C868 – Software Development Capstone

Task 2 Section A

# By Gerald Seth Scheller

**Business Problem**

The Customer

The customer is Fictitious Company LLC, a multinational company who provides resume consultations for various customers. Currently, the company resides in five different countries. the consulting company currently employs 150 employees, ranging from assistants, junior consultants and up to the CEO. The consulting company currently averages 1000 appointments at any given time. There has been an increase in the number of customer complaints that the company has had to reschedule or has missed appointments. After reviewing the current process on how they were tracking customer appointments it was determined that process was the issue.

Business Case

Currently, the company receptionists take the phone or check the company inbox for appointments and then added those appointments to an excel spreadsheet. The consultants then pull from that spreadsheet and add it to their own calendar. This process has led to consultants overbooking themselves, and/or missing appointments. There are not checks or balances and no one to help keep track of appointments if the consultant are out of the office and had an appointment. It has become apparent that the company needs a more robust and scalable solution for scheduling appointments.

The appointment scheduler will provide a consistent and centralized location to house appointments. The application will use a robust, relational database capable of handling the current load and allow scalability. The application will allow for each employee to view existing appointments and allow for transparency.

Fulfillment

The scheduling application will be programmed with the Java programming language. The application will be utilized in a virtual environment called the Java Virtual Machine. This allows the company to continue to utilize their current hardware and operating systems. The only requirement will be to install the Java runtime environment on the computers.

The scheduling application will use a MySQL database to store the data. The database will allow for additions, deletions, and modification directly through the application.

The scheduling application will be able to save all customers and appointments. It will allow easy tracking and modifications to appointments and customers. The flexibility, centralization, access, and uniformity the application provides will alleviate the scheduling issues the company is currently having.

**Existing Gaps**

Fictitious Company LLC is experiencing many problems with their current method of tracking appointments. The consultants and assistants are responsible for managing their appointments and manually doing that with whichever system choose. This has led to overbooking of appointments, missed appointments and customers being forgotten about.

The current process for managing appointments can be described disorganized and chaotic. A combination of notebooks, journals, excel documents and other various software applications are utilized to handle this important information. While notebooks and journals can be used to help keep an individual on track to a degree, it is not accessible to other in the office. In a couple different instances, a consultant got sick and was unable to come to work. All the appointments the consultant had were in their notebook. They had that notebook locked in their office drawer. Needless to say, those appointments got missed and the customers weren't notified they would have to be rescheduled. Excel is a widely used software application, however it has its limitation in a large-scale operation. Its ability to handle relational data, the inability to allow multiple users to be editing a document or appointments simultaneously to name a couple things.

**SDLC Methodology**

The initial application will be created by using the Waterfall methodology. The reason we have chosen this method is because the applications features are clearly defined at upfront. This application is frequently requested by our company, and we have a solid foundation to build upon to complete this application in a short amount of time.

The first phase of the Waterfall method begins with the requirements phase. This phase is used to understand the client’s current needs. We have met with Fictitious Company LLC and created a company specific requirement sheet. This document details the specific needs the system will need to perform to satisfy the client's request.

The second phase is the system design phase. In this phase the project team will design the system. There is no coding in this phase. The team will decide on the programming language, database diagrams, deployment, and testing strategies. The team will create a low fidelity wireframe to demonstrate the flow of the application, a high-fidelity prototype that will demonstrate the UI and database models. These items will be created using the requirement sheet we created in in the requirements phase.

The third phase is the implementation/coding phase. The approved prototype from the previous phase will be turned into code. The application will transition from idea to reality. The UI will be designed according to the prototype. The database will be created and formatted. All functionality that was listed in the requirements sheet will be programmed. The output from this phase is a functional software application that meets the documented requirements.

The fourth phase is the testing phase. Now that we have a functional application, we need to test the system. The first round of testing is called the alpha test and is performed by the development team. The next test is the beta test. This test will take place with a select group of employees at Fictitious Company LLC. During these testing phases, the groups will document any bugs or issues they encounter. The final step in the testing is the user acceptance test. This test will be used to garner the approval of the employees and ensure that all required items were provided.

The fifth phase is the deployment phase. Now that all the bugs have been worked out and the system meets the requirements defined by the customer, we need to put this into the live environment. All computers should have the latest version of the Java runtime environment and access to the installed version of the application on their computers.

The sixth and final phase is the maintenance phase. This phase is post deployment and allows the client to turn to the development team if any issues arise. Maintenance can include newfound bugs, system patches and new updates. A contract should be created to outlines ongoing services.

**Deliverables**

Project Deliverables

* + Requirements document
    - Covers the applications required functionality.
    - Will be used throughout all stages of the project.
    - Details about UI, functions, data structures, constraints, and limitations.
  + Project schedule
    - This will detail the timeline of each phase.
  + Wireframes
    - Mockup of the entire application to demonstrate the flow.
    - Does not have details or usable data.
    - Will clearly show application navigation and ways to interact with the application.
  + UML Class and Entity Relationship Diagram
    - Class diagrams that show each class, variables, functions and how they interact.
    - The Entity Relationship diagram which shows the information for each table in the database and how they are related.
  + Testing plan
    - Testing will be performed by the development team and a select group of employees from Fictitious Company. Each step will test functionality and have clearly defined inputs, outputs to ensure expected outcomes are produced.

Product Deliverables

* + Functional database and scalable database
  + Fully developed application which meets the client's requirements document
  + Secure login access for users and admin.
  + A standard operation procedure on how to use the system.
  + A simple and intuitive user interface.

**Implementation**

The implementation of this project is expected to go very quickly and smooth. During the testing phase the selected staff will be trained have access to the system before all others. This will allow for a few employees who can be utilized as power users. Between the power users and the operation guide there will be several resources for system navigation questions.

The integration of the software component does not require a specific operating system. It only requires the installation of the Java runtime environment. Our staff will assist with the installation as necessary prior to the scheduling applications release.

**Validation and Verification**

We will ensure a testing plan is in place and the application meets all the requirements outlined in the requirements documents as well as provides a user-friendly interface. In addition to writing code, our developers run their own unit tests. We will run function tests. These tests will be used to ensure the system performs all the specified required actions that were identified in the requirements document. The final testing can only be performed in the real world with real information. The acceptance testing will validate that the product meets the company needs and is usable in day-to-day tasks. These tests will determine if the application is suitable and ready for full scale deployment.

**Environment and Costs**

Programming Environment

The scheduling application will be written in the Java language. Java is an object-oriented programming language. The company will be using a Java Runtime Environment to run the application. This is being done so that we don’t need to specify the operation system the employees are using. The data will be stored on a MySQL database and will be house on the company’s server.

Environment Costs

Java Runtime Environment 8 or newer: No Cost

HP 748303-s01 Proliant G8: $6200.00

Microsoft SQL Server: $589.00

Human Resource Requirements

In terms of human resources, the project requires a project manager, a designer, two software developers, and a quality assurance specialist. The program manager and developers will do a majority of the work on this project. The designer will be involved heavily up front in the design with the quality assurance specialist being more heavily involved at the end during the testing.

|  |  |  |
| --- | --- | --- |
| **Resource** | **Rate \* Time** | **Total** |
| Project Manager | $60/hr \* 50h | $3000.00 |
| Designer | $40/hr \* 20h | $800.00 |
| Software Developer | $50/hr \* 50h \* 2 | $5000.00 |
| QA Specialist | $30/hr \* 30h | $900.00 |
| **Total HR cost** |  | $9700.00 |

**Environment and Costs**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Phase** | **Task** | **Deliverable** | **Description** | **Dates** |
| Requirements | Determine requirements | Requirements document and project schedule | Meet with customer to determine system requirements. | 07/10/22-07/11/22 |
| System Design | Wireframe | Low Fidelity Wireframe | Wireframe gets created and approved. | 07/12/22-07/12/22 |
| System Design | Prototype | High Fidelity wireframe/  prototype | Prototype created and presented for approval. | 07/12/22-07/12/22 |
| System Design | Diagrams created | UML Class and ERD database diagrams | Diagrams created to show the customer the relationships and how they work with the application. | 07/13/22-07/14/22 |
| Coding | Application is coded | Function application | Developers will program the application according to the customer requirements. | 07/15/22-07/22/22 |
| Testing | Alpha test/beta test | Testing plan for alpha and beta | Specialist creates and runs testing to very all application requirements and functionality | 07/25/22-07/28/22 |
| Testing | Acceptance testing | Customer conducts acceptance testing | The customer will test the application to verify satisfaction and requirements. | 07/29/22 |
| Deployment | Prepare Deployment | SOP and software preparedness | Ensure server is ready, runtime environment and application are installed. Deliver training | 08/01/22-08/02/22 |
| Maintenance | Set up Mainenance plan | Maintenance service contract | Finalize the terms of service for the application maintenance. | 08/03/22-08/04/22 |