Get My

A simple list application with syncing and scaling for web or mobile.

- Install Get My

Install Get My

https://www.youtube.com/embed/hbGNTzU24hI

I built Get My to fill a gap I perceived in the open source world of list apps. There are tons of them to be certain, but none really fit couple of things I wanted, and my family needed. Also, I just wanted to make something useful.

Quick References

Get My Source Code: https://gitlab.com/bmcgonag/get_my

Docker Hub for Get My: https://hub.docker.com/r/bmcgonag/get_my

What you'll need

- A host system / server to run the application on.
- Docker and Docker Compose installed on the host system
- (optional) reverse proxy (I use NginX Proxy Manager)
- (optional) a domain or subdomain that you can set A Records for
- (optional) a VPN
- About 30 minutes of your time

Installation via a Simple Script

You can easily install Docker-CE, Docker-Compose, Portainer-CE, and NGinX Proxy manager by using this quick install script I created and maintain on Github. Just use the command:

```
wget -O install-docker.sh https://gitlab.com/bmcgonag/docker_installs/-/raw/main/install_docker_nproxyman.sh
```

To download the script to your desired host.

Change the permissions to make the script executable:

```
chmod +x ./install-docker.sh
```

and then run the script with the command:

```
./install-docker.sh
```

When run, the script will prompt you to select your host operating system, then will ask you which bits of software you want to install.

Simply enter 'y' for each thing you want to install.

At some point, you may be asked for your super user (sudo) password as well.

Allow the script to complete installation.

At this point, you might want to log out and back in, as this will allow you to use the `docker` and `docker-compose` commands without the need of sudo in front of them.

Install Get My in Docker

First, let's make a directory structure, where we have a parent level folder called "docker" to put all of our docker applications in. This is just organizational, so you don't have to do it, but it helps keep things easier to find in my opinion.

`mkdir -p docker/get-my`

The above command will create the full path with a docker folder as the parent, and the 'ge-my' folder inside of it. Now, let's move into the get-my folder:

`cd docker/get-my`

Inside this folder, we'll make a new file called "compose.yaml".

`nano compose.yaml`

Once inside the nano text editor, we'll copy the code below, and paste it in, either using the right-click context menu, or you can paste into the linux terminal with CTRL + Shift + V.

> NOTE: if you have a CPU without the AVX instruction set, you can change the line `image: mongo` to be `image: mongo:4.4` in the docker compose below.

```
---
services:
  get_my:
    container_name: get_my
    image: bmcgonag/get_my:latest
    ports:
      - "80:3000"    # you can change the left side to a less common port number if needed
    healthcheck:
      test: curl --fail -s http://localhost:3000/ || exit 1
```

```
      interval: 30s
      timeout: 10s
      retries: 3
    links:
      - mongo
    depends_on:
      - mongo
    restart: unless-stopped
  mongo:
    container_name: get_my-mongo
    image: mongo
    volumes:
      - ./data:/data/db    # this will store the application data in the same directory where
you have the compose.yaml file.
    restart: unless-stopped
```

In the snippet above, you can change the left side port in the port mapping. You should do this if you are going to be setting up a Reverse Proxy in order to reach your application from outside your network.

In my video, I changed it from 80 to 8030. This is the port you'll access the app on locally if you only want to use it on your local network, or if you want to access it from outside on a VPN, you can use the VPN IP of the host machine, and this port.

Save the file with CTRL + O, then press Enter to confirm, and exit the nano editor with CTRL + X.

Now we'll pull down the images needed for Get My to work. These are the Get My image, and the mongo image from docker.

`docker compose pull`

You should see some output showing the various parts of the images being pulled down, and then be back at a prompt int he terminal window.

Now we can start up Get My with the command:

`docker compose up -d && docker compose logs -f`

This one-liner, is actually two commands concatenated with the '&&' symbols. The first part tells docker to start the containers based on the images it just pulled down. The second part says we want to follow the logs as the applications are brought up. You don't have to follow the logs, I just do this in order to watch for anything that looks like it might be an error.

If all goes well, you'll see the application containers start up, and the logs will scroll la bunch of JSON type data up the screen. IF you don't see anything that jumps out as an error, then you can

quit watching the logs with CTRL + C.

Now you're ready to access your GetMy application. OPen your favorite modern browser, and go to http://<ip address of your host>:<port you entered on the left side of the mapping>. For me, it was

http://192.168.10.157:8030

Once you have the main page loaded, you should see the Get My title, and a "login" option at th upper right corner. Tap Login, then select the "Register" option at the bottom of the login form.

Register with your name, email, and password. The first registered user is automatically made a system admin user, with access to everything. All subsequently registered users are set as 'user' role, and will have access only to things that are shared by other users (e.g. lists) and items that are theirs (their own private lists, tasks assigned to them, and menus.

I suggest going to the Manage option int he upper title bar (or the hamburger menu on a mobile device) and adding some stores. Then if you have a bunch of products you regularly buy, go to Products, and add those items as well. This just builds a list of items for you and / or your rusers to choose from when adding items to a list.

If you want a grocery list, for instance, and you want that shared with your family, make sure to create it under the "My LIsts", then check the box to share it with everyone. Now you can all add items for the list.

Tapping on a list item will mark through it with a red line, and tapping it again will unmark it. This is intended to be used as a "Ordered but not received" type of mark. Once received, you can tap the check mark for that item to mark it as received. NOTE: you don't have to mark an item ordered in order to mark it received.

You can also delete items from the list if you have duplicates, or items you don't want to get after all. Just tap the trash can icon, and then confirm the deletion in the pop-up confirmation box.

You can add tasks as well. from 'My Tasks'. you can add tasks for yourself, and from Manage >> Tasks you can add tasks for yourself, or other users on the system. You add tasks by typing the task name, then pressing Enter, and you'll see it turn into a chip. This is to help make multiple tasks that will be assigned to the same user for the same date(s). You can also add multiple dates before submitting the task(s) so that you can enter multiple repeating tasks.

You can create Menus, then add items to the menus, and assign which date you want to have that menu item on. I'll be expanding this functionality in time to help build out a shopping list as well.

Access Get My from outside your Local Network

If you want to have access to the Get My app outside of your local home network, you can use a VPN, then make sure the host computer for Get My is on it, and you'll need to put the mobile

device(s) on the VPN as well, then use the hosts VPN address and the same port number to access Get My through your VPN connection. If setup for Wireguard for instance, this can give you encrypted traffic outside of your home, and won't require you to open any firewall ports.

If, however you want to access Get My without a VPN, you should setup a Reverse Proxy. There are many reverse proxies out there, but I like NGinX Proxy Manager. It's a web GUI that helps setup NGinX as a reverse proxy for all of your services inside your network.

First, you need to own a domain that you want to use for the Get My site. I own the domain "routmehome.org".

Next, you need to setup an A record in your domain registrar's DNS settings for the subdomain / domain you'll be using to get to the Get My app. I setup list.routemehome.org, ,and made sure the A record pointed to my public IP address. You can check your public IP address by going to a site like https://ipchicken.com .

Next, you need to forward ports 80 and 443 through your firewall (router / firewall) to the IP address of the machine you are hosting Get My from. There are literally dozens fo router and firewall makers out there, and they all differ slightly, so I recommen you look up how to do port forwarding on your model router / firewall.

Now, you need to set a new entry in your reverse proxy. In NGinX Proxy Manager, we go into the Proxy Hosts tab, then click the 'Add' option in the upper right.

Now fill in the subdomain / domain you want to access Get My from. I entere "list.reoutemehome.org". Next, move to the IP address field, and enter the host IP address (local / internal network IP address). Now move to the port field, and enter the port number you put on the left side of the port mapping in the compose.yaml file. IF you left it as 880, you still need to type in 80 in this field.

Next, enable 'Block Common Exploits', and 'Websocket Support'.

Now move to the SSL tab. Click on the drop-down that says 'None', and select 'Request a new certificate'.

Enable the options 'Force SSL' and 'HTTP/2 Support' at a minimum. I recommend you also enable the two HSTS options.

Now enter your email if its not already entered, then accept the Terms of Service for LetsEncrypt, and click 'Save'.

This will take about 30 seconds to one minute to complete, but the pop-up should just close with no errors. This means you now have an SSL encrypte site for your Get My app. You can go to your browser, and enter the address you made for your Get My app, and you should be taken to it.

Support My Channel and Content

Support my Channel and ongoing efforts through Patreon:

https://www.patreon.com/awesomeopensource

Buy me a Beer / Coffee:
https://paypal.me/BrianMcGonagill