**C868 – Software Capstone Project Summary**

**Task 2 – Section A**



|  |  |
| --- | --- |
| **Capstone Proposal Project Name:** | Scheduling Application |
| **Student Name:** | Isam Elder |

# Table of Contents

[**Business Problem** 3](#_Toc1937181)

[**The Customer** 3](#_Toc1937182)

[**Business Case** 3](#_Toc1937184)

[**Fulfillment** 4](#_Toc1937185)

[**Existing Gaps** 5](#_Toc1937186)

[**SDLC Methodology** 6](#_Toc1937187)

[**Deliverables** 8](#_Toc1937188)

[**Project Deliverables** 8](#_Toc1937189)

[**Product Deliverables** 10](#_Toc1937190)

[**Implementation** 11](#_Toc1937191)

[**Validation and Verification** 12](#_Toc1937192)

[**Environments and Costs** 12](#_Toc1937193)

[**Programming Environment** 12](#_Toc1937194)

[**Environment Costs** 13](#_Toc1937195)

[**Human Resource Requirements** 14](#_Toc1937196)

[**Project Timeline** 14](#_Toc1937197)

# **Business Problem**

## **The Customer**

The customer who will benefit from the completion of this software application are businesses or organizations that require an efficient and organized system for managing their customer appointments (Bennett & McRobb, 2010). The size of the operation can vary from small businesses to larger corporations, as the software can be customized to suit the needs of businesses of all sizes.

The function of the business/organization can also vary, but any business that relies on appointments with customers, such as medical practices, salons, spas, gyms, education institutions, government agencies, and professional services firms, will benefit from this software application.

Key players in the setting may include receptionists, schedulers, managers, and owners who are responsible for managing appointments, ensuring that appointments are kept on time, and ensuring that customers are satisfied with the level of service provided.

Related IT infrastructure may include hardware and software such as computers, servers, databases, and networks that are used to support the operation of the business (Bennett & McRobb, 2010). This software application can be integrated with existing IT infrastructure to ensure seamless communication and data sharing.

The mission of the business/organization can be enhanced through the use of this software application by improving customer satisfaction, reducing wait times, and streamlining the appointment booking process.

Short-term goals of the business/organization can be achieved through the implementation of this software application by increasing the efficiency of the appointment booking process, improving customer service, and reducing the workload of staff responsible for managing appointments.

Long-term goals of the business/organization can also be supported by the use of this software application by allowing for scalability as the business grows and expands, providing a centralized and accessible repository of customer appointment information, and improving overall operational efficiency.

Projected growth can be supported through the use of this software application by providing a reliable and scalable system for managing appointments, ensuring customer satisfaction, and allowing for future expansion of the business.

**Business Case**

For businesses that rely on appointments with customers, managing those appointments efficiently can be a significant challenge (Bennett & McRobb, 2010). Many businesses still rely on manual methods such as paper-based appointment books, spreadsheets, or even phone calls to manage their appointments. This can lead to errors, double bookings, and missed appointments, which can result in dissatisfied customers and lost revenue.

The software application we are proposing will provide a centralized, easy-to-use interface for managing customer appointments (Bennett & McRobb, 2010). The application will allow businesses to book, reschedule, or cancel appointments, as well as manage customer information and preferences. Additionally, the software application will allow businesses to send appointment reminders and notifications to customers via email or SMS, reducing the likelihood of missed appointments.

The software application will function by integrating with existing IT infrastructure, such as customer relationship management (CRM) systems, calendars, and other applications (Bennett & McRobb, 2010). The application will allow for real-time updates and synchronization across all devices, ensuring that everyone involved in the appointment process has access to the latest information.

By implementing this software application, businesses will be able to streamline their appointment booking process, reduce the workload of staff responsible for managing appointments, and improve customer satisfaction by reducing wait times and ensuring that appointments are kept on schedule.

For example, let's say that a medical clinic is currently using a paper-based appointment book to manage their patient appointments. This method is time-consuming, prone to errors, and can lead to double bookings or missed appointments (Bennett & McRobb, 2010). By implementing the software application, the medical clinic can provide an easy-to-use web interface for patients to book appointments online, reducing the workload of receptionists and other staff responsible for managing appointments. The software application can also send appointment reminders and notifications to patients, reducing the likelihood of missed appointments and improving patient satisfaction.

In summary, the software application we are proposing will help businesses streamline their appointment booking process, reduce errors, and improve customer satisfaction by providing a centralized, easy-to-use interface for managing appointments.

## **Fulfillment**

The software application we are proposing will fulfill the needs of our clients by providing a web-based application that is user-friendly, reliable, and scalable. The application will have a modern and intuitive user interface that is easy to navigate and use, with a customizable layout that can be tailored to the specific needs of each client.

The basic functions of the software application will include:

1. Appointment booking: The software application will allow customers to book appointments online, with real-time availability and confirmation of appointment times. The application will also allow staff to book appointments on behalf of customers, with the ability to view and manage multiple calendars.
2. Appointment management: The software application will allow staff to manage customer appointments, including rescheduling, cancelling, and confirming appointments. The application will also allow staff to view customer information and preferences, and to send appointment reminders and notifications to customers.
3. Reporting and analytics: The software application will provide detailed reporting and analytics on appointment bookings, cancellations, and customer satisfaction. This data can be used to identify trends, improve processes, and optimize customer service.

The software application will interface with existing IT infrastructure, such as CRM systems, calendars, and other applications, using APIs or other integration methods. This will allow for real-time updates and synchronization across all devices, ensuring that everyone involved in the appointment process has access to the latest information.

The user through an easy-to-use web interface will enable the functions of the software application. The interface will be designed to be user-friendly and intuitive, with clear instructions and prompts to guide users through each step of the appointment booking and management process.

The results provided by the software application will include real-time updates on appointment availability, customer satisfaction ratings, and other key metrics. The software application will also provide detailed reports and analytics on appointment bookings, cancellations, and other data points, allowing businesses to identify trends and optimize their appointment management processes. In summary, the software application will fulfill the needs of our clients by providing a reliable, scalable, and user-friendly solution for managing customer appointments.

# **Existing Gaps**

Many businesses currently rely on manual methods for managing customer appointments, such as paper-based appointment books, spreadsheets, or even phone calls. These methods are often time-consuming, prone to errors, and can result in missed appointments, dissatisfied customers, and lost revenue.

The deficiencies of the old process can include:

1. Time-consuming: Manually managing appointments can be a time-consuming process, especially for businesses that have a high volume of customer appointments. Staff responsible for managing appointments must manually enter customer information, check availability, and schedule appointments, which can be a time-consuming and tedious process.
2. Error-prone: Manual processes are often prone to errors, such as double bookings or missed appointments. This can lead to dissatisfied customers, lost revenue, and damage to the business's reputation.
3. Lack of scalability: As businesses grow and expand, manual processes can become increasingly difficult to manage. Businesses with multiple locations, for example, may struggle to manage appointments across all locations using manual methods.
4. Poor customer experience: Customers expect a seamless and efficient appointment booking process, and manual methods can often fall short of these expectations. Long wait times, confusion over appointment times, and missed appointments can all contribute to a poor customer experience.

By replacing the old process with a software application, businesses can address these deficiencies and improve the appointment booking process for both staff and customers. The software application will automate many of the manual tasks associated with appointment booking, reducing the workload of staff responsible for managing appointments and improving the accuracy and efficiency of the process. Additionally, the software application can provide real-time updates and notifications to customers, reducing wait times and improving the overall customer experience.

In summary, the old process of manually managing customer appointments is time-consuming, error-prone, lacks scalability, and can result in a poor customer experience. By replacing the old process with a software application, businesses can address these deficiencies and improve the appointment booking process for both staff and customers.

# **SDLC Methodology**

The SDLC methodology chosen for this project is Agile. Agile methodology is an iterative and incremental approach to software development that emphasizes collaboration, flexibility, and continuous improvement. It is well-suited for projects that require flexibility and adaptability, as it allows for changes and updates to be made throughout the development process.

Agile methodology is a good fit for the appointment booking software application project because it allows for continuous collaboration between the development team and stakeholders, including end-users, which is critical for ensuring the final product meets the needs of the client (Dennis et al., 2018). The iterative nature of Agile methodology also allows for frequent feedback and testing, which can help to identify issues and address them early in the development process.

The Agile methodology chosen will include the following phases:

1. Planning: In this phase, the development team will work with stakeholders to identify project requirements, establish project goals, and define the scope of the project. The team will also create a project roadmap and identify the key milestones of the project.
2. Design: During the design phase, the development team will create a detailed design of the application, including user interfaces, data models, and workflows. This phase will involve collaboration with stakeholders to ensure that the design meets the needs of the client.
3. Development: In this phase, the actual coding and development of the software application will take place. The development team will work in sprints, with each sprint focusing on a specific set of features or functionality. During each sprint, the team will work closely with stakeholders to ensure that the product is meeting their needs.
4. Testing: Throughout the development process, the software application will be thoroughly tested to ensure that it meets the requirements and is free from defects. Testing will occur throughout each sprint and will include both automated and manual testing.
5. Deployment: Once the software application has been developed and tested, it will be deployed to production. The development team will work closely with stakeholders to ensure a smooth deployment and transition to the new system.
6. Maintenance: After deployment, the development team will continue to provide maintenance and support for the software application, including bug fixes and updates.

In summary, the agile methodology was chosen for the appointment booking software application project due to its flexibility, collaboration, and iterative approach. The project will follow the Agile methodology phases of planning, design, development, testing, deployment, and maintenance.

# **Deliverables**

As mentioned above, the Waterfall method of software development has a well-defined set of deliverables tied to each stage of the project lifecycle (Dennis et al., 2018). These can further be subdivided into project deliverables that are under the purview of the Project Manager and product deliverables that represent the actual software product delivered to the customer. We will expand upon both types of deliverables in the following section.

## **Project Deliverables**

For the Agile SDLC methodology, there are several types of deliverables that are associated with each phase of the development process. These deliverables are as follows:

1. **Planning phase:**

* Project roadmap: This document outlines the project goals, timeline, and key milestones.
* Product vision statement: A brief statement that outlines the overall goals and vision for the product.
* User stories: These are brief, narrative descriptions of specific features or functionality from the perspective of the end-user.
* Release plan: A high-level plan that outlines the features or functionality that will be delivered in each sprint.

1. **Design phase:**

* Wireframes: These are rough, visual representations of the user interface and layout of the application.
* Data models: This document outlines the data structures and relationships that will be used in the application.
* Workflow diagrams: These diagrams provide a visual representation of the flow of data and actions within the application.

1. **Development phase:**

* Code: This is the actual code that is developed to implement the features and functionality of the application.
* User acceptance tests: These are tests that are developed to ensure that the application meets the requirements and is usable by end-users.

1. **Testing phase:**

* Test plans: These documents outline the testing strategy and approach for each sprint.
* Test cases: These are specific scenarios that are used to test the functionality of the application.
* Defect reports: These documents are used to report any defects or issues that are identified during testing.

1. **Deployment phase:**

* Deployment plan: This document outlines the steps that are necessary to deploy the application to production.
* User manuals: These documents provide instructions on how to use the application.

1. **Maintenance phase:**

* Bug reports: These documents are used to report any defects or issues that are identified in the application after deployment.
* Change requests: These documents are used to request changes or updates to the application.

For the appointment booking software application, specific deliverables could include wireframes for the user interface, user stories for specific features such as appointment creation and editing, code for the application itself, and user acceptance tests to ensure that the application meets the requirements (Dennis et al., 2018). Additionally, a deployment plan and user manuals may be necessary for the deployment and maintenance phases of the project.

## **Product Deliverables**

Product Deliverables are the tangible items produced during the SDLC that will be delivered to the customer. These items are what the customer will use to evaluate the progress of the project and ultimately determine if it meets their needs.

1. Wireframes: A wireframe is a low-fidelity, rough representation of the user interface of the software application (Bennett & McRobb, 2010). It is used to provide a basic visual representation of the application and to ensure that the layout and functionality are aligned with the customer's requirements. For an appointment booking software application, an example of a wireframe could be a basic layout of the appointment-booking screen, showing the fields for appointment date and time, location, and contact information.
2. Mock-ups/Layout: These are high-fidelity designs that typically do not contain any functionality (Dennis et al., 2018). They are used to give the customer a more realistic idea of what the finished product will look like. A mock-up can help to ensure that the design and layout of the application meet the customer's requirements. For an appointment booking software application, an example of a mock-up could be a detailed design of the appointment-booking screen, showing the exact placement of each field and button.
3. Prototype: A prototype is a working model of the software application. It is used to allow the customer to test the functionality of the application and provide feedback on any issues or changes they would like to see. A prototype can help to ensure that the software application meets the customer's needs and is functioning correctly. For an appointment booking software application, an example of a prototype could be a fully functional version of the appointment-booking screen, allowing the user to book appointments and test the system's functionality.
4. Requirements document: The requirement document outlines the functional and non-functional requirements for the application. This document is crucial because it helps to ensure that the development team and the client are on the same page about what the application should and should not do.
5. User manual: This document provides instructions and guidance to users on how to use the software application effectively (Dennis et al., 2018). The user manual should be written in a clear and concise language and should include screenshots and diagrams to help users understand the application's features and functions.
6. A training plan: A training plan is another product deliverable that can be created to help users learn how to use the application. The training plan should be designed to meet the needs of the users and should include both online and offline training options.
7. Software application is the ultimate product deliverable. This is the tangible output of the entire development process and includes all the features and functionalities specified in the requirements document. The final software application should be thoroughly tested and validated to ensure that it meets all the client's needs and requirements.

# **Implementation**

The implementation of the appointment booking software application will involve several stages. First, the application will be developed and tested in a non-production environment to ensure that it is stable and meets all of the required specifications. Next, the application will undergo validation and verification to ensure that it is functioning properly and meets the customer's requirements.

Once the application has been validated and verified, it will be deployed to the production environment. This will involve coordinating with the customer to ensure that the timing is right and that there will be minimal disruption to their business operations (Dennis et al., 2018). The deployment process will also involve testing and verification to ensure that the application is working as expected in the production environment.

During the implementation process, several personnel will be involved in different capacities. The Project Manager will oversee the entire process and ensure that all tasks are completed on schedule. The development team will be responsible for coding and testing the application, while the testing team will be responsible for ensuring that the application meets all requirements and is free from defects.

The customer will also be involved in the implementation process, providing feedback and input throughout the process. Finally, the IT department will be responsible for deploying the application to the production environment and ensuring that it is properly configured and integrated with existing systems

# **Validation and Verification**

To prove that the appointment booking software application functions sufficiently well to meet the customer's needs, a combination of testing methods will be used. First, unit testing will be performed by the development team to ensure that each component of the application works correctly (Dennis et al., 2018). Integration testing will also be conducted to ensure that the different modules of the application work seamlessly together.

The testing team will then conduct system testing to ensure that the entire system functions as expected. This testing will include verifying that all features are functional and that data is being stored and retrieved correctly. The customer will be involved in the acceptance testing process, which will ensure that the application meets their needs and expectations.

The acceptance testing will be conducted by the customer, with support from the development team. The testing will be performed in a test environment that mirrors the production environment. The customer will create test cases and test scenarios that will be used to validate the functionality of the application. Any defects found during the testing process will be logged and tracked until they are resolved.

To ensure that all requirements are met, a requirements traceability matrix (RTM) will be created. The RTM will link each requirement to specific test cases, which will ensure that all requirements are tested and validated. The testing team will also document test results and create test reports to track the progress of testing and ensure that all test cases have been executed.

Overall, the testing process will involve collaboration between the development team, testing team, and customer to ensure that the application meets all requirements and functions as expected.

# **Environments and Costs**

## **Programming Environment**

The programming environment required for the appointment booking software application includes the following:

**Hardware:**

• A computer or laptop with a minimum of 4GB RAM and 500GB hard disk space

• An internet connection with a minimum speed of 5 Mbps

• A printer for printing appointment schedules and reports

**Software:**

• Operating System: Windows 10 or later version

• Database Management System: Microsoft SQL Server 2016 or higher

• Programming Language: C# or Java

• Development Environment: Visual Studio 2017 or higher

• Web Server: Internet Information Services (IIS) 10 or higher

• Web Technologies: HTML5, CSS3, JavaScript, jQuery, AJAX, Bootstrap 4 or higher

• Third-Party Libraries: FullCalendar, DataTables, toastr, Moment.js, etc.

These hardware and software requirements will be used by the development team to build, test and deploy the appointment booking software application.

## **Environment Costs**

Based on the information provided, the costs associated with the software application include:

1. Database maintenance fee: $500 per year: This fee covers the maintenance of the database, including unlimited storage size and 99.8% uptime.
2. Web server fee: $300 per year: This fee covers the maintenance and upgrades of the web server, including Windows Server, IIS, and ColdFusion.
3. Device fee: $40 per device per year: This fee covers the upgrades of the operating system and network for each device that is attached to the network.

It is important to note that the final cost will depend on the number of devices that are connected to the network, as each device will incur an additional $40 fee. The costs associated with the software application are relatively low, as the environment where the system resides is a shared environment where costs are shared by the organizations.

## **Human Resource Requirements**

Based on the scope and complexity of the project, the human resource requirements will include a team of developers, a project manager, a business analyst, and a quality assurance specialist. The estimated time for the completion of the project is 6 months, with a total cost of $67,200.

The breakdown of human resource requirements is as follows:

* Developers: The development team will consist of 5 full-time developers who will work on the project for 6 months. Each developer will work an average of 40 hours per week at a rate of $30 per hour, resulting in a total cost of $28,800.
* Project Manager: The project manager will oversee the project and ensure that it stays on schedule and within budget. The project manager will work an average of 20 hours per week for 6 months at a rate of $20 per hour, resulting in a total cost of $9,600.
* Business Analyst: The business analyst will work with the customer to gather requirements and ensure that the application meets the needs of the customer. The business analyst will work an average of 20 hours per week for 6 months at a rate of $20 per hour, resulting in a total cost of $9,600.
* Quality Assurance Specialist: The quality assurance specialist will test the application and ensure that it meets the customer's requirements. The quality assurance specialist will work an average of 20 hours per week for 6 months at a rate of $40 per hour, resulting in a total cost of $19,200.

The total cost for labor is $67,200.

# 

# **Project Timeline**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Phase | Milestone/Task | Deliverable | Description | Dates |
| Pre-development | Task 1 | Requirements | Meeting with customer and procedure review | 3/1/2023 – 3/10/2023 |
| Design | Task 2 | Low fidelity wireframe | Develop a basic representation of the application's layout | 3/10/2023 – 3/15/2023 |
| Design | Task 2 | High fidelity mockup | Develop a detailed representation of the application's layout and design elements | 3/16/2023 – 3/31/2023 |
| Development | Task 3 | Prototype | Create a functioning but incomplete version of the application | 4/1/2023 – 4/3/2023 |
| Development | Task 4 | Beta version | Complete the application and release to select group of users for testing and feedback | 4/4/2023 – 4/6/2023 |
| Testing | Task 5 | Internal testing | Conduct rigorous testing and troubleshooting to ensure application is functioning properly | 4/16/2023 – 4/30/2023 |
| Testing | Task 6 | External testing | Release application to a larger group of users for testing and feedback | 5/1/2023 – 5/15/2023 |
| Deployment | Task 7 | Final release | Release the fully functional application to the customer | 5/16/2023 5/30/2023 |
| Maintenance | Task 8 | Ongoing support | Provide technical support and address any issues that arise after release | Starting 6/1/2023 |

References

Bennett, S., & McRobb, S. (2010). Object-Oriented Systems Analysis and Design Using UML. McGraw-Hill Education.

Dennis, A., Wixom, B. H., & Roth, R. M. (2018). Systems Analysis and Design. John Wiley & Sons.