

"Delectable" - A Catering Business

1. Introduction

The purpose of the project is to give you the opportunity to practice the concepts discussed in this class. The requirements are purposely somewhat vague such that you can interact with the client (that's your instructor) to figure out the detailed requirements.

Before we get into the [functionality detail](#), I want to remind you that detail about the programming language, the platform on which the project will be tested, unit-testing, and how to submit your work is available in the [syllabus](#).

Sections 2 through 4 in this document cover acceptance criteria for the project, Q&A, and a description of how you can earn extra credit for the project.

Lastly, the best project will receive a prize of \$300 in cash, curtesy of our sponsor, theSleeples, Inc. Your instructor reserves the right to not award the prize or to split it between multiple outstanding submissions.

2. Acceptance Criteria

The basic implementation of the project assumes the application runs on a server and that various clients access it through a REST interface.

Sample XML/JSON files used to test your functionality will be available here soon.

Scenario that will be used to test your application: coming here soon.

3. Questions and Answers

Check [here](#) for answers before asking your TA or the teacher.

4. Extra Credit

The main purpose of this project is to practice object-oriented analysis, design and programming, not to teach you web development frameworks or relational databases. As such, the use of a database is expressly prohibited for the core project implementation.

There is a finer point to be made about the databases ban, namely that most students would start their from with the design of a database schema, the way they learned in the Databases class, instead of thinking about use cases, entity objects, etc.

In fact, the database should be a plug-in to your application, the same way the delivery mechanism should be, whether command line, web, etc.

We can talk about extra credit iff (if and only if) your core project implementation works impeccably. This point will not be debated.

If you complete your core project implementation on time and feel adventurous, then you can get extra credit for implementing a web client for the customer.

5. Project Description

"Delectable" is a catering business for the new age. The food is great and we don't break our customers' pockets.

We started the business using pen, paper and phones. It's time that we get some software to help us run the business better and grow it.



Like any other catering business, we maintain a menu from which our customers can order. Each food on the menu has a price (per person) and the minimum number of persons that it can be ordered for.

Each food can belong to one of more categories, e.g. "organic", "gluten free", "vegetarian", "vegan", "dairy free".

Our customers should be able to browse the menu to see what's available to order. They can see all the foods available on the menu or select just those in a specific category, e.g. "vegan".

Our customers can create an order by selecting one or more foods from the menu.

NOTE: Please don't forget that for each food selected the customer needs to specify the number of people they're ordering for.

The system will then calculate the total price for the order. When the customer submits the order, they need to provide shipping and billing information, in addition to special instructions for the order (e.g. "delivery door is in the back", "we need red napkins", etc.)

NOTE: we apply a surcharge for all orders that are delivered on week-ends and federal holidays. The surcharge needs to be shown as a separate line item on the order.

Once the order is placed, the system will return to the customer two URLs, one that allows the customer to view their order and another one that allows the customer to cancel their order. Orders can be cancelled no later than the beginning of the day when they're supposed to be delivered.

When customers call with a question about their order, our customer service people should be able to look them up by email address, or their last name, or by their phone number. If found, then our customer service people should be able to see current and past orders, including cancelled orders.

Our business manager should be able to view all the orders that need to be delivered today such that she can have them prepared and ready to go.

Our business manager should be able to view all the orders that need to be delivered tomorrow as well. For each food that is being ordered for tomorrow, she should be able to see the number of people it's been ordered for.

Our business manager will change the price of menu items from time to time. She may also change the surcharge for week-end and holiday delivery.

The other thing that I need is some sense of how we're doing as a business. Here are some of the things I want to know:

- How much money have we made this month, last month, over the last three, six, nine, 12 months?
- How many orders have we delivered this month, last month, over the last three, six, nine, 12 months?

Here is a list of possible future enhancements:

- Customers can modify an order after it's been submitted and accepted by the system.
- Get the customers to sign-off on delivery.
- Allow customers to rate foods.
- Allow customers to rate the quality of service for each delivery

That's all for now ... is this super-complicated? Can I get this by mid-April?

Last update: Feb 2, 2016

[Virgil Bistriceanu](#)

[cs445](#)

[Computer Science](#)

\$Id: project.html,v 1.1 2016/02/02 21:07:21 virgil Exp \$