Gathering and Specifying Requirements

Part One of Two

Topics

- Project Specification
- Functional Requirements Artifacts
  - Glossary
  - User Scenarios
  - User Story (US)
  - Use Case (UC)
  - User Story (US) vs. Use Case (UC)
  - to be continued …

Project Specification

Some Project Excerpts

Project Specification [Example]

- Name: Many Labs – Clinical Analysis Management System

- Full and Updated Specification is available on Moodle

- Following slides provide:
  - A preview of some excerpts
  - Warning: Simulates some conversations with the software client (for demo)
  - **During the project development, the conversations should be real (via Moodle) and not simulated**

Project Specification Excerpts (1/4)

Many Labs is an English company that has a network of clinical analysis laboratories and that wants an application to manage the clinical analyses performed in its laboratories.

Many Labs is a company that operates in the English market, it has headquarters in London and has a network of clinical analysis laboratories in England where analysis of blood (samples are collected) are performed, as well as Covid-19 tests.

In England, Many Labs has exclusivity for Covid-19 tests throughout the territory, which means that no other company can perform this type of testing. All Many Labs clinical analysis laboratories perform clinical blood tests, and a subset of these laboratories also performs Covid-19 tests.

The set of Many Labs clinical analysis laboratories form a network that covers all England, and it is responsible for collecting samples and interacting with clients.

The samples collected by the network of laboratories are then sent to the chemical laboratory located in the company's headquarters and the chemical analysis are performed there.

Project Specification Excerpts (2/4)

Typically, the client arrives at one of the clinical analysis laboratories with a lab order prescribed by a doctor. Once there, a receptionist asks the client's citizen card number, the lab order (which contains the type of test and parameters to be measured), and registers in the application the test to be performed to that client.

Then, the client should wait until a medical lab technician calls him/her to collect the samples required to perform a given test.

All the tests (clinical blood tests and Covid-19 tests) performed by the network of laboratories are registered locally by the medical lab technicians who collect the samples. The samples are sent daily to the chemical laboratory where the chemical analyses are performed, and results obtained. When sampling (blood or swab) the medical lab technician records the samples in the system, associating the samples with the client/test, and identifying each sample with a barcode that is automatically generated using an external API.

Project Specification Excerpts (3/4)

At the company's headquarters, the clinical chemistry technologist receives the samples (delivered by a courier) and performs the chemical analysis, recording the results in the software application.

Each test is characterized by an internal code, an NHS code, a description that identifies the sample collection method, the date and time when the samples were collected, the date and time of the chemical analysis, the date and time of the diagnosis made by the specialist doctor, the date and time when the laboratory coordinator validated the test, and the test type.

Blood tests are frequently characterized by measuring several parameters which for presentation/reporting purposes are organized by categories. For example, parameters such as the number of Red Blood Cells (RBC), White Blood Cells (WBC) and Platelets (PLT) are usually presented under the blood count (Hemogram) category.

Project Specification Excerpts (4/4)

Covid tests are characterized by measuring a single parameter stating whether it is a positive or a negative result.

Despite being out of scope, the system should be developed having in mind the need to easily support other kinds of tests (e.g., urine). Regardless, such tests rely on measuring one or more parameters that can be grouped/organized by categories.
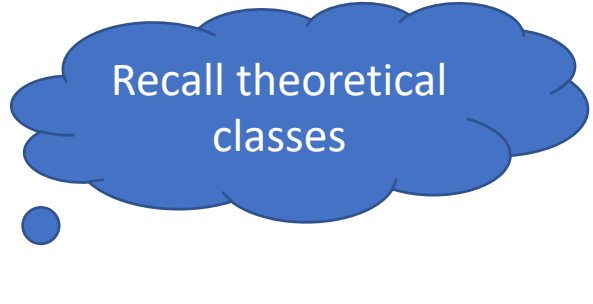
During system development, the team must: (i) adopt best practices for identifying requirements and for OO software analysis and design; (ii) adopt recognized coding standards (e.g., CamelCase); (iii) use Javadoc to generate useful documentation for Java code.

All those who wish to use the application must be authenticated.

Artifacts

Glossary

Glossary

- In its simplest form is just a list of terms and their meanings in the business scope
  - Interact with the client about the intended meaning of the terms
  - External sources of information might be also used
- Aims to facilitate communication between:
  - Members of the development team
  - Development team and client
- It can detail any element: an attribute of an object or terms used in other artifacts
  - Although it is just used for terms that are important in the project and whose ambiguity is intended to be reduced or eliminated

Glossary – Which are the important terms? The marked ones?

Typically, the client arrives at one of the <u>clinical analysis laboratories</u> with a <u>lab order</u> prescribed by a doctor. Once there, a <u>receptionist</u> asks the client's citizen card number, the lab order (which contains the <u>type of test</u> and <u>parameters</u> to be measured), and registers in the application the <u>test</u> to be performed to that client.

Then, the client should wait until a <u>medical lab technician</u> calls him/her to collect the samples required to perform a given test.
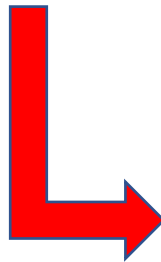
All the tests (clinical <u>blood tests</u> and <u>Covid-19 tests</u>) performed by the network of laboratories are registered locally by the medical lab technicians who collect the samples. The samples are sent daily to the <u>chemical laboratory</u> where the <u>chemical analyses</u> are performed, and results obtained. When sampling (blood or swab) the medical lab technician records the samples in the system, associating the samples with the client/test, and identifying each sample with a <u>barcode</u> that is automatically generated using an external <u>API</u>.

Glossary – Some Rules

- Glossary terms must be placed in alphabetical order
- A term must appear in its singular form in the glossary.
  - E.g., it should have "test" and not "tests"
- Abbreviations must also be included
  - E.g., "MLT" as acronym for "Medical Lab Technician"
- Terms with the same meaning must also be in the Glossary
  - In the description of the term itself; or
  - In another entry in the Glossary

Glossary – An example

Term or Expression (EN)	Termo ou Expressão(PT)	Description (Definition and Information)
Client	Cliente	It refers to the person requesting a clinical analysis test.
MLT	MLT	Acronym for Medical Lab Technician. It is a end-users' role.
Test	Exame	It refers to a clinical analysis test.
…	…	…

Optional. Used to clarify terminology when development team use more than one natural language (e.g., English, Portuguese).
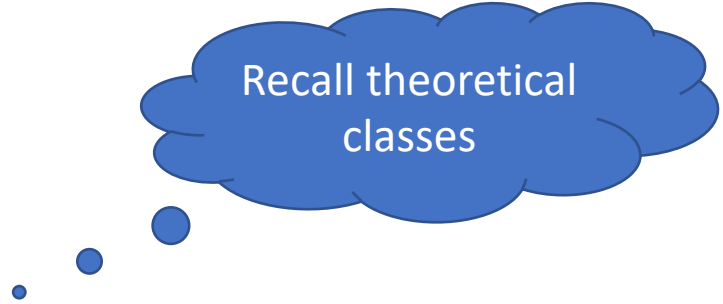
Glossary – Starting and Ending

- When to start?
  - It should start very early, but, as with other artifacts, it can and should be modified many times over the project development

- When is it done?
  - New terms may be added over time, but some definitions may also be refined, and new details may be added as they become known

Artifacts

User Scenarios

User Scenarios

- Aim at capturing the goals of the system from the application end-user's perspective
  - Focused on what the end-users need to do in their day-to-day job
  - Based on **Roles** and/or *personas*

- Actor
  - it is something with behavior, such as a person, a computer system or organization

User Scenarios are usually captured by

- **User Story**
  - It is a short description of a functionality told from the perspective of a user that is valuable to a us

and/or

- **Use Case**
  - It is a text story of how the system is used to achieve a certain business objective

Artifacts

User Story(US)

User Story

- A user story is a short, simple description of a feature told from the perspective of the person who desires the new capability, usually a user or customer of the system.

- User stories typically follow a simple template:
  - As a *<user role>*, I *<want to do something>*.
  - As a *<user role>*, I *<want to do something>*, so that *<benefit>*.

- Examples:
  - As a cashier, I want to process a sale of a customer.
  - As a cashier, I want to handle a product return made by a customer.

User Stories provided by the SW client

- US4: As a <u>receptionist</u> of the laboratory, I intend to <u>register a test</u> to be performed by a registered client.

- US5: As a <u>medical lab technician</u>, I want to <u>record the samples collected</u> in the scope of a given test.

- US9: As an <u>administrator</u>, I want to <u>specify a new type of test</u> and its collecting methods.

- US10: As an <u>administrator</u>, I want to <u>specify a new test parameter</u> and categorize it.

- US11: As an <u>administrator</u>, I want to <u>specify a new parameter category</u>.

Checking the User Stories

- Three distinct actors can be identified:
  - Receptionist
  - Medical Lab Technician
  - Administrator
- All US are missing 2 out of 3 C's
  - Card → Got it!
  - Conversation → To do
  - Confirmation (i.e., the acceptance criteria) → To do
- Lack of details regarding the underlying business processes

US11: As an administrator, I want to specify a new parameter category.

- What the development team already knows about it?
  - *"Blood tests are frequently characterized by measuring several parameters which for presentation/reporting purposes are organized by categories. For example, parameters such as the number of Red Blood Cells (RBC), White Blood Cells (RBC) and Platelets (PLT) are usually presented under the blood count (Hemogram) category."*
  - *"Regardless, such tests rely on measuring one or more parameters that can be grouped/organized by categories."*
- What the development team still needs to know? E.g.:
  - What are the data that characterize a parameter category?
  - What are the business rules applicable to such data?

US11: As an administrator, I want to specify a new parameter category. (cont.)

After some conversations with the software client, **suppose** that the development team got the following answers:

- What are the data that characterize a parameter category?
  - Simply consider a code, a description and an NHS identifier
- What are the business rules applicable to such data?
  - Code must be unique having 4 to 8 chars
  - Description cannot be empty and has, at maximum, 40 chars
  - NHS identifier is not mandatory

US11: As an administrator, I want to specify a new parameter category. (cont.)

After some conversations with the software client, **suppose** that the development team got the following answers:

- What are the data that characterize a parameter category?
  - Simply consider a code, a description and an NHS identifier
- What are the business rules applicable to such data?
  - Code must be unique having 4 to 8 chars
  - Description cannot be empty and has, at maximum, 40 chars
  - NHS identifier is not mandatory

ESOFT assumes a User Story is equivalent to a Use Case → Use Cases are richer than User Stories in details → Lets create the corresponding Use Case

Artifacts

Use Case (UC)

Use Case Formats

- **Brief**
  - It assumes a high-level point of view
  - Has a Title (ID and name of UC)
  - A paragraph describing the main success scenario

- The description of the use case does not include details about the system interface.
- Do NOT write something like "User writes in the text box and clicks the OK button"!

- **Fully-dressed, describing:**
  - Primary actor
  - Stakeholders and Interests
  - Preconditions
  - Success Guarantee (or Postconditions)
  - Main Success Scenario (or Basic Flow)
  - Extensions (or Alternative Flows)
  - Special Requirements
  - Technology and Data Variations List
  - Frequency of Occurrence
  - Open Issues

Brief Format

- **Brief format** – it assumes a high-level point of view
  - A paragraph describing the main success scenario

UC 11 Create a Parameter Category

The administrator starts the definition of a new parameter category. The system requests the required data (i.e., code, description, and NHS id). The administrator types the requested data. The system validates and presents the data to the administrator, asking her/him to confirm. The administrator confirms. The system records the data and informs the administrator of the operation's success.

Brief Format

- **Brief format** – it assumes a high-level point of view
    - A paragraph describing the main success scenario

Id and Name
of the UC

UC 11 Create a Parameter Category

The administrator starts the definition of a new parameter category. The system requests the required data (i.e., code, description, and NHS id). The administrator types the requested data. The system validates and presents the data to the administrator, asking her/him to confirm. The administrator confirms. The system records the data and informs the administrator of the operation's success.

Brief Format

- **Brief format** – it assumes a high-level point of view
  - A paragraph describing the main success scenario

Id and Name
of the UC

UC Successful
Scenario

UC 11 Create a Parameter Category

The administrator starts the definition of a new parameter category. The system requests the required data (i.e., code, description, and NHS id). The administrator types the requested data. The system validates and presents the data to the administrator, asking her/him to confirm. The administrator confirms. The system records the data and informs the administrator of the operation's success.

Brief Format

- **Brief format** – it assumes a high-level point of view
  - A paragraph describing the main success scenario

Id and Name
of the UC

UC 11 Create a Parameter Category

The administrator starts the definition of a new parameter category. The system requests the required data (i.e., code, description, and NHS id). The administrator types the requested data. The system validates and presents the data to the administrator, asking her/him to confirm. The administrator confirms. The system records the data and informs the administrator of the operation's success.

UC Successful
Scenario

The description of the use case does not include details about the system interface.

Brief Format

- **Brief format** – it assumes a high-level point of view
  - A paragraph describing the main success scenario

Id and Name of the UC

UC 11 Create a Parameter Category

UC Successful Scenario

The administrator starts the definition of a new parameter category. The system requests the required data (i.e., code, description, and NHS id). The administrator types the requested data. The system validates and presents the data to the administrator, asking her/him to confirm. The administrator confirms. The system records the data and informs the administrator of the operation's success.

The description of the use case does not include details about the system interface.

Do NOT write something like "User writes in the text box and clicks the OK button"!

Fully-Dressed Format for UC11 (1/5)

- Several sections:
  - Primary actor
  - Stakeholders and Interests
  - Preconditions
  - Success Guarantee (or Postconditions)
  - Main Success Scenario (or Basic Flow)
  - Extensions (or Alternative Flows)
  - Special Requirements
  - Technology and Data Variations List
  - Frequency of Occurrence
  - Open Issues

Fully-Dressed Format for UC11 (2/5)

- **Primary actor:**
  - Administrator
- **Stakeholders and Interests:**
  - Administrator : Wants accurate, fast/easy entry as test parameters cannot be defined without a parameter category
  - Many Labs: Wants to provide clients with a clear and well-organized/presented test report
  - Client' Doctor: Wants a test report compliant with NHS regulations
- **Preconditions:**
  - Administrator is identified and authenticated.
- **Success Guarantee (or Postconditions):**
  - Parameter category is saved.

Fully-Dressed Format for UC11 (3/5)

- **Main Success Scenario (or Basic Flow):**
  1. The administrator starts the definition of a new parameter category.
  2. The system requests the required data (i.e., code, description, and NHS id).
  3. The administrator types the requested data.
  4. The system validates and presents the data to the administrator, asking her/him to confirm.
  5. The administrator confirms.
  6. The system records the data and informs the administrator of the operation's success.

Fully-Dressed Format for UC11 (4/5)

- **Extensions (or Alternative Flows)**
    - ***a**. The administrator asks to cancel the creation of the parameter category.
        - 1. The use case ends.
    - **4a.** Lacking mandatory data.
        - 1. The system informs which data is lacking.
        - 2. The system allows the introduction of the lacking data (step 3).
            - 2a. The administrator does not change the data.
            - The use case ends.
    - **4b.** The system detects that the introduced code already exist in the system.
        - 1. The system alerts the administrator.
        - 2. The system allows administrator to change the code (step 3)
            - 2a. The administrator does not change the data.
            - The use case ends.

    - …

Fully-Dressed Format for UC11 (5/5)

- **Special Requirements:**
  - Code must be unique having 4 to 8 chars
  - Description cannot be empty and has, at maximum, 40 chars
  - NHS identifier is not mandatory
- **Technology and Data Variations List:**
  - (not identified)
- **Frequency of Occurrence:**
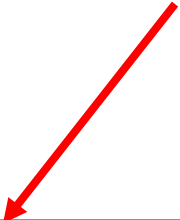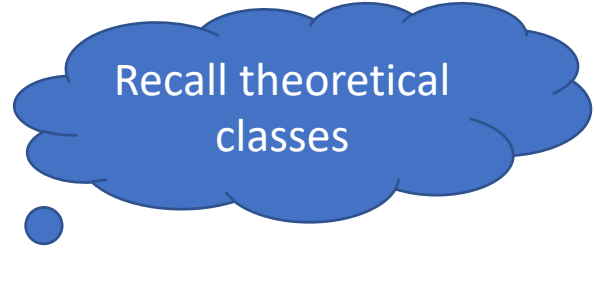 - Very occasionally.
- **Open Issues:**
  - Is there any relationship between parameter categories and the types of test?
  - What is the purpose of the NHS identifier?
  - NHS identifier follows any format/structure?

Fully-Dressed Format for UC11 (5/5)

- **Special Requirements:**
  - Code must be unique having 4 to 8 chars
  - Description cannot be empty and has, at maximum, 40 chars
  - NHS identifier is not mandatory
- **Technology and Data Variations List:**
  - (not identified)
- **Frequency of Occurrence:**
  - Very occasionally.
- **Open Issues:**
  - Is there any relationship between parameter categories and the types of test?
  - What is the purpose of the NHS identifier?
  - NHS identifier follows any format/structure?

Fosters more conversations with SW client
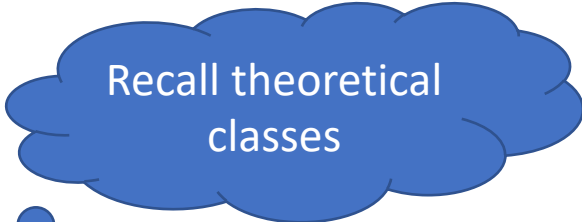
Artifacts

User Story vs. Use Case

Similarities and Differences (1/3)

Recall theoretical classes

- Both are focused on achieving a particular goal for a user
  - US is more target to capture who, what and why of a functionality
  - UC is more target to capture who, what and how the system plays (the flow)
  - US is more about user needs while UC is more about behavior to meet needs

- Degree of detail
  - US are normally, and purposely, vaguer → Lack of details
  - US are meant to promote elicit conversations with the client
  - UC shows how user and system interact with each other → Richer in details
  - UC are a more structured approach demanding more up-front details
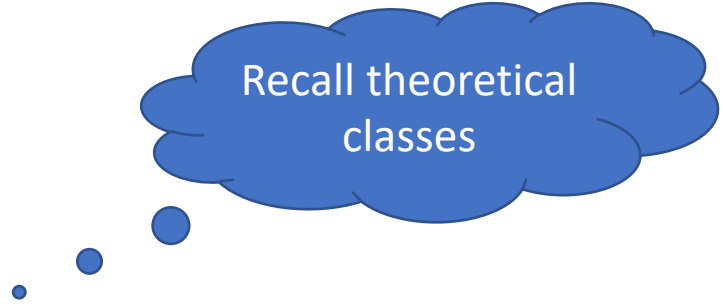
Similarities and Differences (2/3)

Recall theoretical classes

- Size
  - US/UC are sized to deliver business value
  - US are more suitable for sprints/iterations
  - UC are commonly split across multiple sprints/iterations

- Communication
  - UC emphasize written communication, which is often very imprecise
    - Hard to elaborate → More verbose → Who really reads it?
  - US emphasize verbal communication, becoming easier to clarify something
    - Easier/faster to elaborate → Easier to read

Similarities and Differences (3/3)

- Longevity
  - US are intended to outlive the sprint in which they are added to the software
    - However, it is possible to archive US cards
  - UC are often permanent artifacts that continue to exist as long as the product is under active development or maintenance

- Useful for planning
  - US are typically smaller and easier to estimate difficulty and time-consuming
  - UC are generally too large and thus are harder to estimate

Summary

- After describing some artifacts of the requirements engineering, that focus mostly on Functional Requirements
  - Glossary
  - User Scenarios
  - User Stories (US)
  - Use Cases (UC)
  - User Stories (US) vs. Use Cases (UC)
- On the next session we'll focus on other artifacts for these requirements and also Non-Functional Requirements

References & Bibliography

- Larman, Craig; Applying UML and Patterns; Prentice Hall (3rd ed.); ISBN 978-0131489066

- Eeles, P. (2005). Capturing architectural requirements. Available on http://www.ibm.com/developerworks/rational/library/4706.html.

- Abran, A., Bourque, P., Dupuis, R., & Moore, J. W. (2001). Guide to the software engineering body of knowledge-SWEBOK. IEEE Press.

- Mike Cohn (2004). Advantages of User Stories for Requirements. Available on https://www.mountaingoatsoftware.com/articles/advantages-of-user-stories-for-requirements

- Andrew, Stellman(2009). Requirements 101: User Stories vs. Use Cases. Available on https://www.stellman-greene.com/2009/05/03/requirements-101-user-stories-vs-use-cases/