How to create a magic mirror calendar

Zach Martin

28 December 2021

The purpose of this document is to document the steps taken to create a “magic mirror” type of digital calendar/information station. The goal of this digital calendar, dubbed “SmartWall” by myself, is for a google calendar to be displayed on the wall on a TV monitor. Later on, this smartwall may include weather reports and news headlines if desired. These are standard modules in the software I will be using.

Materials required

- Raspberry pi 3,3b+, 2.
- HDMI capable monitor
- HDMI cables, power adapters
- SD card for pi image

Configuration

1 Pi Image

Setup the raspberry pi image using the raspberry pi imager software
https://www.raspberrypi.com/software/

Install the imager (usually .deb). Once installed, run the imager program.

Pretty simple program. Choose the SD card and type of OS. Use raspbian with desktop environment.

Use the advanced settings to setup SSH and wifi by using the key combo control + shift + x while in the raspbian installer. This brings up the menu in figure 2.

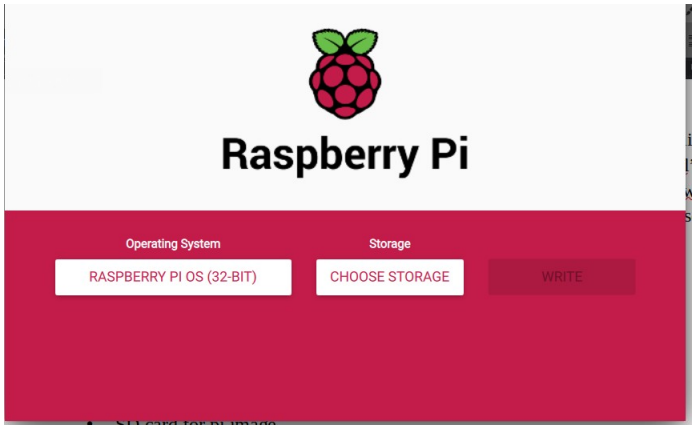

Figure 1: Imager

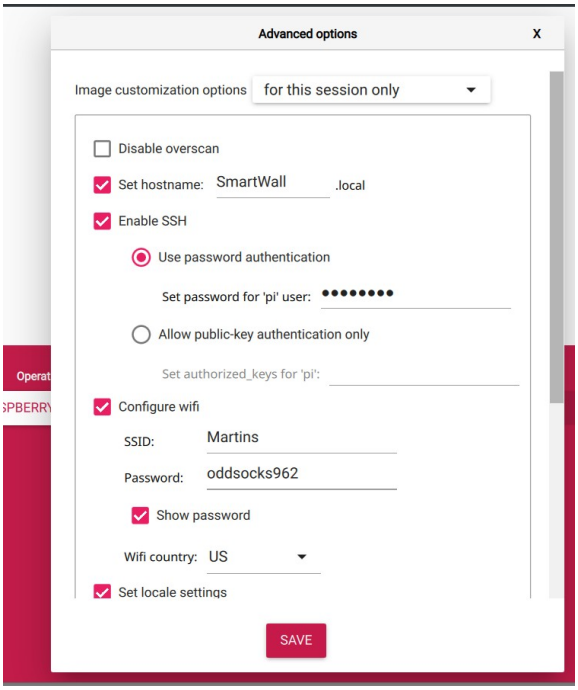

Figure 2: Advanced Options

Once installation is complete, insert card into Raspberry pi and apply power.

2 Pi config

Disable screen saver / screen blanking by setting the raspberry pi configuration screen blanking to “disabled.” There’s probably a CLI way to do this. Supposedly you can do this via “xset s off” in cli, but it did not work via SSH.

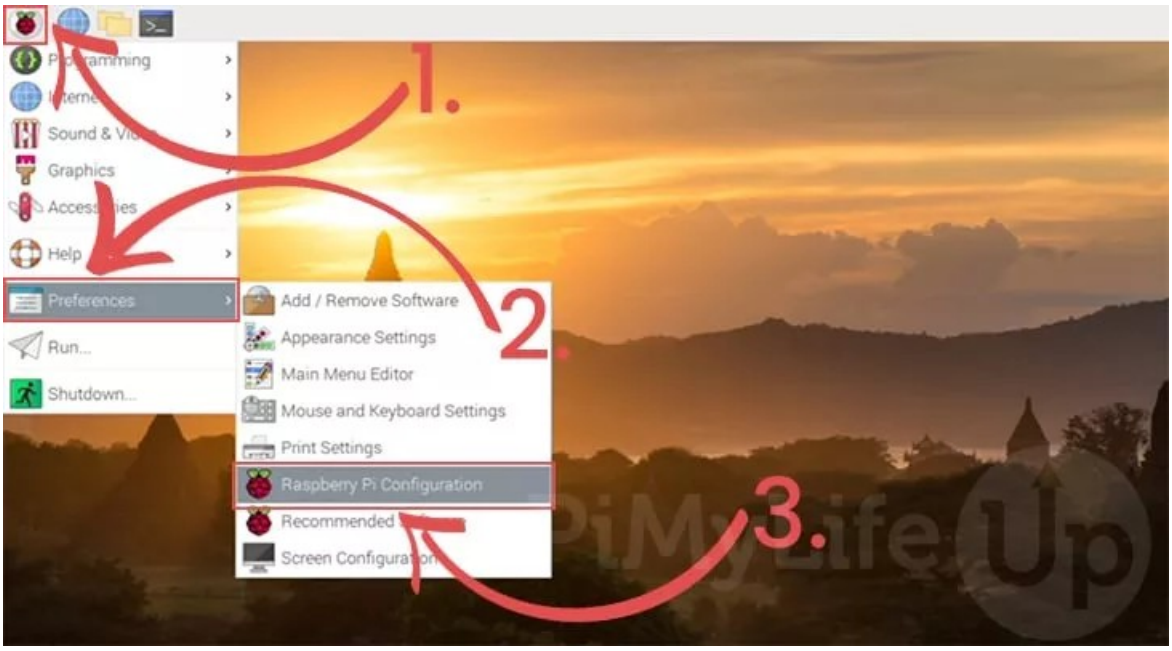

Figure 3: Disable screen blanking steps pt 1

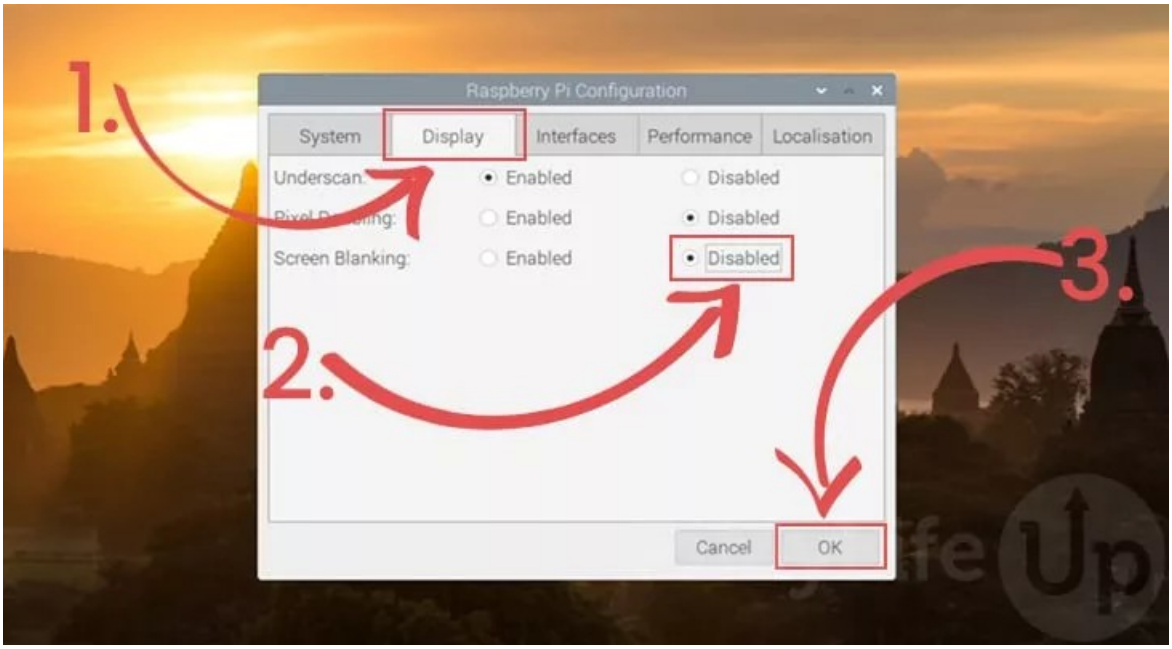

Figure 4: Disable screen blanking pt 2

Rotate screen

As the raspberry pi has switched from “legacy” display drivers to x11 display drivers, the old trick of using display_rotate=1,2, or 3 in the configuration text file does not work anymore. Instead, the display has to be rotated using xrandr. For some reason I was having trouble getting the .xprofile or .xinitrc trick to work, so I added an xrandr config to my startup.sh. However, there needs to be a delay in order for this to work. 15 seconds works OK.

Here’s my startup.sh

```
#!/bin/sh
export XAUTHORITY=/home/pi/.Xauthority
export DISPLAY=:0.0
sleep 15
xrandr --output HDMI-1 --rotate right
chromium-browser calendar.google.com/calendar/u/0/r --start-
fullscreen
```

Start this by adding a crontab entry @reboot.

3 Magicmirror2 Install/config

1. Download and install the latest *Node.js* version:

- `curl -sL https://deb.nodesource.com/setup_16.x | sudo -E bash -`
- `sudo apt install -y nodejs`

2. Clone the repository and check out the master branch: `git clone https://github.com/MichMich/MagicMirror`
3. Enter the repository: `cd MagicMirror/`
4. Install the application: `npm install`
5. Make a copy of the config sample file: `cp config/config.js.sample config/config.js`
6. Start the application: `npm run start`

This concludes a basic setup of the magicmirror software. It will launch with a default page.

Settings for the program are stored in config/config.js. Open that file in a text editor (IE vi or nano) and you can change the time settings (24/12h) the measurements (metric/imperial) and which modules are enabled.

4 Google Calendar Setup

There needs to be a google calendar for this project in Ical format. Supposedly this is a URL to a “secret” address. Meaning, the calendar is not “public” but the address can really be accessed by anyone if given the URL. Go to calendar.google.com, to your settings, and click on your calendar in the bottom left as in the image below.

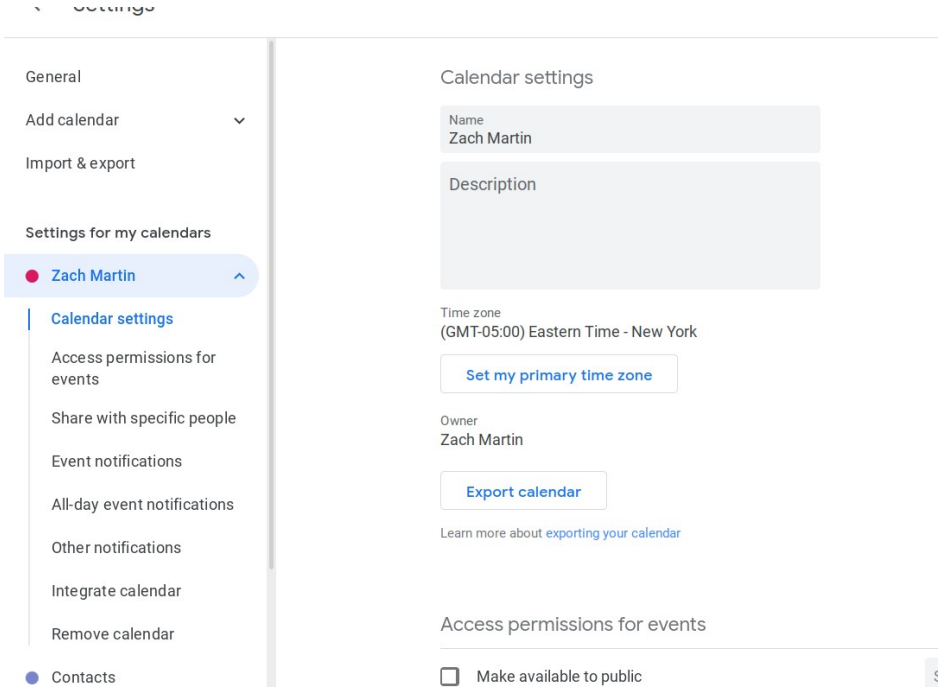

Figure 5: Settings Page

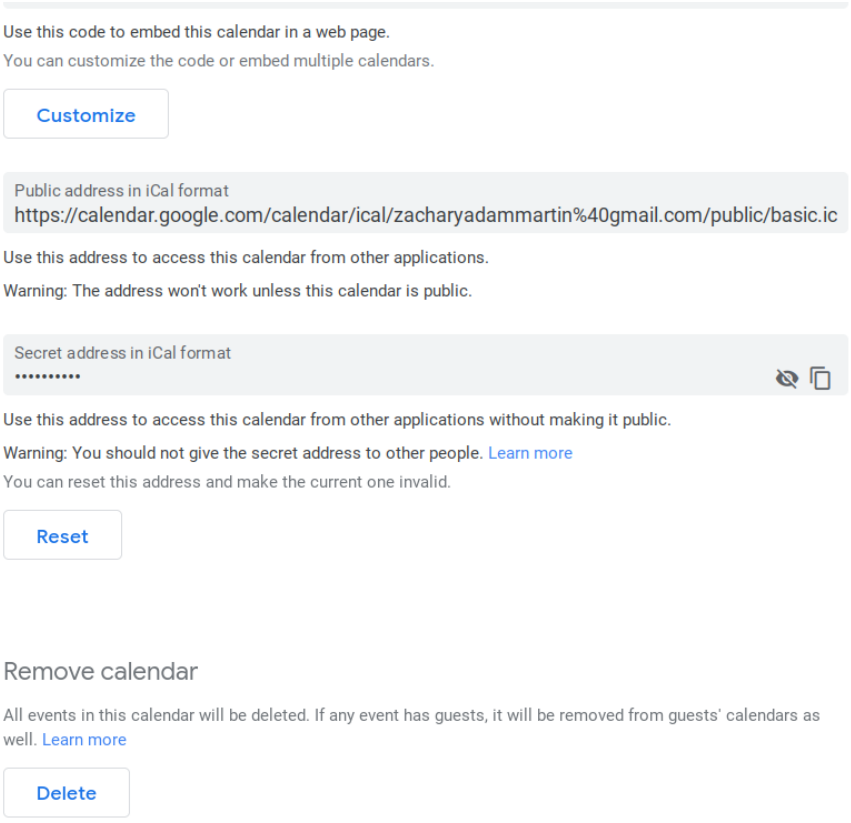

Figure 6: Secret Address location

Scroll down to the “Secret Address” section in the bottom of the page and click the eye icon to the right to view the address. Copy this address in the MM2 config file in the “modules” section under “calendar”

```
modules: [
        {
                module: "alert",
        },
        {
                module: "updatenotification",
                position: "top_bar"
        },
        {
                module: "clock",
                position: "top_left"
        },
        {
                module: "calendar",
                header: "US Holidays",
                position: "top_left",
                config: {
                        calendars: [
                                {
                                        symbol: "calendar-check",
                                        url: "https://calendar.google.com/calendar/
                                }
                        ]
                }
        },
        {
                module: "compliments",
                position: "lower_third"
        },
        {
                module: "weather",
                position: "top_right",
                config: {
                        weatherProvider: "openweathermap",
                        type: "current",
```

Figure 7: Calendar Module Settings

This sets up a pretty basic calendar with a scrolling “agenda” style of window. For a full screen full calendar setup, I can use the MMM-MonthlyCalendar module. https://github.com/kolbyjack/MMM-MonthlyCalendar

1 Installation of MMM-cal

In your terminal, go to your MagicMirror's Module folder:

```
cd ~/MagicMirror/modules
```

Clone this repository:

```
git clone https://github.com/kolbyjack/MMM-MonthlyCalendar.git
```

Configure the module in your `config.js` file.

Note: After starting the Mirror, it will take a few seconds before events start to appear.

2 Using the module MMM-cal

To use this module, add it to the modules array in the `config/config.js` file:

```
modules: [
```

```
  {
    module: "MMM-MonthlyCalendar",
    position: "bottom_bar",
    config: { // See "Configuration options" for more information.
      mode: "fourWeeks",
    }
  }
]
```

3 Configuration options

The following properties can be configured:

Option	Default	Description
`mode`	`"currentMonth"`	Which type of calendar to show. Possible values are `currentMonth`, `currentWeek`, `twoWeeks`, `threeWeeks`, and `fourWeeks`.
`firstDayOfWeek`	`"sunday"`	Which day to use as the start of the week.
`showWeekNumber`	`false`	Whether to show the week number of the first day of each row.
`displaySymbol`	`false`	Whether to show symbols next to events.
`hideCalendars`	`[]`	A list of calendar names to hide from the view.

NOTE: the built in calendar module MUST be enabled with the google ical URL otherwise the MMM-cal module will not work. You can leave the position line blank for the default calendar this way the default calendar is not displayed.

5 Weather

There is a built in weather forecaster that may be useful for this application. Use the following settings

```
        {
            module: "weather",
            position: "top_right",
            config: {
                weatherProvider: "openweathermap",
                type: "current",
                location: "New York",
                locationID: "5128581", //ID from http://bulk.openweathermap.org/sample/city.list.json.gz; unzip the gz file and find your city
                apiKey: "YOUR_OPENWEATHER_API_KEY"
            }
        },
        {
            module: "weather",
            position: "top_right",
            header: "Weather Forecast",
            config: {
                weatherProvider: "openweathermap",
                type: "forecast",
```

```
            location: "New York",
            locationID: "5128581", //ID from http://bulk.openweathermap.org/sample/city.list.json.gz; unzip the gz file and find
your city
            apiKey: "YOUR_OPENWEATHER_API_KEY"
        }
    },
```

Find an API key by creating a free account with openweathermap and clicking the “subscribe” button for an API key via email.

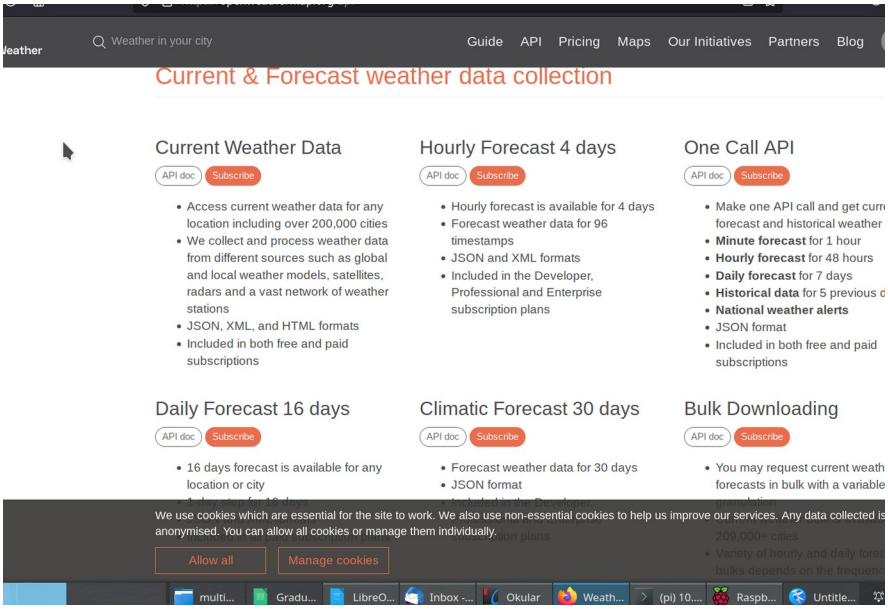

Figure 8: OpenWeatherMap

6 Using MM2 and Chromium in side-by side Mode

Ultimately, we decided to remove the MM-calendar and instead run a window with chromium below a window with MM2. This way we could have a full color google calendar in addition to the features that MM2 allows. The settings are as follows:

1 Create a script and add it to cron

Touch startup.sh

crontab -e

add the folowing line: `@reboot export DISPLAY=:0 && sh /home/pi/startup.sh`

In startup.sh, add the following:

```
#!/bin/sh
export XAUTHORITY=/home/pi/.Xauthority
export DISPLAY=:0.0
sleep 15            # necessary for xrandr else xrandr will run too soon
xrandr --output HDMI-1 --rotate right # portrait mode
chromium-browser --app=https://calendar.google.com/calendar/u/0/r --window-size=1080,1080 --
disable-translate --fast --fast-start --disable-infobars &
npm run start --prefix /home/pi/MagicMirror/
```

change your MagicMirror/config/config.js to have the following after “let config = { “

```
electronOptions: {y: 1080, width: 1080, height: 840, fullscreen: false, kiosk: false },
```

This way you can choose the y-coordinate or x-coordinate, height, width, etc of the thingy.