



CS 353 - Database Systems

Project Proposal

Flover

Group 30

Muzaffer Yasin Köktürk - 21703552

Muhammed Berk KÖSE - 21704277

Alptekin Önder - 21602728

İsmail Şahal - 21703596

27.02.2020



1.0 Introduction	3
2.0 Project Description	3
2.1 Why Do We Need A Database?	3
2.2 How Do We Use Database for Our Project?	3
3.0 Requirements	4
3.1 Functional Requirements	4
3.1.1. Customers	4
3.1.2 Shop Owner	4
3.1.3 Courier	4
3.1.4 Administrators	4
3.1.5 System Requirements	4
3.1.6 Customer Service Employees	5
3.2 Non-Functional Requirements	5
3.2.1 Authentication & Security	5
3.2.2 Quick Response Time	5
3.2.3 Reliability	5
3.2.4 Performance	5
4.0 Limitations	5
5.0 ER Diagram	6
6.0 Conclusion	7
7.0 Website	7

1.0 Introduction

This report aims to explain the functionality, requirements and specifications of our flower shopping website, FLOver. We will start with a description of the project explaining the necessity of a database system for the project and how we use it. Then, we will be explaining the functional and non-functional requirements of the system from the perspective of different users. Next, we will be listing some limitations of the project and finish with the E/R Diagram for the database of the system which is our main focus. You can find the website for the project at the end of this report.

2.0 Project Description

An Online Flower Shopping System is a web-based application which aims to distribute flowers online to all of Turkey. In this application, an user can choose different flowers offered by the sellers to order. User can add extra materials to the flower order like chocolates. The application creates an environment for manufacturers to get in contact with customers in a reliable and fast way.

2.1 Why Do We Need A Database?

Since this is an online system where suppliers and customers meet, there will be various types of information relating to the suppliers, customers, couriers, customer service and relationships among them. In order to make the site functional, we need to store and manage all that data. Since the data that is going to be stored will be accessed and updated frequently, it is a requirement that we use a database system which will establish basic principles of the data management such as constraints and relationships, and allow any user of the application to create, update and transmit relevant data.

2.2 How Do We Use Database for Our Project?

Since the system we are building consists of many users of various types and we need the move data from one side to another, we will design a central database to handle all the data requests and transmissions. The database will hold all the information about the customers, flower-sellers couriers as well as the orders. We will use the database to monitor and manage any function of the site that is available to the users of the site such as

searching for an item, placing an order, updating personal information, receiving an order, giving feedback etc. The application will perform these functions by creating, storing, updating and transmitting the data in the database by using requests and updates.

3.0 Requirements

3.1 Functional Requirements

Our system will support 4 different type of users which are flower-sellers, customers, couriers and admins. All users should authenticate before using any of our services.

3.1.1. Customers

- Customers can choose their favorite flower shop.
- Customers can choose their favorite flowers.
- Customers can order a floral arrangement from any shop that is available.
- Customers can place orders to be delivered in a later time.
- Customers can gain points by shopping.
- Customers can use their points for discounts.
- Customers can order flower.
- Customers should provide their address in the order.
- Customers can add a note to their order.
- Customers can choose delivery type for their order.
- Customers can choose an occasion for their order.
- Customers can choose method of payment for their order.
- Customers can rate shops.

3.1.2 Sellers

- Sellers can add new flowers to their shop.
- Sellers can assign orders to couriers.
- Sellers can assign a price to their flowers.

3.1.3 Couriers

- Couriers can reach the addresses of customers.
- Couriers can accept or reject assigned orders.

3.1.4 Administrators

- Administrators can close flower shops.
- Administrators can delete any account from the system.

3.1.5 System Requirements

- The system should prevent selling already sold flowers.

The system should keep list of all available flowers.
The system should keep transportation data for orders.
The system should keep check points of customers before sending them gift flowers.

3.1.6 Customer Service Employees

These users can view, modify the status of and act accordingly for a complaint report.
They will work according to different shifts for availability.

3.2 Non-Functional Requirements

3.2.1 Authentication & Security

- Users of the system should signup and login with their emails and passwords.
- User information should be kept securely without access to third party services without consent.

3.2.2 Quick Response Time

- The system should be able to respond the requests under 1 second.

3.2.3 Reliability

- The system should function the way it is specified.

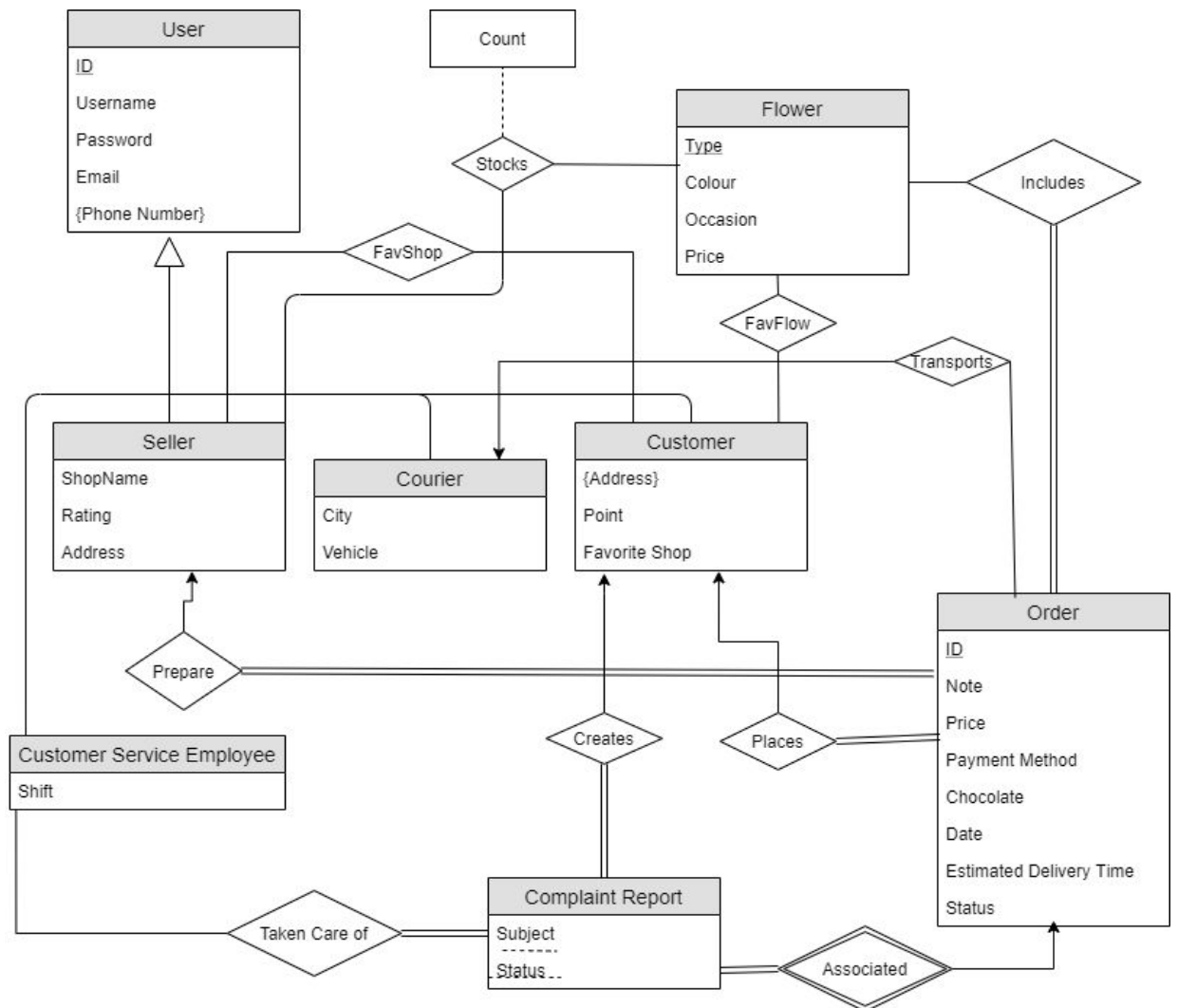
3.2.4 Performance

- The system should be able to handle 1000 simultaneous users.

4.0 Limitations

1. A customer cannot rate a seller if they did not buy any products from the seller.
2. A customer cannot order a chocolate by itself.
3. A customer cannot send letters by itself.
4. A customer cannot stack discounts.
5. A customer is not allowed to cancel an order if less than 1 hour is left for delivery.

5.0 ER Diagram



6.0 Conclusion

In this report, we have described the requirements and descriptions of our flower shopping website, FLOver. FLOver is a website where the flower sellers and buyers will be able to interact with each other. Customers can order flowers with several options.

Introduction, project description, requirements and E/R diagram can be found in this report in the given order.

7.0 Website

<https://github.com/SerialHugger/FLOver-CS353>