Rebecca Nickerson, Jessica Pancer, Fandi Tang, Payal Modi

CPSC 437

**Final Project Report**

**0. Overview**

* Project name: Welp: recommending New Haven restaurants based on menu items
* Concept: on a high level, we created a system with randomized recommendations based on user preferences. We compiled data from menus of restaurants, including main ingredients, cuisine, and price range. Users can submit preferences and also search for a specific ingredient, and will receive a dish recommendation as well as a restaurant recommendation for their preferences.
* Link: <http://35.161.174.85/>
* Schema

**1. Working Split**

All of the members of the team contributed to all the concepts behind this project.

Fandi designed the schema and coded the SQL to create the various tables in the database.

Payal, Rebecca and Fandi pulled the data and menu items from the restaurant websites to input to the database.

Rebecca set up the VM and server that our website is hosted on.

Payal and Fandi drafted the algorithm to use for restaurant recommendations.

Jessica put the data in the database.

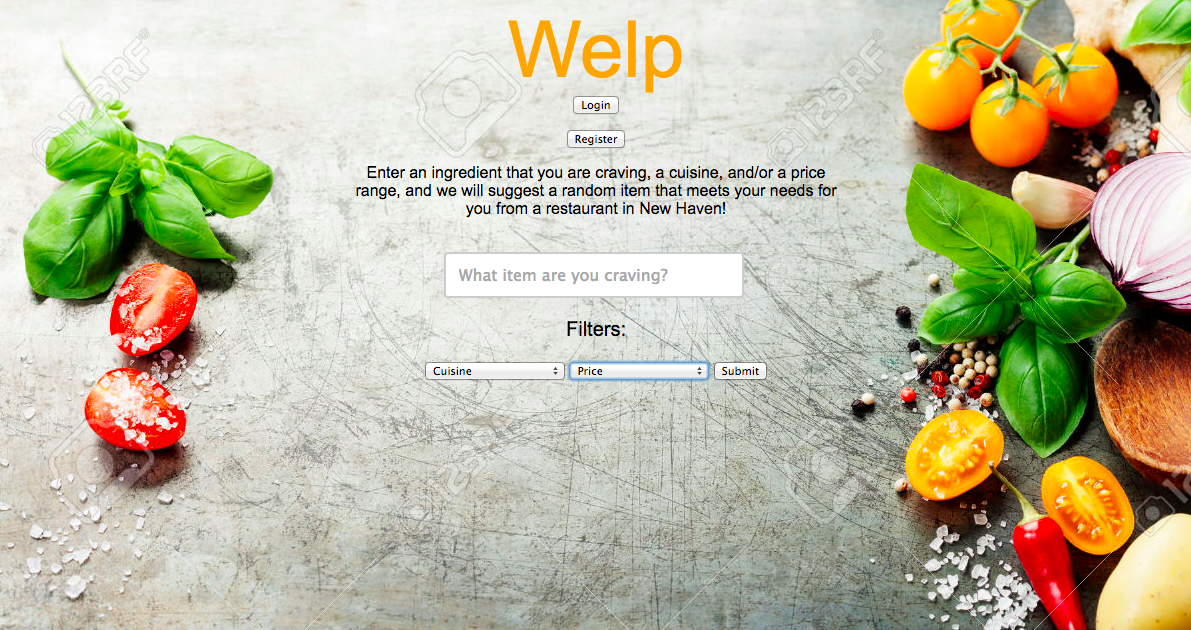
Jessica and Payal did the UI/Web design.

Rebecca coded user registration, login/logout, and following system.

**2. Functions the system supports**

Our system supports:

* Query to find restaurant based on ingredient, cuisine or price
  + If ingredient entered, a dish containing that ingredient will be returned as well
* Priority system (first matches ingredient, then cuisine, then price range)
* Create account, login and logout of account
* Comments -- displays comments and ratings on restaurants
* Follower network
  + Any user can follow any other user to see their comments and who they follow
* Algorithm for recommendation
  + Randomness emphasized for spontaneity if multiple results match query

Example: Login or register:

OR:

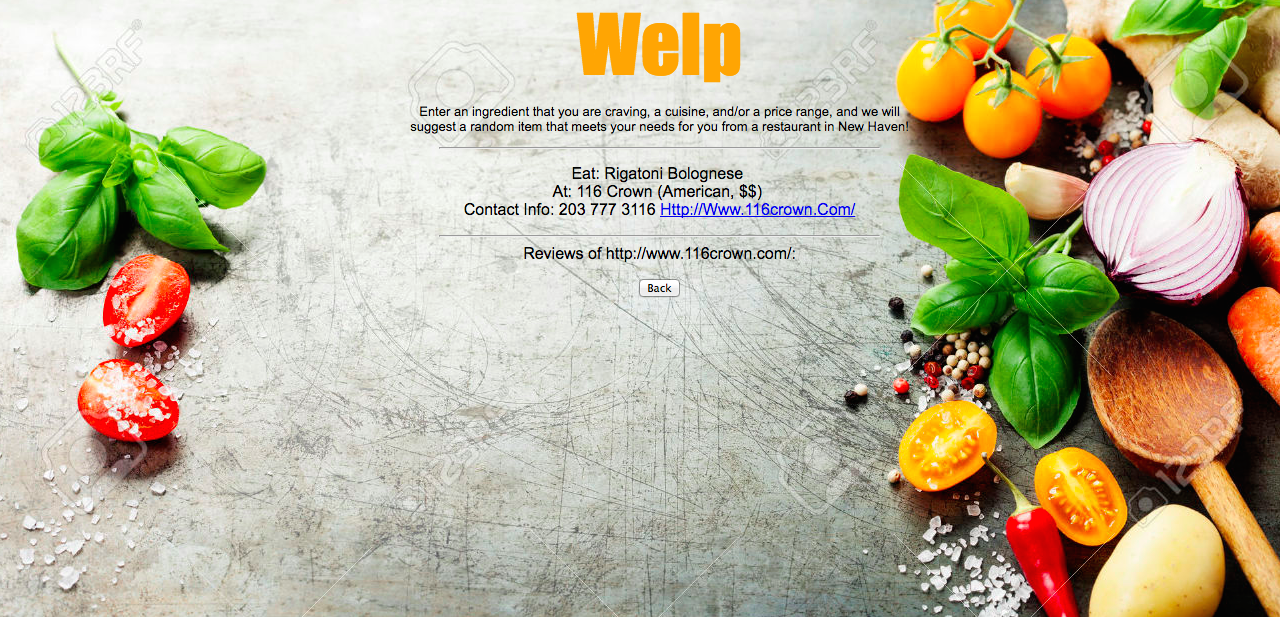
I wish to eat pasta, but dislike Italian restaurants. Thus, I search for pasta at American restaurants:



Result: Eat Lamb Ragu at Heirloom



If I want another option, I simply return and query again, to receive the suggestion of Rigatoni Bolognese at 116 Crown.



Technology used:

Amazon EC2 to host our site

MySQL for the SQL programming

HTML, CSS, for front end

PHP with embedded SQL for queries

**3. Limitations**

The system is not completely robust in that it would not be able to suggest similar menu items or return more generalized results; for example if someone searched “Langostino”, our system will not be able to translate that and return a Lobster dish (which might be fairly similar).

Similarly, we cannot check for allergies robustly because sometimes restaurants do not list all ingredients in the item description (e.g. someone allergic to peanuts would not be able to see if peanut oil were used in a dish).

We also do not have passwords for logging in for the sake of simplicity - users only enter a username to sign in. If we were to deploy the website, we would support account passwords and more secure logins.

We were unable to access APIs to scrape nationwide menu data, so we had to limit our menu data to only New Haven restaurants.

**4. Main issues and problems**

The main issue we encountered was the lack of API we could use to scrape the data. Locu was our first option, as it boasted APIs for all restaurants and listed all items as well as ingredients. However, the website was down for over a month, and other APIs we considered (such as OpenMenu) did not give us access to the ingredients, only the item names.

Another problem we encountered was in setting up the server. Originally we thought to use a Google cloud based server, but realized we would be charged for that. We ultimately used Amazon VM (a default configuration) to host our site.

**5. Main contributions our system makes**

Our system stores restaurant data (cuisine, price, link, phone number) as well as menu items and a list of ingredients for each item. Each restaurant also stores comments and ratings that users can leave, which display when a user receives a result after a query. Furthermore, our system allows a follower system in which any user can follow any other user to see their comments/reviews for restaurants.

Our system is New Haven targeted, so it is great for those at Yale. The main contribution, however, is its emphasis on menu items and specific ingredients a user might wish to eat. Instead of showing the most popular restaurants or highly rated, we display one food recommendation at a time and the restaurant it can be found at.

**6. Selling points**

No other system allows you to filter based on item you want to eat, or suggests a dish at that restaurant with that item, so our suggestion algorithm is innovative in that regard. Our system is thus targeted towards helping make choices through one suggestion rather than overwhelming the user.

Additionally, beyond the database aspect of our system, we have a follower system that allows users to sign up for accounts and follow other users. This is makes it very relevant in the age of social networks.