**C868 – Software Capstone Project Summary**

**Task 2 – Section A**



|  |  |
| --- | --- |
| **Capstone Proposal Project Name:** | Customer Database and Appointment Solution for Sam Sam Tech. |
| **Student Name:** | Isabella Cerda Grigolla |

**Table of Contents**

*Create a professional looking Table of contents that includes your main and subheadings and the related page numbers. Use the automatic TOC generating function of Word or other word processing packages to make the process easier.*

*The headings that follow are only examples of what might be included. You will need to create headings that are appropriate for your application and process.*

***Remember that this needs to be a professionally formatted document with detailed information about your project that is easily accessible.***

# **Business Problem**

**The Customer**

The customer for this project is Sam Sam Tech. They currently provide consultations and repairs for tech-related issues; Their current services include diagnosing hardware/software-related issues, component/software upgrades, troubleshooting, component optimizations, and mobile phone repairs. They started small in their area but due to a significant lack of repairability of today’s devices, their business grew exponentially. At the start of their venture, they only had three employees, all of them co-founders, now two years later have thirteen total employees. They have plans to expand into different ventures in the tech related field. Such as selling devices, offering at home installation, remote diagnosing, and cellular upgrades. They want a system that will set a foundation for them in their continuing effort to grow and strive in this harsh economy.

Sam Sam Tech’s current structure has three divisions, equipment and stock, repairs, and management; Each co-founder runs a division and with this structure, they have a solid foundation for their exponential growth. Currently, only the Equipment and Stock division has a program in place to allow employees to keep track of their current stock and equipment that is available to them. Since their mission is to continue to grow and allow anyone to be able to repair their devices at reasonable costs, they need to ensure their costs and demand don’t exceed their expectations. They’re hoping to continue expanding on their current clientele, along with being able to provide the services the intend, they will then gain more attention from businesses as well. Along with their current prospects of working on new services they want to expand from just families to the enterprise level as well.

Their short-term goals consist of maintaining current operations will keeping the wait as short as possible. Since this is the goal of all three co-founders, they are quite confident in what needs to be done to ensure this stays the same. They show an absolute concern for maintaining customer satisfaction, which ensures that they will come back in the future.

Their current long-term goals differ quite significantly, but remain unified in their overall mission to provide their essential services. To ensure that each member is on board with this, they agreed on their need with this program. This will help keep track of customers and their history, so with any future new employees and services they will better be able to serve each and every person. Since they also want to provide future services that require heavily on appointments, they need to get this done sooner, rather than later. Once they gain this program and are able to fully utilize it, they will then be able to proceed into further ventures and better cater to their customer base.

## **Business Case**

The current operations of Sam Sam Tech, allow for each employee to keep track of which customers they help and the issues they are fixing. Their three divisions allow for a complete and unified structure for their internal communications. Where they lack currently is organizing which employee has meetings with specific customers. Their current system has employees in the management division emailing employees in the repairs division. Those emails consist of customer names, the time of their appointment, and their concerns. This process can be rather lengthy and tedious as the repair employees have to check their email every time, they want to review what the issue is and the names of their customers. During this time there have been miscommunications between employees. These are the following for some of the issues: mislabeled reasoning for appointments, wrong customer names, wrong time/day of appointment, and wrong employees being emailed.

These common issues show there is an express need from the company to help dwindle these issues. For the project we have in mind, we will have a centralized application for employees to access as well as a database to store the information. This application gives employees the ability to not only create appointments but to view when appointments are. This negates the need for management employees to email repair employees, as well as for repair employees to have to check their emails. The process for onboarding this program will be seemingly smooth as they will have a minimal amount of time for training with the program as it is very seamless for user interactions.

## **Fulfillment**

Sam Sam Tech expressed that they’d like to host this application locally as well as the database. Since the cost of cloud software and hosting is increasing, they deemed it to be too costly. They also verified that they already have a centralized server on location, which already hosts their inventory program for the inventory and parts department.

We will be designing the program with functionality at its core. There is no need for it to be graphically intensive as they just need the information to be up to date and accurate. The program will be desktop based for Windows OS Systems; the application will then communicate with a MySQL database hosted off of their local server.

The basics of this application will include, log in functionality, customer data, appointment creation, calendar to view appointments as well as filter by time, and reports for based off of time and users. These functionalities will cover anything that the company will need for the foreseeable future in regards to this applications functionality. Each user has to log in first to be able to gain access to the applications functionality, there is a process taken to ensure that this first step is locked and guarded to prevent intrusions of external sources. Once the user logs in to the application, they will be greeted with the application directory which clearly defines which button takes them where and what they will provide.

Since Sam Sam Tech expressed a need for a better communication system around their customer appointments our system will be operating with user simplicity in mind. There will be no complex menus or unnecessary menus. They are already struggling with setting up and maintaining their issues, they have no need for a system to have additional issues.

Overall, we believe that each and every function that is provided in this application will be a true benefit to Sam Sam Tech in their current dilemma and for the foreseeable future.

# **Existing Gaps**

Their current method of setting up meetings involves a tedious process; Usually a customer can call and make an appointment ahead of time or they can schedule a follow up meeting from the previous appointment. This seems to be a strain on their current architecture as it makes the customer jump through hoops as the repair employees are unsure whether they have current appointments or not.

1. The following process is to setup an initial appointment;
2. For a phone call will usually proceed as the customer dials in.
3. An employee from the management department will answer.
4. The management employee will then confirm with the customer the date and time best for the customer.
5. They will then send a confirmation email to the customer for their records.
6. The employee will then notate it in an email for a select repair employee to take on.
7. Depending on the availability of that day they are usually within the next day or so.
8. The day of the customer’s appointment: the management employee then emails the repair employee for that days appointments.
9. Then finally: the repair employee then preps and get’s ready to receive the customer and their device.
10. Once the customer and the tech are done with their appointment, they are officially concluded.
11. (Optional) The customer then would like to setup a new appointment or follow up, the repair employee would then be referred to step 2 in the follow up appointment chart.
12. The following process is for a follow up appointment;
13. After the initial appointment is finished the customer would then be able to request a follow up visit. Whether it pertain to the issue for the current visit or a new issue entirely.
14. The repair technician will then email the management.
15. The management employee will then confirm and check the calendar for availability
16. Once the appointment is set, the management employee will then send a follow up email to the customer confirming that their appointment is set up.
17. The day of the follow up appointment, the repair employee will then get an email from management notifying them of the customer coming in and their issues.
18. The repair technician will then proceed with their appointment and fulfill what the customer needed help with.
19. (Optional) The follow up process is then started over again from step 1.

Once this system is replaced with the new program, all the strenuous and tedious parts of this process will no longer be necessary. There will be less of a need to email different departments for 1 appointment, and all employees will be on the same page. The number 1 goal for this application is to gather a unified base for all employees in regard to appointments. There has been tremendous delay and miscommunications due to the strenuous older method listed above.

# **SDLC Methodology**

The methodology we will be adopting for this project is Agile. Our repertoire consists of projects completed with the Agile methodology. We gain great feedback from not only our employees but from our clients as well; Each client is with us every step of the way versus the final results, via waterfall. The reason why we go with agile often is because it allows us to test at every stage we’d like. For example, if we complete a whole page of a program but we do not have the rest of the application to go with it yet, we can test it by itself and document if that functions as it should completely by itself. Once smaller testing is completed, we can then test in bigger pieces of the program. This allows us to keep a handle on bugs when they arise, to keep in mind every program will have their issues; Agile does not prove to be 100% efficient at squashing out all bugs in a program, but it does help to alleviate what keeps the program from functioning.

I will go over the phases in our Agile methodology to give a better overall understanding:

1. Planning
   1. In the Planning phase of the Agile Methodology, we gather requirements and wants from the client.
   2. Once we gather these pieces of information we then set up a process to begin the introductory phase.
   3. We have our employees then begin brainstorming the appropriate steps to present what we have developed and test from there.
   4. We would usually present a low-fidelity wireframe and sample data to show how it would function.
   5. We would then gather the results from the client and change what would need to accordingly.
   6. We then split off some employees to begin in the Development phase.
   7. We get confirmation again from the client to begin full development of the project.
2. Development
   1. We onboard the employees who have begun development from step f in the planning phase
      1. These employees will develop only the parts of the application that have gotten full approval from department heads and the clients.
   2. Once we get full approval from the clients we will begin bringing on the rest of the employees to start the process of testing.
   3. Employees will begin to test the pages/application bits they have already compiled.
   4. Once testing of those pieces have been done, we will then show the clients how it functions and if this is in line with what they were expecting.
      1. If client approval is not accepted in this phase there well be backtracking to then redo the entire components.
   5. The next step will then allow employees to continue on with the development of another component until the entire program is created.
      1. All steps from the development cycle will be used until the application is fully developed.
3. Testing
   1. This is the formal Testing phase.
   2. With Agile’s methodology, it allows testing to be part of every step of the way, or whenever we would like it to be.
   3. In the formal Testing phase, we will test the program as a whole with both combined efforts of white box testing and black box testing.
   4. Those two testing methods combined usually allow for squashing more bugs and finding incompatibilities with user intuition when using the program.
   5. Once testing is complete and in a satisfactory rate, it will be taken to the clients to show the almost finalized project.
4. Deployment
   1. Once full client approval is received the Deployment phase will commence.
   2. This step usually has a lot of communication between the employees and the clients onboarding team.
   3. With the clients onboarding team, there will be a stream of information and uploads coming from the internal employees.
   4. There will be documentation received as well as the program itself.
   5. This phase can be different depending on clients needs but if a client has an internal hosting method there are two choices.
      1. The internal employees can head there to install the program themselves.
      2. The onboarding team will receive the documentation including files needed to setup the entire application along with it’s necessities to run.
5. Maintenance
   1. Once the Deployment phase is fully completed, the Maintenance cycle takes over.
   2. The Maintenance cycle mainly consists of taking care and managing any issues that arises once the clients take over the completed program.
   3. Issues can range from inputs not registering to crashes on newer operating systems.
   4. There will be a point in different types of cycles where once maintenance ends there will be a retirement cycle.
      1. Since this application does not rely on any sort of cloud infrastructure or the hosting capabilities of another company, this step relies mainly on the client themselves.
      2. Once the client decides it Is time to move onto something better or that fulfills their new requirements will the process start over once again.

# **Deliverables**

We will be having two different types of deliverables for this project. An internal deliverable which will have employees show certain aspects and designs to higher up. The other is external deliverables which will be designs, mockups, and test results to be shown to the client.

## **Internal Deliverables**

These are the deliverables only to be seen within the programming company.

* Internal Timeline
  + When a project goal will be hit and when testing of each phase will be completed.
* Testing methodologies for each phase.
  + i.e. White box testing, black box testing, grey box testing.
  + Once the methodologies are confirmed who and from where will they conduct the testing.
* Presentable results of tests to be diagnosed.
  + Once diagnosed there will be an opportunity to present the findings to the clients.
* Communication Updates
  + These don’t necessarily have to be full in depth documentations but at the very least need to communicate where progress is currently at in the given phase of development.

## **External Deliverables**

These are the deliverables created specifically to be shown to the client.

* A low-fidelity wireframe
  + This wireframe will not be interactive but will just show the overall look and in-depth explanation for what each thing does
* A design chart for the database
  + Similar to the wireframe, this will just show the key tables in the database for how each table looks, their primary keys and foreign keys.
* Presentation of test results
  + These results can include failures, which lead to being able to get closer to what the client needs.
* A localized demo via RDP
  + This demo can give a hands on impression not only for testing but for showing how the client can see their application is coming.

# **Implementation**

The Implementation part of the process will be quite a strenuous one, not only will both teams have to communicate during this, there will be rigorous testing to ensure that the project is not only fully transferred but in working condition. The Validation and Verification process is usually done right before this step. Since Agile takes communication hand in hand, there will be no issues with what the client is expecting once implementation is finished.

I will go over the team members needed from both the internal team and the external team for the implementation process.

**Internal** **Members**

These will include employees from the company which created the application.

* Project Coordinator
  + The Project Coordinator has been with this project since the beginning and has overseen all developers in this process.
  + They will be the ones communication with the owners of the client company.
  + They will also be receiving updates from the deployment employees.
  + They will be conducting close communications with the client and their employees if there is any sort of miscommunications from either side.
* Deployment Employees
  + The Deployment Employees only embark upon the project once the implementation phase has begun.
  + Their abilities usually consist of Customer Service and a minimal degree of programming related experience.
  + They partake on communications with the External employees that are onboarding the program onto their systems.
* Stand-By Programming Employees
  + Usually at this stage of the process there is a backup made just in case of corruptions or missing files.
  + The Stand-By Employees are here to fix any issues that are caused by corruptions or mistakes from the client.
  + This is usally relegated by the backup created, they are extremely helpful to the Deployment Employees, just in case if there are any questions or concerns they cannot answer.

**External** **Members**

These will include employees from client company which is receiving the application.

* Owners
  + Similar to the Project Coordinator, they will be communicating with almost everyone in their business along with the Deployment Employees.
  + The have already given confirmation in the Validation and Verification phase.
    - Which is the step that triggered the process for the implementation phase.
  + They will be communicating with the Management Employees all throughout this process.
  + Since management Employees will not be involved in implementation, they will be involved with training; Along with overwatching how each employee will handle their part of this program.
* Receiving Employees
  + The receiving employees will be the programmers of the client company.
  + Their duties will primarily consist of communicating with the Deployment Employees, and Stand-By Programmers if needed.
  + Since Sam Sam Tech already has a localized server, there will be no need for down time.
  + The database from this new program will work in tandem with the existing architecture of the MySQL Database.
    - Documentation for installation will be included to follow protocols.
  + These employees will also be testing their end of the program during this process to make sure nothing was corrupted or lost when the files were sent over.
* Management Employees
  + These employees will not be actively in the process of implementation
  + Although, they will be in the loop to be aware of how the program works.
  + The Clients may have already shown these employees the program before this step.
    - Communication is key, as once implementation is done, training is to follow.

The overall process of Implementation can be quite lengthy as anything can go wrong. Implementation usually consists of transferring over the essential files from the programming company to the client company. In this example since a server is already implemented there will be no need for the clients to review the database installation documentation; It is recommended anyways for good measure as there may be a need for it to avoid issues.

The best time for this phase to start is on a slow day during operating hours. Since there is a minimal system in place already there will really not be a need for it to be taking down operations of other employees. The repair and management employees can continue with their normal operations. Although Management employees will be kept in the loop for this process, they are once again not integral for integration. The ones that are needed for this process will be heavily involved with communicating with other team members and the other companies employees. Each employee in their respective company will work in tandem with their other departments to not only ensure functionality is still up, but that their progress is in good shape.

# **Validation and Verification**

Since we are using the agile method we will not only be testing at each phase but notifying the clients of their progress. To ensure that we are covering the requirements of the clients we are following a rigorous testing model during each phase. We will do primarily white box testing as the programmers would need to know what they are keeping an eye out for; For example, if the client wants there to be a filtering option for the data fields, then we would have the programmer with test data try to find if that functionality has been implemented correctly. There will be no need for black box testing in this current phase as we are not looking for usability, we are looking at core functionalities.

Once testing is completed for each phase we will then move on to the usability testing. Which will be a mix of grey box and white box testing. Those two tests will be the precursor for the acceptance tests. Grey Box testing will be a programmer from a different department, who will be given a list of functions to look out for and when to test the functions themselves. The Black Box test will be someone from an internal QA Department, whom would have no prior programming experience whatsoever. The purpose of the black box test would be to test the overall usability of the program itself; To look for any faults and flaws within the program, once completed we will then proceed with the Acceptance Test. The following acceptance test will be among the first releases of the program, a 1.0 release. This build will provide the foundation for the maintenance phase and will be the program that the client was asking for. We will be providing a copy of the requirements from the agreed upon contract from the start of the project to the client. Along with the list of requirements the client is then encouraged to test out the application in it’s capacity.

The client can conduct several tests of their own, as we encourage to test each program to their fullest. Once the client has given the confirmation that the program has meet their requirements and has passed the Acceptance phase, the project will then move onto the Implementation phase.

# **Environments and Costs**

## **Programming Environment**

The system currently used to program these projects is quite simple. We have several users with desktop at their stations linking to their project repository. The project repository consists of GitHub Enterprise accounts which then links to each employees computer for any sort of update. These programmers are then split into different divisions that tackle on multiple projects.

We will now go more into the specifics of the current technology used.

List of Employee and Company Technology

* Windows Server 2019
  + MySQL Databases Hosted on the server itself
* Windows 10 and 11 Operating Systems
  + Some Employees will utilize different versions of Windows between Home and Pro

## **Environment Costs**

Provide an explanation of the costs associated with the software application. Some might be startup, first-time costs while others might be a percentage of licensing costs. Environment costs are relatively minimal. The environment where the system resides in a shared environment where costs are shared by the organizations. There is a nominal fee associated with maintaining the database of $500 a year that allows for unlimited storage size and 99.8% uptime. The web server is another fee of $300 a year that includes maintenance and upgrades of the following; Windows Server, IIS, and ColdFusion. The final cost is based on the thick or thin clients utilized by the customer. Each device that is attached to the network has a $40 annual fee which covers Operating System and Network upgrades.

The environment costs for this project will be at a minimum expenditure, due to the customer owning most of the resources that are required for the application to function. The client has requested

## **Human Resource Requirements**

What is the time and cost for the labor to complete the application?

For example: The larger share of human resource is by the developers of the project followed by the PM. Developers consume approximately 75% of the hours and dollars associated with …etc.

# **Project Timeline**

For this section, you'll need to look at the phases of the project and provide information about the time required to complete each phase.

For example:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Phase | Milestone/Task | Deliverable | Description | Dates |
| Pre-development | Task 1 | Requirements | Meeting with customer and procedure review | 6/1/2018 – 6/30/2018 |
| Design | Task 2 / Design files | Low fidelity wireframe  High fidelity mockup | Create the UI that relates the look and feel of the project | 7/1/2018 – 7/15/2018 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |