Bilkent University

Department of Computer Engineering

Project Proposal

CS353 DATABASE SYSTEMS

Onat Korkmaz
Muhammed Maruf Şatır
Ümit Çivi
Erdem Ege Eroğlu

October 20, 2021

http://onat.korkmaz.ug.bilkent.edu.tr/courses/cs353.html

Table of Contents

Introduction

In this report, we will talk about the Shipping Company Data Management system. This paper will include a description about the project, functional and non-functional requirements, the limitations which will be enforced by the developers and lastly a conceptual design to make everything clear.

In the description part, we will make a brief introduction to the Shipping Company DBMS. We will explain why a database is required in this context and how we can use a database for this project.

In the functional requirements part, we will talk about the functional and non-functional requirements of the project. Functional requirements play a key role in defining the system capabilities, constraints, functionality etc.. Therefore, they will provide a descriptive list of details regarding what each entity does in our project. In the non-functional requirements part, we will deal with issues like performance, security, reliability, maintainability, usability etc..

In the limitations part, we will talk about the constraints that we will enforce in our project which will be regarding the boundaries of our project.

In the conceptual design part, we will provide our entity/relationships diagram to simplify the understanding of how we plan on designing this project.

- Project reports will be at: http://onat.korkmaz.ug.bilkent.edu.tr/courses/cs353.html

Description

This project is planned to be a web-based application for shipping companies that will enable them to store their huge amount of data which is related to each other by certain common attributes. The persistent data storage is the reason why we need a database in this project so that companies can perform certain operations such as insertion, deletion and update on the data from any place and any time. Moreover, using a database we will enable users to fetch data efficiently. This way, the shipping companies will be able to manage the orders from customers, their delivery addresses, the way of delivery and the process and so on.

Requirements

Functional Requirements

Common Functionalities

- Each user will sign-up to the system.
- Users who have registered to our system can change their personal information such as passwords.

Customer

- Customers can specify cargo size and get an estimated price.
- Customers can file a report regarding a damaged or lost order.
- Customers can see the reports they have written before.
- Customers can see the result of the order whether or not it is solved.
- Customers can give the package to the courier or the employees in a branch.
- Customers can rate courier if they want.
- Customers can save their payment methods for future transactions.

Employee

- Employees can create orders when they get the packages.
- Employees can view the complaints reported to their branch.
- Employees can assign orders to the couriers.
- Employees can transfer packages between branches.
- Employees can perform transactions when the customer wants to pay cash.

Courier

- Couriers deliver the shipment to the destination from the branch.
- When the delivery is completed, the courier informs the customer about the cargo situation.
- Couriers are informed about the destination of the cargo.
- Couriers are informed about the address of the customers if customers do not bring their cargos to any branches.

Non-Functional Requirements

Security

Since we are storing sensitive user information such as address, transactions etc., the system should be resistant to any attacks such as SQL injection. The database servers should also provide safety precautions for the admins.

Maintainability

The system should be developed in such a way that when an error is encountered, the error should be fixed with minimal effort. In addition to this, any updates made to the code should not corrupt the whole system, therefore, the code should be written in a maintainable way.

Scalability

The project should be open to improvements in the future since the number of entities or relationships can increase if the company develops. Moreover, the project should be improved in a case where the number of current entities increases significantly.

Performance

The project will be displayed on a single system for each user. Thus, This feature makes the system executable with no noticeable delay. Indeed, a response takes a maximum of 0.25 milliseconds. Networking performance requirements are not needed to be specified.

Error Handling

In order to eliminate errors, we will make use of try-catch blocks in the implementation.

Usability

The GUI for the project should be easy to use for employees, customers and employees. The development of a simple GUI with maximum efficiency is a crucial part of the project.

Constraints

- The backend will be implemented in PHP.
- For database MySQL will be used.
- For front end HTML and CSS will be used.

Limitations

- Each user has to sign up with a unique email and password.
- Customers can not see others' information.
- Customers can not see couriers’ names.
- Customers have to pay before the transfer when the order is placed.
- Customers can only file a report for the order they have sent.
- Customers have to give the destination address.
- An employee can only work for one branch.
- An employee can assign a specific order to one courier only.
- An employee can not delete or modify the report filed by a customer.
- An employee cannot modify any user and order information.
- Couriers cannot see the transactions for a specific order.
- Couriers cannot create order.
- Employees cannot create orders without packages.
- Employees cannot fill a report about an order unless they placed it as a customer.

Conceptual Design