

THE UNIVERSITY OF WINNIPEG

Dept. of Applied Computer Science

ACS-4901

Senior Systems Development Project

Prove IT – Project Proposal

Team Members

<u>Name</u>	<u>ID</u>
Arshjot Ghuman	3114007
Liam Kristjanson	3116156
Usmaan Sahar	3065852
Vi Le	3119710
W A Shadman	3132520
Xiao Zhang	3131956

IS Director: Victor Balogun

Date: 25 September 2023

Table of Contents

1. Executive Summary	3
2. Project Overview	3
3. Agile Methodology Selection.....	3
4. Project Scope.....	5
5. Project Team and Roles	8
6. Timeline and Sprints.....	10
7. Product Backlog.....	11
8. Sprint Planning and Execution.....	12
9. Daily Standups	12
10. Continuous Integration and Testing.....	13
11. Retrospectives and Continuous Improvement	13
12. Risk Management.....	14
13. Budget and Resources	15
14. Benefits and Expected Outcomes.....	15
15. Proposal Approval and Signatures.....	16

1. Executive Summary

HomeTrumpeter has innovated a novel approach to property management through a centralized platform for a multitude of home services. They set out to create an API economy, allowing other platforms to utilize this streamlined workflow. Our team is tasked to construct a platform built on the foundations of the HomeTrumpeter API gateway, in order to "Prove IT" works.

The project is split into EPICs corresponding to a HomeTrumpeter system, each containing tasks derived from real-world user stories. Each user story will revolve around one main API endpoint or microservice sold by HomeTrumpeter. Aggregation of results allows for analysis of commonly and uncommonly used API endpoints, further allowing HomeTrumpeter to optimize resource allocation within their API economy.

Through this project, HomeTrumpeter will be able to stress-test each component of their microservices in a real-world scenario.

2. Project Overview

The main purpose of this project is to develop an interface that wraps around the HomeTrumpeter API gateway.

We will design and build a web interface in React that will port over to iOS and Android using React Native, integrating into the HomeTrumpeter API gateway. A CI/CD workflow will be developed using GitLab to streamline the Agile process, allowing for fast and safe releases. Deployment will be done using OpenShift Kubernetes Platform—containerizing systems to ensure maximum compatibility throughout development.

While the interface may not be deployed to production, the expected outcome is the evaluation of the feasibility of the API economy created by HomeTrumpeter. From this point, HomeTrumpeter may re-evaluate or continue development of this system.

3. Agile Methodology Selection

3.1 Agile

As an agile team, we will be adopting the Scrum framework for this project. Adopting Scrum will allow us to focus on continuous learning and process improvement through the project's duration. Learning and improvement will be particularly important for our development team, as we are all young, inexperienced developers. The ability to share and

reflect on lessons we have learned and implement changes based on them will be paramount to the project's success.

As we are building an entirely new front-end app leveraging an existing API, it will be crucial that we solicit regular feedback from stakeholders to ensure that the product meets their needs. The sprint review will facilitate such feedback well. After the completion of each two-week sprint, we will meet with our stakeholders to demonstrate the progress we have made. The stakeholders will then be able to provide feedback, allowing us to prioritize and adjust requirements for the next sprint accordingly.

Following the sprint review, we will hold a sprint retrospective amongst ourselves. This meeting will allow us to candidly share challenges faced and lessons learned during the previous sprint. By documenting what processes worked well, and which ones need improvement, we will be able to continuously improve our performance as the project progresses. As a team taking on our first software project as a unit, there will be many lessons learned. Scrum's built-in framework for sharing such lessons at a regular interval will prove especially useful for our specific circumstances.

3.2 Scrum

The Scrum methodology allows teams to respond to changes, collaborate with flexibility, and manage work in progress. The basis of Scrum follows the scientific process of hypothesizing, testing, and adjusting our approach. This workflow enables us to understand the status of ongoing work and encourages communication with stakeholders.

Scrum fits perfectly in our project as it permits us to incorporate the frameworks used by HomeTrumpeter. Additionally, 2–4-week iterations allow for ample time to develop each sprint and to receive timely feedback from the stakeholders.

4. Project Scope

4.1 General Scope

The "Prove IT" project for HomeTrumpeter aims to develop a versatile platform that interfaces with the HomeTrumpeter API gateway. The scope encompasses the following key components and functionalities:

1. Web Interface Development:

- Design and construct a web interface using React.
- Plan for seamless porting of the web interface to iOS and Android platforms through React Native integration with the HomeTrumpeter API gateway.

2. Continuous Integration/Continuous Deployment (CI/CD) Workflow:

- Establish a robust CI/CD workflow utilizing GitLab.
- Implement Agile development practices to enable rapid and secure software releases.

3. Deployment Infrastructure:

- Utilize the OpenShift Kubernetes Platform for deployment to ensure compatibility throughout the development process.

4. Feasibility Evaluation:

- Evaluate the feasibility of HomeTrumpeter's API economy.
- Assess the potential for further system development and refinement based on the outcomes.

5. User Interface (UI) Development:

- Develop UI flows for key features, including:
 - **Login:** Allow users to access the platform securely.
 - **Tenant Onboarding:** Facilitate the onboarding process for Tenants.
 - **Service Provider Onboarding:** Streamline the registration process for Service Providers.
 - **Background Checks:** Implement background checks for Service Providers.
 - **Service Provider Behavior Prediction:** Explore predictive capabilities for Service Providers' behavior.

6. Collaboration and Methodologies:

- Collaborate through regular stand-up meetings.
- Maintain effective communication with an Agile coach.
- Incorporate feedback from stakeholders to drive improvements.

7. Agile Methodologies:

- Implement Scrum and Kanban methodologies to emphasize continuous learning, adaptability, and feedback-driven improvements throughout the project lifecycle.

8. Success Metrics:

- Measure project success based on the achievement of milestones and the realization of project objectives outlined in each sprint.

4.2 Entities and Tasks

Homeowners:

- **Sign-up Process:** Enable Homeowners to create accounts for property management.
- **Add Properties:** Allow Homeowners to add their properties to the platform.
- **Invites Tenants:** Enable Homeowners to invite Tenants to their properties.
- **Approves Tenant:** Provide Homeowners the ability to approve Tenant applications without requiring background checks.
- **Invites Private Service Provider:** Allow Homeowners to invite Private Service Providers to offer services.
- **Allocate Task to Service Provider:** Facilitate task assignment from Homeowners to Service Providers.
- **Status of Ticket:** Enable Homeowners to track the status of maintenance tickets.

Private Service Providers:

- **Accepts Invitation:** Allow Private Service Providers to accept invitations from Homeowners.
- **Sign-up:** Enable Private Service Providers to create accounts on the platform.
- **Submit Maintenance Ticket:** Allow Service Providers to receive and respond to maintenance requests.
- **Status of Ticket Update:** Enable Service Providers to update the status of maintenance tickets.

- **Mark Job Completed:** Allow Service Providers to mark maintenance jobs as completed.

Tenants:

- **Accepts Invitation:** Enable Tenants to accept invitations from Homeowners to rent properties.
- **Sign Up:** Allow Tenants to create accounts and complete the rental process.
- **Submit Maintenance Ticket:** Enable Tenants to report maintenance issues.
- **Track Status:** Allow Tenants to monitor the status of their maintenance requests.
- **Confirm Job Completed:** Enable Tenants to confirm when maintenance tasks are completed.

4.3 Dependencies

- The Sign-up Process for Homeowners, Private Service Providers, and Tenants may involve the use of Signal Wire Tokens or other authentication methods.
- Homeowners must complete the Sign-up Process before performing other tasks.
- Homeowners can invite Tenants and Private Service Providers once their properties are added.
- Private Service Providers need to accept invitations before receiving maintenance tickets.
- Tenants must accept invitations and complete the Sign-Up process before renting properties and submitting maintenance tickets.
- The Status of Ticket update is dependent on maintenance tickets being submitted.

4.4 Note

The project scope encompasses the development of this platform, including user interfaces, functionality, and integration of necessary authentication methods, to support the described workflows and interactions between Homeowners, Private Service Providers, and Tenants. The platform aims to streamline property management and service provision while maintaining user privacy and security.

- Homeowners have the option to approve Tenant applications requiring background checks.
- The platform facilitates the allocation of maintenance tasks from Homeowners to Service Providers.
- Users can track the status of maintenance tickets to stay informed about ongoing maintenance activities.

5. Project Team and Roles

Collaboration within the development team will be facilitated through stand-up meetings, 3 times per week. Further communication will occur through weekly/bi-weekly meetings with the Agile coach and various stakeholders, as required by the respective sprint. Additional weekly check-in meetings will take place with the information system director to ensure smooth operations. All project members will be invited to collaborate on the Microsoft Teams platform, facilitating communication and file sharing.

Product Owner: Liam Kristjanson

- Create and maintain project backlog
- Define a global vision of the product
- Facilitate communication between stakeholders and development team to plan product releases

Scrum Master: Liam Kristjanson

- Ensure the development team follows the Agile process
- Oversee and support the development team using the Scrum methodology

Developers: Everyone

Technical Leader: Xiao Zhang

- Coordinate technical requirements between the stakeholders and project team
- Set up and oversee all technical infrastructure required for development and deployment

Lead Developer: Vi Le

- Oversee and support the development team
- Develop and enforce programming standards

Lead Quality Control Specialist: Arshjot Ghuman

- Ensure adherence to established standards (e.g., Documents, code, presentations, etc.)
- Lead and manage periodic walkthroughs of the codebase

Lead Systems Analyst/Designer: Usmaan Sahar

- Conducts information gathering from users
- Analyze user requirements and system requirements
- Design plans to facilitate development

DevSecOps Engineer: W A Shadman

- Implement automated security checks within the continuous development and deployment of the product
- Identify and address potential vulnerabilities throughout the design and development process
- Collaborate with developers to ensure that security protocols are always followed

Agile Coach: Victor Balogun

- Oversee multiple Agile teams across the organization
- Provide mentorship for implementing Agile principles
- Provide training material pertaining to Agile principles

Industrial Mentor: Gabriel O

- Provide guidance relating to industrial technologies and practices.
- Provide training materials relating to required technologies.

Stakeholders: HomeTrumpeter Team & Users

- Provide user requirements
- Provide pre-existing technical infrastructure
- Test and provide feedback on product releases

6. Timeline and Sprints

Sprint Number	Sprint Objective	Start date	Finish date
1	Create Project Proposal	6/9/2023	25/9/2023
2	Integrate into Sponsor System	25/9/2023	9/10/2023
3	Create CI/CD pipeline using GitLab	11/10/2023	25/10/2023
Project Plan Deadline			6/10/2023
4	Develop Login profile using API Gateway	27/10/2023	10/11/2023
System Study Review Deadline			13/11/2023
5	Develop Tenant onboarding	12/11/2023	26/11/2023
6	Develop Service Provider onboarding	28/11/2023	12/12/2023
7	Develop background check and notification	14/12/2023	28/12/2023
8	Develop service request creation	30/12/2023	13/1/2024
9	Research Service Provider behavior prediction for a service offering	15/1/2024	29/1/2024
10	Develop Web Interface	31/1/2024	14/2/2024
Detailed Design Review Deadline			31/1/2024
11	Develop User Interface	16/2/2024	1/3/2024
12	Develop frontend for IOS and Android	3/3/2024	17/3/2024
Complete System testing and Development			20/3/2024
Get Sign-off from user			20/3/2024
Project Completion Seminar			22/3/2024
Sign-off Course Completion Checklist			1/4/2024

7. Product Backlog

The backlog will focus on enforcing all the epics and accommodating the three major users who the application is targeted for. These are the Homeowners, the Service Providers, and the Tenets.

The beginning of the backlog for will be developed around base functionality and setting up the login portal to the API gateway. This is a top priority for the first sprint as it will allow current members to access their properties/services and allow new members to sign up and join when needed. No other epic can be covered until we address the base functionality.

Once base functionality is established, the backlog will then focus on the second major requirement of onboarding and ensuring that users can properly look through and access the tools necessary to manage their properties or services. These will focus on the homeowners and their ability to add and remove properties, set up service providers and inviting tenants. Tenants becoming familiar with the app and being able to communicate with their landlords is essential, as if there is a problem, they should be able to reach out with little issue.

This will tie into the next two requirements of Service provider onboarding and background checks. A service provider can be anyone who will be providing a service to the property (contractor, electrician, etc.) and will need to properly set up their profile to ensure that the offered service is being displayed. It is also important to vet the service providers so that they are verified for the service they are providing as well as their general conduct.

Once a service provider is onboarded and verified, a tenet will need to put in a request for something in their rented property. Making sure that the ability to create a service request, so that the tenet can inform the landlord of a problem and the landlord can dispatch the proper service is the next step. These service requests will directly involve all three major parties and will need to be meticulously detailed to ensure no mistakes are made when putting in a request.

Finally, we are asked to investigate creating a service provider's prediction for the service offered. This will be complicated and will need to rely on the previous background checks and experience the service provider has had with other tenets. With the users established and the goal of each epic in mind, a proper backlog can be created by making sure each requirement is met and expanded upon for greater functionality.

8. Sprint Planning and Execution

A Sprint Planning Meeting will be held at the beginning of every sprint. During this meeting, our team will select work items from the top of the prioritized backlog to be completed in the upcoming sprint and assign them to members of the team. The amount of work selected will be based on our story point estimations. We will select an amount of work roughly equal to the number of story points completed in the previous sprint. Tasks will be assigned based on team roles, as well as each team member's affinity for the required work.

The result of the sprint planning meeting will be a set of product-based objectives we hope to achieve during the sprint. The assignment of specific tasks will detail how we plan to reach these objectives. Agile methodology is centered around responsiveness to change and acceptance of the unknown. We will be developing a set of objectives we strive to reach during the sprint, rather than creating a detailed day-by-day plan of work to be completed. In doing so, we allow ourselves to be responsive to the inevitability of change.

Using our story point estimations, we will be able to create a 'Burndown chart.' This chart will allow us to see the speed of work being completed compared to the required speed to complete all planned tasks on time. This will be a valuable tool during our sprints to monitor if our sprint plan was realistic, or if action is required to keep the project on time.

9. Daily Standups

Our team will conduct face-to-face Standup meetings every Monday, Wednesday, and Friday at 1:00PM. These will be short, informal meetings where each team member will share what they have worked on the previous day, what they plan to work on the next day, and any issues blocking their progress. These meetings will encourage accountability, as each team member will need to share how much progress they have made during the last day. Agile methodology encourages face-to-face interaction whenever possible. Such interactions result in fewer errors in communication, improving the quality and efficiency of the project. These regularly scheduled standups are our method of implementing regular face-to-face interaction.

On Tuesdays and Thursdays, our team will adopt an asynchronous scrum methodology. All team members will share the same information as would be shared in a standup meeting through our Microsoft Teams chat. As an agile team, we will value continuous learning and seek process improvement. This search for process improvement will extend to our scrum practice as well. With each sprint retrospective, we will explore whether the standup process could be improved in any way. We will be open to meeting face-to-face at a different frequency, in a different format, or at a

different time. This will allow us to respond to unique needs as the project progresses, changes in team members' schedules, or any other unexpected change. This responsiveness to change is a key pillar to success in agile methodology.

10. Continuous Integration and Testing

For this project we will employ a robust Continuous Integration (CI) process using tools like GitLab for version control, alongside CI servers such as Jenkins. This ensures every code push is automatically built and tested, promoting frequent, smaller updates and early detection of integration issues. Automated testing will be a keystone, with unit tests, component tests, and end-to-end tests using tools like React Testing Library. Monitoring tools will be used to ensure high code coverage, guaranteeing consistent quality throughout the codebase.

In our Agile sprints, testing is an intrinsic part. During sprint planning, testing requirements will be defined alongside feature breakdowns. We will encourage Test-Driven Development (TDD), making sure that tests are written before the actual feature code. A feature will be considered "done" only when it meets its functional requirements and passes all its tests. Regular sprint reviews and retrospectives will ensure that we constantly assess and refine our testing and integration methodologies, ensuring continuous improvement in quality and process efficiency.

11. Retrospectives and Continuous Improvement

At the conclusion of each 2-week sprint, we will hold a sprint review with our team and sponsors, followed by a retrospective attended by just the team. The review will demonstrate concrete progress in the software to our stakeholders and solicit feedback that could be acted on in the next sprint. On the other hand, the purpose of the retrospective will be to share what worked well in the sprint, discover processes or interactions that could be improved, and to commit to concrete plans of action according to these findings.

Each member of the team will be strongly encouraged to share a positive aspect of the previous sprint, and an area of improvement. The team will then be able to find common themes among positive experiences and identify pain points that could be removed. This will allow each team member to have a say in the process changes enacted in the following sprint. The Scrum Master will be responsible for organizing and facilitating this meeting. This candid, self-organizing, regularly occurring retrospective will facilitate the continuous learning and improvement sought by agile teams.

12. Risk Management

12.1 Risk Identification

Identifying potential risks and challenges and outlining a plan to monitor, assess, and mitigate them is a crucial component in the modern software development approach. The following is how we as a team will be implementing the best practices and industry standards to maintain security in every phase of the software development lifecycle to ensure the secure and efficient development and deployment of this project:

A thorough analysis of the project will be conducted, including the API which we will be using and the requirements of our front-end app. We will engage with stakeholders and members of our team to identify potential risks and challenges. We would also conduct threat modelling and categorize risks in a way that best defines them, such as technical, resource-related, scope-related, and external factors. Common risks in this project might include API changes, unclear requirements, technology dependencies, background checks for service providers, etc.

12.2 Risk Monitoring and Assessment:

After identifying potential risks, we would assess their potential impact on the project's timeline, budget, and quality. In addition, we will consider both the probability of the risk occurring and the severity of its impact and assign each risk a risk score based on these factors. For example, high-risk items should be prioritized for mitigation planning.

We will also implement strategies to actively monitor risks throughout the project. This could involve regular risk assessment meetings, using risk management tools, and maintaining an up-to-date risk register.

12.3 Risk Mitigation Strategies:

For each identified risk, we would propose specific mitigation strategies. These strategies should outline how we plan to prevent the risk from occurring or how we intend to minimize its impact if it does occur. Mitigation strategies can include actions like regular communication with the API provider, establishing contingency plans, and setting up a change control process for API changes.

12.4 Risk Management in Agile

Since risk management is an ongoing process throughout the project's lifecycle, it should be integrated seamlessly into our agile development methodology. The flexibility of agile methodology in responding to risks and changes allows for continuous planning and adaptation which we will incorporate for our risk management into our agile processes, such as sprint planning and retrospectives. By addressing potential risks proactively and having a well-defined risk management plan, we would increase the chances of a successful project outcome.

13. Budget and Resources

The resources we have access to are all free, primarily using GitLab, React/React Native, Kubernetes, Figma, and the private cloud provided by HomeTrumpeter. Most of the members working on this will be using any of these tools and will have access to additional help in the form of mentoring from the people at HomeTrumpeter.

While no express budget is needed, as we are not using any services that require a fee, the most ambitious tool we are using will require machine learning. This will potentially require additional resources at a further time once the specifics are decided during the later sprints.

14. Benefits and Expected Outcomes

With the agile methodology in practice, we can better prepare and adapt for any future changes or necessary adjustments that need to be implemented. We can also document the process a lot better with each sprint creating slowly going towards our expected outcomes.

We will often collaborate amongst ourselves and our sponsors and discuss improvements or redundancies that need to be accounted for during the project. It will further our ability to work well together as a team and allow us to develop experience with new concepts that need to be implemented.

The expected outcomes of this project will result in a better understanding of the industry, as well as some worthwhile experience with the agile approach. It will also allow us to identify issues we may experience now in the future more clearly and give us a proper understanding of the number of resources and work needed for projects of a similar nature.

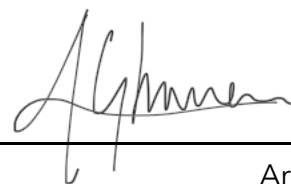
We will expect to push a proper application that functions as intended by the end of the collaboration and hopefully also make some worthwhile additions to the previous software in hopes of improving it the best we can.

15. Proposal Approval and Signatures

15.1 HomeTrumpeter LLC.

Adeyinka Owotuyi
Director of Product Design and Management
HomeTrumpeter LLC

15.2 The University of Winnipeg



Arshjot Ghuman
Lead Quality Control Specialist
The University of Winnipeg



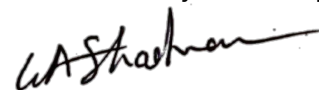
Liam Kristjanson
Project Leader/Scrum Master
The University of Winnipeg



Usmaan Sahar
Product Owner and Design Analyst
The University of Winnipeg



Vi Le
Lead Developer
The University of Winnipeg



W A Shadman
DevSecOps Engineer
The University of Winnipeg



Xiao Zhang
Technical Leader
The University of Winnipeg