Department of Computer Engineering

University of Peradeniya

EasyMed

- AN ONLINE PHARMACY -

E/15/123 Wishma Herath
E/15/280 Pubudu Premathilaka
E/15/316 Suneth Samarasinghe

CO527 Project Report

June 2020

Contents

1 Introduction

1.1 How we got the idea?

Online shopping is one of the most popular usages of technology these days. Buying things you want is a fingertip away from your mobile phone. But, when it comes to medical products, people prefer to buy them offline from a pharmacy rather than ordering from random dealers on a regular online shopping website. Therefore, we thought of creating an online platform that focused on building trust with the customer. The frustration people faced getting their medicines waiting in queues near pharmacies during the current situation in the country showed us the importance of such a platform.

1.2 What is EasyMed?

EasyMed provides a virtual market space for pharmacies to serve their customers. Also it offers better pricing than offline stores, with increased access, lower transaction and product costs, convenience and greater anonymity for consumers. EasyMed offers accessibility to people with limited mobility and people in remote areas. It provides media alerts (personalized medicine reminder service), discounts, doorstep delivery within a short time, and validation of prescription through licensed pharmacists.

Information about substitutes and adverse effects is also available on the site. A prescription for medicines ordered is mandatory, except for sale of over-the-counter products. Orders for medicines can be taken only from areas where the pharmacy retail license applies. All the medicines must be verified and certified by the registered pharmacist before delivery. Also in EasyMed, you can have a way to monitor your prescriptions.

Conventional 'bricks and mortar' pharmacies usually have controlled drug distribution systems from the manufacturer, sufficient validation, and follow good distribution practices. Home delivery of pharmaceuticals is a desirable convenience. The time frame for making deliveries can be greatly reduced by this type of pharmacy. The time required to verify prescription fills for each patient is minimized. A Pharmacy can register in EasyMed and have a separate virtual stall and start selling. Customers can explore the market space and choose what they want from a lot of stalls while comparing reviews/recommendations and ratings.

1.3 How is it different?

During the feasibility study, we found similar applications developed in other countries but not in Sri Lanka. There are a few local services that promote their own business with online marketing. EasyMed has the pharmacies that are already known to the customers. Therefore, customers can use the platform without any hesitation.

2 The overview of the database

2.1 Diagrams

2.1.1 Entity Relationship Diagram

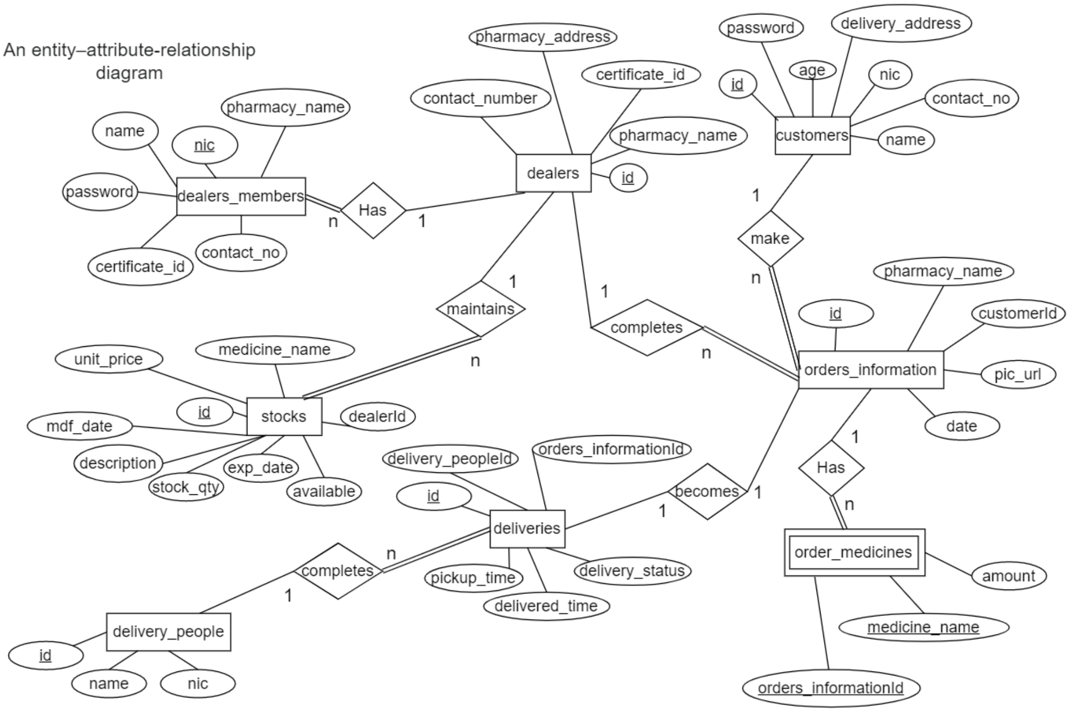

Figure 1: *ER Diagram*

2.1.2 Schema Diagram

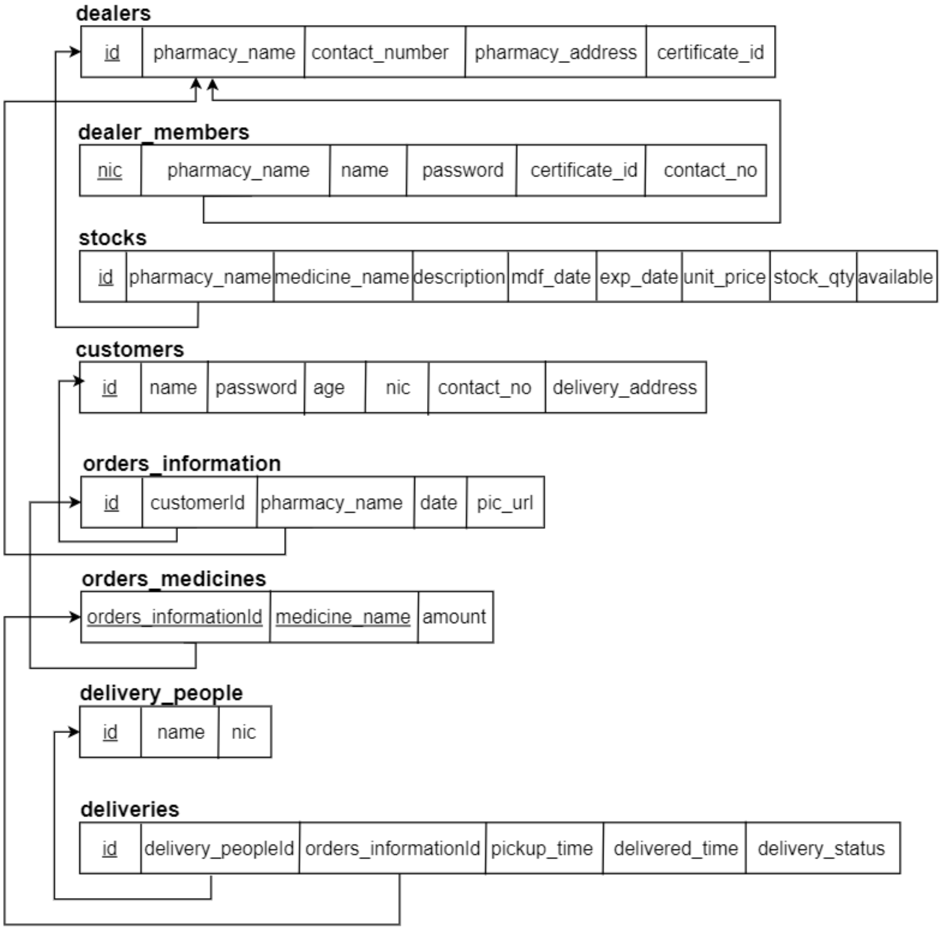
dealers
id | pharmacy_name | contact_number | pharmacy_address | certificate_id
dealer_members
nic | pharmacy_name | name | password | certificate_id | contact_no
stocks
id | pharmacy_name | medicine_name | description | mdf_date | exp_date | unit_price | stock_qty | available
customers
id | name | password | age | nic | contact_no | delivery_address
orders_information
id | customerId | pharmacy_name | date | pic_url
orders_medicines
orders_informationId | medicine_name | amount
delivery_people
id | name | nic
deliveries
id | delivery_peopleId | orders_informationId | pickup_time | delivered_time | delivery_status

Figure 2: *Schema Diagram*

2.1.3 Conceptual Diagram

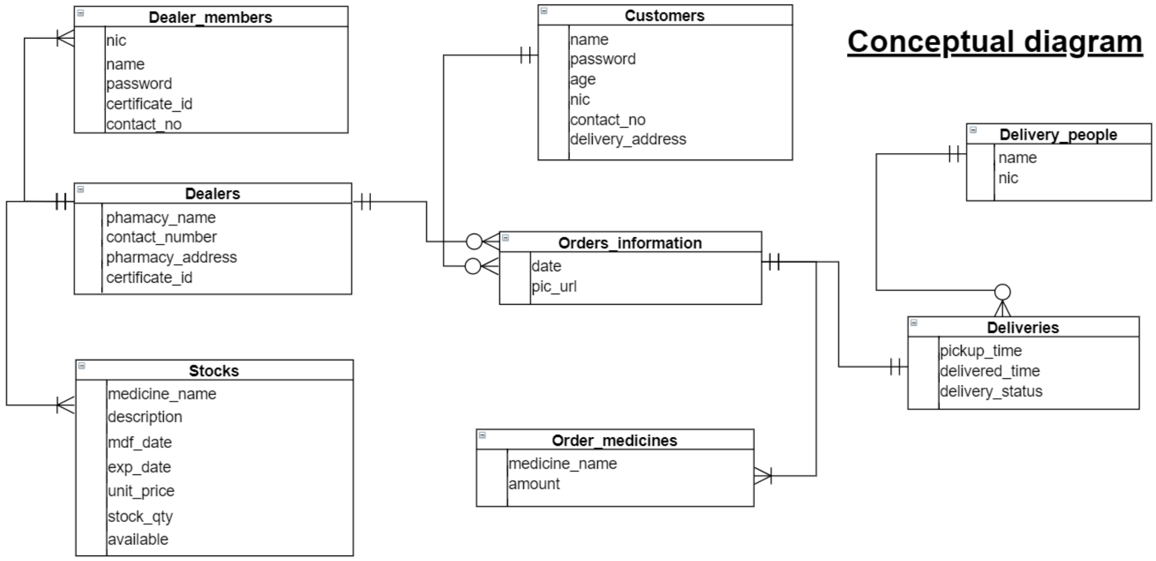

Figure 3: *Conceptual Diagram*

2.1.4 Logical Diagram

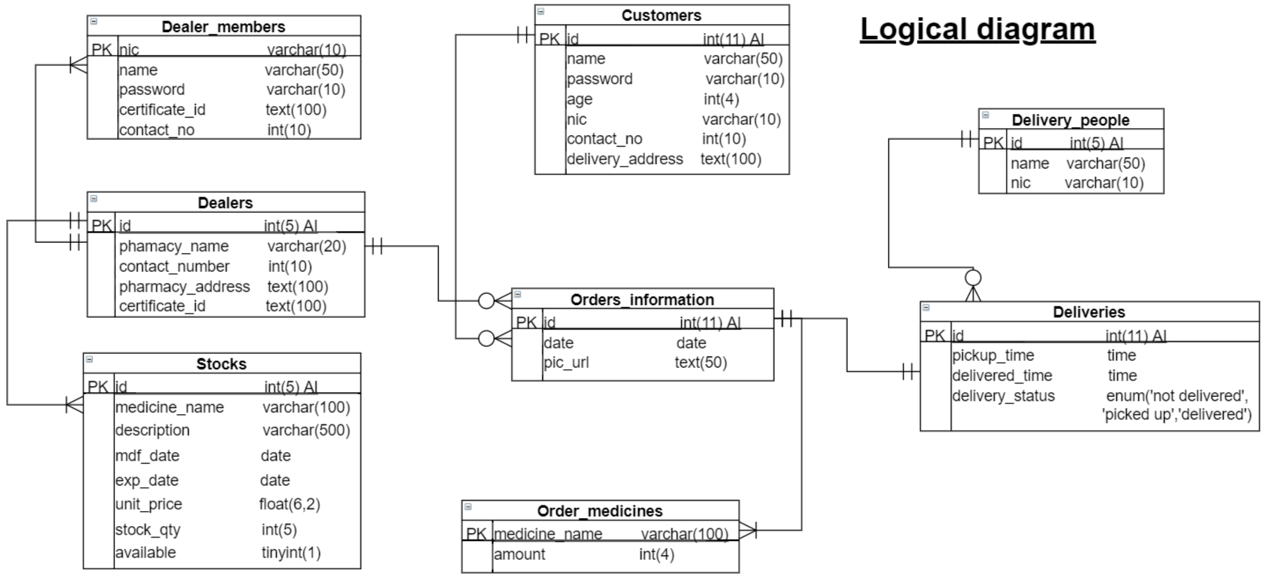

Figure 4: *Logical Diagram*

2.1.5 Physical Diagram

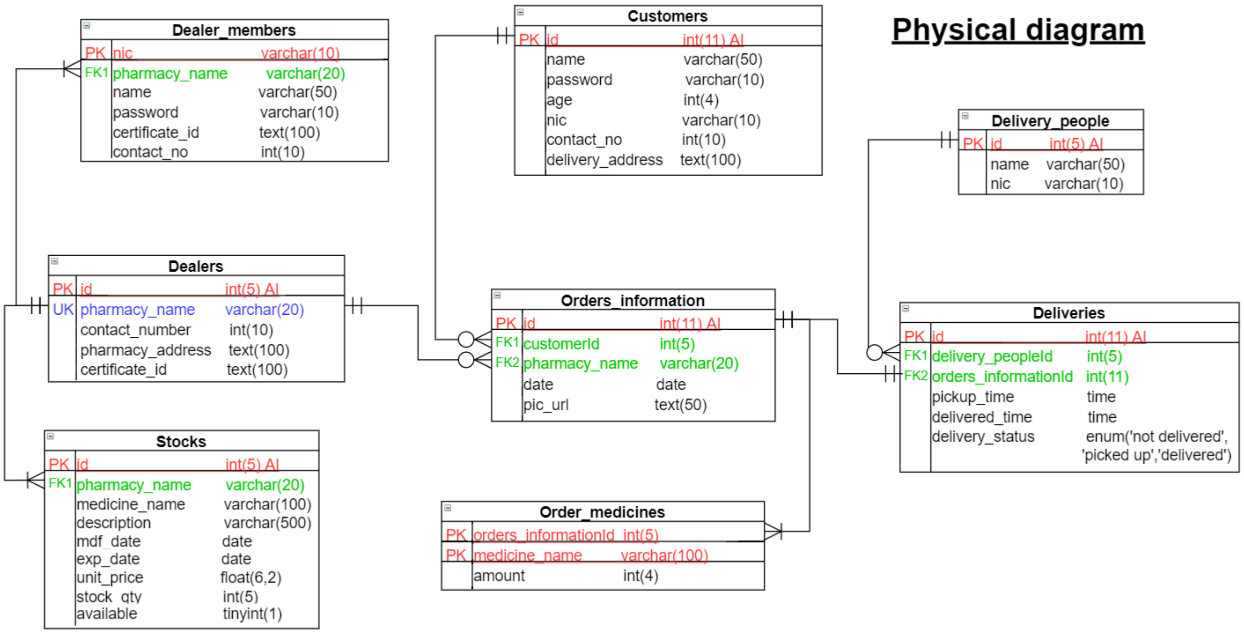

Figure 5: *Physical Diagram*

3 The abstract description on each table

3.1 Customers

This table includes details we keep track of about the customers. When a new customer is registered, a new entry is inserted. When a user login hashed password is retrieved. Other necessary details are taken from here when putting an order.

PK	id
FK	-
Indexes	Primary key index (id)
attributes	name, password, age, nic, contact_no, delivery_address

This table has a (1:m) relation with the orders_information table(which described below).

3.2 Dealers

Dealers represent the pharmacies. In our DBMS many pharmacies can be registered. Each registered pharmacy's details are displays here. Each dealer has its own crew, which is an entity of the ER diagram named dealer_members. Dealers table has relations with the 'stocks' table, 'orders_information' table, and 'dealer_members' as mentioned earlier. Each relation represents 1:m relationship as shown in the above diagrams.

PK	id
UK	pharmacy_name
Indexes	pharmacy_name$_index$
attributes	contact_no pharmacy_address certificate_id

3.3 Deliveries

This table consists of the records of delivery data. Once a customer has requested an order, after the process of preparation, it should be added to the deliveries table (which is done by dealers). Then, the delivery agent can deliver the requested order according to the data in the ‘deliveries’ table.

PK	id
FK	delivery_peopleId, orders_informationId
Indexes	delivery_peopleId$_{i}ndex, orders\_informationId\_index$
attributes	pickup_time, delivered_time, delivery_status

3.4 Delivery people

This table includes details about the agents who do the deliveries. Here, by considering the ease of the database system, this table is made as simple as possible. Because the main concern is to provide a medicine distribution system for the customers and dealers. For the completion of the system, the ‘delivery_people’ entity is created.

PK	id
FK	-
Indexes	Primary key index (id)
attributes	name, nic

3.5 Order_information

Order_information represents the details about the orders which are made by customers. This table can be viewed by all the roles of the system(i.e. Customers, dealers, delivery_agents).

PK	id
FK	customerId, pharmacy_name
Indexes	customer_id_index, pharmacy_name_index
attributes	date, pic_url

3.6 Order_medicines

This is a weak entity. This means it cannot be uniquely identified by its attributes alone but it uses a foreign key(i.e. orders_informationId) in conjunction with its attributes. This table includes the records on the names and amount of medicines ordered by the customers.

PK	id
FK	customerId, pharmacy_name
Indexes	composite_index (orders_informationId, medicine_name)
attributes	date, pic_url

3.7 Stocks

This entity represents the medicine stocks of each dealer (i.e pharmacy) has.

PK	id
FK	pharmacy_name
Indexes	medicine_name, description, mdf_date, exp_date, unit_price, stock_qty, available

4 Queries

Queries are included under three categories

1. Customers

   - Order medication from the desired pharmacy
   - Add medication to the list, update, delete, review the order (basically CRUD)
   - Get the total price, delivery status
   - CRUD profile
   - Order history based on date, pharmacy
   - Display info of pharmacy

2. Dealers

   - Add medicine to stock (CRUD)
   - Login/ registering to the DB
   - CRUD profile
   - Search medicine based on medi_name
   - Get medicine info of medi_name, description, modified date, exp_date, unit price, availability, stock quantity

3. Delivery_people

   - Update the deliveries table's pickup, delivered time and status
   - Get customers info to deliver
   - Get the price that should be collected from the customers

5 Transaction

A transaction is a very small unit of a program and it may contain several low-level tasks. A transaction in a database system must maintain Atomicity, Consistency, Isolation, and Durability commonly known as ACID properties in order to ensure accuracy, completeness, and data integrity.

To archive the above-mentioned properties/qualities, basically 4 store procedures are used.

1. dealersProcedure
2. customerProcedure
3. orderMedicines
4. deliveriesProcedure

The main aspect of these procedures is when inserting or updating the dealers table data or customers table data, The respective queries will be executed using these store procedures.

Especially, ‘orderMedicine’ store procedure is used to update both order_information table and its weak entity order_medicines table concurrently.

All the respected store procedures are included in the store_procedures.sql file.

6 Security

The following shows the security aspects relative to each roll of the database. There are 4 user defined database roles from in the back end. Their jobs are as follows

- admin_dealer: Handle dealer related table. admin_dealers as full access to the stocks table. He can read and update Dealers, Dealer_members table. The reason for that is in order to delete or insert new dealer or dealer member a higher level of access is required.
- admin_customer: admin_customer can read, update customers table
- admin_delivery: admin_delivery can read, update delivery_people table
- admin_orders: Responsible for read, write, delete, insert orders and manage deliveries.

User authentication

When a new user is registered to the system the password is not stored in the database as it is. Instead, the hashed value is stored. Whenever the user tries to log in, the hash value of the entered password is compared with the one stored in the 'customers' table in the database.

All the above security aspects are obtained by granting necessary privileges for each of the user-defined database roles. All the SQL queries regarding the security aspects are included in the security.sql file.

7 Backup features

7.1 Mysqldump utility

The mysqldump utility is a console-driven executable that helps us to backup a database to an external resource such as a file or to a different MySQL server. We use mysqldump utility to recreate the database to a local directory. mysqldump is only executed by the admin role in our database. It can be simply used as follows:

```
mysqldump ---user [user name] ---password=[password]
[database name] > [dump file]
```

In case of a failure in the database, if we want to restore the data back, it can be done simply using the following command:

```
Mysql [database name] < [backup file name]
```

Since mysqldump utility is flexible and has a lot of other features, it can be used in further improvements of the database as well. But, it has to be done manually by the database admin. Therefore we thought of using an automated backup tool.

7.2 AutoMySQLBackup tool

AutoMySQLBackup tool has the ability to create daily, weekly and monthly backups of one or more databases. Also, it facilitates email notification of backups, backup compression and encryption, configurable backup rotation, and incremental database backups. We have not included all the features of this tool for the project yet.

8 The wireframe

A wireframe is a blueprint that illustrates the elements of a web site. Creating a wireframe gives the client, the developer, and the designer an opportunity to take a critical look at the structure of the website and allows them to make revisions easily. Which indirectly indicates that the requirements of the database as well.

Mainly the idea of what should be the queries which should be included in the database is decided based on the design of the wireframe of the website. The wireframe is provided as an attached pdf file. For instance, using the patient page(i.e. the 'customer' roll in the database), a query can be obtained such that the patient wants to see the details about a particular pharmacy.

In this matter, the related query can be built according to the requirement of the wireframe. Most of the important queries are decided based on the wireframe. When deciding the store procedures including the transactions of the database, the arrangement of the wireframe was very useful.

The wireframe arrangement

The proposed wireframe consists of 5 pages.

- The homepage
- One patient page
- Two pharmacy pages
- One delivery service page.

The homepage

Simply this displays the banners, descriptions of the services and etc.

The patient's page

Once a patient logged into the website, the patient can request the order by adding the relevant medications with the respected amount. As an extra feature (for the conformation) photograph of the prescription can be drop to the file input. Then, the order can be requested.

The Pharmacist page1

From this page, a dealer member can observe the requested orders and then, preparing the order for the patient can be done. Sending the total price and the confirmation request can be made using this page.

The Pharmacist page2

By observing this page, the pharmacy member will be informed about the current status of the delivery(now the order has been given to the delivery service.). The status shows whether the delivery is ‘picked up/delivered/not delivered’.

The Delivery Service page

This page is available for the delivery service. By observing this, the delivery person can observe the customer’s information as displays in the wireframe. So the delivery can be made according to the information.

The Delivery Service page

This page is available for the delivery service. By observing this, the delivery person can observe the customer’s information as displayed in the wireframe. So the delivery can be made according to the information.

(In the front end development, a very basic website is built up to the homepage and the customer page. The login functionality is working fine with it. All the front end source codes are provided with the submission.)

9 Why MySQL

- A vast and expert community is available in MySQL.
- SQL is a standard language.
- Each DBMS solves a different need; they are not interchangeable substitutes for each other.
- Relational databases have a goal of minimizing data anomalies and enforcing data integrity in a declarative way. Using a schema, data types, and constraints help you avoid writing lots of application code.
- Non-relational databases have a goal of optimizing data access for a particular set of query types (often at the expense of other query types).
- Traditional relational data management systems also offer a rich set of features around such concepts as ”Atomicity”, ”Consistency”, ”Isolation”, and ”Durability”, the features collectively called ACID. NoSQL solutions provide different guarantees and functionality in this area and no standards have emerged at this time.
- NoSQL databases don't have the reliability functions which Relational Databases have (basically don't support ACID).
- In NoSQL In order to support ACID developers will have to implement their own code, making their systems more complex.
- NoSQL is not compatible (at all) with SQL.
- NoSQL is very new compared to Relational Databases, which means that they are far less stable and may have a lot fewer functionalities.

By considering the above facts, we have chosen MySQL for the database management system.

10 Summary

According to the project requirements, a multi-user database system has to be implemented with the following requirements.

- Multiple users with different access roles
- Use of proper indexes
- Transaction processing
- Optimized queries
- Security and backup features
- Comparison with other data models.

By implementing 'medical_database' in our project, each and every requirement has been covered. But in practical scenarios, these mentioned queries and the arrangement of the database may not be very suited. Here our main concern is to proper use of MySQL for the database management system.

11 References

- https://www.quora.com/Which-is-better-MySQL-or-NoSQL
- https://www.mysqltutorial.org/mysql-transaction.aspx/: :text=MySQL
- https://dev.mysql.com/doc/refman/5.6/en/backup-and-recovery.html
- https://dev.to/lmolivera/everything-you-need-to-know-about-nosql-databases-3o3h
- https://stackoverflow.com/questions/35169922/mysql-stored-procedure-insert-multiple-rows-from-list

12 Step-wise project plan

- Identify and plan for different parties and their roles
- Gather all of the types of information you might want to record in the database
- Divide the information into tables
- Turn information items into columns
- Specify primary keys
- Set up the table relationships
- Initial ER diagram
- Refine ER diagram
- Initial Conceptual diagram and schema diagram
- Refine Conceptual diagram and schema diagram
- Initial logical diagram
- Refine logical diagram
- Initial physical diagram
- Refine physical diagram
- User Interface Sketch
- Web interface
- Testing web interface
- Login and Authentication
- Tool selection (DBMS, front end, mobile app if included, etc)
- Create tables and relations
- Populate with data
- Test the database using sample queries
- Indexing
- Write Queries
- Query optimization
- Security and backup features