Detailed Design Document

for the

Makerspace Automation Suite

Team GOATR

Jasmine Brown, Thomas Coe, Dana Koutsky,
Craig Owenby, & Ryan Strat

Table of Contents

List of Figures

List of Tables

Glossary of Terms

G Suite

Formerly known as Google Apps for Nonprofits, G Suite provides directory services to the Makerspace Automation Suite.

MAS

Initialization of Makerspace Automation Suite - The Wordpress plugin described within this document.

Onboarding

The process of adding a new member to an organization.

Slack

Slack is a community messaging service built upon topic-based discussion rooms and direct messages between users.

Stripe

Stripe is a payments processing service that handles secure payments for merchants.

Wordpress

Wordpress is an open source website content management system (CMS). As the world's most popular CMS, it has a thriving ecosystem of themes and plugins.

Wordpress Plugin

Wordpress plugins are a modular system for adding pages and functionality to sites hosted on Wordpress. Throughout this document, the word *plugin* refers to a Wordpress Plugin.

1. Introduction

Makerspace Automation Suite (MAS)

The Makerspace Automation Suite is a plugin for Wordpress websites that provides a host of convenient and useful features to assist makerspaces around the United States in automating their onboarding workflows for new members. For this project, Team GOATR is working closely with Decatur Makers, a family-friendly makerspace based in Decatur, GA.

The Decatur Makerspace's onboarding workflow is currently 100% manual, requiring the intervention of an administrator at each step of the process. *Figure 1.1* on page 2 demonstrates their current workflow.

The purpose of the Makerspace Automation Suite is to reduce the required steps that the administrators must take in order to successfully add a new member to the makerspace organization. Instead of the administrator having to perform nine intervening steps, they simply need to wait for the information required to start the background check to be received, wait for the background check to be successfully completed, and schedule member orientation. This represents a 300% decrease in the number of intervening steps. The new automation workflow is demonstrated in *Figure 1.2* on page 2. The Makerspace Automation Suite also provides the administrators of the makerspace with a list of members and access to detailed member information. In addition, the Makerspace Automation Suite gives current members a central location to update their personal information such as email address and payment method.

The Detailed Design Document

This document will cover Team GOATR's design decisions and scoping for this project. After reading this document, one should have a complete understanding of how the Makerspace Automation Suite interacts with its various subsystems and components and be able to perform any troubleshooting or maintenance that may arise after the development of this system. Additionally, this document will serve as a starting point for anybody who wishes to further develop the MAS and increase its functionality.

Specifically, this document will explain:
- The higher-level architecture of the system, including its third-party components and interactions
- The format in which data is stored and accessed by the MAS
- The interactions between low-level components of the MAS and how they perform specific functionality
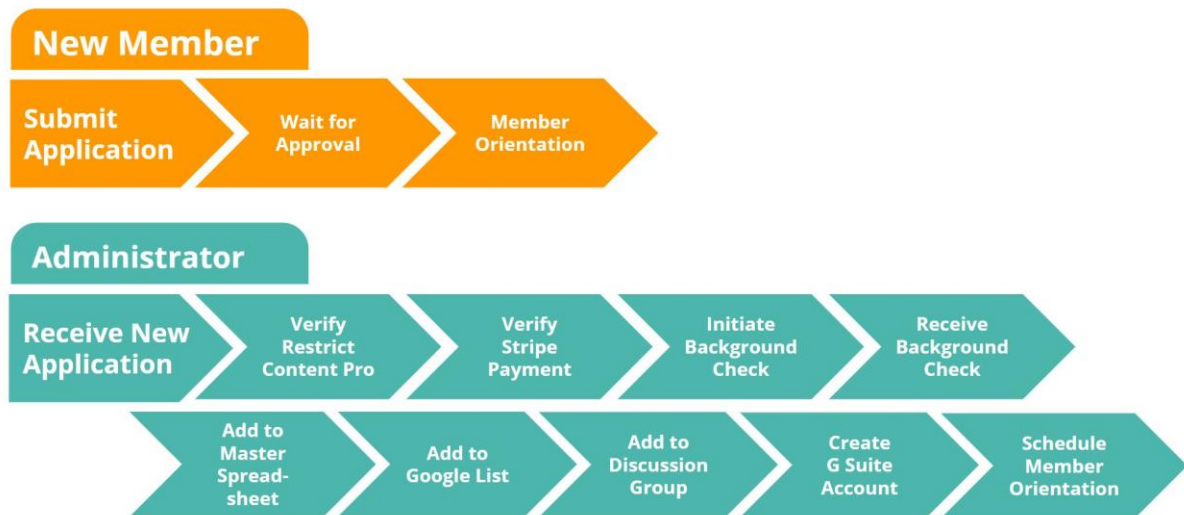- The user interface design and decisions made by Team GOATR

New Member

Submit Application → **Wait for Approval** → **Member Orientation**

Administrator

Receive New Application → **Verify Restrict Content Pro** → **Verify Stripe Payment** → **Initiate Background Check** → **Receive Background Check**

Add to Master Spread-sheet → **Add to Google List** → **Add to Discussion Group** → **Create G Suite Account** → **Schedule Member Orientation**

Figure 1.1: Previous Decatur Makers Onboarding Workflow

New Member

Submit Application → **Wait for Approval** → **Member Orientation**

Administrator

Receive New Application → **Makerspace Automation Suite** → **Initiate Background Check** → **Receive Background Check** → **Schedule Member Orientation**
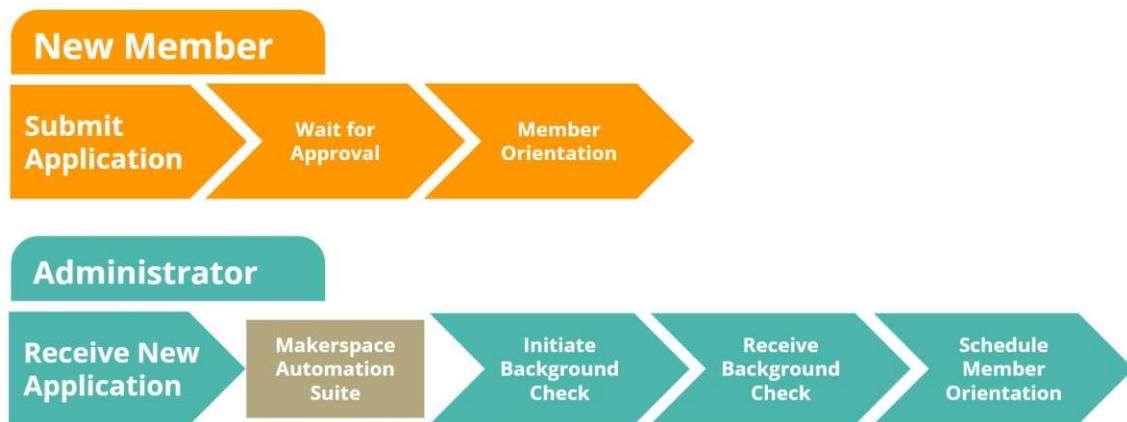
Figure 1.2: Decatur Makers Onboarding Workflow with Makerspace Automation Suite

2. System Architecture

Static System Architecture Diagram

Figure 2.1 details the relationship between the user and the Makerspace Automation Suite. All components connected with a dotted arrow are automatically initiated by the Makerspace Automation Suite. All components connected by a solid line are initiated by human interaction. Boxed components indicate external systems that are interacting with the Makerspace Automation Suite whereas the figures indicate human interaction within the system.
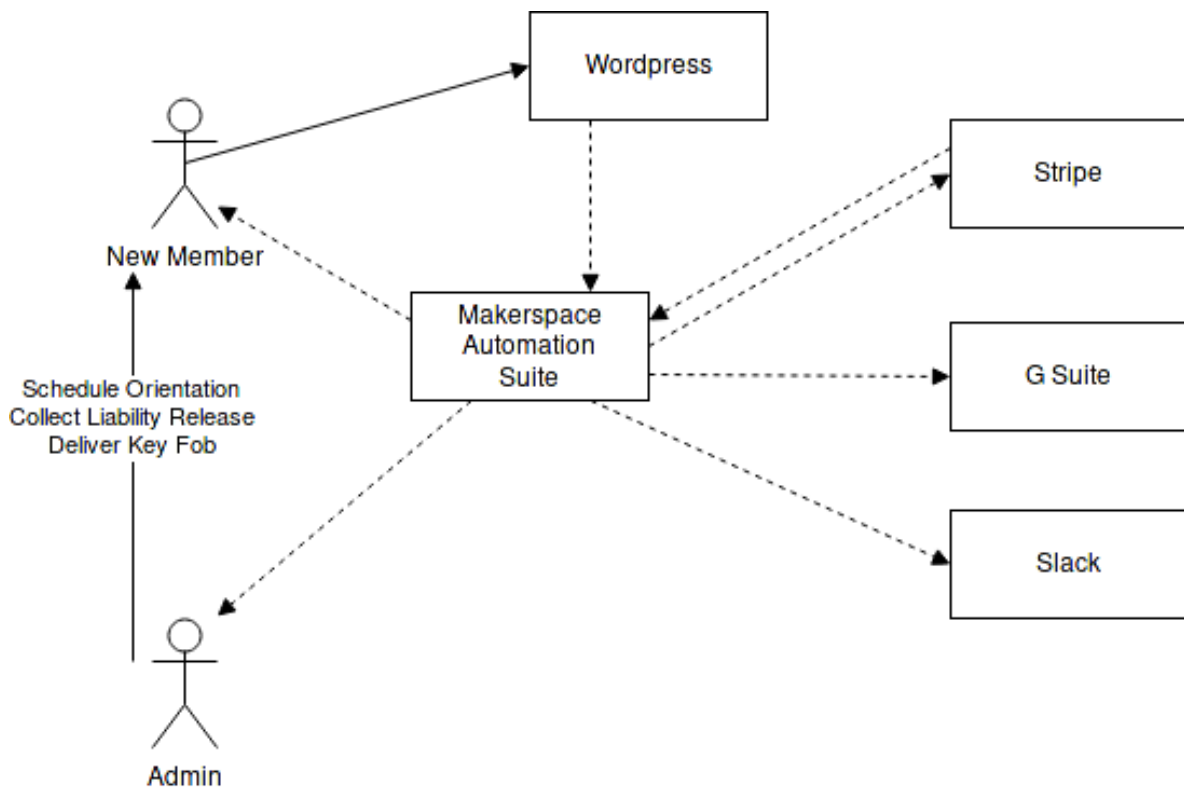
Figure 2.1: Static System Architecture Diagram of Onboarding Process

Dynamic System Architecture Diagram

Registering As A New Member

Figure 2.2 illustrates the steps that the systems goes through once a new user registers to become a member of the makerspace.
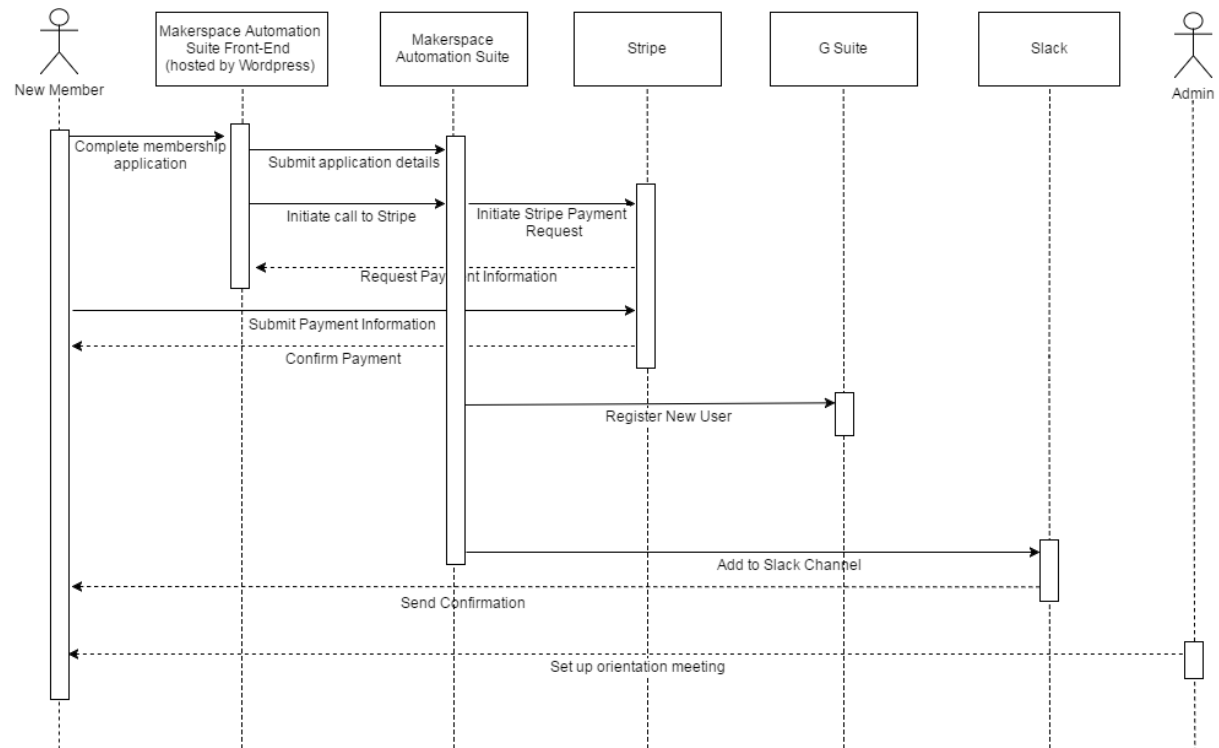

Figure 2.2: Dynamic System Architecture Diagram of Onboarding Process

3. Data Storage Design

G Suite Directory API

All information on each user is stored within the G Suite Directory API. Most of the information that must be retained, such as name and contact information, can be stored in the pre-existing fields. For information that is more specific, like subscription information or access entitlements, custom schemas are added. All data is stored as nested JSON objects in a NoSQL-like database. In *Table 3.1*, the mapping from default field names to an application specific description of content is shown. Any fields described in the Google API Documentation that are not listed below are not utilized by the Makerspace Automation Suite.

Basic Information

Table 3.1: Basic User Contact Information

Property	Data type	Description
name	nested object	User's full name
name.familyName	string	User's last name
name.givenName	string	User's first name
password	string	The user's password. Must be at least 8 characters long.
primaryEmail	string	User's G Suite email. It takes the form of username@domain.tld, where username is the user-selected user ID, and domain.tld is the domain used by the Makerspace's G Suite organization
emails[].address	string	Secondary emails. Email addresses associated with the user that do not end with the organization's domain name. These are the email addresses used for sending notification emails to members.

Subscription Management

Table 3.2: Subscription Management Information

Property	Data type	Description
Subscription_Type	string	Type of subscription the user has at the makerspace
Subscription_Status	string	Contains information on the current status of the user's account. Legal values: ● pending - account has been created and is awaiting administrator approval ● active - account is approved and paid ● expired - account is disabled due to lack of payment
Subscription_Expiration	string	Date of subscription expiration in the format YYYY-MM-DD
Subscription_Recurring	boolean	True if the user opted in to automatic billing of the membership each month
Stripe_ID	string	Stripe customer ID. Links to the corresponding customer object at Stripe for managing payments

Roles

The Makerspace Automation Suite uses an entitlements-based permission system. Roles can be created or read by any aspect of the software for determining whether a user has a particular authorization. There is no limit on the number of roles that a user can possess.

Table 3.3: User Roles

Property	Data type	Description
permissions	string[]	Array of strings, where each string represents a role
rfid-id	int	RFID tag number assigned to the member. This should be the public facing number and not the number embedded in the chip. This number will be publicly queryable to enable integration with the door access system.
founding-member	boolean	True if the user is a founding member of the makerspace

Configuration Data Storage

The Makerspace Automation Suite stores the external API access keys in the Wordpress options store. API keys and administrative notification email addresses can be updated from within the Wordpress settings menu under "MAS Options".

4. Component Detailed Design

Static Component Detailed Design

The Makerspace Automation Suite is developed with the Model-View-Controller architecture. The model is contained within the third-party services GSuite and Stripe, whereas the GSuiteAPI.php, StripeAPI.php, and SlackAPI.php files are the controllers, and the HTML files are the views.
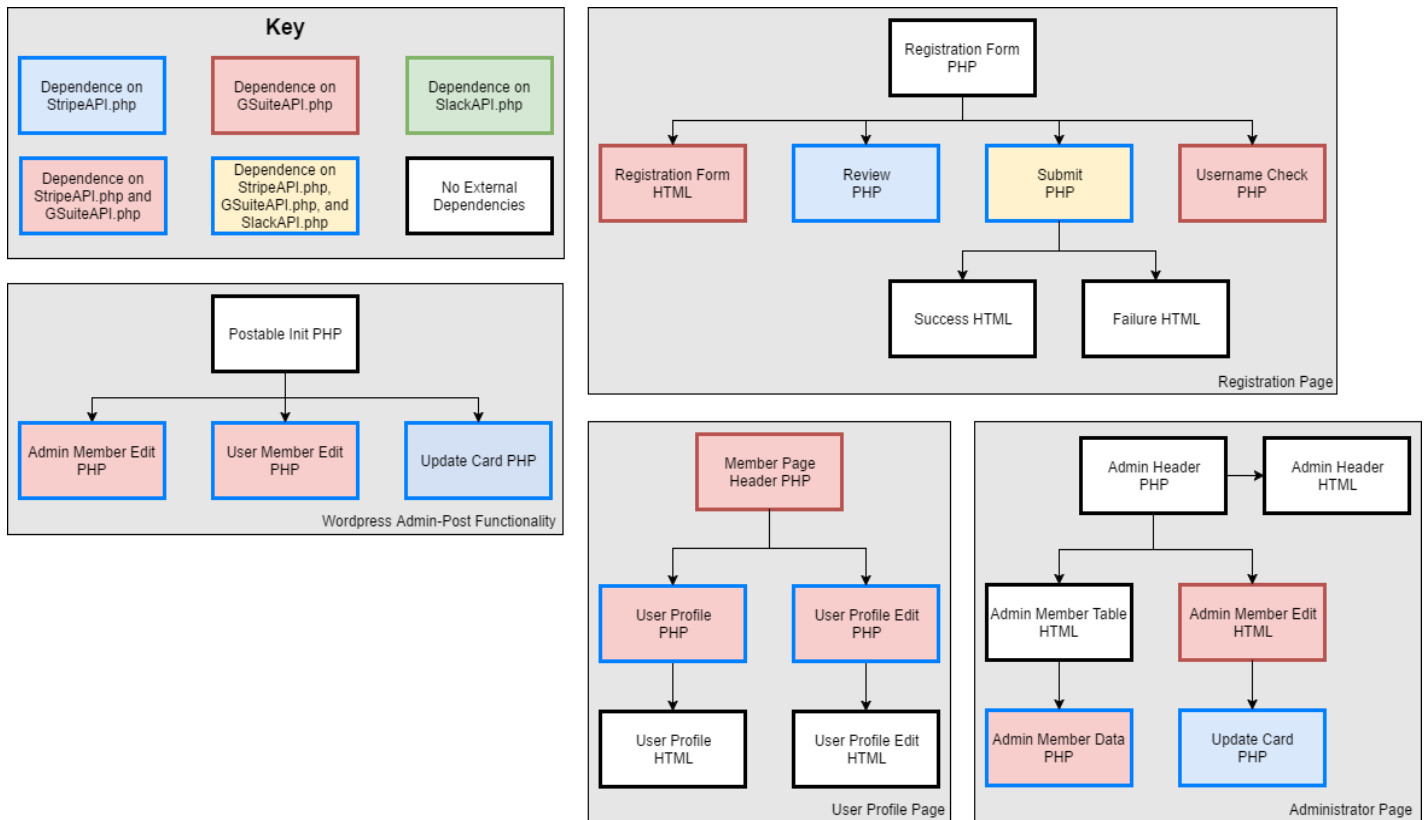

Figure 4.1: Static Component Diagram for MAS

For the interactions between different components of the Makerspace Automation Suite, each arrow represents a direct dependency between components. Additionally, there are three API interface files: GSuiteAPI.php, StripeAPI.php, and SlackAPI.php. These API interface files are wrappers for the official GSuite PHP, Stripe PHP, and the Slack APIs. In essence, they allow easier interaction with the APIs for the developers creating the front-end components.

Dynamic Component Detailed Design

Admin Updating User Profile

Figure 4.2 demonstrates the interactions between the files during a user interaction with the system. In this scenario, the admin is updating user profile information. The Admin Header PHP file serves as a controller for the interaction, sourcing different HTML files to display the data to the admin.
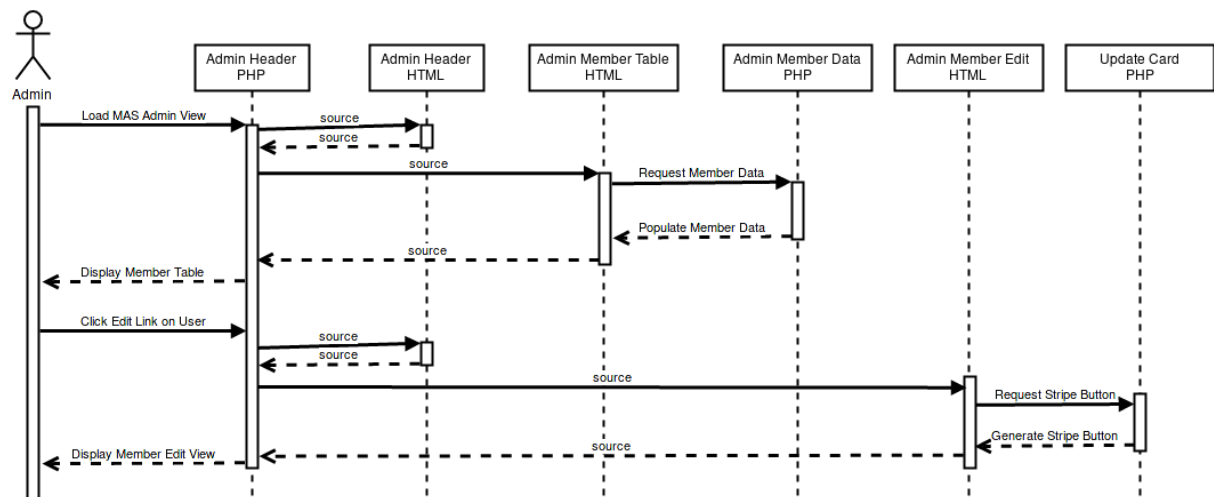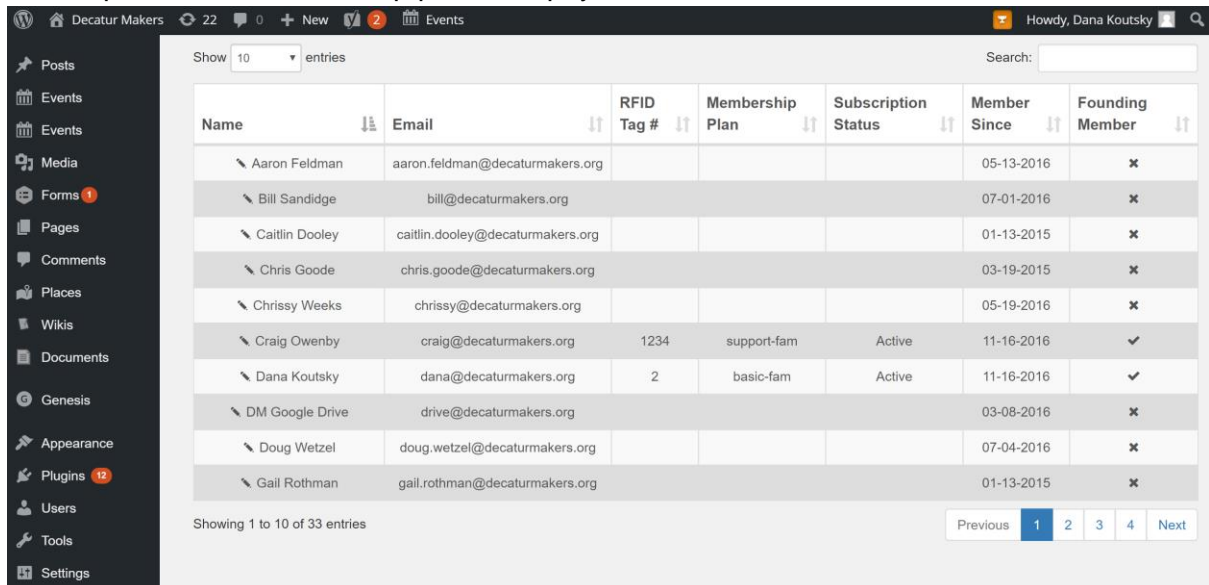
Figure 4.2: Dynamic Component Diagram for Admin Updating User Profile

5. UI Design

The UI design is broken into two parts: the Admin View which is secured and hosted within the Wordpress admin pages, and the Member View which is located on the client facing Wordpress site.

Admin View

Figure 5.1 shows the page that allows the administrator to view all of the members of the makerspace, their membership plan, and payment date.

Figure 5.1: User Interface Mockup for Administrator Members Tab

Figure 5.4 shows the page where a makerspace administrator can manage the integrations that the makerspace uses.

Figure 5.4: User Interface Mockup for Administrator Integrations Tab

Member View

When a member logs into the Decatur Makers website, they will be shown this member profile page (*Figure 5.5*). Here, a member can view and edit their personal information, change their password, and update their payment information.

Figure 5.5: User Interface Mockup for User Homepage