

# MineService

Founder: CJ Miller

Members: Jeremy Brinegar

Ishank Tandon

Max Morgan

# Change Log

## Milestone 5:

Updated the high-fidelity prototype images that changed.

Added "Feature Completed" section.

## Milestone 4:

There has been no significant changes since Milestone three. We have implemented the changes our client suggested and still working on making some of the pages more to their liking.

# Introduction

MineService is a Client-Server application that would help monitor and run Minecraft servers across different systems. It can be split into two different feature sets. One for dealing with physical systems, such as resource allocation, while the other dealing with people, such as support tickets. Between these two sets there will be many sub features.

For physical system features the main ones will be on control. We'll have ways to start, stop, and restart the server and also automated versions of those. There will also be ways to upload, edit, and delete files on the server.

The people system will include both support and permissions systems. The support side will allow people to both submit and resolve issues in a centralized semi-public place. The permissions system will control who has access to what physical system features and what things they can see on the support side.

# User Stories

- **Epic**
  - **Performance**
    - As a developer I want to give typical users better performance when running Minecraft users so that they can have a more enjoyable Minecraft experience.
  - **Time Saving**
    - As a developer I want to give typical users the ability to minimize their time spent on maintenance activities so that they can have their server up and running quicker.

- **Ease of Use**
  - As a developer I want typical users to be able to easily accomplish their everyday tasks so that they do not have to waste time looking up fixes or syntax.
- **Feature**
  - **Performance**
    - **VM Overhead**
      - As a Server Administrator I want to reduce CPU and Memory Usage so that I can dedicate more to my clients.
      - Condition of Satisfaction:
        - Drastically reduce the number of required VMs per client.
    - **Memory Limit**
      - As a Server Administrator or Minecraft Admin I want to be notified when a Minecraft Server is approaching its memory limit so that I can see if it is an issue or just user spike
      - Condition of Satisfaction:
        - I am notified via text or email within a minute of a condition being met.
    - **Frozen Servers**
      - As a Server Administrator or Minecraft Admin I want to be notified when a Minecraft Server is frozen so that I can fix it promptly
      - Condition of Satisfaction:
        - I am notified via text or email within a minute of a condition being met.
  - **Ease of Use**
    - **Terminal**
      - I'm a person who is responsible for a couple of Minecraft servers but I don't like to program. I prefer not having to memorize code or commands and type it in the terminal because there are so many of them. To make my life easier, I need to have a User Interface which would do all the stuff the terminal commands do with just a click of a button.
      - Condition of Satisfaction:
        - Acknowledge that I don't have to use terminal to manage my servers
    - **Multiple Applications Open**
      - I've realized that the less number of applications I have open, the less CPU usage and memory leaks. Having multiple applications open can potentially slow down the servers and is also really hard to manage since I need to switch between them at the right time for the right thing. It would be so much better if I just had to have one application

open which would display all the information which I seek from multiple applications and perform an action based on that.

- Condition of satisfaction:
  - Just one User Interface which would display all the information needed

## ○ Time Saving

### ■ Terminal Commands have Multiple Steps


- I'm an individual who owns a few Minecraft servers, however I don't write programs as my profession. I would rather not have to enter commands into a terminal, because it takes time to memorize the commands, and if I don't memorize the procedure of how to navigate the terminal, I will have to take time to write a document with steps outlining the commands I have to execute. It would save me time if I only had to click a button on a graphical user interface, and all of the terminal commands would be done for me.
- Conditions of satisfaction:
  - Make sure that I don't have to enter any commands into a terminal to manage my server.



### ■ Regularly having to login to Check the Servers

- I've been told that Minecraft servers with a lot of mods on them can cause memory leaks, and memory leaks can cause a lot of memory, disk, and possibly CPU strain on my server. This can slow down all of my servers, and possibly cause them to crash. Therefore, I must log in to the windows application every three days or so to check the memory, disk, and CPU usage of each server. If a server is taking up too much hardware, I have to hope that no one is on the server, so that I can stop the server, then start it again. If there are people on the server, I either have to stop the server anyway, which will kick off all of the people who are on the server, or I have to wait until another time, log in again, and check and see if anyone is on the server. It would be much more convenient if a server automatically restarted if no players were on and the combination of CPU, memory, and disk usage reached a certain level. Also, if the server was not able to be automatically restarted, due to players being on the server, and CPU, disk, or memory usage got too high, I would like to receive a text message, e-mail, or both, describing the issue to me, so that I can manually restart the server if I saw the need.
- Conditions of satisfaction:
  - If there are players on a server, and the combination of CPU, memory, and disk usage exceeds a certain level, notify the server owner via their specified means of contact, either text message, e-mail, or both.

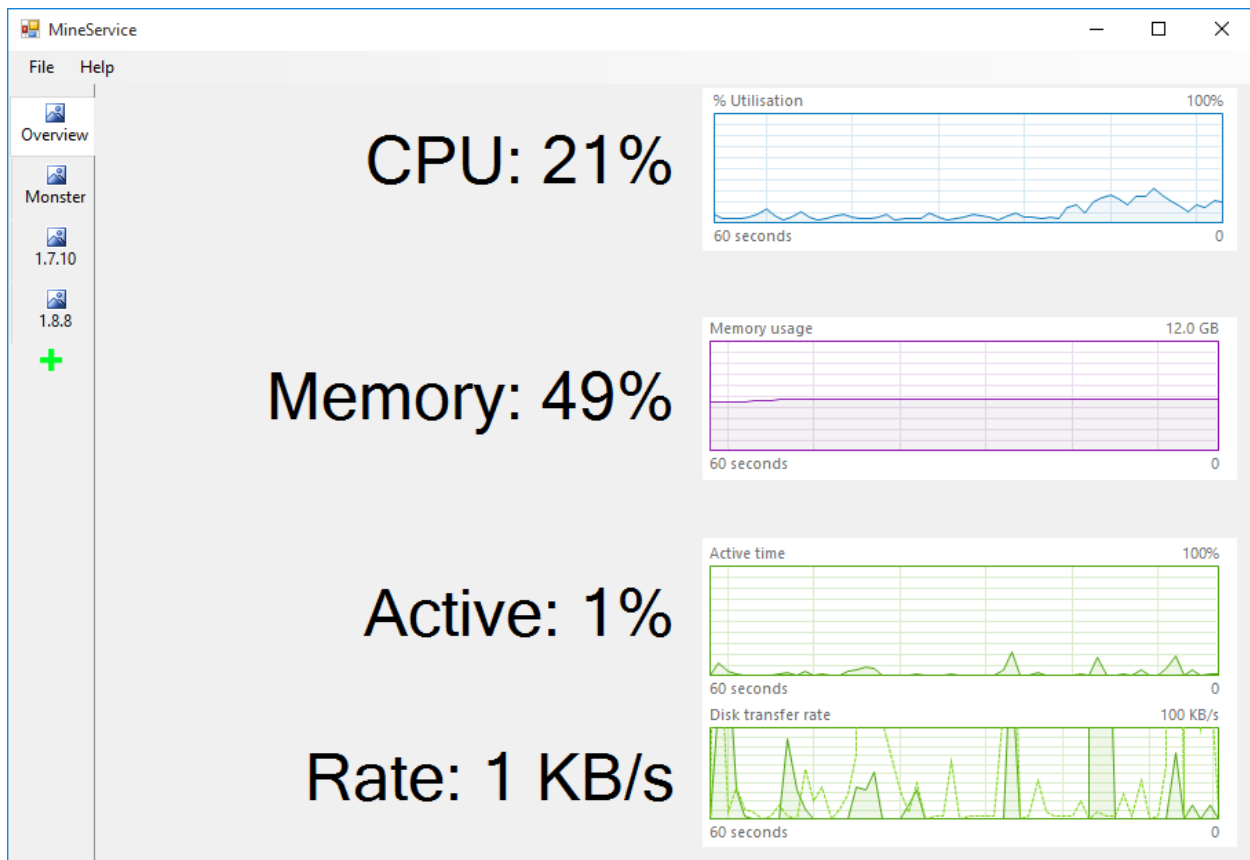
- Have the server automatically restart if the combination of CPU, memory, and disk usage exceeds a certain level, and there are no players on the server.

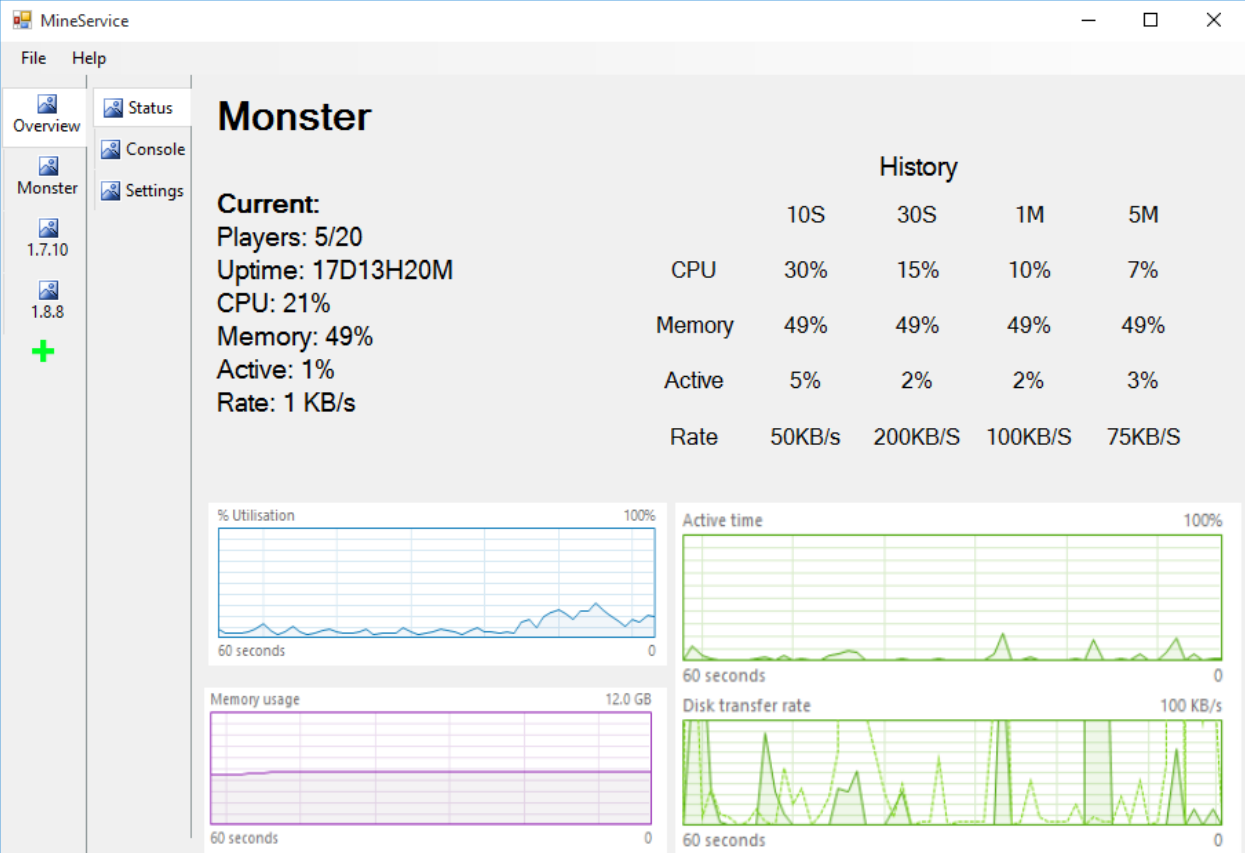
## Low-Fidelity Prototype



Select Server  

☐ Remember Login







## **Alternative Design**

## Log In

Server

User name

password

Remember password

☐

Mine Service

↓  
pop-up window

Port

## Server information

Server1

Server2

Server3

Server4

Q Switch by Server

Server Name: \_\_\_\_\_

Current usage: \_\_\_\_\_

past usage: 

graph

restart

set options



## Options Screen

### ☐ server & options:

☐ set time for restart

hr

min.

timezone

☐ restart when ~~restart~~ body is on

### Notifications:

☐ high memory usage

☐ on reset

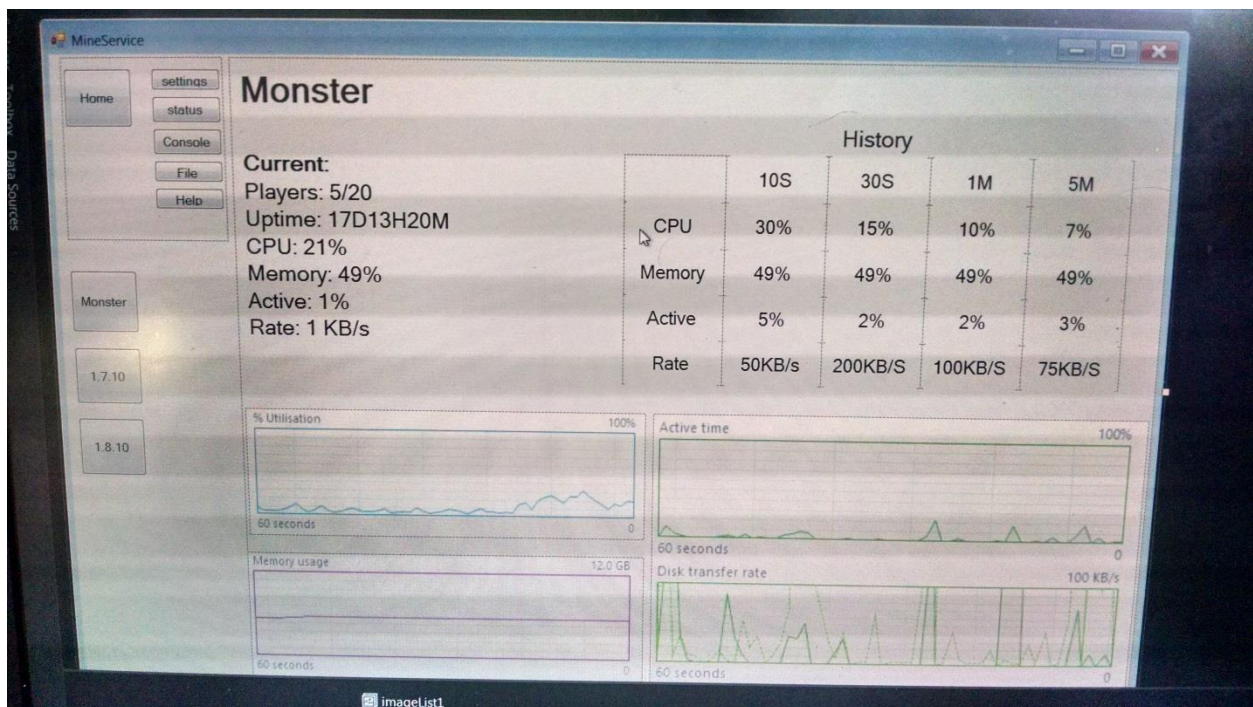
## Plan for Evaluation

A couple clients that currently pay CJ were given the prototype designs. The process of them using the prototypes was recorded and put into a video file. We hoped to accomplish the layout of the user interface and obtain feedback on the display of the analytical data of the servers. We were unsure especially of which side the tabs should be on for displaying the other server's data. In addition, we hoped to obtain feedback from the where the commands should be issued.

## Result for Evaluation

Based on the feedback, we saw that the "file/help" bar should be removed and create a new tab called "home" that this will integrate in with. The elements under the current overview tab will also be under the "home" tab. Overall the feedback was good, just requested a few things being moved around the UI. They also said the first prototype would be good for the desktop application.

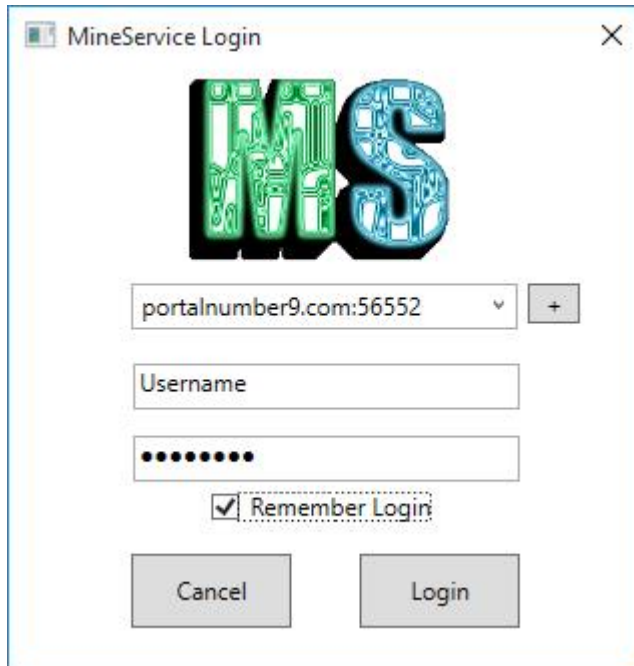
## Changes to Prototype



Home tab now involves file and help and the place for entering commands is in the home tab as well.

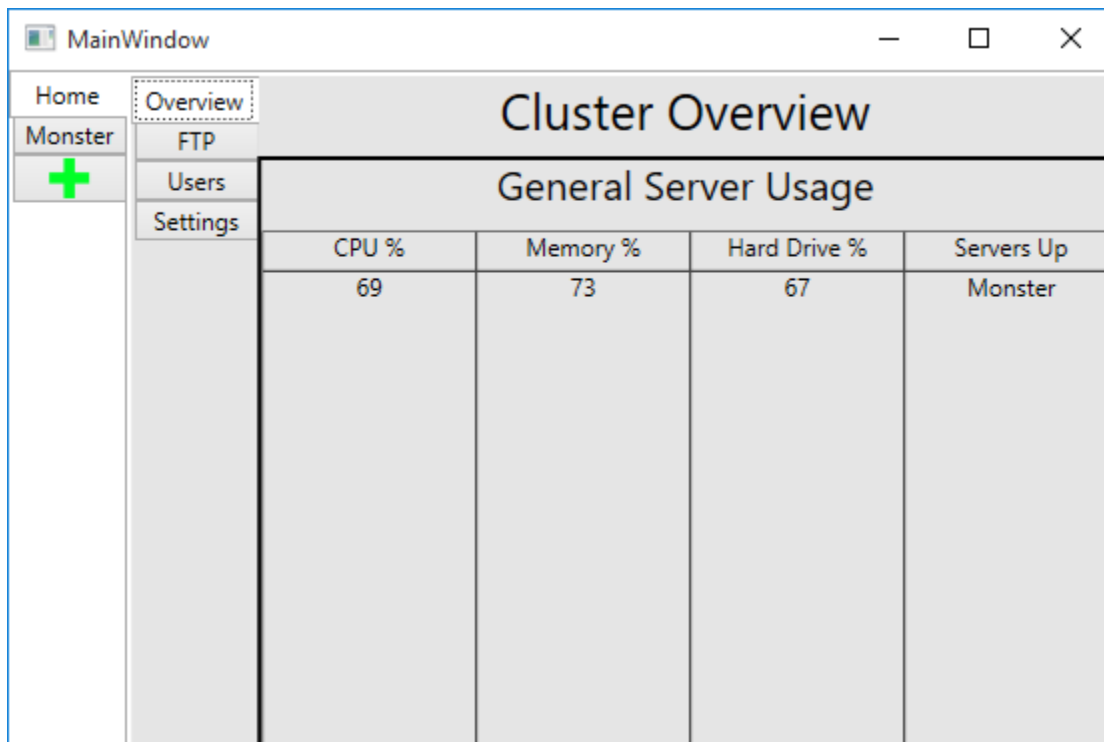
## High-Fidelity Prototype

This is the first window you will see. General Login info that can store the servers you connect to.



The image shows a 'MineService Login' window. It features a logo with the letters 'MS' in a stylized, circuit-like font. Below the logo, there is a dropdown menu showing 'portalnumber9.com:56552' with a '+' button to its right. Underneath are input fields for 'Username' and a password field represented by dots. A checkbox labeled 'Remember Login' is checked. At the bottom are 'Cancel' and 'Login' buttons.

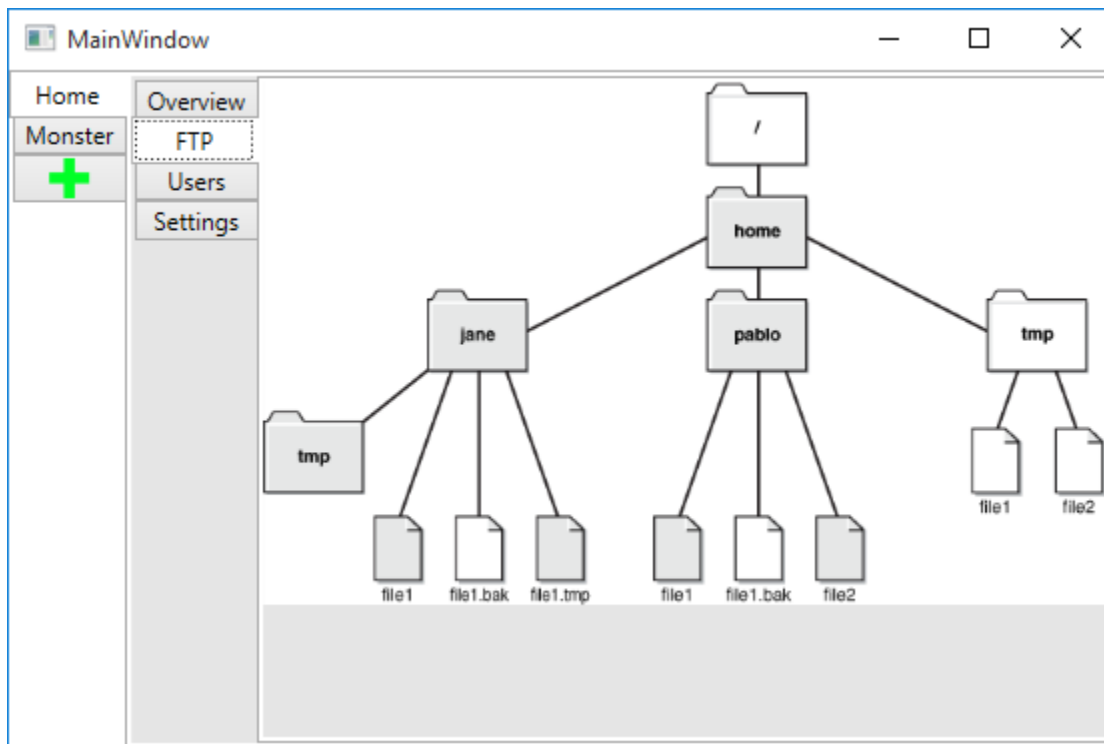
This is the window you will see after logging in. General Information about the cluster



The image shows a 'MainWindow' titled 'Cluster Overview'. It has a sidebar with 'Home', 'Monster' (with a green plus icon), and 'Overview' (which is selected). Under 'Overview' are sub-tabs for 'FTP', 'Users', and 'Settings'. The main content area displays 'General Server Usage' in a table.

| CPU % | Memory % | Hard Drive % | Servers Up |
|-------|----------|--------------|------------|
| 69    | 73       | 67           | Monster    |

This is the FTP: Design still in progress might go to more of a text based solution. Waiting from response from test clients.

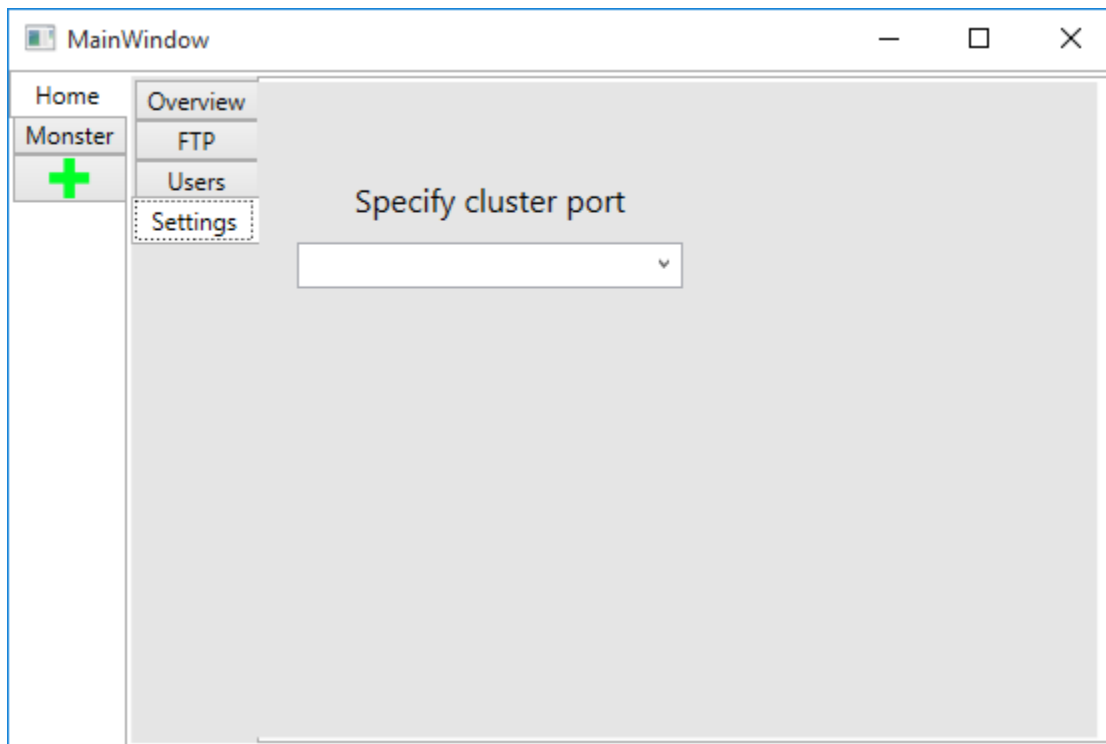


This is basic user control to see who has access to what and in what group.

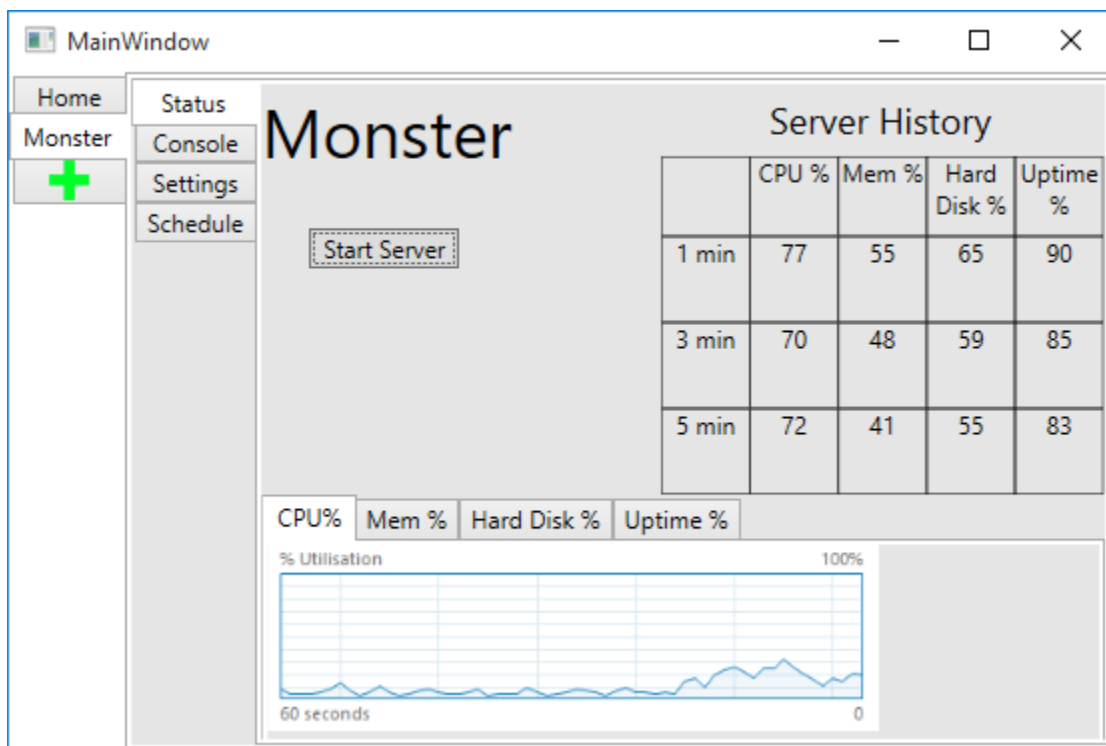
The screenshot shows a window titled 'MainWindow' with a sidebar on the left containing 'Home', 'Monster', and a green plus icon. The main area displays a table with two columns: 'User' and 'Groups'.

| User                 | Groups  |
|----------------------|---|
| max_l33t_sk1llz420   | max_iz_#1<br>420forlyfe<br>Rose-Hulman yea boiii        |
| cj_l33t3st_sk1llz420 | cj_da_man<br>420forlyfe<br>Rose-Hulman yea boiii        |
| Jeremy_HR_420        | Its_mr_grandslam<br>420forlyfe<br>Rose-Hulman yea boiii |
| Ishank_cooldude_420  | Ishank_dude<br>420forlyfe<br>Rose-Hulman yea boiii      |

This page will house cluster settings as they are needed. Currently only port is needed, but latter if we extend to additional features this will be needed.

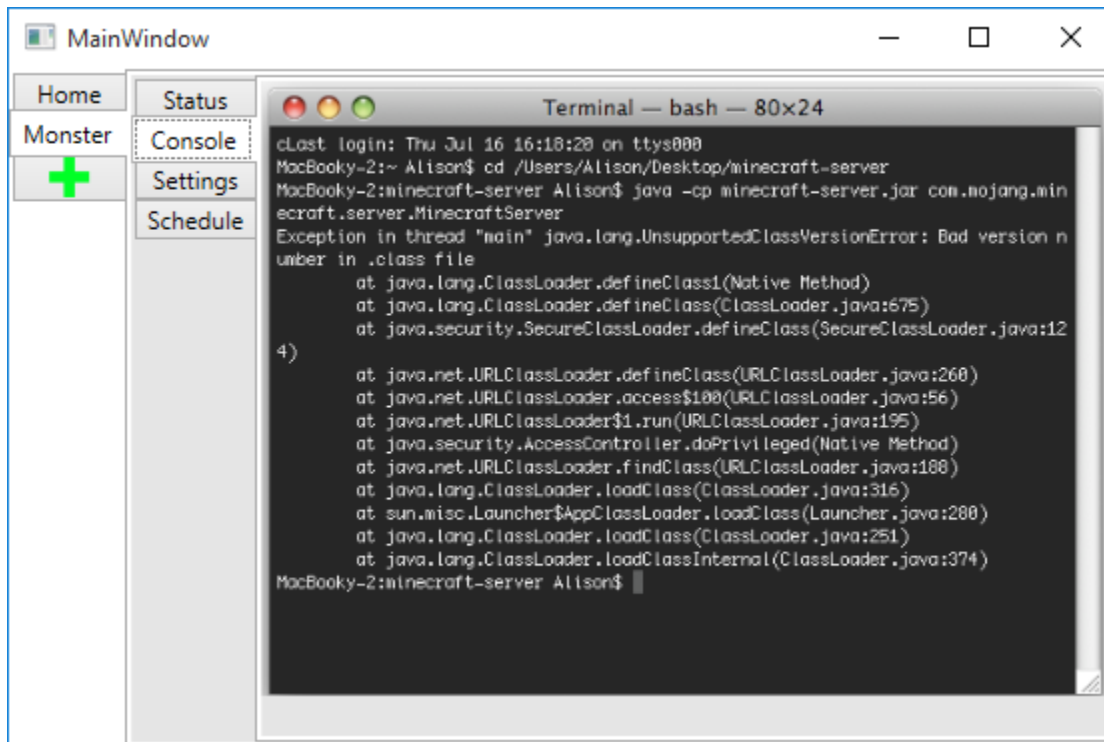


This is the default page for a specific server.

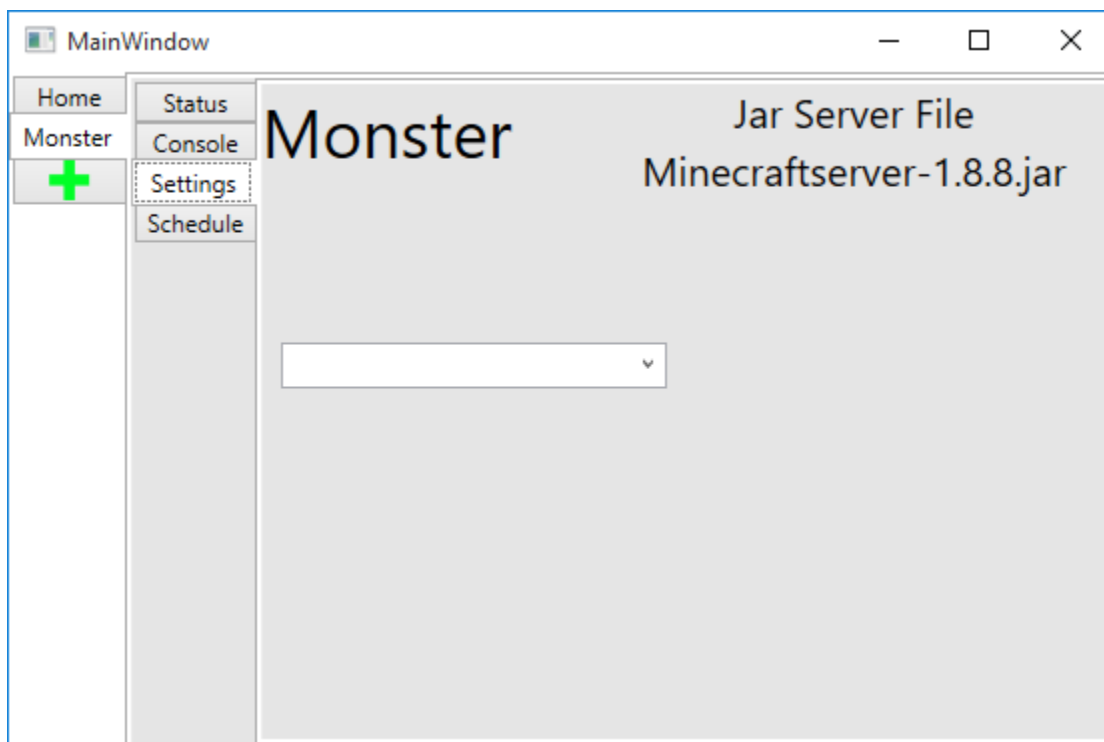




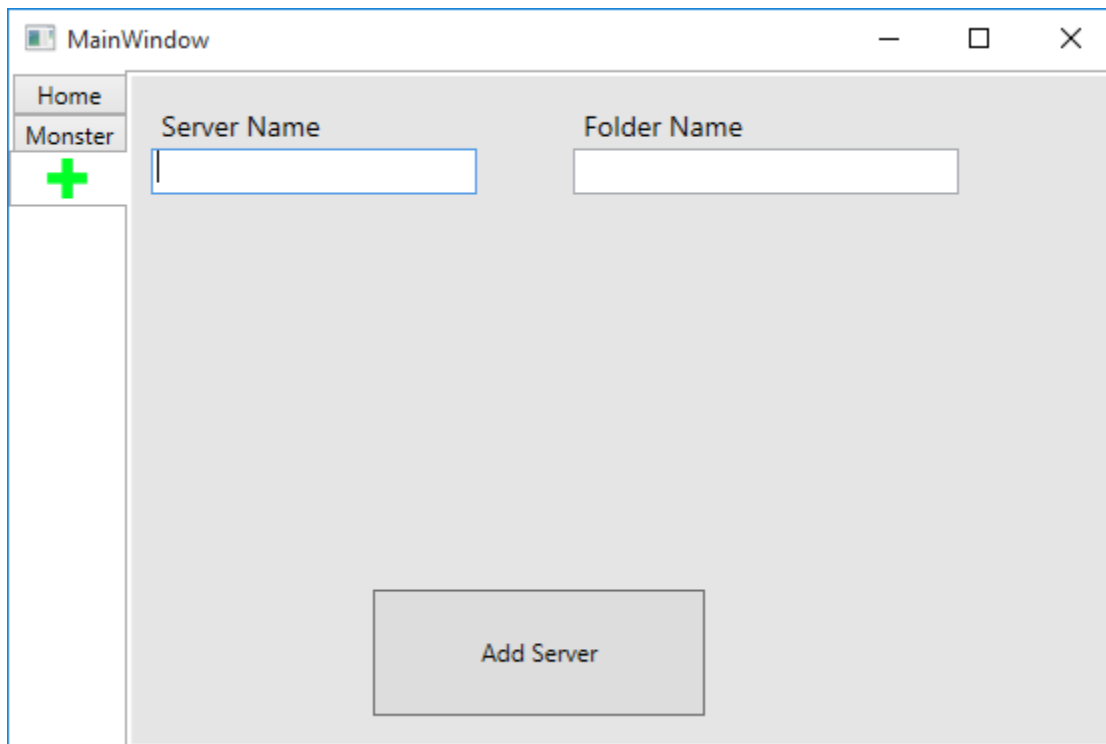
This is the console window for a specific Minecraft server, window still in progress.



Server specific settings. Still in progress.



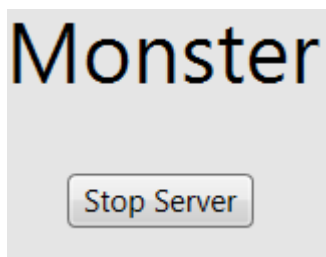
Page to add a new server to the cluster.



## Features Completed

(note: names listed in approximate order of contribution to feature)

Feature: Add a start/stop server button to the GUI, send the correct message to the server, and make sure the server is receiving the message in the correct format.




---

```
Client Connected: /127.0.0.1
Message: {"type":"Login","message":{"Username\":\"Username\",\"Password\":\"System.Security.SecureString\"}}
command type is: mineService.JSON.Message@3252f033
login info Username

Message: {"type":"MCCCommand","message":{"type\":\"Stop\",\"Server\":\"Monster\",\"args\":\"\"}}
command type is: mineService.JSON.Message@9e399a6
```

---

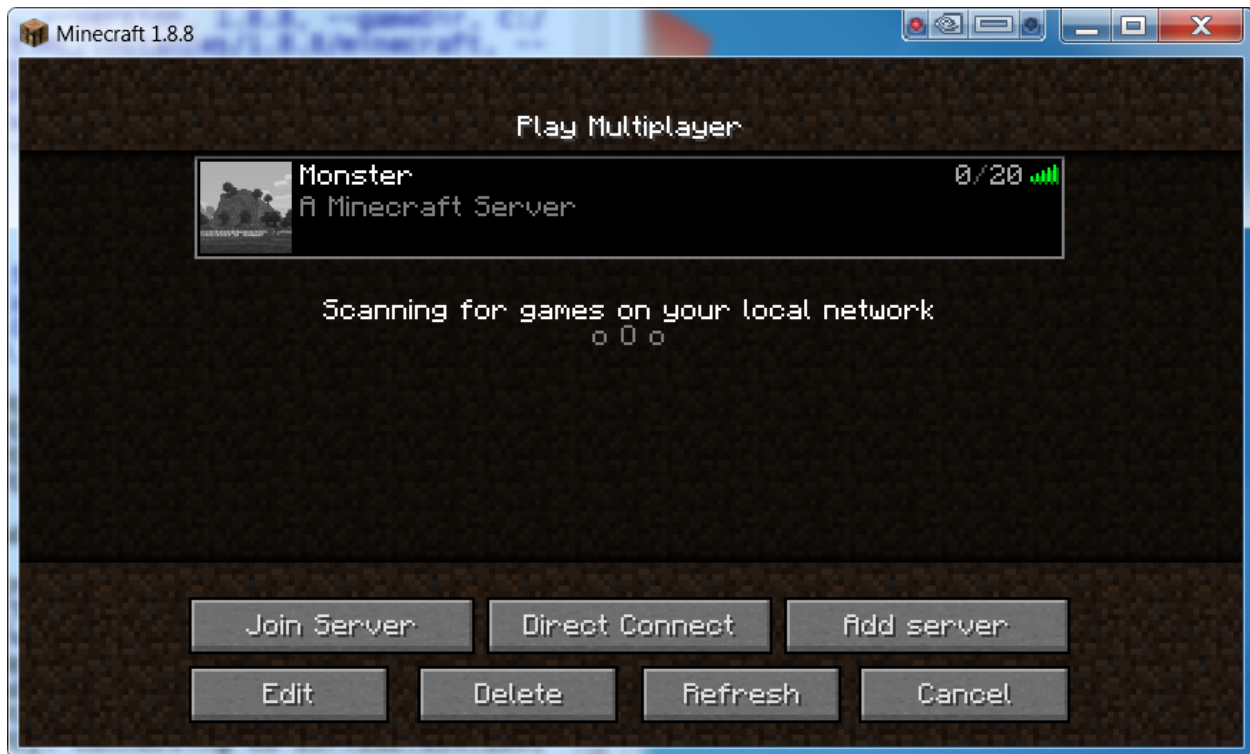
## Conditions of Satisfaction

- Acknowledge that I don't have to use terminal to manage my servers

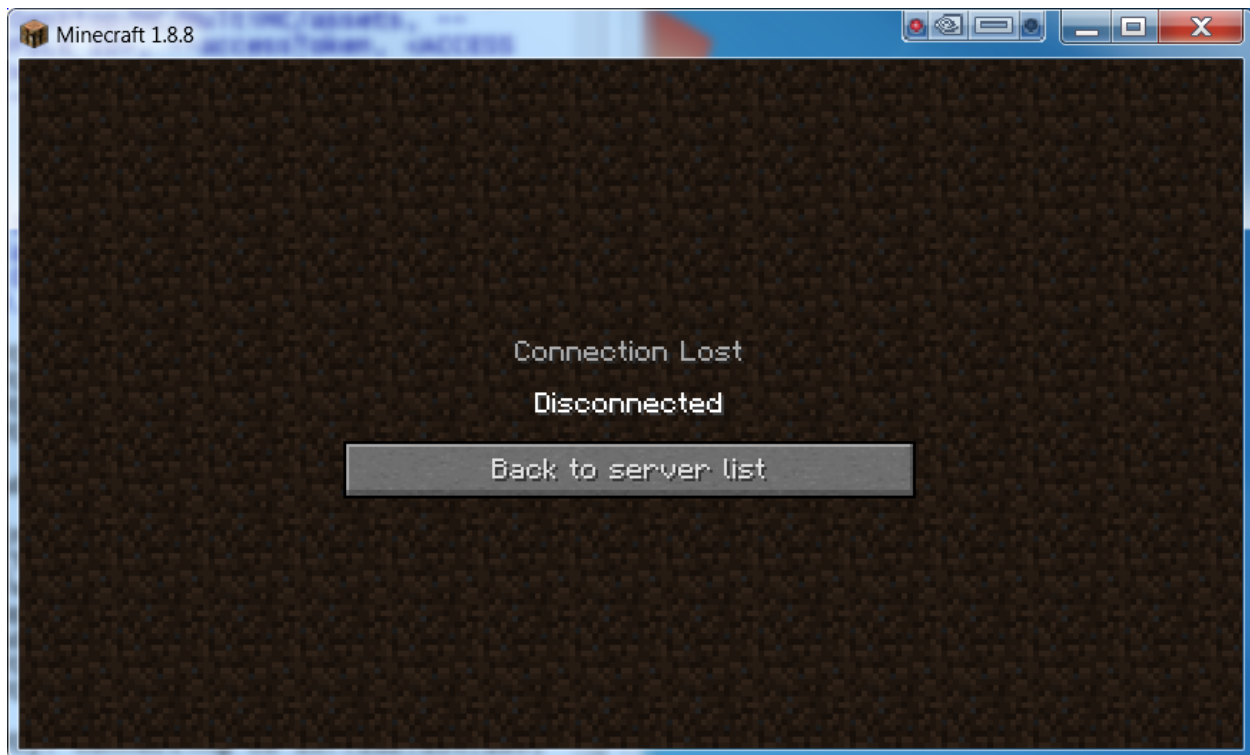
Contributors: Ishank, C.J., Max, Jeremy

Feature: Handle start and stop requests on the server so that a Minecraft server is started or stopped when requested.

Minecraft client a few second after the start message has been received.



After clicking stop on MineService this is seen on the Minecraft client.



On the MineService Server Console the following is seen:

```
{ "type": "MCCommand", "message": "{ \"type\": \"Start\", \"Server\": \"Monster\", \"args\": \"\" }" }
{ "type": "MCCommand", "message": "{ \"type\": \"Stop\", \"Server\": \"Monster\", \"args\": \"\" }" }
```

## Conditions of Satisfaction

- When a stop message is received the running server is stopped
- When a start message is received the server is started and only if it is not already running.

Contributors: CJ, Jeremy, Ishank

Feature: Users are authenticated against a database to be able to login.

Shows on the server when a user logs in a good password:

```
Client Connected: /127.0.0.1
{ "type": "Login", "message": "{ \"Username\": \"Username\", \"Password\": \"System.Security.SecureString\" }" }
login info Username

Login: Success|
```

From a bad user/pass combo.

```
Client Connected: /127.0.0.1
{ "type": "Login", "message": "{ \"Username\": \"Username\", \"Password\": \"System.Security.SecureString\" }" }
login info Username

Login: Denied
```

## Conditions of Satisfaction

- A login that is stored allows you to connect.
- A login that is not valid disconnects you.

Contributors: Jeremy, CJ

## Appendix

- Items available upon request:
  - Interview documents
  - Usage Diagrams
  - Initial Concept Pictures and Notes
  - Notes Taken on User usage