**C868 – Software Capstone Project Summary**

**Task 2 – Section A**



|  |  |
| --- | --- |
| **Capstone Proposal Project Name:** | Engine-uity Rebuilds Inventory Manager |
| **Student Name:** | Philip Parks |

Table of Contents

[Business Problem 3](#_Toc182470838)

[The Customer 3](#_Toc182470839)

[Business Case 3](#_Toc182470840)

[Fulfillment 4](#_Toc182470841)

[Existing Gaps 4](#_Toc182470842)

[SDLC Methodology 5](#_Toc182470843)

[Deliverables 6](#_Toc182470844)

[Project Deliverables 6](#_Toc182470845)

[Product Deliverables 8](#_Toc182470846)

[Implementation 8](#_Toc182470847)

[Validation and Verification 9](#_Toc182470848)

[Environments and Costs 9](#_Toc182470849)

[Programming Environment 9](#_Toc182470850)

[Environment Costs 10](#_Toc182470851)

[Human Resource Requirements 10](#_Toc182470852)

[Project Timeline 11](#_Toc182470853)

# Business Problem

## The Customer

Engine-unity Rebuilds is an engine rebuild service provider specializing in the rebuild of engines where the manufacturer of parts for the engines no longer exists or makes parts for that engine. They are a small business at one location owned by a veteran. The owner currently employs four engine technicians and two machinists. They are easily able to keep up with their current workload.

The owner uses an office computer to communicate with customers and perform other administrative responsibilities. 2 computer systems in the shop are used for diagnostic work. They use the Chamber of Commerce’s recommended internet service provider for wired and wireless connections to the internet.

At a recent town hall meeting, the owner recognized an opportunity to take on large contracts for construction, military, and public service vehicles. The owner has contacted the local construction companies and town officials to propose contracts for repairing and maintaining the older vehicles in use with positive feedback. The town officials are adamant that the owner establish a better records-keeping system before initiating a contract between the owner and any other business or government agency. The owner currently keeps a paper inventory that he transfers to an Excel Spreadsheet on his office computer once a month.

## **Business Case**

Engine-uity Rebuilds Inventory Manager – the proposed software solution – is an inventory management system that will track engines in inventory and the parts needed to repair them. The owner of Engine-uity currently keeps paper records that they transfer to an Excel Spreadsheet once a month, and the town officials are insisting that the owner transitions into a more modern record-keeping system to start a maintenance and repair contract with local construction companies (who already have a modern system) or any other entity. The construction companies agree that a more modern system is more efficient for their needs. The inventory manager will provide reporting, record keeping, and retrieval, as recommended by the town officials.

## **Fulfillment**

Engine-uity Rebuild Inventory Manager will be a stand-alone Java application for use by designated inventory managers, machine operators, and administrative staff. Administrative users can create new accounts. All records will be maintained behind a secure sign-in to comply with the town officials' request. The application will interface with a MySQL database.

The application will allow for the entry of parts that are machined on-site and parts that are obtained from distributors. For parts made on-site, there will be a workstation associated with the part. The parts obtained from a third party will have a distributor associated with each part. An administrative user must add distributors and workstations before they can be related to a part.

The application will also be able to provide reports for engines by their status, in-house parts by the workstation used to create them, and outsourced parts by the distributor that they were procured from.

# Existing Gaps

The owner currently has a work logbook that he uses to track the engines currently in inventory. The logbook also keeps track of the parts used to rebuild the engine.

The owner manually captures inventory information in a logbook for parts made in-house. When a part is used on an engine, the logbook is updated. The parts they receive from distributors have the owner's invoices in a filing cabinet. When an outsourced part is used on an engine, they find the part on the associated invoice in the filing cabinet and use a pen to cross it out to indicate that it has been used.

The owner has separate Excel spreadsheets for in-house parts, outsourced parts, and engines that he updates from the logbooks and invoices once a week.

The inventory manager application will eliminate the need to update spreadsheets and put all records about engines and the parts required to rebuild them in one secure application.

The application's main view will have three tabs. The first visible tab will be a table showing all the engines in inventory. There will also be a parts tab that will include two tables; One will be the parts made in-house, and the other will be parts ordered from a distributor. The last tab will be for reports and adding new users (admin Tab).

# SDLC Methodology

The Waterfall method of developing software applications has been chosen to deliver the features for the Engine-uity Rebuild Inventory Manager application. This method was chosen because the project’s features are known and laid out at the outset. Based on the set of agreed-upon features, development can follow a predictable series of steps or stages, each with its deliverables. The Waterfall method suits a project like Engine-uity Rebuild Inventory Manager since the rebuild shop's requirements are well understood.

The Waterfall method begins with the **requirements phase**, which seeks to understand the current situation of Engine-uity Rebuilds and their needs. This is the stage where the comprehensive set of features is enumerated, and the desired functionality of the software solution is described in detail. The main output from this phase is a requirements document that has been distributed to all stakeholders. In addition, a project schedule that lists the project stages with approximate time estimates will also be provided at this junction.

The requirements phase is followed by the **system design phase**, in which the system is developed from the ground up. While this stage involves no coding, the system's technical specs, such as the programming language(s), database diagrams, deployment, and testing strategies, are decided upon. The deliverables tied to this stage are a low-fidelity Wireframe to show the flow of the application. This will demonstrate the UI design, a database diagram that sketches the structure and relations between the data models, and a testing plan to ensure complete and correct functionality.

The next step is the **implementation phase, which is** when the actual design and development of the software project takes place. The application's building can commence based on the outputs from the previous stages. The output from the implementation phase is a functional software product that fulfills all the functionality specified in the requirements document.

Following implementation, the **testing phase** aims to find and report all deviations from the agreed-upon requirements. Quality assurance involves thoroughly battle-testing the application and listing bugs and errors. The engineering team will immediately act on all bug reports and fix any problems.

Once all the known problems have been resolved, the **delivery or deployment phase** can begin. This will put the application into the live production environment and prepare it for use by Engine-uity Rebuilds. The deliverable here is a software application that Engine-uity Rebuilds can start using.

The **maintenance phase** ensures Engine-uity Rebuilds can still turn to the developer as issues arise. Maintenance consists of fixing newly discovered bugs, creating patches, and making updates as requested by Engine-uity Rebuilds. There will be a maintenance contract that will specify the terms for supplying ongoing maintenance services.

# Deliverables

The Waterfall software development method has well-defined deliverables for each project lifecycle stage. These can further be subdivided into project and product deliverables representing the software product delivered to Engine-uity Rebuilds.

## **Project Deliverables**

* Requirements document
  + Outlines features and intended functionality of a software application in detail and represents the software developer’s understanding of Engine-uity Rebuilds needs as well as the exact nature and constraints of the system
  + It needs to be written to be clearly and unambiguously referenced throughout all project stages.
  + Most importantly, it should include details about user interfaces, functional capabilities, performance levels, data structures, quality and reliability metrics, and constraints and limitations.
* Project Schedule
  + Detailed timeline that lists the main stages and substages of the project along with their respective deliverables and estimated time ranges
  + It should include tasks and milestones and provide a brief description of each.
* Low-fidelity Wireframe
  + Mockups of the entire application are designed to demonstrate the basic flow of the website and the most important functional aspects.
  + It does not need details of the design elements that will eventually be used. Still, it should clearly show the main navigational components and the different ways to interact with the application.
* High-fidelity Design Prototype
  + Based on the wireframe, the high-fidelity prototype will focus on the look and feel of the application and contain the complete set of theme elements such as typography and color scheme.
* Database Diagram
  + It will show the blueprint for the database schema, including structures, data types, and relations between the different database tables.
  + It will give a comprehensive picture of the database representation of real-life entities such as doctors, patients, and appointments.
* Testing Plan
  + Outline of the included unit tests suite to ensure the correct functionality of the software application. Unit tests are closely based on the code designed to test and can take advantage of white-box testing methods. The unit test suite will run automated and produce structured output.
  + Functional test plans will involve performing a series of steps to check the end-to-end functionality of the application. Each step will have clearly defined inputs and outputs, and the goal is to ensure that the expected outcomes are produced.

## **Product Deliverables**

* Functional database with custom schema that matches the tables laid out in the database diagram and powers the data storage needs of the application
* A fully developed application that fulfills all the functionality specified in the requirements document
* Navigation and flow of the application that match the patterns shown in the wireframe
* Graphical user interface that matches the prototype
* Secure admin access for several selected staff members
* Deployment to the office computer
* Support with initial data import to facilitate the process of getting started with using the application

# Implementation

The implementation of this project is expected to go smoothly and not cause any significant disruptions to the operations of Engine-uity Rebuilds. In the requirements phase, the leadership team at Engine-uity Rebuilds will be worked with to understand the exact requirements and later to make any needed amendments. This should only require a few 60 to 90-minute meetings. One of the advantages of the Waterfall method is that Engine-uity Rebuilds only need to be involved in the early and the late stages of the project lifecycle rather than needing to make representatives available throughout the project. In addition to the project requirements, stakeholders from Engine-uity Rebuilds will also need to sign off on the project timeline and the low and high-fidelity wireframes. All this will ensure that Engine-uity Rebuilds has a solid idea of the product they will be receiving.

During the testing phase, Engine-uity Rebuilds staff will be selected to pioneer the adoption of the new software application and will be adequately trained to use it. This will also put Engine-uity Rebuilds in an excellent position to perform acceptance tests successfully before releasing the project for deployment.

The application will be deployed to the production environment with a database with a generic admin account so the owner can initially implement user accounts. After that step is completed, Engine-uity Rebuilds should be able to use the new and greatly improved workflows to manage the tracking of engines and the parts used to rebuild them.

# Validation and Verification

A comprehensive testing plan will be devised to ensure the new inventory manager application looks and works as outlined in the requirements document. Testing and quality assurance are central to all projects, and nearly complete test coverage of the code bases will be pursued to the greatest extent.

In conjunction with writing code, **unit tests** will be created to verify the functionality of every method called in a running program. Unit tests are low-level tests that check methods in isolation and confirm an expected behavior or output given inputs. The suite of unit tests can be run automatically. Hence, it is ideal for use during development as more and more features are added and on deployment events to ensure that changes to the code do not break or alter existing functionality.

**Functional tests** will be devised and executed at the next level as QA. Functional testing strives to test specific requirements and parameters of the inventory manager. A set of functional tests will be created to verify the functionality of all requirements identified in the requirements document. This will go a long way in ensuring that all the agreed-upon functionality is implemented fully and correctly.

Finally, the inventory manager is only viable if it can perform its function in Engine-uity Rebuilds’ real-life environment. **Acceptance tests** ensure that the product meets Engine-uity Rebuilds’ needs and is valuable and usable in performing everyday day-to-day tasks in Engine-uity Rebuilds’ environment, such as adding and modifying parts, adding and modifying engines, adding users, running reports, etcetera.

# Environments and Costs

## **Programming Environment**

The machine being used to develop the application is a 2020 MacBook Pro. The macOS is Sequoia 15.0.1. The software used to create the proposed application is as follows:

* IDE: IntelliJ IDEA CE 2024.1
* UI components: SceneBuilder
* Database: MySQL version 8.0.37
* Database interface: MySQL Workbench version 8.0.34
* Database connection: mysql-connector-java-8.0.25

## **Environment Costs**

The database will be using MySQL CE, the free version. The database and application will be deployed on the owner’s computer in their office. The computer must have a dual-core processor and at least 4GB of RAM. The computer will also need 1.2 GB of disk space for the initial installation of MySQL. My SQL workbench will require 500MB of free disk space to install. The cost of the 13” MacBook 2020 was roughly $1,500 when it was purchased.

The deployment environment will be identical to the development environment and thus can easily meet the disk and RAM space requirements.

## **Human Resource Requirements**

This project will take a total of 120 hours to complete. One project manager is taking on this project to add to their portfolio and demonstrate to the local officials and chamber of commerce the benefits of automating and modernizing their records systems. This project will hopefully lead to more local businesses hiring the project manager for other projects related to software development to improve their business processes. The following is an overview of what the cost might be were there to be an entire team working on this project:

|  |  |  |
| --- | --- | --- |
| **Resource** | **Rate \* Time** | **Total** |
| Project Manager | $62/h \* 80h | $4,960 |
| Software Developer | $52/h \* 120h | $6,240 |
| QA Specialist | $45/h \* 40h | $1,800 |
| **TOTAL Human Resource Cost** |  | **$13,000** |

# Project Timeline

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Phase | Milestone/Task | Deliverable | Description | Dates |
| Requirements gathering | Discuss and finalize requirements with owner and employees | Requirements document and project schedule | Meetings between the project team and Engine-uity Rebuilds stakeholders to gather and prioritize application requirements | 10/15/24 - 10/18/24 |
| System Design | Create and review low fidelity wireframe and high-fidelity design prototype | Low and high-fidelity application mockups | The software developer will create a lo-fi wireframe to show navigation and interactivity and a design prototype that will show the final look of the application | 10/19/24 -10/22/24 |
| System Design | Create database schema for site | Database Diagram | The developer will define and model entities to be represented as database tables and identify relationships between them | 10/19/24 -10/22/24 |
| System Design | Create overall testing plan | Testing Plan | The PM and developer will create a comprehensive plan for unit, functional, and acceptance testing | 10/19/24 -10/22/24 |
| Implementation | Create functional application including unit tests | Final alpha version of the application | The developer will implement feature requirements and match prototype design to create a fully functional application according to specs. All unit tests will run without errors. | 10/23/24 -10/24/24 |
| Testing Phase | Devise and execute functional tests | All functional tests are executed successfully | QA specialists creates and runs functional test to verify all the requirement spelled out in the initial requirements document. | 10/25/24 – 10/26/24 |
| Testing Phase | Devise and execute acceptance tests | Engine-uity Rebuilds executes acceptance tests to satisfaction | In coordination with PM, Engine-uity Rebuilds will run acceptance test to verify that the application meets requirements. | 10/25/24 – 10/26/24 |
| Deployment | Prepare deployment environment and deploy application | Deployed application ready to be used by Engine-uity Rebuilds | The developer will set up the database and deploy the application after final checks. | 10/27/24 |
| Maintenance | Set up maintenance plan | Maintenance service contract | The PM will discuss and finalize terms of service for ongoing maintenance with Engine-uity Rebuilds. | 10/27/24 |