Health Tracker - Report

Contributors : Miti Jhaveri(maj529), Parita Dhanda (ppd241), Priyanshi Shah(ps3296), Sneha Ghosh(sg3533)

Github Link

Overview

Healthy and conscious eating has been followed by a lot of people these days. People are becoming more calorie conscious as our lifestyles are getting more sedentary with the growing technology. However, finding the correct recipe in such case is a much of hassle. Moreover not having all the ingredients for the recipe decided, is another big issue. Health Tracker App is a web-based application that comes to your rescue in such a scenario.

Solution

HealthApp is a web-based application that suggests you recipes based on the calorie requirement and keyword specified by the user. The keyword can be an ingredient name like eggs or any adjective related to the recipes; example : green. We have provided a filter based on the calorie. The user can input a calorie range and the recipes suggested will fall in that range.

The required calorie per day is estimated by the BMR of the user which is calculated on multiple factors including the age, height, weight, lifestyle(sedentary,light exercise, heavy exercise etc) of the user. These information bits are taken as input from the user at registration time and are stored in the profile of the user.

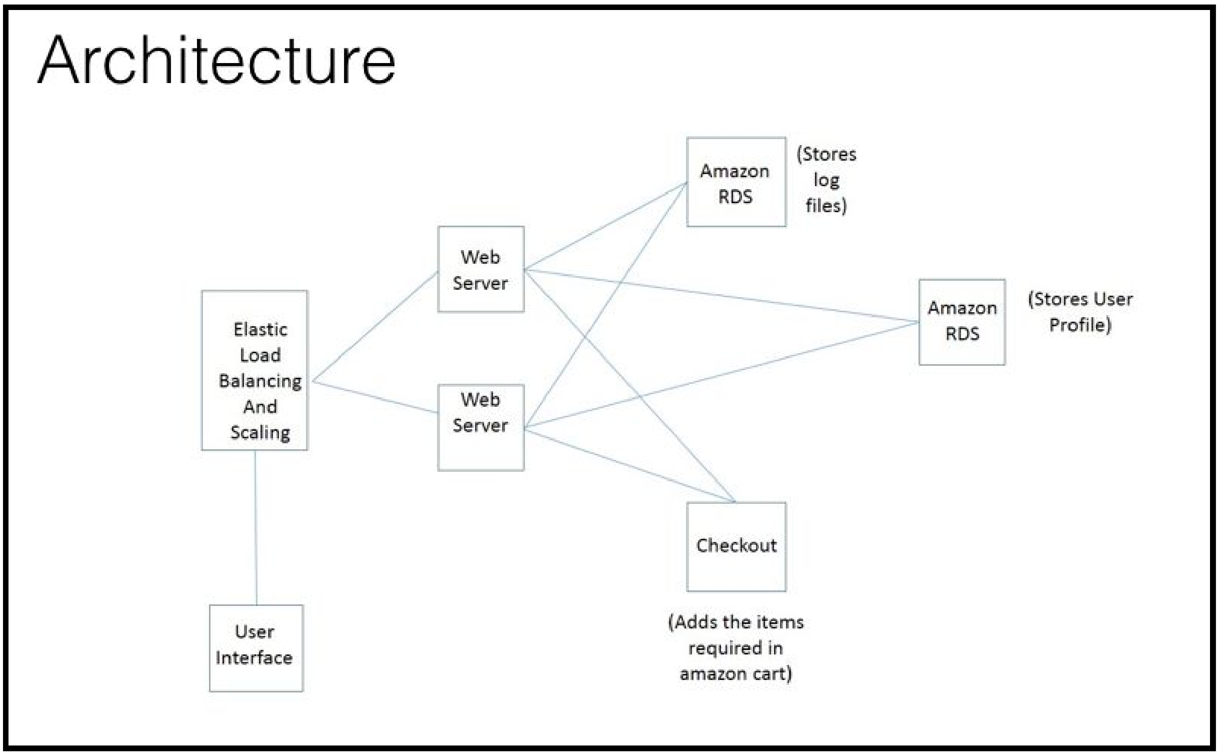
Upon finding an interesting recipe, the user can mark it for breakfast, lunch or dinner. This helps HealthApp in keeping track of the calorie intake for each meal and monitors your intake. There is a possibility that the user might eat a midday meal like a snack. As these calories are also important aspects to be considered during monitoring the calories, the application provides an option add any additional calorie intake.

The Recipes are from edamam API which uses a collection of recipes from multiple sources and gives the details of the recipe as response including the calories, ingredients and health labels (like low sodium, low carb etc).

The application also has a recommendation module which recommends recipes to the user based on user past eaten recipes. The application uses a similarity function which finds the match between the recipes eaten by all users. When the similarity between two recipes is more than 30%, and the user has not already used that particular recipe, it is suggested to the user. For example if a user xyz has eaten egg omelet and another user pqr has eaten egg toast, then the two users might have similar interests. Hence egg toast is suggested to the user xyz.

To alleviate the issue of missing ingredients in the pantry, upon selection of the recipe for a meal, the user is given option to place order through Amazon. This feature adds all the ingredients in the amazon cart and allows you to place an order of all the items required ingredients. So forget about any missing ingredients for tomorrow's lunch. Just click through, select, place order and cook.

Architecture Design



Component Description

User Interface

User can access the user interface of our website using the elasticbeanstalk url. The interface allows user to register on our website, search for recipes, view his daily calorie intake, view recommended recipes based on recipe matching algorithm that matches his recipes with other registered users, detailed view of recipe, add the recipe ingredients on Amazon.com cart.
We will later discuss different features of the website.

Elastic Load Balancing and Scaling

We have deployed our application on elasticbeanstalk which makes it scalable and switches load on webserver nodes when needed.

WebServer

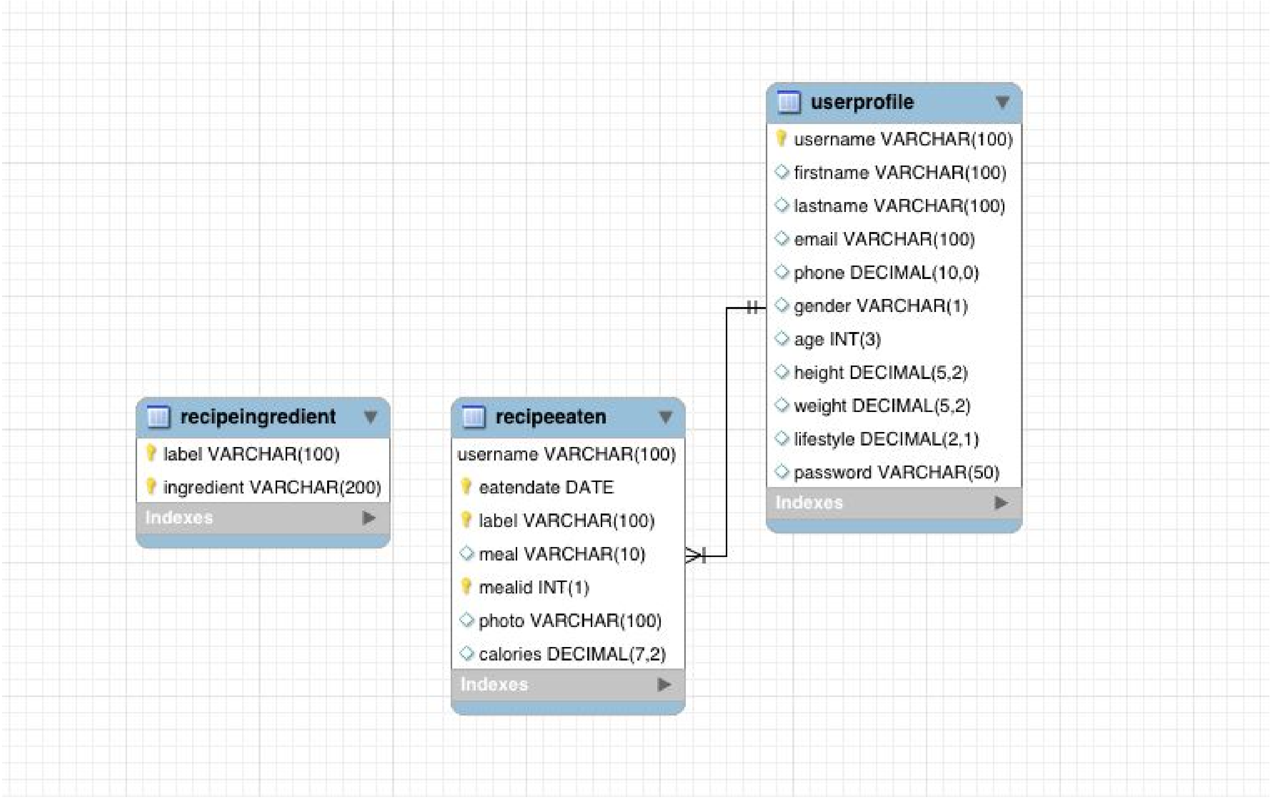
These are Amazon EC2 instances where code is running.

Amazon RDS (Database design)

We use the Amazon Relational database schema for the following data:

- **User Profile:** We store the profiles of all registered users. We use this to compute the BMR of users.
- **Recipes Eaten:** Users are allowed to mark a recipe as consumed. We store this in the database to allow us to recommend recipes. We will talk later in detail about our recommendation algorithm.
- **Recipe Ingredients:** We store the ingredients for all the recipes which have been marked as consumed by some user. This is also used to find the similarity between two recipes, which is useful during recommendation.

Following is the relational schema representation:



Checkout

The webserver interacts with Amazon API to search for ingredients and create Amazon cart. We will talk in detail about the Cart creation flow in the Implementation section below.

Implementation

Recommendation algorithm

As mentioned before, we store the recipes eaten by a user and the ingredients of all those recipes in the database. We use this data to compute the recommended recipes for a user.

Similarity of two recipes

To find the similarity between two recipes, we compute the percentage intersection of their sets of ingredients. Following is an example.
Consider R1 and R2 are the set of ingredients of recipe 1 and recipe 2 respectively.

```
Similarity (R1, R2) {
   N1 = count(R1)
   N2 = count(R2)
   M = count(set_intersection (R1, R2))
   return max(M*100/N1, M*100/N2)
}
```

SimilarityThreshold

We consider two recipes to be similar only if the Similarity >= SimilarityThreshold.
This allows us to prevent false positives by having a high similarity threshold. Eg. with a very low similarity threshold of, say 10, Vegetable Biryani and Dosa might be considered similar (as both contain rice).
With a higher similarity threshold like 60, that would not be the case, as they don’t have 60% ingredients in common. However, Vegetable Biryani and Vegetable Fried Rice might be considered similar with a threshold of 60% and that would be a good recommendation.

Candidate Recipes

Consider we are computing the recommended recipes for User U.
R = Set of recipes eaten by U.
T = Set of recipes eaten by some user other than U and **not** eaten by U.
Ing(X) = Set of ingredients of recipe X.

We compute the candidate recipes and their score as follows:

```
CandidateRecipes(U, R, T, SimilarityThreshold) {
   score = {}
   for t in T:
      recipeScore = 0
      for r in R:
         recipeScore = max(score[t], Similarity(Ing(R), Ing(T)));
```

```
    if (recipeScore >= SimilarityTreshold) {
      score[t] = recipe_score;
    }
  }
  return score
}
```

Recommended Recipes

Recommended recipes are just the list of candidate recipes sorted by their score. We also show which users ate the recommended recipes so that one can explore other recipes of that user.

Cart creation flow

Once the user reaches the recipe detail page (via either searching for a recipe or through the list of recommended recipes), we allow the user to place an order of the recipe ingredients using the Amazon and Mashape API. We use the following APIs to achieve this.

Edamam Recipe Search and Diet API (on Mashape)

We use this API to find the ingredients of a recipe. Sometimes the ingredients returned from the API were plural which produced undesirable results when searching on Amazon. Hence, we also made the ingredients singular.

ItemSearch API (on Amazon)

Once we find the ingredients of the recipe using the above Edamam API, we use the ItemSearch API on Amazon to search each ingredient. We only search within “Grocery” category and filter out the “out of stock” items.

To compute the most relevant result among the returned links, we compute which item in the list has most of the words in our ingredient (as generally the ingredients are multi-word strings). To resolve conflicts, we use the ranking in the list returned by Amazon.

CartCreate API (on Amazon)

Once we find the most relevant items from Amazon corresponding to each ingredient using the above method, we add then to Amazon’s cart using CartCreate API. The response of the API call contains a purchase URL which is then shown to the user, where one can confirm the items before they are added to the cart in amazon.com.

In case, even after searching for each ingredient, we are not able to find any item from Amazon’s ItemSearch API, then we prompt the user accordingly instead of calling the CartCreate API with 0 items.

Calorie Calculator

The Calorie Calculator is used to find the appropriate number of Calories required by the user on daily basis. The profile of the user has all the required information for the Calorie calculator; like the height, weight, age, lifestyle, gender.

BMR Calculation

First the BMR of the user in computed and this BMR is later used to calculate the Daily calorie requirement.
The formulas used for the BMR (Harris Benedict equation) are as follows:

MALE:
bmr = 655 + (9.6 * weight in kg) + (1.8 * height in cm) - (4.7 * age);

FEMALE:
bmr = 655 + (9.6 * weight in kg) + (1.8 * height in cm) - (4.7 * age);

LifeStyle Factor Computation

We compute the **lifestyle factor** based on lifestyle type as follows:

1. Sitting/Lying all day = 1.2
2. Seated work, no exercise = 1.3
3. Seated work, light exercise = 1.4
4. Moderately physical work, no exercise = 1.5
5. Moderately physical work, light exercise = 1.6
6. Moderately physical work, heavy exercise = 1.7
7. Heavy physical work,heavy exercise = 1.8
8. Above average physical work/exercise = 2.0

Daily Calories Required

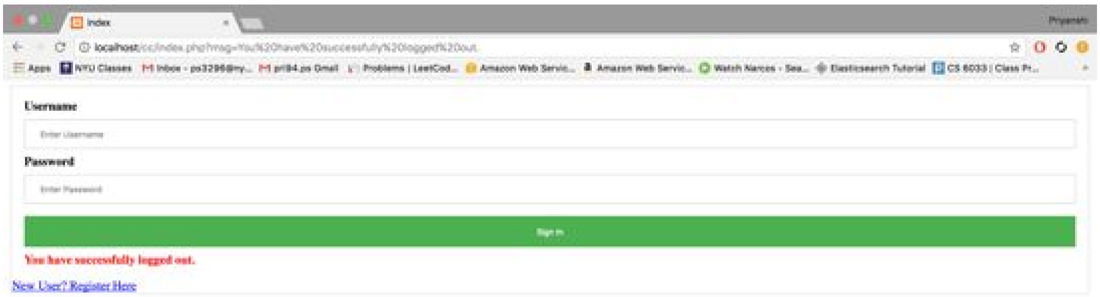
BMR multiplied by the lifestyle factor gives the calorie required per day.
Calorie required per day = bmr * lifestyle factor

User Interface and Flow

Login and Registration

The registration form includes Username, Firstname, password which is hashed and stored, gender, age, height, weight, lifestyle, email and phone number. This email is user for adding items in amazon cart.
Upon registration the user has to login using the username and password.

Login Page

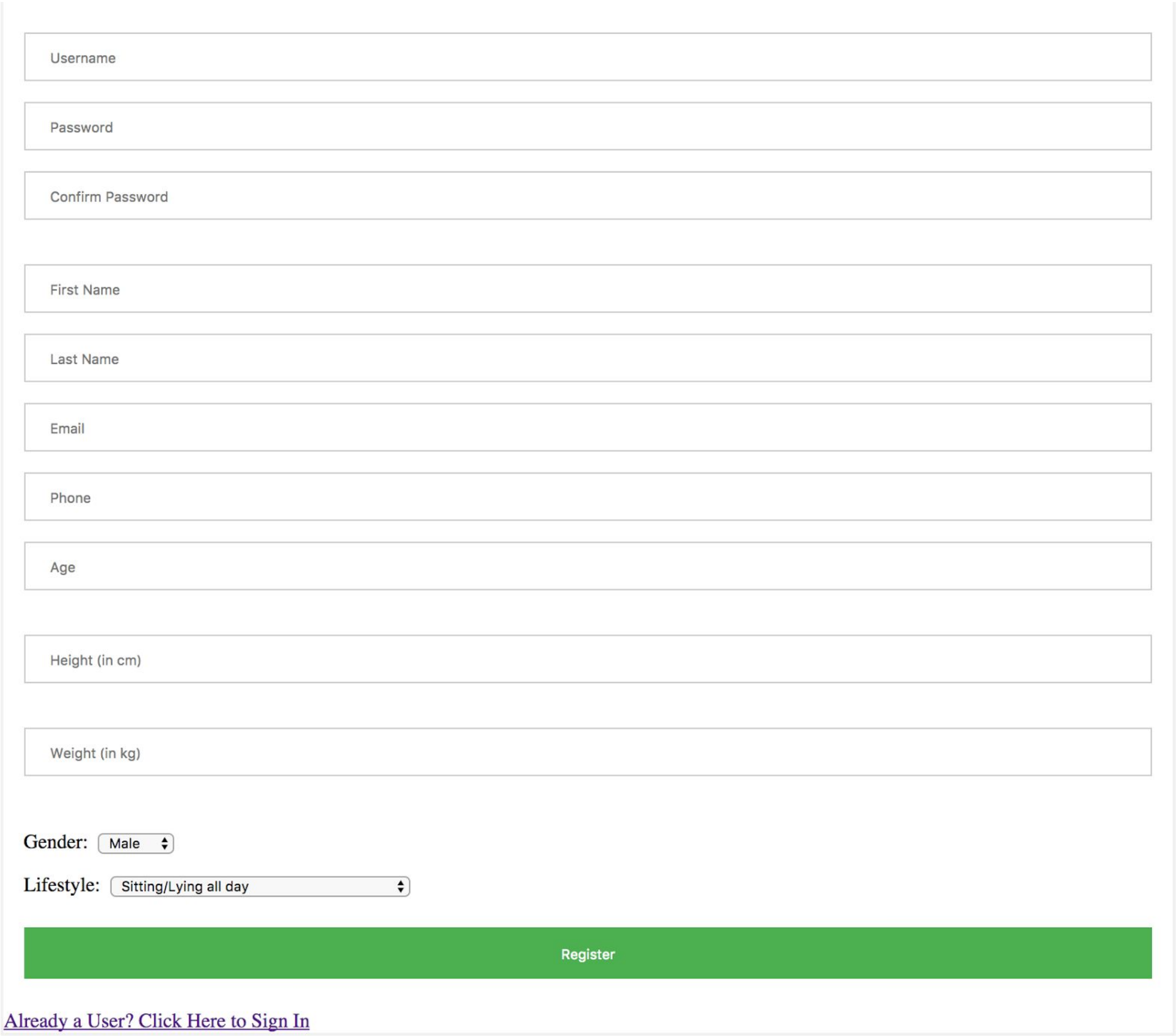


localhost/cc/index.php?msg=You%20have%20successfully%20logged%20out.

You have successfully logged out.

New User? Register Here

Registration Page



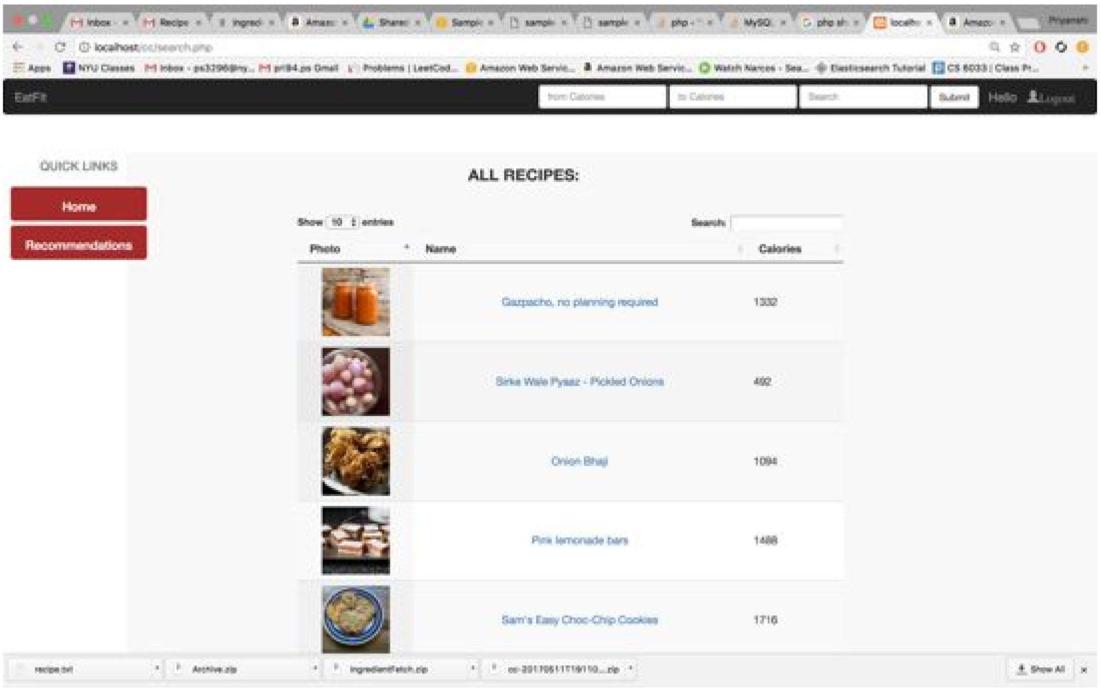
Gender: Male

Lifestyle: Sitting/Lying all day

Already a User? Click Here to Sign In

Search flow

User can use the searchbar provided in the navigation bar to find recipes based on the parameters specified. The searching parameters include any keyword which can be an ingredient name or any adjective like green, low sodium etc.
The search can also be on the basis of calorie range where the user specifies from and to calorie range.
Any possible combinations of lower limit, higher limit of calorie and keyword is possible. If the field is blank it is considered as all possibilities for that field, For example calorie range from 200 to 1500 and keyword noodles.



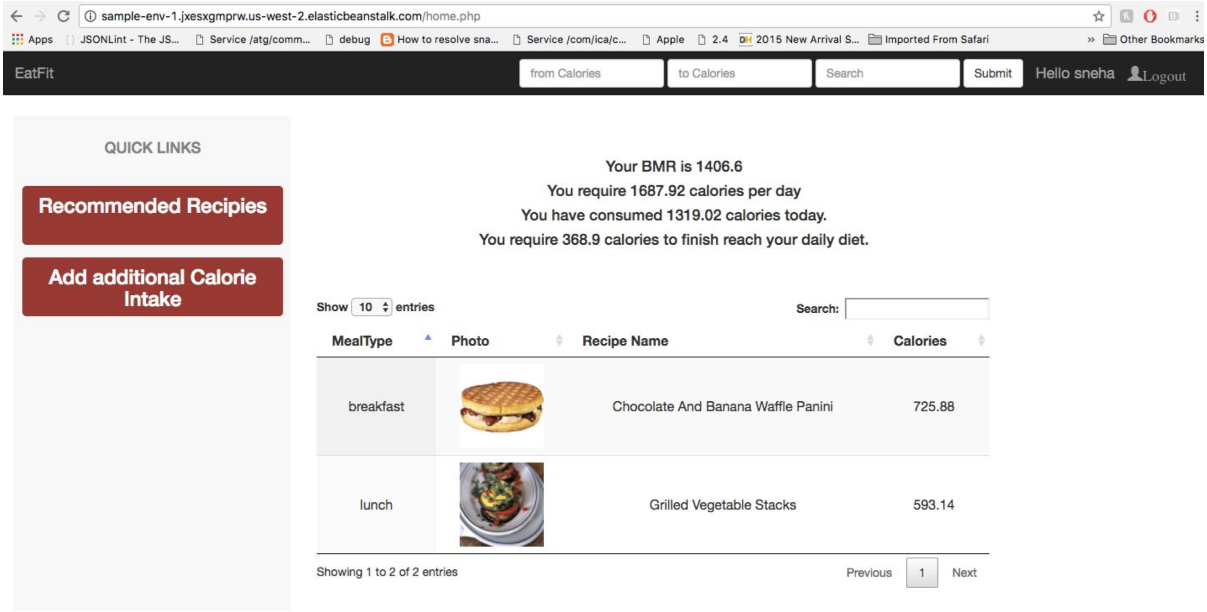
ALL RECIPES:

Photo	Name	Calories
	Gazpacho, no planning required	1332
	Sirka Wale Pyaaz - Pickled Onions	492
	Onion Bhaji	1094
	Pink lemonade bars	1488
	Sam's Easy Choc-Chip Cookies	1716

Today’s activity

When the user finds a recipe which interests him, he can mark the recipe as chosen by clicking on one of the three following
Mark for breakfast
Mark for lunch
Mark for dinner.
Upon selecting one of the above options the recipe is marked as eaten for that particular meal and it appears in the list of items consumed on the home page.

Home page shows today’s activity.



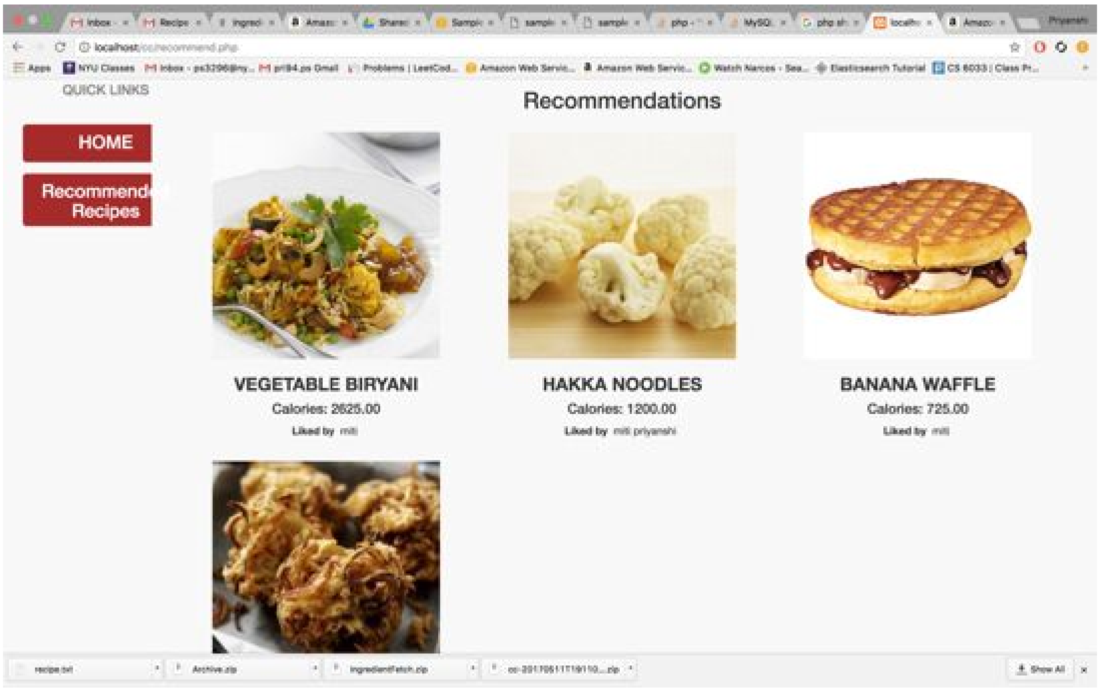
Your BMR is 1406.6
You require 1687.92 calories per day
You have consumed 1319.02 calories today.
You require 368.9 calories to finish reach your daily diet.

Show 10 entries Search:

MealType	Photo	Recipe Name	Calories
breakfast		Chocolate And Banana Waffle Panini	725.88
lunch		Grilled Vegetable Stacks	593.14

Showing 1 to 2 of 2 entries

Recommended recipes



Recommendations

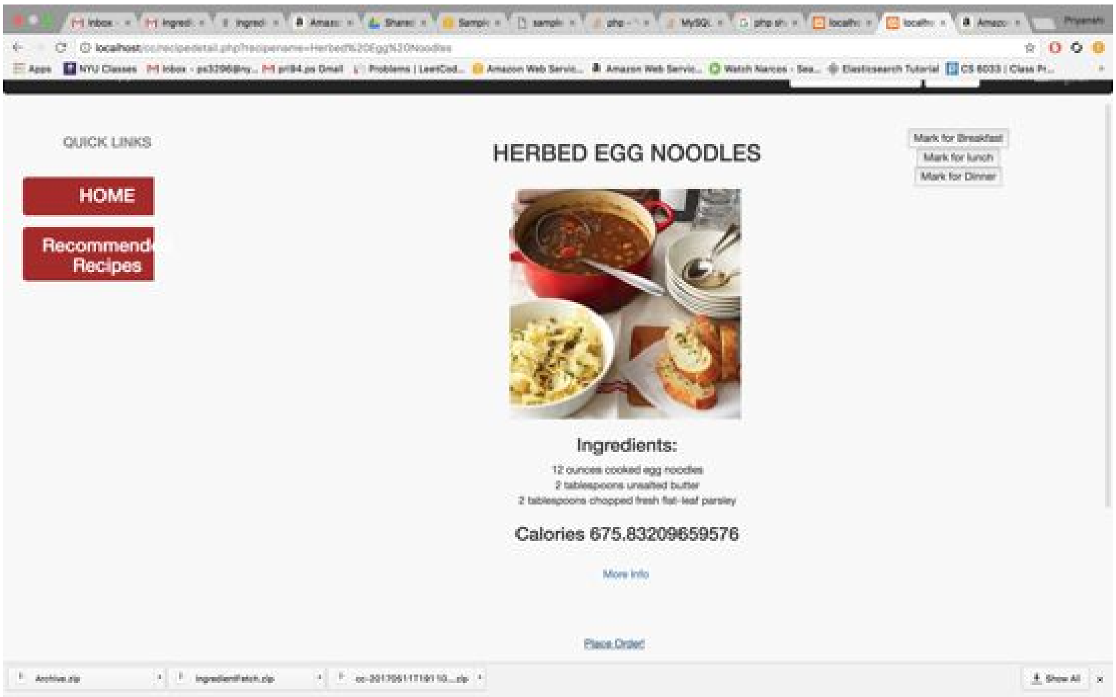
VEGETABLE BIRYANI
Calories: 2625.00
Liked by miti

HAKKA NOODLES
Calories: 1200.00
Liked by miti priyanshi

BANANA WAFFLE
Calories: 725.00
Liked by miti

Recipe Detail and Cart Creation

Upon clicking on a recipe search page or recommendation page, user gets to recipe detail page.



QUICK LINKS

HOME

Recommend Recipes

HERBED EGG NOODLES

Mark for Breakfast

Mark for lunch

Mark for Dinner

Ingredients:

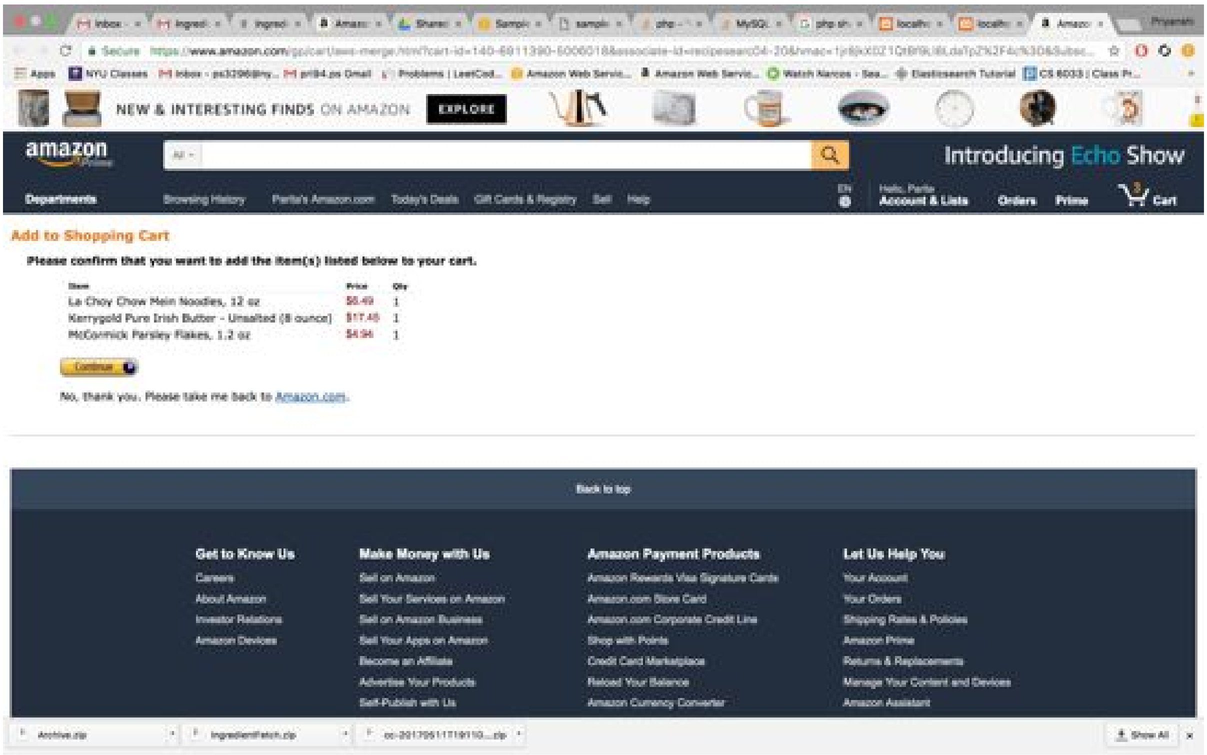
12 ounces cooked egg noodles
2 tablespoons unsalted butter
2 tablespoons chopped fresh flat-leaf parsley

Calories 675.83209659576

More Info

Place Order!

User can add the ingredients of a recipe to Amazon cart upon clicking Place order on above page:



Add to Shopping Cart

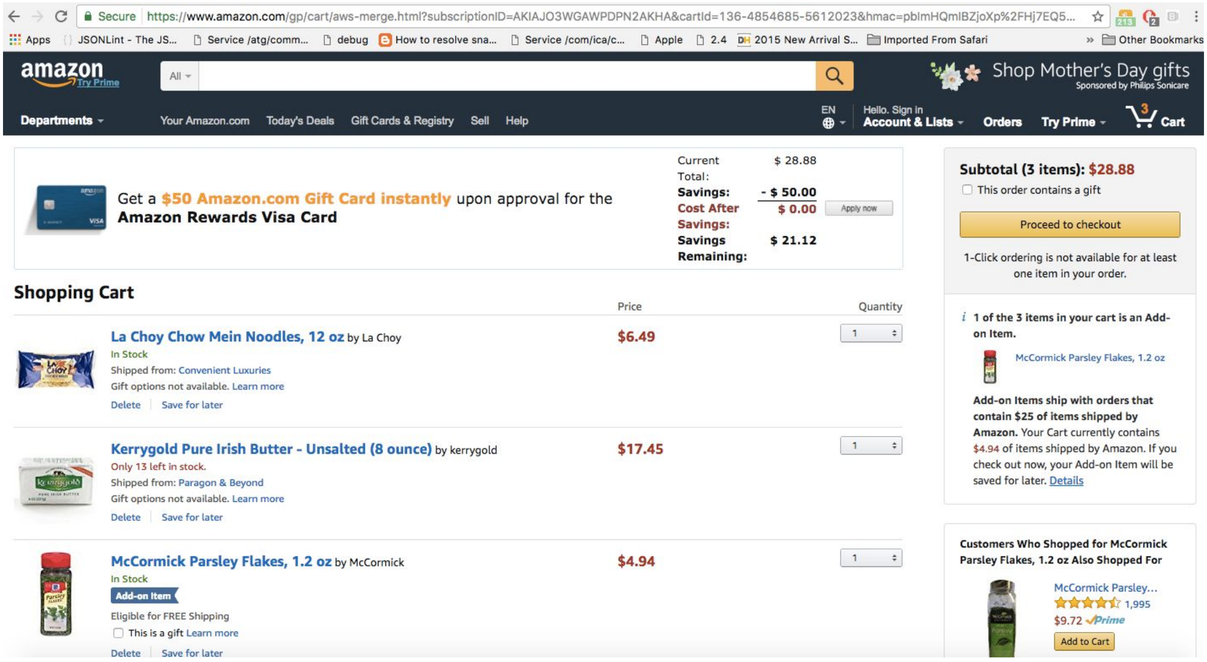
Please confirm that you want to add the item(s) listed below to your cart.

Item	Price	Qty
La Choy Chow Mein Noodles, 12 oz	$0.49	1
Kerrygold Pure Irish Butter - Unsalted (8 ounce)	$17.48	1
McCormick Parsley Flakes, 1.2 oz	$4.96	1

Continue

No, thank you. Please take me back to Amazon.com.

When user clicks on confirm the items are added to his cart.



Get a $50 Amazon.com Gift Card instantly upon approval for the **Amazon Rewards Visa Card**

Current Total:	$ 28.88
Savings:	**- $ 50.00**
Cost After Savings:	**$ 0.00**
Savings Remaining:	**$ 21.12**

Subtotal (3 items): $28.88

□ This order contains a gift

1-Click ordering is not available for at least one item in your order.

Shopping Cart

	Price	Quantity
La Choy Chow Mein Noodles, 12 oz by La Choy In Stock Shipped from: Convenient Luxuries Gift options not available. Learn more Delete | Save for later	$6.49	1
Kerrygold Pure Irish Butter - Unsalted (8 ounce) by kerrygold Only 13 left in stock. Shipped from: Paragon & Beyond Gift options not available. Learn more Delete | Save for later	$17.45	1
McCormick Parsley Flakes, 1.2 oz by McCormick In Stock Add-on Item Eligible for FREE Shipping □ This is a gift Learn more Delete | Save for later	$4.94	1

i **1 of the 3 items in your cart is an Add-on Item.**

McCormick Parsley Flakes, 1.2 oz

Add-on Items ship with orders that contain $25 of items shipped by Amazon. Your Cart currently contains $4.94 of items shipped by Amazon. If you check out now, your Add-on Item will be saved for later. Details

Customers Who Shopped for McCormick Parsley Flakes, 1.2 oz Also Shopped For

McCormick Parsley... 1,995 $9.72 Prime

Summary

Challenges

- IOS Mobile App implementation was hard to do as none of us knew mobile app development. We spent time learning IOS development but then migrated to do the web app which consumed some of our time. We should have thought realistically and not try to combine IOS app development learning with project development.
- It required lot of research to find Recipe APIs which were freely available and gave detailed view of recipe along with calories. We switched from Campbell Kitchen API to Edamam after doing the research as it gave calories for each recipe.
- For implementing recommendations we wanted to use recipes categories but Edamam did not provide any such categorization of recipes on the bases of cuisines so we were not able to filter recommendations on the basis of cuisines.
- We tried using ElasticSearch for storing recipes and created recipe indexed with the recipe content we needed but later on we realized we were just duplicating data from Edamam and it had to be merged and updated with the content from Edamam. So we used the Edamam API directly for searching recipes.
- Coordination with team members was also challenging and sometimes it happened that some of us were out of sync on the functionalities. So a lot of time was consumed in discussing and finalizing the design. Designing and implementing the modules in parallel wasted a lot of time for us. We could have designed first and then implemented sticking to that design.

Lessons Learned

Although we faced the challenges we learned a lot doing this project -

- For Amazon product advertising API we joined the Amazon affiliate program which allows developers to consume Amazon API for different purposes like item search, item lookup, cart creation, cart modify etc.
- BMR calculation - We implemented our own calorie calculator which gives calories that you can consume during a day using weight, height, lifestyle.
- Recommendation Algorithm - We recommended recipes by relating users on the basis of recipes they have eaten in the past. Users with similar taste belong to the same group and see recipes other like them have had in the past.
- REST API calls and research for APIs which are available. https://market.mashape.com website helped us to explore the APIs. Nutritionix, Campbell's Kitchen, Edamam are some APIs we looked at for recipes.

Github Link