Share Cloths

Comp204 Term Project

Phase 1

Instructor

Ahmet Soran

Samet Tonyalı

Group Members

Tacettin Batuhan Bostancı

Ayşe Şeyda Çalışkan

Mustafa Demiröz

Mehmet Anıl İrfanoğlu

Dhiya Ulhaq

TABLE OF CONTENTS

LIST OF FIGURES

Abstract

This article contains means of implementation of a charity database system. Charities have significant obligation of credibility. The aim of this project is to make certain of the credibility of the help foundation by acquiring trust from the donators. The application offers for the donators to see status of their donation and to the charities to get the reputation of being trustworthy. The general mechanism of the project is outlined in the ER diagram, database design, system requirements, tools used, analysis and specifications in detail. Lastly, some screenshots from the application are given in the paper for better understanding of the project.

SUMMARY

This report is based on the project that creates a database that has been established to ensure that the connection between people who want to give their clothes to help and those who need them is faster and more effective. In the realization of this project, SQL to create the connection between people and systems, Flutter for the creation of the mobile application, and programs such as Lucid App and Draw.io as additional resources were utilized. In the following parts of the report, general description, requirement analysis, specifications, IDEs, UML, High-level diagram, E-R diagram, design philosophy, cardinalities, and user permission of the project are mentioned in detail.

REQUIREMENT ANALYSIS

The Share Clothes mobile application have charity page that user will enter the needed information about themselves and the clothes that they gave. These clothes will have different attributes such as cloth id, type, size, the date that they were given.

The person that gives cloths to charity will have name, gender and address. The user will be able to track down the status of the cloth by entering id of the cloth.

Gathering hangar will keep three attributes cloth id’s of clothes that arrived today, clothes that remain in the hangar and the clothes that left the building.

Recipient will have name, gender, upper size, lower size as body size, the number of clothes that he got aided and address. Once the cloth is aided to the recipient, status of the cloth will be changed.

The mobile application will offer list of clothes in specified gathering hangar to the charity organization so that they could list by specified size and type.

SPECIFICATION

As we know, the need for clothing is an important part of every person's life. Many organizations provide clothing aid to poor regions where this need cannot be met. After long observations, although we saw that these charities did not keep records to keep the donation confidential, we came to the conclusion that this actually caused many problems. The reason for this is that the clothes given go to other places rather than the actual destination. Therefore, we decided to set up a database. Thanks to this system, users will open an account and donate clothes and give details about the clothes they give. The charity employees collecting these clothes will make a classification process based on the characteristics of the dress givers entered into the system. Finally, people who need clothes will register to the system and give some information. To give an example of this information, these are some elements that are important in giving clothes such as cloth size and cloth type. When all these things are implemented, a very effective cloth sharing system emerges. and thanks to these records, curious people will be able to confirm that the dresses are reaching the people who really need it.

GENERAL DESCRIPTION

The cloth aid system our purpose is to make sufficient and organizable application with the MySQL database management system. In the system, there are 2 significant data one of them is people who need clothes and the other one is people who sent his/him clothes. In the cloth aid system, we will keep the recipient, donor information. Determining the size of the clothes that will go to help beforehand, understanding who needs which clothes by the system, we will be late for the wrong clothes to go. At the same time, we aim to make the aid campaign more contractual and efficient by keeping the information of the people who help and the people helped, the information of the transfer center, and the information of the people who will help. In this way, by having a general background by the charity organization, aid campaign information can be easily sent to the same people in future aid campaigns.

TOOLS/IDES

MySQL Workbench:
It is the most important program in order to complete Project. It will be used to create and manage database system of the Project.

Android Studio:
Android studio is the most common interface to create mobile application. İt will be used with flutter.

Dart:
This language is used fort to create mobile application both on android and IOS. So, it will be used in our Project.

Flutter:
Flutter is dart based mobile programming system. By the help of flutter, application could run on both android and IOS and flutter will be used in our Project in order to create mobile application.

GitHub:
It will be used to share the versions of mobile application.

HIGH LEVEL DIAGRAM

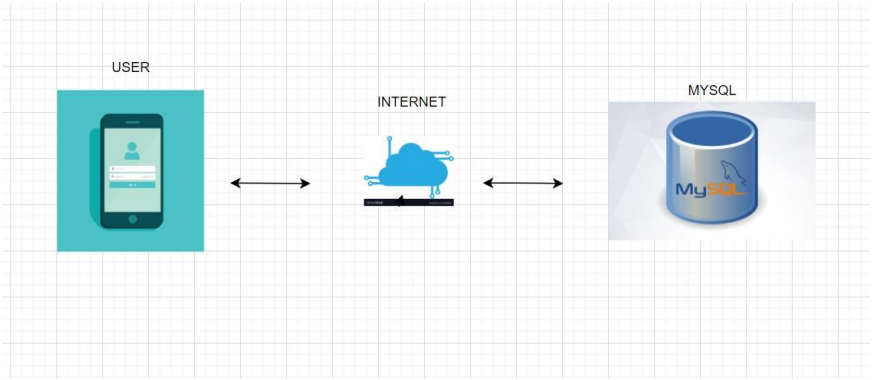

Figure1.: Tools

UML USE CASE DIAGRAM

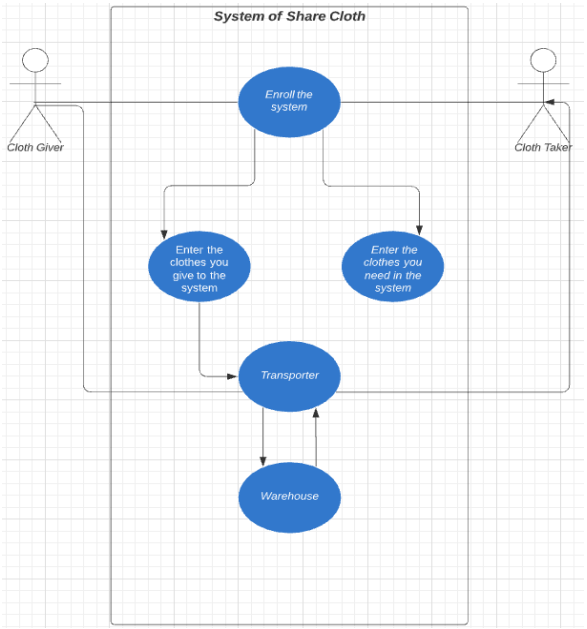

Figure2. UML Diagram

E-R DIAGRAM

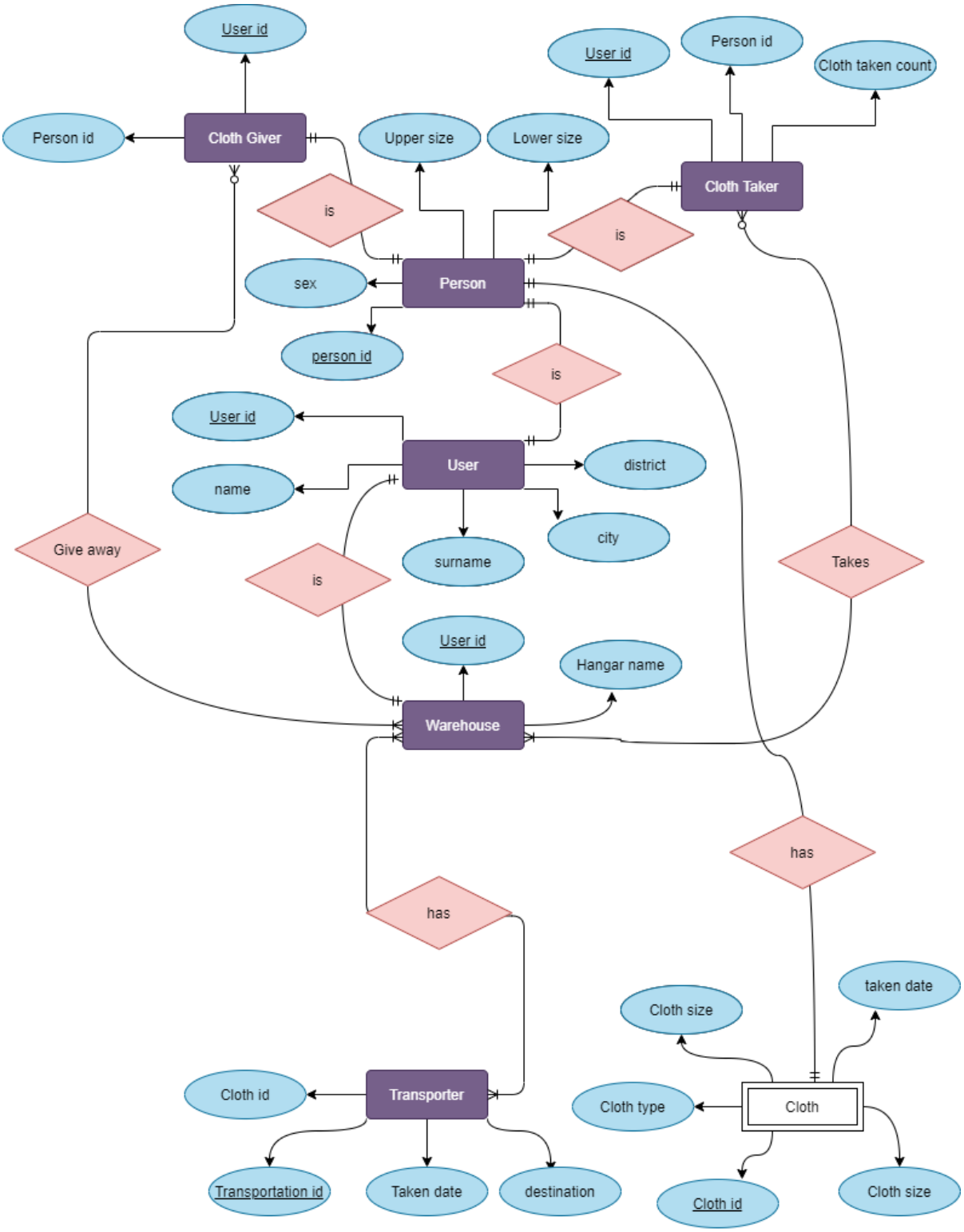

Figure2.: E-R Diagram

DESIGN PHILOSOPHY

ER Diagram:

- **User**
- **Person**
- **Cloth Giver**
- **Cloth Taker**
- **Clothe**
- **Warehouse**
- **Transporter**

In this part we will show the ER diagram to explain the whole system. By showing the ER diagram, sections that are not understood in the ER diagram will be understood.

User is the super class for this project and it gives its information to the **Cloth_giver**, **Cloth_taker**, Hangar and **Warehouse**. Basically, it keeps the general information of the user like: name, surname, city and **user_id**. **User_id** will be generated automatically and with this way other entities takes user entity information's.

Person class keeps **upper_size, lover_size, sex** and **persond_id (generated automatically).** Also, there are another two subclass which are **Cloth_taker, Cloth_giver** sub classes, those are connected to the Person class as Sub-Classes.

Cloth_giver is a sub-class of Person entity. **Cloth_giver** keeps **user_id, cloth_type**, **cloth_size**. Finally, it has a foreign key comes from User Entity.

Cloth Taker table represents the person who is in need of cloth and also is a user which can state his or her need in the system. He or she has a unique **user_id** and by the help of this **user_id** operations become much simpler format.

Cloth table is the most important table in the system. The given and taken clothes will be selected from this table and cloth table has a unique key which is **cloth id** and with this cloth id some operations will be tracked.

Hangar table stands for the main center of complete cloths. The taken cloths will be stored in this hangar and hangar personals will see the stocks of cloths and depending on the cloth needs they will create transportation route which will be handled by **transporters** and also there is a specific transporter table.

Transporter is a tablet o keep transportation information's in it. The main key for transportation table is **transportation id** and with this id, we will learn who is transporting the goods and also in this tablet here is **cloth id** and we will know what he or she is transporting.

Cardinalities

ONE TO ONE:
User, Person
Person, Cloth Giver
Person, Cloth Taker
User, Warehouse
Person, Cloth

MANY TO OPTIONAL MANY
Warehouse, Cloth Giver
Warehouse, Cloth Taker

MANY TO MANY
Warehouse, Transporter

USER PERMISSIONS

CLOTH GIVER
Add: User, Clothe type
Delete: User, Clothe type,
Update: User, Clothe type,
View: User, Clothe type, Warehouse

CLOTH TAKER
Add: User, Clothe type
Delete: User, Clothe type,
Update: User, Clothe type,
View: User, Clothe type, Warehouse

WAREHOUSE
Add: Hangar name
Delete: Hangar name
Update: Hangar name
View: Hangar name, Cloth giver, Cloth Taker