**C868 – Software Capstone Project Summary**

**Task 2 – Section A**



|  |  |
| --- | --- |
| **Capstone Proposal Project Name:** | Patient Scheduling App: SimpleBook |
| **Student Name:** | Victoria Brown |

Table of Contents

[**Table of Contents** 2](#_Toc130645000)

[**Business Problem** 3](#_Toc130645001)

[**The Customer** 3](#_Toc130645002)

[**Business Case** 3](#_Toc130645003)

[**Fulfillment** 3](#_Toc130645004)

[**Existing Gaps** 3](#_Toc130645005)

[**SDLC Methodology** 4](#_Toc130645006)

[**Deliverables** 4](#_Toc130645007)

[**Project Deliverables** 4](#_Toc130645008)

[**Product Deliverables** 5](#_Toc130645009)

[**Implementation** 5](#_Toc130645010)

[**Validation and Verification** 5](#_Toc130645011)

[**Environments and Costs** 5](#_Toc130645012)

[**Programming Environment** 5](#_Toc130645013)

[**Environment Costs** 6](#_Toc130645014)

[**Human Resource Requirements** 6](#_Toc130645015)

[**Project Timeline** 6](#_Toc130645016)

# **Business Problem**

**The Customer**

The customer is Tennessee Family Medical Care, a newly opened doctor’s office in Nashville, Tennessee. The office currently employs six doctors, including general practitioners and specialists, as well as an adequate number of nurse practitioners and medical assistants. The patient base of the office is growing rapidly, as Tennessee Family Medical Care has just been named one of Nashville’s best care facilities. Appointments are currently being booked months in advance, and the office’s ability to track the appointments is severely lacking. The current systems are outdated, which is causing a strain on the general staff. Currently, appointments are taken over the phone and written on each doctor’s calendar. It is then up to the doctors to accurately track their own appointments. As the doctors have limited time throughout the day, this has become a more pressing issue with the rise in patient appointments. An up-to-date technology solution is a dire need to support the current and future operations of this facility.

## **Business Case**

SimpleBook is an appointment scheduling system designed specifically for medical offices. It allows office staff to track patients with a specified doctor, as well as add, delete, and modify said appointments. Currently, the staff is overwhelmed with the number of appointments, and their tracking system is causing even more unwanted stress. The manual tracking of appointments using calendars has led to double-booking consultations and doctors forgetting appointment times as they are required to track their calendars themselves. Additionally, returning customers have been booked with slightly different information each time. This has led to customers not getting the help they need and becoming frustrated with an otherwise fantastic medical office. SimpleBook will address these issues by centralizing the data into a single location.

## **Fulfillment**

SimpleBook will be a standalone application written in C#, and it will also use a MySQL database for the storage of all office data. To log in to the system, a consultant will enter personal credentials at the login screen that will then be validated against the database. Once a consultant enters the system, they will be taken to the main page of the application. This page provides a view of all the appointments, search options, and control buttons to navigate to the other application pages. Users can search for appointments based on customer ID, appointment type, and date range. Clicking on the ”View Appointments” button will navigate to a page where the user can filter the appointments by week and month. Users who click on the “Appointment Types Report” button will be shown a report that shows the types of appointments for the previous, current, and upcoming month. The “Consultant Schedule” button will show a report of filtered appointments scheduled for each consultant. The “Patients” button will show the user a list of patients and allow the user to add, edit, and delete patients. The ”Appointments By Patient” button will show a report that filters the appointments based on patients that have upcoming appointments.

# **Existing Gaps**

The existing system utilizes a manual and handwritten system for tracking appointments. Once a patient calls to schedule an appointment, a staff member writes this appointment in the respective doctor’s calendar. This calendar is then given to the doctors who are expected to track appointments themselves. This leads to double-booking and lost appointments, as the doctors are often too busy to track the appointments accurately. By using SimpleBook, the staff can enter patient and appointment information in a single application, and the data is saved and tracked as it is entered. This will lead to better data validity and reduce redundancy in customer records. Doctors will no longer be required to track the appointments themselves as they will be given reminders for upcoming appointments as they happen.

# **SDLC Methodology**

Considering no projects of this kind have been implemented before at Tennessee Family Medical Care, it is unlikely that the first iteration will provide everything they need for the long term. The application should be implemented as soon as possible to prevent losing customers and damage to the company’s reputation due to missed appointments. This is why we recommend the Agile methodology. This will allow for the implementation of the main application as soon as possible, while less critical items can be added during further development.

The six stages of the Agile methodology are as follows:

* Planning
* Design
* Coding/Development
* Testing
* Deployment
* Review

Each stage will be used while developing SimpleBook. During the planning phase, interviews with the staff will be conducted by the developers to develop stories and understand the perspectives of everyday users. The interviews will provide an understanding of the main needs of the company to determine what aspects of the application should be implemented first. The design phase will also be present during some points of the planning, as the developers must ensure the application meets the requirements outlined by the power users of the company.

Development can begin once the planning and design phases are completed. The coding, as well as the User Interface (UI), of the application is done in this stage. Testing will overlap with the development stage as unit tests are done frequently to ensure a faster total development time. Once the individual pieces of the application have been tested and combined, the application can be tested in full. Testing will be completed by a quality assurance team before being tested by real users in beta. On completion of the beta testing, urgent fixes can be finished before deploying the application to the office’s workstations and cloud environment. Feedback will then be taken before starting the second iteration of the Agile process. The separate iterations of the Agile process are the main benefit of choosing it over another methodology. The application can be planned and delivered quickly, while less vital features can be added later.

# **Deliverables**

There are two types of deliverables associated with the Agile SDLC, project and product deliverables. The details for the deliverables are outlined below.

## **Project Deliverables**

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

* Project Schedule
  + What must be done and the time it will take to complete.
* Product Backlog
  + This is the document that will be used to decide what will be done for each development iteration. Both in and out-of-scope requirements are listed for the current cycle.
* Wireframe/UI Documentation
  + A rough outline of the user interface.
* Sprint Backlog
  + A list of what will be completed by the development team in the development cycle.

## **Product Deliverables**

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

* Customized database
  + The database used by the SimpleBook application, which will be contained in the office’s cloud environment.
* GUI
  + A fully developed UI that ensures users can access the sections of the application.
* Prototype
  + General staff will have access to a prototype of the application before deployment.

# **Implementation**

Tennessee Family Medical Care has a small IT department that we will coordinate with before deployment. Because the IT department is small, the requirements for deployment are simple. This is also a new system, so no outages are necessary before implementation. SimpleBook will be installed on all endpoints, and the office should have a good network connection to access the cloud servers. As competent on-site technicians are vital to the success of the application, training will be completed with the IT staff to ensure they have adequate knowledge of the application. Trained IT staff will be able to answer inquiries without reaching out to us. The entire deployment process, including training, should not take more than five days to complete.

# **Validation and Verification**

The core functions of this application are to create, delete, modify, and store records, as such, the main testing should be done on its data entry and storage capabilities. Testing will be done to ensure each type of record can be added, deleted, updated, and saved accordingly. In addition to this, testing will be completed to ensure only the users in the database have login capabilities. The search functionality on the main application page will also be tested. After separate pieces are added to the application, unit testing with be completed to make certain the record management is working as intended. Once the application is fully assembled, end-to-end testing will be completed using sample data. Lastly, black box testing will be completed by key users during their training periods. This testing will provide information vital for future development cycles.

# **Environments and Costs**

## **Programming Environment**

To maximize the functionality of this application, the client will need a Windows environment, preferably Windows 10, as well as a dedicated database server. SimpleBook will utilize a MySQL cloud database server which the client already has. Full software requirements are listed below:

* Microsoft Windows 10 (any version)
* MySQL Cloud Database (client owned)

## **Environment Costs**

Costs for the application environment will be minimal as the client already utilizes Windows 10 on the relevant machines, and they have existing cloud servers. The client has a current MySQL license; however, they will be required to pay their cloud provider for extra data usage. According to most cloud-based solutions, the fee will be nearly $500. This will be the only fee added to the yearly budget.

## **Human Resource Requirements**

The highest human resource requirement for this project will be the development team. Nearly 85% of the budgeted hours and funds will be used by the team of developers. 15% will come from the IT team’s administrative duties and user testing.

# **Project Timeline**

The proposed project timeline is listed below. Dates are open to change based on project needs.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Phase | Milestone/Task | Deliverable | Description | Dates |
| Planning | Gathering requirements and planning | Requirements | Meeting with client and gathering user stories | 04/01/2023 – 04/02/2023 |
| Design | Core Design | Low fidelity wireframe  High fidelity mockup | Mockup of the wireframe as well as UI prototypes | 04/03/2023 – 04/05/2023 |
| Development | Complete the first Iteration of the application | Functional application with sample data | Development of the application based on UI prototypes and requirements gathered during planning | 04/07/2023 – 04/18/2023 |
| Testing | Application and Unit testing | List of future bug fixes and test results | Testing of the application will frequently occur during the development phase to lessen development time | 04/21/2023 – 04/25/2023 |
| Deployment | Deployment of the full application | Functional application | Key users will gain access to the application first. General users will gain access after completion of training | 04/28/2023 – 05/03/2023 |
| Review | Application critique | Future goals and process reviews | Development process review and determination of improvement areas for future cycles. Requirements gathering for the next project stage will also be completed | 05/06/2023 |