

C868 – Software Capstone Project Summary

Task 2 – Section A



Capstone Proposal Project Name: Appointment Scheduler for Local College

Student Name: Ned Imhoff 005917082

Table of Contents

Business Problem	3
Business Case	3
Fulfillment	3
Existing Gaps	3
SDLC Methodology	4
Deliverables	4
Project Deliverables	4
Product Deliverables	4
Implementation	4
Validation and Verification	5
Environments and Costs	5
Programming Environment	5
Environment Costs	5
Human Resource Requirements	5
Project Timeline	6

Business Problem

The Customer

Northern Metro College (NMC) is a small liberal arts community college. They have approximately 100 faculty and 2,000 undergraduate students. Currently, the teaching faculty have been responsible for certain administrative tracking such as student appointments, grades, and curriculum development.

The college is preparing to streamline student/faculty appointments to improve faculty and student experience by investing in an administrator appointment application manager. Northern Metro College believes this investment will support the college's mission of delivering high quality education for their students by standardizing the appointment process thus making it more consistent and reliable. The goal of this is to improve faculty and student appointment experience.

Business Case

The proposed software will funnel all student/faculty appointments through one source, the administrator application manager.

Prior to the application manager solution, faculty were responsible for scheduling student appointments. These appointments were documented by the faculty member in various forms from excel documents to google calendars to hand written planners. The college was receiving feedback from faculty noting their frustrations with this responsibility, and student feedback noted problems with faculty availability and reliability. This feedback made it appear as though some faculty appointments were consistently full, some faculty were poor communicators, and some appointment times were inaccessible to students.

The application manager will solve these problems. All appointments will be scheduled by the administrators interacting with the application software. This will take the responsibility of scheduling appointments off the plate of the faculty and it will ensure reliability for students. All appointments will be saved in a centralized database. This will allow for analysis of appointment trends to help the college make decisions for improving student/faculty interaction and experience. The software is a tool to store all appointments in one place.

Fulfillment

The programming of the application will use the Java programming language, operating within the Java Virtual Machine (JVM), a virtual environment. This feature enables Java programs to execute on various operating systems such as Windows 10, MacOS, and Linux.

The appointment scheduler possesses the capability to securely store and modify all student and appointment data to a MySQL database from the application main screen. There is also functionality that

Appointment Scheduler for Local College

allows administrators to create new student profiles, and update existing profiles. There will be a Reports section that will display appointments by faculty and by student when searched.

The MySQL database uses the SQL language for executing database queries and managing information storage. Following the initial database setup, no direct maintenance is required. However, it remains adaptable for potential expansions necessitated by the client's growth or new goals. The appointment scheduler manages all communication to and from the server, ensuring the protection of all data from unauthorized tampering.

Existing Gaps

Existing systems of appointment collection was documented by faculty using excel spreadsheets, google calendars and hand written planners. These tools were independently managed by the faculty and there was no ability for the college to see appointments except by contacting the faculty member directly. This process led to lengthy time spent gathering appointment data and the reliability of the data was low due to unstructured documentation processes.

SDLC Methodology

The Waterfall method will be used for this project due to its suitability for the short timeframe. Appointment scheduling applications are commonly requested, therefore past plans and elements can be adapted for the client's needs. In the Waterfall method, each phase must be approved before moving on.

Requirements (1 day): Meet with client to define project functionalities. This sets the project scope and expectations, documented in a Software Requirement Specification (SRS) document.

Analysis (1 day): Review the SRS, create the project schedule, and ensure agreement between the development team and client before moving to the design phase.

Design (3 days): Create a functional design based on the SRS. Plan the user interface (UI) using wireframes, turning them into a prototype for the client's approval.

Coding (8 days): Translate the approved prototype into source code, coding the UI and creating/formatting the database. Perform unit testing for each completed unit.

Testing (5 days): Rigorously test the application, starting with alpha testing by the development team, followed by beta testing with the client's faculty. Resolve any issues found, then conduct the acceptance test with the client.

Deployment (2 days): Install the application on client administrators computers. The installation is platform-independent but requires the most recent Java Runtime Environment (JRE).

Maintenance (2 days): Handle any unforeseen issues or bugs post-deployment to ensure a smooth experience for the client.

Deliverables

This section will detail the deliverables associated with the development cycle. These deliverables are categorized into two groups: Project Deliverables and Product Deliverables.

Project Deliverables

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

Project Schedule:

- An overall timeline is outlined in the SDLC Methodology section. This document will provide more detailed timelines for each phase. Due by the end of the analysis phase.

Software Requirements Specification (SRS):

- This document details the application's functionality and serves as the main guidance throughout the project. It defines the project's scope and resolves problems. Due by the end of the analysis phase.

Wireframes:

- Low-fidelity representations of the application's UI and functionality. They lack functionality and are due during the design phase before prototyping.

Prototypes:

- Developed based on wireframes, prototypes have navigational functionality but lack logical functionality. They allow for a preview of the application, with the flexibility to quickly change features. Data in the prototype is placeholder material. Due during the design phase after wireframe approval.

UML Class Diagrams and Entity Relationship Diagram (ERD):

Appointment Scheduler for Local College

- Class diagrams depict the content of each class in terms of variables and functions, as well as interactions between classes. The ERD illustrates the information in each database table and their relationships. Due by the end of the design phase.

Testing Plan:

- Testing by the development team and client will follow a plan with specific instructions. Due during the testing phase.

Product Deliverables

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

- Due during deployment phase:
 - Wireframe
 - UML Class Diagram
 - Entity Relationship Diagram
 - Intuitive graphical user interface
 - Secure login
 - Application capable of creating, updating, and deleting students
 - Application capable of creating, updating, and deleting faculty
 - Application capable of creating, updating, and deleting appointments
 - Ability to filter appointments by week, month, or all, and by faculty
 - Ability to search through customer name and appointment title via user-entered text
 - Scalable database
 - User Guide for application operation
- Due during maintenance phase:
 - Maintenance Guide

Implementation

The appointment scheduler software needs a Java Runtime Environment (JRE) to be installed on the computers of the administrators using it. A team will help install the JRE before setting up the appointment scheduler software. Within two business days, three computers can be prepared as well as training for the administrators to use the software.

Validation and Verification

Testing will involve a thorough full lifecycle examination to ensure the application meets the specified requirements. The development team will conduct the initial round of testing using white box testing to validate each function. If any issues are detected, they will be promptly addressed. The second round of testing will involve client administrators, who will perform specific tasks with clear instructions to assess the application's functionality. Any identified issues will be resolved before the final round of testing. The last phase, the Acceptance Test, will be conducted by the client owner and the administrator staff. This test is expected to be error-free, with testers freely using the application to verify that it fulfills business needs and adheres to the requirements in the SRS. The Customer will perform Acceptance Testing before taking ownership of the application, ensuring its readiness for production.

Environments and Costs

Programming Environment

The application will be built using the Java programming language, which works on any operating system as long as Java Runtime Environment version 17 is installed. No additional upgrades or installations are needed. The data will be stored on a MySQL database hosted by Google Cloud service.

Environment Costs

Estimated database cost using Google Cloud (supports MySQL): \$ 10.00 per month

Human Resource Requirements

The project team includes one project manager, one software designer, two software developers, and one quality assurance specialist. The project manager and developers will invest the most time in the project. The quality assurance specialist will contribute to testing and train On-Track Recovery staff. The designer will be involved only during the design phase. The breakdowns for time and costs are as follows:

Appointment Scheduler for Local College

Resource	Rate * Time	Total
Project Manager	\$60/h * 40h	\$2,400
Designer	\$40/h * 24h	\$960
Software Developers	\$70/h * 128h * 2	\$17,920
QA Specialist	\$40/h * 40h	\$1600
TOTAL Human Resource Cost		\$22,880

Project Timeline

Phase	Milestone/Task	Deliverable	Description	Dates
Requirements	Task 1 / Define Requirements	Software Requirements and Specifications document	Meeting with customer to identify software needs	12/4/23
Analysis	Task 2 / Finalize schedule	Project timeline	Share finalized schedule with client. Authorize design stage.	12/5/23
Design	Task 3 / Wireframe	Low Fidelity Wireframe	Finish wireframes	12/6/23
Design	Task 4 / Prototype	High Fidelity Prototype	Present prototype for client approval	12/6/23 - 12/7/23
Design	Task 5 / Complete Diagrams	ERD database diagram and UML Class Diagram	Diagrams shared with client to demonstrate application relationships	12/7/23-12/8/23
Coding	Task 6 / Code all parts to application	Software Application	Following the SRD document, UML and ERD diagrams, software developers will program the application.	12/11/23-12/20/23
Testing	Task 7 / Alpha Testing	Testing Part 1	Developer team will complete round 1 of testing. Results	12/21/23-12/22/23

Appointment Scheduler for Local College

			will be delivered to the client.	
Testing	Task 8 / Beta Testing	Testing Part 2	Client administrator staff will test part 2 of testing.	12/27/23-12/28/23
Testing	Task 9 / Acceptance Test	Acceptance approval	Client reviews the completed application with SRS document.	12/29/23
Deployment	Task 10 / Application installation	Functioning Application and user instructions	Connect cloud server. Install JRE to computers. Install application. Train administrative staff.	1/2/24-1/3/24
Maintenance	Task 11 / Maintenance Plan	Maintenance Instructions	Provide maintenance directions and discuss ongoing maintenance with client	1/3/24-1/4/24