Crave on campus (expanded)  
Project Scope

DSBA-6160 | 11/15/2021

# Overview

## Team Members

* Saurabh Vijay
* Shina Dhingra
* Sammie Srabani
* Ali Camlibel
* Lakshmi Medineni

## Project Background and Description

The ERD system consists of the 1st Level Dataflow Diagram and Second Level Dataflow Diagram. The 1st Level Dataflow Diagram indicates that the system is divided into sub-systems such as vehicle details, drivers’ details, people involved-Student/Staff, Location-faculties for students and staff, orders, and restaurant details/Ratings. Each unit deals with one or more data flow. All of them combined to give the functionality of the college/university restaurant management system.

In the 1st level Dataflow Diagram, the main entities and outputs include processing the user records and generating a report of user categories (Student, Staff, Drivers - ratings), processing all orders category records and producing a report of all orders, and processing locations of all orders made and reports of those locations (Different faculties). It also supports processing details of vehicles used in the delivery of orders in different locations (faculties), processing delivery details and generating a report on all deliveries made, processing orders made by the student and producing reports, and processing orders made by staff and producing reports

The 2nd level Data Flow Diagram gives deeper information of the college/university restaurant management system. It contains more details of the hotel staff, customers, means of payments, exact locations, order IDs, Driver IDs, Drivers employment details, Rating about the driver, restaurant and services and Details of student graduation year.

The 2nd level Dataflow Diagram functionalities where admin can log in and manage all the functionalities of the system. It also allow the admin to add, remove, and update student, staff, or driver details and provide the ratings to food delivery ratings. The other functionality is that it allows the admin to generate accurate reports of all restaurant sections-faculties, orders, staff, drivers, and location, delivery, or payment details.

## Project Scope

The main purpose of this college/university restaurant management system is to show the production of reports, inventories, staff/student/driver/location/faculty/vehicle records/ratings as well as managing orders. This system is only useful to users which are involved in restaurant interactions such as staff, students, and drivers as well as ratings.

## Use Cases

|  |  |  |
| --- | --- | --- |
| US-001 | Secure Food Delivery on Campus | |
| Dependencies | OBJ 001: To mange secured food delivery with in a campus (Objective) | |
| OBJ 002: Access to various local restaurants(Objective) | |
| OBJ 003: Restricted Drivers and accessible restaurants system (Business Rules) | |
| OBJ 004: Delivery food to correct person and location (Business Rule) | |
| OBJ 004: Delivery food to correct person and location (Business Rule) | |
| Description | The system will behave as described in the following user case when a person request for a food delivery | |
|
| Precondition | The person who deliver the food and the driver who is going to deliver the food have been identified by their ID, paid for the item it placed the order and provide correct direction | |
|
| Ordinary Sequence | Step | Action |
| 1 | Person requested for a food delivery |
| 2 | System request for identification data or provide an option to open an account |
| 3 | Person provide the identification or create a new account |
| 4 | System provide the option of food to order |
| 5 | Person pick up the restaurant and food from the menu |
| 6 | System display the payment information and collect the data |
| 7 | Person provide the payment detail and placed the order |
| 8 | System provide a confirmation to the user |
| 9 | System sent the information to the restaurants and to all the registered driver |
| 10 | restaurant validate the order and prepare the food. Once food is ready it will update to the system |
| 11 | System then notify the driver to pick up the food and delivery location |
| 12 | Driver will pick up and then deliver to the given address and update it in the system |
| 13 | System make the transaction as complete |
| Post condition | System can handle any delivery issue by customer support | |
| Exceptions | If the payment will not go through or restaurant is closed or restaurant information is not correct, if the drivers are not available to deliver then transition will not go through and the use case is canceled | |
| Comments | Every transaction should meet the corporate policy | |

## Business Rules

The new system must include the following:

* Ability to easily retrieve ratings on both drivers and restaurants that are a part of the system.
* Ability to see information about the individual drivers and restaurants in our system.
* Ability to see statistical analysis on ratings, such as average, min, and max.
* Ability to query all related data within a reasonable time. Query times should not exceed 5 seconds.

## Deliverables

* Use Case, Business Rules, Enhanced ERD, Schema (Due 11/18/2021)
* New Rating Table, Test Data (Due 11/25/2021)
* Queries (Due 12/2/2021)
* Final Project (Due 12/8/2021)

## Implementation Plan

Implementing this plan will require the following:

1. Import the current design and data of the existing database schema.
2. Normalize any tables, if necessary. All tables should be in 3rd normal form.
3. Break apart any many-to-many relationships that may exist.
4. Generate an ERD diagram of the current, or normalized, implementation.
5. Add in rating tables for both drivers and restaurants.
   * Should be in 3rd normal form with the proper foreign key constraints.
   * Should not have any many-to-many relationships.
   * Should contain auto-incrementing surrogate keys for the tables’ primary keys. No composite keys should be used.
   * Make sure common metadata within the design is refactored out and joined together.
6. Generate an ERD diagram of the expanded design.
7. Insert in fake, but reasonable, data into the new rating tables.
   * Can be done manually by generating and importing data from [generatedata.com](https://generatedata.com/)
   * Can be done in Python to automatically generate and run the proper INSERT statements.
8. Create functions to the schema to allow quick access to the data.
9. Create stored procedures to the schema to allow quick access to the retrieval and insertion of data.
10. Create views that will be used for reporting.
11. Index the views.
12. Record a video demo of the system and the data associated with it.