

Project Milestone 3

Team #: 13-05

Team Name: Fruit Leather

Demo Features:

- Demo 1: Website
 - Working Register Page which can interface from angular to springboot to mysql, can take in information and add it to the database
 - Working Login Page which checks if the user exists and redirects if true
 - Enter Ingredients Page in which ingredients can be typed in, not interfacing with backend
- Demo 2: Web Scraping for finding recipes
 - Using Scrapy to return ("md / Daunte")

What Worked In the Demo:

- All features developed so far

What issues were faced during development/demo:

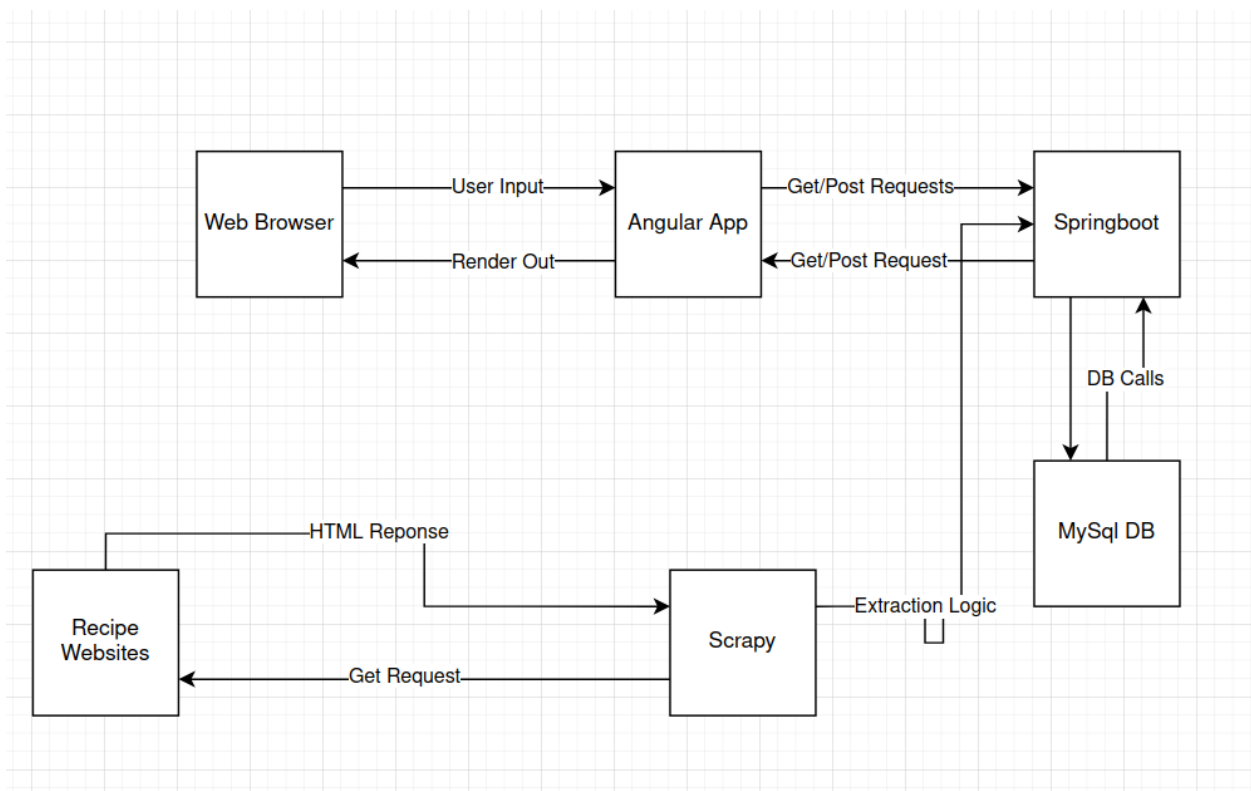
- Mismatch of software versions between our computers caused problems if working together
 - Especially java mismatches
- Not matching different peoples work together, for example two sql creation scripts ended up getting made

What were the suggestions offered by the TA:

- We should ensure we know how things will change in the gui if a user is signed in or .
- We should flesh out the logic for processing a user registration.
- We should not have two apps, one for the front end service and one for webscraping that may be reading and writing to the same database at the same time. This could cause errors. To deal with this, we will use springboot to trigger the webscraping at certain times (application architecture updated to reflect this).
- TA said we should look into checks to make sure that we do not scrape malicious code For example, scrape sql which could be used to mess up our database. Planning on using springboot JPA built in functionality to handle this.
- Also should implement checks in webscraping to make sure we scrape recipes and not home pages or something else
- We should determine the exact websites we are scraping and their url structure so we know what to scrape/crawl

- Ta said we should look further into the database/queries we use to generate recipes from ingredients, we need to look at:
 - Cases in which the ingredients may be misspelled, hyphenated, or otherwise different, in the original source
 - Cases in which no matches arise but some options in the database are very close
 - How to tie together recipes by ingredients when we scrape them to reduce query time

Application Architecture:



Individual Contributions:

- Carson
 - Created the initial database creation script.
 - Latest Commit: <https://github.com/CU-CSCI-3308-Fall-2021/CSCI-3308-Fall21-013-05/commit/190e370c1b93c0e918133b06cddbda797f56a78dc>
- Rachel
 - Developed front end views

- Latest Commit:
<https://github.com/CU-CSCI-3308-Fall-2021/CSCI-3308-Fall21-013-05/commit/ce a1a91e1cfeaba3cc33040c108c7e09cf213c04>
- Ferin Von Reich
 - Worked on conjoining sql to springboot to angular for login and register page with lan (worked locally so no commit)
 - Latest Commit:
<https://github.com/CU-CSCI-3308-Fall-2021/CSCI-3308-Fall21-013-05/commit/40 f328089bcc4fa1ee58bf942b86c21442323f62>
- Md
 - Created web scraper/crawler for simplyrecipes.com
 - Latest Commit:
<https://github.com/CU-CSCI-3308-Fall-2021/CSCI-3308-Fall21-013-05/commit/62 d05f9baf92d212b72d1eaba362bc7ecddd00be>
- Daunte
 - Created alternative webscraper
 - Latest Commit:
<https://github.com/CU-CSCI-3308-Fall-2021/CSCI-3308-Fall21-013-05/commit/65 795d65f34b0bde243dc79a14703ffa260da025>
- Ian McKibben
 - Created login and register
 - Worked on back end
 - Worked on queries for login and register
 - <https://github.com/CU-CSCI-3308-Fall-2021/CSCI-3308-Fall21-013-05/commit/a3 a5f6ab1d4a8bf1d11769c883184c9963646cdc>