Github Repo: <https://github.com/prime-electric/pnw-prefab-calendar>

This doc should show all the information in the README plus a bit more information

**What is it:**

An Express.js web app (basically a web server written in Javascript with fancy modules) that uses OAuth2 to access the Microsoft Graph API to get event data to be displayed in a 2 month calendar view (other views are also available but not used). **Ubuntu** will be the preferred OS to run on. For more info on Express.js - <https://expressjs.com/>

**How to set it up on a machine:**

First of all, get chrome installed before anything

1. Install Node.js
   1. On Ubuntu run the commands

Update package list

$ sudo apt update

Install using package manager

$ sudo apt install nodejs

Verify installation

$ nodejs --version

1. For Windows just install from this link: <https://nodejs.org/en/download/>
2. Install npm (Node Package Manager) *Ubuntu Machines ONLY*
   1. Run the command

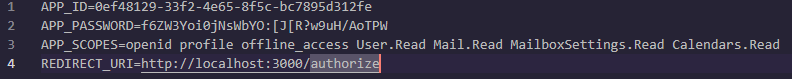
$ sudo apt install npm

$ npm --version

1. Configuring the Express server
   1. There's a few ways to extract the files that will be up to you
      1. Easy way: Just zip it from the GitHub repo and extract the folder in the documents

This is good if you don't anticipate any changes to the code

1. Harder way: install git and initialize a local repo. This takes some more knowledge of git, but it does allow you to pull changes from the remote repo. More info: <https://git-scm.com/book/en/v2/Git-Basics-Getting-a-Git-Repository>
2. After you have the folder extracted, run the command $ npm install in the folder to install all the necessary dependencies for the app. This should make a folder called node\_modules
3. The app also needs a .env file to get necessary info about the app. The .env file should be placed in the main app directory and should look like the following. The Password should be a secret and not stored many places so don’t lose this document



1. To make sure everything is installed correctly, run $ npm start. This should pull up stuff in the terminal and if on Ubuntu will pull up a chrome window in kiosk mode most likely on a MS login page. Windows has some difficulties with kiosk mode so you might have to mess around with chrome flags in bin/www.js where there's the line let app = […]; where the body of the list is all the flags chrome is run with
2. Running on startup/ reboot (Ubuntu only, Windows will have to find another way)
   1. The best way to do this is make it a daemon process (process in the background). Run these commands to install PM2 which is a daemon process manager

$ npm install pm2@latest -g

1. Now we need PM2 to generate the startup script. This will generate a command for you to run

$ pm2 startup

1. Then need to start our app as a process. In the app folder run (spaces matter)

$ pm2 start npm -- start

$ pm2 list

#Verify that npm shows up in process list

1. Save the process list you want to respawn on machine reboot with:

$ pm2 save

1. Reboot your machine and verify the app launches with a chrome browser or the process is started with

$ pm2 list

PM2 Reference: <http://pm2.keymetrics.io/docs/usage/startup/>

1. Make sure autologin is on for the account the app is running on
2. For the account you want the calendar from, make sure you turn on save password on the MS login, so it's not prompted every time

Note: The Ubuntu keychain might bug you every time you open chrome so it may be necessary to mess with those settings

**How it starts:**

If all is set up correctly, on boot the app should pull up chrome and show the 2 month view of the calendar

**Notes:**

* This app should only require the login once since the access token should refresh often. This only works with cookies though so don’t disable them
* If you want to change to a different account and you’ve already logged in a MS account that has save password, it won’t let you change the user unless you clear the browser cache