D424 – Software Capstone

Task 2

Capstone Proposal Project Name: SemesterSync

Student Name: David Ogden

Table of Contents

Business Problem

The Customer

The customer for SemesterSync is Western Governors University (WGU). WGU is an online college boasting 176,000 students, offering a variety of degrees (Western Governors University, n.d.). The fields that WGU offers degrees in are business, technology, health and nursing, and education. The mission of WGU is to expand access to education, which they exemplify by offering fully online degree programs and tuition rates roughly 50% lower than the national average (Western Governors University, n.d.).

While the customer of SemesterSync is WGU, the application's users will be WGU students. Of the 176,000 students, the average student age is 38 (Western Governors University, 2023). Further, 79% of WGU students are employed full-time in addition to their studies (Western Governors University, 2023).

Currently, WGU needs a system for students to track their progress and help them better prepare for upcoming classes and exams. This is why SemesterSync is imperative to student success and readiness.

Short-term Goals:

- Increase personalized on-time completion rate
- Increase student readiness for upcoming classes and exams

Long-term Goals:

- Expand access to education
- Establish WGU as the top online school in the United States

Business Case

For WGU, the success of their students is one of their top priorities. However, the average personalized on-time completion rate for the last five years is only 30% (Western Governors University, n.d).

SemesterSync aims to improve the personalized on-time completion rate by providing an application for students to manage their terms, classes, and exams. Students will benefit from SemesterSync because they can customize their degree path, track their progress, and be notified of upcoming classes and exams. Another benefit of SemesterSync is that the power of managing degree plans will be in the hands of the students. This will allow students to plan their degrees, better prepare for classes and exams, and give instructors more time to focus on other duties.

SemesterSync will fill the gap of not having a tracing system by offering functionalities such as:

- Term, Class, and Exam Management: SemesterSync will allow students to add, edit, delete, and associate terms, classes, and exams, ensuring students can create an accurate and customized degree plan.
- Upcoming Event Notifications: The application allows students to schedule reminders for upcoming classes and assessments. This will enable students to prepare themselves better.

- Progress Reporting: SemesterSync will come with the ability to provide robust progress reporting to students, allowing them to set milestones and measure milestone completion
- Security: SemesterSync will utilize authentication to ensure users can only access their data

Fulfillment

SemesterSync is a mobile application that will fulfill the needs of WGU by providing its students with a way to customize their degree path, track their progress, and be notified of upcoming classes and exams.

A notable feature of SemesterSync is an easy-to-use user interface (UI), ensuring only pertinent information is shown. This will enhance the application's user experience and help streamline user interaction.

SemesterSync will be developed with the .NET MAUI framework, allowing one code base to be used for multiple operating systems. Using .NET MAUI, SemesterSync can be delivered at a lower cost and is still available for most WGU students. SemesterSync will also utilize a SQLite. Using SQLite will enhance query speeds since all data will be stored locally on the user's device, resulting in a smoother user experience.

- Term Management: Students can create, update, and delete terms. Term details include the term name, start date, end date, and associated classes.
- Class Management: Students will also be able to create, update, and delete classes. Class details include class name, start date, end date, instructor information, upcoming exams, and class notes. Students will have the ability to share their class notes for collaboration.
- Progress Reporting: SemesterSync will compile class and term information into graphics showing students' progress. The graphic filters will have options for overall progress and progress by term. This ensures students always know where they stand in their degree plan and current term.
- Notification Scheduling: Students can set reminders for upcoming classes and exams, ensuring they never miss a class or forget to prepare for an exam.
- Security: Students will set their login and password upon registration and must authenticate themselves to ensure data privacy.

SDLC Methodology

SemesterSync will be developed using the waterfall software development life cycle methodology. The waterfall methodology consists of six stages, and the next stage starts when the previous stage ends. SemesterSync will benefit from the waterfall methodology because:

- The well-defined requirements allow a streamlined development cycle and a lower chance of scope creep.
- The well-defined requirements assist in making more accurate time and cost estimates.
- At the end of development, WGU will receive a fully completed application.

The six stages of the waterfall software development lifecycle are:

- Requirements: During the requirements stage, the team at SemesterSync will gather the resources needed, assign responsibilities to team members, and create a timeline for the project.
- Design: In the design stage, team members will create the wireframes, UML class diagrams, and entity relationship diagrams and determine deliverables.
- Implementation: In the implementation stage, an implementation plan will be created, data will be collected, research will be conducted, and the project will be created based on the design artifacts.
- Verification: During the verification stage, the quality assurance (QA) team will receive the project during verification. The QA team will write and execute test cases, documenting any bugs or errors that need fixing.
- Deployment: Assuming SemesterSync passed QA, the application will go live, and access will be given to WGU and its students.
- Maintenance: Once the product is released to WGU, the development team at SemesterSync will continue to provide support for the product. Maintenance activities include bug fixes, customer support, and security updates.

Deliverables

Since SemesterSync will be developed using the waterfall methodology, each stage will have its own deliverables. The deliverables by stage are:

- Requirements:
  - Requirements Documentation – Sets the requirements that the product will meet
  - Project Schedule – Timelines for each stage and work within the stages
  - Responsibility Assignment Matrix – Shows the responsibility level of each team member for each task
- Design:
  - Wireframe – A low-fidelity mockup of the user interface
  - UML Class Diagram – Shows how classes will be organized and the available methods
  - Entity-Relationship Diagram – Shows the relationship of data in the database
  - Test Plan – How the QA team will verify that SemesterSync is operational and works as expected
- Implementation:
  - Untested Application – An untested version of the application is complete and ready to move to the next stage
- Testing:
  - Test Results – The results of the testing
  - Application sign-off – The approval that the application passed all tests and is ready for production
- Deployment:
  - Production Application – The completed, production-ready application will be deployed and become available to WGU and its students
  - User Guide – A walkthrough of how students will interact with SemesterSync
- Maintenance:
  - Support – Ongoing support will be provided as needed for the duration of the maintenance contract

Deployment Plan and Outcomes

Deployment of SemesterSync will be relatively easy because it is a new product. This means there will be no downtime, and the deployment will not impact users. Furthermore, SemesterSync can be published and verified before students are given notice that the application is available.

Since SemesterSync is using the waterfall methodology, the deployment phase cannot begin until the testing phase has concluded. Also, the maintenance phase cannot start until the deployment phase is completed.

The parties involved in deployment are the following:

- Project Manager – The project manager is responsible for coordinating all deployment tasks and ensuring deployment runs smoothly and according to schedule
- Quality Assurance Team – The QA team will approve that the application passes all tests and is ready to move to deployment
- DevOps Team – The DevOps team will be responsible for taking the application from the development environment to the production environment, ensuring a seamless transition

The steps for deployment are:

- Verification – Ensuring results from the QA team show passing tests and that the application is ready to go live. The outcome of this step is to make a final check before deploying to ensure there are no loose ends and that deployment will run smoothly.
- Preparation – The project manager will check in with WGU and all internal teams to plan the correct time for deployment and distribute the deployment plan. This will ensure that all parties are available for deployment if something goes awry and that deployment can happen seamlessly.
- Deployment – The project manager will coordinate with WGU staff and all internal teams throughout the actual deployment of SemesterSync. The DevOps team will initiate deployment by uploading the repository to the Google Play Console. The DevOps team will also ensure that all information on the store listing is accurate. Once the application has been successfully added to the Google Play Console, there is a mandatory 14-day wait period before SemesterSync is published in the Google Play Store.
- Training – User training will occur during the mandatory 14-day wait period. User training will be teaching WGU staff how to use SemesterSync. While the actual users of SemesterSync will be the students, WGU staff must understand how the application works. This will allow students to contact their program mentors for assistance in using the application instead of filing a support ticket directly to SemesterSync.
- Maintenance – After the conclusion of the 14-day wait period, the application will go live for public use. At this point, the deployment stage of the waterfall methodology is complete, and the maintenance stage will begin.

Project Timeline

Phase	Milestone/Task	Deliverable	Description	Dates
Requirements	Task 1	Requirements Documentation	Drafting and finalizing the scope of the project	7/22/24 – 7/25/24
Requirements	Task 2	Project Schedule	Create a schedule for stages	7/25/24 – 7/31/24
Design	Task 3	Wireframes ERD UML class diagram	Create a Mockup of the UI Create DB entities & relationships Create classes, entities, methods, and relationships	8/1/24 – 8/5/24
Design	Task 4	Test Plan	A plan for how the functionality will be verified	8/5/24 – 8/9/24
Implementation	Task 5	Untested Application	SemesterSync will be fully created but not be tested yet	8/12/24 – 8/26/24
Testing	Task 6	Test Execution	Testing will begin	8/27/24 – 8/30/24
Deployment	Task 7	Production App User Training	The application will go live and become available for public use. WGU staff will receive training on how to use SemesterSync.	9/2/24 – 9/18/24 (This includes the 14-day wait time)
Maintenance	Task 8	Support	Ongoing support and maintenance	9/18/24 – End of contract

Environments and Costs

Programming Environment

Hardware Requirements:

- Development machines – Each team member will need a machine capable of running the .NET SDK, Visual Studio 2022, and SQLite. Machine minimum requirements are as follows:
    - CPU Architecture:
        - ARM64 or x64
    - Memory:
        - 4 GB RAM
    - Storage:
        - 850 MB
    - Graphics:
        - GPU or on-board video supporting 1366x768

Software Requirements:

- Operating System – The minimum version of each OS listed below is necessary to utilize all other software requirements
    - Windows Minimum Version – Windows 10 1607
    - macOS Minimum Version – macOS 12
    - Linux – Alpine 3.17, Debian 11, Fedora 39, Ubuntu, 20.04, and more
- .NET SDK 8.0 - The .NET SDK is required to use the MAUI framework for building cross-platform applications.
- Visual Studio 2022 - Visual Studio 2022 will be the integrated development environment of choice for developers, as most documentation and Microsoft support uses this.
- Git – Git will be used to manage version control. This will enhance developer collaboration, help mitigate bugs in the production code, and allow for easier rollbacks if an application-breaking change is made.
- GitLab – Gitlab will be the cloud storage repository for all versions of SemesterSync. This will enable developers to make and receive changes in the code quickly.
- Google Play Store – SemesterSync will be deployed onto the Google Play Store via the Google Play Console, making the application accessible to the public.

Environment Costs

The costs for SemesterSync can be divided into two categories: start-up and recurring expenses.

- Start-up Costs:
    - Google Play Console – To publish an application to the Google Play Store, Google requires a one-time fee of $25. After the payment is paid, the account can publish unlimited apps.
- Recurring Costs:

  - GitLab – GitLab has a recurring cost of $348 annually. This cost is justified because it will give the development team access to more CI/CD tools, planning tools, support, enterprise user and incident management, and 10,000 monthly compute minutes. This will improve the application's development experience, speed, and quality.

With the start-up and recurring costs, the expenses for the first year will be $3,157, accounting for three developers, 3 DevOps members, 2 QA members, and one project manager. Every year after, there will be $3,132 plus $348 for each new team member.

Human Resource Requirements

The labor costs to complete SemesterSync with nine people working for various hours across 8.5 weeks are as follows:

- Cost per Developer: $47/hour * 160 hours = $7,520
- Cost per QA tester: $44/ hour * 72 hours = $3,168
- Cost per DevOps Engineer: $53 / hour * 24 hours = $1,272
- Cost for Project Manager: $47 / hour * 112 hours = $5,264

Total Cost by Team:

- Developer Team: $7,520 each * 3 members = $22,560
- QA Team: $3,168 each * 2 members = $6,336
- DevOps Team: $1,272 each * 3 members = $3,816
- Project Manager: $5,264 each * 1 member = $5,264

This brings the total labor costs for SemesterSync to $37,976.

Total Cost by Task:

1. $47 / hour (PM) * 32 hours = $1,504
2. $47 / hour (PM) * 40 hours = $1,880
3. $47 / hour (Dev) * 32 hours * 3 members = $4,512
4. ($47 / hour (Dev) * 40 hours * 3 members) + ($44 / hour (QA) * 40 hours * 2 members) = $9,160
5. $47 / hour (Dev) * 88 hours * 3 members = $12,408
6. $44 / hour (QA) * 32 hours * 2 members = $2,816
7. ($53 / hour (DevOps) * 24 hours * 3 members) + ($47 / hour (PM) * 40 hours) = $5,696
8. N/A

Validation and Verification

Multiple techniques will be used to test the application and ensure that SemesterSync is up to standard. Testing will ensure that SemesterSync has all the necessary functionality and that it does not have any bugs. The following methods that will be used are as follows:

- Unit Testing: All developers will create unit tests for the functionalities they create. Unit tests will cover all methods and services. The unit tests will be verified by ensuring the test produces the desired outcome every time.
- Integration Testing: Before moving onto the testing phase in the waterfall methodology, the development team will conduct integration testing. The integration testing will ensure that all components work together cohesively and produce no unexpected results.
- Functional Testing: In the testing stage, the QA team and WGU staff will conduct functional tests. These tests will ensure that SemesterSync can handle different conditions, such as device type, and respond as intended.
- Bug Fixing and Re-testing: All documented bugs will be fixed or deemed a non-issue once all testing has concluded. Then, the application will be retested to ensure the bug was corrected.

Citations

- Western Governors University. (n.d.). *Student Experience.* https://www.wgu.edu/student-experience/outcomes/students-graduates.html#:~:text=Nation%20of%20Night%20Owls%3A%20175%2C983%20Students&text=WGU%20has%20a%20diverse%20student,as%20of%20June%2030th%2C%202024
- Western Governors University. (n.d.). *The Story of WGU.* https://www.wgu.edu/about.html
- Western Governors University. (n.d.). *Tuition and Financial Aid.* https://www.wgu.edu/financial-aid-tuition.html
- Western Governors University. (n.d.). *Student Diversity.* https://www.wgu.edu/about/story/measuring-impact/student-diversity.html
- Western Governors University. (n.d.). *2023 Annual Report.* https://www.wgu.edu/about/annual-report.html