**GroGro Project Proposal**

**ITWS 4200 Web Science**

**By: Team 1**

[**Sarah Abrahamson**](https://www.facebook.com/sarahxabrahamson)**, Robert Russo, Peter Ryder, Jessie Sodolo**

**Project Description:**

The project team intends to design a Grocery list website “GroGro”. This web application will allow a user to create a grocery list/(s), the application will then show the nutritional value of your their grocery list on a chart. All charts will display the Protein, Energy, Fat, Carbohydrate, fiber and total sugar per 100 grams for the list as a whole. The Protein, Energy, Fat, Carbohydrate, fiber and total sugar per 100 grams on individual ingredients will also be available to user.

**Why this needed?**

Anyone who has tried to lose weight can attest to the great difficulty involved in doing so. Habits are hard to break, gym visits are hard to schedule, and diets are hard to follow. To assist users with this difficult task we are proposing our application. By providing analytics on the grocery list, we alleviate a majority of the need for users to track individual meals. What’s more, we are encouraging users to eat at home, which saves both money and promotes healthy eating. The application is designed to encourage consumer responsibility and give fair, unbiased, and clear information about food being purchased and the total intake of a consumer. By displaying the information in a graph, we can also encourage a balanced diet through a simple visual metaphor.

**Use Cases/Basic Features:**

* User logs into the app, and creates a grocery list. The app then tells them the nutritional facts about their list.
* Using the same list a user wants information about a specific ingredient or food item. The user selects the ingredient and the app shows the user a nutrition list.
* User can search for food items to add to their grocery list. Autocomplete will be used to assist the user.
* Frequently used list items can be saved as ‘Favorites’ so the user can easily add them whenever they make a new list.

**Technologies Used:**

* MEAN Stack
* HTML5
* CSS3
* JavaScript
* JQuery
* Bootstrap 3
* Chart.js / D3.js / Google Charts
* USDA API
* Facebook Login API

**Why These Technologies?**

The MEAN stack fits our project because we will be interacting primarily with JSON data. The data returned from the USDA comes in the form of JSON data, and since Mongo stores its database as JSON files, it will be easy for us to integrate a MongoDB database. Node and Express will also be used for this project. JSON data is very easy to manipulate and use with JavaScript, and since Node uses JavaScript in the backend, moving and manipulating the JSON data will be easier. Express makes serving different web pages to the user easier when combined with Node. Bootstrap will make it easy for us to create a clean looking and responsive website for both mobile and traditional computers.

We chose to use the Facebook Login API to provide a simpler experience for our users. This also enables us to gather much more data about our users and opens the door to adding social/gamified aspects down the road.

Architecturally speaking, by implementing numerous other services for mundane and trivial aspects, like data collection and user validation, allows our team to focus on the core product. This also creates modularity in our program. If we prefer another data set, we could feasibly just grab that data instead. This modularity, coupled with our RESTful architecture, also enables us to expand our software onto various other platforms if we so choose.