OneTeam **μProject**short line

**A micro project management tool**

David Blair

Christian Carducci

Juan Landaverde

Sang-Joon Lee

Jacqueline Pham

Srivathsa Rajagopal

June 3rd 2016

# **Table of Contents**

[Section A. μProject Background](#h.gexkng1roy73)

[Vision for μProject](#h.pnc9pd3mcelk)

[Summary of Discussion with Client](#h.7yq2q1lebbr)

[Target Population](#h.4ceto6yrkxk4)

[Strategy for Success](#h.1e2w7qbk9oxe)

[Challenges](#h.3hnl8fw6tz99)

[Client Feedback and Interaction During Development](#h.2vode8bsbra)

[Feasibility of μProject](#h.t481aqcnpqr3)

[Amending the Proposal](#h.9uapym3jn3k)

[Section B. Design and Development Activities for Application](#h.vx23hjh28tvo)

[Queued: Project Management Tool](#h.jcfy550nc4n)

[ChatNow: Chat system](#h.8zzsqo1hk0ps)

[Bugs: Issue Tracker](#h.belislv7ncsq)

[Section C. Risks Retirement](#h.gvtifywgqfrk)

[Section D. Work Plan](#h.bktzezggvzl3)

# 

# 

# **Section A. μProject Background**

## **Vision for μProject**

*Purpose of Application:* To provide capabilities to support a software project using Agile methodology through a basic project management tool.

*Major Functionalities:*

* Management Component - manages the project's requirements
* Communication Component - allows project team to communicate via real time chat
* Issue Management Component - manages the project’s issues or defects

*Requirements: Described in detail in Section B for each aspect of the application.*

Throughout this proposal each component mentioned above is described as a project within the application. This logically breaks down the development process (see Section B).

## **Summary of Discussion with Client**

After reviewing the written description provided, a meeting was held with the Client to discuss the details of the application. We addressed the scope of the project. It was agreed that the requested scope of work was far too large given the time available. To solve this, we identified elements of a successful project: basic functionality in each of the three major functional areas with additional elements added as time permits. The details of what basic functionality means and potential elements for additional functionality are outlined in Section B. Once basic functionality is complete, focus will shift to improving the project management component of the application. The functional features are inspired by the examples given by the Client, but in no case should it be assumed any feature will faithfully replicate an example.

## **Target Population**

The target demographic for this application will be small to medium sized (1 to 8 people) software teams using Agile methodology who need a basic tool to support the process.

## **Strategy for Success**

We will leverage the Agile software development process to improve the quality, process, and outcomes of the final product. It is anticipated that this process will help develop the project in manageable pieces and allow for the maximum amount of Client feedback during the development process. For each iteration, the team will design, develop, test, conduct code reviews, and revise the various components of the product. At the end of an iteration, a sprint review will be conducted, with Client involvement, to review progress, seek feedback, and set priorities for the next iteration.

This is not an entirely open ended process. A well-defined scope of work exists for development (see Section B). The project team will make their best effort in order to implement as many features as possible, but in the event that there is not enough time in the sprint, they will eliminate lower priority items. In addition, focus on the components of the application are not equally distributed. Each of the three components will be developed to a level of basic functionality and then development will concentrate on the project management and bugs components of the system, with greater emphasis given to project management.

## **Challenges**

The major challenges the project team faces are time and scope. The project team is limited to a 12-week time period. In that time period, the team must ramp up on new technologies in order to implement μProject. The other constraint that we face is scope. Given the time constraint we have, it will be challenging to implement all the must have, and additionally the wants/nice-to-have features.

## **Client Feedback and Interaction During Development**

The Client of the project is Professor Czik. The project team will consult with Professor Czik in order to generate user stories for requested features at the start of each iteration, based on the content of this proposal. The requirements throughout this initiative are somewhat flexible as new design considerations arise or needs change. Feedback will be requested early and frequently form Professor Czik to ensure the project team is delivering the components outlined in the proposal. At the end of each iteration, the software will be demonstrated in class to Professor Czik for feedback.

During the first iteration, Chris (ccard@bu.edu) will coordinate all communication related to development of the application and the Client should direct all messages to them. In order for the Client and project team to ensure visibility and alignment, the Client will have access to the project team’s code repository and will receive weekly meeting minutes and quad reports. This role will be rotated in each iteration to follow.

## **Feasibility of μProject**

Given the time constraint challenge, unforeseen circumstances may lead to reduced scope. The project team will consult with the Client in order to encourage that the must-haves of the project be implemented first. Having the basic functionality of each component (requirements, chat, and issue tracking) will be the goal of our initial iteration in order to have a minimum viable product to demo to the Client.

## **Amending the Proposal**

This proposal represents the entire understanding between the Client and Students with respect to development of the application. The proposal is a commitment by the Students and the Client with respect to the work to be completed and the timelines on which it will be completed. Both parties are aware of the challenges discussed as well as issues related to feasibility of μProject.

To ensure clarity and fairness to everyone involved, this proposal may be amended only by written agreement of both parties. The Project Team representative will change each iteration, for the first iteration the point of contact is Chris Carducci.

# **Section B. Design and Development Activities for Application**

The application (μProject) is comprised of three relatively separate components wrapped in a single web application. The elements are described below as three separate projects. The descriptions outline the project team, goal of the project, and address the background, description of activities, and our proposed activities by iteration. We propose to build all three components using Node.js as the foundation for the backend, with MySQL as the database server, and Bootstrap as the front-end framework. Documentation will be hosted using Read The Docs <https://readthedocs.org>. Details of the Risk Retirement can be found in Section C.

Each project leader will manage communication for the related team and coordinate with the iteration leader to ensure a cohesive effort. In general, communication will be conducted through Slack, code will be committed to GitHub, and documents will be collaboratively edited using Google Docs. Development requirements will be tracked in Pivotal Tracker until the point at which we can use μProject to track the requirements.

|  |  |
| --- | --- |
| Project Title | Queued: Project Management Tool |
| Project Team | Juan Landaverde, Sang-Joon Lee, Srivathsa Rajagopal, Chris Carducci, Jacqueline Pham |
| Project Goal | To create a project requirements management tool for Agile software development projects which allows user to create a project, stories and manage lifecycle of software development tasks. |
| Project Timeline | May 2016 - August 2016 |

**Project Background**

Working on a team towards the same goal can become a challenge. This challenge can be solved by performing Agile software development which addresses a problem by dividing it into smaller pieces that then can be put together. This allows multiple individuals to work on a given piece and work towards the same goal. Software can support the Agile process by helping individuals and teams structure the development process. As such, requirement management is an essential process for software development which involves documenting, analyzing, tracing, prioritizing and agreeing on requirements such that software meets expectation of its customer and stakeholders. μProject provides fundamental capabilities to create, edit, analyze, track and manage requirements for software development.

**Project Objectives**

|  |
| --- |
| 1. Develop foundation of the Project Management Tool |
| 2. Implement primary and secondary functionality |
| 3. Pilot Project Management Tool Tracking system for quality and finalize documentation |

**Project Description**

The Queued project requirement management tool is a simplified tool to create, update, assign, track and prioritize requirement for software development. Queued will be based on successful software such as PivotalTracker, having similar fundamental project management features. The development of Queued will begin with focus on core functionalities such as the ability to create a new user, a new Project and a new Story with set of attributes such as assigning tasks to owners and setting priorities with editing capabilities. In addition, the tool will allow user to update status of a Story such that it can be promoted from Backlog view to Current view, and then to Completed view. Functionality to further support the system may be implemented, as time permits, after the core functionalities are implemented. A summary of prioritized list of functionalities for the project requirement management tool is shown below.

Must haves (functionality necessary for success)

* Create a user
* Create a Project: main project with multiple individuals/users
* Create, edit, delete Stories: small pieces of work or features
  + Ability to assign a story to a user for development
  + Ability to assign and update status of a story throughout development cycle
* Backlog of Stories: list of prioritized stories that can be worked on, ability to re-order stories
* Current Stories: list of prioritized stories that are currently under development
* Completed Stories: list of stories that are completed

Wants (functionality to further support system)

* Create Iterations: Given timeframe in weeks which contain stories to be completed by team.
* Assign type of story (feature, bug or release)

Nice to haves (potential functionality)

* Create and manage multiple projects: ability to create, edit and delete multiple projects
* Search Stories
* View stories per individual.
* Email reminders for Story deadlines
* Project History
* Progress charts and analytics

**Project Objectives by Iteration**

*Objective 1. Develop foundation of the Project Management Tool (Iteration 1).*

Week 1 will be dedicated to gathering the requirements from the Client, analyzing the requirements and translating into design. The proposed requirements will be analyzed, ranked and prioritized prior to development. In addition, the development environment will be finalized, identifying the necessary tools and complete setup with version control system as well as the database schema will be designed. In Week 2, a basic implementation of the project requirement management tool will be the focus, followed by initial testing. By end of the first iteration, a user will be able to create a new Project, create a new Story with ability to add details as needed including assigning task to an owner responsible for development. Initial testing phase will begin by setting up whole team with working test environment, and perform testing of must-have features. A code review will be conducted to ensure overall quality. Any issues or bugs raised during initial testing will be address prior to demonstrating the features to the Client. The project team will seek feedback from the Client via demonstration and revise requirement based on the feedback.

*Objective 2. Implement primary and secondary functionality (Iteration 2).*

In Week 3, feedback from the demonstration will be translated into further requirements. These requirements will be then prioritized, implemented as necessary, and tested. In addition, the remaining Must-Haves functionalities will be designed, implemented, and tested; such as ability to promote a task from Backlog to Current view, ability to update status of Stories in Current view as per development life-cycle, and ability to view Stories in one of three lists: Backlog, Current, or Completed/Done. In Week 4, other remaining items from Wants and Nice-to-Have list will be added to the Backlog, prioritized, designed, and implemented. In this phase, if there are any outstanding needs (Must-Haves), these tasks will be given highest priority. In Week 5, all features will be thoroughly tested by development and QA. A code review will be conducted to ensure overall quality. Documentation will also begin and reviewed during this iteration phase. The project team will make their best effort in order to implement as many features as possible, but in the event that there is not enough time in the sprint, they will eliminate the lower priority items. Progress will be demonstrated for feedback from the Client.

*Objective 3. Pilot Project Management Tool for quality and finalize documentation (Iteration 3).*

In iteration 3, we will have implemented all of the features to be included in the final version, dedicating this iteration strictly to fine-tuning such as eliminating bugs, making the product more polished, and making the documentation comprehensive. In Week 6-7, all project requirement management system components will be tested by development and QA team. In addition, the project requirement management system will be tested by a randomly selected software team for usability testing. In Week 7-8, we will conduct pilot testing with three to five individuals not involved in project development asking them to use the Queued system and provide qualitative feedback about their experience. In Week 8-9, the project team will document and resolve any issues that arise from pilot and usability testing. In Week 9, the documentation will be finalized. The end product will be demonstrated to the Client.

|  |  |
| --- | --- |
| Project Title | ChatNow: Chat system |
| Project Team | Chris Carducci, David Blair, Juan Landaverde, Srivathsa Rajagopal |
| Project Goal | To create an easy-to-use communication environment that allows a team to share ideas in a virtual chat room setting as an alternative to emails. |
| Project Timeline | May 2016 - August 2016 |

**Project Background**

Communication through email is not ideal for projects. Team members can communicate more effectively by using a dedicated chat application rather than trying to dig up emails. Communication is critical to the success of a project, and this tool will enable it by allowing team members to effectively share ideas, provide feedback, and identify risks in advance in an easy-to-use, dedicated environment.

**Project Objectives**

|  |
| --- |
| 1. Develop foundation of the Chat System |
| 2. Implement primary and secondary functionality |
| 3. Pilot Chat System for quality and finalize documentation |

**Project Description**

A simple chat system is based on channels and the ability to look at past discussions. It is the fundamental element of the user experience that must be right. As we refine the chat system it may include features such as private messaging and searching to increase utility and value to the user. Ultimately, this tool will remain simple, however as time permits it may include features such as file sharing and audio/visual chat.

Must haves (functionality necessary for success)

* Create a channel - ability to create a public chat room
* Archive chat history - make chat history persistent (not session based, but always visible)

Wants (functionality to further support system)

* Send private messages to other members
* Search - ability to search through the chat history for keywords

Nice to haves (potential functionality)

* Share files
* Audio/Video chat

**Project Objectives by Iteration**

*Objective 1. Develop foundation of the Chat System (Iteration 1).*

In Week 1, requirements will be gathered and the design will begin, identifying the necessary tools and database schema. In Week 2, a basic implementation of the chat system will be the focus, followed by initial testing. Basic functionality should include the ability to create a channel, chat with others, and see persistent chat histories.

*Objective 2. Implement primary and secondary functionality (Iteration 2).*

After iteration 1 has been presented to the Client, Weeks 3-5 will be spent by translating feedback into further requirements, designing, and implementing, and testing. Additionally, further functionality will be implemented and tested, such as: the ability to send private messages, and the ability to search. If there is still time, the nice-to-have features will be explored (sharing files and audio/video chatting). In Week 5 a code review will be conducted before demonstration of the project to the Client.

*Objective 3. Pilot Chat System for quality and finalize documentation (Iteration 3).*

In iteration 3, we will have implemented all of the features, dedicating this iteration (weeks 6-9) strictly to fine-tuning (eliminating bugs, making the product more polished, etc.) and making the documentation comprehensive. In Week 7-8, we will conduct pilot testing with two to three individuals not involved in the application development and ask them to use the chat system and provide qualitative feedback about their experience. In Week 9 we will make any necessary revisions based on the feedback. The end product will be demonstrated to the Client.

|  |  |
| --- | --- |
| Project Title | Bugs: Issue Tracker |
| Project Team | David Blair, Jacqueline Pham, Sang-Joon Lee, Chris Carducci |
| Project Goal | To develop a basic tracking system to create, assign, update and resolve issues. |
| Project Timeline | May 2016 - August 2016 |

**Project Background**

Issue tracking is essential to a software team using Agile methodology as it is inevitable that applications contain defects. μProject requires an issue tracking system in order to manage and maintain the ongoing list of issues. Such issue tracking system will help the application users work towards ensuring quality software. As defects are found, the tool will allow for the capability to be logged and then assigned a priority and severity, in order to allow the application users to properly triage the issues.

**Project Objectives**

|  |
| --- |
| 1. Develop foundation of the Issue Tracking System |
| 2. Implement primary and secondary functionality |
| 3. Pilot Issue Tracking System and finalize documentation |

**Project Description**

The Bugs issue tracker is a simple tool to create, assign, and prioritize issues. This is an opinionated tool, it guides users into a specific methodology of handling issues. Bugs will be based on similar successful software (e.g. Github issues, Assembla) while focused on the fundamental set of features.

Bugs development begins with the core functionality. The ability to create, edit, update, and destroy issues which contain a basic description of the problem. The tracker will allow users to set a priority, severity, and record who submitted the issue originally. This functionality embraces the core necessities of any issue tracker. To ensure these methods are executed in a useable way, we will test other issue tracking software in order to identify best practices and then implement the Bugs issue tracker based on those findings.

Functionality to further support the system can be added, as time permits, after the base functionality is implemented. The ability to post comments on an issue, assign responsibility for the issue, and attach files related to the issue are the first priority. These components are common in many software programs but not necessary to successfully track and resolve issues. Less common features of issue tracking software may also be incorporated as the project progresses based on progress and priority. These elements include functionality such as tagging, issue-project management linking, favoriting, issue history, and custom fields.

A summary of the primary elements for the issue tracking project is shown below.

Must haves (functionality necessary for success)

* Create, edit, update, and delete issues
* Set priority, severity and status of issue
* Identify who reported issue

Wants (functionality to further support system)

* Comment on issue
* Assign issue responsibility to user
* Attach files to issue

Nice to haves (potential functionality)

* Tag issues and search by tags
* Link issue to Project Management goal
* Favorite issue
* Maintain a history of changes to issue
* Custom fields

**Project Objectives by Iteration**

*Objective 1. Develop foundation of the Issue Tracking System (Iteration 1).*

The Week 1 project staff will review existing products in the same category to identify common design patterns, draft an initial concept design, create user stories, and confer with the Client in order to create a minimal viable product. Development will begin after the features for the issue tool are ranked and prioritized. In Week 2, basic functionality for the issue tool will be designed and implemented. By the end of the iteration, the ability to create a new issue and assign it will exist. The ability to set a priority, severity and status will also be present. A code review will be conducted to ensure overall quality. The project team will seek feedback on the implemented features from the Client at the Sprint Demo at the end of the iteration.

*Objective 2. Implement primary and secondary functionality (Iteration 2).*

Based on the feedback at the Sprint Demo from the Client, project staff will re-evaluate the priorities and assess the state of the project to identify any requirement changes in Week 3. We expect any outstanding basic needs of the application will be assigned a higher priority and implemented first. In Weeks 3-5, new features will be added, based on priority agreed upon with the Client as time is available. It is anticipated that features will first draw from the functionality to further support the system list first and then from potential functionality list as time permits. All features will be tested and documented as they are developed. In Week 5, the Bugs project shown at the Sprint Demo will represent all features to be included in the project. Client feedback will be sought to ensure the quality and clarity of the product.

*Objective 3. Pilot Issue Tracking System for quality and finalize documentation (Iteration 3).*

In Week 6, project staff will trial the components of Bugs in-house first to ensure that the system is functional, but also useable. After making revisions to the product based on this testing, the issue tracking component will be tested as part of the entire application by the team for usability. The usability testing will give us insight into how users interact with the system. In Week 7-8, pilot testing will be conducted with three to five individuals not involved in project development. These individuals will be asked to complete certain tasks using the system and feedback will be sought through qualitative interviews. Pilot testers involvement will be anonymous, voluntary, and they will receive no compensation for their participation. Based on feedback from the usability testing, any necessary revisions to the system will be implemented in Weeks 7-9. During this same time period, documentation will be finalized and any necessary release notes will be written. The end product will be demonstrated to the Client.

# 

# 

# **Section C. Risks Retirement**

|  |  |  |  |
| --- | --- | --- | --- |
| **Identified Risk** | **Impact** | **Probability** | **Risk Response** |
| Technical Expertise with Node.js is limited causing features to be delayed. | High | Medium | Those on the team that aren’t familiar with Node.js will be given a demo by Chris. Additional online training will be taken by project team members without experience in Node.js |
| Timeline for the requested features is very short and may result in reduced scope | High | High | The project team will focus on must-have items first and ensure that Client is aware of status at all times in order to communicate that some features may not be implemented until the next iteration. |
| Communication Method is new to the team, creating potential for missed communications and therefore delaying project status | Medium | Low | The team has tried to keep all communication to Slack and has organized channels based on streams of work. Therefore, all communication is easily referenced in one place. The team has created communication expectations and will be available through Slack on a close to daily basis. |
| Roles are component specific which may make it difficult for developers to jump in on a component that they are not familiar with | Medium | Low | The components will have a lead so there is one person that will be knowledgeable and leading that effort. If there are questions, this resource will be the to-go, mitigating any concerns for that component by assisting those who are not as familiar. |

#### 

# **Section D. Work Plan**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **WORK PLAN/PERSON LOADING** | **DB** | **CC** | **JL** | **SL** | **JP** | **SR** |
| ***Project Activity Responsibilities:*** |  |  |  |  |  |  |
| **Project 1. Queued: Project Management Tool** |  |  | **XX** |  |  |  |
| *Objective 1.* Develop foundation of the Project Management Tool |  |  | X | \* |  | \* |
| *Objective 2.* Implement primary and secondary functionality |  | \* | \* | X | \* | \* |
| *Objective 3.* Pilot Project Management Tool for quality and finalize documentation |  |  | \* | \* | \* | X |
| **Project 2. ChatNow: Chat System** |  | **XX** |  |  |  |  |
| *Objective 1.* Develop foundation of the Chat System |  | X | \* |  |  |  |
| *Objective 2.* Implement primary and secondary functionality | \* | X |  |  |  |  |
| *Objective 3.* Pilot Chat System for quality and finalize documentation |  | \* |  |  |  | X |
| **Project 3. Bugs: Issue Tracker** | **XX** |  |  |  |  |  |
| *Objective 1.* *Develop foundation of the Issue Tracking System* | X |  |  | \* | \* |  |
| *Objective 2.* *Implement primary and secondary functionality* | \* | \* |  |  | X |  |
| *Objective 3.* *Pilot Issue Tracking System for quality and finalize documentation* | \* |  |  |  | X |  |

**Personnel: DB** (Blair) **CC** (Carducci) **JL** (Landaverde) **SL** (Lee) **JP** (Pham) **SR** (Rajagopal)

**Key: X** indicates primary responsibility for a task **XX** indicates project lead **\*** indicates task involvement