية بيانات إذا تبـادل هـذان الجـزءان البيانات الضرورية بينهما، ويعتبر هذا النوع أقل أنواع الاعتمادية سوءاً. الشكل (3) يبين مثالاً لهذا النوع من الاعتمادية، حيث يستدعي البرنامج الفرعي (AVERAGE) البرنامج الفرعي (SUM) فيرسل له العلامات ويعيد بدوره النتيجة (SUM).



الشكل (3): اعتمادية البيانات الضرورية

2. اعتمادية البيانات غير الضرورية (Stamp Coupling)

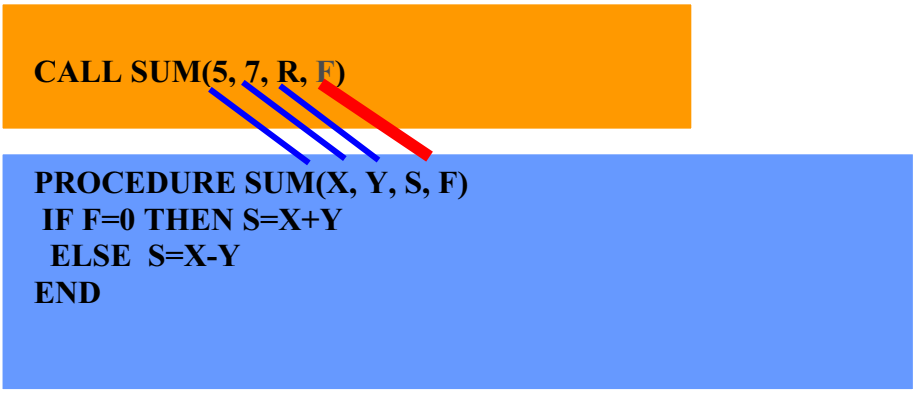
يحدث هذا النوع إذا تبادل جزءان بيانات غير ضرورية بينهما، كما هو مبين في الشكل (4) حيث يستدعي البرنامج الفرعي (AVERAGE) البرنامج الفرعي (SUM) وبدلاً من أن يرسل له العلامات يقوم بإرسال الملف كاملاً بما يحويه من بيانات. وفي هذه الحالة، تقل فعالية النظام لنقل معلومات أكثر من اللازم إضافة إلى أننا أعطينا البرنامج الفرعي (SUM) – من دون أن نقصد– الفرصة ليغير بيانات الملف كلها.



الشكل (4): اعتمادية البيانات غير الضرورية

3. اعتمادية التحكم (Control Coupling)

يحدث هذا النوع حين يتحكم جزء بالمنطق الداخلي لجزء آخر، وذلك عن طريق إرسال متغير من الجزء المستدعي إلى الجزء المستدعى. والشكل (5) يبين مثالاً لهذا النوع، حيث يتم فيه استدعاء جزء يسمى (SUM) من قبل جزء آخر، فيتحكم الجزء المستدعي بالجزء المستدعى عن طريق المتغير (F) الذي تم تمريره بطريقة رسمية (Formal) عن طريق معاملات الجزء (Parameters).



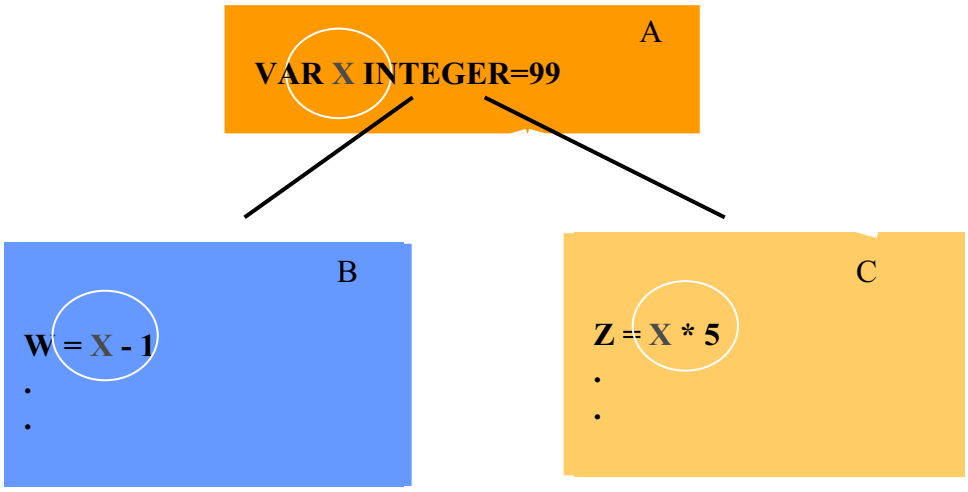
الشكل (5): اعتمادية التحكم

من الواضح- عزيزي الدارس- أن هذا النوع يعتبر أكثر سوءاً من الأنواع الأخرى، وذلك لأن فهم الجزء المستدعى وتعديله وصيانته تكون صعبة وعلينا تتبع الطرق المستدعية وفهمها جيداً قبل القيام بهذا العمل.

4. اعتمادية المشاركة (Common Coupling)

يتم هذا النوع إذا اشترك جزءان أو أكثر في استعمال نفس البيانات. وفي هذه الحالة إذا حدث خطأ أو خلل في هذه البيانات، فإن هذا الخلل سينتقل إلى كل الطرق التي تستخدم هذه البيانات، وسيكون من الصعب اكتشاف هذا الخلل لأنه سيكون قد انتشر بين الطرق.

والشكل (6) يبين مثالاً لهذا النوع من الترابط، حيث يبين هذا الشكل ثلاثة أجزاء (A و B و C). حيث تم تعريف المتغير (X) في الجزء (A) وأعطي القيمة 99، وتم استخدامه في بقية الطرق. فلو فرضنا أن قيمة هذا المتغير هي 9 وكتبت خطأً 99 فإن هذا الخطأ سينتقل إلى بقية الطرق وسيتم استخدام القيمة 99 بدلاً من القيمة الصحيحة وسيكون من الصعب اكتشاف ذلك، وإذا تم اكتشافه سيكون من الصعب وفي بعض الأحيان من المستحيل تصحيح الوضع.



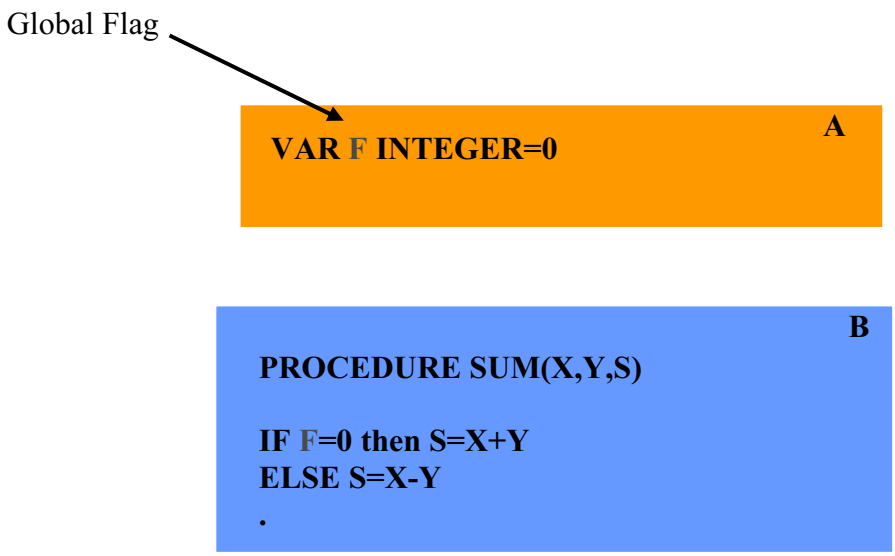
الشكل (6): اعتمادية التشارك

إذاً من السهل جداً عزيزي الدارس أن تستنتج أن هذا النوع من الارتباط يعتبر أسوأ من الأنواع السابقة.

5. اعتمادية المحتوى (Content Coupling)

يعتبر هذا النوع هو الأسوأ مقارنة مع كل الأنواع الأخرى، حيث يتحكم جزء بالمنطق الداخلي لجزء آخر عن طريق استخدام متغير معرف في جزء آخر؛ وربما يختلف عن الجزء المستدعي؛ بدون تمريره في جملة الاستدعاء. ويعود سبب اعتبار هذا النوع كأسوأ الأنواع إلى صعوبة فهم وتتبع وصيانة الجزء المستدعى لتحكم متغير موجود في جزء آخر ربما يكلفنا الكثير من الجهد، ويمكن أن نحتاج إلى تتبع النظام كله، حتى نجد هذا المتغير.

الشكل (7) يبين مثالاً على اعتمادية المحتوى حيث يتضح من الشكل أن هناك متغير (F) يتحكم بالمنطق الداخلي للجزء (B) من دون أن نعرف من أين أتى أو في أي جزء تم تعريفه وإعطاءه قيمة، مما يزيد من صعوبة تتبع وتعديل الجزء (B). وقد يتطلب الأمر جهداً كبيراً، خاصة إذا كان النظام ضخماً، لاكتشاف أن هذا المتغير قد تم تعريفه في الجزء (A)، وبالطبع هذا يتنافى مع مبدأ تغليف وإخفاء المعلومات.



الشكل (7): اعتمادية المحتوى

أما النوع الثاني من الترابط فهو ترابط التوريث. وهذا النوع من الترابط، كما هو واضح من الاسم، يبين ترابط الأصناف في الهيكل التنظيمي الوراثي، كما مر معنا –عزيزي الدارس– عند تعريف الوراثة والترابط الديناميكي. إذ يجب أن تكون درجة التوريث معتدلة وحسب المطلوب لا أكثر ولا أقل، وعلى المحلل أن يوازن بين التوريث وعدم التوريث.

3.6 معيار التماسك (Cohesion)

إن معيار التماسك عبارة عن مقياس للدرجة التي ينفذ فيها الصنف أو الطريقة وظيفة واحدة معروفة ومفهومة، وكلما كان هذا المعيار أقوى، كلما كان النظام أفضل.

حدد العلماء للأنظمة التي تصمم باستخدام الكينونة نوعين من التماسك هما: تماسك الطرق وتماسك الأصناف، حيث يمثل تماسك الطرق قدرة الطريقة على أن تكون منفردة وتنفذ فعلياً عملاً واحداً فقط، وذلك أن الطريقة التي تنفذ العديد من الوظائف تكون أكثر تعقيداً للفهم والمتابعة والفحص والصيانة من تلك التي تنفذ وظيفة واحدة فقط . بينما يعتبر تماسك الأصناف معياراً يعتمد على مستوى التناسق فيما بين الخصائص والطرق في الصنف الواحد، بمعنى أن الصنف يجب أن يمثل شيئاً واحداً، مثل موظف أو قسم وذلك يتضمن أن كل الخصائص والطرق المحتواة في صنف واحد يجب أن تتعلق بذلك الصنف.

فصنف الموظف، على سبيل المثال، يمكن أن يحتوي على خصائص مثل الرقم الوطني، والاسم، والعنوان ولكن ليس من المنطق أن نحدد له خصائص مثل المتطلب السابق أو قوة المحرك لأنها تخص أصنافاً أخرى.

7. توصيف الطرق (Methods Specification)

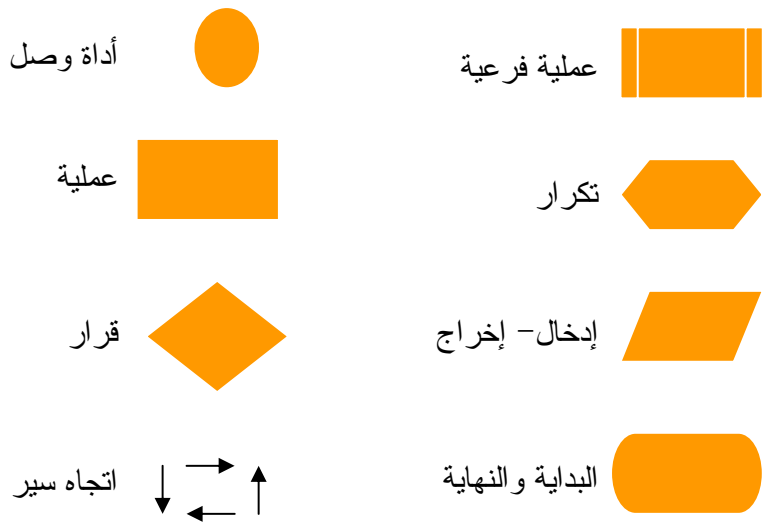
يتم تصميم الأصناف وتحديد الطرق في مرحلة التصميم، وتتم برمجتها في مرحلة البناء حسب تعليمات المصمم، إذ لا يجوز الاعتماد على فريق المبرمجين بالاجتهاد في تحديد منطق البرامج والطرق حسب أهوائهم أو ما يرونه مناسباً. لهذا السبب ينبغي للمصمم توصيف المنطق الداخلي للطرق قبل تسليمها لفريق البرمجة.

ويمكن أن يتم هذا التوصيف إما بطريقة محلية يتفق عليها المصمم والمبرمجون أو بطريقة معيارية عالمية. والواقع يجب تشجيع استخدام الطريقة الثانية لأنها معروفة ومستخدمة على نطاق واسع مما يساعد على العمل ضمن المعايير العالمية. وسيتم، في هذا القسم- عزيزي الدارس- مناقشة ثلاث طرق عالمية للتوصيف تستخدم بشكل واسع بين المحللين ومصممي الأنظمة هي: خرائط سير البرامج (Programs Flowcharts)، وخوارزميات اللغة الطبيعية (Structured English Algorithms)، وأشباه البرامج (Pseudo (Code.

1. خرائط سير البرامج (Programs Flowcharts)

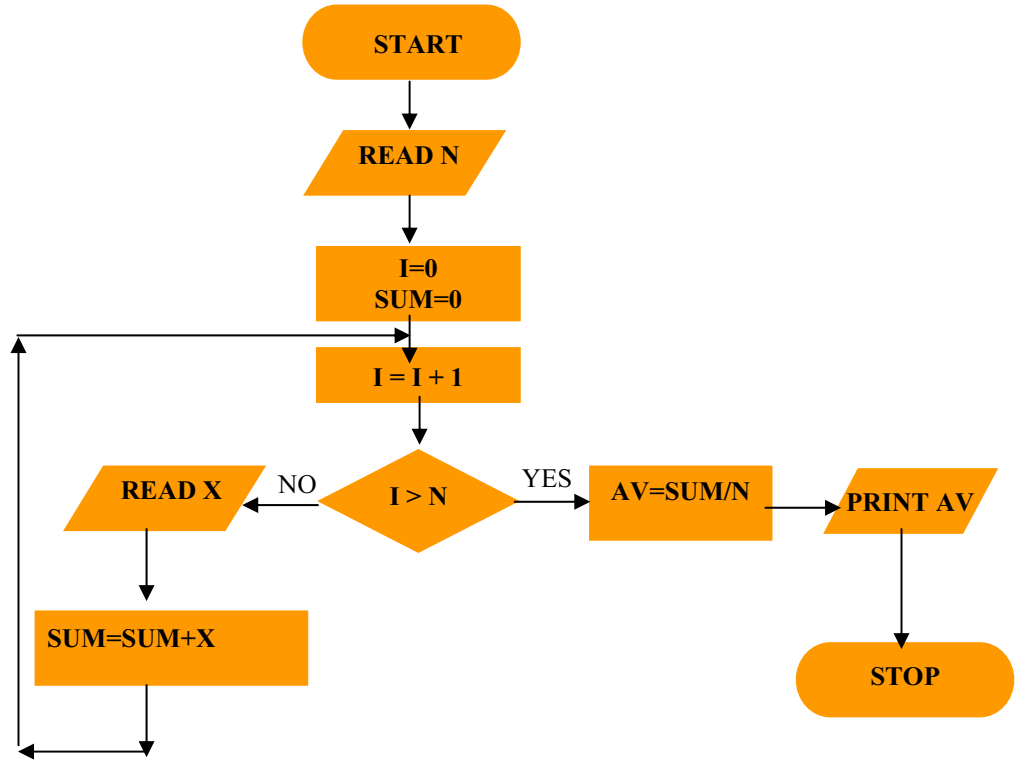
إن استخدام خرائط سير البرامج يساعد على سهولة تتبع البرنامج واكتشاف الأخطاء وتصحيحها، ويساعد كذلك على تعديل البرامج وصيانتها في غياب المبرمج. وتعتبر خرائط سير البرامج من الأدوات المهمة لتوثيق العمليات التفصيلية للبرامج، وتسهل البرمجة باستخدام إحدى لغات البرمجة.

ولخرائط سير البرامج رموز وأشكال معيارية عالمية. الشكل (8) يبين بعض هذه الرموز.



الشكل (8): بعض رموز خرائط سير البرامج

الشكل (9) يبين خريطة سير البرامج لعملية حساب معدل N من العلامات.



الشكل (9) خريطة سير البرامج لعملية حساب معدل N من العلامات

2. خوارزميات اللغة الطبيعية (Structured English Algorithms)

تعتبر الخوارزميات جزءاً محدوداً من اللغة الإنجليزية وتحتوي على مصطلحات محدودة وجمل محددة التعابير تُيسر على المبرمج فهم وتتبع منطق البرنامج.

وللخوارزميات أشكال عدة، وعلى المحلل اختيار أحدها واعتباره معياراً موحداً لتوصيف البرامج في المشروع. وأهم هذه الأشكال عزيزي الدارس- ما يأتي:

أ- الشكل العام (Common Style) حيث تتم كتابة الخوارزمية بالحروف الصغيرة وتكون بداية كل كلمة بالحروف الكبيرة، كما هو موضح بالخوارزمية التالية:

Start
Read Grades
Compute Their Sum
Divide The Sum Over Their Number

Store The Result In Average
Print Average
Stop

ب- خوارزميات الحروف الكبيرة Code Style حيث تكتب الخوارزمية بالحروف الكبيرة، كما هو مبين في الخوارزمية التالية:

START
READ GRADES
COMPUTE THEIR SUM
DIVIDE THE SUM OVER THEIR NUMBER
STORE THE RESULT IN AVERAGE
PRINT AVERAGE
STOP

ج- الخوارزمية المرقمة (Outline Style) حيث يتم ترقيم خطوات الخوارزمية، كما هو موضح في الخوارزمية التالية:

1. Start
2. Read Grades
3. Compute Their Sum
4. Divide The Sum Over Their Number
5. Store The Result In Average
6. Print Average
7. Stop

د- الخوارزميات القصصية (Narrative Style)، حيث يتم في هذا الشكل كتابة الخوارزمية على شكل سرد لحكاية أو قصة. ويستخدم هذا الشكل في بعض الأحيان في حالة مناقشة الخوارزمية مع غير المتخصصين في البرمجة. والخوارزمية التالية تبين مثالاً لهذا النوع:

First, read the grades. Then, compute their summation. Next, divide the summation over their number, and store the result in average. At the end, print average.

3. أشباه البرامج Pseudo Code

يعتبر هذا النوع من أدق الأنواع وأقربها إلى لغات البرمجة الفعلية، وينصح باستخدامه بشكل كبير. وتحتوي الخوارزمية من هذا النوع على كل التراكيب البرمجية

مثـــل التكـــرار (Iteration)، وجمـــل القـــرار(Alteration)، والتتـــابع (Sequence). والخوارزمية التالية- عزيزي الدارس- تبين مثالاً على هذا النوع:

```
Read Grade1, Grade2, Grade3
Sum = Grade1+ Grade2+ Grade3
Average = SUM/3
Print average
Stop
```

أسئلة التقويم الذاتي (3)

ما هو الفرق بين الخوارزميات وأشباه البرامج؟

تدريب (1)

اكتب وصف خوارزمية جمع الأرقام من 1 إلى 100 باستخدام:

1. خرائط سير البرامج.
2. أشباه البرامج.
3. خوارزميات اللغة الطبيعية.

8. الخلاصة

لقد تعلمت- عزيزي الدارس- في هذه الوحدة بعض المواضـــيع المتقدمـــة فـــي التصميم المعتمد على الكينونة، حيث تعرفت على مجموعة مفـــاهيم ذات صـــلة بطـــرق التصميم ولغات البرمجة الكينونية مثل الترابط الديناميكي والوراثة. بالإضافة إلى ذلك تم بيان بعض المعايير التي يمكن استخدامها لزيادة جودة التصميم مثل التماسك، والتـــرابط، وتغليف المعلومات. وأخيراً قمت بالتعرف على مفهوم توصيف الطرق ورأيت كـــم هـــو مفيد للمبرمجين ويسهل عليهم برمجة وصيانة النظام.

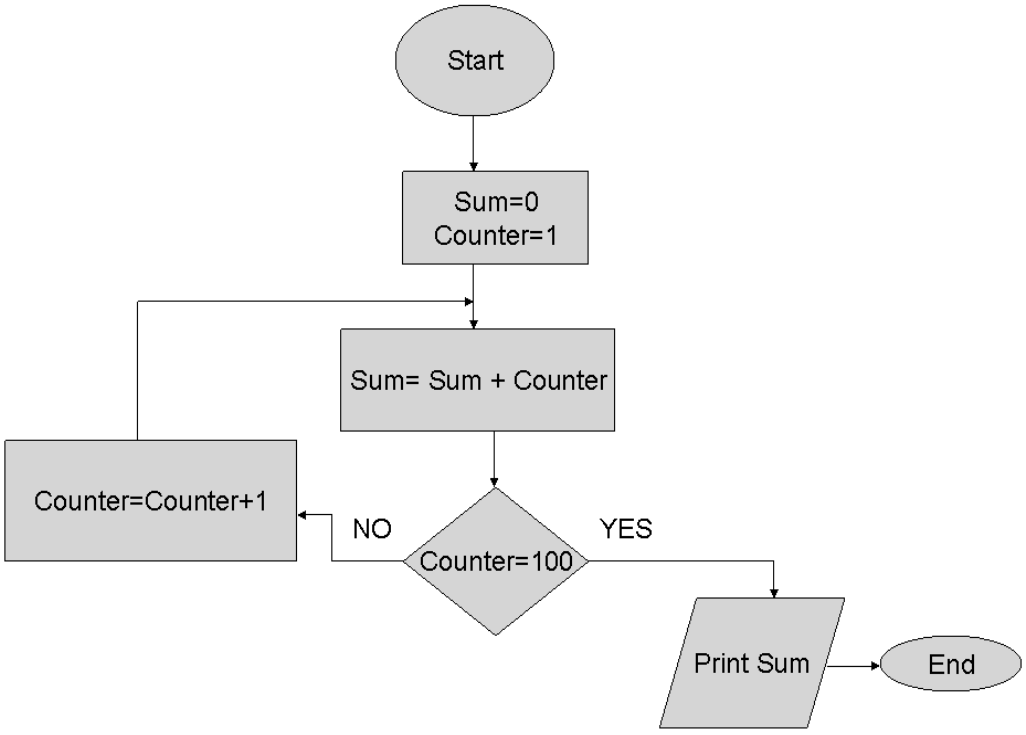
9. لمحة عن الوحدة الدراسية العاشرة

الهدف الأساسي من الوحدة العاشرة هو بيان كيفية تصميم قاعدة البيانات. تبدأ الوحدة ببيان أنواع الملفات وأهمية واستخدام كل منها، ثم تتطرق إلى مناقشة وسائط التخزين وبيان أنواعها وميزات كل نوع. والجزء الثالث من الوحدة يبين طريقة تصميم قاعدة البيانات بالتفصيل. أما الجزء الأخير من الوحدة فيبين أنواع الأنظمة من حيث طرق المعالجة.

10. إجابات التدريبات

تدريب (1)

1. خرائط سير البرامج



2. أشباه البرامج

Pseudo code

1. Start
2. Sum=0
3. Counter=1
4. Sum=Sum+Counter
5. If Counter not= 100 then Go to step 4
6. Print Sum
7. End

3. خوارزميات اللغة الطبيعية

Structured English

- Start the algorithm then initialize both Sum and Counter by 0,1 respectively
- Add the counter to sum
- Then keep looping the adding counter to sum till counter reaches 100
- Once you are done, print out the sum content

11. مسرد المصطلحات

- **إخفاء المعلومات Information Hiding:** عملية الاحتفاظ بالصفات والطرق التي يتطلب استخدامها في كينونةً معينة ولا يجوز للكينونات الأخرى الوصول إليها واستعمالها إلا إذا لزم الأمر لذلك، وفي هذه الحالة يكون الاستخدام باستدعاء هذه الطرق ولا ضرورة لتعريفها وكتابتها مرة أخرى في هذه الكينونات، بل يجب أن تكون أسماء هذه الطرق معروفة لهذه الكينونات حتى تستطيع استدعاءها.
- **الارتباط الديناميكي Dynamic Binding:** عبارة عن قدرة النظام المبني على الكينونة بالرجوع إلى البيانات ومعرفة نوعها في وقت التنفيذ.
- **الترابط Coupling:** يحدد معيار الترابط بين الأصناف ودرجة اعتماد بعضها على بعض، وكلما كان الترابط أقل كلما كان التصميم أفضل.
- **تعدد الأوجه Polymorphism:** يعني التواجد للكينونة الواحدة بعدة أشكال.
- **تغليف المعلومات Encapsulation:** التغليف عبارة عن آلية تدمج ما بين العمليات والبيانات في كينونة واحدة.
- **التماسك Cohesion:** عبارة عن درجة قوة الطريقة من الداخل، وتقاس بعدد الوظائف التي تحتويها الطريقة، فكلما كانت الطريقة تحتوي على وظائف أقل كلما كانت متماسكة أكثر.
- **التوريث Inheritance:** آلية تسمح لمطور النظام بتعريف الأصناف بشكل مرن بحيث نستطيع إعادة استخدام أصناف معرفة مسبقاً كأساسات للأصناف الحالية.
- **توصيف الطرق Methods Specification:** عبارة عن خوارزميات مكتوبة تحتوي على خطوات واضحة تصف المنطق الداخلي للطرق.

12. المراجع

1. Aktas, Z., Structured Analysis and Design of Information Systems, Prentice-Hall, 1987.
2. Baudoin, C. and Hollowell, G., Realizing The Object-Oriented Life Cycle, Prentice- Hall, 1996.
3. Booch, G., Rumbaugh, J., and Jacobson, I., The Unified Modeling Language User Guide, Addison- Wesley, 1999.
4. Dennis, A. et al, Systems Analysis & Design: An Object-Oriented Approach with UML. John Wiley & Sons Inc., 2002.
5. Dewitz, S., Systems Analysis and Design and the Transition to Objects, McGraw-Hill, 1996.
6. Edwards, P., Systems Analysis, Design, and Development with Structured Concepts, Rinehart and Winston, 1999.
7. Henderson, P., Object-Oriented Specification and Design with C++, McGraw- Hill, 1993.
8. Hoffer, J. et al., Modern Systems Analysis & Design, 2nd Ed., Addison-Wesley, 1999.
9. Kendal, K., and Kendal, J., Systems Analysis and Design, 5th Ed.2002.
10. Kruchten, P., The Rational Unified Process: An Introduction. 2nd Ed., Addison -Wesley, 2000.
11. Lejk, M. and Deeks. An Introduction to Systems Analysis Techniques, 2nd Ed., Pearson Education Limited, 2002.
12. Maciaszek, A., Requirements Analysis and System Design, Developing Information Systems with UML, Addison -Wesley, 2001.
13. Quatrani, T., Visual Modeling with Rational Rose 2000 and UML, Addison- Wesley, 2000.
14. Satzinger, J., Systems Analysis and Design, 2nd Ed., Thomson Learing, 2002.
15. Silver, G. and Silver, M., Systems Analysis and Design, Addison-Wesley, 1996.
16. http://www.psut.edu.jo / computer science/ staff/saleh abu-soud
17. www.rational.com
18. www.omg.com

الوحدة العاشرة

تصميم الملفات وقواعد البيانات
Files and DB Design

محتويات الوحدة

الموضوع	الصفحة
1. المقدمـــة	235
1.1 تمهيد	235
2.1 أهداف الوحدة	235
3.1 أقسام الوحدة	235
4.1 القراءات المساعدة	236
5.1 ما تحتاج إليه لدراسة الوحدة	236
2. أنواع الملفات Types of Files.....	237
3. وسائط وطرق التخزين Media and Storage Techniques.....	238
4. تصميم قاعدة البيانات Database Design	241
5. طرق التحكم والسيطرة على قواعد البيانات DB Control Systems	244
6. الخلاصة	245
7. لمحة عن الوحدة الدراسية الحادية عشرة	245
8. مسرد المصطلحات	246
9. المراجع	247

1. المقدمة

1.1 تمهيد

أهلاً بك- عزيزي الدارس- إلى الوحدة العاشرة من مقرر "1380 تحليل الأنظمة وتصميمها"، وهي بعنوان "تصميم الملفات وقواعد البيانات".

تعتبر قاعدة البيانات المخزن الذي يزود النظام بالبيانات المطلوبة وتخزن فيه البيانات اللازمة. وتحتوي قاعدة البيانات على مجموعة من الملفات التي تعتبر اللبنة الأساسية لها. وللملفات أنواع عدة، كل نوع يناسب تطبيقاً معيناً وله واسطة تخزين مناسبة.

ولتصميم قاعدة البيانات خطوات متسلسلة، كل منها يحتوي على عدة مهام. وعند تصميم قاعدة البيانات، لا بد من بناء نظام سيطرة وتحكم بها وذلك للحفاظ عليها من العابثين من الضرر.

وفي هذه الوحدة، سيتم- عزيزي الدارس- مناقشة كل هذه النقاط بالتفصيل.

2.1 أهداف الوحدة

يتوقع منك- عزيزي الدارس- بعد الانتهاء من دراسة هذه الوحدة أن تكون قادراً على أن:

1. تحدد أنواع الملفات.
2. تحدد وسائط وطرق التخزين.
3. تصمم الملفات.
4. تصمم قاعدة البيانات.
5. تتحكم وتسيطر على قواعد البيانات.

3.1 أقسام الوحدة

تقسم الوحدة إلى أربعة أقسام رئيسة: القسم الأول: يبين أنواع الملفات وأهمية تصميمها ويغطي الهدف الأول. أما القسم الثاني: فيبين وسائط وطرق التخزين فيها بالإضافة إلى طرق الوصول للبيانات داخل الملفات وهذا يغطي الهدف الثاني. وأما القسم الثالث: فيناقش طرق تصميم الملفات وقواعد البيانات ويغطي الهدفين الثالث والرابع. وأما

القسم الرابع والأخير: فيبين كيفية المحافظة على سلامة وسرية قاعدة البيانـــات، وهـــذا يغطي الهدف الخامس.

4.1 القراءات المساعدة

أنصحك- عزيزي الدارس- بالإطلاع على المراجع التاليـــة لمـــا تحويـــه مـــن معلومات مفيدة تتعلق بموضوع هذه الوحدة:

1. Fundamentals of Database Systems, 4th Ed., Elmasri et. al, Addison-Wesley, 2004
2. Database System Concepts. 4th Ed., Silberschatz et. Al, McGraw-Hill, 2002.
3. Edwards, P., Systems Analysis, Design, and Development with Structured Concepts, Rinehart and Winston, 1999.
4. Kendal, K., and Kendal, J., Systems Analysis and Design, 5th Ed.,.2002.
5. Silver, G. and Silver, M., Systems Analysis and Design, Addison-Wesley, 1996.
6. Aktas, Z., Structured Analysis and Design of Information Systems, Prentice-Hall, 1987.
7. http://www.psut.edu.jo / computer science/ staff/saleh abu-soud

5.1 ما تحتاج إليه لدراسة الوحدة

عزيزي الدارس: قبل أن تبدأ بدراسة هذه الوحدة تأكد من أنـــك هيـــأت المكـــان الهادئ المناسب للدراسة مما يساعدك على التركيز والتفكير العميق. وتأكد من وجود قلم وأوراق، ثم احرص على تتبع الإرشادات التي توجهك لدراسة الوحدة، وحـــاول بجديـــة الإجابة عن أسئلة التقويم الذاتي، لأنها تساعدك على مراجعة موضوعات الوحدة، وتعمق معلوماتك حولها.

2. أنواع الملفات Types of Files

يعتبر الملف الوحدة الأساسية للتخزين في قاعدة البيانات، حيث يحتوي على سجلات متجانسة من حيث التركيب والنوع وعدد الحقول. وللملفات أنواع عديدة وذلك حسب نوع محتوياتها من البيانات وطريقة ترتيب هذه البيانات فيها. وأشهر هذه الأنواع – عزيزي الدارس– وأكثرها استخداماً:

أ. الملف الرئيس (Master File):

يعرف الملف الرئيس أنه الملف الذي يحوي البيانات الرئيسة، شبه الدائمة قليلة التغيير، كبيانات الطالب في نظام التسجيل أو بيانات العميل في نظام البنك. وغالباً ما يكون هذا النوع مرتبطاً بمخططات الأصناف.

ب. ملف الحركات (Transaction File):

يعرف ملف الحركات أنه الملف الذي يتم تخزين جميع الحركات اليومية فيه والتي يقوم بها المستخدم أثناء تشغيل النظام، كعمليات السحب والإيداع على الحسابات أو حركات البيع والشراء اليومية في نظام تجاري. ونستطيع القول أن ملف الحركات مرتبط بالمخططات التفاعلية.

ج. الملف الاحتياطي (Backup File):

هو الملف الذي يحوي نسخة احتياطية للبيانات. ويستخدم هذا النوع في حالة حدوث تحطم للنظام حيث يستعمل لإعادة النظام لحالته السابقة التي كان عليها قبل حدوث التحطم.

د. الملف المؤقت (Temporary File):

هو الملف الذي يستخدم لتخزين البيانات بشكل مؤقت لاستخدامها لأغراض تشغيل واستعمال النظام بحيث يتم الاستغناء عن هذه الملفات عند الانتهاء من استعمالها. ويعتبر توفير استخدام الذاكرة الدائمة من أهم فوائد هذا النوع من الملفات.

ه. ملف البرنامج (Program File):

عبارة عن ملف نصي (Text File) يستخدم لتخزين أوامر البرنامج نفسه (Source Code) الذي يقوم بتنفيذ النظام.

و. ملف التحكم (Control File):

يعرف هذا النوع أنه الملف الذي يحتوي المعلومات التي يستخدمها النظام للتحكم بفعالياته.

أسئلة التقويم الذاتي (1)

اذكر أنواع الملفات واستخدامات كل منها.

3. وسائط وطرق التخزين
Media and Storage Techniques

يقصد هنا بوسائط التخزين الوسائل التي تستخدم لتخزين البيانات، وطرق التخزين هي كيفية ترتيب البيانات داخل الملف ليسهل استرجاعها. والواقع هناك نوعان من الوسائط هي: الشرائط (Tapes) ويتم تخزين البيانات عليها بشكل تتابعي (Sequential)، والأقراص الممغنطة (Magnetic Disks) وتخزن البيانات فيها بشكل عشوائي (Direct or Random).

وستتم– عزيزي الدارس– في الفقرات التالية مناقشة هذه الطرق وبيان خصائص كل منها:

أ. الشرائط والملفات التتابعية (Tapes and Sequential Files)

تعتبر الشرائط من أقدم وسائط التخزين التي حازت على شهرة كبيرة، حيث إنها تعتبر إحدى أنواع الذاكرة الدائمة المستخدمة لتخزين البيانات، وكانت تعد سابقاً من أكثر طرق التخزين شيوعاً لما كانت تتميز به من خصائص.

الملفات التتابعية هي الملفات التي يتم تخزين السجلات فيها بطريقة متسلسلة وتكون هذه السجلات مرتبة تصاعدياً أو تنازلياً حسب حقل معين غالباً ما يكون مفتاح (Key)، بحيث يتم الوصول إلى السجلات بشكل تتابعي حسب قيمة هذا المفتاح.

إن كون الشرائط تتابعية بطبيعتها، يجعل من البديهي أن تخزن الملفات التتابعية عليها، حيث ترتبط الملفات التتابعية والشرائط بشكل عضوي بحيث لا نستطيع الاعتماد على الشرائط بدون وجود الملفات التتابعية والعكس غير صحيح.

يعتبر ملف الحركات من أهم الأمثلة على الملفات التتابعية حيث يرتب ويخزن على شريط قبل تحديث الملف الرئيس.

وتعتبر الشرائط رخيصة الثمن وذات حجم تخزيني كبير لذلك تستعمل الشرائط عادة للتخزين الاحتياطي. ومن خصائص الشرائط المهمة أنها قابلة للنقل (Portable) من جهاز إلى آخر وبكل سهولة. وإن البطء يعتبر من أهم سيئات استخدام الشرائط حيث إن الوصول للمعلومة لا يتم بشكل مباشر بل تتابعي أي أننا يجب المرور على كل السجلات السابقة قبل أن نصل إلى السجل المطلوب. ومن سيئات الشرائط والملفات التتابعية أنها لا تدعم المعالجات متعددة الوظائف (Multitask Processing) لأننا لا نستطيع الحصول على المعلومة إلا بشكل تتابعي فقط.

ب. الأقراص الممغنطة والملفات ذات الوصول المباشر (Disks & Direct Files)

هي إحدى أشهر وأكثر الوسائل انتشاراً إلى يومنا الحالي حيث تعتمد هذه الوسيلة على الأقراص الممغنطة والمجال المغنطيسي إذ تكون المنطقة الممغنطة مكافئة للرمز (1) ويرمز للمنطقة غير الممغنطة بالرمز (0)، ويمتاز هذا الوسيط بعدم الحاجة للتخزين التتابعي والقدرة على الوصول المباشر للمعلومة حيث إن أجزاء الملفات تخزن أينما كان على سطح القرص بشرط ضمان الوصول للمعلومة.

ومن حسنات الأقراص الممغنطة أنها تعتبر سريعة وتدعم الملفات ذات الوصول المباشر، حيث يتم التحديث (Updating) على نفس القرص، وتدعم المعالجات متعددة الوظائف لكوننا نستطيع الحصول على المعلومة بشكل مباشر.

أسئلة التقويم الذاتي (2)

اذكر الفرق بين الشرائط والأقراص الممغنطة وبيّن ميزات كل منها.

يصنف العلماء الملفات إلى عدة أنواع حسب طريقة الوصول إلى البيانات المخزنة فيها كما هو مبين- عزيزي الدارس- في النقاط الآتية:

أ. الملفات التتابعية (Sequential Files)

الملف التتابعي هو ذلك النوع من الملفات التي يتم الوصول فيه إلى السجل المطلوب بشكل تتابعي حيث يتم البحث من بداية الملف كل مرة والمتابعة بالبحث حتى يتم الوصول للسجل المطلوب. وكما تم ذكره مسبقاً، يجب أن يكون الملف مرتباً ترتيباً تصاعدياً أو تنازلياً اعتماداً على مفتاح معين.

ورغم أن معالجة الملفات التتابعية تعتبر بطيئة مقارنة بالأنواع الأخرى، إلا أن هذا النوع يناسب ملفات الحركات التي يتم تحديث غالبية السجلات فيها، كملف الرواتب مثلاً حيث يتم حساب الراتب لكل الموظفين عدا بعض الموظفين المعارين، على سبيل المثال، أو المجازين إجازة طويلة.

ب. الملفات التتابعية المفهرسة (Indexed Sequential Files)

تختلف الملفات التتابعية المفهرسة عن الملفات التتابعية البحتة في نقطة رئيسة هي أن البحث يبدأ من سجل معين قريب من السجل المراد البحث عنه وليس بالضرورة من بداية الملف كما يحدث في الملف التتابعي، وهذه النقطة تُحدَّد عن طريق فهرس يتم بناؤه اعتماداً على حقل أو أكثر في الملف، والذي يكون الملف مرتباً بناء عليها.

والواقع أن البحث في هذا النوع من الملفات أسرع من النوع السابق، وذلك أن الفهرس؛ بالطبع إذا كان مبنياً بطريقة سليمة؛ يقلل بدرجة كبيرة من منطقة البحث.

ج. ملفات الوصول المباشر (Direct Files)

يتم الوصول للمعلومة في هذا النوع من الملفات بشكل مباشر دون الحاجة إلى المرور بأي من السجلات الأخرى داخل الملف. هذا النوع يعتبر الأسرع مقارنة ببقية الأنواع.

وتجدر الإشارة هنا إلى أمرين؛ أولهما: إنه يجب استخدام الأقراص الممغنطة إذا ما قرر المحلل أن يستخدم هذا النوع من الملفات، وثانيهما: إن هذا النوع من الملفات عادة ما يناسب الملفات الرئيسة.

أسئلة التقويم الذاتي (3)

بين الفروق بين الملفات التتابعية المفهرسة والملفات التتابعية.

4. تصميم قاعدة البيانات Database Design

عزيزي الدارس– ستقوم بدراسة عملية تصميم قواعد البيانات في مقرر قاعدة البيانات وإدارتها "1383"، ولكن ستقوم بدراسته في هذا المقرر من وجهة نظر التحليل؛ أي كخطوة من خطوات تحليل وتصميم الأنظمة، وسترى كيف يمكن تضمينه معلومات تفيد عملية بناء النظام.

تتكون عملية تصميم قاعدة البيانات من عدة خطوات رئيسة لحصرها في النقاط الآتية:

1. تصميم الملفات (DB Tables)

تعتبر الملفات اللبنة الأساسية لقاعدة البيانات، إذ تُعرف قاعدة البيانات بأنها مجموعة مترابطة من الملفات. ولتصميم الملفات خطوات يمكن تلخيصها في الآتي:

أ- استخرج قائمة الملفات Entity-Relationship Diagram

تستخرج قائمة الملفات من مخطط الكينونات– العلاقات (Entity-Relationship Diagram) وذلك حسب القواعد التي سوف تتعلمها ضمن مقرر قاعدة البيانات وإدارتها "1383".

ب- اختر واسطة التخزين وطريقة الوصول المناسبة لكل ملف

ج- ارسم جدول الملف وذلك بالإستعانة بقاموس البيانات (Data Dictionary) ومخطط الكينونات العلاقات، حيث يحتوي على المعلومات الآتية:

1- أسماء الحقول (Fields).

2- نوع الحقل (Format) حيث يستخدم الرمز 9 للدلالة على أن نوع الحقل رقمي، والرمز (A) للدلالة على أن نوع الحقل حرفي، والرمز (X) للحقول التي تحتوي النوعين معاً.

3- فارغ (Null) ولها قيمتان (YES) وتعني أن الحقل يمكن أن لا يحتوي على قيمة وNO وتعني أن الحقل لا يمكن أن يكون فارغاً.

4- مفتاح رئيس (Primary Key) وتكون قيمته (YES) إذا كان الحقل مفتاحاً رئيساً للملف.

5- فريد (Unique) تستخدم القيمة (YES) إذا كان من غير المسموح تكرار قيمة الحقل لسجل آخر. لاحظ أن المفتاح الرئيس دائماً يكون فريداً.

6- شرط الفحص (Check)، يستخدم هذا الفحص في حالة وجود شرط على الحقل.

7- القيمة الأولية (Default Value) تستخدم في حالة الرغبة في تحديد قيمة أولية للحقل يمكن تغييرها فيما بعد أو إبقائها كما هي.

8- المفتاح الأجنبي (Foreign Key) تكون قيمته (YES) في حالة كون الحقل مفتاحاً أجنبياً.

9- المرجع (Reference) تكون قيمته عبارة عن اسم الملف والحقل اللذين أتى منهما المفتاح الأجنبي.

10- جملة الربط (Join Condition) وهي عبارة عن الجملة التي تربط المفتاح الأجنبي والحقل الذي أتى منه.

11- المؤشر (Index) تكون قيمته (YES) إذا تم بناء مؤشر على هذا الحقل.

الشكل (1) يبين مثالاً على جدول لملف يحتوي على خمسة حقول، ومفتاحه الرئيس (Privilege_Assistant_Id, Privilege_Course_Number)

Field	Format	Null	P.K	Unique	Check	Default	F.K	Reference	Join condition	index
PRIVILEGE_ASSISTANT_ID	9(3)	NO	YES	YES	>0	-	-	-	-	YES
PRIVILEGE_COURSE_NUMBER	9(4)	NO	YES	YES	>0	-	-	-	-	YES
PRIVILEGE_SECTION_NUMBER	9(2)	YES	-	NO	-	-	YES	SECTIONS (SECTION_NUMBER)	PRIVILEGES. PRIVILEGE_SECTION_NUMBER = SECTIONS. SECTION_NUMBER	-
PRIVILEGE_SEMESTER	9(1)	NO	-	YES	>0	1	YES	SECTIONS (SECTION_SEMESTER)	PRIVILEGES. PRIVILEGE_SEMESTER = SECTIONS. SECTION_SEMESTER	YES
PRIVILEGE_YEAR	9(4)	NO	-	NO	>1990	2003	YES	SECTIONS (SECTION_NUMBER)	PRIVILEGES. PRIVILEGE_YEAR = SECTIONS. SECTION_NUMBER	-

الشكل (1) مثال على جدول ملف

ب- اكتب معلومات الملف

تجب كتابة معلومات عن كل ملف من ملفات قاعدة البيانات، وتفيد هذه المعلومات عدة جهات أهمها: المبرمجون، ومدير قاعدة البيانات (DB Administrator)، وفريق توصيف الأجهزة (Hardware Specification Team). وتحتوي معلومات الملف على العناصر التالية:

- نوع الملف (File Type).
- واسطة التخزين (Storage Media).
- النسخ الاحتياطية (Backups) تعني وقت عمل النسخ الاحتياطية وما إذا كانت دورية أم لا.
- حجم السجل (Record Size): عدد حروف السجل.
- الحجم الأولي للملف (File Initial Size): الحجم الأولي الذي يعطيه مدير قاعدة البيانات للملف.
- الحجم الأعلى للملف (File Maximum Size): أكبر حجم يمكن للملف أن يصل إليه، وهذه القيمة تقدر تقديراً من قبل المحلل، وبجمعها لكل الملفات تفيد معرفة أقصى حجم تخزين للذاكرة الثانوية يمكن أن نحتاجه.
- حجم الزيادة التراكمية (Increment Size): مقدار زيادة حجم الملف عند امتلائه.
- نظام الحماية (Access Security): من يحق لهم الوصول للبيانات في الملف وما يحق لهم عمله.
- معادلات التأكد (Record Counts).

الشكل (2) يبين– عزيزي الدارس– مثالاً على معلومات ملف الصلاحيات.

PRIVILEGES FILE INFORMATION
File Type: Random Access File
Storage Media: Hard disk
Backups: Weekly
Record Size: 14 Bytes
Initial Size: 500 records X 14 Bytes
= 7000 Bytes = 7KB
Estimated Maximum Size:
100,000 records X 14 Bytes
= 1,400,000 Bytes = 1.4 MB
Increment Size: 100 records
Access Security: Read only for users of Rank 1
Read/Write for Rank 2 users
Record Counts: SUM(Privilege_section_number)

الشكل (2) مثال على معلومات ملف

أسئلة التقويم الذاتي (4)

تتكون عملية تصميم قاعدة البيانات من عدة خطوات رئيسة، اذكر هذه الخطوات وبين محتويات كل خطوة.

5. طرق التحكم والسيطرة على قواعد البيانات
DB Control Systems

يقصد بطرق التحكم والسيطرة على قواعد البيانات بأنها إجراءات يحددها المحلل ويتم اتباعها لحماية قاعدة البيانات من عدة أمور أهمها- عزيزي الدارس- ما يأتي:

- الضرر أو الدمار (Damage) الذي قد يلحق بقاعدة البيانات سواءً أكان مقصوداً أم لا.
- الوصول غير الشرعي (Illegal Access) للبيانات المخزنة في قاعدة البيانات، بمعنى منع أشخاص أو كينونات غير مخولة بالدخول والاطلاع على المعلومات المحمية من عمل ذلك.

ولتحقيق الأهداف المذكورة أعلاه، ينصح باتباع الإرشادات وطرق التحكم والسيطرة التالية:

- الحرص على عمل نسخ احتياطية (Backup) لقاعدة البيانات وبشكل دوري، وذلك للاستعانة بها لإرجاع قاعدة البيانات (Recovery) إلى حالتها الصحيحة عند حدوث ضرر أو دمار فيها.
- عمل نظام حماية يحتوي على كلمات دخول سرية (Passwords) للسماح أو لمنع الدخول لنظام قاعدة البيانات بناءً على رغبتنا، ويحتوي كذلك على صلاحيات (Privileges) لتحديد ما يستطيع عمله من سُمح لهم بالدخول لقاعدة البيانات.
- عمل معادلات تأكد (Record Counts) على البيانات المهمة، بمعنى تنفيذ عمليات معينة على بعض البيانات الحساسة الموجودة في قاعدة البيانات قبيل الانتهاء من العمل على النظام وإغلاقه، وتطبيق نفس العمليات على نفس البيانات عند البدء في استخدام النظام في المرة القادمة. فإذا حصل اختلاف في نتائج هذه العمليات في المرتين، نتأكد عندها من أنه تم التلاعب بالبيانات.
- بناء نظام مراقبة وتدقيق، حيث يمكن أن يكون على شكل سجل (Log File) تسجل فيه كل حركات الدخول لنظام قواعد البيانات بحيث تتضمن اسم المستخدم، ووقت الدخول، والحركات التي قام بها، وأسماء الملفات التي دخل إليها، ووقت الخروج وغيرها من المعلومات التي يراها المحلل ضروريه. ويساعد هذا السجل على معرفة الذين دخلوا أو حاولوا الدخول إلى قاعدة البيانات، وتصحيح ما قد يكون قد بدر منهم من تخريب أو تغيير غير مشروع للبيانات.

أسئلة التقويم الذاتي (5)

ما المقصود بطرق التحكم والسيطرة على قواعد البيانات؟ وما هو الهدف منهـا؟ وما هي الإرشادات التي ينصح باتباعها في هذا السياق؟

6. الخلاصة

لقد تعلمت–عزيزي الدارس– في هذه الوحدة كيف أن تصميم قاعـدة البيانـات يُعتبر من خطوات بناء النظام المهمة، حيث إنها تعتبر المخزن الذي يـستخدمه النظـام لتخزين واسترجاع البيانات التي يحتاجها. إن قاعدة البيانات تتكون من ملفات ذات أنواع مختلفة ولها وسائط تخزين متعددة أيضاً.

ولتصميم قاعدة البيانات خطوات أهمها:

- استخراج قائمة الملفات.
- اختيار واسطة التخزين المناسبة.
- رسم جدول الملف.
- كتابة معلومات الملف.

وعند بناء قاعدة البيانات يجب بناء نظام حماية لهـا لحمايتهـا مـن العبـث أو الوصول غير الشرعي من قبل المستخدمين وغيرهم.

- الحرص على عمل نسخ احتياطية.
- عمل نظام حماية يحتوي على كلمات دخول سرية.
- عمل معادلات تأكد.
- بناء نظام مراقبة وتدقيق.

7. لمحة عن الوحدة الدراسية الحادية عشرة

تناقش الوحدة الحادية عشرة موضوعين رئيسين؛ الموضوع الأول: يبين ماهيـة المعايير (Standards) وأهمية استخدامها في البرمجة، كما يبين المنهجيات المتبعـة فـي برمجة النظام وخاصة منهجية أعلى/أسفل (Top/Down approach) ومنهجية أسفل/أعلى (Bottom/Up approach)؛ ويناقش أهميتها ويقارن بينها. أما الموضوع الثاني من هـذه الوحدة. فيبين أهمية اختبار النظام وخطواته وطريقة تطبيقه وتأثيره على النظام.

8. مسرد المصطلحات

- **الملف الاحتياطي Backup File:** هو الملف الذي يحوي نسخة احتياطية للبيانات. ويستخدم هذا النوع في حالة حدوث تحطم للنظام حيث يستعمل لإعادة النظام لحالته السابقة التي كان عليها قبل حدوث التحطم.
- **ملف البرنامج Program File:** عبارة عن ملف نصي (Text File) يستخدم لتخزين أوامر البرنامج نفسه (Source Code) الذي يقوم بتنفيذ النظام.
- **الملفات التتابعية Sequential Files:** الملف التتابعي هو ذلك النوع من الملفات الذي يتم الوصول فيه إلى السجل المطلوب بشكل تتابعي حيث يتم البحث من بداية الملف كل مرة والمتابعة بالبحث حتى يتم الوصول للسجل المطلوب. ويجب أن يكون الملف مرتباً ترتيباً تصاعدياً أو تنازلياً اعتماداً على مفتاح معين.
- **الملفات التتابعية المفهرسة Indexed Sequential Files:** تختلف الملفات التتابعية المفهرسة عن الملفات التتابعية البحتة في نقطة رئيسية هي أن البحث يبدأ من سجل معين قريب من السجل المراد البحث عنه وليس بالضرورة من بداية الملف كما يحدث في الملف التتابعي، وهذه النقطة تُحدَّد عن طريق فهرس يتم بناؤه اعتماداً على حقل أو أكثر في الملف، والذي يكون الملف مرتباً بناءً عليها.
- **ملف التحكم Control File:** يعرف هذا النوع أنه الملف الذي يحتوي المعلومات التي يستخدمها النظام للتحكم بفعالياته.
- **ملف الحركات Transaction File:** الملف الذي يتم تخزين جميع الحركات اليومية فيه التي يقوم بها المستخدم أثناء تشغيل النظام، كعمليات السحب والإيداع على الحسابات أو حركات البيع والشراء اليومية في نظام تجاري. ونستطيع القول إن ملف الحركات مرتبط بالمخططات التفاعلية.
- **الملف الرئيس Master File:** الملف الذي يحوي البيانات الرئيسة، شبه الدائمة قليلة التغيير وغالباً ما يكون هذا النوع مرتبطاً بمخططات الأصناف.
- **الملف المؤقت Temporary File:** هو الملف الذي يستخدم لتخزين البيانات بشكل مؤقت لاستخدامها لأغراض تشغيل واستعمال النظام بحيث يتم الاستغناء عن هذه الملفات عند الانتهاء من استعمالها. ويعتبر توفير استخدام الذاكرة الدائمة من أهم فوائد هذا النوع من الملفات.
- **ملفات الوصول المباشر Direct Files:** يتم الوصول للمعلومة في هذا النوع من الملفات بشكل مباشر دون الحاجة إلى المرور بأي من السجلات الأخرى داخل الملف. وهذا النوع يعتبر الأسرع مقارنة ببقية الأنواع.

9. المراجع

1. Aktas, Z., Structured Analysis and Design of Information Systems, Prentice-Hall, 1987.
2. Baudoin, C. and Hollowell, G., Realizing The Object-Oriented Life Cycle, Prentice- Hall, 1996.
3. Booch, G., Rumbaugh, J., and Jacobson, I., The Unified Modeling Language User Guide, Addison- Wesley, 1999.
4. Dennis, A. et al, Systems Analysis & Design: An Object-Oriented Approach with UML. John Wiley & Sons Inc., 2002.
5. Dewitz, S., Systems Analysis and Design and the Transition to Objects, McGraw-Hill, 1996.
6. Edwards, P., Systems Analysis, Design, and Development with Structured Concepts, Rinehart and Winston, 1999.
7. Henderson, P., Object-Oriented Specification and Design with C++, McGraw- Hill, 1993.
8. Hoffer, J. et al., Modern Systems Analysis & Design, 2nd Ed., Addison-Wesley, 1999.
9. Kendal, K., and Kendal, J., Systems Analysis and Design, 5th Ed., 2002.
10. Kruchten, P., The Rational Unified Process: An Introduction. 2nd Ed., Addison- Wesley, 2000.
11. Lejk, M. and Deeks. An Introduction to Systems Analysis Techniques, 2nd Ed., Pearson Education Limited, 2002.
12. Maciaszek, A., Requirements Analysis and System Design, Developing Information Systems with UML, Addison- Wesley, 2001.
13. Quatrani, T., Visual Modeling with Rational Rose 2000 and UML, Addison- Wesley, 2000.
14. Satzinger, J., Systems Analysis and Design, 2nd Ed., Thomson Learing, 2002.
15. Silver, G. and Silver, M., Systems Analysis and Design, Addison-Wesley, 1996.
16. http://www.psut.edu.jo / computer science/ staff/saleh abu-soud
17. www.rational.com
18. www.omg.com

الوحدة الحادية عشرة

برمجة النظام واختباره

Programming and Testing the System

محتويات الوحدة

الموضوع	الصفحة
1. المقدمـــة	253
1.1 تمهيد	253
2.1 أهداف الوحدة	253
3.1 أقسام الوحدة	253
4.1 القراءات المساعدة	254
5.1 ما تحتاج إليه لدراسة الوحدة	254
2. برمجة النظام Programming the System	255
1.2 المعايير المستخدمة في البرمجة Programming Standards	255
2.2 طرق برمجة النظام Methods of Programming the System	256
3.2 تتبع البرامج Programs Walkthroughs	259
3. اختبار النظام System Testing	260
1.3 أهمية الاختبار Importance of Testing	260
2.3 مراحل الاختبار Phases of Testing	261
4. الخلاصة	263
5. لمحة عن الوحدة الدراسية الثانية عشرة	263
6. مسرد المصطلحات	264
7. المراجع	265

1. المقدمة

1.1 تمهيد

أهلاً بك- عزيزي الدارس- إلى الوحدة الحادية عشرة من مقرر" 1380 تحليل الأنظمة وتصميمها". وهي بعنوان " برمجة النظام واختباره".

يعتبر التنسيق بين المحللين والمبرمجين من الأمور المهمة التي يجب على الفريق تطبيقه، ويعتبر استخدام المعايير (Standards) من الوسائل التي تدعم هذا التنسيق وتشجع عليه مما يؤدي إلى سهولة البرمجة وسهولة توثيقها. وعند برمجة أجزاء النظام من قبل المبرمجين يستحسن أن تتم البرمجة بترتيب معين وليس بطريقة عشوائية، حيث إن هناك منهجيتين عالميتين لهذا الغرض هما منهجية أعلى/أسفل (Top/Down approach) ومنهجية أسفل/أعلى (Bottom/Up approach).

عند الانتهاء من برمجة النظام وقبل تسليمه للمستخدمين يجب التأكد من خلوه من الأخطاء ومن أنه يؤدي الغرض الذي من أجله تم بناؤه، وهذا ما يسمى باختبار النظام. ولاختبار النظام خطوات ومراحل عديدة.

2.1 أهداف الوحدة

يتوقع منك-عزيزي الدارس- بعد الانتهاء من دراسة هذه الوحدة أن تكون قادراً على أن:

1) تستخدم معايير في البرمجة.
2) تعدد أنواع المنهجيات المتبعة في برمجة النظام.
3) تبين أهمية اختبار النظام.
4) تحدد مراحل اختبار النظام.

3.1 أقسام الوحدة

تقسم هذه الوحدة إلى قسمين رئيسين. القسم الأول: يبين ماهية المعايير (Standards) وأهمية استخدامها في البرمجة، كما يبين هذا القسم المنهجيات المتبعة في برمجة النظام وخاصة منهجية أعلى/أسفل (Top/Down approach) ومنهجية أسفل/أعلى (Bottom/Up Approach)؛ ويناقش أهميتها ويقارن بينها. وهذا يغطي الهدفين الأول

والثاني. وأما القسم الثاني من هذه الوحدة: فيبين أهمية اختبار النظام وخطواته وطريقة تطبيقه وتأثيره على النظام وهذا يغطي الهدفين الثالث والرابع.

4.1 القراءات المساعدة

عزيزي الدارس: حاول أن تتطلع على المراجع التالية لاتصالها المباشر والوثيق بموضوع هذه الوحدة، وهذا سيساعدك على فهمها واستيعابها بشكل عميق، وهذه المراجع هي:

1. Edwards, P., Systems Analysis, Design, and Development With Structured Concepts, Rinehart and Winston, 1999.
2. Hoffer, J. et al., Modern Systems Analysis & Design, 2nd, Ed., Addison-Wesley, 1999.
3. Kendal, K., and Kendal, J., Systems Analysis and Design, 5th Ed., 2002.
4. Lejk, M. and Deeks. An Introduction to Systems Analysis Techniques, 2nd Ed., Pearson Education Limited, 2002.
5. Satzinger, J., Systems Analysis and Design, 2nd Ed., Thomson Learing, 2002.
6. Silver, G. and Silver, M., Systems Analysis and Design, Addison-Wesley, 1996.

5.1 ما تحتاج إليه لدراسة الوحدة

عزيزي الدارس، قبل أن تبدأ بدراسة هذه الوحدة تأكد من أنك هيأت المكان المناسب للدراسة مما يساعدك على التركيز والتفكير العميق. وفي أثناء الدراسة حاول الإجابة عن أسئلة التقويم الذاتي والاطلاع على بعض المعايير العالمية المتبعة في البرمجة.

2. برمجة النظام Programming the System

تقع مسؤولية البرمجة، في الواقع، على مرحلة التصميم في الأساس حيث يقوم المصممون بكتابة خوارزميات البرامج وتجهيزها وترتيبها وتسليمها للمبرمجين. وكما تم ذكره- عزيزي الدارس- في وحدات سابقة من هذا المقرر؛ ويتكون النظام من أجزاء (Modules) عديدة ويقوم عادةً فريق من المبرمجين ببرمجته. فإذا لم يتم الاتفاق على معايير بين المصممين والمبرمجين لإتباعها قبل وأثناء وبعد برمجة النظام، ستكون عملية البرمجة بحد ذاتها صعبة وتحتوي على الكثير من التناقضات مما يؤدي إلى صعوبة عملية اختبار النظام وصيانته.

كما أن عدم اتباع طريقة لترتيب برمجة الأجزاء وتوزيعها على المبرمجين سيؤدي إلى صعوبة بالغة في البرمجة وتأخير في إنجاز العمل.

والبنود التالية تبين ماهية وأهمية استخدام المعايير في البرمجة، كما تبين طرق ترتيب أجزاء البرمجة.

1.2 المعايير المستخدمة في البرمجة
Programming Standards

تُعرّف المعايير بأنها مجموعة من القواعد التي يجب على المصممين والمبرمجين اتباعها عند تصميم البرامج وبرمجتها. وهذه القواعد تطبق في معظم مراكز المعلومات وشركات البرمجة المعتبرة. وتختلف المعايير من شركة إلى أخرى ومن نظام إلى آخر في الشركة الواحدة.

ومن الفوائد العظيمة لاستخدام المعايير كونها تعزز استخدام أسلوب متناغم للبرمجة يوحد الاختلافات بين المصممين والمبرمجين ويسهل عملهم ويجعل من انخراط أعضاء جدد للفريق أمراً سهلاً. كما يؤدي اتباع المعايير إلى سهولة صيانة البرامج واختبارها.

أسئلة التقويم الذاتي (1)

عدّد فوائد استخدام معايير البرمجة.

ويمكن تحديد معايير لأسماء الملفات، والحقول، والأجزاء، والمتغيرات في البرنامج، وشكل البرنامج...

الجزء التالي يبين مثالاً لمعايير يمكن استخدامها عند تصميم وبرمجة نظام ما:

1. أسماء الملفات: يكون اسم الملف اسماً كاملاً ذا معنى مكتوباً بالحروف الصغيرة ولكن أول حرف يكون من الحروف الكبيرة ويكون بصيغة الجمع، مثلاً (Students) وحسب هذه القاعدة يمثل اسم ملف يتعلق بالطلاب.
2. أسماء الحقول: يكون اسم الحقل اسماً كاملاً ذا معنى مكتوباً بالحروف الصغيرة ولكن أول حرف يكون من الحروف الكبيرة ويكون مسبوقاً بالحالة المفردة لاسم الملف الذي يتبعه، مثلاً (StudentName) يعني حقلاً اسمه (StudentName) ويتبع للملف (Students).
3. أسماء المتغيرات المحلية (Local Variables): يكون اسماً كاملاً ذا معنى مكتوباً بالحروف الصغيرة ويكون مسبوقاً بالحرف L وهو أول حرف من الكلمة (Local). مثال: (Lsum).
4. أسماء المتغيرات غير المحلية: (Global Variables) يكون اسماً كاملاً ذا معنى مكتوباً بالحروف الصغيرة ويكون مسبوقاً بالحرف G وهو أول حرف من الكلمة (Global). مثال: (Gmax).
5. يجب أن تكون كل جملة من جمل البرنامج على سطر مستقل.

وتجدر الإشارة هنا – عزيزي الدارس– إلى أنه يمكن كتابة معايير لأي منحى من مناحي البرمجة، يتم الاتفاق عليها من قبل مركز الحاسوب.

نشاط (1)

ابحث عن طريق الإنترنت على معايير عالمية للبرمجة لشركات رائدة في البرمجة مثل (Microsoft) و (Oracle).

2.2 طرق برمجة النظام

Methods of Programming the System

تبدأ البرمجة الفعلية بمراجعة شاملة لوثيقة مواصفات البرامج (Programs Specifications) وهي الوثيقة التي تم بناؤها في نهاية مرحلة التصميم التي تظهر أجزاء النظام (System Modules) وطريقة ارتباط بعضها ببعض بالإضافة إلى المنطق الداخلي لكل جزء والذي يكون موصوفاً إما على شكل خوارزمية (Algorithm) أو خارطة تدفق البيانات (Program Flowchart).

وعادة ما يكون النظام مكوناً من عشرات الأجزاء، وكل جزء يؤدي هدفاً معيناً، وتكون هذه الأجزاء مرتبطة بعضها مع بعض بحيث تشكل في النهاية شجرة يكون جذرها البرنامج الرئيس (Main Program) الذي يستدعي بقية الأجزاء بينما تكون الأجزاء الموجودة على أطراف الشجرة هي الأجزاء الصغيرة التي لا تستدعي أجزاءً أخرى.

والآن هناك خياران لفريق البرمجة، فإما أن يقوموا ببرمجة الأجزاء بشكل عشوائي أو أن يقوموا بذلك حسب ترتيب معين. ويمكن أن يؤدي الخيار الأول إلى صعوبة السيطرة على عملية البرمجة مما يؤدي في النهاية إلى فشل عملية البرمجة برمتها أو إلى صعوبتها وتعقيدها على أقل تقدير. أما إذا تمت البرمجة بطريقة مرتبة وحسب استراتيجية محددة، عندها سنوفر على أنفسنا الكثير من المشاكل التي نحن بغنى عنها.

في الواقع، توجد منهجيتان تستخدمان لهذا الغرض على مستوى واسع، هما منهجية أعلى/أسفل (Top/Down Approach) ومنهجية أسفل/أعلى (Bottom/Up Approach).

أولاً: منهجية أعلى/أسفل (Top/Down approach)

هنا يتم البدء ببرمجة البرنامج الرئيس وعند الانتهاء منه يتم الانتقال إلى المستوى الذي يليه وهكذا ... حتى نصل إلى آخر مستوى. والواقع إن هذه الطريقة منظمة وتبدأ من العام إلى الخاص، ولكن تواجه من يتبع هذه الطريقة مشكلة تكمن في الآتي: عندما ينتهي المبرمج من برنامج في مستوى معين، فإنه لا يستطيع أن يختبره إلا إذا قام ببرمجة البرامج التي يستدعيها هذا البرنامج في المستوى الذي يليه. وذلك لأن هذا البرنامج يحتوي على جمل استدعاء لبرامج المستوى الذي يليه، وفي كثير من الأحيان تبقى هذه المشكلة حتى ينتهي المبرمجون من برمجة كامل النظام ومن ثم يستطيعون اختباره، وهذا أمر غير جيد، إذ إن اختبار النظام جزءاً جزءاً أسهل بكثير من اختباره كوحدة واحدة.

إذن ما هو الحل؟؟ في الواقع، الحل يكمن في استعمال ما يسمى بجمل الأصل (Stubs) حيث إن هذه الجمل عبارة عن جمل تعليق مؤقتة (Comment Statements) تستخدم للدلالة على موقع جمل استدعاء البرامج الفرعية التي لم تتم برمجتها بعد. أي أن جمل الأصل تكون في مستوى والبرامج الفرعية في مستوى أدنى. إن استعمال جمل الأصل يفيد المبرمج بإعطائه الفرصة لاختبار برنامجه حتى ولم ينته بعد من برمجة البرامج التي يستدعيها برنامجه. مثلاً إذا كان البرنامج يحتوي على جملة الاستدعاء التالية:

```
.
.
CALL SUM(X,Y,Z)
.
```

ويمكن تضمين البرنامج لجملة أصل كالآتي:

```
.
.
/* CALL SUM(X,Y,Z)
WRITE ('Procedure SUM is not ready yet')
.
```

ولاستخدام جمل الأصل فوائد كثيرة، من أهمها:

- تساعد على الاختبار المبكر للبرنامج.
- تعطي المجال للمبرمج لكتابة البرامج الفرعية بالترتيب الذي يراه مناسباً.
- تقوي عملية بناء البرنامج.
- تجبر المبرمجين على استخدام طريقة أعلى/أسفل.

أسئلة التقويم الذاتي (2)

عدّد فوائد استخدام جمل الأصل في طريقة أعلى/أسفل.

ثانياً: منهجية أسفل/أعلى (Bottom/Up Approach)

يبدأ المبرمج في هذه الطريقة ببرمجة البرامج الفرعية الموجودة في أدنى مستوى أولاً ثم يبدأ ببرمجة البرامج الموجودة في المستوى الأعلى وهكذا حتى يصل إلى البرنامج الرئيس. وهنا أيضاً توجد مشكلة تكمن في عدم وجود جملة استدعاء للبرنامج الفرعي الذي انتهى من برمجته، أي أن المبرمج لا يستطيع اختبار برنامجه إلا إذا قام ببرمجة البرامج الموجودة في المستوى الأعلى، حيث يمكن لهذه المشكلة أن تبقى إلى حين الانتهاء من برمجة كامل النظام دون إعطاء الفرصة للمبرمج لاختبار برامجه أولاً بأول.

كما هو الحال في طريقة أعلى/أسفل، ويوجد أيضاً في هذه الطريقة حل لهذه المشكلة يكمن باستخدام ما يسمى بجمل الاختبار (Test Drivers)، وهذه الجمل عبارة عن جمل مؤقتة تستخدم لاستدعاء البرنامج الفرعي الذي انتهى المبرمج من كتابته مما يعطيه المجال لاستدعائه واختباره قبل أن يكتب البرنامج الذي يقوم باستدعائه فعلياً.

أسئلة التقويم الذاتي (3)

بين الفوائد التي يجنيها المبرمج من استخدام جمل الاختبار في طريقة أسفل/أعلى.

لاحظ – عزيزي الدارس– بأن المبرمج سواءً استخدم طريقة أسفل/أعلى أم طريقة أعلى/ أسفل فإنه في النهاية سينتهي من بناء النظام. إذن ومن حيث المبدأ، يستطيع المبرمج استخدام أي من هاتين الطريقتين حيث إنها جاءت لتنظيم عملية البرمجة والسيطرة عليها.

3.2 تتبع البرامج Programs Walkthroughs

عند الانتهاء من كتابة البرامج يقوم فريق من المتخصصين في البرمجة بتتبع سريع للبرامج قبل تسليمها لفريق الاختبار بهدف اكتشاف الأخطاء البرمجية وخاصة الأخطاء التركيبية والإملائية (Syntax errors) وكذلك الأخطاء المنطقية (Semantic errors) التي يمكن أن توجد في البرامج الصحيحة تركيبياً كالقسمة على صفر. كما يفيد تتبع البرامج في اكتشاف النواقص إن وجدت كأن ينسى المبرمج إظهار التاريخ على الفاتورة أو الفصل الدراسي على كشف علامات الطالب. ويهدف التتبع أيضاً إلى اكتشاف الاستخدام الخاطئ للّغة.

ينصح هنا بعدم إشراك مستخدمي النظام في عملية التتبع هذه، وذلك لأنهم عادة لا يمتلكون المعرفة الكافية للإفادة في هذه العملية بل يمكن أن يكونوا مصدر إزعاج للفريق في كثير من الأحيان.

وعادة ما يتم تحديد عضو من الفريق لتسجيل الأخطاء المكتشفة وترتيبها وصياغتها بطريقة ملائمة ومن ثم تسليمها للمبرمجين الذين قاموا ببرمجة هذه البرامج، حيث إنه ليس من مهام فريق التتبع تصحيح الأخطاء المكتشفة بل لا ينبغي له القيام بذلك. وإذا كانت الأخطاء رئيسة فيمكن عندها القيام بعملية تتبع أخرى.

وبعد الانتهاء من عملية التتبع السريع للبرامج وتصحيح الأخطاء والنواقص المكتشفة، تكون البرامج جاهزة للاختبار الرسمي من قبل فريق متخصص. والقسم التالي من هذه الوحدة يبين – عزيزي الدارس– أهمية الاختبار ومراحله بشكل مفصل.

أسئلة التقويم الذاتي (4)

بين أهمية القيام بالتتبع السريع للبرامج قبل عملية الفحص الحقيقي للنظام.

3. اختبار النظام System Testing

بعد تتبع البرامج وإزالة الأخطاء البسيطة والسهو والاستخدام الخاطئ للّغة، تكون البرامج في هذه المرحلة جاهزة للفحص الفعلي باستخدام بيانات فعلية أو افتراضية. وأثناء الاختبار يحاول فريق الفحص إزالة الأخطاء التي من الممكن أن تعرقل سير النظام عند تطبيق النظام بشكل فعلي.

1.3 أهمية الاختبار Importance of Testing

إن فحص النظام واكتشاف الأخطاء الموجودة فيه عملية مهمة جداً ويستنفذ أكثر من 30% من الوقت والميزانية المخصصة للنظام، لذلك يجب إعطاء هذه العملية الاهتمام والوقت والجهد الكافي لإنجاحها وتحقيق الأهداف المرجوة منها والتي تتمثل في الهدفين الرئيسين الآتيين:

- التحقق من أن البرامج تؤدي الوظائف المتوقعة لها بالشكل الصحيح.
- التحقق من أن النظام يؤدي وظائفه وفقاً لما هو مخطط له في مرحلتي التحليل والتصميم.

أسئلة التقويم الذاتي (5)

ما هي أهداف اختبار النظام؟

يمكن فحص النظام- عزيزي الدارس- باستخدام بيانات يمكن الحصول عليها من عدة مصادر أهمها:

1. **بيانات حقيقية من النظام السابق:**

عادة ما يكون حجم هذه البيانات كبيراً جداً وتحتوي على كل الحالات والاحتمالات للأوضاع التي من الممكن للنظام أن يواجهها عند التطبيق الفعلي. لذلك فهي توفر بيئة حقيقية لاختبار النظام. وتستخدم هذه الطريقة إذا كانت هيكلية البيانات في النظام القديم قريبة نوعاً ما من بيانات النظام الجديد وإلا سنحتاج إلى برامج تحويل ومن الممكن أن تتعقد العملية ولا تصبح مجدية.

2. **بيانات يتم بناؤها يدوياً:**

تستخدم هذه الطريقة من قبل الطلاب والمحللين المبتدئين لسهولة بناء البيانات فيها. وفي هذه الحالة، يقوم المحلل ببناء ملف للبيانات يحاول فيه تضمين البيانات لكل

الاحتمالات التي يتوقع أن يواجهها النظام في المستقبل. ولكن يجب أن نلاحظ بأنه من الصعب بل من المستحيل في كثير من الأحيان أن نستطيع بناء بيانات يدوياً بهذه المواصفات، وخاصة أن حجم البيانات التي يتم بناؤها بهذه الطريقة يكون قليلاً. لذلك فإن هذه الطريقة لا تناسب فحص النظام تحت الظروف الحقيقية حيث حجم العمل كبير والبيانات ضخمة.

3. **بيانات يتم بناؤها عن طريق برامج خاصة:**

تمتاز البيانات الناتجة عن هذه الطريقة بأنها ذات حجم كبير وتناسب فحص النظام تحت الظروف الطبيعية ولكنها قد لا تحتوي على كل الاحتمالات حيث يعتمد ذلك على قوة البرنامج في إنتاج بيانات بهذه المواصفات.

2.3 مراحل الاختبار Phases of Testing

يتم عادة إجراء اختبار النظام على عدة مراحل يمكن تلخيصها – عزيزي الدارس– كالآتي:

1. **فحص الأجزاء (Module Testing)**

يركز هذا الفحص على التأكد من صحة المنطق الداخلي لكل جزء على حدة دون الأخذ بعين الاعتبار علاقته ببقية الأجزاء. وهنا يحاول الفاحص دفع الجزء إلى الفشل بإعطائه البيانات التي تحتوي كل الاحتمالات،وذلك يتم بفحص كل المسارات المنطقية في الجزء فإذا فشل الجزء باعطاء النتيجة الصحيحة يكون فيه خلل لابد من تصحيحه، وعكس ذلك يكون الجزء صحيحاً.

2. **فحص ترابط الأجزاء (Modules Integration)**

عندما يتم التأكد من أن الأجزاء صحيحة وتؤدي الغرض الذي أُنشئت من أجله، يتم فحص ترابط الأجزاء بعضها ببعض، أي عندما يستدعي الجزء جزءاً آخر يجب التأكد من أن البيانات المراد إرسالها للجزء الآخر قد وصلت بشكل سليم وهذا الجزء عندما يقوم بتنفيذ مهمته يجب التأكد من استلام الجزء المستدعي للنتائج بشكل صحيح.

3. **فحص البرامج (Programs Testing)**

هنا يتم فحص كل برنامج (الذي يتكون من مجموعة من الأجزاء) على حدة والتأكد من أنه يؤدي الهدف منه وذلك باستخدام بيانات الفحص التي تم إعدادها مسبقاً.

4. **فحص ترابط البرامج (Programs Integration)**

بعد فحص كل برنامج والتأكد من صحته، يتم التأكد من أن البرامج تعمل بعضها مع بعض بالشكل الصحيح، ويتم ذلك على مرحلتين:

أ- فحص التسلسل (String Test)

حيث يتم التأكد من أن مخرجات برنامج هي مدخلات صحيحة لبرنامج آخر، كأن يتم التأكد من أن عدد الساعات الدراسية التي سجلها الطالب عن طريق برنامج تسجيل المواد الدراسية يتسلمها برنامج المحاسبة بالشكل الصحيح ليتم احتساب الرسوم لهذا الطالب.

ب- فحص الربط (Link Test)

حيث يتم التأكد من أن مجموعة البرامج المرتبطة بعضها ببعض تعمل بالشكل الصحيح. كأن يتم التأكد من أن نظام التسجيل بأقسامه المكونة من نظام الإرشاد ونظام تسجيل المواد ونظام احتساب الرسوم الجامعية يعمل كوحدة واحدة بشكل متكامل وصحيح.

أسئلة التقويم الذاتي (6)

ما الفرق بين فحص التسلسل وفحص الربط؟

5. فحص النظام (System Testing)

تعتبر هذه هي المرحلة الأخيرة من الفحص حيث يتم فيها التأكد من أن النظام ككل يعمل كما هو متوقع منه. ويتم في هذه الحالة فحص النظام على أرض الواقع وفي بيئة العمل الحقيقية ولمدة طويلة نسبياً قد تمتد إلى أسابيع أو أشهر، حيث تعتمد مدة الفحص على عدة عوامل أهمها: درجة تعقيد النظام والدورة المحاسبية للتنظيم الذي يطبق النظام، حيث إن النظام يجب أن يتم فحصه لدورة محاسبية واحدة على الأقل ليتم التأكد من أن كل وظائف النظام تعمل بالشكل الصحيح. ومن الممكن أن يتم إشراك المستخدمين أنفسهم في هذا الفحص. ويتم التركيز على فحص التقارير التي تنتج من النظام.

6. فحص القبول (Acceptance Testing)

يتم التأكد هنا من أن النظام قد حقق أهدافه التي تم الاتفاق عليها في دراسة الجدوى، ويتم هذا الفحص من قبل المستخدمين وإدارتهم. وإذا اجتاز النظام فهذا الفحص، هذا يعني أن النظام جاهز للاستخدام الفعلي.

أسئلة التقويم الذاتي (7)

عدد خطوات اختبار النظام وبين الهدف من كل خطوة.

4. الخلاصة

عزيزي الدارس، إن لاستخدام معايير للبرمجة أثراً بالغ الأهمية في التنظيم من حيث سهولة البرمجة وتعديل البرامج وصيانتها. كذلك تحسن هذه المعايير من درجة تخاطب أعضاء الفريق وتفاهم بعضهم مع بعضهم الآخر.

وكما رأينا في هذه الوحدة فإن إتباع طرق منظمة لترتيب برمجة أجزاء النظام يؤدي إلى السيطرة بشكل جيد على عملية البرمجة ويؤدي إلى تقوية بنية البرامج. وتعتبر طريقتا أسفل/أعلى (Bottom/Up approach) وأعلى/أسفل (Top/Down approach) من الطرق الشهيرة في هذا السياق. ولكل منهما نقاط ضعف ونقاط قوة ويمكن للمبرمج أن يتبع أي واحدة منهما.

وكما رأينا- عزيزي الدارس- في هذه الوحدة أيضاً أن اختبار النظام من العمليات المهمة في إنتاج أنظمة صحيحة، خالية من الأخطاء، وتحقق الأهداف التي تم بناؤها لأجلها. ولاحظنا أن اختبار النظام يتم على عدة خطوات ومراحل، وذلك باستخدام بيانات من عدة مصادر أهمها النظام القديم، ومن الممكن بناؤها يدوياً أو باستخدام برامج لهذا الغرض. وفي جميع الأحوال كلما كانت البيانات المستخدمة في الاختبار كبيرة وتحتوي على كل الاحتمالات الممكنة كلما كانت أفضل وذات تأثير أقوى على نتائج اختبار النظام.

5. لمحة عن الوحدة الدراسية الثانية عشرة

تناقش الوحدة القادمة موضوع التحول، حيث يقصد بالتحول عملية الانتقال من بيئة النظام القديم إلى بيئة النظام الجديد وتهيئة المستخدمين والوضع بشكل عام ليكون مناسباً للعمل بالنظام الجديد. وهذا يعني أن أشياء كثيرة سيتم تغييرها كالأجهزة والتجهيزات العامة وبعض البرمجيات، وغيرها. وسنتعرف في هذه الوحدة على أهمية التحول وطرقه بالإضافة إلى الأشياء التي يجب تغييرها.

والجزء الثاني من هذه الوحدة: يناقش عملية توثيق النظام، حيث تعتبر هذه العملية من العمليات المهمة جداً لما تقدمه من خدمات للمستخدمين ومهندسي النظام على حد سواء. كما يقدم التوثيق حماية دائمة للنظام على المدى البعيد. لكننا سنرى بأن بناء التوثيق الجيد ليس بالعملية السهلة حيث يتطلب وقتاً ومهارةً ومالاً. سيتم خلال هذه الوحدة، بيان أهمية وأنواع التوثيق بشكل مفصل. كما سيتم التطرق إلى بيان خصائص التوثيق الجيد وفائدته للنظام.

6. مسرد المصطلحات

- **تتبع البرامج Programs Walkthroughs:** عند الانتهاء من كتابة البرامج يقوم فريق من المتخصصين في البرمجة بتتبع سريع للبرامج قبل تسليمها لفريق الاختبار بهدف اكتشاف الأخطاء البرمجية والنواقص والاستخدام الخاطئ للّغة.
- **فحص التسلسل String Test:** حيث يتم التأكد من أن مخرجات برنامج هي مدخلات صحيحة لبرنامج آخر.
- **فحص الربط Link Test:** حيث يتم التأكد من أن مجموعة البرامج المرتبط بعضها ببعض تعمل بالشكل الصحيح.
- **المعايير المستخدمة في البرمجة Programming Standards:** تُعرّف المعايير بأنها مجموعة من القواعد التي يجب على المصممين والمبرمجين اتباعها عند تصميم البرامج وبرمجتها.
- **منهجية أسفل/أعلى Bottom/Up approach:** طريقة لترتيب برمجة أجزاء النظام حيث يبدأ المبرمج فيها ببرمجة البرامج الفرعية الموجودة في أدنى مستوى أولاً ثم يبدأ ببرمجة البرامج الموجودة في المستوى الأعلى وهكذا حتى يصل إلى البرنامج الرئيس.
- **منهجية أعلى/أسفل Top/Down approach:** طريقة لترتيب برمجة أجزاء النظام حيث يتم البدء فيها ببرمجة البرنامج الرئيس وعند الانتهاء منه يتم الانتقال إلى المستوى الذي يليه وهكذا... حتى نصل إلى آخر مستوى.
- **وثيقة مواصفات البرامج Programs Specifications:** هي الوثيقة التي تم بناؤها في نهاية مرحلة التصميم والتي تظهر أجزاء النظام (System Modules) وطريقة ارتباط بعضها ببعض بالإضافة إلى المنطق الداخلي لكل جزء الذي يكون موصوفاً إما على شكل خوارزمية (Algorithm) أو خارطة تدفق البيانات (Program Flowchart).

7. المراجع

1. Aktas, Z., Structured Analysis and Design of Information Systems, Prentice-Hall, 1987.
2. Baudoin, C. and Hollowell, G., Realizing The Object-Oriented Life Cycle, Prentice- Hall, 1996.
3. Booch, G., Rumbaugh, J., and Jacobson, I., The Unified Modeling Language User Guide, Addison- Wesley, 1999.
4. Dennis, A. et al, Systems Analysis & Design: An Object-Oriented Approach with UML. John Wiley & Sons Inc., 2002.
5. Dewitz, S., Systems Analysis and Design and the Transition to Objects, McGraw-Hill, 1996.
6. Edwards, P., Systems Analysis, Design, And Development With Structured Concepts, Rinehart And Winston, 1999.
7. Henderson, P., Object-Oriented Specification and Design with C++, McGraw- Hill, 1993.
8. Hoffer, J. et al., Modern Systems Analysis & Design, 2nd Ed., Addison-Wesley, 1999.
9. Kendal, K., and Kendal, J., Systems Analysis and Design, 5th Ed., 2002.
10. Kruchten, P., The Rational Unified Process: An Introduction. 2nd Ed., Addison- Wesley, 2000.
11. Lejk, M. and Deeks. "An Introduction to Systems Analysis Techniques", 2nd Ed., Pearson Education Limited, 2002.
12. Maciaszek, A., Requirements Analysis and System Design, Developing Information Systems with UML, Addison- Wesley, 2001.
13. Quatrani, T., Visual Modeling with Rational Rose 2000 and UML, Addison- Wesley, 2000.
14. Satzinger, J., Systems Analysis and Design, 2nd Ed., Thomson Learing, 2002.
15. Silver, G. and Silver, M., Systems Analysis and Design, Addison-Wesley, 1996.
16. http://www.psut.edu.jo / computer science/ staff/saleh abu-soud
17. www.rational.com
18. www.omg.com

الوحدة الثانية عشرة

تحول النظام وتوثيقه

System Conversion and Documentation

محتويات الوحدة

الموضوع	الصفحة
1. المقدمـــة	271
1.1 تمهيد	271
2.1 أهداف الوحدة	271
3.1 أقسام الوحدة	272
4.1 القراءات المساعدة	272
5.1 ما تحتاج إليه لدراسة الوحدة	272
2. تحول النظام System Conversion	273
1.2 أهمية التحول Importance of Conversion	273
2.2 المتغيرات والثوابت بين النظام القديم والجديد	273
3.2 طرق التحول Conversion Methods	275
3. توثيق النظام System Documentation	277
1.3 أهمية التوثيق Importance of Documentation	277
2.3 فوائد التوثيق الجيد Benefits of Good Documentation	277
3.3 مشاكل التوثيق Problems of Documentation	278
3.3 أنواع الوثائق Types of Documents	279
4. الخلاصة	282
5. مسرد المصطلحات	282
6. المراجع	284

1. المقدمة

1.1 تمهيد

أهلاً بك- عزيزي الدارس- إلى الوحدة الثانية عشرة والأخيرة من مقرر "1380 تحليل الأنظمة وتصميمها". وهي بعنوان "تحول النظام وتوثيقه".

المقصود بالتحول هنا هو عملية الانتقال من بيئة النظام القديم إلى بيئة النظام الجديد وتهيئة المستخدمين والوضع بشكل عام ليكون مناسباً للعمل على النظام الجديد. وهذا يعني أن أشياء كثيرة سيتم تغييرها كالأجهزة والتجهيزات العامة وبعض البرمجيات، وغيرها. وسنتعرف في هذه الوحدة على أهمية التحول وطرقه بالإضافة إلى الأشياء التي يجب تغييرها.

والجزء الثاني من هذه الوحدة يناقش عملية توثيق النظام، حيث تعتبر هذه العملية من العمليات المهمة جداً لما تقدمه من خدمات للمستخدمين ومهندسي النظام على حد سواء. كما يقدم التوثيق حماية دائمة للنظام على المدى البعيد. لكن بناء التوثيق الجيد ليست بالعملية السهلة حيث تتطلب وقتاً ومهارةً ومالاً. وسيتم خلال هذه الوحدة، بيان أهمية وأنواع التوثيق بشكل مفصل. كما سيتم التطرق إلى بيان خصائص التوثيق الجيد وفائدته للنظام.

2.1 أهداف الوحدة

يتوقع منك- عزيزي الدارس- بعد الانتهاء من دراسة هذه الوحدة أن تكون قادراً على أن:

1. تدرك أهمية التحول.
2. تحدد طرق التحول.
3. تذكر أهمية التوثيق.
4. تحدد أنواع التوثيق.
5. تفرق بين التوثيق الجيد والتوثيق السيئ.
6. تتمكن من توثيق نظام.

3.1 أقسام الوحدة

تقسم هذه الوحدة إلى قسمين رئيسين:

القسم الأول: يبين أهمية التحول وطرقه وهذا يغطي الهدفين الأول والثاني. أما القسم الثاني: فيناقش توثيق النظام ويبين أهميته ويحدد أنواعه ويبين خصائص التوثيق الجيد، وهذا يغطي الأهداف من الثالث حتى السادس.

4.1 القراءات المساعدة

أنصحك- عزيزي الدارس- بالإطلاع على المراجع التالية لما تحويه من معلومات مفيدة تتعلق بموضوع هذه الوحدة:

1. Edwards, P., Systems Analysis, Design, and Development with Structured Concepts, Rinehart and Winston, 1999.
2. Kendal, K., and Kendal, J., Systems Analysis and Design, 5th Ed.2002.
3. Silver, G. and Silver, M., Systems Analysis and Design, Addison-Wesley, 1996.
4. Aktas, Z., Structured Analysis and Design of Information Systems, Prentice-Hall, 1987.
5. http://www.psut.edu.jo / computer science/ staff/saleh abu-soud

5.1 ما تحتاج إليه لدراسة الوحدة

عزيزي الدارس: قبل أن تبدأ بدراسة هذه الوحدة تأكد من أنك قمت بتهيئة المكان الهادئ المناسب للدراسة مما يساعدك على التركيز والتفكير العميق. ثم احرص على تتبع الإرشادات التي توجهك لدراسة الوحدة، وحاول بجدية الإجابة عن أسئلة التقويم الذاتي، لأنها تساعدك على مراجعة موضوعات الوحدة، وتعمق معلوماتك حولها.

2. تحول النظام System Conversion

المقصود بالتحول–عزيزي الدارس– هو عملية تهيئة الوضع في المؤسسة صاحبة النظام الجديد للعمل على هذا النظام الجديد من حيث تجهيز المكان والبيانات القديمة والعدد اللازم من المستخدمين والتجهيزات والأجهزة وغيرها مما يلزم لتشغيل النظام. ويبدأ العمل بالتحول قبيل الانتهاء من بناء النظام.

1.2 أهمية التحول Importance of Conversion

يساعد التحول على سهولة الانتقال من الوضع الحالي إلى الوضع الجديد بسهولة ويسر. وقد يظن بعضنا أننا إذا انتهينا من بناء النظام قبل تسليمه لمستخدميه نكون قد تجاوزنا مرحلة الخطر ولم يبقَ إلا خطوة سهلة وغير مهمة نكون واهمين، حيث إن أي فشل في التحول قد يؤدي إلى فشل النظام الجديد برمته. لذلك يجب التخطيط جيداً قبل إجراء عملية التحول.

أسئلة التقويم الذاتي (1)

بين أهمية التحول والخطر الكامن في عدم النجاح في إنجازه بالشكل المطلوب.

2.2 المتغيرات والثوابت بين النظام القديم والجديد

يجب أن نستذكر– عزيزي الدارس– بأن عملية بناء نظام جديد لتنظيم ما، لا تقتصر على بناء البرمجيات فقط وإنما قد يلجأ محلل الأنظمة وحسب ما تقتضيه الحاجة؛ إلى إجراء تغييرات في الأجهزة والتجهيزات، والإجراءات المتبعة في التنظيم، والقواعد والقوانين المعمول بها في التنظيم، كما يمكن للمحلل إجراء تغييرات على المستخدمين من حيث عددهم وتنظيمهم وطريقة عملهم على النظام الجديد.

ولنكون أكثر تفصيلاً، فإن الأشياء التي قد يشملها التغيير عند التحول للنظام الجديد يمكن بيانها في النقاط التالية:

1) أجهزة الحاسوب القديمة:

يتم عادة تغيير الأجهزة القديمة إلى أجهزة جديدة إذا لم تعد مناسبة للعمل عليها في النظام الجديد وإذا لم تعد صالحة للتحسين أو تعديل قدراتها، حيث يتم استخراج البرامج والبيانات اللازمة المخزنة فيها وتنزيلها على الأجهزة الجديدة بعد إجراء

التغييرات اللازمة عليها إذا كانت غير متوافقة مع هذه الأجهزة، أما إذا كانت متوافقة فيتم إنزالها مباشرة كما هي.

ويمكن التصرف بالأجهزة القديمة بأية طريقة تراها المؤسسة مناسبة، حيث يمكن بيعها لشركات تتعامل مع الأجهزة المستعملة. ولكن عادة لا تكون أسعار الأجهزة القديمة مغرية، مما يشجع أصحاب المؤسسة إلى توزيعها على موظفيهم أو التبرع بها.

2) التجهيزات والخدمات Facilities and Services

يقصد بالتجهيزات والخدمات هنا، الأشياء المساعدة اللازمة للنظام الجديد مثل الكهرباء والإضاءة إذ من الممكن أن تكون متطلبات الأجهزة الجديدة من الكهرباء أكثر من الإمكانيات الحالية كأن تكون الكهرباء الحالية (2-Phase) مثلاً ويلزم للوضع الجديد (3-Phase). بالنسبة للإضاءة يمكن أن يتم زيادتها أو إنقاصها أو تغيير مكانها حسب متطلبات النظام الجديد.

ومن الممكن إضافة وحدات تكييف أو إذا كان النظام مطبقاً في مصنع يمكن إضافة أجهزة تحكم بالرطوبة والغبار. والأثاث المكتبي من الأشياء التي يتم تغييرها في أحيان كثيرة.

3) البيانات القديمة والملفات الرئيسة

عند بناء نظام جديد لتنظيم ما، إما أن يكون النظام القديم يدوياً أو نظاماً حاسوبياً لا يفي بالغرض. وفي جميع الأحوال هناك مدخلات (Inputs) وعمليات معالجة (Processing) تنفذ ونتائج (Outputs) يتم الحصول عليها من النظام وبيانات يتم تخزينها في قاعدة بيانات (Data Base) للرجوع إليها على المدى البعيد. عند وبعد بدء العمل على النظام الجديد، يجب أن يتم الاستمرار باستخدام البيانات القديمة، فإذا كانت هذه البيانات متطابقة مع قاعدة بيانات النظام الجديد ولم يحصل تغيير على بيئة البيانات القديمة فليس من الضروري تحويلها بل يمكن استخدامها كما هي، أما إذا كانت غير متطابقة معها فيجب تحويلها حتى يتسنى استخدامها. وتحويل قاعدة البيانات يتم إما بإضافة حقول جديدة أو بالتخلص من الحقول الزائدة. من الممكن بناء برامج لتحويل البيانات خاصة إذا كانت قاعدة البيانات كبيرة الحجم.

أسئلة التقييم الذاتي (2)

بين الأسباب والظروف التي تدعو إلى تحول قاعدة البيانات أو عدم تحولها.

4) طريقة العمل والإجراءات Working Procedures

لقد تم بناء النظام الجديد بناءً على عملية تحليل شاملة أخذت بعين الاعتبار تحسين طريقة العمل في التنظيم وربما تعديل بعض الإجراءات والقوانين التي تؤدي إلى فعالية أكبر في معالجة البيانات وتنفيذ الأعمال بطريقة أفضل.

وعند تغيير طريقة العمل أو الإجراءات، يجب على المحلل كتابة هذه الإجراءات وتدريب المستخدمين على طريقة العمل الجديدة.

5) النظام القديم

تغيير النظام القديم إلى النظام الجديد عملية محورية في التحول من الوضع الحالي إلى الوضع الجديد، والتي بسببها أساساً قامت عملية التحليل. وتعتبر هذه الخطوة من الخطوات المهمة جداً والخطيرة في نفس الوقت، حيث يؤدي الفشل فيها إلى فشل النظام الجديد برمته. إذن لا بد من تطبيق هذا الأمر بعد تفكير عميق وتخطيط شامل بحيث تتم عملية التحول بسلاسة وبدون مشاكل. وهناك عدة طرق للتحول سيتم بيانها بالتفصيل – عزيزي الدارس– في الفقرة الآتية.

3.2 طرق التحول Conversion Methods

يقصد بطرق التحول هنا التوقف عن استخدام النظام القديم واستبداله بالنظام الجديد. وهنالك ثلاث طرق رئيسة لفعل ذلك:

1) التحول المباشر (Direct Conversion)

هنا يتم التحول بشكل فوري ولكل النظام ككتلة واحدة، حيث تتم إزالة النظام القديم ويستبدل بالنظام الجديد. وتمتاز هذه الطريقة بقلة التكلفة وبقصر الفترة الزمنية اللازمة للتحول.

وتعتبر هذه الطريقة من أخطر طرق التحول لأنه في حالة فشل النظام الجديد يكون من الصعب العودة للنظام القديم. ولكن يمكن اتباع هذه الطريقة إذا كان قد تم فحص النظام الجديد جيداً وتم كذلك تدريب المستخدمين عليه.

2) التحول الموازي (Parallel Conversion)

في هذه الحالة يعمل النظامان بشكل متزامن حيث يتم إدخال البيانات إلى النظامين وكذلك يتم الحصول على المخرجات من النظامين. ويستبدل النظام القديم بالنظام

الجديد عند تشابه مخرجات النظامين مما يعني أن النظام الجديد يعطي نتـائج صـحيحة كونها تتشابه مع نتائج النظام القديم التي هي أصلاً صحيحة. رغم أن هذه الطريقة تقـدم خدمة مجانية بمقارنة النظامين بعضهما ببعض مما يعطي المجال للتأكد من النظام الجديد، إلا أن هذه المقارنة يمكن أن تكون خطرة حيث تعطي المجال للمستخدمين للتفكير بالعودة للنظام القديم الذي كانوا قد اعتادوا عليه مقابل النظام الجديد الذين لم يتعودوا عليه بعـد. وزيادة على ذلك، فإن هذه الطريقة قد تسبب إرباكاً للمستخدمين كونهم يستخدمون نظامين في آن واحد.

ولكن هذه الطريقة تعتبر من أقل طرق التحول خطورة لأنه في حالة فشل النظام الجديد يكون من السهل العودة للنظام القديم كونه ما زال يعمل ولم يتم استبداله بعد.

3) التحول التدريجي المرحلي (Phased Parallel Conversion)

تعتبر هذه الطريقة حلاً وسطاً بين الطريقتين السابقتين حيث يتم تحـول النظـام بأجزائه كل جزء (Module) على حدة، حيث يتم اختيار الجزء المـراد تحولـه للعمـل الفعلي عليه من قبل المستخدمين مع النظام القديم، ويتم سحب الجزء المقابل لهذا الجـزء في النظام القديم من الخدمة بعد التأكد من أن الجزء الجديد يعمل بشكل صـحيح، ويـتم الاستمرار باستبدال الأجزاء حتى يتم استبدال كل النظام.

وتعتبر هذه الطريقة أقل تكلفة من التحول المتوازي ولكنها تسبب إرباكـاً أكثـر للمستخدمين من الإرباك الذي تسببه بقية الطرق كونهم يستخدمون أجزاء من النظام القديم فقط وأجزاء من النظام الجديد فقط وأجزاء من النظامين في نفس الوقت.

أسئلة التقييم الذاتي (3)

ما هي طرق تحول النظام وما الفرق بينها؟

3. توثيق النظام System Documentation

1.3 أهمية التوثيق Importance of Documentation

يعرف التوثيق – عزيزي الدارس– بأنه عملية بناء مواد مكتوبة تستخدم لوصف النظام، كما يمكن استخدامها لدعم عملية تدريب المستخدمين على النظام. ويحتوي التوثيق على العديد من الوثائق (Documents) كتبت لتلبي حاجات جهات معينة بعينها، وهذا يعني أن لكل نوع من مستخدمي النظام نوعاً مختلفاً من أنواع الوثائق.

وتبنى الوثائق بطريقة تجميعية ومستمرة حيث تبدأ عملية التوثيق من مرحلة التحليل، وليس كما يظن بعضنا من أن التوثيق يبدأ بعد الانتهاء من النظام، علاوة على ذلك يجب أن يكون التوثيق جيداً حتى يستطيع الجميع استخدامه بشكل فعال ولكن كتابة التوثيق الجيد ليست بالعملية السهلة حيث يحتاج التوثيق الجيد إلى مهارة، ووقت، وميزانية.

2.3 فوائد التوثيق الجيد

Benefits of Good Documentation

للتوثيق الجيد أهمية خاصة وفوائد كبيرة للنظام، يمكن تلخيصها– عزيزي الدارس– بالنقاط الآتية:

1. يعتبر التوثيق الجيد صلة وصل لكل الجهات التي لها علاقة بالنظام:

وعند ظهور مشكلة في النظام لأحد المستخدمين، فإنه يكون بحاجة ماسة لمن يساعده على حل هذه المشكلة، فإذا كان التوثيق جيداً فإن هذا المستخدم سيستخدم هذا التوثيق لحل مشكلته. وكذلك الأمر بالنسبة للمبرمجين، الذين سيستخدمون التوثيق أثناء البرمجة لفهم نقطة في التصميم. ومصممو النظام يحتاجون للتوثيق لفهم التحليل، وهكذا نرى بأن التوثيق الجيد يصل إلى كل من لهم علاقة بالنظام.

2. يوفر حماية للنظام

يعتبر محللو الأنظمة والمبرمجون وغيرهم من المختصين ببناء الأنظمة من أكثر المختصين تنقلاً بين الموظفين. فلذلك عند حدوث أي خلل في النظام فإن الشيء الوحيد الذي لا يتغير تقريباً هو التوثيق، فهو إذن الضمانة الرئيسة لحماية النظام.

3. يوفر تكاليف على المدى البعيد

التوثيق الجيد وعند استخدامه يوفر بديلاً فعالاً عن المختصين الذين قاموا بتحليل وتصميم وبناء النظام. فبدلاً من استشارة هؤلاء المختصين وعند الحاجة لذلك، يستخدم

التوثيق لهذا الغرض مما يؤدي إلى توفير تكاليف نحن بغنى عنها. ويستخدم التوثيق الجيد أيضاً في تدريب المستخدمين بدلاً من بناء توثيق خاص للتدريب.

4. يسهل عملية صيانة النظام

تنفذ صيانة النظام عند وجود خلل في النظام أو عند إجراء تحسينات عليه، وفي الحالتين تجب إعادة دراسة النظام أو جزء منه من تحليل، أو تصميم، أو برمجة. والتوثيق الجيد يساعد على فهم النظام بكل مراحله مما يسهل عملية صيانته.

5. يعتبر بمثابة مرجع دائم للنظام

بما أن التوثيق عبارة عن مواد مكتوبة ودائما متوفرة للاستخدام، ويمكن الرجوع إليها عند الحاجة، فهي تعتبر مرجعاً دائماً للنظام.

أسئلة التقويم الذاتي (4)

بين أثر التوثيق الجيد على النظام.

3.3 مشاكل التوثيق Problems of Documentation

هناك عدة عيوب إن وجدت في التوثيق ستجعله سيئاً وغير قابل للاستخدام بشكل فعال. ويمكن استعراض هذه العيوب– عزيزي الدارس– في النقاط الآتية:

1. عدم اكتمال الوثائق

يقصد هنا بعدم اكتمال الوثائق هو كتابة وثائق ناقصة إما أنها لا تغطي كل أجزاء النظام كأن يتم عدم توثيق مرحلة من مراحل بناء النظام كمرحلة التحليل مثلاً أو مرحلة التصميم، وإما أنها كاملة بالعدد وتغطي كل جوانب النظام إلا أن بعض الوثائق ناقصة من الداخل، كأن لا يتم وصف شاشة من شاشات النظام في دليل المستخدم (User Manual).

2. كتابة الوثائق بشكل سيّئ

يمكن أن تكون الوثائق مكتوبة بشكل سيّئ من حيث عدم وضوح الكتابة نفسها كاختيار نوع حرف سيّئ أو وثيقة لا تحتوي على قائمة بالمحتويات أو لا تحتوي على أرقام صفحات أو غيرها من الأشياء التي تؤثر على شكل الوثيقة. وتكون الوثيقة سيئة كذلك إذا كانت طريقة الكتابة تتسم بعدم الوضوح كالكتابة بلغة صعبة تفوق مستوى فهم مستخدم الوثيقة.

3. الوثائق العتيقة (Out of Date)

المقصود بالوثائق العتيقة هي الوثائق التي لا تصف آخر نسخة من النظام بل

تصف نسخاً أو طبعات قديمة من النظام.

4. الوثائق المليئة بالمصطلحات الفنية المتخصصة ومفردات لغوية صعبة

يكون مستخدمو النظام عادة من غير المتخصصين في الحاسوب ومصطلحاته، لذلك وحتى تكون الوثائق مفهومة من قبل مستخدميها يجب أن تكون خالية من المصطلحات التي لا يفهمها إلا المتخصصون وكذلك يجب أن تكون اللغة المستخدمة سهلة وبسيطة حتى يتم استخدام الوثائق بفعالية.

5. نفور المحللين من التوثيق

إن عملية التوثيق تحتاج إلى مهارة، كما تم ذكره سابقاً. والرغبة في الكتابة تعتبر إحدى مقومات المهارة، حيث إن شخصاً لا يرغب بالكتابة من الصعب أن ينتج وثائق ناجحةً. والواقع أن الدراسات تشير إلى أن المحللين يكرهون كتابة الوثائق وينفرون منها، وفي حال إجبارهم على الكتابة سينتجون وثائق لا تتصف بالجودة. وتجدر الإشارة هنا- عزيزي الدارس- إلى أنّه لا بد من البحث عن أناس متخصصين يرغبون في الكتابة وعدم إجبار من لا يرغب في الكتابة ولا يجيدها على ذلك.

4.3 أنواع الوثائق Types of Documents

هنالك العديد من مستخدمي الوثائق، ويمكن تصنيفهم كالآتي:

1. مستخدمو النظام (Users): حيث يحتاجون إلى وثيقة تبين كيفية العمل على النظام.
2. موظفو الإدارة: ويحتاجون إلى وصف للتقارير الرئيسة التي يخرجها النظام والتي تتعلق بهم.
3. موظفو التشغيل (Operators): وهم المسؤولون عن تشغيل النظام وحمايته.
4. المحللون والمختصون ببناء النظام: وهم الذين قاموا بعملية تحليل وتصميم وبرمجة النظام، وهم الذين ينتجون الوثائق التي تصف ما تم تنفيذه في هذه المراحل.

وللوثائق أنواع عديدة تصنف حسب طبيعة مستخدميها. وهذه الأنواع وخصائصها ومكوناتها سيتم بيانها- عزيزي الدارس- كما يلي:

1. التوثيق الخاص بالإدارة (Management Documentation)

يستخدم هذا التوثيق من قبل الإدارة، حيث يحتوي على وصف للتقارير الإدارية الرئيسة وكيفية استخراجها والاستفادة من المعلومات التي تحويها. ويحتاج هذا النوع من التوثيق إلى أقل التفاصيل كون المستخدم مديراً.

ويحتوي هذا التوثيق على الأقسام التالية:

أ- مقدمة عن النظام .

ب- أهداف النظام وغاياته.

ج- أمثلة عن تقارير الإدارة الرئيسة.

د- الميزانية التقديرية والفعلية.

2. التوثيق الخاص بالمستخدمين (User Documentation)

يستخدم هذا التوثيق من قبل مستخدمي النظام، ويسمى هذا التوثيق في كثير من الأحيان بدليل المستخدم (User Manual). يجب أن يحتوي على كل المعلومات اللازمة للمستخدم بحيث يستطيع استخدام النظام بكل سهولة ويسر، لذلك يحتاج هذا النوع من التوثيق إلى تفاصيل أكثر من التوثيق الخاص بالإدارة إذْ يحتاج المستخدم إلى وصف كل حركة في النظام. ويفضل إشراك المستخدمين في عملية بناء هذه الوثيقة عن طريق إعطائهم الفرصة لقراءة الطبعات الأولى ومراجعتها وتجربتها ووضع الملاحظات عليها.

وفي الواقع يحتاج المستخدم لهذا الدليل عند حدوث خلل في النظام أو في أحسن حال عندما يحتاج إلى مساعدة في استخدام النظام. لذلك يجب أن يكون دليل المستخدم مكتوباً ومرتباً بشكل واضح وخالياً من المصطلحات اللغوية الصعبة والمصطلحات التقنية التي لا يفهمها إلا المختصون، ويسهل الوصول إليه عند الحاجة، ويسهل الوصول إلى الجزء الذي يحتاجه المستخدم بيسر، ولذلك ينصح باحتواء الدليل على قائمة بمحتوياته وأرقام للصفحات.

3. التوثيق الخاص بموظفي التشغيل (Operators Documentation)

موظفو التشغيل هم الموظفون المسؤولون عن تشغيل النظام وحمايته والمحافظة عليه. ويحتاج هذا التوثيق إلى تفاصيل أكثر من غيره من أنواع التوثيق. ويسمى دليل التشغيل (Run Manual). وحتى يحقق هذا التوثيق الهدف الذي بني لأجله، يجب أن يحتوي على النقاط الآتية:

أ- الهدف من النظام.

ب- طريقة تشغيل النظام.

ج- كل الأخطاء المحتملة وطريقة التعامل معها.

د- معلومات تشغيل النظام، مثل:

- نماذج الطابعات.
- أسماء الملفات في قاعدة البيانات وأماكن وجودها.

- معلومات الأقراص (Disks) والشرائط (Tapes).
- طريقة توزيع المخرجات.

هـ- طريقة خلق مستخدمين جدد.

و- طريقة إعطاء كلمات المرور (Passwords) والصلاحيات (Privileges) للمستخدمين.

ز- طريقة عمل النسخ الاحتياطية (Backups) واسترجاعها (Recovery).

4. توثيق البرامج (Program Documentation)

يكون توثيق البرامج ببيان الهدف من كل برنامج وطريقة عمله ومنطقه. ويساعد هذا النوع من التوثيق على جعل البرامج أكثر قابلية للفهم والتعديل من قبل فرق مراجعة البرامج وفرق الفحص وفرق الصيانة، حيث يساعد هذه الفرق على القيام بعملها بسهولة ويسر. وقد أثبتت التجارب أن متابعة وفهم وتعديل البرامج التي تم بناء توثيق لها أسهل كثيراً من البرامج المكتوبة بدون توثيق.

ويحتاج هذا التوثيق إلى تفاصيل أكثر من بقية الأنواع حيث إن مستخدميه من المختصين فإنه يحتوي على مصطلحات تقنية متخصصة.

وتعتبر الجمل التوضيحية في البرنامج (Comments) أحد أشكال هذا التوثيق وتكون في بداية أجزاء البرنامج حيث تفسر الأفكار الجديدة المستخدمة في البرنامج أو الجمل غير الواضحة التي تحتاج إلى توضيح، ويمكن أن يصل عددها إلى نسبة واحد إلى واحد مع جمل البرنامج نفسها.

وبناء على ما سبق- عزيزي الدارس- فإن توثيق النظام يحتوي على الوثائق الآتية:

أ- التقرير المبدئي (Preliminary Report).

ب- دراسة الجدوى (Feasibility Study).

ج- توصيف النظام (System Specifications).

د- توصيف البرامج (Program Specifications).

هـ- دليل المستخدم (User Manual).

و- دليل التشغيل (Run Manual).

ز- دليل الإدارة (Management Documentation).

ح- توثيق البرامج (Programs Documentation).

4. الخلاصة

يعتبر تحول النظام من العمليات المهمة التي تساعد على إنجاح النظام الجديد والتي تهيئ الوضع للمستخدمين على العمل عليه بدون مشاكل. وإن الفشل في عملية التحول هذه قد يعني الفشل في تطبيق النظام الجديد ممّا يؤدي إلى فشل النظام نفسه.

وقد لاحظنا – عزيزي الدارس– من خلال هذه الوحدة أن توثيق النظام الجديد يحتوي على أنواع عديدة من الوثائق المهمة وبعضها يساعد على الحفاظ على النظام على المدى البعيد وبعضها يساعد على سهولة فهمه وتعديله وبعضها الآخر يساعد على استخدام النظام بطريقة فعالة.

5. مسرد المصطلحات

- **التحول التدريجي المرحلي Phased Parallel Conversion:** تعتبر هذه الطريقة حلاً وسطاً بين طريقة التحول المباشر والتحول الموازي حيث يتم تحول النظام بأجزائه كل جزء (Module) على حدة.
- **التحول المباشر Direct Conversion:** هنا يتم التحول بشكل فوري ولكل النظام ككتلة واحدة، حيث تتم إزالة النظام القديم ويستبدل بالنظام الجديد.
- **التحول الموازي Parallel Conversion:** في هذه الحالة يعمل النظامان بشكل متزامن حيث يتم إدخال البيانات إلى النظامين وكذلك يتم الحصول على المخرجات من النظامين.
- **تحول النظام System Conversion:** هو عملية تهيئة الوضع في المؤسسة صاحبة النظام الجديد للعمل على هذا النظام الجديد. ويبدأ العمل بالتحول قبيل الانتهاء من بناء النظام.
- **التقرير المبدئي Preliminary Report:** التقرير الناتج عن مرحلة التحليل المبدئي، ويبين ما إذا كانت المشكلة التي من أجلها قامت عملية التحليل حقيقية أم لا.
- **توثيق البرامج Programs Documentation:** أحد أنواع وثائق النظام ويبين الهدف من كل برنامج وطريقة عمله ومنطقه.
- **توصيف البرامج Program Specifications:** التقرير الثاني من تقارير مرحلة التصميم، ويبين وصف أجزاء النظام وبرامجه.
- **توصيف النظام System Specifications:** أحد التقارير الناتجة عن مرحلة تصميم

النظام، ويبين وصف الشاشات، والتقارير، وقاعدة البيانات، وهيكلية النظام.

- **دراسة الجدوى Feasibility Study:** التقرير الناتج عن مرحلة التحليل، ويبين الجدوى من بناء النظام الجديد.
- **دليل الإدارة Management Documentation:** أحد أنواع وثائق النظام ويستخدم من قبل الإدارة ويبين محتويات وطريقة استخدام والاستفادة من التقارير الإدارية الرئيسة.
- **دليل التشغيل Run Manual:** أحد أنواع وثائق النظام ويستخدم من قبل مشغلي النظام. يبين طريقة تشغيل النظام وحمايته والحفاظ عليه.
- **دليل المستخدم User Manual:** أحد الوثائق التي تستخدم من قبل المستخدمين ويساعدهم على استخدام النظام بطريقة فعالة.

6. المراجع

1. Aktas, Z., Structured Analysis and Design of Information Systems, Prentice-Hall, 1987.
2. Baudoin, C. and Hollowell, G., Realizing The Object-Oriented Life Cycle, Prentice- Hall, 1996.
3. Booch, G., Rumbaugh, J., and Jacobson, I., The Unified Modeling Language User Guide, Addison- Wesley, 1999.
4. Dennis, A. et al, Systems Analysis & Design: An Object-Oriented Approach with UML. John Wiley & Sons Inc., 2002.
5. Dewitz, S., Systems Analysis and Design and the Transition to Objects, McGraw-Hill, 1996.
6. Edwards, P., Systems Analysis, Design, and Development with Structured Concepts, Rinehart and Winston, 1999.
7. Henderson, P., Object-Oriented Specification and Design with C++, McGraw- Hill, 1993.
8. Hoffer, J. et al., Modern Systems Analysis & Design, Second Edition, Addison-Wesley, 1999.
9. Kendal, K., and Kendal, J., Systems Analysis and Design, 5th Ed.,2002.
10. Kruchten, P., The Rational Unified Process: An Introduction. Second Edition, Addison- Wesley, 2000.
11. Lejk, M. and Deeks. An Introduction to Systems Analysis Techniques, 2nd Ed., Pearson Education Limited, 2002.
12. Maciaszek, A., Requirements Analysis and System Design, Developing Information Systems with UML, Addison- Wesley, 2001.
13. Quatrani, T., Visual Modeling with Rational Rose 2000 and UML, Addison- Wesley, 2000.
14. Satzinger, J., Systems Analysis and Design, 2nd ed., Thomson Learing, 2002.
15. Silver, G. and Silver, M., Systems Analysis and Design, Addison-Wesley, 1996.
16. http://www.psut.edu.jo / computer science/ staff/saleh abu-soud
17. www.rational.com
18. www.omg.com