**SACLI-Q**

**USER**

**MANUAL**

**JPCS**

**BSIT-3**

**Almaden, Justine Jude**

**Maningas, Genesis Johann**

**Dumaop, Andrei**

**Abrea, Jude Ivan**

**Del Monte, Reiner**

**Alvarez, Mark Louie**

**SY:2024-2025**

**System Setup**

# Prerequisites

Before setting up the project, make sure the following tools are installed on your system:

* **XAMPP** – for Apache and MySQL server
* **Composer** – PHP dependency manager
* **Visual Studio Code** – recommended code editor

# Installation Guide

## 1. Download the Project Source Code

- Visit the GitHub repository: <https://github.com/JudeAlmaden/SACLI-Q>

-Click the green "Code" button and select "Download ZIP"

-Extract the ZIP file

-Move the extracted folder into your **xampp/htdocs directory**

-Open the project in **Visual Studio Code**

## 2. Configure the Environment

-Update .env File

-Open the .env file in the root of the project and update the following values:

APP\_URL=http://192.168.1.12:8000 🡨Assign with device local IP  
  
DB\_CONNECTION=mysql 🡨Your database connection  
DB\_HOST=127.0.0.1  
DB\_PORT=3306  
DB\_DATABASE=Sacli\_Queue  
DB\_USERNAME=root  
DB\_PASSWORD=  
  
REVERB\_HOST=192.168.1.12 🡨Assign with device local IP

-Update vite.config.js. Locate and update the vite.config.js file with your local IP address:

server: {  
 host: '0.0.0.0',  
 port: 8880,  
 strictPort: true,  
 hmr: {  
 host: '192.168.1.12' 🡨Assign with device local IP  
 }

## 3. Application & Database Setup

-Open a terminal in Visual Studio Code (inside the project directory)

-Run the following commands one by one:

*composer install  
php artisan migrate:fresh  
php artisan db:seed  
php artisan storage:link*

**NOTE: The Admin login credentials are defined in the seeder file SACLIQueue/database/seeders/DatabaseSeeder.php**

## 4. Running the Program

-Open three new terminals in Visual Studio Code  
-Run the following commands in separate terminals:

*“php artisan serve --host=0.0.0.0 --port=8000”  
“php artisan reverb:start --host=0.0.0.0 --port=8080”  
“npm run dev”*

The application should now be up and running on your local network

**User Guide**

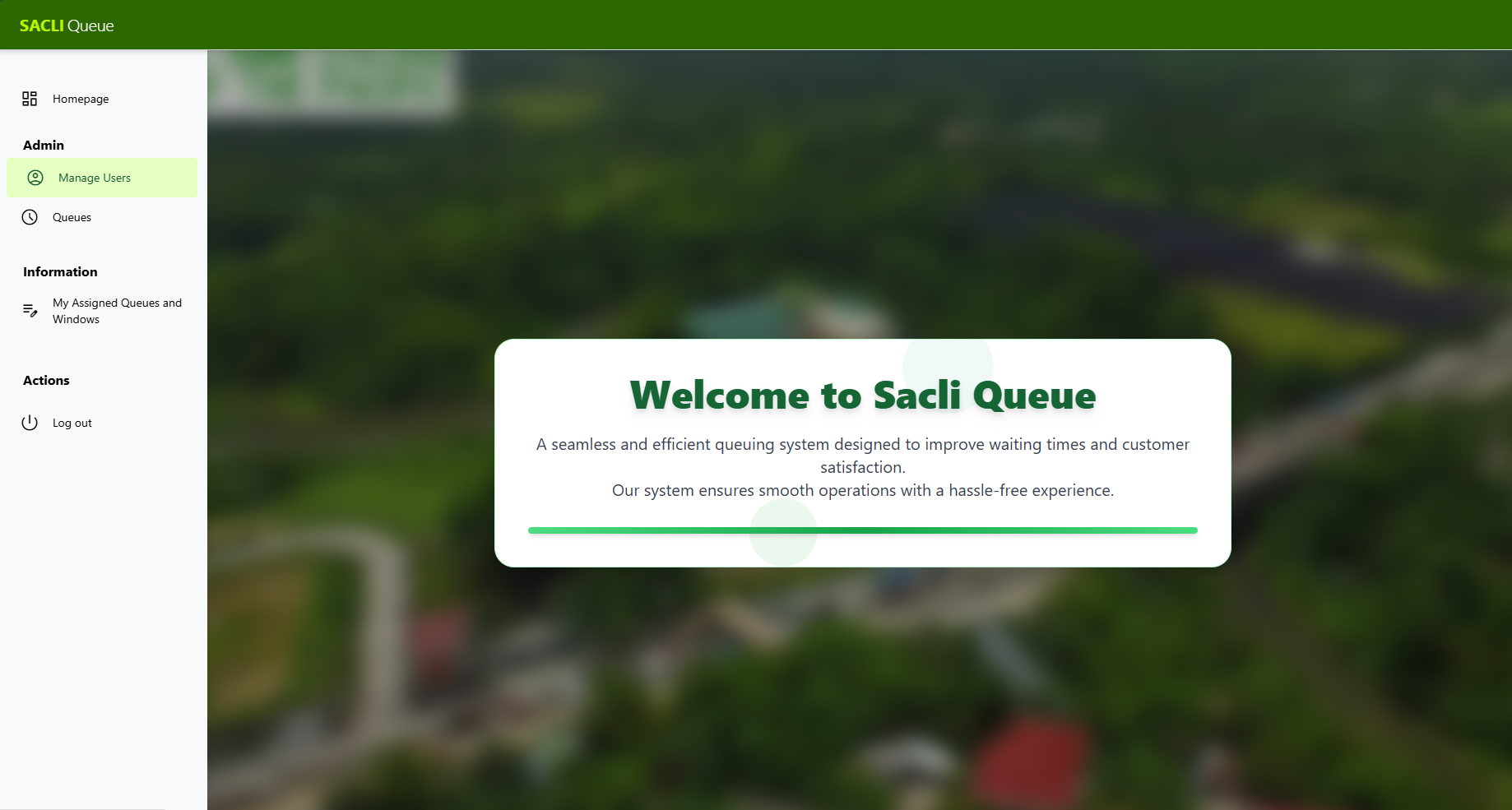
# Logging in

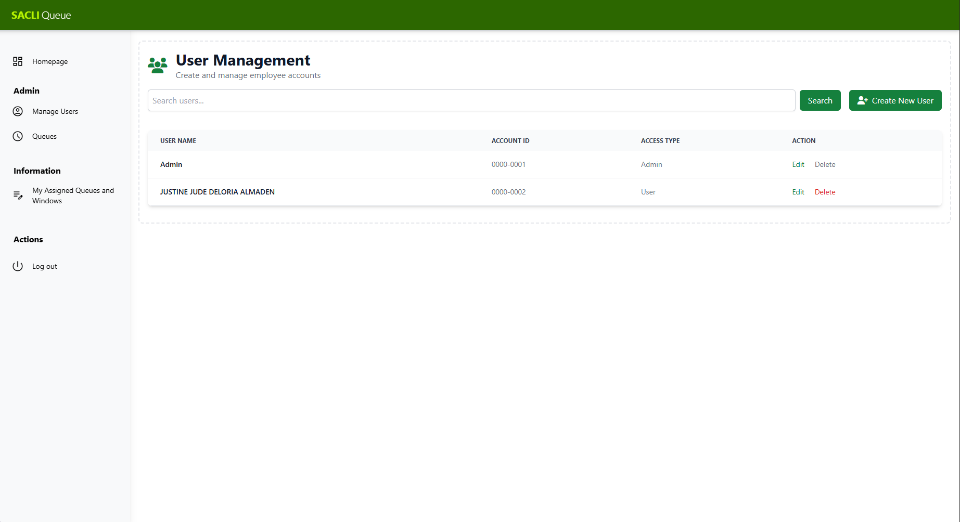
The login page is found at <http://192.168.x.x:8000/Sacli-Q.com/login>. Where in the IP and port is host server on the local network.

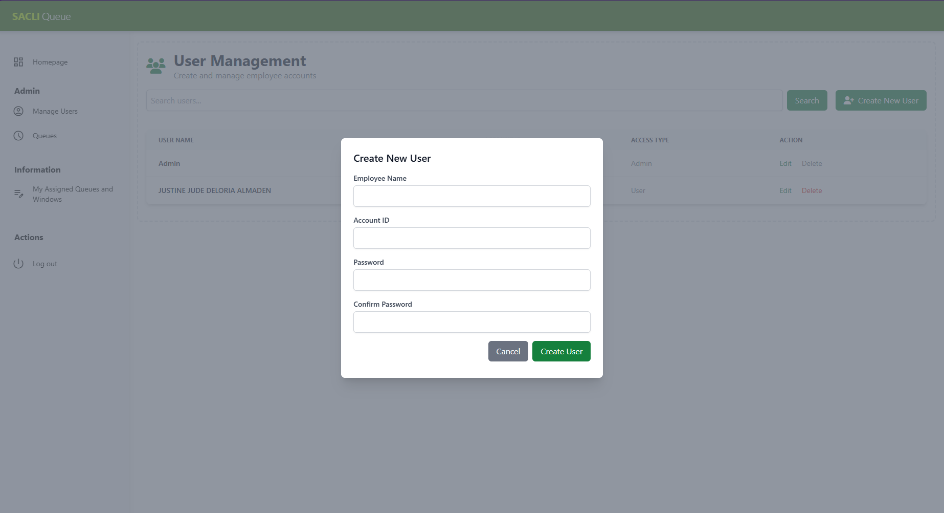


# creating And managing accounts

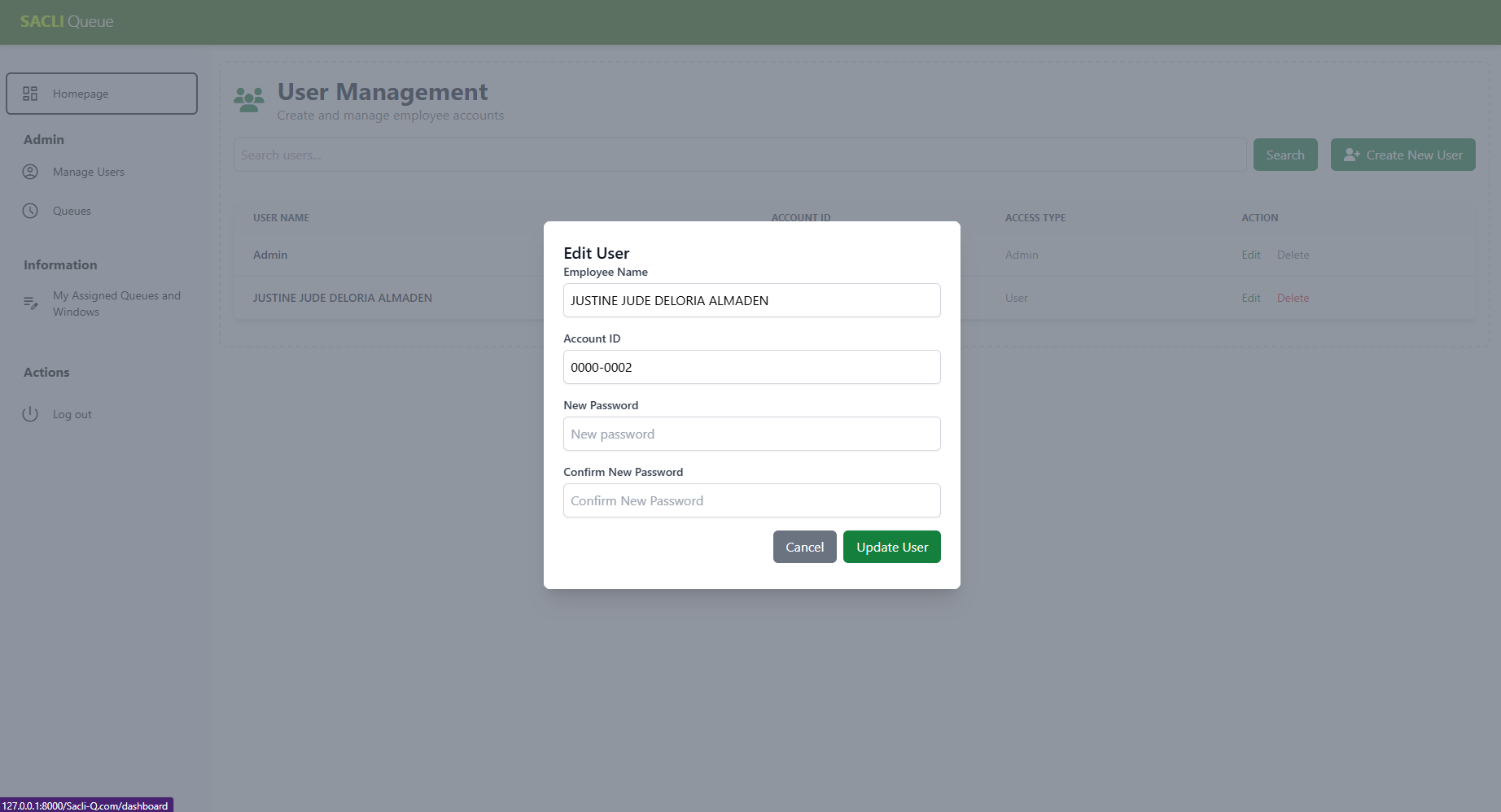
Accounts can be created by the admin by visiting the “Manage Accounts” section on the side panel





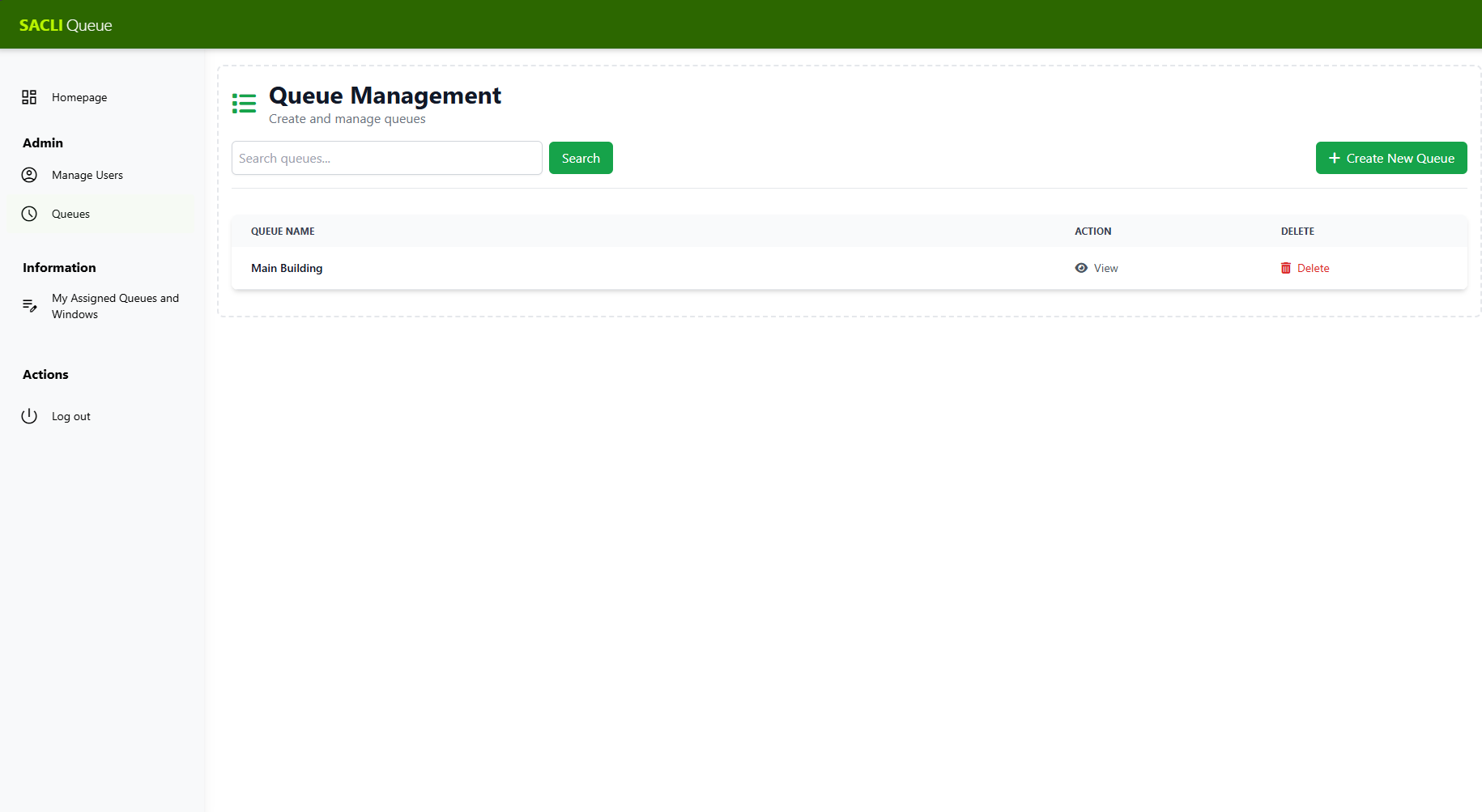
Clicking the “Create account” Button will provide a modal form for account creation.

In the event of forgotten passwords and login credentials, Contact the system admin to change your password.

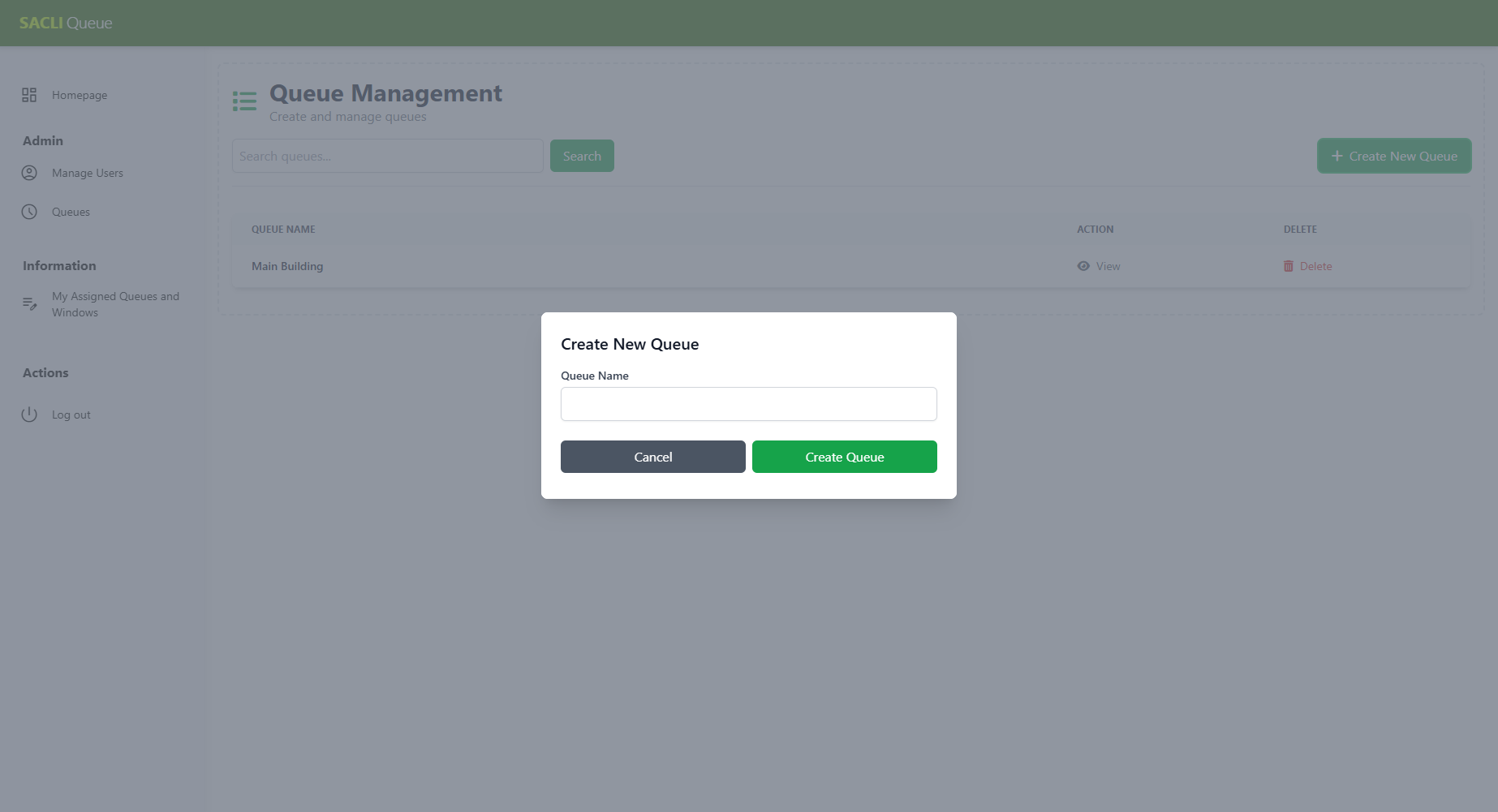


# Managing queues

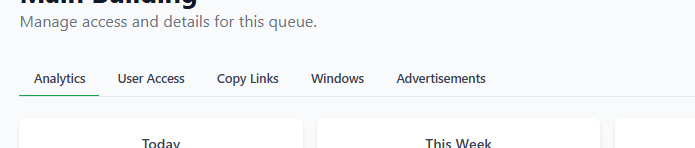
Queues are made by the system admin; users are assigned to a window where in multiple users can have access to the same window.

****

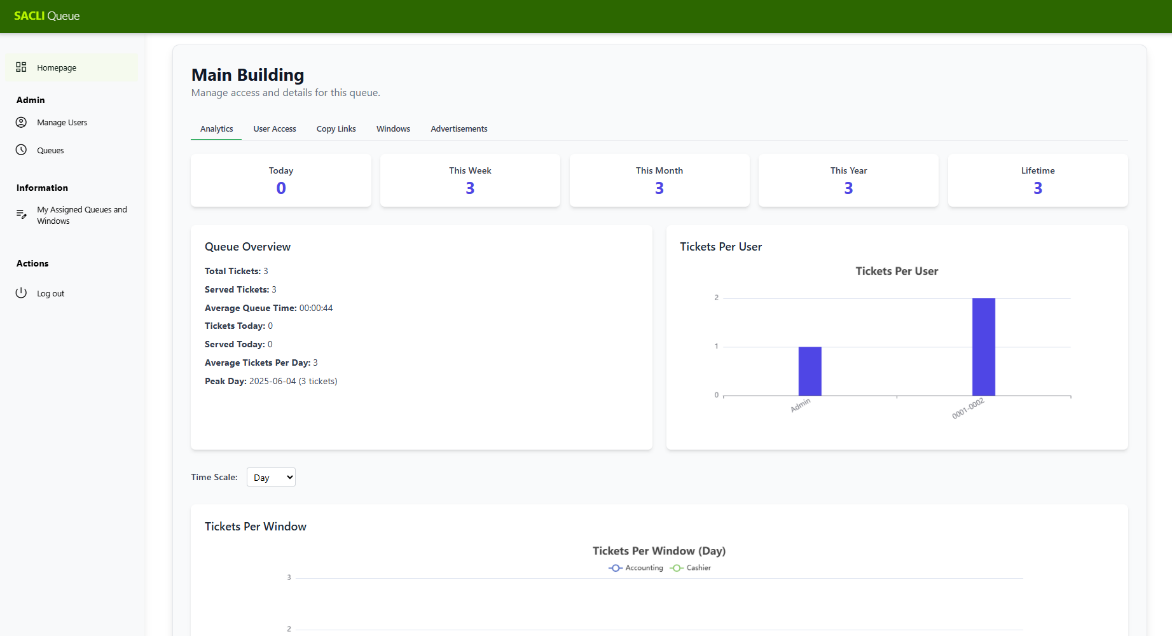
*List of queues*

**

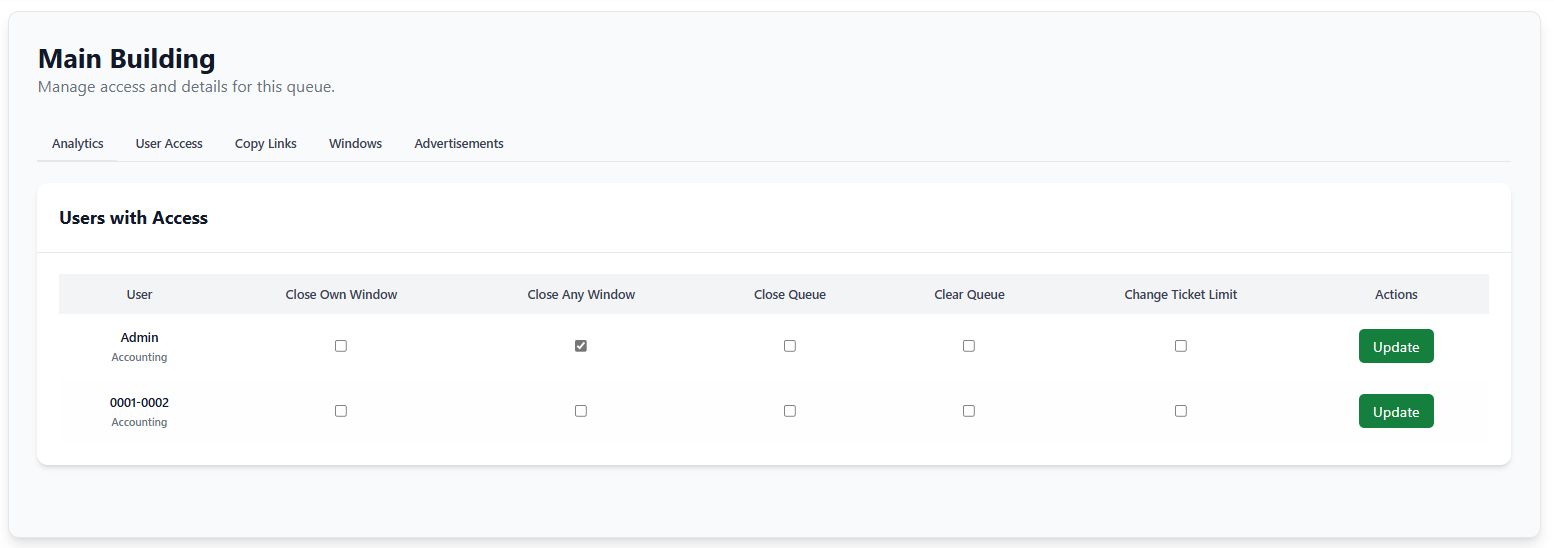
*Creating a new queue*

**

*Switching tabs*

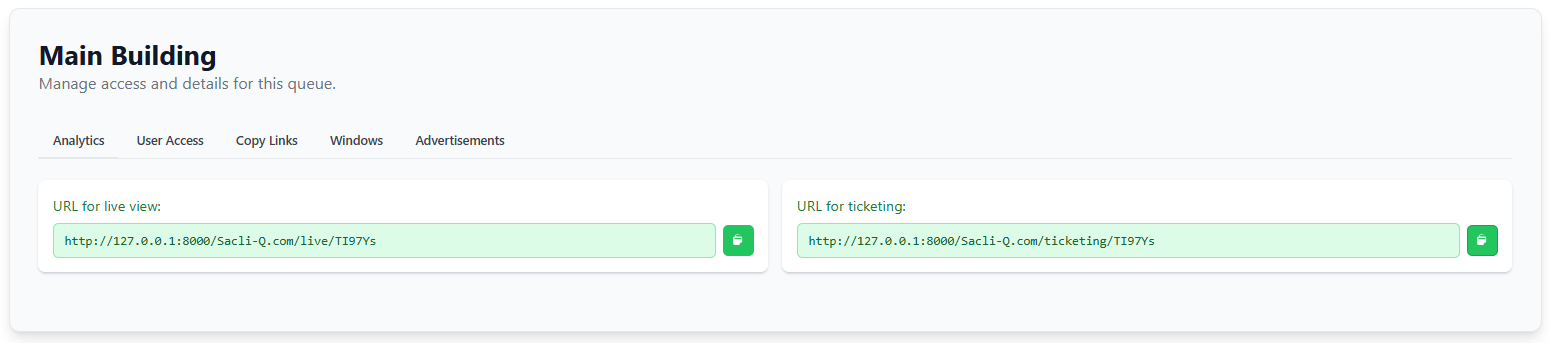


*Analytics Section*

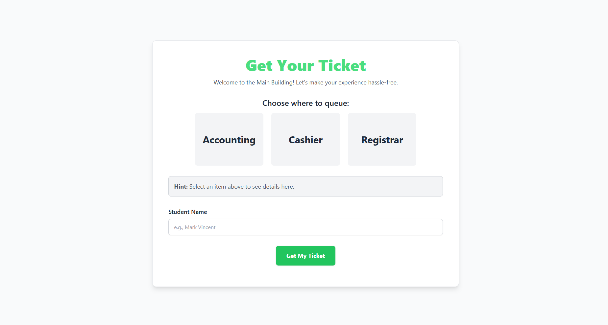
Users with acess to any window are added here and their priviledge can be modified 

*User Access Section*

Copy Links have the url for live view and ticketing

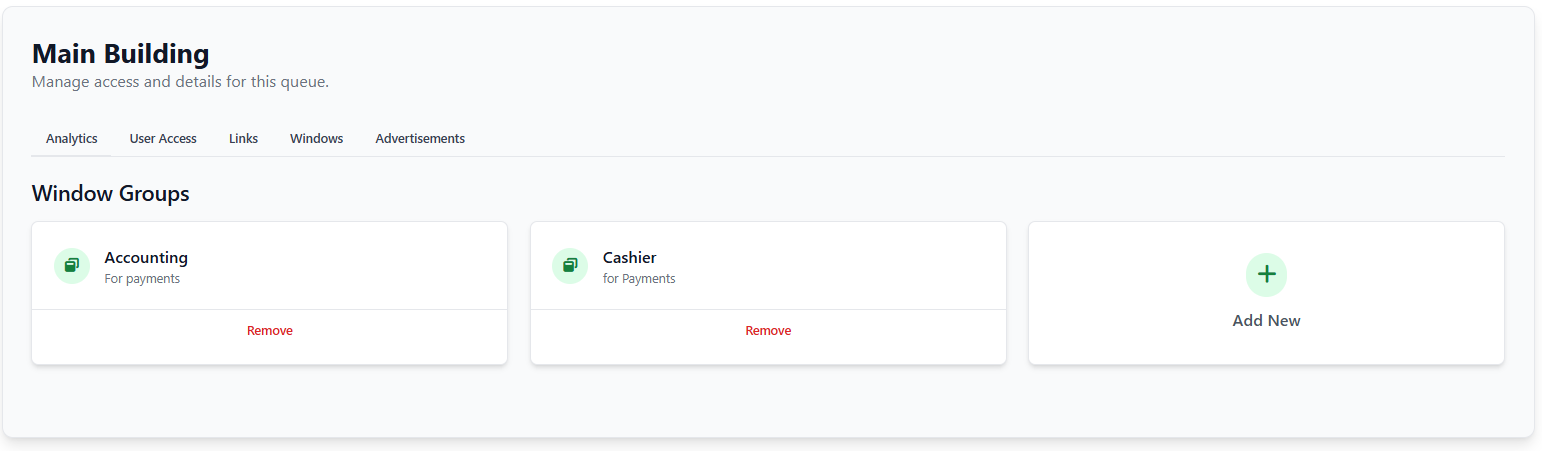
**

*Links Section*



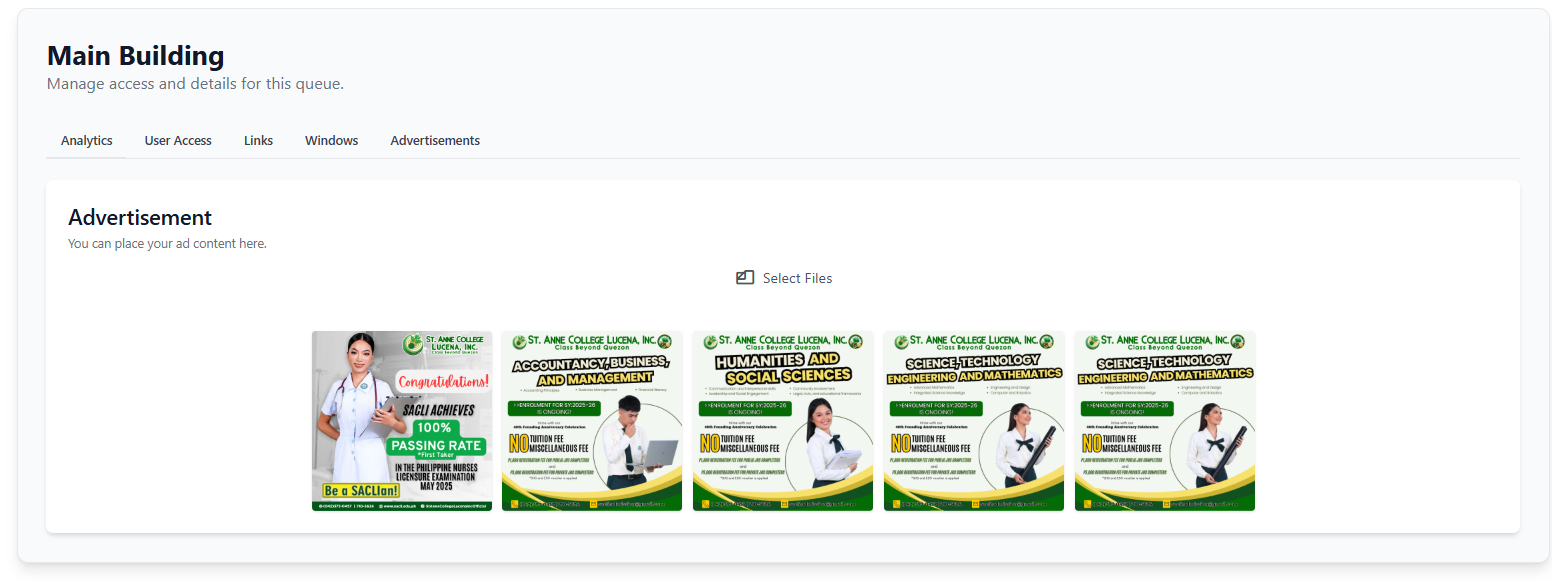
*URL for live queue URL for ticketing*

Windows Section contains the list of windows associated with your queue. You can make one or view a window to assign users to have access on that window

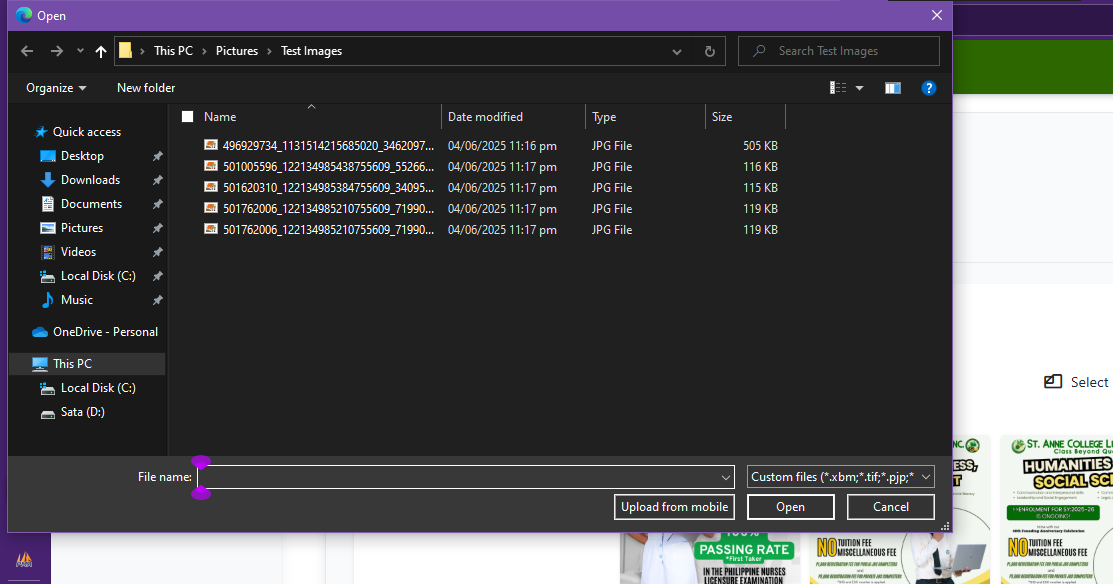
**

*Windows Section*

Advertisement Section Is where the user can insert files to be displayed for live viewing. This **OVERWRITES** the exsisting files so please be careful.



*Advertisement Section*

**

*Clicking Select Files*

***NOTE: Uploads can take time for videos.***

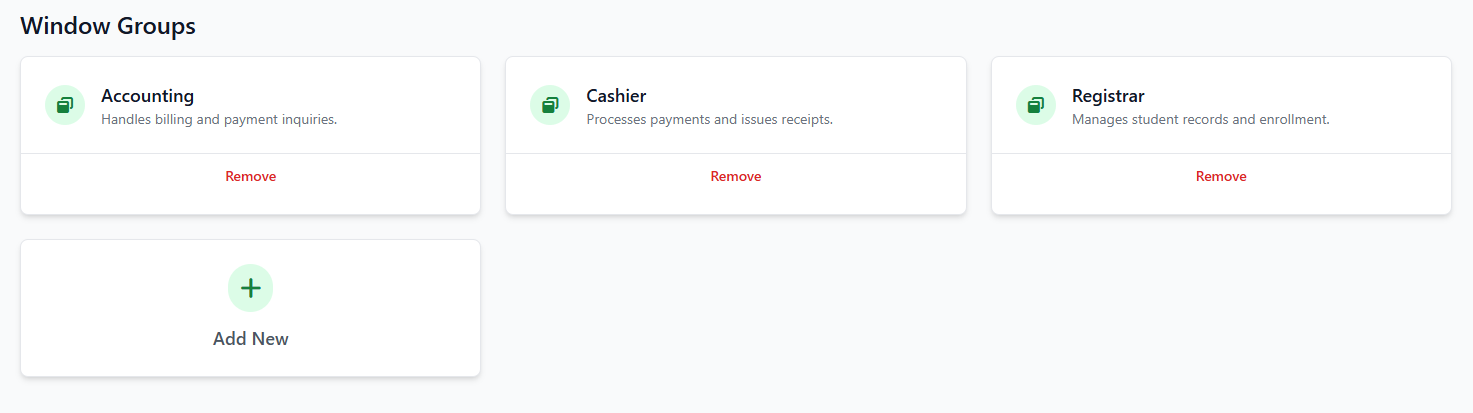
***XAMPP Limits uploads to 20mb.This can be changed by Opening the XAMPP Control Panel → clicking Config next to Apache → select php.ini. And changing the following values***

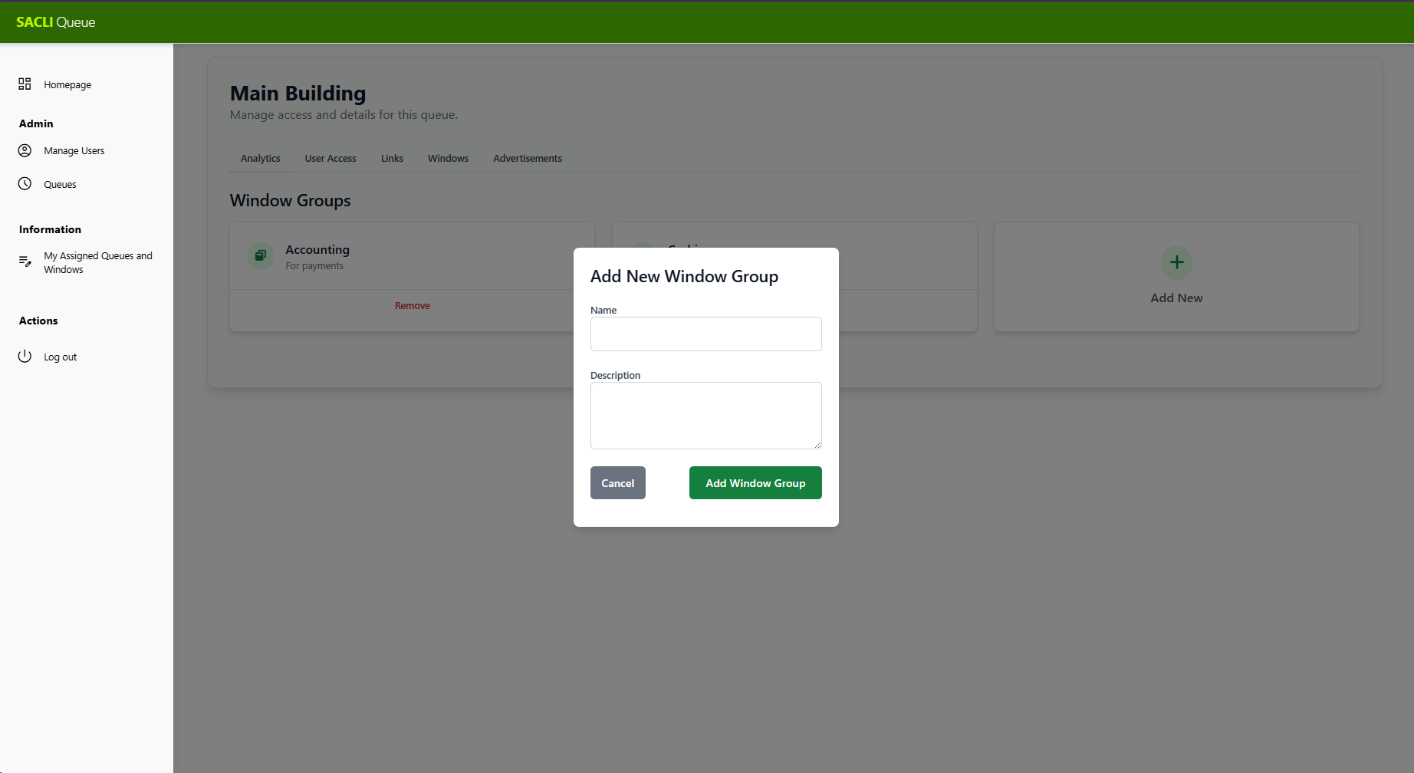
*upload\_max\_filesize = 20M → change to 100M or more*

*post\_max\_size = 20M → change to 100M or more*

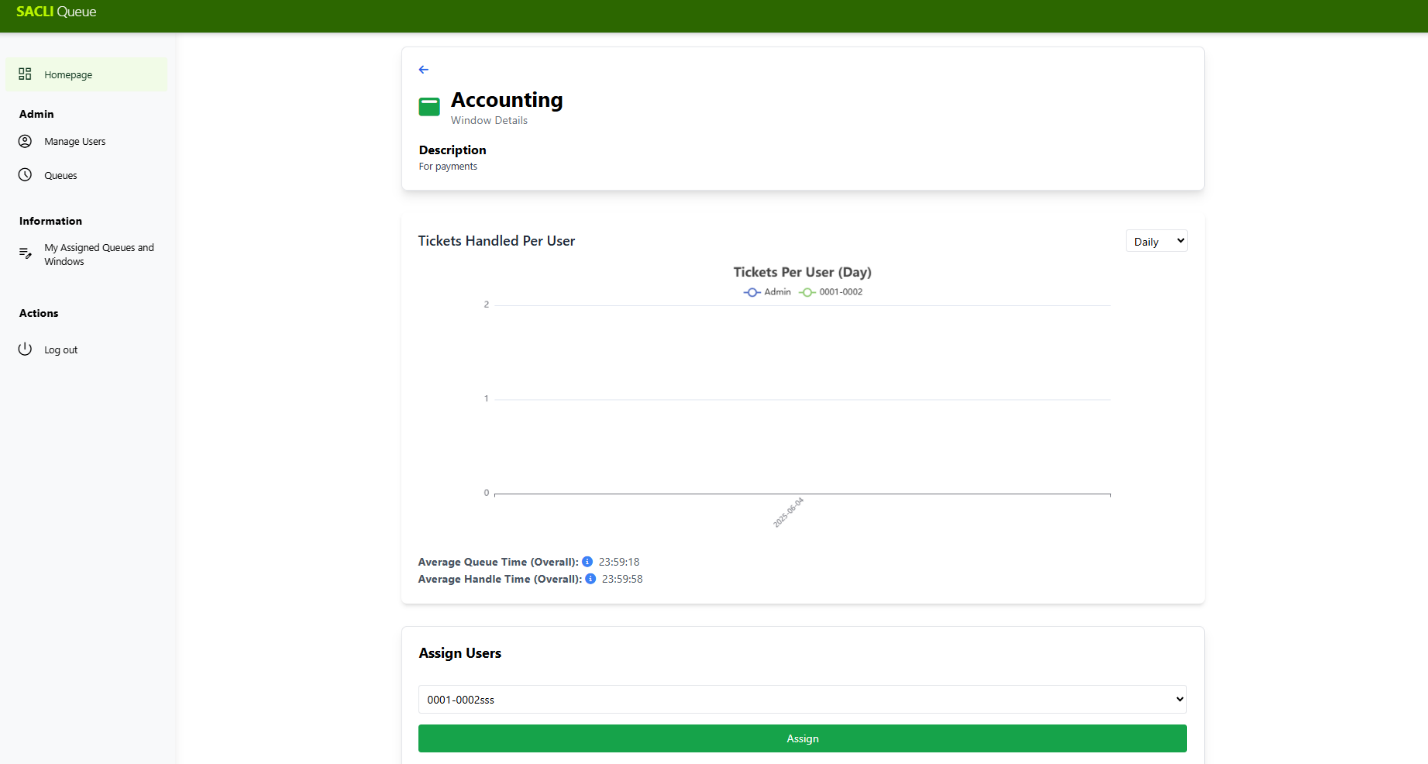
*memory\_limit = 128M → increase to 256M or more*

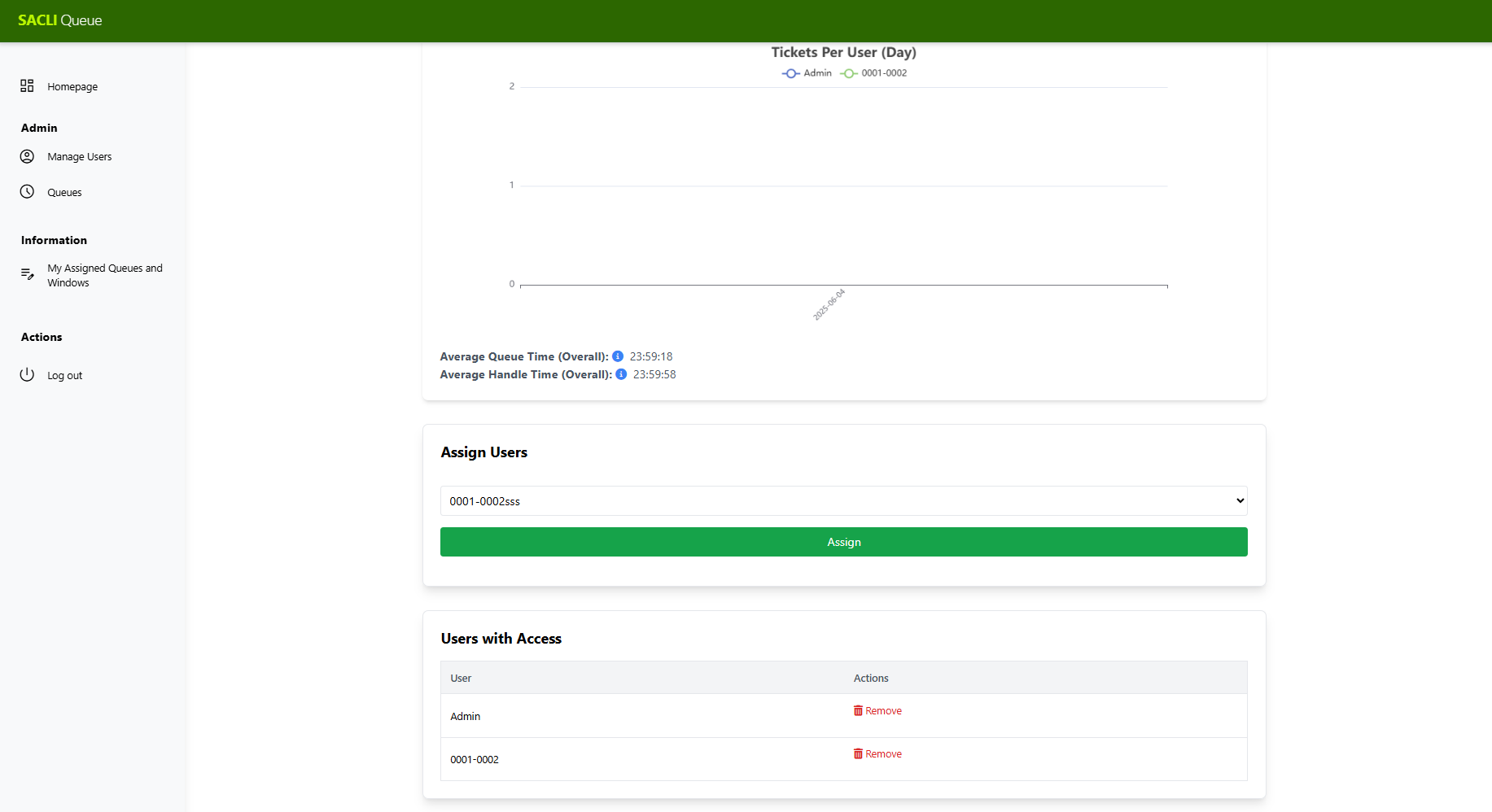
# Managing Windows

A New window can be created for a queue, and old windows can be removed. When clicked, the admin can view accounts that have access to the window. Description will be displayed for the ticketing to provide more information regarding the queue.



*Creating a New Window*

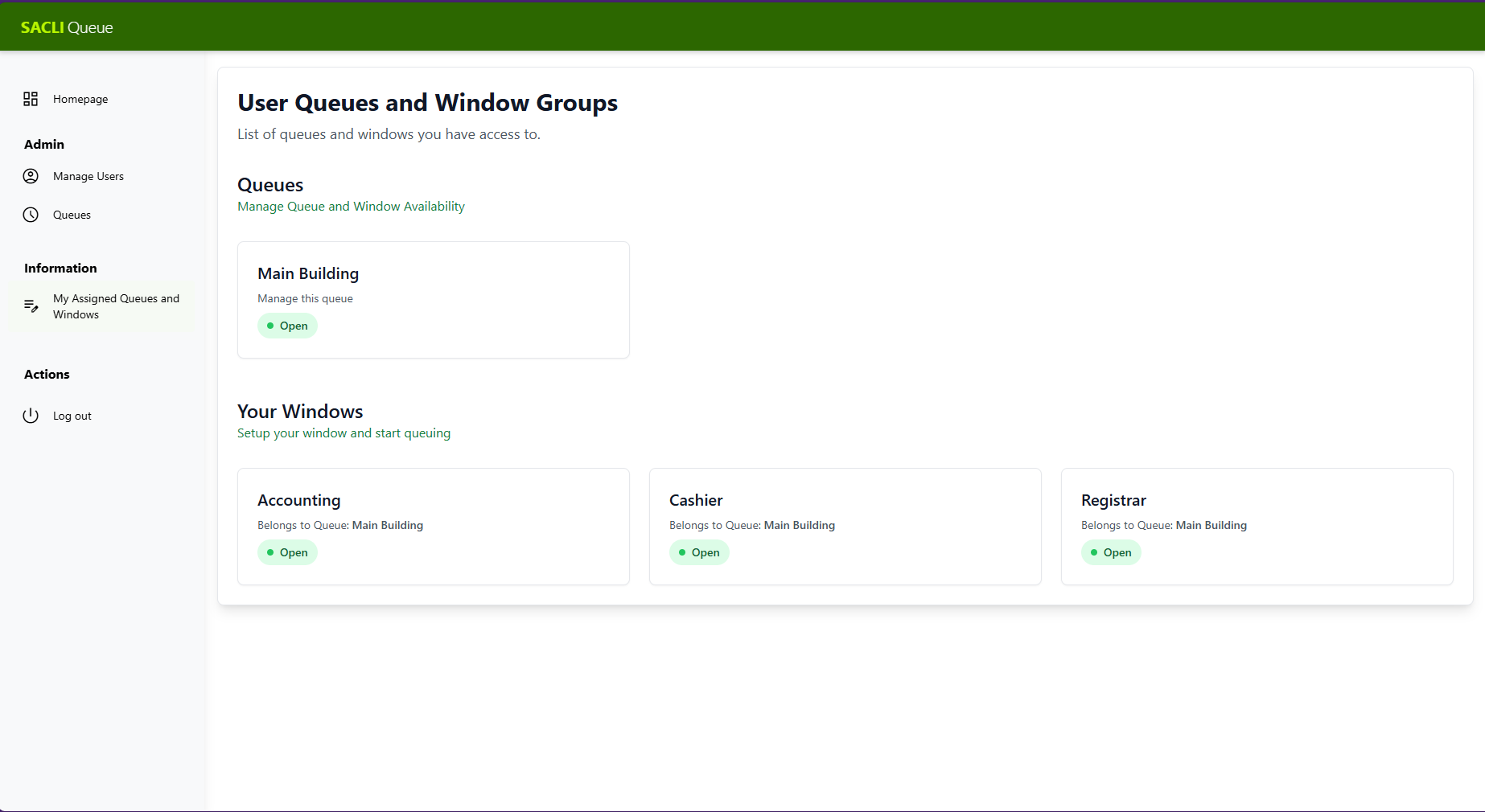




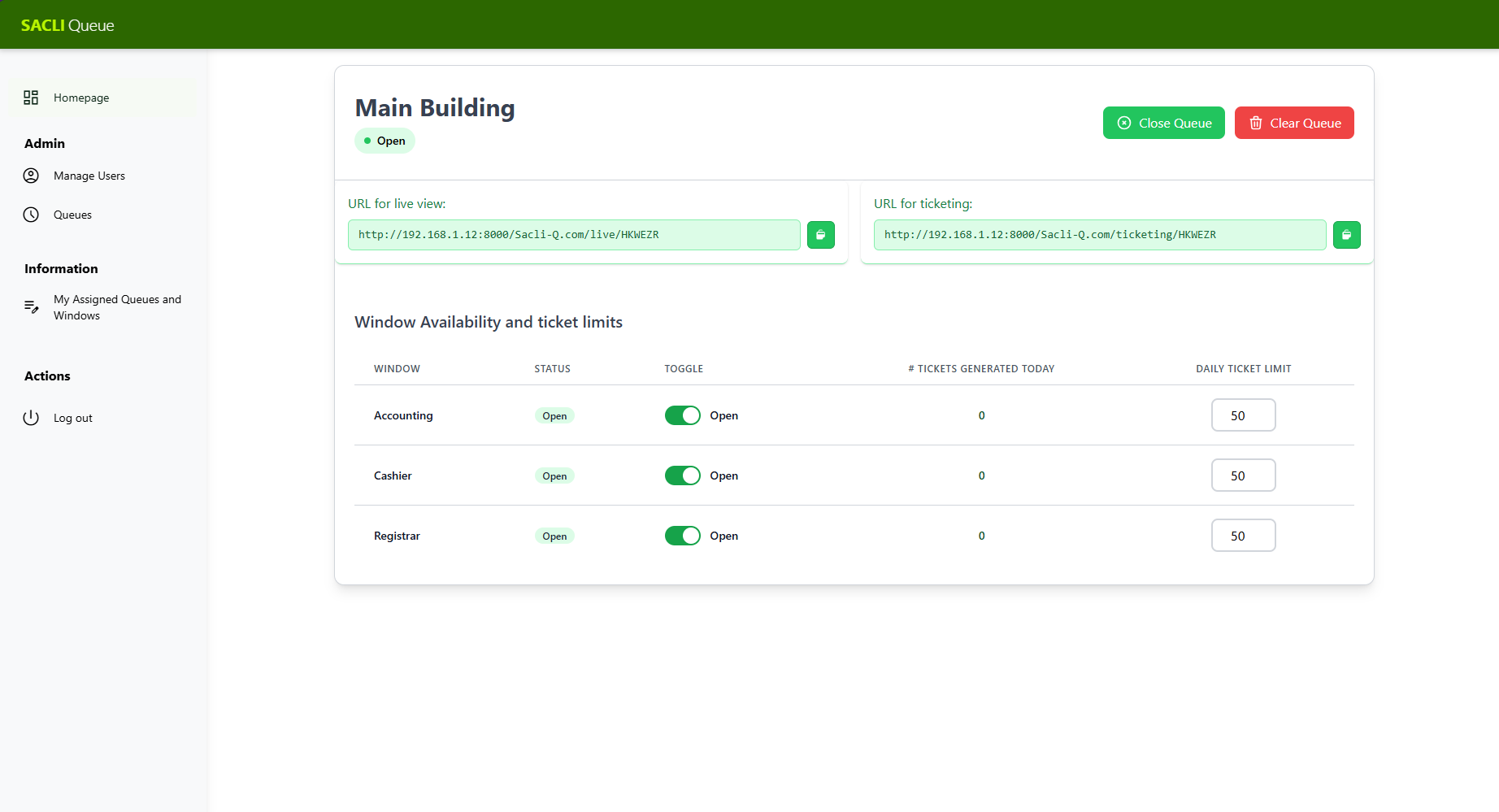
*Viewing a window and Accounts with access*

# MY Assigned queues and windows

Users can view queues and windows where they are assigned.

**

When the user views a Queue that they are assigned eg. Main Building. Certain controls can be done depending on their level of access provided by the Admin.

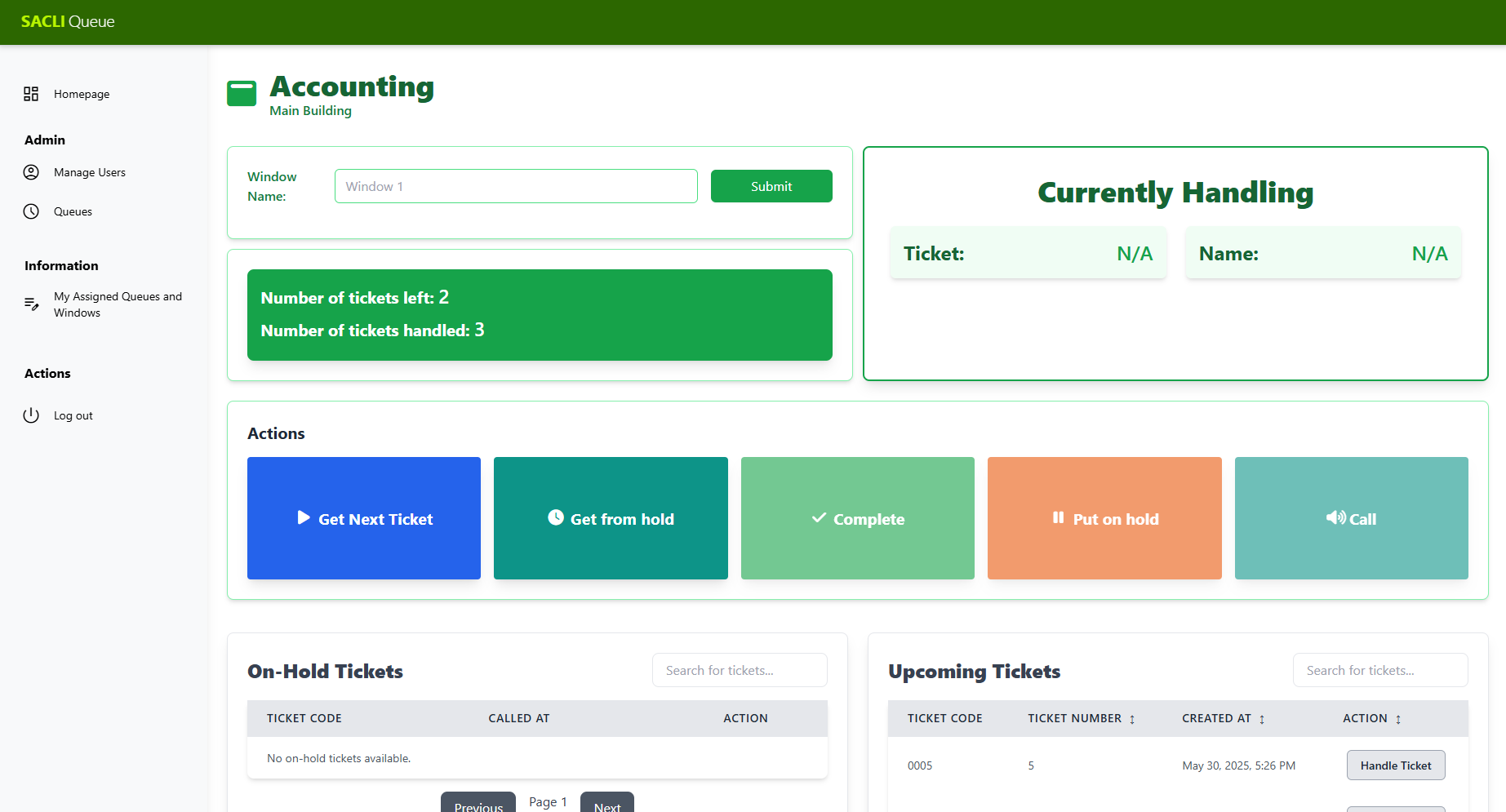




|  |  |
| --- | --- |
| Close own Window | * Close A Window where the current user is assigned and prevent new tickets from being generated. * Allow Existing tickets to still be handled. |
| Close Any Window | * Close any Window on the queue. * Allow Existing tickets to still be handled. |
| Close Queue | * Can Close/Open the entire queue and prevent all windows from taking new tickets. * Prevents any window from being opened unless the entire queue is opened. * Allow Existing tickets to still be handled |
| Clear Queue | * Cancels All tickets that are, On hold/ Waiting |
| Ticket Limit | * Set a limit on how many tickets can be generated for each window |

# The Queuing Dashboard

Users with access to a window can open it and handle queues.

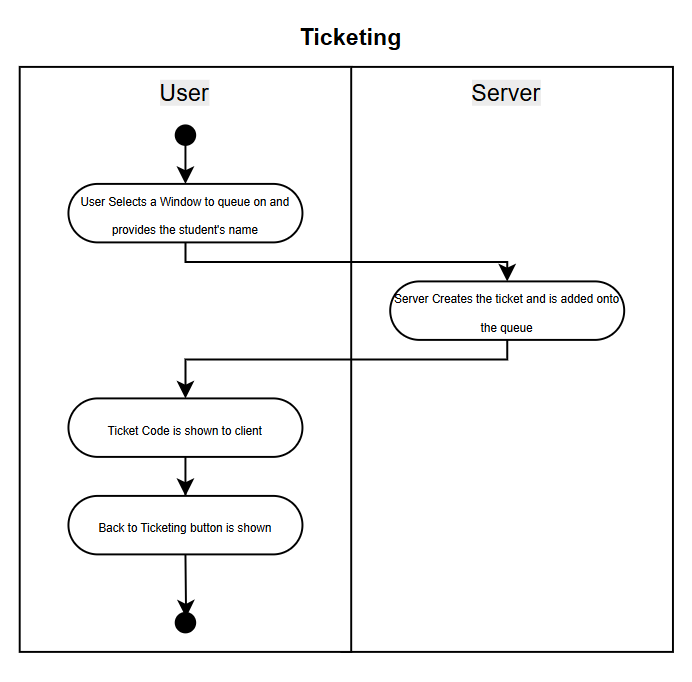


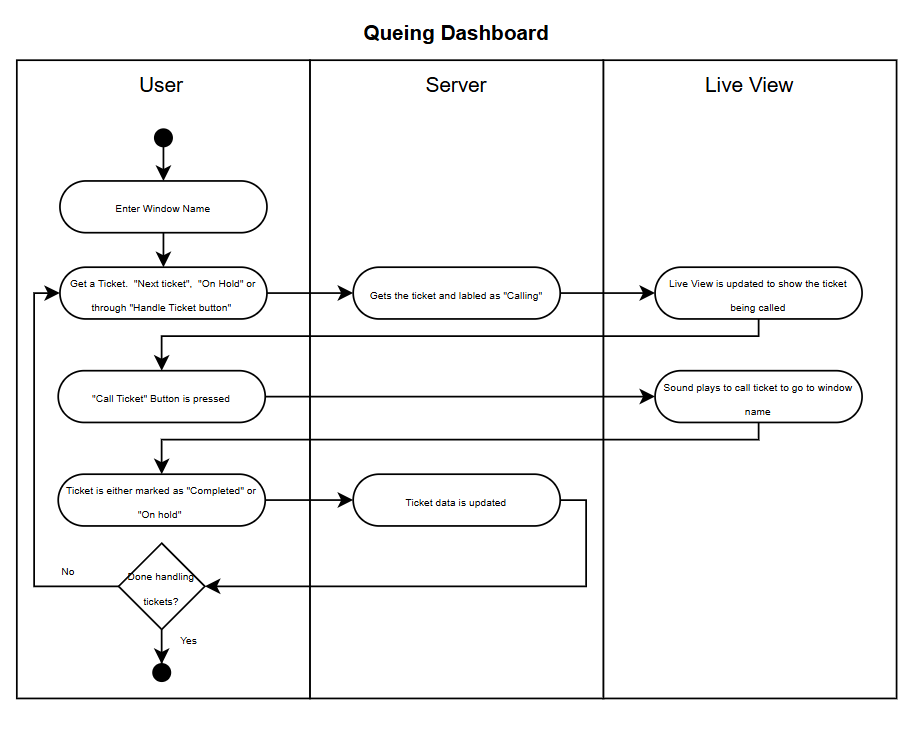
|  |  |  |
| --- | --- | --- |
| Window Name | * Name for your window to be displayed for the live view |  |
| Get Next Ticket | * Get a ticket from the list of upcoming tickets |  |
| Get From Hold | * Get the oldest ticket that has been put on hold. |  |
| Mark Ticket as Complete | * Mark the ticket as complete after it has been handled |  |
| Put ticket on hold | * Put the currently selected ticket on Hold |  |
| Call ticket | * Play a sound on live queue to call the currently selected ticket |  |
| Handle Ticket | * Select the ticket regardless of position in queue |  |

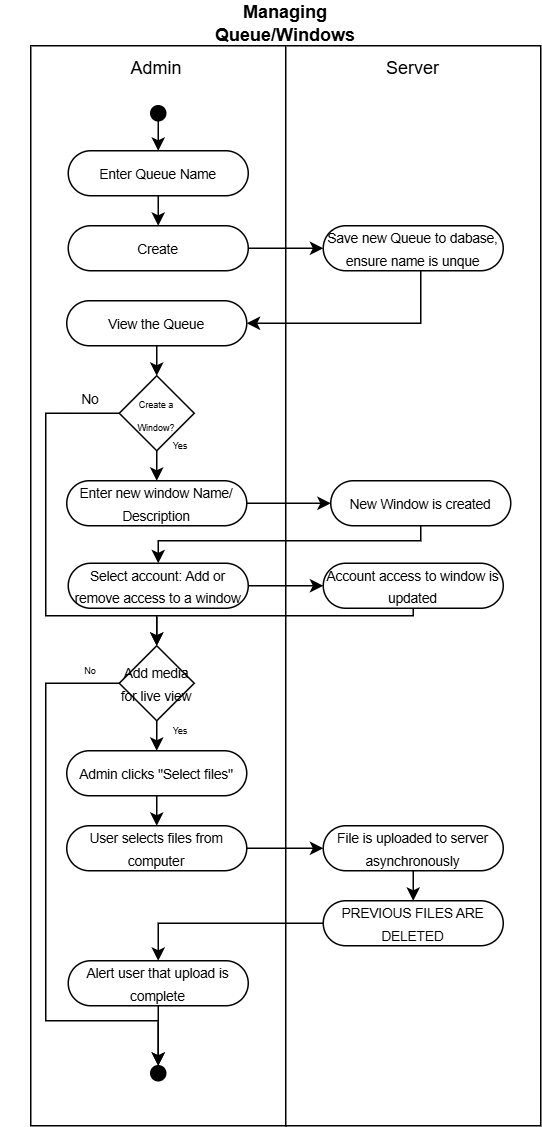
**Note:**

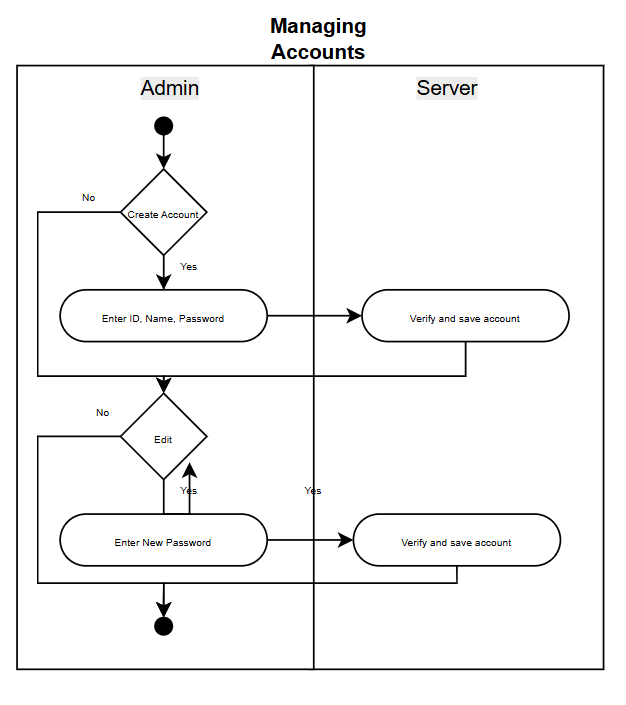
**To prevent “Camping” and cutting behavior. Require ID to be presented and match the name associated with the ticket.**

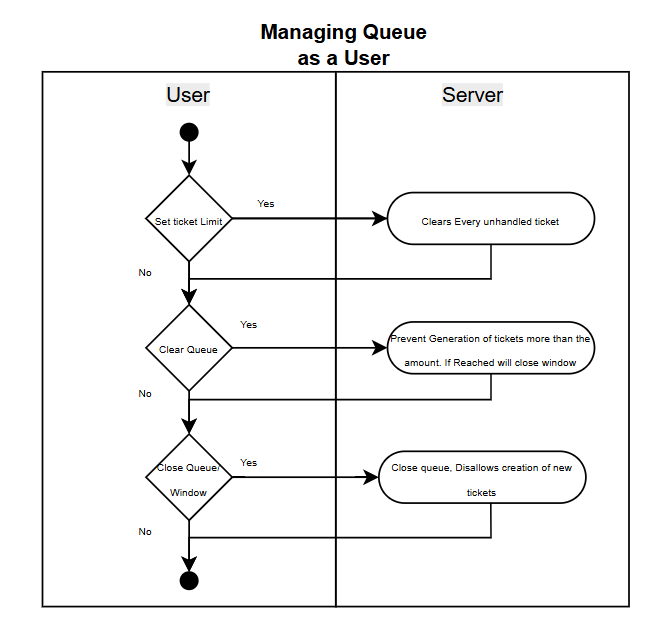
**Activity Diagrams**

****

****



****

****