Team JSON Bourne

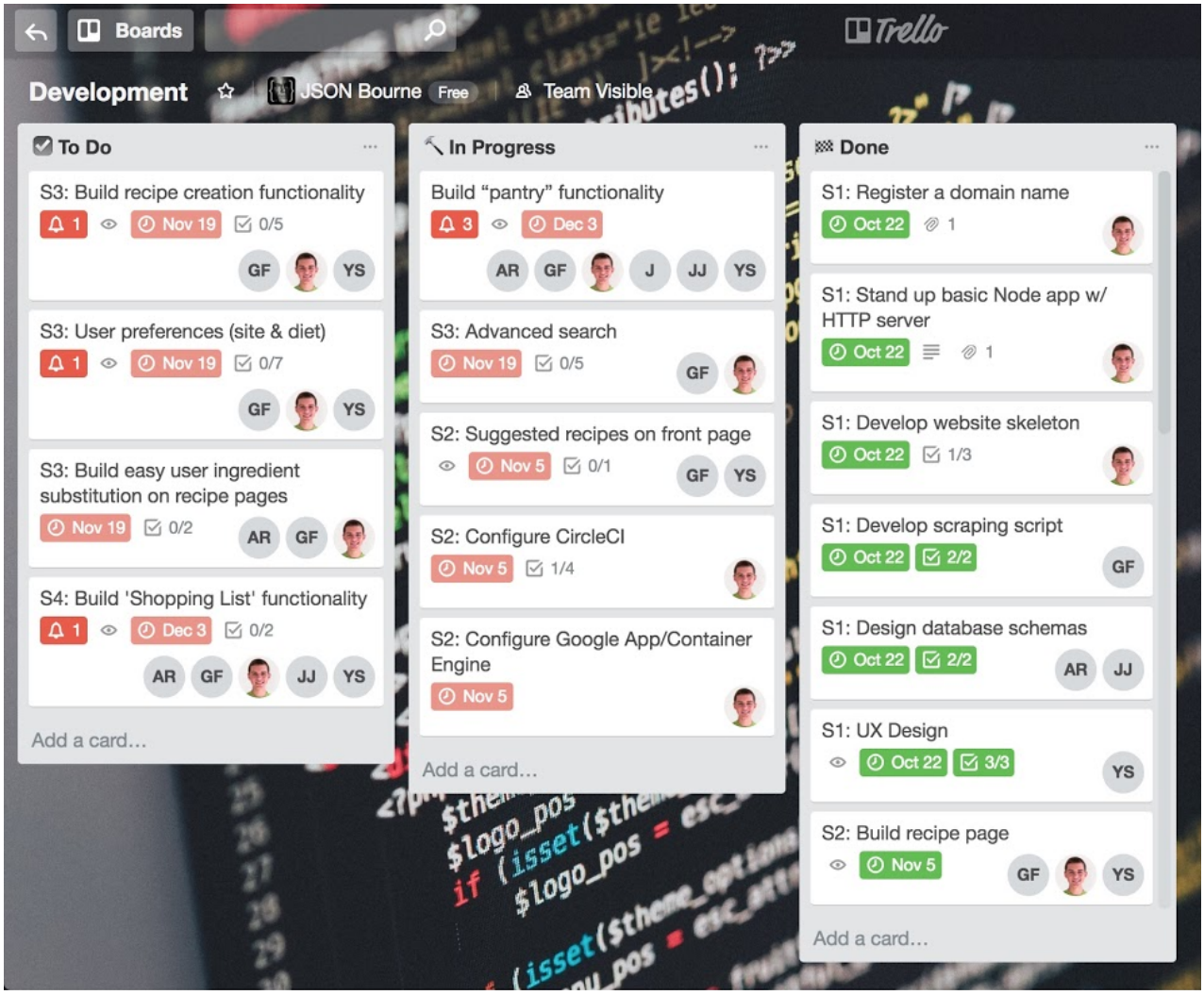
Milestone 7

Title: FindMyAppetite

Who: Grant Burry, Jake Johnson, Yongbo Shu, Gabe Faber, Austin Rugh

Project Tracker: Trello

- https://trello.com/b/lh1oVnvp/development



- **VCS:** GitHub
    - https://github.com/Burry/JSON-Bourne
    - Commit contributions:

Deployment Instructions:

1. Install Yarn.js:
    a. macOS:
        i. Install Homebrew if you haven't already: `/usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"`
        ii. `brew install yarn`
    b. Windows: https://yarnpkg.com/latest.msi
    c. Ubuntu/Debian:
        i. `curl -sS https://dl.yarnpkg.com/debian/pubkey.gpg | sudo apt-key add -`
        ii. `echo "deb https://dl.yarnpkg.com/debian/ stable main" | sudo tee /etc/apt/sources.list.d/yarn.list`
        iii. `sudo apt-get update && sudo apt-get install yarn`
2. Install Node.js v8.x with nvm:
    a. `curl -o- https://raw.githubusercontent.com/creationix/nvm/v0.33.8/install.sh | bash`
    b. `nvm install 8`
3. Install Gulp.js globally:
    a. `yarn global add gulpjs/gulp#4.0`
4. Install Python dependencies:
    a. Install pip if you haven't already: https://pip.pypa.io/en/stable/installing
    b. `sudo pip install requests beautifulsoup4 selenium nltk`
5. Install & Start MongoDB:
    a. macOS: `brew install mongodb`
    b. Windows: https://docs.mongodb.com/manual/tutorial/install-mongodb-on-windows
    c. Ubuntu: https://docs.mongodb.com/manual/tutorial/install-mongodb-on-ubuntu
    d. Debian: https://docs.mongodb.com/manual/tutorial/install-mongodb-on-debian
    e. Create an empty directory in your root path named *data* and within that a directory named *db*
    f. If you aren't already running it as a daemon, start the database with `mongod`. You can open another shell and run `mongo` to access the MongoDB client.
6. Install & Start PostgreSQL:
    a. macOS: http://postgresapp.com
    b. Windows: https://www.postgresql.org/download/windows
    c. Ubuntu: https://www.postgresql.org/download/linux/ubuntu
    d. Debian: https://www.postgresql.org/download/linux/debian
    e. Start PostgreSQL server (for non-macOS Postgres.app): https://www.postgresql.org/docs/9.3/static/server-start.html
    f. Note the server credentials
7. Create *findmyappetite* PostgreSQL database:
    a. `psql postgres`
    b. `CREATE DATABASE findmyappetite;`
8. Clone/download repository: `git clone https://github.com/Burry/JSON-Bourne.git`
9. Enter the repository directory: `cd JSON-Bourne`
10. Install Node dependencies: `yarn install`
11. Create a new file named *.env* and add the following text. Remember to input your PostgreSQL server credentials:

```
NODE_ENV=development
PORT=3000

# Session secret
SESSION_SECRET=u=8Adv3Hq]VmWMc[7ftgKRmp]2TZrnQ.u34bpGEC[B]22mqXfCj6avUYo4{/p}fM

# Set the username and password for your PostgreSQL database
PGUSER=[username]
PGPASS=[password]

# Set the URL for your MongoDB database. Will auto-create the database
MONGOURL=localhost/findmyappetite
```

```
# Algolia API credentials
ALGOLIA_APP_ID=4U2NCQ3L4M
ALGOLIA_ADMIN_API_KEY=d9324691a50725be6163ecc61965aa5d

# Setting for how many recipes the scraper should attempt to retrieve
RECIPE_COUNT=100

# Facebook API credentials
FACEBOOK_CLIENT_ID=133189940663934
FACEBOOK_CLIENT_SECRET=ba2687fdb83dba485f31b73ae7ceeaf7
FACEBOOK_CALLBACK_URL=http://localhost:3000/auth/facebook?cb=true
```

12. Build and start the app: `gulp`

Repository Structure:

a. **.circleci** - CircleCI config
b. **bin** - Application entry point (www) and Python scripts
c. **client** - frontend content
    - **images** - site images and icons
    - **scripts** - frontend JavaScript
    - **styles** - SASS stylesheet
    - **views** - Pug page templates
        - **includes** - Reusable template modules
        - **mixins**
            - **auth** - Authentication form modules
        - **pages** - Page templates
    - *layout.pug* - A layout for entire frontend
d. **controllers** - Controller files that handle page routing and db logic
e. **doc** - Documentation
f. **milestones** - Milestone documentation
g. **models** - Database schemas and connection setup
h. **test** - Unit tests
i. *app.js* - Main application
j. *gulpfile.js* - Development task definitions
k. *package.json* - Project dependencies and script definitions

How to Run Unit Tests:

- `yarn test`

Continuous Integration (not fully configured):

- https://circleci.com/gh/Burry/JSON-Bourne