Setting up the server environment.

Note: The following instructions are targeted for Linux and macOS environments. If you are using a Windows environment, adjust the CLI commands accordingly.

Prerequisites:

1. Docker
    a. To install docker, go to https://docs.docker.com/install/ and choose the appropriate installation instructions.
    b. Select the Community Edition.
2. Node and npm
    a. Go to https://docs.npmjs.com/downloading-and-installing-node-js-and-npm and follow the instructions for your environment.
    b. It is also recommended to install NVM which can be found on the same page. NVM is for managing node versions.
3. Git
    a. Go to https://git-scm.com/downloads and follow the instructions for installing git in your environment.

Clone:

1. Open a terminal in you preferred directory, such as home.
   `cd ~`
2. Clone the repository:
   `git clone https://github.com/kenbonilla/irondb.git`

Automatically install and launch containers:

Note: This automatic method assumes you are running Linux or macOS. If you are using Windows, follow the instructions for running manually.

1. Open the root directory:
   `cd ~/irondb`
2. (Optional - do this if the next step does not work) Set permissions:
   `chmod a+x iron.sh`
3. Run the setup script:
   `./iron.sh`
   You will be given several options.
   1 - Install dependencies, build the containers, and launch
   This should be run the first time you are launching IronDB or in case of wanting to rebuild all dependencies and containers.
   2 - Rebuild containers and launch

Rebuild the containers but do not reinstall dependencies.

3 - Launch pre-built containers

Perform a normal launch of the containers that have been built prior.

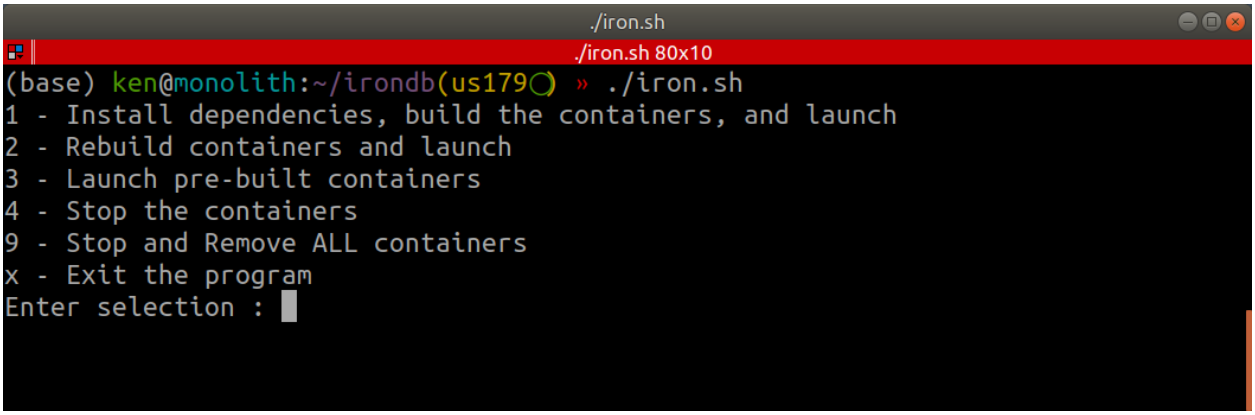
4 - Stop the containers

Performs the docker-compose down command

9 - Stop and Remove ALL containers

If performing a fresh install (1) does not correct the issues, do this and then try a fresh install again.

x - Exit the program



4. Go to your browser and navigate to http://localhost:3001/ . This should bring you to the landing page.

Manually install and launch the containers:

1. Enter the root directory:
   `cd ~/irondb`
2. Install the npm dependencies:

```
npm install
sudo npm install -g gulp-cli
sudo npm install -g jest-cli
gulp sass
gulp js
```

3. **Warning!** *This will delete the local copy of the postgres database, backup important files before continuing.* If the root directory has a folder name **pg-data** then:
   `sudo rm -rf pg-data`
4. Make the postgres folder:
   `mkdir pg-data`
5. Remove previous files and directories:

```
sudo rm -rf pg-data
sudo rm -rf node-modules
```

```
sudo rm -rf rabbitmq/data/.erlang.cookie
```

6. Build and launch the containers: `docker-compose up --build -d`
   The `--build` flag is important for ensuring that you are not trying to run an outdated container.
   The `-d` flag is for running the containers in a detached state.
7. Navigate to http://localhost:3001/ which will take you to the landing page.

Shutdown procedures:

Open the controller application `./iron.sh` and then select **4**.
OR
Enter the command `docker-compose down`

Restart the server:

To restart the server using pre-built containers, open the controller application:
`./iron.sh` and then select **3**.

--- OR ---

To rebuild the containers before restarting the server, open the controller application:
`./iron.sh` and then select **2**.

In case of docker errors:

Note: When docker is shut down improperly, it may result in errors launching containers in the future. Attempt this before launch in case of docker errors.
Open the controller application `./iron.sh` and then select **9**. Reopen ./iron.sh and select 1.

--- OR ---

If using Windows, instead of `./iron.sh` do this before following the instructions to manually build and run containers.

```
docker-compose down
docker stop $(docker ps -aq)
docker rm $(docker ps -aq)
```

(EOF)