**BIT Project Management Plan**

**Monstrosity Inc. Network**

rev. 3/8/2020

**Team Members**

Ryan Shepherd

Andrew Taylor

Liam Floyd

Document Control

**Change History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision** | **Change Date** | **Completed by** | **Description of changes** |
| V1.0 | 3/04/2020 | RS, AT, FL | Initial baseline release |
|  |  |  |  |

**Document Storage**

This document is stored in the project’s repository at: <https://github.com/umkc-cs-451-2020-spring/semester-project-group-17>

**Document Owner**

Ryan Shepherd is responsible for maintaining this document. Ryan Shepherd, Andrew Taylor, and Liam Floyd are responsible for updating this document.**Table of Contents**

[**1**](#_3znysh7) **OVERVIEW 4**

[**1.1**](#_23ckvvd) **Purpose and Scope 4**

[**1.2**](#_ihv636) **Goals and Objectives 4**

[**1.3**](#_4d34og8) **Project Deliverables 5**

[**1.4**](#_32hioqz) **Assumptions and Constraints 6**

[**1.5**](#_2s8eyo1) **Schedule and Budget Summary 7**

[**1.6**](#_17dp8vu) **Success Criteria 9**

[**1.7**](#_3rdcrjn) **Definitions 10**

[**1.8**](#_1hmsyys) **Evolution of the Project Plan 11**

[**2**](#_41mghml) **STARTUP PLAN 11**

[**2.1**](#_26in1rg) **Team Organization 12**

[**2.2**](#_lnxbz9) **Project Communications 12**

[**2.3**](#_35nkun2) **Technical Process 13**

[**2.4**](#_2jxsxqh) **Tools 13**

[**3**](#_z337ya) **WORK PLAN 13**

[**3.1**](#_3j2qqm3) **Activities and Tasks 13**

[**3.2**](#_1y810tw) **Release Plan 15**

[**3.3**](#_4i7ojhp) **Iteration Plans 16**

[**3.4**](#_3whwml4) **Budget 19**

[**4**](#_qsh70q) **CONTROL PLAN 19**

[**4.1**](#_3as4poj) **Monitoring and Control 19**

[**4.2**](#_1pxezwc) **Project Measurements 20**

[**5**](#_49x2ik5) **SUPPORTING PROCESS PLANS 21**

[**5.1**](#_2p2csry) **Risk Management Plan 21**

[**5.2**](#_147n2zr) **Configuration Management Plan 26**

[**5.3**](#_3o7alnk) **Verification and Validation Plan 27**

**5.4 Product Acceptance Plan 27**

# **Overview**

## *Purpose and Scope*

This document outlines the plan to implement the requirements set forth in the design of the information technology infrastructure for Monstrosity Inc., its employees and customers. The purpose of this document is to present the action plan that the design team will follow in execution of this design project.

Monstrosity Inc. has contracted with our firm to design and recommend an infrastructure package that will provide high speed wireless internet connectivity for their employees and customers, access to apps such as secure email, and a database system to manage the real estate companies information. This system will be used by approximately 200 employees of Monstrosity Inc. as well as its customers and will have to be implemented in 5 satellite offices and 1 corporate office.

This document doesn’t address the requirements of the project in detail as this is addressed in a separate project document. Additionally, this document does not address detailed user interface implementation which would be accomplished at the software level.

## *Goals and Objectives*

Our project goal is to research, design, and propose the infrastructural solution to the technical needs of Monstrosity Inc which includes designing the following objectives:

1. secure network with intrusion detection, as well as wireless internet services for employees and customers,
2. cloud services hosting for secure email,
3. onsite database hosting infrastructure.

We will complete this over the course of 5 planning and design iterations that will end with the final proposal being presented by May 4, 2020. A 6th iteration will cover the physical build-out and will take place after this project has been completed and approved by stakeholders. This infrastructure will give Monstrosity Inc employees consistent access to company tools and digital resources across its many locations and while out in the field, thus allowing employees to be more productive, knowledgeable, and effective in their jobs. Additionally, from a customer experience perspective, customers will benefit from limited internet access

## *Project Deliverables*

This section lists the outputs of the project that will be delivered to Monstrosity Inc.

The following design items and their implementation recommendations will be delivered to the customer on or before 5/4/2020 as a proposal document:

1. The network infrastructure spanning across all offices
2. The database infrastructure, fully connected to the network
3. Customers and Employees have access to internet with varying permissions for access
4. A comparison of cloud hosting platforms and a recommendation of which platform will best fit Monstrosity Inc requirements.
5. A document detailing how security will be maintained for the product
6. Network (and other necessary infrastructure) diagrams
7. Any necessary configuration considerations
8. Hardware recommendations that will be necessary to implement the product
9. Other general recommendations

## *Assumptions and Constraints*

Assumptions:

1. It is assumed that Monstrosity Inc is a local real estate company with all offices residing in the same state, managing several properties that are for sale to the public and businesses.
2. Monstrosity Inc is being considered a small to medium sized business.
3. It is assumed that Monstrosity has a limited budget, characteristic of a small/medium real estate company. For example they may look more to hosted services instead of building extensive in house applications and on-prem infrastructure..
4. It is assumed that this project will be treated as a simulation of a “real-world” scenario, making decisions based on considerations of likely consequences in a practical implementation.
5. Due to the technical nature of physical infrastructure (hardware configurations), the average user (customers, managers, employees) will not be able to manage this system should problems arise and that Monstrosity Inc will likely need to contract with a third party IT support professional who will be on-call to provide maintenance and assistance with operations.

Constraints:

1. We are designing a network layout for a purely hypothetical company with no actual finances, existing infrastructure, or offices.
2. With no offices we must rely heavily on assumptions developed by the team.
3. We are also operating as a group for a school project, giving limited time with schedules involving work, other courses, and more.
4. Operating with no defined financial constraints of the infrastructure. In general, we will try to keep the costs to a minimum as to what we feel would be logical costs/budget constraints of a business of this size.
5. As students, we have limited experience and knowledge in applying architectural design and will require research to complete this project. All variables that would affect a real-world design such as this may not be considered due to inexperience.
6. As a hypothetical company with no ability for physical implementation, testing designs is limited.
7. The database system will be based on MySQL.
8. The system will need to be accessible from a variety of devices (desktops, laptops, tablets, phones, etc) that run on various operating systems (Windows, Mac, Android, etc).
9. Due to the technical nature of physical infrastructure (hardware configurations), the average user (customers, managers, employees) will not be able to manage this system should problems arise.

## *Schedule and Budget Summar*y

* + 1. **Iteration 1**: Requirements Specification and Needs Analysis

Milestone: Completion of Project Charter and Requirements Document. Scope of project determined. Each member of the design team will be assigned a component of the product and complete a needs analysis. The design team will fully understand the needs the product will satisfy for Monstrosity Inc.

Iteration Period: 2/17/2020 - 3/2/2020

Project Charter completion date: 2/23/2020

Component Needs Analysis completion date: 2/28/2020

Requirements Baseline completion date: 3/1/2020

* + 1. **Iteration 2**: Project Planning, Risk Assessment, and Initial Implementation Proposal draft (prototype)

Milestone: Completion of Project Plan, Project Risks Report, and PowerPoint slide deck outlining initial implementation of product. The design team will fully understand how the project will be organized, work completed, and the outcomes that will be achieved for each of 5 iterations. The design team will research their assigned components to produce a high-level prototype project presentation.

Iteration Period: 3/2/2020 - 3/16/2020

Project Plan completion date: 3/8/2020

Project presentation initial draft/prototype (powerpoint format) completion date: 3/13/2020

Risk Report: 3/15/2020

* + 1. **Iteration 3**:

Milestone: Completion of first project presentation covering a high-level implementation of the system to stakeholders. Each team member will have completed research into their assigned component. The team will use this information to begin documenting the architecture implementation.

Iteration Period: 3/16/2020 - 4/6/2020

Project Presentation to stakeholders: 3/20/2020

Architecture Document completion date: 4/3/2020

* + 1. **Iteration 4**: Architecture implementation design finalized

Milestone: Final report on architecture implementation will be completed. The design team will focus on implementation of their assigned component, but the team as a whole will coordinate how components will be implemented within the system as a whole as each component is interconnected. Hardware will be selected and final recommendations to Monstrosity Inc will be detailed.

Iteration Period: 4/6/2020 - 4/20/2020

Final architecture report completion date: 4/20/2020

* + 1. **Iteration 5**: Project Wrap-up and closeout

Milestone: Completion of project test plan, user and system guide, project results, and final presentation to stakeholders. The team will work together to finalize documents that will accompany the final architecture proposal report and present the project to stakeholders. Additional graded tasks will be completed. The project will be considered completed.

Iteration Period: 4/20/2020 - 5/4/2020

Project test plan completion date: 4/26/2020

User/System Guide completion date: 4/27/2020

Project results PowerPoint completion date: 5/1/2020

Github/Slack communication tasks completion date: 5/1/2020

Final presentation dates: 5/4/2020, 5/6/2020, 5/8/2020

* + 1. **Iteration 6**: Post project completion physical build-out

Milestone: Monstrosity Inc approves proposal. Build teams engaged to start and complete build. Hardware ordered and installed.

Iteration Period: 5/4/2020 - 8/5/2020

Firewall setup - 6/10/2020

Completed network: 7/1/2020

Database connection: 7/15/2020

Email conversion: 7/20/2020

Testing: 7/1/2020 - 8/5/2020

* + 1. Budget Summary

1.5.6.1 Amount Monstrosity Inc has budgeted for the project: $175,000

1.5.6.2 Budgetary Constraints of Monstrosity Inc.

Monstrosity Inc is considered a small-medium business and does not require the same enterprise level infrastructure as a large nationwide (or global) tech company would. The budget reflects this lowered need. Monstrosity Inc would expect the project to come under the amount budgeted for the product. The design team should make every effort to meet this requirement. As such, some concessions may need to be made to meet budgetary constraints if necessary. The final proposal should reflect options that would lower costs if available. Monstrosity Inc would have yearly budget constraints regarding continued operation of the infrastructure. The budgeted amount set forth in this document covers the initial buildout of the infrastructure. Some elements of yearly operational costs are not included in this initial budget, but should be considered by the design team when making the final architecture proposal.

1.5.6.3 Build-out Cost Estimates:

* Project manager: $100.00 an hour for 8 hours a week for 12 weeks = 48,000.00
* 2 Architects: $67.97 an hour for 8 hours a week for 12 weeks = 65,251.20
* Consulting expenses: $1,000
* Servers and cluster infrastructure: $15,000.00
* Network Equipment: $4,000.00
* Security Infrastructure: $15,000
* Misc. Hardware: $500.00
* Buildout Contractors: $2000
* First month cloud services: $2000

Total Labor, Hardware, Services Estimate = $152,751.20

Ongoing Cost Estimates:

* Full-time IT staff person (yearly wages): $55,000
* Ongoing IT management and Maintenance (yearly): $2510
* Security Licenses for approximately 200 users (yearly): $4800
* Cloud Hosting Services Email and Filter hosting for approximately 200 users (yearly): $24,000

## *Success Criteria*

The project will be deemed a success if:

1. We produce a network and systems architecture design that meets Monstrosity Inc.’s requirements.
2. Since this is a school project, the project will be a success if the team gets a grade no lower than a B.
3. There is an expectation that each team member will contribute equally to the project not only in on-time completion of their assigned work but also in providing complete and adequate detail so that other team members are not having to “pick up the slack”. This will be measured in the reviews that will be completed at the end of the project.
4. The team will communicate effectively and often so as to avoid any miscommunication.
5. Team meeting schedules and deadlines will be adhered to.

## *Definitions*

This section defines potentially unfamiliar or ambiguous words, acronyms and abbreviations.

**Team/Design Team** - refers to the design team of Ryan Shepherd, Andrew Taylor, and Liam Floyd.

**On-Prem** - “on premises”, infrastructure that is on-prem will be hosted physically on site at the physical location opposed to hosted in the cloud through a third party provider.

**TBD** - To be determined. When used in this document, this refers to material facts or decisions that will need to be made at a later date and will be added to this document as an addendum.

**Component** - the features of the product/system that is being designed. The system is being divided into the following components:

1. Network Infrastructure
2. Database and Server infrastructure
3. Security Infrastructure
4. Cloud Hosting Infrastructure

**Prototype** - a high-level, early draft of the implementation of the product components. This prototype will give stakeholders a general idea of the implementation.

**Use case** – describes a goal-oriented interaction between the system and an actor. A use case may define several variants called scenarios that result in different paths through the use case and usually different outcomes.

**Actor** – user or other system that receives value from a use case.

**Product** – what is being described here; the system specified in this document.

**Project** – activities that will lead to the production of the product described here. Project issues are described in a separate project plan.

**Shall** – adverb used to indicate importance; indicates the requirement is mandatory. “Must” and “will” are synonyms for “shall”.

**Should** – adverb used to indicate importance; indicates the requirement is desired but not mandatory.

**May** – adverb used to indicate an option. For example, “The system may be taken offline for up to one hour every evening for maintenance.” Not used to express a requirement, but rather to specifically allow an option.

**ISP** - Internet service provider, a third party company that will provide the infrastructure needed to access and utilize internet capabilities

**IT Professional** - It is assumed that Monstrosity INC will not have a full time IT manager on staff to maintain the system and that this will be contracted out to a third party who will be on-call.

**Cloud Service Provider** - A third party provider who maintains and manages their own data center who provides cloud services to the public

**Network** - refers to the physical infrastructure of a group of two or more computer systems linked together. There are many types of computer networks, including Local Area Networks (LAN), Wireless Internet (wifi), and Wide Area Networks (WAN).

**SLA** - Service Level Agreement. A service-level agreement is a commitment between a service provider and a client. Particular aspects of the service – quality, availability, responsibilities – are agreed between the service provider and the service user. As used in this document, SLA’s refer to agreements between the cloud service provider and Monstrosity Inc.

**UI** - User interface. Any number of front-end software that will allow the users to interact with the system.

## *Evolution of the Project Plan*

This section describes plans for updating the project plan throughout the project.

Before an iteration begins, the Project Plan will be updated to reflect actual effort for each activity in the prior iteration. In the event of an uncompleted task, it shall be carried forward and the plan updated to reflect these changes. Afterwards a schedule will be laid out in the Project Plan to determine the path forward for the iteration. At the beginning of each iteration we will lay out the plan for the iteration including what tasks need to be completed, who they will be assigned to, and account for any risks and how to mitigate them.

# **Startup Plan**

This section details how the design team will operate.

## *Team Organization*

This section explains project roles and the authorities and responsibilities associated with these roles.

Project Manager: The project manager is responsible for reviewing the project plan, monitoring deadlines, submitting work to stakeholders, managing risks, running the weekly team meeting, communicating items for review with the team, and providing monthly status reports to senior management.

Role will be filled by Ryan Shepherd

Architects: Architects are primarily responsible for researching, analyzing, designing, reporting, and documenting the infrastructure details pertaining to the components needing to be implemented the product. They are also expected to take part in architecture planning and review meetings. Each architect is responsible for knowing not only how their component will be implemented, but also how their component will connect with other components of the system.

Roles will be filled by:

Network Design Architect: Andrew Taylor

Network Hardware Architect: Andrew Taylor

Security Architect: Liam Floyd

Cloud Services Architect: Liam Floyd

Server Cluster Architect: Ryan Shepherd

Database Implementation Architect: Ryan Shepherd

Report Reviewer: All team members shall review all reports and documents for accuracy and holding each other accountable for project thoroughness and results. Each team member will review the whole reports and documents and compare against grading rubrics and adhering to the standard of getting full points.

## *Project Communications*

This section contains the project communications plan. The communications plan describes how information is gathered and distributed.

Every Monday, Wednesday and Friday, the group will meet between the hours 11AM and 1PM to discuss the project, assign tasks, discuss deadlines and what progress is being made. Messaging over slack and discord will also be done throughout the project to maintain an understanding of progress in between meetings. Documents will be stored on Github and Google drive. Google drive allows for live collaboration of documentation before being stored in the repo on Github.

## *Technical Process*

The phases of this project will be to research & design, develop, and analyze each of the four main pillars; network, database, security, and cloud hosting. Afterwards compiling a total list of hardware and software needs and how to implement them together, along with a laid out budget for Monstrosity Inc’s IT. Completion of each pillar will be a milestone. For organization we communicate primarily through slack, with a general check in on Wednesdays and Fridays. We host a meeting at 11am every Monday in person with remote members calling in through discord. For each iteration, the members each work on one of the pillars, distributed as such: Network: Taylor, Database: Shepherd, Security and Hosting: Floyd. Each member focuses on said area, with all coming back together weekly to contribute and keep consistent all paths.

## *Tools* Utilized

Azure/AWS/Google Cloud services

Cisco Packet Tracer

Cisco GNS3

Draw.io

Microsoft Office

Google docs

Github

Slack

Windstream

Port scanner

Kali Linux

# **Work Plan**

This section details the tasks that will need to be completed for this project and the plans to complete them.

## *Activities and Tasks*

**Design Network Architecture**

Research and develop a plan on the network setup for Monstrosity INC, covering the internal corporate network, external facing to web, and guest networks across the main office and how the satellite offices connect.

Owner: Andrew Taylor

Effort Estimate: 4 hours

Actual Effort: 6 hours

Planned start and stop dates: 2/17/2020 - 3/2/2020

Actual start and stop dates: 2/17/2020 - 3/2/2020

Dependencies among other tasks: Work with Security

**Create Network Infrastructure Hardware plan**

Research best hardware for use cases of Monstrosity Inc to cover network infrastructure. Layout the hardware needs and how it will be implemented.

Owner: Andrew Taylor

Effort Estimate: 5 hours

Actual Effort: TBD

Planned start and stop dates: 3/9/2020 - 3/16/2020

Actual start and stop dates: TBD

Dependencies among other tasks: Design of Network, Database, Security and Hosting

**Design Database**

Research and develop a plan for a MySQL database implementation and management for Monstrosity Inc.

Owner: Ryan Shepherd

Effort Estimate: 4 hours

Actual Effort: TBD

Planned start and stop dates: 3/16/2020 - 4/6/2020

Actual start and stop dates: TBD

Dependencies among other tasks: Work with Security and Network

**Designate Security Needs**

Research and design Monstrosity Inc's security needs, including both software and hardware solutions to cover both digital and physical aspects of cyber security, protecting company information, access control, database security, and network.

Owner: Liam Floyd

Effort Estimate: 4 hours

Actual Effort: TBD

Planned start and stop dates: 3/16/2020 - 4/6/2020

Actual start and stop dates: TBD

Dependencies among other tasks: Work with database and network

**Design Hosting Services**

Research and develop a plan on which cloud hosting services to use, how they will be implemented, what hardware and infrastructure will be required, and the cost to Monstrosity INC.

Owner: Liam Floyd

Effort Estimate:

Actual Effort: TBD

Planned start and stop dates: 3/16/2020 - 4/6/2020

Actual start and stop dates: TBD

Dependencies among other tasks: Mostly none, will work in tandem with Network

**Create On-Prem Server Infrastructure Plan**

Research best hardware for use cases of Monstrosity Inc to cover an on-prem server cluster to host database and data backup nodes. Layout the hardware needs and how it will be implemented.

Owner: Ryan Shepherd

Effort Estimate: 10

Actual Effort: TBD

Planned start and stop dates: 3/16/2020 - 4/6/2020

Actual start and stop dates: TBD

Dependencies among other tasks: Design of Network, Database, Security and Hosting

**Outline Budget**

Generate an estimated budget to Monstrosity Inc on the entire network setup, including hardware, labor costs, installation costs, hosting services fees, domain name providers, and all costs encompassed by network side IT. Not including the cost of employee machines and help desk.

Owner: Ryan Shepherd

Effort Estimate: 5

Actual Effort: TBD

Planned start and stop dates: 2/23/2020 - 4/20/2020

Actual start and stop dates: 2/20/20 - TBD

Dependencies among other tasks: Network, Database, Security and Hosting design

## *Release Plan*

This section details pertinent dates of when portions of this project will be completed and released to affected stakeholders.

**Iteration 1**:

Iteration Period: 2/17/2020 - 3/2/2020

Project Charter completion date: 2/23/2020

Component Needs Analysis completion date: 2/28/2020

Requirements Baseline completion date: 3/1/2020

**Iteration 2**:

Iteration Period: 3/2/2020 - 3/16/2020

Project Plan completion date: 3/8/2020

Project presentation initial draft/prototype (powerpoint format) completion date: 3/13/2020

Risk Report: 3/15/2020

**Iteration 3**:

Iteration Period: 3/16/2020 - 4/6/2020

Project Presentation to stakeholders: 3/20/2020

Architecture Document completion date: 4/3/2020

**Iteration 4**:

Iteration Period: 4/6/2020 - 4/20/2020

Final architecture report completion date: 4/20/2020

**Iteration 5**:

Iteration Period: 4/20/2020 - 5/4/2020

Project test plan completion date: 4/26/2020

User/System Guide completion date: 4/27/2020

Project results PowerPoint completion date: 5/1/2020

Github/Slack communication tasks completion date: 5/1/2020

Final presentation dates: 5/4/2020, 5/6/2020, 5/8/2020

**Iteration 6:**

Iteration Period: 5/4/2020 - 8/5/2020

Firewall setup - 6/10/2020

Completed network: 7/1/2020

Database connection: 7/15/2020

Email conversion: 7/20/2020

Testing: 7/1/2020 - 8/5/2020

## *Iteration Plans*

This section details each iteration and the tasks that will need to be assigned and completed.

* + 1. Iteration 1: Project Initiation, Requirements Specification and Needs Analysis (Iteration Period: 2/17/2020 - 3/2/2020)
* Group assignments by Professor Bingham
* Schedule and conduct initial meeting
  + decide roles
  + decide on communication routes and regular meeting schedules
  + brainstorm scope of project
  + delegate iteration 1 tasks
  + discuss everyone's strengths and weaknesses
  + assign components to own
* Complete Project Charter report
* Complete Requirements Baseline report
* Complete network needs analysis
* Complete database infrastructure needs analysis
* Complete security needs analysis
* Complete cloud services needs analysis
* Other tasks TBD
* Continue to meet and communicate regularly
* Complete iteration beginning documentation and upload to github
* Complete iteration close documentation and upload to github
  + 1. **Iteration 2**: Project Planning, Risk Assessment, and Initial Implementation Proposal draft (prototype) (Iteration Period: 3/2/2020 - 3/16/2020)
* Complete Project Plan report
* Complete Project Risks Report
* Complete PowerPoint slide deck outlining initial implementation
* Conduct research:
  + Network infrastructure implementation
  + Intersite network links
  + Networking hardware
  + Network configuration/settings
  + Security protocols of network, database, servers, etc
  + Physical security implementation
  + Intrusion detection and other cybersecurity systems
  + Cloud hosting service providers
  + Cloud hosted services
  + Cloud hosting cost comparison
  + Database design implementation
  + Database analysis
  + Server clustering
  + Server hardware
  + Other topics as needed
* Design rough implementation of network infrastructure
* Design rough implementation of server clustering infrastructure
* Design rough implementation of cloud services infrastructure
* Design rough implementation of security infrastructure
* Design rough implementation of database infrastructure
* Continue to meet and communicate regularly
* Complete iteration beginning documentation and upload to github
* Complete iteration close documentation and upload to github
* Other tasks TBD
  + 1. **Iteration 3**: Project Presentation and architecture design (Iteration Period: 3/16/2020 - 4/6/2020)
* Present initial implementation design to stakeholders
* Complete research into assigned components
* Use virtual tools to complete design diagrams and configuration schematics
* Write architecture plan detailing implementation of:
  + Network infrastructure
  + Database design
  + Server cluster infrastructure
  + Security infrastructure
  + Cloud hosting recommendations
  + Include any applicable design diagrams
  + Present comparisons of solutions to maximize budget
* Continue to meet and communicate regularly
* Complete iteration beginning documentation and upload to github
* Complete iteration close documentation and upload to github
* Other tasks TBD
  + 1. **Iteration 4**: Architecture implementation design finalized (Iteration Period: 4/6/2020 - 4/20/2020)
* Finalize architecture implementation report
* Select hardware to be used in the product implementation
* Make final recommendations to Monstrosity Inc.
* Continue to meet and communicate regularly
* Complete iteration beginning documentation and upload to github
* Complete iteration close documentation and upload to github
* Other tasks TBD
  + 1. **Iteration 5**: Project Wrap-up and closeout (Iteration Period: 4/20/2020 - 5/4/2020)
* Complete project test plan
* Complete user and system guide
* Complete project results assessment
* Create PowerPoint presentation
* Complete final presentation to stakeholders
* Finalize any additional documents that will accompany proposal
* Make sure Github repository is complete
* Make sure Slack use requirement is completed
* Complete final presentation to stakeholders
* Complete evaluations of project
* Any additional tasks TBD
* Continue to meet and communicate regularly
* Complete iteration beginning documentation and upload to github
* Complete iteration close documentation and upload to github

3.3.6 **Iteration 6:** Post-Project Physical Buildout (Iteration Period: 5/5/2020 - 8/5/2020)

* Purchase equipment
* Monstrosity Inc approves proposal
* Present final approved proposal to contracted build-out company (Windstream)
* Set up Network with Windstream
* Connect Database to network
* Transfer Emails to cloud

## *Budget*

* + 1. Monstrosity Inc budget for project (not including ongoing costs): $175,000

## Buildout Cost Estimates:

* Project manager: $100.00 an hour for 8 hours a week for 12 weeks = 48,000.00
* 2 Architects: $67.97 an hour for 8 hours a week for 12 weeks = 65,251.20
* Consulting expenses: $1,000
* Servers and cluster infrastructure: $15,000.00
* Network Equipment: $4,000.00
* Security Infrastructure: $15,000
* Misc. Hardware: $500.00
* Buildout Contractors: $2000
* First month cloud services: $2000

Total Labor, Hardware, Services Estimate = $152,751.20

Ongoing Cost Estimates:

* Full-time IT staff person (yearly wages): $55,000
* Ongoing IT management and Maintenance (yearly): $2510
* Security Licenses for approximately 200 users (yearly): $4800
* Cloud Hosting Services Email and Filter hosting for approximately 200 users (yearly): $24,000

Total ongoing annual costs = $86,310

* + 1. Iteration 1 Expense Report:
* Labor of Design Team:

Project Manager: $8000

Architects: $10875.20

* Consulting Fees Paid:

$150 - Database and server cluster design consultant

$200 - Network consultant

* + 1. Iteration 2 Expense Report: TBD
    2. Iteration 3 Expense Report: TBD
    3. Iteration 4 Expense Report: TBD
    4. Iteration 5 Expense Report: TBD
    5. Iteration 6 Expense Report: TBD

# **Control Plan**

This section details how quality work will be maintained throughout the project.

## *Monitoring and Control*

This section outlines key dates when reviews will be completed by the team and presented to sponsors for grading

Weekly – Team meeting. Project participants report status, progress and potential problems. Participants also go over what their next goal is.

3/2/2020 – Critical Design Review. Review of collected data.

3/8/2020 – The project participants present the Project plan to the project sponsor.

3/13/2020 – Executive Review. The project participants present current project status to project sponsor and senior executives.

3/16/2020 – The project participants present the Risk Management document to the project sponsor.

3/17/2020 – Critical Design Review. Review Iteration 2 tasks to ensure quality.

4/3/2020 – The project participants present architecture documents to the project sponsor.

4/6/2020 – Critical Design Review. Review of Iteration 3 tasks to ensure quality.

4/20/2020 – Critical Design Review. Review of Iteration 4 tasks to ensure quality.

4/26/2020 – The project participants present the test plan to the project sponsor.

4/27/2020 – The project participants present the user and system guide to the project sponsor.

5/1/2020 – The project participants present the project results to the project sponsor.

5/1/2020 – The project participants present their communication logs to the project sponsor.

5/4/2020 – Critical Design Review. Review of Iteration 5 tasks to ensure quality.

## *Project Measurements*

This section reviews how the team will measure the success of the project throughout the entire project timeline.

|  |  |  |
| --- | --- | --- |
| **Phase** | **Measurement** | **Source** |
| Iteration Planning | Record effort estimates for scheduled tasks  Update effort estimates for product features  Update estimated dates in release plan | Architects  Architects  Mgr |
| Iteration Closeout | Record actual effort for scheduled tasks  Record actual effort for product features  Upload documentation to Github | Architects  Architects  Mgr |
| Project Closeout | Archive project performance data in process database. | Mgr |

# **Supporting Process Plans**

This section details additional factors that will be addressed throughout this project.

## *Risk Management Plan*

This section provides a summary of technical and managerial risks of the project. A complete and full accounting of risks will be detailed in a separate Risk Management Report document.

The risk management plan will be maintained and reviewed at the beginning and close of each iteration during team meetings. The project manager will own updating the document with input provided by the architecture team. Risks will continue to be reviewed on an ongoing basis once the project is completed by the IT professionals who will be managing the system.

* + 1. Risk: Internet failure (ISP related)

Priority: Low

Probability of turning into a problem: 20%

Consequences if risk turns into a problem: System will go down with no access to resources across sites, decreased employee productivity

Actions to minimize risk: Accept risk

Contingency plan: Contact ISP and Network Management company to resolve problem ASAP

* + 1. Risk: Power Failure

Priority: Low

Probability of turning into a problem: 20%

Consequences if risk turns into a problem: Loss of data, loss of access to resources, decreased employee productivity

Actions to minimize risk: Avoid risk

Contingency plan: Have a backup power system in place that will kick in automatically and secure a backup of data resources.

* + 1. Risk: Unauthorized access to system

Priority: High

Probability of turning into a problem: 40%

Consequences if risk turns into a problem: Data integrity changed producing incorrect information being used in decision making, systems being held hostage for ransom, access to confidential information causing the company legal problems, etc

Actions to minimize risk: Mitigate risk

Contingency plan: Implement a full security suite of tools/systems that will detect and prevent unauthorized access before it happens. Once there is unauthorized access, have IT professionals follow a plan that details action steps to remedy the problem (for example, change passwords and rectify the hole in the system when access was gained)

* + 1. Risk: Unauthorized access within the system

Priority: High

Probability of turning into a problem: 50%

Consequences if risk turns into a problem: Data integrity changed producing incorrect information being used in decision making, systems being held hostage for ransom, access to confidential information causing the company legal problems, etc

Actions to minimize risk: Avoid Risk

Contingency plan: Set up stringent access control systems to avoid risk. Once there is unauthorized access, have IT professionals follow a plan that details action steps to remedy the problem (for example, change passwords and rectify the hole in the system when access was gained)

* + 1. Risk: Hardware failure

Priority: High

Probability of turning into a problem: 85%

Consequences if risk turns into a problem: Loss of access to system resources, employee downtime and decreased productivity, increased costs to repair or replace hardware.

Actions to minimize risk: Accept

Contingency plan: Contact IT professionals to repair failures ASAP

* + 1. Risk: Fire

Priority: Medium

Probability of turning into a problem: 20%

Consequences if risk turns into a problem: Damage to equipment and facilities, rendering equipment unusable and costs to replace equipment and costs due to downtime of resources.

Actions to minimize risk: Mitigate risk

Contingency plan: Implement fire control systems (sprinklers) to put out fire if it takes place. Avoid risk as much as possible by following industry standards of cooling systems and electrical systems safety standards.

* + 1. Risk: Overheating of server cluster

Priority: high

Probability of turning into a problem: 90%

Consequences if risk turns into a problem: Fire, shutdown of systems, loss of access to resources, company costs to loss of productivity, costs to repair/replace hardware, data loss

Actions to minimize risk: Mitigate risk

Contingency plan: Implement industry standard cooling systems to keep equipment at correct operating temperatures. Engage IT professionals ASAP if downtime occurs to repair equipment

* + 1. Risk: Cloud service provider changes terms of service

Priority: low

Probability of turning into a problem: 15%

Consequences if risk turns into a problem: Increased costs, loss of services, decreased employee productivity

Actions to minimize risk: Buy information

Contingency plan: Fully research and understand the services that cloud service providers do and do not provide. Have contracts in place to ensure consistent level of agreed upon services

* + 1. Risk: Database misuse

Priority: Medium-High

Probability of turning into a problem: 40%

Consequences if risk turns into a problem: Incorrect or inconsistent information being in database results in lowered employee performance which will impact the service their customers receive

Actions to minimize risk: Mitigate risk

Contingency plan: Have access control systems in place to control various levels of authorized access to make changes to the database. Have employees and managers trained on how to use the database correctly. Have an easy to use and intuitive front end UI for users of the database.

* + 1. Risk: Viruses, phishing, other malicious acts contracted through email

Priority: High

Probability of turning into a problem: 75%

Consequences if risk turns into a problem: Loss of access to systems, confidentiality breaches, unauthorized access to resources, employee downtime and reduced productivity, costs to repair damage

Actions to minimize risk: Mitigate risk, Transfer risk, avoid risk

Contingency plan: Have secure systems in place that will automatically detect and protect the system from malicious software. Have a robust firewall system in place. Have email security drills to educate employees on these risks and how to avoid them. Engage IT support professionals to repair damage.

* + 1. Risk: Monstrosity Inc reduces budget

Priority: Medium

Probability of turning into a problem: 25%

Consequences if risk turns into a problem: Increased project time as changes to requirements will affect proposal outcomes, increased costs due to increased project timelines requiring additional time spent on project re-working what has been completed, loss of quality of a robust system

Actions to minimize risk: Accept risk

Contingency plan: Have this risk in mind from the beginning when doing research and be prepared to implement alternative system recommendations.

* + 1. Risk: Monstrosity Inc cancels project

Priority: Medium

Probability of turning into a problem: 20%

Consequences if risk turns into a problem: Wasted time and costs of design team and Monstrosity Inc, increased exposure to risks of maintaining the current system

Actions to minimize risk: Accept risk

Contingency plan: Close project and turn what has been completed over to Monstrosity Inc.

* + 1. Risk: Monstrosity Inc changes requirements to project

Priority: Low

Probability of turning into a problem: 10%

Consequences if risk turns into a problem: Increased project time as changes to requirements will affect proposal outcomes, increased costs due to increased project timelines requiring additional time spent on project re-working what has been completed, loss of quality of a robust system

Actions to minimize risk: Accept risk

Contingency plan: Have this risk in mind from the beginning when doing research and be prepared to implement alternative system recommendations.

* + 1. Risk: Design team goes past schedule

Priority: High

Probability of turning into a problem: 70%

Consequences if risk turns into a problem: Increased costs exceeding budget

Actions to minimize risk: Buy Information, Accept, Mitigate, Transfer

Contingency plan: Adhere to deadlines and engage consultants as available to ensure timely completion

* + 1. Risk: Design team goes over budget

Priority: high

Probability of turning into a problem: 70%

Consequences if risk turns into a problem: Requires executive decision to continue project or make trade-offs to lower costs

Actions to minimize risk: Buy information, Accept, Mitigate

Contingency plan: Adhere to deadlines, conduct complete and accurate research to recommend the best solutions to stay within budget, research alternative solutions.

* + 1. Risk: Design team does not perform adequate research

Priority: high

Probability of turning into a problem: 50%

Consequences if risk turns into a problem: Any number of various problems could result, some foreseen and some unexpected. For example, possible compromised system security, loss of productivity, additional costs to rectify holes in the system, increased time to fix the issues, etc

Actions to minimize risk: Mitigate risk

Contingency plan: Perform complete research, research multiple alternative recommendations, engage consultants and other resources to verify accuracy

* + 1. Risk: Design team does not propose an optimal implementation recommendation in final report

Priority: high

Probability of turning into a problem: 50%

Consequences if risk turns into a problem: Any number of various problems could result, some foreseen and some unexpected. For example, possible compromised system security, loss of productivity, additional costs to rectify holes in the system, increased time to fix the issues, etc

Actions to minimize risk: Mitigate risk

Contingency plan: Perform complete research, research multiple alternative recommendations, engage consultants and other resources to verify accuracy

* + 1. Risk: Miscommunication within design team

Priority: medium

Probability of turning into a problem: 40%

Consequences if risk turns into a problem: Any number of various problems could result, some foreseen and some unexpected. For example, possible compromised system security, loss of productivity, additional costs to rectify holes in the system, increased time to fix the issues, etc

Actions to minimize risk: Mitigate risk, accept risk

Contingency plan: Conduct regularly scheduled meetings and progress reviews to make sure everyone is on the same page.

* + 1. Risk: Compromised private data

Priority: High

Probability of turning into a problem: 50%

Consequences if risk turns into a problem: Possible legal ramifications of disclosure of confidential information, increased costs to rectify the problem.

Actions to minimize risk: Mitigate risk

Contingency plan: Implement a full security suite of tools/systems that will detect and prevent unauthorized access before it happens. Once there is unauthorized access, have IT professionals follow a plan that details action steps to remedy the problem (for example, change passwords and rectify the hole in the system when access was gained)

* + 1. Risk: Vague project details (specifications and client requirements) leading to incorrect assumptions

Priority: Medium

Probability of turning into a problem: 70%

Consequences if risk turns into a problem: Incorrect assumptions leading to wrong proposal recommendations resulting in increased costs to rectify, reduction in security, efficiency reduction, and decreased productivity.

Actions to minimize risk: Buy information

Contingency plan: Ask questions of sponsor and stakeholders when anything is unclear, present multiple options to solve problem, discuss as team to come to consensus on decisions

* + 1. Risk: Design team being inexperienced in architecture design

Priority: High

Probability of turning into a problem: 80%

Consequences if risk turns into a problem: Bad grade, incorrect final recommendations in proposal leading to additional costs, lowered productivity, and increased time to rectify any problems

Actions to minimize risk: Buy information

Contingency plan: Ask questions of sponsor and stakeholders when anything is unclear, present multiple options to solve problems, discuss as a team to come to consensus on decisions. Conduct complete and thorough research.

## *Configuration Management Plan*

This section describes how changes to the project will take place, updated, managed, and maintained throughout the project period.

* All work is spearheaded by the owner of the task, with other architects reviewing and advising in review.
* Final budget will be laid out annually for recurring costs unless otherwise stated.
* Items will be sorted under their proper category: network, database, security, and cloud hosting. For example any work pertaining to the acquisition, setup and cost of routers and access points will be covered under network. If an item, such as database security, crosses between multiple categories it will be covered in one of the two or more.
* Included with each document is a change history of the document as the team has researched and added to it.

## *Verification and Validation Plan*

The Verification and Validation plan is specified as a separate documented located in the document repository at:

<https://github.com/umkc-cs-451-2020-spring/semester-project-group-17/blob/master/Verification%20and%20validation%20plan.docx>

## *Product Acceptance Plan*

The project will be deemed acceptable and a success with a final, complete report detailing the network lay out, database setup, cybersecurity measures, how and what applications will be cloud hosted, a complete budget and equipment requirement. This report should be in a state that could be presented to the hypothetical company if real, and be used to fully implement the network. This means that the report should outline realistic security protocols that are shown to help defend and protect other businesses. The network architecture should be based upon existing technology and proven methods. The database setup should be thoroughly laid out to construct a database within the constraints, but does not require specific layout of the tables and relations. Hosting should layout which third party providers will be used, fees, and how it will be integrated into the company network. All of the above research and design should also result in an implementable list of networking hardware and estimated implementation and operating cost. All should be completed before the final due date.