

RUP: Rational Unified Process

RUP Is an iterative software development process هي عملية تطوير برمجيات بتكرارية

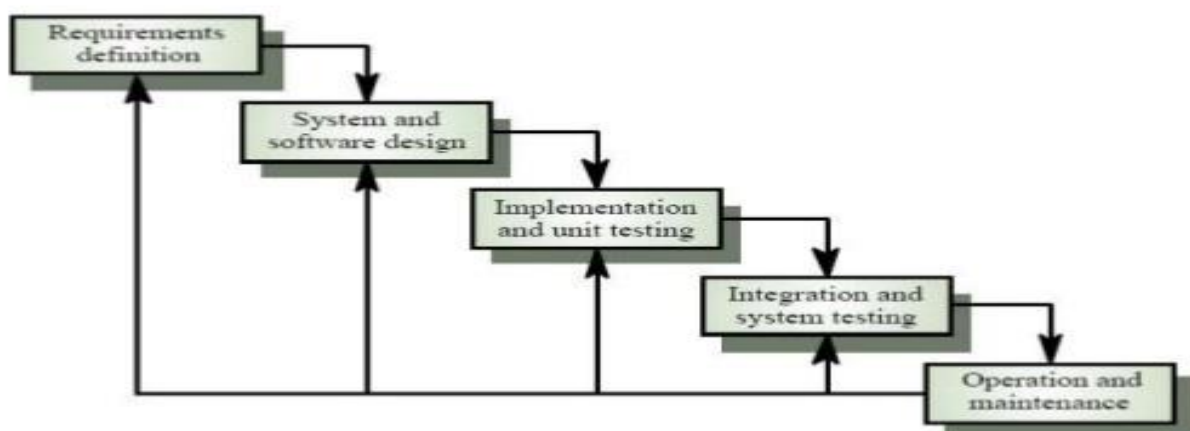
Software process models can be combined for designing large systems

يمكن دمج نماذج عملية سير البرمجيات لتصميم أنظمة كبيرة

Waterfall Model نموذج الشلال

transform mathematical model of a system into executable code using mathematical transformation that preserve its consistency.

تحويل نموذج رياضي لنظام الى كود يتم تنفيذه باستخدام تحويل رياضي للمحافظة على تناسقية النظام



Requirements and definition المتطلبات والتعريفات	The system contains, services and goals قيود النظام وخدماته وأهدافه
System and software design تصميم النظام والبرنامج	System design establish an overall architecture تصميم النظام يأسس هيكل عام Software design identify and describe the fundamental software system abstractions and their relationships تصميم البرنامج يعرف ويوصف المكونات الرئيسية المجردة للنظام البرمجي وعلاقاتهم
Implementation and unit testing التطوير واختبار الوحدات	Implementation: software design is realized as a set of programs or program units التنفيذ: تصميم البرنامج يدرك على انه مجموعة من البرامج او وحدات برامج unit testing verify that each unit meets its specifications اختبار الوحدات: يتأكد من ان كل وحدة تؤدي المواصفات المفروضة لها مسبقا
Integration and system testing التكاملية واختبار النظام	Individual program units or programs are integrated and tested as a complete system to ensure that the requirements have been meet وحدات البرنامج المنفصلة او البرامج يتم دمجها معا واختبارها كنظام كامل للتأكد من تطبيق كل المتطلبات After testing the software is delivered to customers بعد الاختبار: يتم تسليم البرنامج للعميل
Operation and maintenance التشغيل والصيانة	Operation the system installed and put into practical use التشغيل: النظام يتم تثبيته ووضع في موضع التنفيذ العملي Maintenance involves correcting errors which were not

discovered in earlier stages of the life cycle
الصيانة: تتضمن تصحيح الأخطاء التي لم تكتشف في المراحل الأولى لدورة حياة البرنامج

Advantages of waterfall model

مميزات نموذج الشلال

- 1- documentation is produced with each phase
إنتاج توثيق لنهاية كل مرحلة
- 2- fits with other engineering process models
ملائم للاستخدام مع نماذج هندسة العمليات

Disadvantages (problems) of waterfall model

عيوب أو مشاكل استخدام نموذج الشلال

- 1- the difficulty of accommodating change after the process is underway
صعوبة إضافة تعديلات أثناء عمل المرحلة
- 2- One phase has to be complete before moving onto the next phase
مرحلة واحدة لا بد أن تنتهي قبل الانتقال إلى المرحلة التالية لها
- 3- partitioning of the project into distinct stages
تجزئة المشروع إلى مراحل مستقلة

this model is only appropriate when the requirements are well-understood and changes will be limited during the design process

هذا النموذج ملائم فقط عندما تكون المتطلبات مفهومة جيدا والتعديلات ستكون محدودة خلال مرحلة التصميم

Applicability of waterfall model

قابلية تطبيق نموذج الشلال

Used in some old large systems, now its recommended to design large systems if mixed with other models or sub-systems in large systems

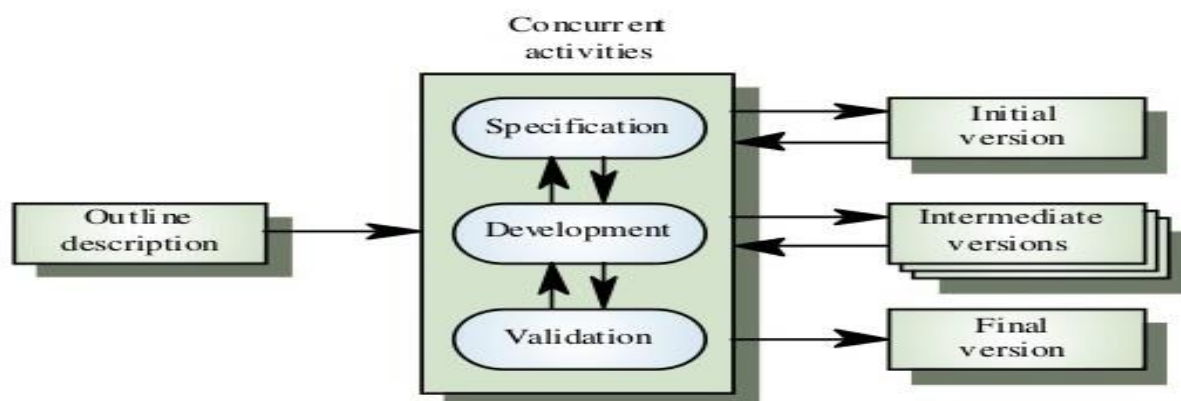
استخدم في الأنظمة الكبيرة القديمة الآن يوصى به لتصميم الأنظمة الكبيرة بشرط أن يدمج مع نماذج أخرى أو تصميم الأنظمة الفرعية بالأنظمة الكبيرة

Evolutionary (iterative) development

التطوير بالتكرارية

Based on the idea of developing an initial implementation then refine it through customer comments until a stable system has been developed.

قائم على فكرة تطوير نسخة ابتدائية للنظام ثم إعادة تحسينها من خلال تعليقات العميل عليها وتعد العملية إلا أن تطور نظام ثابت



Types of Evolutionary (iterative) development

أنواع التطوير بالتكرار

1- Exploratory development

التطوير الاستكشافي

1.1 Objective of the process is to work with customers to explore their requirements and deliver a final system. هدف المرحلة العمل مع العميل لاستكشاف متطلباته و انتاج نظام نهائي

1.2 The development starts with the parts of the system that are understood.

التطوير يبدأ من أجزاء النظام المفهومة

1.3 the system evolves by adding new features proposed by the customer.

يتم تطوير النظام باضافة مميزات جديدة مقترحة من المستهلك

2- Throwaway prototyping

النماذج المهمة

2.1 Objective is to understand the system requirements. الهدف فهم متطلبات النظام

2.2 Should start with poorly understood requirements to clarify what is really needed. يجب ان تبدأ من فهم غير صحيح للمتطلبات لاستيضاح ما يجب فعليا معرفته

Advantages of Evolutionary (iterative) development model

مميزات التطوير بالتكرار

1- specification can be developed incrementally المواصفات يتم تطويرها تدريجيا

2- customers develop a better understanding of their problem العملاء ينسجرو فهم افضل لمشكلاتهم

Disadvantages (problems) of Evolutionary (iterative) development model

عيوب التطوير بالتكرار

1- Lack of process visibility عدم وجود رؤية لاتمام العملية

2- Systems are often poorly structured الانظمة غالبا تصمم بشكل سئ (ضعيف)

Applicability of Evolutionary (iterative) development model

التطبيق

Used for small and medium-size systems يستخدم فى الانظمة متوسطة او صغيرة الحجم

Component-based software engineering (CBSE)

هندسة البرمجيات القائمة على التجزئة

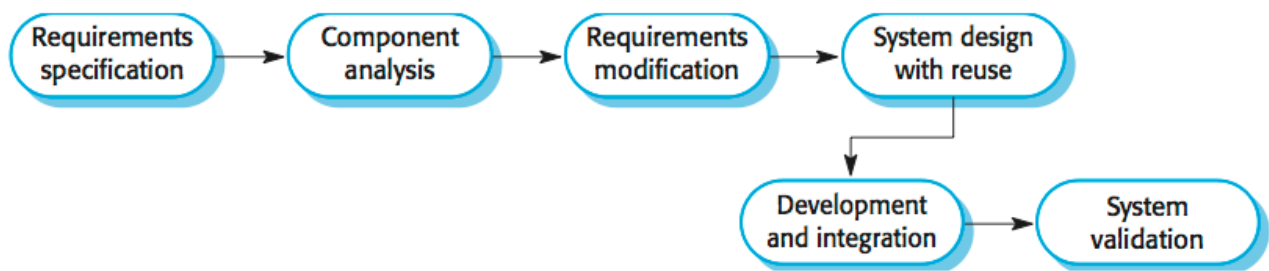
Based on systematic reuse where systems are integrated from existing components or **COTS** (Commercial Off-The-Shelf) systems

قائم على اعادة استخدام الانظمة بمعنى الانظمة يتم تكاملها من اجزاء موجودة او انظمة البرمجيات التجارية الجاهزة

COTS provide specific functionality such as text formatting or numeric calculation

البرمجيات التجارية الجاهزة تقدم وظائف محددة مثل تنسيق النصوص او حسابات عددية

Reuse-oriented development



Component analysis تحليل المكونات	A search is made for the components to implement the requirements specification يتم عمل بحث للمكونات التي سوف تنفذ مواصفات المتطلبات
Requirements modification تعديل المتطلبات	Requirements are analyzed using the information discovered about the components then modified المتطلبات يتم تحليلها باستخدام المعلومات المكتشفة عن الجزاء ثم يتم تعديلها
System design with reuse تصميم النظام مع اعادة الاستخدام	Design the framework of the system or existing framework is reused صمم هيكل النظام او اعد استخدام هيكل موجود مسبقا
Development and integration التطوير والدمج	Develop the system if it can't be externally developed التطوير: النظام لا يمكن تطويره خارجيا (يعني مفيش مكونات جاهزة مسبقا) Integrate the COTS and components to create the new system الدمج: البرمجيات التجارية الجاهزة والمكونات التي تستخدم لانشاء النظام الجديد

Advantages of Component-based software engineering model المميزات

- 1- reduce the amount of software to be produced تقليل وقت انتاج البرمجيات
- 2- reduce cost تقليل التكلفة
- 3- reduce risks تقليل المخاطر
- 4- faster delivery for software تسليم اسرع للبرمجيات

Disadvantages (problems) of Component-based software engineering model العيوب

- 1- compromises requirements تسوية المتطلبات (بمعنى الاتفاق على حل وسط للمتطلبات)
- 2- resulted system may not meet the customer needs النظام الناتج لا يحقق احتياجات المستخدم
- 3- some control over the system evolution is lost as some reusable components are not under the control of the same organization using them

بعض التحكم في تطوير النظام يتم فقده حيث بعض المكونات المعاد استخدامها ليست تحت تحكم نفس المؤسسة التي تستخدمهم

Applicability of Component-based software engineering model التطبيق

Used for large systems design تستخدم في تصميم الانظمة الكبيرة

Process iteration تكرار العملية

System requirements **ALWAYS** evolve in large systems so process iteration where earlier stages are reworked is always part of the process for these systems

متطلبات النظام دائما تتطور في الانظمة الكبيرة ولذلك تكرر العملية في مراحل الابتدائية يتم اعادة صياغتها ويعد هذا جزء من عمليات هذه الانظمة

Types of Process iteration

أنواع تكرارات العمليات

1- Incremental delivery

التسليم التدريجي

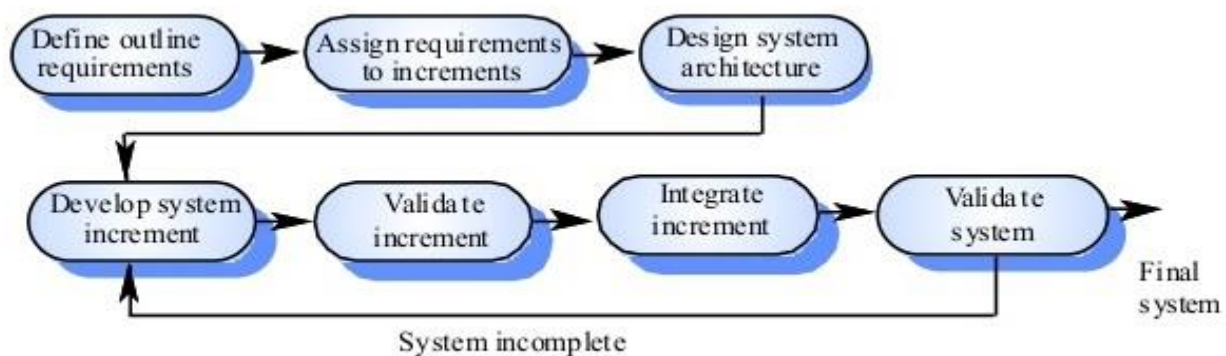
Software specification and implementation are broken down into a series of increments and each turn an increment is being developed

مواصفات البرنامج وتنفيذه يتم تجزئتهم الى سلسلة من التدرجات وفي كل خطوة يتم تطوير تدرج واحد

Customers define the priority of each increment the higher the priority is the earlier it will be included
المستهلكون يحددوا اولوية كل تدرج كلما علت الاولوية يضاف التدرج باكرا

Once the development of an increment is started its requirements are frozen

عندما يبدأ تطوير تدرج ما تتجمد (لا يمكن التعديل فيها او الجوع اليها) متطلباته



Advantages of the Incremental delivery

المميزات

1- Customers can benefit for the system while its being developed

العملاء يستفيدوا من النظام قبل انتهاء تطويره

2- Early increments act as a prototype to help elicit requirements for later increments
التدرجات الباكرا تعمل كنماذج ابتدائية للمساعدة في انتخاب (ترشيح) المتطلبات للتدرجات الاتية

3- Lower risk of overall project failure

تقليل المخاطر لفشل النظام في بشكل عام

4- The highest priority system services tend to receive the most testing

خدمات النظام الاعلى في الاولوية تميل الى ان تمنح اختبارات مكثفة

Disadvantages of the Incremental delivery

العيوب

1- Increments should be small

التدرجات يجب ان تكون صغيرة

2- each increment should deliver some system functionality

كل تدرج يجب ان يمدنا ببعض وظائف النظام

3- difficulty of mapping user requirements onto increments of the right size

صعوبة توفيق متطلبات المستخدم مع تدرجات بحجم مناسب

4- hard to identify the common facilities which are needed by all increments

صعوبة التعرف على التسهيلات العامة التي يتم احتياجها لكل التدرجات

2- Spiral development

التطوير الحلزوني

Development of the system spirals outwards from the initial outline through to the final developed system
تطوير النظام يتدرج حلزونياً الى الخارج ابتداء من المخطط التمهيدي الى انتاج المشروع النهائي

Software Process is represented as a spiral rather than as a sequence of activities with backtracking from one activity to another

سير عملية البرمجة يمثل على شكل حلزوني بدلاً من متتالية من الانشطة مع امكانية الجوع من نشاط الى نشاط اخر

Each loop in the spiral represents a phase in the process

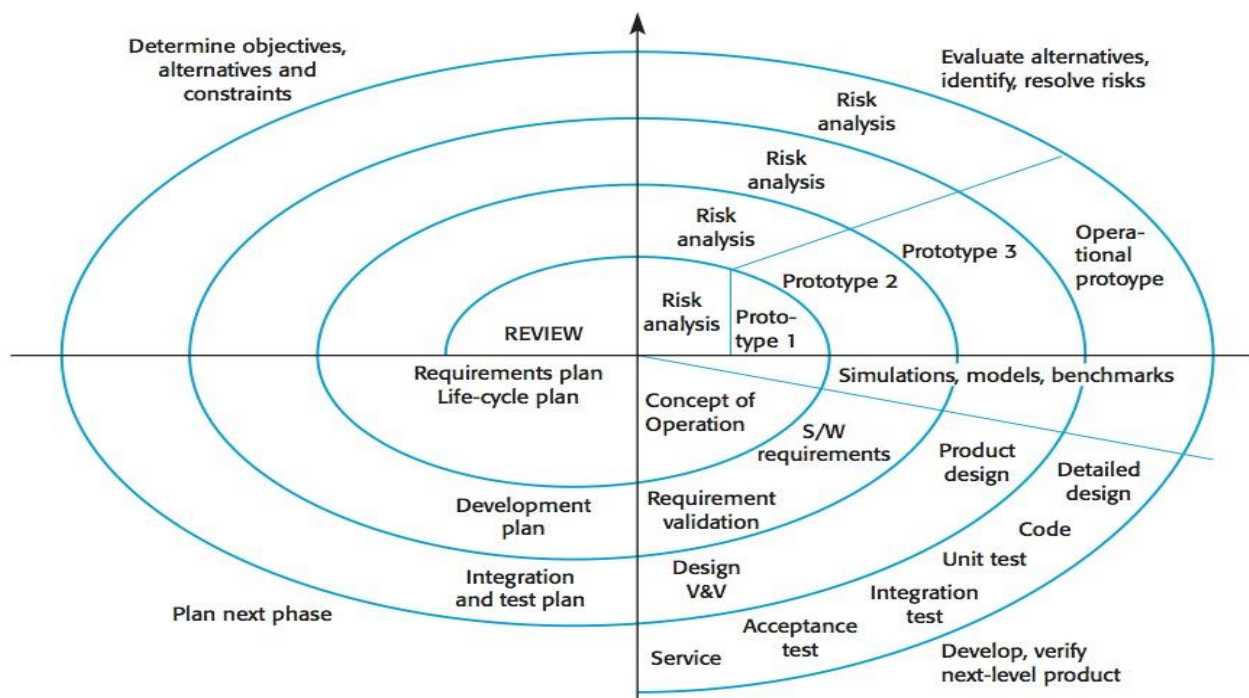
كل لفة في النموذج الحلزوني تعبر عن مرحلة من مراحل العملية

loops in the spiral are chosen depending on the required phase

عدد اللفات في النموذج الحلزوني تحدد بناء على عدد المراحل

Risks are explicitly assessed and resolved throughout the process

المخاطر تقيم بشكل واضح ويتم حلها خلال شير العملية



Spiral model sectors

قطاعات النموذج الحلزوني

Objective setting تحديد الاهداف	Specific objectives for the phase are identified تحديد أهداف محددة للمرحلة
Risk assessment and reduction تقييم المخاطر وتقليلها	A detailed analysis is carried out for each identified risk then steps are taken to reduce this risk تحليل مفصل يتم عمله لكل خطأ يتم ايجاده
Development and validation التطوير والتحقق من صحة النظام	After risk evaluation a development model for the system is chosen using any of the generic models بعدها يقيم الخطأ يتم اختيار نموذج تطوير للنظام باستخدام اى نموذج من النماذج العامة
Planning التخطيط	The project is reviewed and the next phase of the spiral is planned if its required يتم استعراض المشروع و يتم البدء فى التخطيط للمرحلة التالية فى النموذج الحلزوني ان كانت مطلوبة

Advantages of the Spiral development

فوائد النموذج الحلزوني

1- risk minimization

يقلل من المخاطر

How Process Activities can be organized

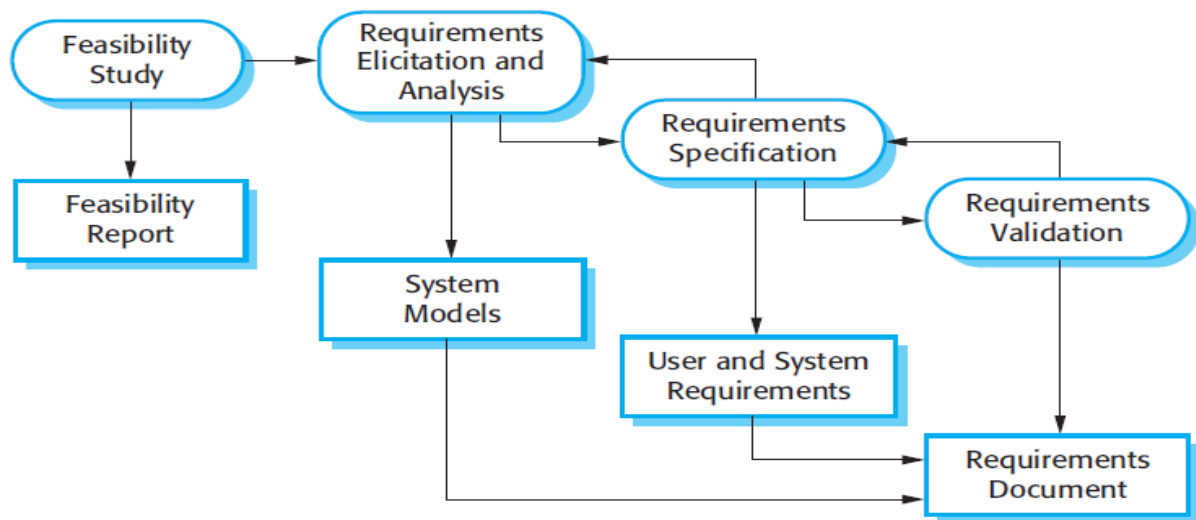
كيف يمكن تنظيم الانشطة الخاصة بسير عملية ما

1- Software specification (requirements engineering process)

توصيفات المتطلبات (متطلبات هندسة العملية)

Process of understanding and defining what services are required form the system as well as the constrains on the system operations

هى مرحلة فهم وتعريف الخدمات المراده من هذا النظام بالاضافة الى القيود الخاصة بعملياته



Main phases of the requirements engineering process

المراحل الرئيسية للمتطلبات

Feasibility study دراسة الجدوى	The study consider whither the proposed system will be cost-effective from a business point of view الدراسة تأخذ بعين الاعتبار ما اذا كان النظام المقدم على الكفاءة بالنسبة لتكلفته من وجهة نظر استثمارية
Requirements elicitation and	Acquire the system requirements through the observation (analysis) of existing systems

analysis انتخاب المتطلبات والتحليل	الحصول على متطلبات النظام من خلال متابعة (تحليل) الانظمة الموجودة مسبقا
Requirements specification مواصفات المتطلبات	Translating the information gathered during the analysis into a document which defines the set of requirements Has two types User requirements and system requirements تحويل المعلومات التي جمعت خلال مرحلة التحليل الى توثيق يعرف مجموعة المتطلبات المراده وله نوعان توثيق للمستخدم وتوثيق للنظام
Requirements validation التحقق من صحة المتطلبات	The discovered errors in requirements document must be modified الخطاء التي تم اكتشافها في وثيقة المتطلبات يجب ان يتم تعديلها

2- Software design and implementation

تصميم وتنفيذ النظام

The process of converting the system specification into an executable system.

عملية تحويل مواصفات النظام الى نظام قابل للتنفيذ

Software design: Design a software structure that realizes the specification

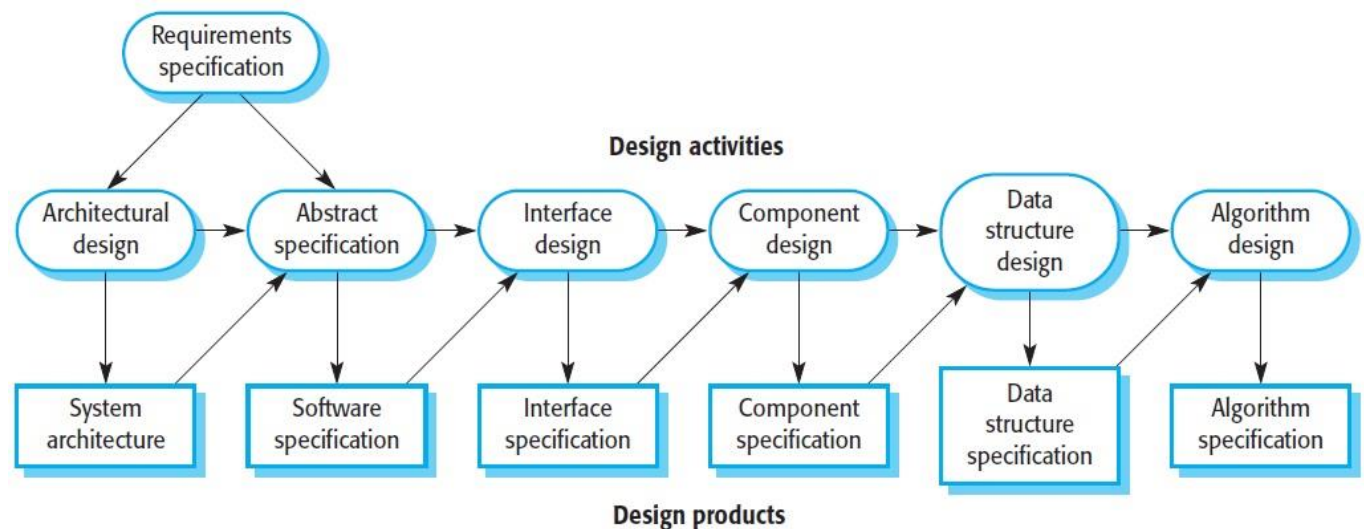
تصميم النظام: تصميم هيكل البرنامج الذي يحقق المواصفات

Implementation: Translate this structure into an executable program

التنفيذ: يترجم هيكل التصميم الى برنامج قابل للتنفيذ

The activities of design and implementation are closely related and may be interleaved

الانشطة الخاصة بالتصميم والتنفيذ متقاربة جدا من بعضها وقد تكون متداخلة



Design process activities

أنشطة مرحلة التصميم

Architectural design التصميم المعماري	Subsystems makeup the system and identify the relationships and document them الانظمة الفرعية تصنع النظام وتعرف العلاقات والتوثيق الخاصة به
Abstract specification المواصفات المجردة	Defines the abstract of each subsystem as well as its services and constrains which it must meet تعرف المجردات الخاصة بكل نظام فرعي كما انها تعرف الوظائف والقيود التي يجب تحقيقها
Interface design تصميم الواجهه	Each subsystem interface which work with other subsystems is designed and documented

	كل واجهه لنظام فرعى والتي تعمل مع انظمة فرعية أخرى تصمم وتوثق
Component design تصميم المكونات	Services are allocated to components and components interfaces are designed الخدمات تمنح للمكونات وواجهات المكونات تصمم
Data structure design تصميم هياكل البيانات	The data structures used in the system implementation are designed in detail هياكل البيانات المستخدمة في تنفيذ النظام تصمم بالتفصيل
Algorithm design تصميم الخوارزمية	Algorithms used to provide services are designed in detail الخوارزميات تستخدم لتقديم الخدمات يتم تصميمها بالتفصيل

Algorithms adaptation methods

أساليب تأقلم الخوارزميات

1- data structure design and algorithm design may be delayed until implementation process
تصميم هيكل بيانات و تصميم خوارزمية قد يتأخر حتى مرحلة التنفيذ

2- interface may be designed after the data structure is specified if the design is exploratory
الواجهه قد تصمم بعد تحديد هيكل البيانات اذا كان التصميم استكشافى

3- Abstract specification stage maybe skipped although it's an essential part of critical system design
مرحلة المواصفات المجردة قد يتم تعديها بالرغم من اهميتها فى تصميم النظام

Programming and debugging

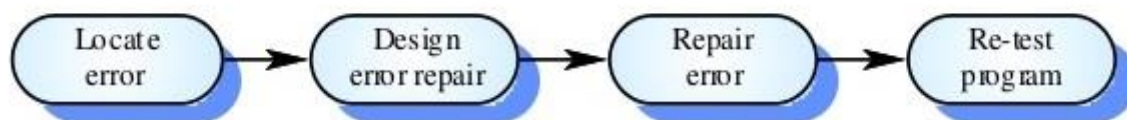
البرمجة وتصحيحها

Translating a design into a program and removing errors from that program

ترجمة التصميم الى برنامج وازالة الاخطاء من البرنامج

Programmers carry out some program testing to discover faults in the program

المبرمجين ينفذوا بعض برامج الاختبار لاكتشاف الاخطاء فى البرامج

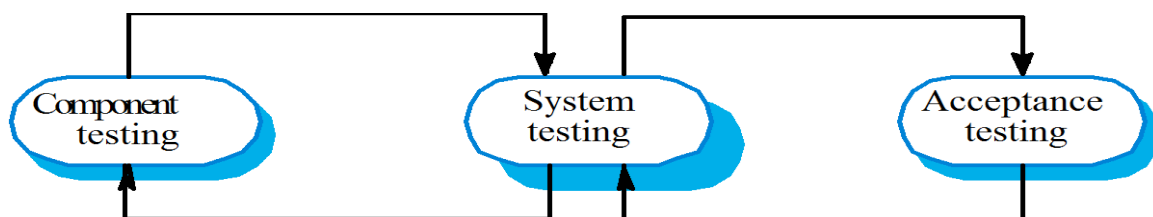


3- software validation

التحقق من البرامج

Involves checking and review processes and system testing to conforms if the system meets the requirements of customer

تتضمن التحقق من ومراجعة مراحل واختبارات النظام للتأكد من النظام يحقق متطلبات المستخدم

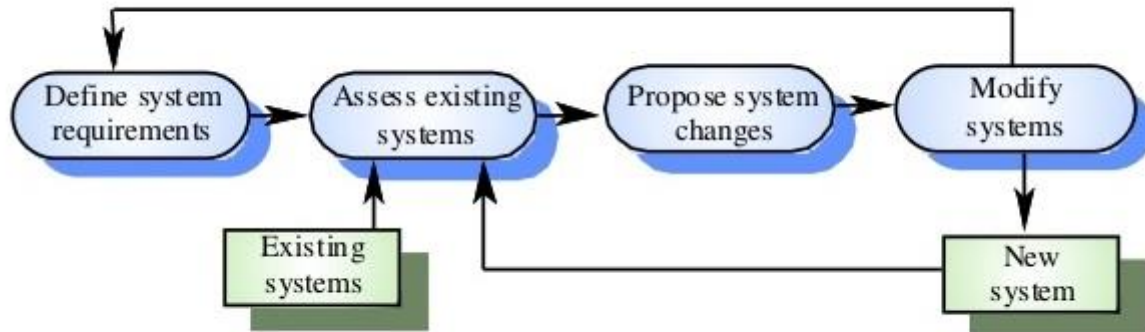


4- software evolution

تطوير البرامج

As requirements change through changing business circumstances, the software that supports the business must also evolve and change.

بما ان المتطلبات تتغير بسبب تغير ظروف العمل كذلك البرامج الداعمة للعمل يجب ان تطور وتتغير



Computer-Aided Software Engineering (CASE)

هندسة البرمجيات بمساعدة الحاسوب

Automated activities using CASE استخدام الأنشطة الآلية في هندسة البرمجيات بمساعدة الحاسوب

1- development of graphically system models as part of the requirements specifications
تطوير نماذج انظمة رسومية كجزء من مواصفات المتطلبات

2- understanding a design using data dictionary
فهم تصميم باستخدام قاموس بيانات

3- generation of user interface from a graphical interface
توليد واجهه المستخدم من واجهه رسومية

4- program debugging using previous data about existing program

تصحيح اخطاء البرنامج باستخدام بيانات سابقة عن برنامج موجود مسبقا

5- translation of a program form an old programming language to a more recent version
ترجمة برنامج من لغة برمجة قديمة الى اصدار احدث

improvements limitations in CASE قيود تطور هندسة البرمجيات بمساعدة الحاسوب

1-attempting to use artificial intelligence to provide support for design have not been successful.
محاولة استخدام الذكاء الاصطناعي لتقديم الدعم في التصميم لم تنجح

2- doesn't support working as a team.
لا تدعم العمل كفريق