

Final Project Proposal

Project

FeedMe – A Meal Planning App

Description

The FeedMe app takes the hassle out of selecting meals each week by randomly generating five meals, using Yummly's API, that abide by a users' specified preferences (ex: allergies, diet). The app produces a grocery list for each week's meal plan, which is editable by the user. Users can select meals from their meal plan to review detailed information (ex: total cook time, servings, source, ingredients). Lastly, users can interact with the device camera and photos in order to add pictures of their prepared meals.

Fundamental Functionality

At least one container view controller

- Tab Bar Container View Controller
- Navigation View Controller

At least 4 non-trivial scenes

1. Diet: User Selects Dietary Preferences
2. Meal Plan: Display 5 randomly generate meals (Based on selected dietary preferences)
3. Recipe: Display recipe details for selected recipe
4. Grocery List: List all ingredients for the meal plan
5. Photos: Display user generated meal photos

Interesting Functionality

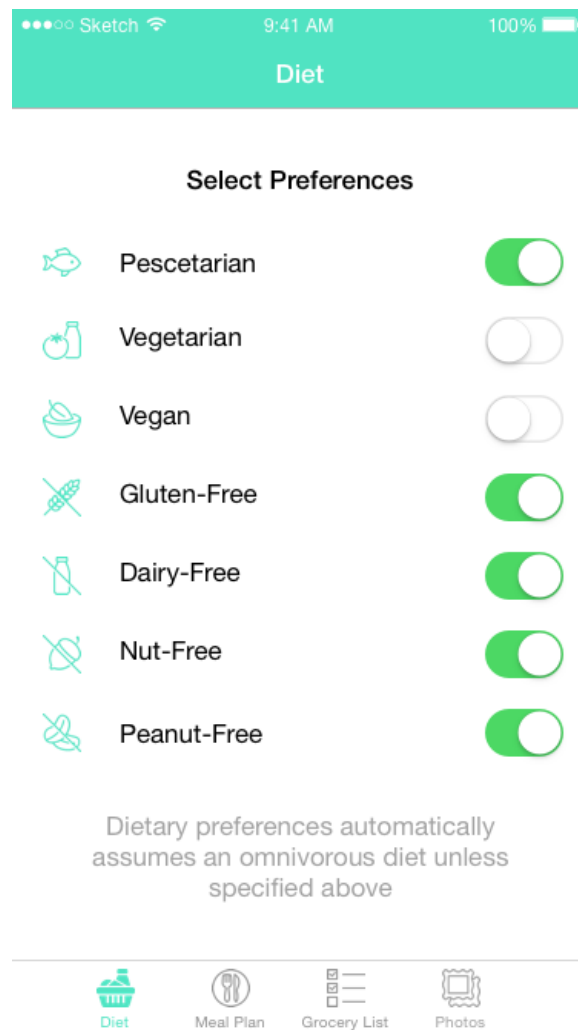
Integrate Yummly API to get recipes

Integrate Device Camera and Photos

Scene Details/Mockups

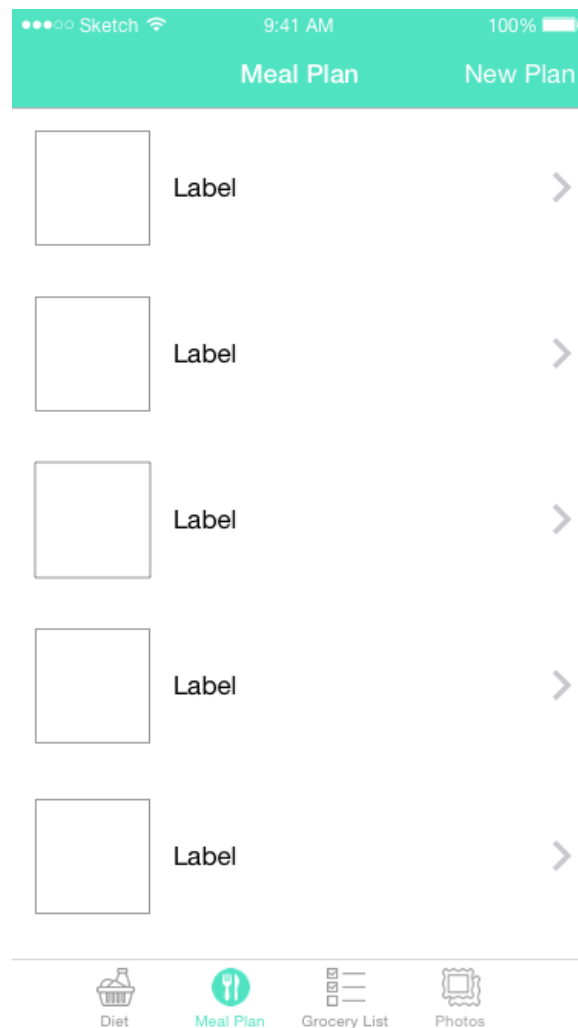
Scene 1 – Diet

- The user can navigate to Diet scene using the icon labeled 'Diet' in the Tab Bar
- The user can select the following dietary preferences using ON/OFF switch in order to dictate the types of recipes included in their meal plan
 - Vegetarian
 - Pescetarian
 - Vegan
 - Gluten-Free
 - Dairy-Free
- The dietary options will be displayed as a cell in a Table View
- Each cell in the Table View will include an icon (Image View), a dietary option (Label), and an ON/OFF Switch



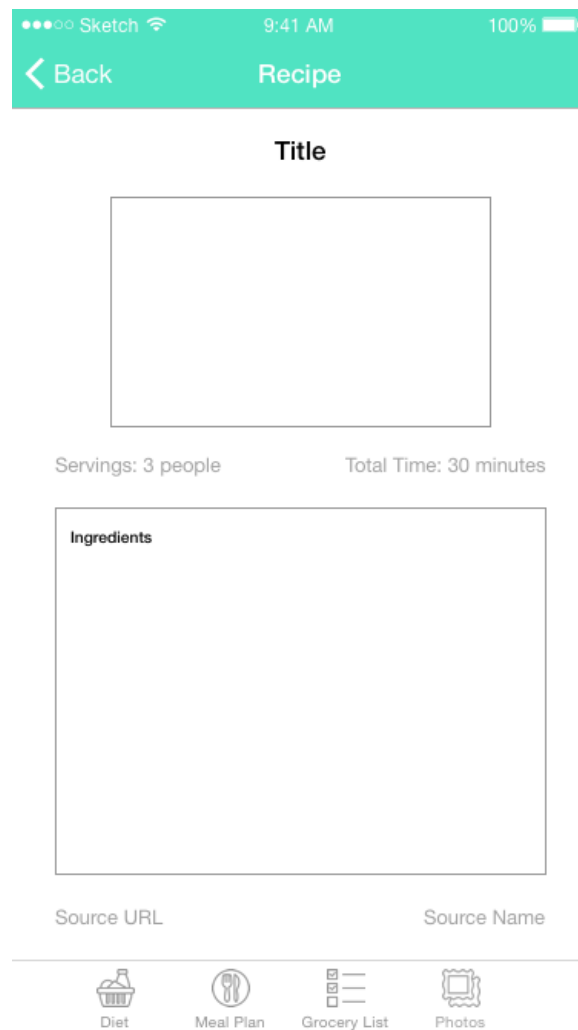
Scene 2 – Meal Plan

- The user can navigate to Meal Plan using the icon labeled 'Meal Plan' in the Tab Bar
- The view will display 5 randomly generate meals (based on selected dietary preferences) for each week
- The user can request a new meal plan by selecting the 'New Plan' button in the navigation bar
- The recipes will be displayed as a cell in a Table View
- Each recipe cell will include a recipe photo (Image View), a title (Label), and a disclosure indicator (navigates to the corresponding recipe)



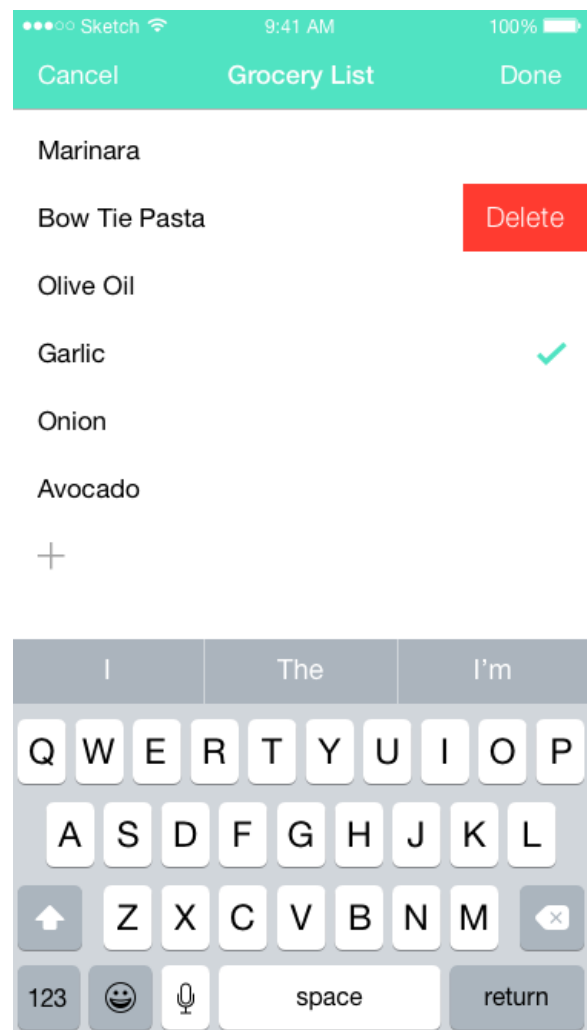
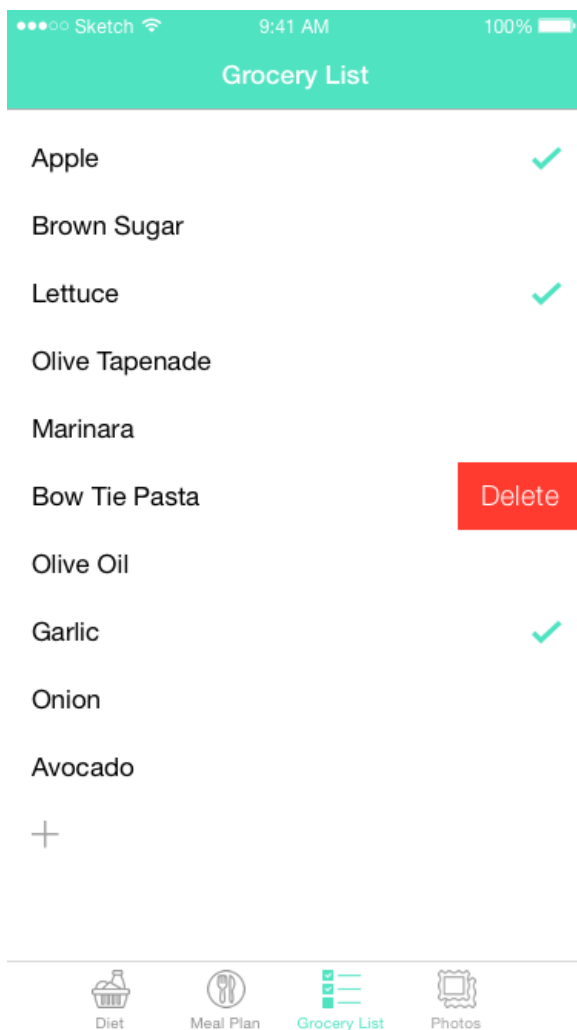
Scene 3 – Recipe

- The Recipe view will be a modally presented and embedded in a Navigation View Controller
- The view will display recipe details for the recipe (selected from the Meal Plan view) including:
 - Photo (Image View)
 - Title (Label)
 - Total Time (Label)
 - Number of Servings (Label)
 - Source Name (Label)
 - Source URL (Label)
 - List of Ingredients
- The user can select the 'back arrow/label' to return to the Meal Plan View



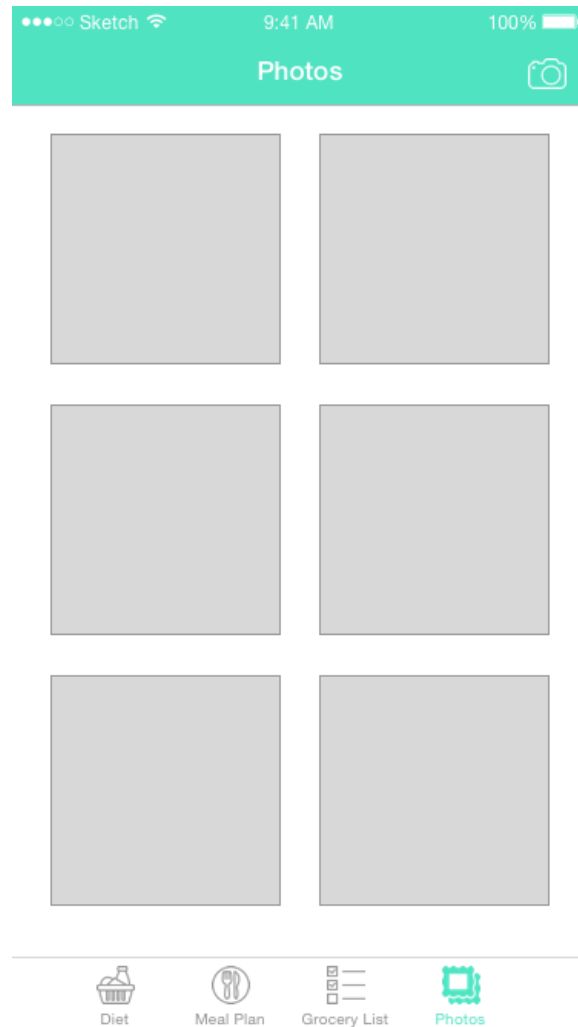
Scene 4 – Grocery List

- The user can navigate to Grocery List using the icon labeled 'Grocery List' in the Tab Bar
- The view will list all ingredients for the meal plan as cells in a Table View
- The user can 'check' each ingredient
- The user can delete each ingredient (swipe left for delete button)
- The user can add a custom ingredient by selecting the '+' button, which will notify the keyboard to appear
- The user can select the 'done' button in the navigation bar to save the custom ingredient, which will notify the keyboard to hide
- The user can select the 'cancel' button to proceed without saving the custom ingredient, which will notify the keyboard to hide



Scene 5 – Photos

- The user can navigate to Photos using the icon labeled 'Photos' in the Tab Bar
- The view will display all user meal-related photos generated by the user in a Collection View, where each photos is an Image View within a cell
- Users can add photos by selecting the camera icon button, located in the navigation bar, from their device's photos OR camera



Core Data Entities

Recipe

- Recipe ID
- Name
- Image
- Total Time (Prep + Cook)
- Number of Servings
- Source Name
- Source URL
- To-Many relationship with Ingredient

Ingredient

- Name
- To-One relationship with Recipe

Photo

- Photo ID

