**7.1 Activities and Tasks Assignment**

|  |  |  |  |
| --- | --- | --- | --- |
| Activity | Purpose/Description | Artifact(s) produced | Assigned member(s) |
| 7.1.1  Form and divide team | Forming a team allows the project to start. Dividing the team into a coding and documentation sub-teams allows members to focus on particular tasks. Such division requires the evaluation of the strengths and weaknesses of all members to determine their role. | 1.4.1 Human resources | Claudia Della Serra  Dimitri Topaloglou  Philip Lim  Ryan Lee  Daniel Di Corpo  Erin Benderoff  Aline Koftikian  Ideawin-Bunthy Koun  Kevin Yasmine  Marc-Andre Leclair  Lori Dalkin  Bryce Drewery-Schoeler |
| 7.1.2  Complete the system overview | The system overview consists of finding a name for the system, determining the main purposes and elements involved in it, and drawing a domain model of the system along with the description of each entity found in the domain model. | Deliverable 0  1.3.2 Domain model | Claudia Della Serra  Dimitri Topaloglou  Philip Lim  Ryan Lee  Daniel Di Corpo  Erin Benderoff  Aline Koftikian  Ideawin-Bunthy Koun  Kevin Yasmine  Marc-Andre Leclair  Lori Dalkin  Bryce Drewery-Schoeler |
| 7.1.3  Write a project description | The project description introduces the entire document. It informs the reader about its goals, the information to be found in it, as well as the purposes and objectives of the project. | 1.2 Project description | Ryan Lee |
| 7.1.4  Determine the functional requirements and draw use case diagrams accordingly | Determine the requirements that the system should have, including those that may later be scoped out, their relative importance and relative difficulty. Define the actions that must take place in the software and illustrate each requirement by the mean of a use case diagram. | 1.3.1 Functional requirements | Dimitri Topaloglou  Bryce Drewery-Schoeler Ideawin-Bunthy Koun  Erin Benderoff  Lori Dalkin  Marc Leclair |
| 7.1.5  Review and correct domain model | Make any necessary changes according to the feedbacks of Deliverable 0. | 1.3.2 Domain model | Aline Koftikian |
| 7.1.6  Find and describe constraints and qualities | Describe any design constraints, qualities and non-functional requirements that the system should meet. | 1.3.3 Constraints and qualities | Dimitri Topaloglou |
| 7.1.7  Review and format human resources document | After the profile of each team member has been determined, the document should be revised and formatted correctly. | 1.4.1 Human resources | Lori Dalkin |
| 7.1.8  Determine technical resources | List the computer resources and tools available to complete the project. | 1.4.2 Technical resources | Dimitri Topaloglou |
| 7.1.9  Reduce the amount of features to be realistic according to time and resources | Outline the scope of the software. List all features, goals and qualities that are scoped out, and provide an explanation for each element that is scoped out. | 1.5 Scoping | Kevin Yasmine |
| 7.1.10  Choose the architecture of the system | Explain the high-level architecture of the system, and give a design rationale. A diagram should accompany the explanation. | 1.6.1 Architecture | Daniel Di Corpo |
| 7.1.11  Decide which technologies are relevant to the project development | List technologies that will be used in the project and give a rationale for each of them. | 1.6.2 Technologies in use | Dimitri Topaloglou |
| 7.1.12  List and describe the main activities involved in the project | List all activities that produce at least one artifact and include a description for each activity. | 1.7.1  Activities | Philip Lim |
| 7.1.13  List all artifacts | List and describe all artifacts that need to be produced for the project. | 1.7.2 Artifacts | Aline Koftikian |
| 7.1.14  Estimate the cost and time of production | Estimate the cost and time of production of each artifact and sum them up. | 1.7.3 Project estimates | Daniel Di Corpo |
| 7.1.15  Delegate activities to different team members | Assign each activities of the project to a team member or to a group of team members. | 1.7.4 Activities assignments | Philip Lim |
| 7.1.16  Produce a schedule for the project | Create a schedule/timeline of all activities for the project using a Gantt chart. Start dates, due dates, names of activities, and participants are all information to be included in the schedule. | 1.7.5 Schedule | Ryan Lee |
| 7.1.17  Determine the risks of the project | Outline and explain the risks associated and presented by this project. | 1.7.6 Risk | Dimitri Topaloglou |
| 7.1.18  Produce a prototype report | Describe the work done in the development of the prototype. Explain how the chosen technologies are appropriate, and how the team members are comfortable using such technologies. | 1.8 Prototyping |  |
| 7.1.19  Ensure professionalism of the document | Assemble all parts of deliverable 1. Ensure the document is professional looking and well-organized. | Deliverable 1 | Claudia Della Serra |
| 7.1.20  Update the introduction for Deliverable 2 | Update the introduction to give the reader an overview of the content found in Deliverable 2: Architecture and Design | 2.2 Introduction to part 2 |  |
| 7.1.21  Produce an architecture diagram | Produce a 4+1 Architectural View. Include a rationale for the design. Explain any differences between the old design and this one. | 2.3.1 Architecture Diagram |  |
| 7.1.22  Specify the interactions between the components of the software | Describe the interactions between the components of the software through their interfaces. Include the function calls, the description of the parameters and the range of accepted values of those parameters. | 2.3.2 Subsystem interfaces specifications |  |
| 7.1.23  Illustrate the internal structure of the system | Provide a graphical representation of the structure of each subsystem by means of a UML class diagram. A description of each class should be included. | 2.4.1 Detailed design diagram |  |
| 7.1.24  Describe each class in the subsystem | Provide the programmers with the descriptions of each class in the UML diagram along with any necessary detailed relevant to the development. | 2.4.2 Unit description |  |
| 7.1.25  Produce dynamic design scenarios | Draw the dynamic design of two use cases. | 2.5 Dynamic design scenario |  |
| 7.1.26  Update project estimates | Update the cost of the project for each module and include the cost of integration, testing and documentation. | 2.6 Estimation |  |
| 7.1.27  Produce a report about the prototype and risk | List and describe all elements that have been implemented and describe the effect that those implementations had on the design decisions, risks, estimate and scope. | 2.7 Rapid prototyping and risk |  |
| 7.1.28  Ensure professionalism of the document | Assemble all parts of deliverable 2. Ensure the document is professional looking and well-organized. | Deliverable 2 |  |
| 7.1.29  Update introduction | Update the introduction so that it includes an overview of the content presented in deliverable 3. | 3.2 Introduction |  |
| 7.1.30  List all tested items | Create a list of all items that have been tested, the test cases that were used, and a rationale for the test. | 3.3.1.1  Tested items |  |
| 7.1.31  List all untested items | Create a list of items that are to be tested, an explanation of why those items should be tested, and how they could be tested. | 3.3.1.2 Untested items |  |
| 7.1.32  Unit testing report | Describe the test cases, the technique used, the code used and the result of the testing for two units. | 3.3.2.1 Unit testing |  |
| 7.1.33  Requirements testing report | Provide a list of test cases for all tested requirements by means of scenario of system usage and system reaction. | 3.3.2.2 Requirements testing |  |
| 7.1.34  Stress testing report | Describe situations of extreme system usage, the tests designed to evaluate the performance of the system under such condition, and the test results. | 3.3.2.3  Stress testing |  |
| 7.1.35  Security testing report | Describe tests performed to protect the system against security threats such as SQL injections and automated tools. | 3.3.2.4 Security testing |  |
| 7.1.36  Installation manual redaction | Guide any administrators to install and execute the software. | 3.4.1 Installation manual |  |
| 7.1.37  User’s manual redaction | Guide any users to use the system with all of its features. | 3.4.2 User’s manual |  |
| 7.1.38  Update cost estimate | Update the table of costs with all components of all phases. | 3.5 Final cost estimate |  |
| 7.1.39  Ensure professionalism of the document | Assemble all parts of deliverable 2. Ensure the document is professional looking and well-organized. | Deliverable 3 |  |
| 7.1.40  Correction of various part of the deliverables | Make all necessary changes in the deliverables for the final report. | Corrected deliverables |  |
| 7.1.41  Assemble all deliverables | Put all corrected/updated deliverables together for the final report. Ensure professionalism of the document. | Final report |  |