C868 – Software Capstone Project Summary

Task 2 – Section A

Capstone Proposal Project Name: Appointment Ace – An Appointment Scheduler

Student Name: Adam Wright

Table of Contents

Contents

Business Problem

The Customer

The Wright Consultants (TWC) is a leading global consulting firm with 100 employees across offices in New York, Wisconsin, Michigan, Seattle (USA), and Toronto (Canada). They serve a large client base of approximately 100,000 customers annually. TWC is experiencing growth and anticipates further expansion soon. The current software is outdated and lacks scalability, hindering TWC's ability to support its growing client base and employee count. The outdated interface will impact the user experience for internal staff and external clients.

Business Case

The proposed solution is implementing a modern all-in-one Java application to manage customer appointments. It will leverage a MySQL relational database with normalized tables to ensure data integrity and efficient querying. The application will enable tracking appointments associated with specific customer contacts, and comprehensive management of customer appointment records. Java offers scalability and platform independence which is crucial for future growth. MySQL provides a robust and secure foundation for data storage, and normalized tables optimize database performance and minimize inefficient queries. This solution is designed to address the scalability and data integrity challenges TWC has been experiencing.

Fulfillment

This application will be a native desktop application developed in Java. Trained technical staff at the TWC will utilize the application. A login page with robust authentication will be implemented to safeguard customer and contact information. The login form design will adhere to the best practices to prevent SQL injection attacks. The creation of new contacts will be disabled by default to maintain data integrity. Authorized personnel can be granted access through user privilege management. There will be three dedicated reporting sections that will be integrated into the application, enabling users to manage customer and appointment data with improved reliability and efficiency.

The first section allows users to filter and view customer appointments based on appointment type and month. Appointments will be displayed in a user-friendly tabular format for easy analysis. The application will automatically calculate and display the total number of appointments for each unique category (type and month combination). The Modify Appointment functionality will provide an intuitive interface to facilitate the editing/updating of information on customer appointments.

The second section will be designed to address contact schedule management. The contact schedule view focuses on facilitating improved contact time management by providing a comprehensive schedule. A list of contacts will be presented, along with their associated appointment schedules. Each appointment within the report will include the following attributes: Appointment ID, Title, Type, Description, Start Time, End Time, and Customer ID.

The final section will be designed to address Contact appointment hours management. This section will provide insight into the total appointment hours for each contact. The appointments will be linked to specific contacts and corresponding customer data. The application will calculate and display the total duration (in hours) of appointments scheduled for each contact, presented in a user-friendly format. There will be date validation rules implemented to prevent overlapping appointments for contacts and customers and appointment scheduling outside of pre-defined business hours.

MySQL, a robust relational database management system (RDBMS), will be leveraged for data storage. Optimized table structures adhering to normalization principles will be implemented to ensure efficient data retrieval and manipulation. This design prioritized TWC's performance requirements. The application will incorporate robust security mechanisms to safeguard sensitive customer and contact information within the database. These measures will mitigate potential malicious attacks and guarantee data integrity.

Existing Gaps

Current system challenges experienced by users and technical staff at TWC have reported encountering scheduling errors and missed notifications. These issues are suspected to be caused by software bugs within the existing system. These errors have resulted in missed appointments by TWC employees, negatively impacting their ability to serve customers effectively. The unreliable scheduling system disrupts the structured work environment needed for optimal performance.

The current software solution utilized by TWC has not undergone any updates since the company's inception ten years ago. This lack of updates has resulted in features becoming outdated or reaching their end-of-life stage. These features compromise the system's overall reliability and security posture. To sustain TWC's growth trajectory, a complete system redevelopment will leverage modern software development practices and technologies to ensure a robust and secure platform.

The current system lacks the precision and organization required for effective TWC customer appointment tracking. This inefficiency contributes to user difficulties. A native Java application will be developed to manage all TWC's data within a single, secure location. This application will leverage a centralized MySQL relational database with a user-friendly interface. Technical staff will gain the ability to perform CRUD (Create, Read, Update, Delete) operations on customers and appointment records with ease, security, and efficiency. The streamlined data management will significantly reduce the potential for errors and minimize memory usage.

SDLC Methodology

The Agile development methodology has been chosen for this project due to its iterative and user-centric approach. This methodology employs a series of short development cycles (sprints) that deliver working functionality incrementally. This allows for continuous adaptation based on feedback and evolving requirements. Agile development accommodates the potential for changing needs throughout the project lifecycle. User testing within each sprint allows for client feedback and potential feature adjustments or additions. This approach is well suited for TWC’s project because core challenges are well-defined and with the potential for evolving requirements user feedback during the development process may necessitate adjustments or new functionalities.

During the initial phase (concept phase) the product owner will lead efforts to establish the project's overall vision and scope. The key project requirements will be identified and documented through collaborative discussions with stakeholders. This documentation will serve as a foundational reference throughout the development process. The product owner will conduct a thorough analysis to estimate the project time and cost requirements This assessment is crucial for determining project feasibility and securing necessary approvals. This phase culminates in the creation of key project documents: The project vision document which outlines the project goals and objectives, The sprint backlog which is a prioritized list of features to be developed in iterative cycles (sprints) and The Dynamic Product backlog which is a comprehensive list of all project requirements, subject to refinement and ongoing prioritization as the project progresses.

Following the concept phase is the inception phase. In this phase, the project team will be assembled. This may involve the product owner sourcing and hiring necessary talent which will consist of developers and designers. The product owner will ensure the team has access to all essential tools and resources required to begin development activities effectively. The UX/UI designers will commence the development of a UI mockup to visualize the application's look and feel. Concurrently, project architects will define the overall system architecture to establish a foundation for development. This phase focuses on creating the initial design artifacts to facilitate decision-making by providing supplementary documentation to aid development efforts. A low and high-fidelity wireframe will be designed enabling stakeholders to provide feedback on design decisions.

Following the inception phase is the iteration phase. The project will leverage an iterative development cycle known as sprints focusing on delivering a specific set of functionalities based on prioritized requirements and user feedback. During each sprint, the development team works closely with UI/UX designers to translate the product requirements and customer feedback into functional code. By the end of each sprint, a potentially shippable product increment will be delivered. This increment represents a functional portion of the overall application. Throughout the development process, the produced software undergoes rigorous quality testing (QA) testing to ensure quality code and adherence to requirements.

The testing phase focuses on rigorous testing to ensure the developed software increment meets all quality standards. The first activity includes creating comprehensive test plans that outline the testing strategies and procedures. The next activity includes performing thorough testing to verify all functionalities aligned with sprint backlog features and user requirements. Compatibility across various platforms and environments as specified in the project scope. Following successful testing and bug resolution, a controlled release of the developed software increment will be implemented into production. The main deliverables in this phase are the test plan which will be a comprehensive document outlining the testing strategy, scope, and procedures, and user documentation which will include clear and concise user guides to facilitate application adoption and user experience.

Upon successful completion of the testing phase, the production phase will begin. In this phase, a deployment of the software application will be executed to transition into the production environment. This ensures the software becomes fully operational and accessible to authorized users. Following deployment, the development and/or designated support team will transition into the maintenance phase. During this phase, the team will focus on maintaining application stability and reliability and addressing reported defects.

In the review phase, a comprehensive project review will be conducted by the product owner and project team members. A progress assessment is conducted to evaluate the project's overall progress against initial goals and objectives. Lesson-learned documentation is produced in this phase and key project learnings are captured, including both successes and encountered challenges. This documentation serves as a valuable resource for future iterations and continuous improvement. Based on the review findings and ongoing stakeholder feedback which is solicited by the product owner, the project team will initiate planning for the next development iteration. This iterative approach allows teams to leverage lessons learned and incorporate valuable stakeholder insights to optimize future development efforts.

Deliverables

While Agile methodologies emphasize a focus on delivering value over extensive documentation, essential artifacts are still produced throughout the development lifecycle to ensure project transparency, communication, and maintainability. These artifacts can be broadly categorized into two main groups which will be in the following section.

Project Deliverables

- Project Vision
    - A concise description of the project’s overall goal and desired outcome.
    - Clearly outlines the organization’s aspirations for the project’s impact.

- Sprint Backlog
    - A prioritized list of work items the team plans to complete during a specific development sprint.
    - Derived from the product backlog during sprint planning sessions.
    - Assign ownership of each work item to prevent scope creep.
- Product Backlog
    - A prioritized list of work items for the development team, derived from the project roadmap and its requirements.
    - The most important features are listed at the top, guiding the team on delivery priorities.
- Support Documentation
    - Training materials, troubleshooting guides, templates, and examples designed for staff use.
    - Written in a clear unambiguous manner for user understanding.
- Low-Fidelity Wireframe
    - A basic design blueprint for UI screens.
    - Focuses on core functionalities and the application's overall flow.

- Lowers emphasis on visual detail but emphasizes navigation and user interaction.

- High-Fidelity Prototype
    - A more detailed and interactive prototype built upon the finalized wireframe.
    - Showcases most major features and design elements, focusing on the application's look and feel.
- Testing Plan
    - A comprehensive plan outlining unit testing, integration testing, system testing, and acceptance testing procedures.
    - Ensures the application's functionalities meet requirements.
    - Testing methodology (Manual or automated) and expected results should be documented.
- User Documentation
    - Content aimed at educating users on successful application use.
    - Also known as instruction manuals, user guides, or user manuals.

Product Deliverables

- The application must deliver all functionalities detailed in the comprehensive requirements documents. This includes:
    - Core features essential for user workflows, verified through unit testing.
    - Ancillary features that enhance user experience and system capabilities, verified through integration testing.
    - User acceptance testing will confirm successful implementation from an end-user perspective.
- The application UI flow and navigation must demonstrably mirror the established wireframe layouts. This ensures:

    - User experience aligns with initial design concepts and user expectations.
    - Deviations require documented justification and potential user testing to assess usability impact.
- The final application’s UI design must closely resemble the previously approved high-fidelity prototype. This guarantees:
    - A visually cohesive and consistent user experience across all functionalities.
    - Any deviations require documented justification and potential user testing to assess user experience and brand alignment impact.
- The application must incorporate a robust login mechanism that enforces secure user access, including:
    - User authentication protocols with appropriate password complexity requirements.
    - Role-based access control to restrict functionalities based on user permissions.
    - Secure data transmission protocols to protect user credentials and sensitive information.

Implementation

To minimize disruptions to TWC’s core operations, the project implementation will be strategically scheduled outside of regular business hours. This out-of-bounds deployment window ensures reduced impact on user productivity and allows for enhanced control over the deployment process itself. TWC will actively participate in the design validation stages through acceptance and usability testing. This proactive approach facilitates the early identification and resolution of potential usability issues, ultimately leading to a more user-friendly application. Furthermore, TWC’s involvement in the testing phases fosters user confidence and familiarity with the deployed application due to prior exposure and active participation. By deployment, TWC will possess a solid understanding of the application's functionalities and user interface through comprehensive testing participation.

The application’s production deployment signifies the completion of the development phase. However, user access requires further configuration. In collaboration with TWC’s IT department, user accounts will be provisioned within the application, potentially leveraging existing directories or implementing a separate user management system. Subsequently, role-based access control will be established based on user permissions. The project team will collaborate with TWC to develop user training materials. Once these post-deployment activities are finalized, controlled user access will be granted, enabling TWC staff to leverage the application's functionalities.

Validation and Verification

Our agile methodology emphasizes quality assurance practices throughout the development lifecycle. A comprehensive test plan will be established to ensure the applications align with customer requirements as outlined in the requirements documentation. Unit testing will be created by a software engineer alongside code development. A unit test verifies the functionality of individual software methods and modules in isolation. This granular approach facilitates early identification and resolution of coding errors.

Following successful unit testing, integration testing will validate how different software components interact and exchange data. This ensures seamless system functionality. To confirm the application effectively addresses customer needs, User Acceptance Testing will involve active customer participation. Through user acceptance testing will test and validate core functionalities, recommend potential changes or new features for future iterations, and provide feedback for final system acceptance.

Environments and Costs

Programming Environment

The application will developed using Java, a robust and widely adopted programming language known for its platform independence and object-oriented capabilities. FXML, a declarative UI framework for JavaFX, will be utilized to design the application's user interface. FXML promotes code separation and simplifies UI creation, facilitating efficient development and maintenance. MySQL a relational database will serve as the application's data storage solution. MySQL is an open-source database management system known for its scalability, reliability, and support for complex queries. The database will be designed following normalization principles to minimize data redundancy and improve data integrity. The application will be packaged as a downloadable installer specifically designed for Windows 10 operating systems, aligning with TWC's current IT infrastructure. Upon application launch, an automated connection will be established to the MySQL database, ensuring seamless data access for core functionalities.

Environment Costs

The application is anticipated to incur minimal ongoing costs within the production environment. The project will leverage the MySQL database with a flexible pricing structure. The initial monthly cost is estimated to be $10 based on TWC's current user base of approximately 100,000 users. TWC can monitor database usage and storage capacity. Should the database approach its maximum capacity, the project team can collaborate with TWC to explore cost-effective upgrade options and tailor the database plan to their specific needs. The annual cost for the database is $120.

Human Resource Requirements

The project will require a project manager, UI/UX designer, Software engineer, and a QA engineer. The project manager is responsible for the overall lifecycle management, including activity planning, resource allocation, team organization and motivation, adherence to timeframes, and budget estimation/cost control. The project manager estimates the labor hours to be 40 hours at a rate of $80/hour, totaling $3,200. The UI/UX designers focus on crafting a user-friendly and intuitive user interface and user experience for the application. Their involvement will be concentrated in the inception phase and design phase. Considering the scope of the project, an estimated 20 hours at a rate of $50/hour, totaling 1.000. The software engineer will be responsible for developing core functionalities of the application during the development and iteration phases. Depending on the project's complexity, an estimated timeframe of 30 hours at a rate of $70/hour, totaling 2,100. The QA engineer will be responsible for designing and executing comprehensive test plans to ensure the application's quality and functionality throughout the development cycle. Their involvement will be most prominent during the testing phase, with an estimated effort of 15 hours at a rate of $60/hour, totaling 900.

Human Resource	Bill Rate	Total Cost
Project Manager	$80/hour * 40 hours	$3,200
Software Engineer	$70/hour * 30 hours	$2,100
UI/UX Designer	$50/hour *20 hours	$1,000
QA Engineer	$60/hour * 15 hours	$900
Human Resource Total		$7,200

Project Timeline

Phase	Milestone/Task	Deliverable	Description	Dates

Concept	Finalize requirements	Project vision, Sprint backlog, Product backlog	The requirements will be discussed with the stakeholders to be defined and prioritized.	5/1/24
Inception	Design and test Low/High fidelity wireframes	Low/High fidelity wireframes	The designers will design a low-fidelity wireframe that focuses on establishing core functionalities, user flows, and navigation structure. The high-fidelity prototype will provide the final intended look and feel of the application.	5/2/24 -5/3/24
Inception	Develop support documentation	Support Documents	The project manager will oversee user support documentation development to facilitate knowledge transfer and user adoption.	5/4/24-5/5/24
Iteration	Build unit test	Application version	The development team will utilize the high-fidelity prototype and requirement documents to build a fully functional Java application according to best practices. Manual unit testing will be implemented throughout development to minimize errors.	5/7/24 - 5/8/24
Testing	Build test plan user documentation	Testing plan	The project manager will collaborate with the engineering team to establish a testing strategy. The strategy will encompass unit testing and acceptance testing. User support documentation will be developed to facilitate staff and customer adoption of the new application.	5/9/24 – 5/10/24

Testing	Build and execute acceptance test	The users execute the acceptance test.	User acceptance testing will be conducted. Authorized users will be involved in evaluating the application functionality.	5/10/24
Production	Application Deployment	The application will be delivered to TWC	Deploy to the customer after final checks.	5/11/2024
Review	Conduct Retrospective	Sprint Retrospective	Upon project completion, a facilitated project retrospective will be conducted. This session will involve the project team and key stakeholders to analyze the success and challenges of the project. This meeting will also discuss improvement planning strategies.	5/11/24