Ch.4

Software Engineering

Mohammad Jadallah

Ch 4: Requirements Engineering

Introduction

- The requirements for a system are the **descriptions** of what the system should do, the **services** that it provides and the **constraints** on its operation.
- These requirements reflect the needs of customers for a system that serves a certain purpose such as controlling a device, placing an order, or finding information

2

* Requirements

وصف النظام (الخدمات التي سيقدمها النظام

(functional & non functional

* فالبداية يتم كتابة ال Req على شكل (high Level) .

* non functional = constrains

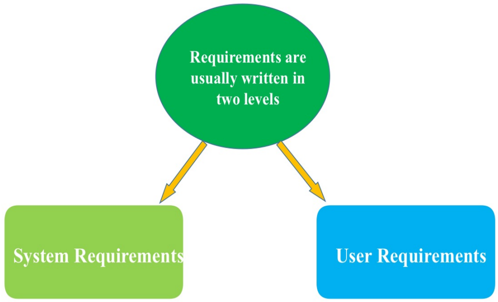
=> نعني بـ high Level كتابة ال main fetchers (كتابة المهام الأساسية فقط)

=> بعد ذلك يتم كتابة تفصيل كامل لهذه ال Req .

* هذه الطريقة هي من أكثر الطرق المحببة لدى الشركات

(كتابة ال Req على 2 Levels)

Requirements' Levels



نسمي هذين ال Levels ع التالي :

1 User Req

وهي ال high level (main fetchers)

2 System Req

وهي ال low level (تفصيل كل fetcher)

نعود للمثال المذكور في نهاية Ch 2 مثلاً Search book in Library system ← وهي main fetcher (User Req) التفصيل لها (System Req)

ملاحظة : فالملاحظة التي يوافق عليها ال customer ← جميع ال Req
تعتبر هذه ال Req بمثابة عقد ومرجع لكلا الطرفين.

Example: User vs. System Requirements

مثال تفصيلي ← User & System Requirements

User Requirement Definition

1. The MHC-PMS shall generate monthly management reports showing the cost of drugs prescribed by each clinic during that month.

System Requirements Specification

1.1 On the last working day of each month, a summary of the drugs prescribed, their cost, and the prescribing clinics shall be generated.

1.2 The system shall automatically generate the report for printing after 17.30 on the last working day of the month.

1.3 A report shall be created for each clinic and shall list the individual drug names, the total number of prescriptions, the number of doses prescribed, and the total cost of the prescribed drugs.

1.4 If drugs are available in different dose units (e.g., 10 mg, 20 mg) separate reports shall be created for each dose unit.

1.5 Access to all cost reports shall be restricted to authorized users listed on a management access control list.

User Requirement Definition

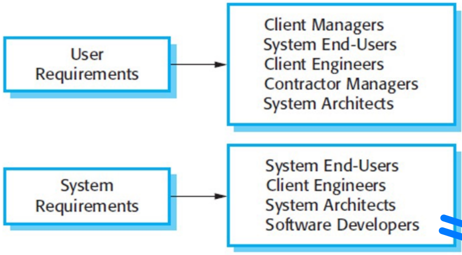
1. The MHC-PMS shall generate monthly management reports showing the cost of drugs prescribed by each clinic during that month.

The user Requirement:
شرح النقطة بشكل عام
لنظام مجمع طبي
⇐ النظام يجب أن يقوم بعمل إيصالات بشكل شهري تقوم باظهار تكلفة الأدوية التي تم وصفها من قبل كل دكتور

System Requirements Specification

1.1 On the last working day of each month, a summary of the drugs prescribed, their cost, and the prescribing clinics shall be generated.
1.2 The system shall automatically generate the report for printing after 17.30 on the last working day of the month.
1.3 A report shall be created for each clinic and shall list the individual drug names, the total number of prescriptions, the number of doses prescribed, and the total cost of the prescribed drugs.
1.4 If drugs are available in different dose units (e.g., 10 mg, 20 mg) separate reports shall be created for each dose unit.
1.5 Access to all cost reports shall be restricted to authorized users listed on a management access control list.

System Requirements:
جميع هذه النقاط هي تفصيل وشرح تتبع الـ user Req السابقة.
⇐ ليس بالضرورة أن تكون هذه تشمل جميع التفاصيل من الممكن أن تكون نسخة أولى ويتم التعديل عليها وإضافة النواقص فيما بعد
⇐ نلاحظ أن الترقيم مهم للتوضيح فإذا كانت الـ user Req رقمها 1 جميع نقاط الـ System تكون بهذا الشكل 1.1, 1.2, 1.3... إذا كانت 2 تكون 2.1, 2.2, 2.3... وهكذا وتسمى هذه الطريقة Requirements tracing
⇐ يجب أن يكون تفصيل كامل ودقيق مثلاً عند طباعة الـ report يجب أن يطبع رسالة للمستخدم، هذه النقطة ناقصة يجب أن يكون مكتوب ما هو محتوى الرسالة التي ستظهر مثلاً تمت العملية بنجاح وإلا فإن هناك نقص في هذه النقطة
⇐ باختصار يجب كتابة الفكرة العامة في الـ user Req وتفصيلها بشكل كامل دقيق في الـ System Req



① مصطلح Stake holders
يعني من المستفيدين من الشيء
مثلاً من هم ال Stake holders
من Ritaj ؟ الطلاب ، المعلمين
الموظفين وهكذا ..

* هنا من هم ال Stake holders من كل من ال user و ال System Reqs

② تذكيري ال Req (functional, non functional) ns

Functional and Non-Functional Requirements

- **Functional Requirements:** These are statements of services the system should provide, how the system should react to particular inputs, and how the system should behave in particular situations.

- **Non-Functional Requirements**: These are constraints on the services or functions offered by the system.

⊗ Requirements Properties

Requirements Properties

كيف يجب أن تكون مانتوية هذه ال Req (صفاتها)

- In general, the system requirements should be
  1. • Correct
  2. • Consistent
  3. • Complete
  4. • Verifiable
  5. • Unambiguos
- This however will not be easy in large and complex systems with many **stakeholders**

[1] Correct: تعني أن هذه ال Req مفيدة لل customer

[2] Consistent: وجود تناقض وتعارض بين ال Requirements مثلاً فالتسجيل يوجد استثناءات تخص رؤساء الدوائر وأقوى من التقييد يوجد بينهم تناقض

[3] Complete: The system should be able the users to register by him information هنا ليست complete لأنه لم يتم توضيح ما هي ال information

[4] Verifiable: تم التحقق منها ومن صحتها

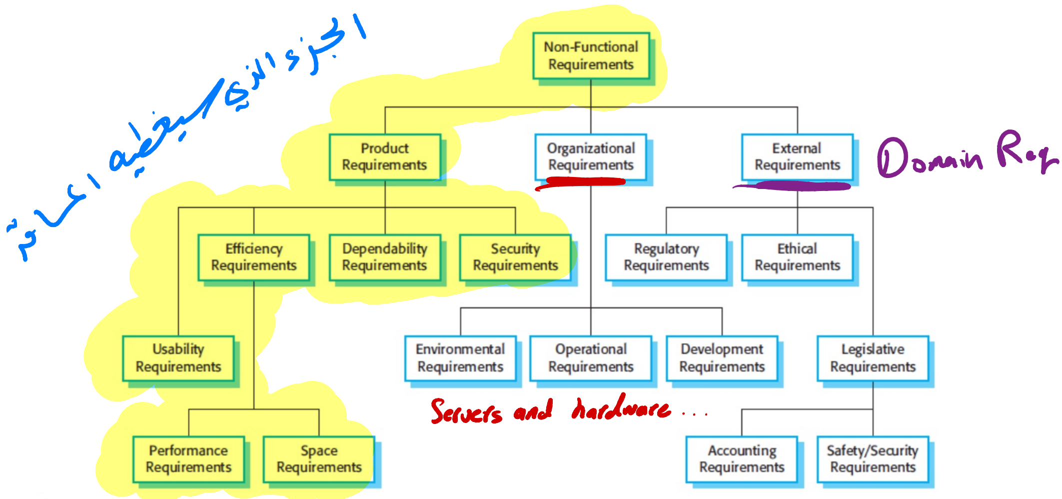
[5] Unambiguous: أن تكون واضحة ويتم فهمها بشكل واحد فقط

في بعض ال Requirements يكون لا يوجد فاصل واضح بين ال functional وال non functional مثل: all user shall be register throw his name and password

ملاحظة: إجابية functional & non functional → هي functional non ولكن أيضاً هذه العملية تحتاج إلى وجود عملية تسجيل الدخول ووضع خيار في حال نسيان ال password

هكذا أصبح لدينا functional Req ...

ملاحظة: عند وجود كلمة shall في الـ Req يعني أن هذا البند إلزامي (Mandatory, Must) أما should فهي قابلة للتفاوض من الممكن وضعها ومن الممكن لا.



Domain Requirements

- Domain requirements in software analysis refer to the specific needs, constraints, and conditions that are inherent to the domain or industry in which the software will operate.

- These requirements are influenced by the characteristics, practices, regulations, and standards of the domain and must be considered during the analysis and design phases of software development.

أي أنها تتبع قوانين هذا النوع من الـ apps ولا تأتي من الـ customer

⇐ هي Requirements تفرض على جميع التطبيقات في مجال معين لتتبعها
مثل: أن تفرض وزارة الصحة شروط معينة على جميع التطبيقات لاعتمادها وقبولها
حتى لو كان هذا السيستم ليس critical مثلا هو مجرد تنظيم أدوات الموظفين ومعلوماتهم يجب الالتزام بهذه القواعد

Domain Req أمثلة عليها

Example Domain Requirements for E-Commers web App

- **Payment Gateway Integration**: In the domain of ecommerce, one of the primary requirements is to integrate with various payment gateways to facilitate secure online transactions. The software needs to support multiple payment methods such as credit/debit cards, digital wallets, and bank transfers. Additionally, it must comply with relevant payment card industry (PCI) standards to ensure the security of sensitive financial information. Payment card industries
- **Product Categorization and Filtering**: Ecommerce platforms often require sophisticated product categorization and filtering functionalities to help customers easily find and navigate through a large inventory of products. Domain requirements might specify the need for features such as hierarchical category structures, product attribute filtering, and sorting options based on relevance, price, or popularity.

بناء ويب لشراء أغراض مختلفة

يجب أن يدعم أكثر من طريقة للدفع أون لاين، ليس بالضرورة أن يطلب الـcustomer ذلك، ولكن هذا من طبيعة هذا النوع من التطبيقات

كذلك هناك ستاندرد عالمية للسيورتي يجب اتباعها كأن الـcustomer (زبون الموقع) سيدخل معلومات بطاقة ائتمانية ويجب الحفاظ على سريتها وتشفيرها

تصنيف المنتجات بناءً أيضاً على الستاندرد العالمية لذلك، غالب الحوانيت المشهورة من هذا النوع تجد أن شجرة التصنيفات للمنتجات عندها متشابهة، لأنهم يتبعون نفس الستاندرد

=> More Examples From Dr. Samer slides

- **Shipping and Logistics Integration**: Another important domain requirement is the integration with shipping and logistics services to facilitate order fulfillment and delivery. The software should support real-time calculation of shipping costs based on factors such as package weight, destination, and preferred shipping method. It should also generate shipping labels and tracking information to keep customers informed about the status of their orders.
- **Tax Calculation and Compliance**: Ecommerce websites operating in different regions or countries must comply with local tax regulations and calculate appropriate taxes for each transaction. Domain requirements may include the ability to automatically calculate taxes based on the customer's location, product type, and applicable tax rates. The software should also generate accurate tax invoices and reports for accounting and tax compliance purposes.
- **User Authentication and Authorization**: Security is paramount in the ecommerce domain, so domain requirements often include robust user authentication and authorization mechanisms. The software should support secure user registration, login, and password management features. Additionally, it should enforce access controls to restrict certain functionalities or data based on user roles and permissions.

موضوع توصيل المنتجات وحساب تكلفة الطرد عن طريق معادلات خاصة عالمية مثلاً الإعتماد على الحجم والوزن وهكذا

حساب الضرائب بناءً على موقع الزبون (قوانين الدولة التي يعيش فيها)

عملية تسجيل الدخول وربطه مع الإيميل الشخصي وإرسال الرسائل لكل شخص على إيميله

=> صحيح جداً Domain Req ليس بالضرورة أن يطلبها الـcustomer

Example..3

الدكتور ذكر أمثلة تحتها

- Example1: a train control system has to take into account the braking characteristics in different weather conditions.
- Example2: a PMS has to enforce all confidentiality rules in accordance with national medical domain practices

May generate new functional requirements, and new constraints on existing requirements or define specific computations
If domain requirements are not satisfied, the system may be unworkable

② الـ Domain Req بتقل إضافة لـ Reqs جديدة سواء functional أو non functional ولازم يتم التماشي معها، بتشكل عبء إضافي عالنظام

Goals and Requirements

Non-functional requirements may be very difficult to state precisely and imprecise requirements may be difficult to verify.

Goal

A general intention of the user such as ease of use

Verifiable non-functional requirement

A statement using some measure that can be objectively tested

Goals are helpful to developers as they convey the intentions of the system users

① الـ non functional Reqs

فالبداية تكتب من وجهة نظر الـ user تسمى (Goal)، ولا تكون مكتوبة بالشكل النهائي والصحيح (لا تكون measurable)

مثلاً التعميم بأن نقول نريد أن يكون التطبيق سهل الاستخدام هذا هدف ولين non-functional Req

Non-Functional Requirements Should be Quantifiable

= not Subjective

يعني معناه يجب أن لا تكون هناك إمكانية لفهمها بأكثر من طريقة عند قراءتها من قبل أكثر من شخص

يجب أن تكتب بطريقة يمكن قياسها

not measurable → goal not (non functional Req)

measurable non functional Req

- **A system goal**
  G.8.1 The PMS system should be easy to use by medical staff and should be organized in such a way that user errors are minimized. (**Goal**) ✗
- **A verifiable non-functional requirement**
  8.4.3 Medical staff shall be able to use all the PMS system functions after four hours of training. After this training, the average number of errors made by experienced users shall not exceed two per hour of system use. (**Testable non-functional requirement**) ✓

⑧

④ أمثلة على كيفية جعل الـ non functional Req يمكن قياسها (measurable)

Metrics for Specifying Non-functional Requirements

Property	Measure
Speed	Processed transactions/second User/event response time Screen refresh time
Size	Mbytes Number of ROM chips
Ease of use	Training time Number of help frames
Reliability	Mean time to failure Probability of unavailability Rate of failure occurrence Availability
Robustness	Time to restart after failure Percentage of events causing failure Probability of data corruption on failure
Portability	Percentage of target dependent statements Number of target systems

Speed: الوقت اللازم لإنجاز عمل معين

Size: كم عدد الـ bytes

Ease of use: Training time → usability: كم الوقت اللازم للتدريب على التطبيق؛ Number of help frames ← كم يحتاج مساعدة

Reliability: كم يحتاج وقت حتى يتعطل؛ احتمالية عدم القدرة على تشغيله؛ Availability → كم الوقت الذي سيكون فيه متاح

Robustness: احتمالية خسارة الداتا عند تعطل النظام

Portability: الأنظمة الموجه لها التطبيق مثل: mobile app (android, IOS)

ملاحظة: مثلاً في بعض الأنظمة ليس من الصحيح قياس الـ usability على وقت التدريب مثل أنظمة البنك والـ critical الأصح هو القياس بناءً على كمية العمل المنجز حتى لو كان وقت التدريب كبير كان الخطأ مثل هذه الأنظمة مكلف جداً فيجب أن يكون الشخص على علم تام بطريقة عمل النظام وهذا يتطلب وقت كبير لتعلم ذلك.

④ بالتالي لا يمكن التعميم لاستخدام طريقة قياس موحدة لجميع الأنظمة، كل نظام يختلف عن الآخر في طريقة التعامل معه.

The Software Requirements Specidication Document (SRS) Software Requirements Specification

- It is an official statement of what the system developers should implement.
- It should include both the user requirements for a system and a detailed specification of the system requirements
- **Agile methods** do not use official SRS, instead they use **User Stories**
- **Critical and Outsourcing Projects** needs detailed SRS document

SRS: يحتوي على الـ analasiys Model كامل (Req, Prototypes ...) يستخدم بالمشاريع التي تكون very formal (عادةً ما تكون موجودة بالمشاريع التي تتبع agile)

Level of Details When Writing Requirements

- The level of detail depends on the type of system that is being developed and the development process used.
- **Critical systems** need to have detailed requirements because safety and security have to be analyzed in detail.
- When the system is to be developed by a **separate company** the system specifications need to be detailed and precise.
- If an **in-house**, iterative development process is used, the requirements document can be much less detailed and any ambiguities can be resolved during development of the system.

Chapter	Description
Preface	This should define the expected readership of the document and describe its version history, including a rationale for the creation of a new version and a summary of the changes made in each version.
Introduction	This should describe the need for the system. It should briefly describe the system's functions and explain how it will work with other systems. It should also describe how the system fits into the overall business or strategic objectives of the organization commissioning the software.
Glossary	This should define the technical terms used in the document. You should not make assumptions about the experience or expertise of the reader.
User requirements definition	Here, you describe the services provided for the user. The non-functional system requirements should also be described in this section. This description may use natural language, diagrams, or other notations that are understandable to customers. Product and process standards that must be followed should be specified.
System architecture	This chapter should present a high-level overview of the anticipated system architecture, showing the distribution of functions across system modules. Architectural components that are reused should be highlighted.

تكملة

System requirements specification	This should describe the functional and non-functional requirements in more detail. If necessary, further detail may also be added to the non-functional requirements. Interfaces to other systems may be defined.
System models	This might include graphical system models showing the relationships between the system components, the system, and its environment. Examples of possible models are object models, data-flow models, or semantic data models.
System evolution	This should describe the fundamental assumptions on which the system is based, and any anticipated changes due to hardware evolution, changing user needs, and so on. This section is useful for system designers as it may help them avoid design decisions that would constrain likely future changes to the system.
Appendices	These should provide detailed, specific information that is related to the application being developed; for example, hardware and database descriptions. Hardware requirements define the minimal and optimal configurations for the system. Database requirements define the logical organization of the data used by the system and the relationships between data.
Index	Several indexes to the document may be included. As well as a normal alphabetic index, there may be an index of diagrams, an index of functions, and so on.

Requirements Specification

- Requirements specification is the process of writing down the user and system requirements in a requirements document.

- At this stage, **focus on what the system should do, not how the system will do it (design)**
- User requirements should be **simple** natural language who don't have technical background.
- User requirements can have some **intuitive diagrams** such as use case diagram.

Requirements Specification: Natural language

* Requirements مكتوبة بشكل بسيط ولغة محكومة

3.2 The system shall measure the blood sugar and deliver insulin, if required, every 10 minutes. (*Changes in blood sugar are relatively slow so more frequent measurement is unnecessary; less frequent measurement could lead to unnecessarily high sugar levels.*)

3.6 The system shall run a self-test routine every minute with the conditions to be tested and the associated actions defined in Table 1. (*A self-test routine can discover hardware and software problems and alert the user to the fact the normal operation may be impossible.*)

Requirements Specification: Structured Format

* Requirement مكتوبة بناءً على Structured Format وتستخدم تحديداً عندما يكون النظام very formal و very critical

Insulin Pump/Control Software/SRS/3.3.2

Function	Compute insulin dose: Safe sugar level.
Description	Computes the dose of insulin to be delivered when the current measured sugar level is in the safe zone between 3 and 7 units.
Inputs	Current sugar reading (r2), the previous two readings (r0 and r1).
Source	Current sugar reading from sensor. Other readings from memory.
Outputs	CompDose—the dose in insulin to be delivered.
Destination	Main control loop.
Action	CompDose is zero if the sugar level is stable or falling or if the level is increasing but the rate of increase is decreasing. If the level is increasing and the rate of increase is increasing, then CompDose is computed by dividing the difference between the current sugar level and the previous level by 4 and rounding the result. If the result, is rounded to zero then CompDose is set to the minimum dose that can be delivered.
Requirements	Two previous readings so that the rate of change of sugar level can be computed.
Pre-condition	The insulin reservoir contains at least the maximum allowed single dose of insulin.
Post-condition	r0 is replaced by r1 then r1 is replaced by r2.
Side effects	None.

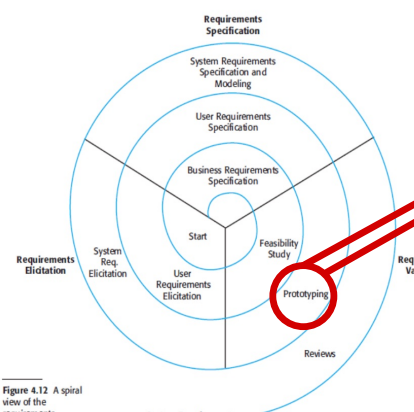
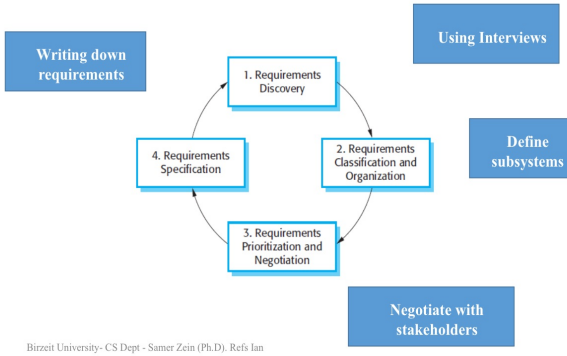


Figure 4.12 A spiral view of the requirements engineering process

ضرورة عمل Prototyping بعد انجاز كل Requirements

+ عمل اجتماع مع كل من ال team و ال customer لل review ال Reqs المكتوبة

Requirements Elicitation & Analysis Process (2+1)



Requirement Analysis (1)

The process of understanding customer requirements and their implications. It comes after requirements discovery

It involves technical staff working on the discovered requirements, iteratively with customers, to understand their technical implication and importance.

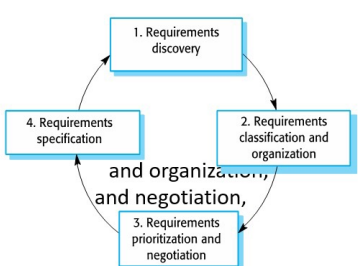
It includes the processes of classifying and organising requirements, negotiating (with customers) and prioritizing them in the order of their importance to the customers.

The output of this process is the requirement specification document (SRS) negotiated and accepted with customers.

Problems of requirements analysis (2)

- Stakeholders don't know what they really want.
- Stakeholders express requirements in their own terms.
- Different stakeholders may have conflicting requirements.
- The requirements change during the analysis process
- Political factors

- Stages include:
  - Requirements discovery,
  - Requirements classification
  - Requirements prioritization
  - Requirements specification.



Stakeholders in the MHC-PMS

- **Patients** whose information is recorded in the system.
- **Doctors** who are responsible for assessing and treating patients.
- **Nurses** who coordinate the consultations with doctors and administer some treatments.
- **Medical receptionists** who manage patients' appointments.
- **IT staff** who are responsible for installing and maintaining the system.
- etc

Techniques

There are many requirement engineering techniques for requirement elicitation and analysis, some of the often used ones:

1. Interviewing
2. Scenario generation
3. Use case analysis
4. Ethnography

Interviewing

Formal or informal interviews with stakeholders are part of most RE processes.

Types of interview

- [1] Closed interviews based on pre-determined list of questions
- [2] Open interviews where various issues are explored with stakeholders.
- [3] Focused interviews, with clusters of stakeholders

Effective interviewing

- Be open-minded, avoid pre-conceived ideas about the requirements and are willing to listen to stakeholders.
- Prompt the interviewee to get discussions going using a springboard question, a requirements proposal, or by working together on a prototype system.

⊛ قبل عمل الـ interview نقوم بإرسال email للإتفاق على الموعد واقتراح خلال الـ email كذلك المواضيع التي سيتم طرحها

⇐ يفضل أيضاً عدم مقابلته في مكتبه لتجنب إنشغاله خلال العمل والذهاب إلى meeting room

⊛ Types of Interview

[1] Closed Interview: أن يكون الكلام والأسئلة من طرف واحد فقط

[2] Open Interview: أن يكون الكلام والأسئلة متبادلة بين الطرفين

[3] Focused Interview: اجتماع مع مجموعة من الـ actors وليس مع شخص واحد، وذلك لتجنب حدوث تعارض بالـ Requirements أو حل هذا التعارض إن وجد.

⇐ كذلك من الممكن أن يكون شخص لم يتفطن لنقطة معينة فيذكرها شخص

في من شاكل الـ Focused أنه من الممكن وجود مدير المجموعة بينهم، والذي قد لا يتيح للآخرين فرصة لأخذ راحتهم في الكلام.

Interviews

Meeting introductory protocol
- Ensure cultural introduction protocols are followed

First meeting
- Aim: to understand the business and its context with a clear aim to understand business processes and services.

Effective meetings:
- Ensure a chair is assigned at the beginning, to keep time-controlled progress
- Ensure an agenda is defined with clear objectives of the target outcome of the meeting
- Ensure a timescale is set for each agenda item and is kept/controlled by the chair
- Ensure clear actions and decisions (and who is responsible for and by when) are identified and reached by the end of the meeting
- Ensure the actions and decisions are summarised at the end of the meeting

سلايد إضافي من سلايدات الدكتور لقراءة عن طريقة تنظيم الاجتماعات للحصول أكبر فائدة ممكنة.

Scenarios

تخيل أمثلة عملية وواقعية عن السيناريو وطريقة عمله

Scenarios are real-life examples of how a system can be used.

They should include
- A description of the starting situation;
- A description of the normal flow of events;
- A description of what can go wrong;
- Information about other concurrent activities;
- A description of the state when the scenario finishes.

مثلاً في بناء سيناريو مكتبة نتخيل عملية إضافة كتاب والخطوات التي سنقوم بها خطوة بخطوة، كذلك هذا يساعدني في بناء Prototype لاحقاً

كذلك يمكن كتابة هذا الـ Scenario وتحويله إلى Use Case

مثال على Scenario كامل

Scenario for collecting medical history: Example

الوضع الأولي

- **Initial assumption**: The patient has seen a medical receptionist who has created a record in the system and collected the patient's personal information (name, address, age, etc.). A nurse is logged on to the system and is collecting medical history.
- **Normal**: The nurse searches for the patient by family name. If there is more than one patient with the same surname, the given name (first name in English) and date of birth are used to identify the patient.
- The nurse chooses the menu option to add medical history.
- The nurse then follows a series of prompts from the system to enter information about consultations elsewhere on mental health problems (free text input), existing medical conditions (nurse selects conditions from menu), medication currently taken (selected from menu), allergies (free text), and home life (form).

الوضع الطبيعي (بكل الحالات)

Scenario for collecting medical history: Example

في حال حدوث مشكلة ما يمكن أن تكون

What can go wrong?	Successful output?
Alternative: The patient's record does not exist or cannot be found. The nurse should create a new record and record personal information.	Yes
Alternative: Patient conditions or medication are not entered in the menu. The nurse should choose the 'other' option and enter free text describing the condition/medication.	Yes
Error: Patient cannot/will not provide information on medical history. The nurse should enter free text recording the patient's inability/unwillingness to provide information. The system should print the standard exclusion form stating that the lack of information may mean that treatment will be limited or delayed. This should be signed and handed to the patient.	No?
Other activities: Record may be consulted but not edited by other staff while information is being entered.	
System state on completion: User is logged on. The patient record including medical history is entered in the database, a record is added to the system log showing the start and end time of the session and the nurse involved.	

Ethnography (Observation)

في بعض الأحيان يتم تجميع الـ Reqs عن طريق الـ interviews ولكن يبقى هناك تفاصيل قد لا يعلم بعض الأمور، أو بإيصال بعض الأفكار من الـ client

A social scientist spends a considerable time observing and analysing how people actually work.

People do not have to explain or articulate their work.

Social and organisational factors of importance may be observed.

Ethnographic studies have shown that work is usually richer and more complex than suggested by simple system models.

⇐ هنا يتم الإلتقاء مع الـ client على سوء التقدم ومراقبة طريقة العمل فالمؤسسة المطلوب عمل سوفتوير لها حتى يتم فهم طريقة العمل بشكل صحيح ودون التأثير على سير العمل.

أحد مشاكل هذه الطريقة أنه من الممكن وجود هذا الشخص يؤثر على سير العمل وتدل من الممكن أن يكونوا الموظفين غير مرحبين بهذه الفترة.

Requirements validation

Concerned with demonstrating that the requirements define the system the customer really wants.

Requirements error costs are high so validation is very important

Fixing a **requirements error** after delivery may cost up to **100 times** the cost of fixing an implementation error.

بعد كل مرة من جمع الـ Reqs يجب عمل Validation لها (التأكد أنها تتبع المعايير المطلوبة في حال)

⇐ حل هذه المشاكل مبكراً كما ذكرنا سابقاً أقل تكلفة وقت وجهد ومراحل من اكتشافها لاحقاً وفي وقت متأخر.

Requirements Validation Check list

- **Validity Checks**: sometimes additional requirements are needed.
- **Consistency Checks**: no conflicts.
- **Completeness Checks**: nothing is missing.
- **Realism Checks**: can such requirements be done using current knowledge and technology?
- **Verifiability**: requirements should be verifiable

Requirements Validation Techniques

- **Requirements reviews**: systematic analysis of requirements by a team of reviewers.
- **Prototyping**: executable model of system is developed and presented to stakeholders.
- **Test-case generation**: requirements should be testable

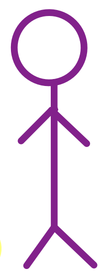
⊛ Use Case Diagram

يوجد في سلايدات (37, 38, 39) عن الـ use case diagram شرح.
ارجعو شوفوهن ☺

⇒ It's a Modified Modeling language

⇐ يعطينا نظرة عامة عن السيستم الذي سنقوم ببنائه بحيث يوضح الـ main fetchers الذي سيقدمها هذا النظام ومن سيقوم باستخدامه

※ يتم تمثيل كل من سيقوم باستعمال النظام بالشكل التالي:



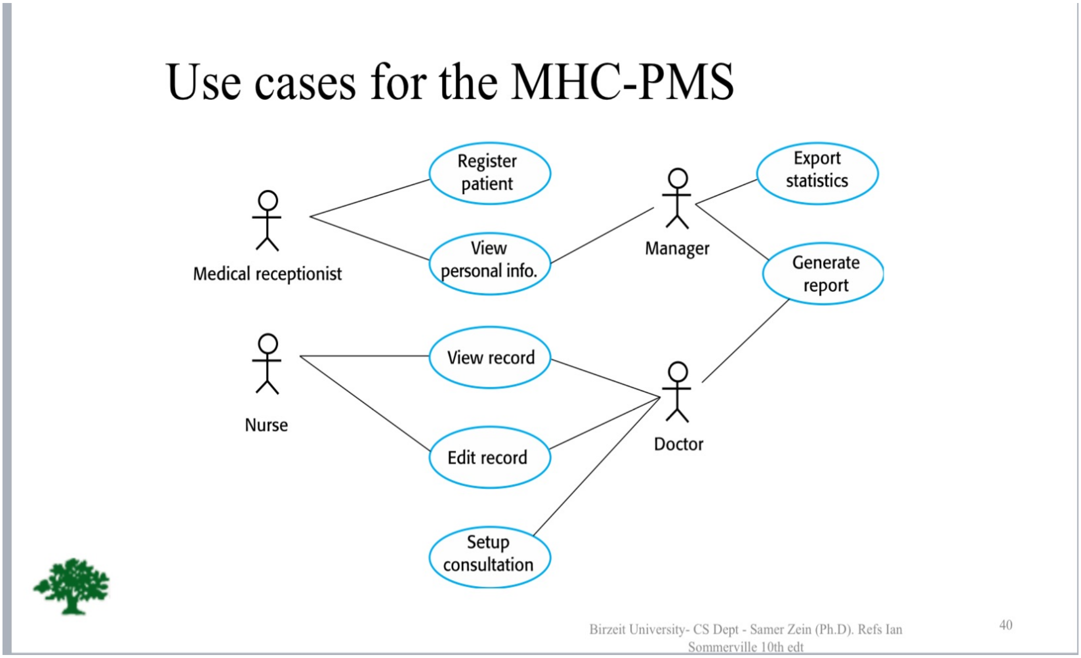
يسمى actor يمثل كلاً من الأشخاص الذين سيستعملون النظام وأيضاً يمثل أي سوفتوير سيستم آخر يتم استعماله داخل نظامنا

⇐ مثلاً: الطلاب في رتاج actor
⇐ المعلمين في الجامعة هم actor ولكن رؤساء الدوائر actor آخر لأن لهم صلاحيات تختلف عن المعلمين
⇐ مثلاً رتاج يتعامل مع سوفتوير (نظام) للمحاسبة بالتالي هذا النظام هو actor

※ تمثل النظام على شكل مستطيل

※ تمثل كل main fetcher (use case) بهذا الشكل

مثال من سلايدات دكتور سامر



* في ال case diagram التالية لدينا 4 actors وهم: Manager, Doctor, Nurse, Medical reception

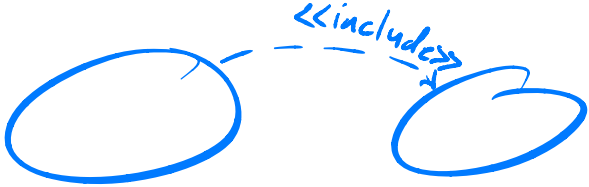
* لدينا 7 use cases لكل واحدة متصلة مع ال actors وذلك حسب صلاحية والمهام المطلوبة منه

ملاحظة: في كل سيستم يكون موجود update, delete, insert بدلاً من عمل use case لكل واحدة، نعمل use case نسميها Maintain أي Edit كما في المثال أعلاه.

Relationships

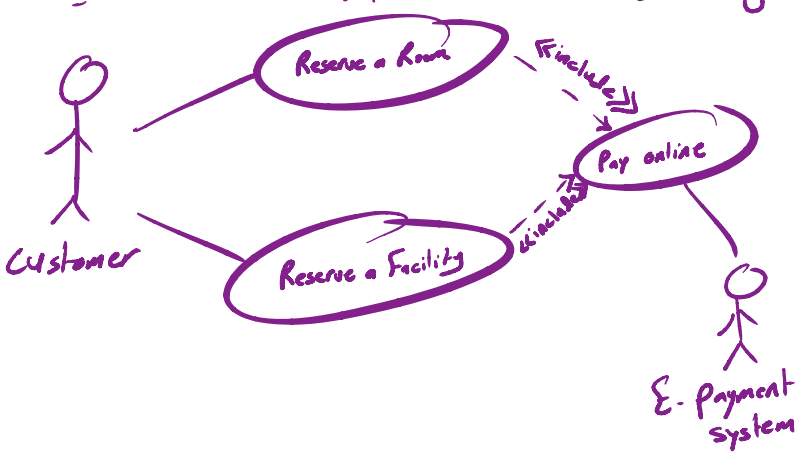
1. include relationship :

عندما يكون هنالك behavior مشترك بين أكثر من use case
إلا في حالة الـ login .



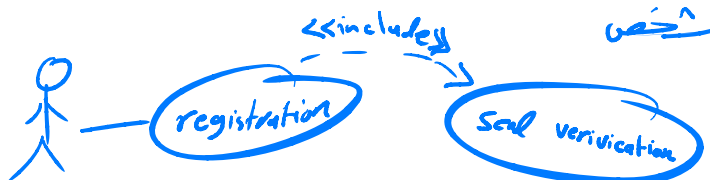
base case

مثل : رفع أول لاين لحجز غرفة أو شقة في نظام فندقي



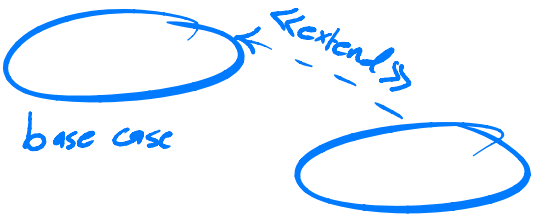
* استخدام آخر لـ include عندما يكون هناك شيء إجباري حصوله بعد عمل behavior معين

مثلاً : في نظام معين عند عمل تسجيل دخول يتم إرسال رسالة تحقق مباشرة إلى email الشخص



2 Extend Relationship

عندما يكون هناك احتمالين (نعم / لا) لعمل behavior معين
behavior آخر قبله



* طريقة تمثيلها :

ملاحظة: في العلاقة بين انتبه!!
عكس اتجاه السهم.

مثلاً : من الممكن أن أحجز كتاب بعد البحث عنه في نظام مكتبة

@Case Study (Dr. Saner)

BIRZEIT UNIVERSITY

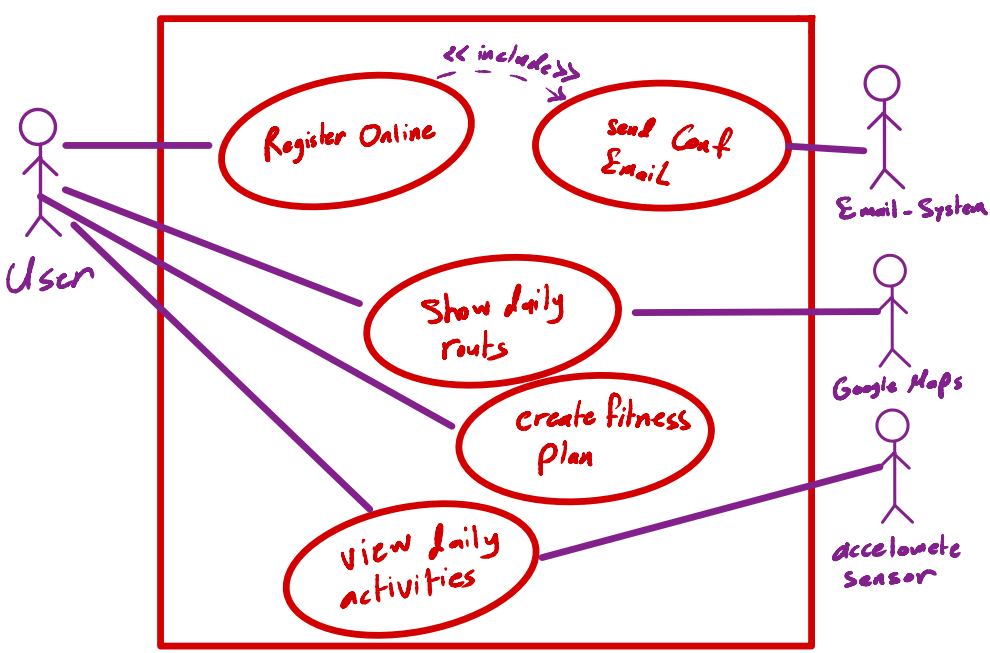
Computer Science Department

Software Engineering Course Comp433,

Read the following case study, and draw a UML Class diagram.

DialyFitPro is a mobile app for tracking user activity, calories, and overall fitness status. After installing the app, the user needs to create a profile (name, weight, gender, height, etc). After successful registration, the app sends an email notification. Later on app can monitor user daily activities and movements using accelometer sensor. The app can also show daily routes taken by user through integration with Google Maps. Further, the app allows the user to specify a fitness plan, for example, the user can specify that she/he wants to loose certain weight in specific time and so on.

Solution:



توتوريال من الدكتور سامر عن كتابة الـ Requirements

Software Engineering COMP433
Tutorial -4- Requirements Engineering

1. Consider the following two sets of Requirements.

Requirement Set-1:

-: The system shall provide a service for users (students) to register and create an account. ambiguous
-: Heads of Departments shall be able to submit queries about students in their departments. too general
-: The system shall adhere to the guidelines set by the ministry of higher education. → domain requirment
-: System Users should be able to view weather forecast when using the website → incorrect to off topic
-: The system shall allow only the registered users to use the services of the website.
-: During registration process, students shall be able to view registration requests for other students. → incorrect

Requirement Set-2:

R1.0: Registered users shall be able to submit a new application to study at the university during normal working hours, adhering to the education submission procedures. → incomplete

R2.0: The system shall create an application template and opens it in a new web page when users press new application button, from the "create new application" web page. The template should have the following data fields: Full Name, DoB, address, telephone numbers, Tawjihi Grade, and three Subjects to be studied in the order of preference. design details - wrong / The rest is fine

R3.0: The system shall fill the template automatically and detect and extract, intelligently, user details, as per the template, using advanced detection technologies, e.g. machine learning.. → Not feasabile

R4.0: The system shall check all the applications' data fields are complete and valid before submission and within a reasonable time. The system shall check number data fields contain only number values and text data fields contain at least some text values. → Not measurable / ambigous / incomplete

a) What do you consider the type of each of the above two sets of requirements? user requirements, system requirements (both)?
Why? Justify your Answer. Set-1 : User / Set-2 : System

and each one has different Feature ← High level language ✓ Depth into system details.

b) You are asked to validate the above two sets of above requirements on the following characteristics, identify the ones that do not validate. Don't talk about UI/UX from now
Correctness: set-1 and Set-2
Unambiguous: Set-2
Completeness: Neither
Consistency: Both ← لا متطلبين بعارضا بعض (تناقض)
Traceability: set-2
Realistic/Feasibility: Both
Measurability :
Why? Justify your Answer.

2. *If you were a project manger responsible to develop a system for improving the efficiency of a complex integrated manufacturing and ordering system. The environment is very complex and has multilevel hierarchies of end-users. The manufacturing part is complex, has large number of stakeholders, who work in a large area, highly fragmented and could not provide valuable input individually to influence the efficiency of the overall of the manufacturing process. The ordering system is simpler and has limited (or smaller number of) stakeholders, who could provide direct input.*

Which requirement discovery technique (or techniques) would you think would be most suitable or efficient for engineering the requirements? Justify your answer.