

# MVHS Schedule Tracker Guide

William Hooper and Aleksis Vilums

May 5, 2022

## Contents

<b>1</b>	<b>Modifying the Schedule</b>	<b>1</b>
1.1	Schedule Format . . . . .	1
1.2	Default Schedules . . . . .	2
1.3	Special Schedules . . . . .	2
<b>2</b>	<b>Setting up a new Raspberry Pi</b>	<b>2</b>

## 1 Modifying the Schedule

### 1.1 Schedule Format

In order to start modifying and creating schedules for the tool you must first understand how the schedules are read by the tracker. Lets give an example schedule (Figure 1).

Each schedule is represented by a string containing all of the periods. Each period is separated by a ',', so a schedule can be simplified as:

`<period 1>,<period 2>,...,<period n>`

Each period has its own syntax aswell. Each bit of information about it is separated by a ';'. The first bit is the start time, the second is the name, and the third is the end time. The times are set in 24-hour time.

`<start time>;<name>;<end time>`

So, using our schedule as an example, first period would look like this:

`7:34;Period 1;8:28`

And, finally, the schedule will look like this:

`7:34;Period 1;8:28,8:28;Passing Period;8:32,8:32;Period 2;9:26,...`

Any period of time not included in the interval will result in the progress bar disappearing.

As for the problem of B lunch, it was solved by making a separate schedule for 5th period (4th on even block day).

## 1.2 Default Schedules

Most days will follow a specific pattern. Monday, tuesday, and friday are the same, with Wednesday and Thursday being unique. In the bar.js file located in the js directory, default schedules are set by the dateSchedule() method. The second schedule for each is simply the B lunch schedule.

## 1.3 Special Schedules

Not everyday will have the normal schedules, so a basic api has been setup so that the page can dynamically fetch different schedules, without modifying the javascript.

This is done by using the schedules.json file located in the api directory. To update it, simply make a commit to the file on the github repository that hosts the site.

It is made up of a "schedules" array, with each element containing a schedule. The first item is "date", which specifies which day it should replace, in MM/DD/YYYY format. The "times" item is an array of two schedules, the first being the main schedule, and the second schedule being the B lunch schedule.

# 2 Setting up a new Raspberry Pi

First, flash the raspberry pi with the latest version of raspberry pi os. Make sure that it is the full version and not the lite version, so that you start with a desktop environment. After booting, follow the directions, but make sure to call the user "schedule".

Second, download setup.sh from the tools directory in the repository. Now open the terminal and navigate to the directory in which it was placed, and run these commands to make it executeable and run it:

```
chmod a+x setup.sh
./setup.sh
```

Finally, run

```
sudo raspi-config
```

, and navigate to display options; screen blanking and disable it to prevent the Pi from going to sleep while displaying the schedule.

Reboot the Pi to ensure everything is working properly, you may have to enable auto-login, but that should be enabled by default.

<b>Monday</b>
<b>Period 1</b> <b>7:34-8:28</b>
<b>Period 2</b> <b>8:32-9:26</b>
<b>Period 3</b> <b>9:30-10:24</b>
<b>Period 4</b> <b>10:28-11:22</b>
<b>Period 5</b> <b>A Lunch 11:22-11:56</b> <b>Class 12:00-12:54</b>  <b>Class 11:26-12:20</b> <b>B Lunch 12:20-12:54</b>
<b>Period 6</b> <b>12:58-1:52</b>
<b>Period 7</b> <b>1:56-2:50</b>

Figure 1: Example schedule.