# ABC’s Inventory Management System

# Elaboration Phase Project Status Assessment

# Introduction

ABC’s Inventory Management System is a project aimed to develop a new digital system for the ABC group to handle their stock management and distribution. This document aims to provide the status of the project in the Elaboration Phase. This document will describe the expectations, achievements and finally review and evaluate the project in the Elaboration Phase. All in all, this document will showcase where the project stands so far.

# Expectations

The main goal of the Elaboration phase is to validate the chosen architecture in the Inception phase and address the major risks of the project. The following are the list of expectations at the end of the Elaboration phase:

* Establish a detailed architecture document along with the non-functional requirement document.
* Establish a detailed domain model and design class diagram.
* Establish Full Use Case Descriptions for all use cases.
* Establish Sequence Diagram for all major use cases.
* Establish a prototype of the application involving a major use case as a proof of concept.
* Update Master Test Plan established in LCOM.
* Establish Test Cases and Test Scripts for testing the implemented core use case.
* Establish Unit Testing for the implementation done on the major use case with documentation.
* Conduct User Acceptance Test for the core use case.
* Identify various technical risks that might affect the project in the long term.
* Identify issues that have occurred during the elaboration phase.
* Establish an updated Project Plan.

# Evaluation Criteria

The following are the evaluation criteria for assessing the elaboration phase:

* Project Architecture finalized.
* Executable architecture established.
* Search Product Test conducted.
* Risks identified.
* Project Plan updated.
* Project Vision updated.

# Issues

The following are the issues that were met during the Elaboration Phase. The project team must carefully note the issues to prevent them from happening in the future.

|  |  |  |
| --- | --- | --- |
| **Issue** | **Status** | **Notes** |
| Lack of regular communication between team members. | Solved | Busy schedule with work commitments and other subject load hampered communication between team members. This made team members not up to date on where everyone stands in terms of task completion. Team members were advised to reply and update team members on their tasks on Telegram every day. Team meeting were conducted as usual every week. |
| Lack of compliance with the schedule made. | Solved | Team members had issues with completing tasks on time mostly because of work commitments and other subject load. Team members were given extensions on their tasks |
| Lack of communication with Sponsor during semester break and public holidays. | Noted | Team member hesitation to contact Sponsor through email during the break caused tasks to be done without a view of what needed to be done. Team members were notified about this so that it can be prevented in the future. |
| Sequence diagram were not produced significantly delaying implementation tasks. | Solved | Implementation tasks were delayed because the delay in establishing design documents. Sequence diagrams were made a priority and completed within a week. |
| Error in Jersey version compliance. | Solved | Jersey dependency in the Maven Project created errors due to bugs in the versions used. Jersey 2.26-b06 was used with added dependency to solve the issue. |
| MySQL connector dependency issue. | Solved | Issue in connecting server with database caused due to corrupted artifact. Deleting and creating a new artifact solved the issue. |
| Windows and Mac environment issues during collaboration. | Solved | Development done on Windows caused issues when on Mac and vice versa during collaboration using Git. Deleting workspace.xml file under .idea folder in the project folder when code is shared solved the issue. |
| Issues with accessing MySQL database on Amazon Web Service. | Solved | The MySQL database deployed on AWS could not be accessed because of security settings on AWS Relational Database Service allowing only particular IP addresses to access the database. Changing security settings to allowing all IP addresses to access the database solved the issue. This will be changed to allowing only stores and warehouse IP addresses to access the database once development is completed. |
| Issues with Database Connection when multiple query connections are made to the database. | Noted | The Database has been deployed to a free Amazon Web Service. This allows only a limited number of connections to be made to the database in a set period of time. Therefore, during product deployment a paid version of AWS can be used to deploy database and server to solve this issue. Connections must also be closed every time they are created. |

A detailed description of the risks of the project and its management plan has been described in the latest Risk List document.

# Achievements

The following are the tasks completed during the Elaboration phase. Only a brief description of the tasks completed haven been mentioned below. The version control document can be viewed to check each version of the established documentation and application with brief description of what updates have been made in each version.

|  |  |
| --- | --- |
| **Tasks** | **Description** |
| Updated Architecture Notebook and NFR documents. | The Architecture Notebook and the NFR specification that were established during inception phase were updated according to the feedbacks received. More analysis on choosing an architecture was done as per the feedbacks received. |
| Updated Domain Model and Design Class diagram. | The domain model and the design class diagrams were updated as soon as requirements became clearer. These include adding entities and removing entities as required. |
| Establish Full Use Case Description | A full description of all use cases was established. This included all the possible flows of each use cases along with a description of the scenarios and the NFR of the use case. |
| Establish Sequence Diagram | A sequence diagram was established for all use cases showing the interaction between objects in each use case. This was used during implementation of the core use case. |
| Establish an Executable Architecture | Search Product was chosen for implementation as a proof of concept. This was done so since searching for a product in each location (i.e. two stores and warehouse) is one of the key requirements of the system. Moreover, searching for a product acts as the core of the system and allows other use cases to be build on top of it during future implementation. Similarly, other key use cases like send stock and accept stock cannot function if search product is not available in the system.  The implementation of Search Product has been done using a 3-layer web architecture. A Restful web application has been developed in the server side to process all request from client, retrieve data from the database and send a JSON response. This server-side application has been deployed in AWS using Elastic Beanstalk. MySQL has been used in the database which has been deployed separately in AWS. Initial data were added to MySQL database for testing purpose. Finally, a restful client has been developed to send a request to the server using JavaFX and Jersey Client. An .exe for Windows and .dmg for MacBook has been generated for testing purposes. Only the .exe file will be used in the production environment when the whole application is developed and tested.  The search product implementation allows users to enter a product code or a product item code to view its details and contents in each location. Moreover, searching using a product code allows user to see details of each product item belonging to that product code and the contents in each location. A product search using either product code or product item code takes less than 3 seconds therefore fulfilling the performance requirement of the use case. Similarly, all data displayed are dynamic and changes when it is changed in the database. All security aspects of the system have been ignored during the implementation of search product. This has been done to make development process smooth. The security aspect of the application will be taken into consideration once Log in/Log out use case has been implemented. |
| Update Master Test Plan | The Master Test Plan document which was established during LCOM was updated. Test Scripts and Test Case were added as a milestone and in the test input output section. The test environment which was realized during implementation of Search Product was added too. |
| Establish Test Cases and Test Scripts for the implemented use case testing. | Test cases and scripts for each test case was established for the search product implementation. These documents were followed when conducting the required tests. |
| Conduct Unit Tests for Search Product implementation | Unit Tests were conducted for necessary classes to validate they perform the tasks they have been designed for. Junit and Mockito libraries were used to conduct these tests. Test cases and scripts have been documented for each test conducted. |
| Conduct User Acceptance Test for Search Product Implementation | A User Acceptance Test for the Search Product feature was conducted. An executable file was created similar to the one that will be used in the production environment which consists of the Search Product feature. This executable file was used to conduct the User Acceptance Test for Search Product unit of the system. |
| Update Risk List | Technical risks that were identified during system design and implementation of search product were added to the risk list. The risk description, area it effects, symptoms, triggers and strategy to control the risks were added in the risk list. |
| Update Project Plan | The Project Milestone and Objectives for each iteration were updated when schedules needed to be changed for task completion. |
| Update Project Vision | The Project Vision document was updated by updating the system requirements that was realized as the project progressed. The Development Environment section in the Project Vision was also updated according to the needs that were realized during the implementation of Search Product Use Case. |

# Assessment against Evaluation Criteria

The following are the assessment made against each evaluation criteria:

|  |  |  |
| --- | --- | --- |
| **Criteria** | **Description** | **Assessment** |
| Project Architecture finalized. | The architecture notebook and NFR specification were updated to better analyse architectural options and make a decision based on the analysis. | Completed |
| Executable architecture established. | Search Product which is one of the major use case was implemented allowing users to search for products using product code or product item code from a database. A production level executable jar was produced allowing multiple users to use the application. This validates the architecture chosen allowing other use cases to be build on top in the future. | Completed |
| Search Product test conducted | The Master Test Plan was updated according to the feedbacks received from LCOM. Test Cases and Test Scripts were established for Search Product Tests. Unit tests for required classes was conducted using Junit and Mockito validating the various classes and methods used. Finally, a UAT was conducted using the executable file of the Search Product Implementation which passed all requirements. | Completed |
| Risks identified. | Various technical risks that might hamper the project in the future were identified in the updated risk list. Their description, symptoms, causes and control strategy have been described in the updated risk list. | Completed |
| Project Plan updated. | Milestones and objectives have been updated regularly as required to complete tasks that have could not be completed according to the initial project plan. | Completed |
| Project Vision updated. | An updated project vision was established. All requirement that were further narrowed down and realized during the elaboration phase was added in the updated project vision. Development environments which was realized during the implementation were also updated. | Completed |

# Conclusion

In conclusion, the elaboration phase has been a success. The required tasks for elaboration phase have been completed as described above. All three members had active participation in the process. Even though there were a few team issues specially with task completion during Iteration 3 and 4, proper task allocation and management allowed these issues to be solved. Team members have learned the value of time and that time lost at the beginning of the project is a time that never comes back. It has been known to team members that a more robust scheduling and task allocation is required in the future to make proper utilization of the time available. Moreover, team members have realized that the responsibilities must not be changed often as this can create chaos during the transition of responsibilities. Change of responsibilities will be conducted more carefully and less frequently in the future. Similarly, communication with the Sponsor will be made frequently if any issues arise through email. This will be focused during term break in the future as it caused some tasks to be delayed during the elaboration phase. All in all, team members will continue to put this effort and dedication towards the project to make it a huge success.