C868 – Software Capstone Project Summary

Task 2 – Section A

Capstone Proposal Project Name: Oregon Dentistry Scheduling Application

Student Name: Spencer Swift

Table of Contents

Contents

Business Problem

The Customer

Oregon Dentistry is a small dental office that provides quality dental care and has an extremely high standard of customer service provided by a small full-time staff. The following are Oregon Dentistry's objectives:

- Provide professional care in a friendly environment.
- Offer preventative, cosmetic, restorative, and emergency services.
- Have technology that makes visits smoother and more efficient.

Business Case

Oregon Dentistry currently manages all appointments using a pen and paper system. The receptionist currently records the appointments along with some basic customer information on pen and paper. The new system will manage customers and appointments in a Windows desktop application it will prevent creating duplicate appointments and have functionality to remind the user of upcoming appointments. The application can be installed on multiple desktops making it so doctors can see and modify the schedule from different locations in the building. Having the scheduling done digitally will speed up the process of creating, modifying, and deleting appointments. This will give the office higher reliability of data and help to build the office's customer database.

Fulfillment

A Windows desktop application will have a user login where upon login they can see an overview of appointments for the week or month (based on a toggle). There will be buttons for generating different types of reports, functionality to add, delete, and modify customers and appointments. There will also be functionality to select a customer or consultant and view appointments tied to the individual. The system will utilize a MySQL database to provide data consistency among different desktops.

Existing Gaps

The existing system is a manual and handwritten system that uses a log sheet. The receptionist receives a phone call and manually writes down customer information and appointment information in the book. This system is limited in its reliability and is prone to data entry errors. In addition to this customer data is not stored so phone number and address must be inputted every time. The new system will only require you to create a customer one time and then you can add appointments to that customer.

SDLC Methodology

The new system will be created using the waterfall methodology. This will allow development time to be much quicker as the requirements are not expected to change and will be clearly defined prior to development. The 5 steps of the waterfall model and how they will be implemented in the project are as follows:

1. Requirements – clearly define the requirements for the project. This phase is where clients needs will be established. There may be multiple meetings to establish clear and concise requirements. A schedule for the development of the application will be developed and signed off during this phase.
2. Design – create a wireframe and implement software architecture. In the design phase the ERD is created to show how the database is structured and how objects within it relate to one another. Mockups of high-fidelity UI elements will be created at this point as well. These deliverables will be attached in Part C "Class Design" and "UI Design".
3. Implementation – take the specifications and design and write the code for the application. The developer will at this point use the requirements and design documents to create a working and complete application. The deliverable for this phase will be the working application.
4. Verification – perform unit testing on the application and deliver the complete application to the user. During this phase unit testing will be done to confirm that the application's development was a success. Bugs will be reported and fixed before delivering the application. Once the application is delivered the user will perform acceptance testing. The key deliverable for this phase is the unit test.
5. Maintenance – as the customer is using the application address problems that they may have. One the application is delivered the customer will report any bugs that they may find, and the developers will patch bugs and fix any problems that may occur.

Deliverables

There are 2 types of deliverables that are associated with the Waterfall SDLC that the customer has requested. They are project and product deliverables.

Project Deliverables

These consist of items that are part of the Project Manager's realm of responsibilities.

- Project Schedule
- Test Plans
- Mockups/Layout

Product Deliverables

Product Deliverables represents what is produced to deliver to the customer.

- GUI Mockups
- ERD for the application
- C#.net Application

Implementation

The Implementation of the waterfall methodology to ultimately deliver a complete application will begin with the requirements phase. An initial meeting will be held to determine what functions the user will need to be present in the application. A follow up meeting will be held to confirm the previously agreed upon requirements and have a final opportunity for the customer to make changes. Following that meeting a schedule will be delivered to the client and then signed off on.

After requirements are established the design phase will begin. The development team will create an ERD to show relationships for the database and create UI Design mockups. Once the mockups and ERD have been received and signed off on by the client the next phase will begin.

At this point implementation will begin. The developers will create the application according to the design document's ERD and design mockups. Once the application has finished the implementation phase of the waterfall the product will under-go testing to ensure that the login feature of the application works as expected. After testing has concluded and any bugs that arose have been squashed the application will be delivered to the customer along with a detailed user manual. Since it will be running on less than 10 PCs (front desk, and some offices at the dental office) installation will be very simple and straight forward. Oregon Dentistry will run the old handwritten system and the new application concurrently for one week to verify functionality and to allow time to report bugs. This should all be able to be done by the Project Manager and the Software Engineer.

Validation and Verification

Validation and verification will be confirmed when the application is completed according the system requirements, unit testing, and all associated deliverables will be required for verification.

The testing phase will be performed by the team developing the application and will ensure that best practices are followed. The unit testing will occur during the implementation phase by inputting test cases into the unit to verify functionality and that requirements are met.

The customer will perform acceptance testing upon receiving the application and ensure requirements are met.

Environments and Costs

Programming Environment

- Visual Studio 2019
- MySQL Database Version 5+

Environment Costs

Environment costs are relatively minimal. The environment where the system resides in a shared environment where costs are shared by the organizations. There is a nominal fee associated with maintaining the database using Microsoft Azure of $600 a year that allows for unlimited storage size and 99.8% uptime. The final cost is based on the thick or thin clients utilized by the customer. Each device that is attached to the network has a $40 annual fee which covers Operating System and Network upgrades.

Human Resource Requirements

The larger share of human resource is by the developers of the project followed by the PM. Developers will consume approximately 75% of the hours and dollars associated with the project. The project manager will be responsible for roughly 25% of the budget and hours associated with the project. The developers will be responsible for design, testing, and maintenance of the application and the project manager will be most involved during the requirements and interacting with the customer.

Project Timeline

Phase	Milestone/Task	Deliverable	Description	Dates
Requirements	Task 1	Requirements	Meeting with customer and procedure review	7/1/2020-7/7/2020
Requirements	Task 2	ERD	Meet with customer to verify accurate schema of the DB	7/7/2020
Requirements	Task 3	Test Plan	Developers create a plan for testing that gets signed off by the customer.	7/8/2020
Design	Task 4	Low fidelity wireframe High fidelity mockup	Create the UI that relates the look and feel of the project	7/8/2020-7/11/2020
Implementation	Task 5	C# Application	Create the application using the design documents and requirements.	7/12/2020-7/20/2020
Testing	Task 6	Test Results		7/21/2020-7/22/2020
Verification	Task 7	C# Application	Test the application using the unit tests. Deliver the application to the customer for use. Acceptance testing will be performed by the customer.	7/23/2020
Maintenance	Task 8		Ongoing maintenance will be provided to the customer.	7/23/2020-7/23/2021