Team 60 (*CMU Food*)

Consensus Statement

**Team Members: Luke Jasko and Karthik.Murugan**

**Features Implemented by Luke:**

1. All front end aspects of web application
2. CSS Styling
3. Database modeling (constructed basic ERD for reference; created/configured models.py)
4. Permission delegation to *Student*, *RestaurantEmployee*, and *Administrator* users (used built-in Django Permission model)
5. Administrator can add new location (included use of Google Maps JavaScript API; included *get\_location\_marker\_json* and *get\_restaurants\_json* operations)
6. Administrator can activate any user
7. Administrator can deactivate any user
8. Administrator can add new restaurant to system
9. Administrator can invite new employee by providing person’s email address (subsequent email is sent for validation of registration)
10. Administrator can add new administrator (partially completed)
11. Administrator can view all registered users in system (including view of just students and just employees)
12. Student registration
13. Restaurant employee registration
14. Students can view all active restaurants in the system (included use of Google Maps JavaScript API; included *get\_location\_marker\_json* and *get\_restaurants\_json* operations)
15. Students can view an active restaurant’s details
16. Creation/maintenance of all fixtures (updating JSON fixtures as necessary; natural ordering of fixtures for improved migration to heroku)
17. Configuration/maintenance/troubleshooting of heroku deployment
18. Refactoring navigation snippets (used include statements to minimize some redundant navigation HTML code)
19. Refactoring form template snippets (used include statements to minimize some redundant forms HTML code)

**Features Implemented by Karthik:**

1. Search by Restaurant Name Functionality
2. Reset Password
3. Forgot Password and associated email confirmation
4. Employee can add a food item to Restaurant (including picture for a food-item and other fields in the Food Item model)
5. Employee can edit a food item in a restaurant
6. Employee can delete a food item from a restaurant
7. Student can view Food-items corresponding to a restaurant
8. Student can view details of a specific food-item
9. Add Food-items to order-cart functionality (i.e. Student can select food-items to order)
10. Remove Food-items from order-cart (i.e. Student can remove food-items from an order)
11. Place order and associated email confirmation that order has been placed (Student can place an order)
12. Quick Order: Students can view a list of their previous orders so that they can place orders fast
13. Employee of the particular restaurant can Accept or Cancel and appropriate email confirmation will be sent to the user