C868 – Software Capstone Project Summary

Task 2 – Section A

Capstone Proposal Project Name: Football Roster

Student Name: Casey White - 1303102

Table of Contents

Contents

Business Problem

The Customer

The customer is a football organization called Seminole State Junior College. They have several coaches and assistants on staff with typically a team of 70+ football players. They are looking for a way to keep track of their rosters digitally versus on paper. They've requested software to be able to keep track of rosters, depth charts, and injury lists. They will have one person keep records, so the program needs to be single user use only.

They expect to be able to export the depth chart so they can present it to officials before a game. They would also like to be able to import or export data so they can either share information or transport it to a different computer.

For the players, they want several attributes to be tracked so they can see their progress. A list of the attributes are as follows:

- First Name
- Middle Name
- Last Name
- Birthday
- Age
- Weight
- Height
- Max Bench Press
- Max Squat
- 40 Yard dash time
- Position

Everything listed other than first name, middle name, last name, and birthday will be unique to the specific year. This will allow them to traverse between years and track player progress.

Football Roster Program

Business Case

With how the team is currently tracking players, it is inefficient and manual work to be able to track player progress. The software will allow them to do this with a click of a button navigating years. This will be more efficient in easily managing players and exporting lists. When they need a quick list for the current injury list and depth chart, they can simply export it quickly.

Their current process takes a lot of time to write down all players and isn't easy to make changes. They must either redo the whole sheet or try to erase and write over what was there. The software will make these task much easier and clean.

The system is scalable for the future if the customer wants to be able to change to a multiuser program. The database can be stored remotely in the cloud or on a local server. The program will simply need to be updated to connect to the central database and that's it! The authentication will be controlled by the database management system.

Fulfillment

The product is written in Java and will not have a login form. The database is secured down by the operating system. The database is stored locally in the user's home directory. This allows the operating system to handle the authentication and security. Only the user can access their home directory which in turns allows only that specific user access to the data. It will be stored under %UserProfile%\FootballRoster\FootballRoster-main.db.

When the application is launched, it shows the home screen where many choices can be made from adding a player, changing years, changing the team name, adding a year, and many other features. Functions the user can do are listed below:

- Add a year
- Edit team name for the year
- Add a player to the roster

- Modify player attributes for the roster year
- Delete player from the roster
- Create a new player
- Modify player personal information
- Completely delete a player
- View and modify the depth chart
- Export the depth chart and select attributes to be exported
- View and modify injury list which is included in the depth chart export
- Import or export the data

The only interface any system will have with another is with import or exporting data. Exporting The data exports everything in the current user system and importing replicates bit by bit. It produces a direct representation of the export. It allows the customer to do what they wish the data.

Existing Gaps

The current process is on paper which can be burned or fade over time. Having it digital, will allow it to be backed up and saved for archival purposes. As stated earlier, if the customer wants to track how much progress a player has made, they have to filter through the file cabinet and find they years they are looking for and then spread it across a table hoping that it's still visible. The program will allow them to easily click between years to view the progress of a player. They can even filter the roster by the player's name.

With the current process be written, it can also be unreadable because of sloppy handwriting. All font is clear and readable on the program. The program will also have type validation such that letters can be entered into numerical data sets.

All these issues with the current process present a reason for developing a program so that things can be recorded digitally.

SDLC Methodology

Considering the nature of your project, select a Software Development Life Cycle (SDLC) methodology that will be used to manage the project. Those may include……. Be sure to describe the process you select first and why it's a good fit. Then review the methodology phases and what part of the project will align with each.

For example:

The SDLC Methodology utilized in this project is Waterfall because... The requirements are well understood and defined. The customer is located a significant distance … testing is not logistically possible. The system will be implemented fully…The Waterfall methodology chosen will include the following phases…etc…

The project will use the waterfall method. This is best because the requirements are clear and simple. The coaches are busy and don't want to be bothered as well. It will be a quick and simple streamlined development process and will satisfy the customer. The phases are detailed as followed.

Requirements*:* This phase will focus on the expected data recorded by the program. The customer has a it's current process with all current data that they want to keep track of. We can simply mimic the type of data recorded in their current process and implement into the software.

Planning & Designing*:* This phase focuses on planning the functionality of the software and the graphical user interface design. Planning will involve figuring out the design of the database and classes for java. Designing will focus on the presentation. The customer stated that they don't want or expect anything fancy. Their focus is the function of the program so the design will be geared towards user usability.

Implementation*:* As stated before the program will be developed with java using IntelliJ by our well-versed software engineer. Everything will be laid out for the engineer to follow and build from the ground up following requirements by the customer.

Football Roster Program

The database will be setup using SQLite with CRUD (Create, Read, Update, and Delete) functionality. The tables with their respective primary keys and foreign keys designed in the prior phase will be developed.

After the backend is setup and ready for data, the engineer will develop the graphical user interface. During that development, connections and queries will be programmed as well.

***Verification**:* This phase will follow a testing plan set forth in the planning and design stage. The testers designated earlier in the process will go through the program and look for the requirements specified by the customer. After all testers feel that the requirements are met, we will let the customer have a test time using the software transferring their old archived data from paper into the software.

When the customer is finished inputting old data into the software and given time to try it out, only then if the customer is satisfied will it be verified for use.

Maintenance: This part focuses on any bugs discovered by the customer. If the customer finds any bugs can be reported and will fixed with a software update overwriting the existing program. Any bugs fixed will be thoroughly tested and validated before implementing.

Deliverables

The deliverables will be sent to the customer as they are completed and validated. The value presented by these will show the progress and direction of the process. It will give the user an abstract idea of the expected program coming to reality.

Project Deliverables

- Test Plan
  - This shows a single unit test to show as an example of type of test for validation.
- Project Schedule
  - When and what part of the process is expected to be completed
- Diagrams
  - Two diagrams: class diagram and design diagram.

Product Deliverables

- A fully functional frontend
- An embedded database to local user that can be easily managed by experienced SQL Developers
- Exports/imports of data
- Generate list of depth chart

Implementation

The program is a standalone exe and can be ran on any system. We will put the executable on the customers system specified by them. We will install java runtime so that the program can be run by the operating system. The program is local to the user so whatever account they are logged into, is the only account that can access the data unless they export the data.

Validation and Verification

The customer's requirements will be tested thoroughly. After our test team fully vets the program, our engineer will go to the customer and help input the old archived data into the program. While doing this, they will ensure that they can perform all functions that are required by the program and check the expected output.

At the end of that testing, we will hold a meeting going over any concerns with the customer and what is expected for future use.

Environments and Costs

Programming Environment

The program will be developed using java development kit 14.0.1 with JavaFX module on Windows 10 Version 1909. SQLite will be the backend for the program. It creates a simple backend but reliable persistent storage.

Environment Costs

Costs to be expected are the development of the program itself. All other costs are covered by the customer. They have their own computing resources and information technology staff.

The program will cost $2,324.29 overall. This covers the entire process from design and planning to implementation and validation.

Human Resource Requirements

The developer will be the main resource. Other resources include program manager, test team, and cyber security. Another resource will be the maintenance group, any bug fixes in the future will be handled by them.

Project Timeline

Phase	Milestone/Task	Deliverable	Description	Dates
Requirements	Task 1	Requirements	Meet with coaches to go over what is expected	10/10/2020 – 10/11/2020
Design	Task 2	Design diagram	Create a diagram showing the flow of the application	10/12/2020 – 10/15/2020
Planning	Task 3	Class diagram	Create a diagram showing the structure of the database	10/17/2020 – 10/22/2020
Implementation	Task 4	Project schedule	Develop Backend	10/25/2020 – 10/29/2020

Implementation	Task 5	Project schedule	Develop Frontend which is most of the work	11/02/2020 – 11/11/2020
Validation and Verification	Task 6	A single unit test, displaying how testing is done.	Develop a test plan and unit test	11/13/2020 – 11/16/2020
Maintenance	Task 7	Updates	Fix any bugs reported for two years	12/01/2020 – 12/01/2022