INSTITUTO SUPERIOR DE TECNOLOGIAS AVANÇADAS DE LISBOA

Hands On – Lab. 4.1

Projeto Ágil (Scrum)

- **Reunião do cliente**
- **Tarefas de Elicitação e Análise**
- **Entrevistando as partes interessadas**

16 de novembro de 2022

Hands-On – Laboratório. 4.1

Projeto Ágil (Scrum)

16 de novembro de 2022

- **Metas**
  - Você, como proprietário do produto**(PO)**estão se reunindo com o**CryptoIsCool**gerente de negócios para reunir requisitos para os produtos e serviços que você vai entregar
  - Você, como um**Equipe**, iniciará o projeto executando as principais tarefas de engenharia de requisitos e gerenciará o projeto seguindo a abordagem Scrum

- **Tarefa**
  - Como um**Equipe**, consulte a folha de dicas do Scrum compartilhada no Classroom
  - Como um**Equipe**, discutir como aplicar a metodologia para elicitação e análise de requisitos
  - Como um**PO**, colete o máximo de informações possível
  - Como um**PO**, avaliar o impacto comercial que cada funcionalidade pode ter no sistema global
  - Como um**Equipe**, comece a pensar nos detalhes da especificação
  - Como um**Equipe**, comece a construir o seu**Documento de Requisitos**modelo

- **Especificação de linguagem natural (dois requisitos)**
  - Este método pode ser muito vago. Depende da habilidade do escritor em colocar os requisitos em palavras
  - **Tem pouco formalismo**

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.*)

- **Alternativas à especificação de linguagem natural**
  - Às vezes, esses métodos podem ser combinados para ilustrar mais detalhadamente alguns requisitos do cliente
  - **É mais trabalho intensivo**

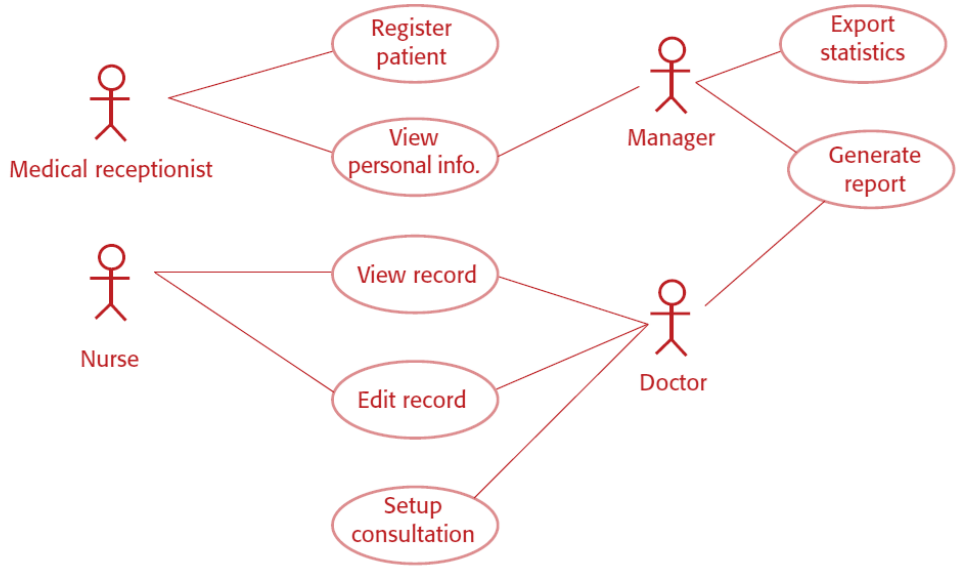
Notation	Description
Natural language sentences	The requirements are written using numbered sentences in natural language. Each sentence should express one requirement.
Structured natural language	The requirements are written in natural language on a standard form or template. Each field provides information about an aspect of the requirement.
Graphical notations	Graphical models, supplemented by text annotations, are used to define the functional requirements for the system. UML (unified modeling language) use case and sequence diagrams are commonly used.
Mathematical specifications	These notations are based on mathematical concepts such as finite-state machines or sets. Although these unambiguous specifications can reduce the ambiguity in a requirements document, most customers don't understand a formal specification. They cannot check that it represents what they want, and they are reluctant to accept it as a system contract. (I discuss this approach, in Chapter 10, which covers system dependability.)

Hands-On – Laboratório. 4.1

Projeto Ágil (Scrum)

16 de novembro de 2022

- **Casos de uso**
  - Os casos de uso são uma maneira de descrever as interações entre os usuários e um sistema usando um modelo gráfico e um texto estruturado
  - **É relativamente fácil de realizar**



- **Especificação estruturada (um requisito)**
  - Este método é**mais detalhado, portanto, mais preciso**sobre os requisitos do cliente.
  - **É mais trabalho intensivo**

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. (see Figure 4.14)
Requires	Two previous readings so that the rate of change of sugar level can be computed.
Precondition	The insulin reservoir contains at least the maximum allowed single dose of insulin.
Postcondition	r0 is replaced by r1 then r1 is replaced by r2.
Side effects	None.

Hands-On – Laboratório. 4.1

Projeto Ágil (Scrum)

16 de novembro de 2022

- **Documento de Requisitos de Software**
  - Depende do tipo de software que está sendo desenvolvido
  - Para um sistema de engenharia complexo que inclui hardware e software desenvolvido por diferentes empresas, o documento de requisitos provavelmente será longo e detalhado
  - Para um produto de software interno, muitos capítulos detalhados serão deixados de fora. O foco estará na definição dos requisitos do usuário e requisitos de sistema não funcionais de alto nível

Chapter	Description
Preface	This defines 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 describes 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 defines 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 nonfunctional 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 presents 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.
System requirements specification	This describes the functional and nonfunctional requirements in more detail. If necessary, further detail may also be added to the nonfunctional requirements. Interfaces to other systems may be defined.
System models	This chapter includes graphical system models showing the relationships between the system components and the system and its environment. Examples of possible models are object models, data-flow models, or semantic data models.
System evolution	This describes 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 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.