Aline Koftikian

Project Deliverable 1 —Section 7.2 – Artifacts

1.2 Project Description

The project description is an introduction to the entire project. It’s a brief that allows the reader and the user of the system to understand the goal and purpose of the project. This document must be presented as an opening to the rest of the manual, and must include the idea, objective, background, approach, and result of the entire project.

* + 1. Functional Requirements

Requirements are the tasks that are necessary for the completion of the system. They are the components of the system that must be documented, measured and tested in order to ensure the ideal conclusion to the project in question. The requirements that need to be specified for this system are functional and non-functional requirements. The functional requirements are the tasks that define what the system is trying to accomplish. These are the elementary actions that constitute the main goal of the system. They include the user viewing weekly schedule, adding classes, dropping classes, swapping, etc.

Each individual requirement must be presented as a use case. A use case is a description of the behavior of a system that explains the process and results obtained through interactions of the user with the system. The functional requirements must be known before writing use cases, because they are the details that compose the main actions and tasks of the system. The use cases are present in both list form and diagram form.

* List Use Cases: The list-based use cases are full formal documents that list all use cases of the system in detail, with each property of the use case explained in full detail. These include the name of the use case, the general description, the actors, preconditions, postconditions, etc.
* Diagram Use Cases: Use case diagrams are used to depict a visual representation of the use cases listed. The actors, which typically include the user or an external entity, are depicted using stick figures. The use cases themselves are drawn as circles with a name that describes the title of the action that can be initiated by the actors. There are lines connected from users to the use cases called connections, which describe the actor’s interactions with those use cases.

1.3.2. Domain Model

The Domain Model is the model that describes all core concepts and an overview of the system at hand. Its main purpose is to list the objects of interest that form the system, and presents their attributes, constraints, and relationships between them. The model is designed in UML format, with arrows describing relationships from one object to another, and ensuring that there are multiplicity values at both ends of the arrow denoting multiplicity factors of the relationship. The finalized domain model will come with a description that will provide a more detailed explanation of the model as well as each object present in the model.

1.3.3. Constraints and Qualities

The constraints and qualities segment describes the main standards of quality and non-functional requirements that are expected to be met by the system. The non-functional requirements are the general properties of the system. They are the constraints and qualities that will make up the system when in use. These include response times, maintenance times, security, etc. It’s important that the constraints and qualities are as specific as possible, which means they must have concrete metric specifications wherever possible.

1.4.1 Human Resources

The human resources of a project is the group of individuals who compose the main workforce of the project. They are the source of labor of a project, and in the case of large projects, are typically divided into smaller groups for more organized work.

1.4.2. Technical Resources

The technical resources of a project are the computer resources that will be used to complete the project. This includes the computer programs, software, libraries, and websites that are available for use.

1.5. Scoping

The scoping section describes all elements, features, qualities, and goals that have been scoped out from the end result. Each scoped element must come with a description of what it is, in addition to a paragraph that describes why it was chosen to be scoped out. These scoped elements have been typically scoped out in the early phases of a project, where the requirements are being planned and discussed in between team members.

1.6.1. Architecture

The architecture of a project is the high-level system architecture that is planned for the complete project. It must be presented with a UML class diagram that describes the details and relationships between different components of the system, along with a detailed description of the mindset behind the architecture (also known as a design rationale). In addition, each module of the architecture must be accompanied with its own design rationale and responsibilities.

1.6.2. Technologies in Use

This segment must provide a technical description of all the technologies that will be used to build the project. These include programming languages, libraries, servers, databases, IDE and compilers, etc. For each technology listed, there must be a short description that explains the rational for the choice and use of the specific tool.

1.7.1 Activities

The activities section is a section that will list a step-by-step process of how the separate artifacts of the project will be produced in order to reach the completion of the project. Each activity must be presented with a clear purpose, description, and must produce at least one artifact in consequence.

1.7.2. Artifacts

An artifact is a by-product that is produced as a consequence of the development of a software project. The artifacts segment will list each document that will be produced in the project. Each artifact listed must come with a description that provides details about the role of that specific document in the completion of the project.

1.7.3. Project Estimates

The project estimates section is for providing a realistic estimate for the cost, as well as the schedule for the length of the project. This is done by summing the evaluated cost of each artifact that’s produced throughout the conception of the project. Each cost must be backed up by a basis of the estimation.

1.7.4. Activities Assignments

The activities assignments are the specific assignments of tasks and activities that are made to each team member based on the members’ skills, requirements, capacities, limitations and schedules.

1.7.5. Schedule

The schedule section must present all the project’s deadlines, milestones, sprints, iterations and other target dates in the form of a diagram or table. A Gantt chart is the ideal way to present a schedule as it outlines the dates all important events from start to finish in a clean and organized fashion.

1.7.6. Risk

The risk section will present all elements of risk of the project, including both early factors and late factors of risk. Each risk must be accompanied by a description and a consequence should the risk occur. They may be presented in a table or list form.

1.8. Prototyping

The prototype section is a description of what work has been completed so far in terms of developing a prototype of the project. A prototype is an early model that is built before the final release of a system or device. The goal of this section is to decide whether the tools and technologies chosen for the completion of the project are appropriate for the final release.

2.2. Introduction to Part 2

The second introduction will introduce the user to the second large deliverable of the project. It is simply a brief description of what information is included in the entire document.

2.3.1. Architecture Diagram

The architecture diagram section is an improved and more detailed version of the architecture diagram presented in the previous deliverable. The system must be divided into at least two systems for this diagram. It must include a high-level description of the architecture of the system in addition to descriptions for each separate module included in the system, accompanied with the visual diagram.

2.3.2. Subsystem Interfaces Specification

The subsystem interfaces specification section is the section that will describe the specifications of the interactions between the software interfaces of the components of the project. These are most likely presented as the function calls and message passing that occur. They must be accompanied with descriptions of the parameters passed of the function calls, including possible valid and invalid ranges of values.

2.4.1. Detailed Design Diagram

The detailed design diagram is a diagram that describes the internal structure of the subsystems depicted in the project. This must be presented as a UML diagram, and must include descriptions that explain the rationale of the designs, in addition to any extra information that is not present in the diagram.

2.4.2. Unit Description

The unit description section is the section that lists each class that’s present in the subsystem and accompanies them with a description of the purpose and function of the class. In addition, they are accompanied by all attributes and methods of the class. They may also be notes and reminders that are there to give any extra information for the programmers who are implementing the classes.

2.5. Dynamic Design Scenario

The dynamic design scenario is a list of two vital and substantial use cases of the system. These use cases must have at least 3 system operations, and the scenarios listed must include system sequence diagrams and operational contracts.

2.6. Estimation

The estimation section is a list of all modules that have been identified so far, accompanied by a new estimate for each one. The revision must also include the newly estimated costs of integration, testing, and additional documentation.

2.7. Rapid Prototyping and Risk

The rapid prototyping and risk section lists the prototypes of the system completed so far. The list must be accompanied by all information pertaining to the models, including classes, modules, drivers, frameworks, database, and other technologies used. Each prototype has a description of the effect of its model, in addition to the effects on the risks previously mentioned, the effects on the estimate, and the changes made in the scope because of them.

3.2. Introduction to Part 3

The third introduction will introduce the user to the third large deliverable of the project. It is simply a brief description of what information is included in the entire document.

3.3.1.1. Tested Items

The tested items section is a list of all items and components that have been tested so far, including the test cases that were applied during the process. Each tested item is accompanied with a description that explains the test case along with why the testing was necessary.

3.3.1.2. Untested Items

The untested items section is a list of all items and components that have yet to be tested, but are deemed necessary to be tested. They are accompanied with an explanation that highlights why it should be tested, along with how it would be tested.

3.3.2.1. Unit Testing

The unit testing section includes two mid-level testable units of the system, which could be modules, classes, or subsystems. Each test unit is accompanied by a list of test cases and the code for the drivers used. In addition, an explanation of the techniques and description of the test are provided, along with the results of the testing.

3.3.2.2. Requirements Testing

The requirements testing section is a list of test cases that accompany each tested requirement of the system. This includes a realistic scenario of the system usage during the test, along with the expected reaction of the system under the conditions of the test.

3.3.2.3. Stress Testing

The stress testing section is a description of the potential extreme situations of system usage. This is necessary to view the behavior of the system in unlikely but extreme conditions to ensure stability of the system as a whole. A design of tests is provided along with the description that verifies how the system performs under those extreme conditions.

3.3.2.4. Security Testing

The security testing section is a description of the tests that have been undergone to ensure maximum security of the system. This includes SQL injection attack resistance to avoid breach of user information.

3.4.1. Installation Manual

The installation manual is a report that is aimed for a competent administrator. It is a precise description of the steps to take to install (i.e. from a compressed file or disk) and execute the software. The reader is expected to install the system simply using the code and this manual provided.

3.4.2. User’s Manual

The installation manual is a document that is aimed for the general user. It is a precise description of how to use the system, and it provides guidelines for all usable components present. All features of the system must be included and described in this document.

3.5. Final Cost Estimate

The final cost estimate is a table that neatly lists all components that are and have been present in the completion of the project. These estimates include documentation, design, implementation, data acquisition, and testing costs.