CMPE 275 Section 2 Spring 2016

Term Project

Last updated: 4/14/2016 (status: draft)

In this project you build a takeout order and fulfillment management system for a restaurant. This must be a web app whose server you build and host in the cloud. The primary language you use for server implementation must be Java. You do not have to use Spring, but you need to exercise the principles, patterns, and methodologies that you learn in the class, such as DI, AOP, MVC, ORM, and transactions. You must use either a relational database, or Datastore if you choose AppEngine.

If any feature described below is unclear or ambiguous, and you fail to get a clear answer from Canvas's discussion board, you can use your best judgement to interpret and add the missing details, provided that you clearly document and explain your reasoning in the product report.

Project Topic - Takeout Order Management System

This app manages the ordering and fulfillment of takeout orders for a restaurant. The interface must be web based, and the server needs to be hosted on the cloud, and accessible from anywhere with Internet connection. The major features include:

Users and Authentication

There are two types of users, admin and customer.

1. Admin user accounts can be hardcoded, with predefined usernames and passwords of your own choice.
2. A customer is able to create his account using an email as the username and password of his choice. Your app must send an email to the user with a verification code. The user needs to use that verification code to complete his account registration.

Menu Provisioning

An admin must be able to add and remove menu items.

3. Each menu item contains the following properties
    a. Category
        i. Drink
        ii. Appetizer
        iii. Main course
        iv. Desert
    b. ID (integers between [0-999], inclusive)
    c. Name
    d. Picture (one per item)

    e. Unit price (e.g., $5.5)
    f. Calories (integers only, can be zero)
    g. Preparation time(1, 2, …, or 10 minutes. This is NEVER shown to the customers.)
4. Add a menu item
    a. An admin must specify all the properties given above for a menu item, except that the ID needs to be generated automatically by your app (or your database or something else, depending on how you implement it)
    b. The added item needs to be instantly available for ordering
5. Remove a menu item
    a. An admin can remove any menu item from the ordering system, in the sense that the item will no longer be available for ordering.
    b. Any existing order for the removed item, however, still need to fulfilled.
    c. It is up to you whether you want to have the capability of re-enabling an item that is removed so that it comes orderable again

Ordering Food

A customer must be able to make takeout order through your web app

6. Menu items needs to be ordered by Categories
7. A customer must be able to navigate and see most of the properties of an item (except the preparation time)
8. When ordering, the quantity of each item needs to be default to one, but the customer must be able to specify a bigger number (up to 100) at his choice
9. Pickup time
    a. You must show the earliest time the food is available for pickup
    b. The customer is able to optionally specify a pickup time, anytime within the office owner of the next 30 days from now, including today
    c. If the pickup time is not possible due to reasons like fulfillment capacity, you must alert the customer and not allow him to submit the order. The customer must either revise the order or cancel it
    d. Your restaurant runs between 6am-9pm, 24 for 7
    e. Food can only be picked up during office hours, of course.

Ordering Fulfillment

For simplicity, you restaurant has only three food pipelines, i.e., at any given time during office hours, there are *only three* persons preparing food

10. For simplicity, each order can only be prepared in a single pipeline, i.e., the only one person can work on an order, and must do so continuously. No single order can go beyond the end of each day's office hours.
11. The pickup time cannot be more than an hour after it is prepared; i.e., you cannot start fulfilling an order too early

i. E.g., if the order takes 10 minutes time to prepare, and the customer sets the pickup time as 11am tomorrow, then you can only start preparing the order between 9:50am and 10:50am tomorrow.

System Defaults and Order Reset

12. When the app starts, the are no existing orders, and there are three menu items only, with 1, 2, and 3 minutes of preparation time respectively.
13. An admin should be able to reset the ordering and fulfillment by clear all the orders and pipeline status, so that there is no order placed in the system at all. Order menus, however, are not affected.

Grouping

This project is group based, with group size up to four people. Once the project plan is submitted, group membership **cannot** be changed.

Source Code Management

You are recommended to use a Source Control Management (SCM) system to manage your team's source code. This can be a private Bitbucket repository or your local git. Commit history will be checked for understanding the frequency of contribution to the project work.

Cheating Policy

Your app must be built by yourself, and cannot be based on the code base of any existing app. If you used any code not written by yourself, it must be clearly documented in your README.TXT file, unless it is part of publicly available libraries. *If your app is already used to serve the requirements of any other class, it will not be accepted by this class*. In the case **any** form of cheating is confirmed, your will get an **F** grade for this class.

Deliverables and Grading

The project is worth 25 points in total. The actual *due dates* of the deliverables will be specified in Canvas.

Project Plan (1 point)

The project plan needs to include

- The platform and technology choices (Spring/EJB, MySQL, etc)
- Architecture diagram
- Division of work with feature owner specified
- Milestones with expected dates

You are recommended not to exceed 3 pages. Must be submitted through Canvas as a PDF file.

Project Presentation and Demo (5 points)

To be presented in class.

- The presentation should cover introduction, high level design, and major features with screenshot. Time limit: 3 minutes.

- You must also do a live demo. The guideline for how to do demos is to be added. Time limit: 5 minutes.
- Grading will be based on successfulness of the demo, the content and clarity of the slides, and the delivery of the presentation

The presentation slides must be submitted through Canvas as a PDF file.

Project Report (4 points)

The report needs to cover the following topics.

1. Motivation and introduction of your app
2. High level and component level design
3. Technology choices
4. Description of features with final **screenshots**
5. Testing plan executed and results
6. Lessons learned and possible future work

You are recommended **not to** exceed 15 pages, but you will **not be penalized** just because the report is too long or too short, as long as the level of coverage for the required topics is reasonable and clear. The report must be submitted through Canvas as a PDF file.

Project App (15 points)

Note: the instruction for submission is still *subject to change*.

1. Your must submit all your source code / resource files through Canvas
2. Features richness, complexity, correctness, choice of technology and implementation are worth 12 points
3. User interface and user experience are worth 3 points
4. README.TXT, including
    a. The names, email IDs, and students IDs of the members
    b. The URL to access your app, and the admin account username/password, so that the user can test the provisioning features
    c. Build instructions;