CS 364 Project 01

Fall 2016

**Stoner’s Late Night Cravings**

Derek Stoner

Chris Sterr

Matt Salvatore

Marcus Lowe

**Synopsis:**

Everyone has had late night hunger spells and food cravings that can't be satisfied by their near empty refrigerator. How great would it be to have all the late night food sources available to you in an instant? Whether you want something delivered or to find a place to get out and get some quality dine-in food this is your place to find what you need. Students staying up late studying or late shift workers won't have to spend time searching countless websites trying to find the food they crave from a nearby restaurant. We will store restaurant hours, menu items with user reviews of the food, whether or not delivery is available, and the location of the restaurants. We can also provide pricing of the items so those who are living on a tight budget can find good food that will not break the bank.

**Functionality:**

* Find address
* Find current time
* Filter restaurants by food type, open or closed, delivery, and distance.
* Get food items from restaurants
* Create user database
* Create reviews from users and link to restaurants

**Database:**

C:\Users\Matt\Downloads\CS 364 Project 01.png**Restaurant table:** Being the biggest table in the database, it will have information contained within it about specific locations, down to the local level as well as three methods to contact the restaurant. It will have an ID as a key, which will be utilized to create a genre self-referential relationship. Additional information about the restaurant will be whether or not it delivers as well as the hours associated with each.



**Items table:** Items table will hold menu items that are available to purchase from restaurants. Each food item will have a name along with a description of what each item is. There will also be pricing for the customers to compare the value of each item. This table will have a foreign key from the restaurant ID that serves it.

**User table:** The User table will have location information associated with it as well as a unique username. The entity allows for reviews to be created and submitted for information on personal experiences at individual restaurants.

**Genre Table:** The genre table will contain the general type of odd that a particular restaurant mainly serves along with a description of the genre. This table will have a foreign key from the restaurant ID associated with it. This will allow users to get a listing of places that serve the type of food they are craving.

**Review Table:** Containing information created by users, the review table has foreign keys to specify which user and restaurant they are associated with. The table will have the ability to hold specific comments about restaurants as well as an overall rating.

**Stakeholders:**

This application is intended for a wide variety of people including college students, late-shift workers and just about anybody that finds themselves hungry during the late night hours.

**Technological Requirements:**

We’ve all determined it would be best to stick with a desktop application for this project, as we all have a solid foundation in desktop java applications. We intend to use SQLite and JDBC to implement our database in a GUI-based java application with the possibility of developing a simple web interface.

**Create Table Statements: (Currently missing set-valued indications)**

CREATE TABLE Restaurant ( RestaurantID int PRIMARY KEY,

ClosingTime Time INT,

email TEXT,

Website NONE,

Phone INT,

Name TEXT,

Delivery TEXT,

Zip INT,

State TEXT,

City TEXT,

Street TEXT)

CREATE TABLE Items ( RestaurantID INT REFERENCES Restaurant.RestaurantID

RestaurantID PRIMARY KEY,

Name TEXT,

Price INT,

Descrption TEXT)

CREATE TABLE Genre ( RestaurantID INT REFERENCES Restaurant.RestaurantID,

Description TEXT,

Name TEXT)

CREATE TABLE User ( Username TEXT PRIMARY KEY,

Password TEXT NOT NULL,

Zip INT,

State TEXT,

City TEXT,

Street TEXT)

CREATE TABLE Review ( FOREIGN KEY RestaurantID INT REFERENCES Restaurant.RestaurantID,

FOREIGN KEY Username TEXT REFERENCES User.Username,

PRIMARY KEY ( RestaurantID, Username ),

Comments TEXT,

Rating INT)