

# **Medicine Review**

**Database Systems (IT252)**

*Submitted by*

**Akshith N M (181IT104)**

**Laharish S (181IT125)**

**Harshvardhan R (181IT217)**

**IV SEM B.Tech (IT)**

*Under the guidance of*

**Dr. Melwyn D'souza**

**Dept of IT,NITK Surathkal**

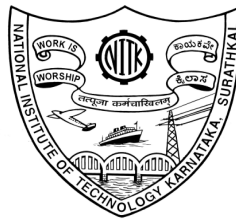
*in partial fulfillment for the award of the degree  
of*

**Bachelor of Technology**

**in**

**Information Technology**

**at**



**Department of Information Technology**

**National Institute of Technology Karnataka, Surathkal.**

***June 2020***

# **ABSTRACT**

Medicines are always best taken under a prescription from a doctor. But that does not stop one from knowing the details about the medication he or she receives. Review of medicines and medicine-taking is seen as an essential aspect of health care. The web application provides the users to search for a prescribed drug and get more information about it. Since a drug must not only display the information given on paper, there is also an option to view the comments written by other users which serve as feedback listing out the side effects they suffered from or any general review. In case some users are allergic to the constituents of one medicine they can view similar medicines having the same drug.

# TABLE OF CONTENTS

<b>ABSTRACT</b>	<b>2</b>
<b>TABLE OF CONTENTS</b>	<b>3</b>
<b>Chapter 1: PROJECT OVERVIEW</b>	<b>4</b>
Introduction	4
Functional Requirements	5
<b>Chapter 2: DATABASE DESIGN</b>	<b>6</b>
ER Diagram	6
Mapping to Schema	7
Functional Dependencies	8
Normalization	9
Views	10
Trigger	10
Index	11
Stored Procedures	11
Transactions	12
<b>Chapter 3: IMPLEMENTATION</b>	<b>13</b>
Tools and Technology	13
Application Architecture	14
Screenshots	15
<b>INDIVIDUAL CONTRIBUTION</b>	<b>18</b>

# **Chapter 1: PROJECT OVERVIEW**

## **Introduction**

The web application provides a solution for searching for information on medicines. A user can sign up with his details including his or her name, email, phone number, blood group, date of birth, height, weight, gender and role. The user role can be either a pharmacist or a general user. A general user can also rate the medicine and add comments. On searching for a particular medicine the user is presented with the drugs page, which has the following details:

- The description of the medicine
- The diseases the medicine can cure
- The average rating of the medicine
- The alternate medicines that can be used
- The comments by other users

The pharmacist has the permission to :

- Add medicines and their description
- Add diseases

The user can keep track of his medication in his profile page.

## **Functional Requirements**

- Users can register and login using their email and password.
- Users can search for a drug.
- Users can rate and comment on a drug.
- Users can get to know about similar drugs ,diseases treated.
- Users can keep track of their medical records.
- Pharmacists can add drugs and its details.
- Pharmacists can add diseases.

# Chapter 2: DATABASE DESIGN

## ER Diagram

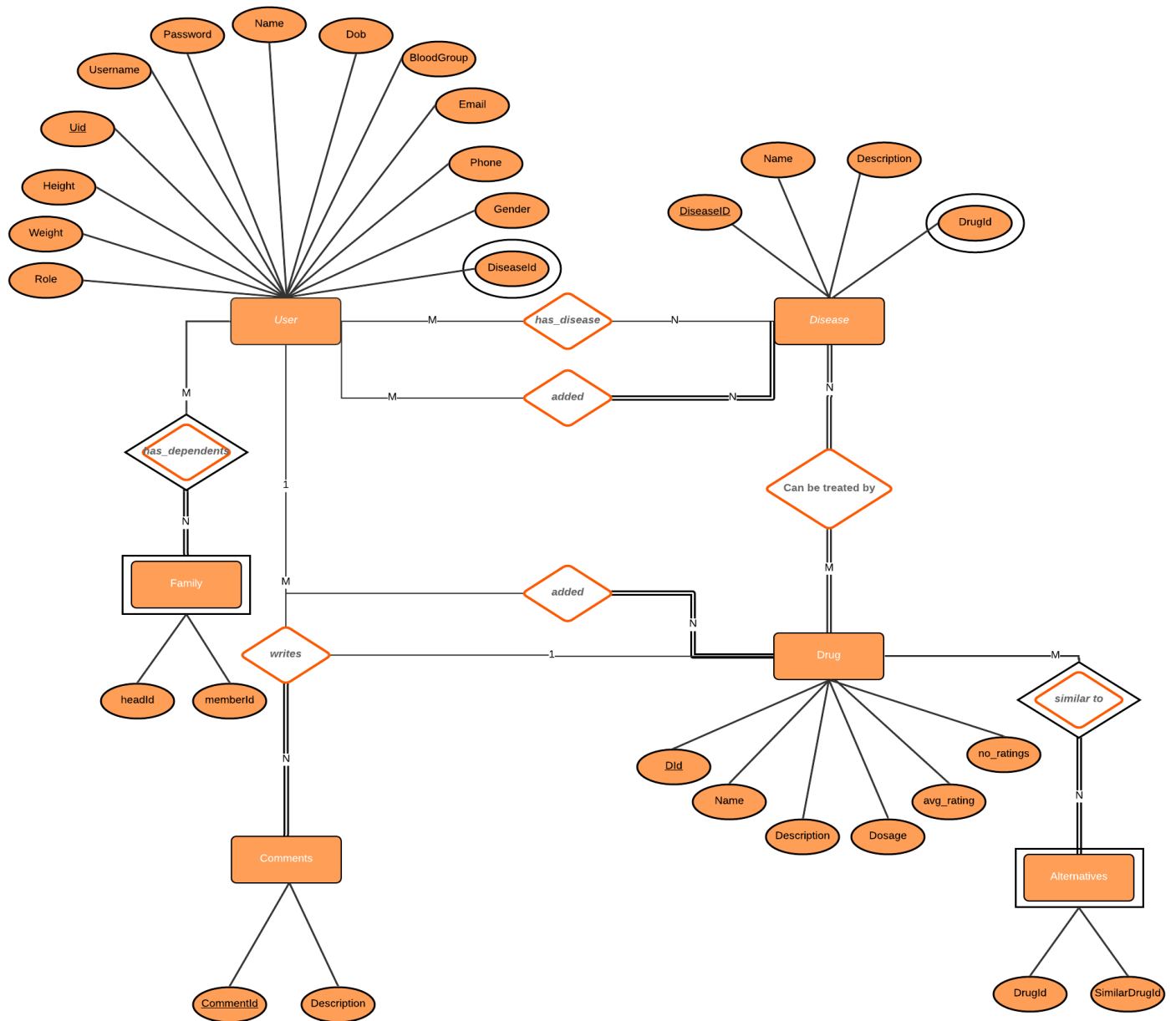
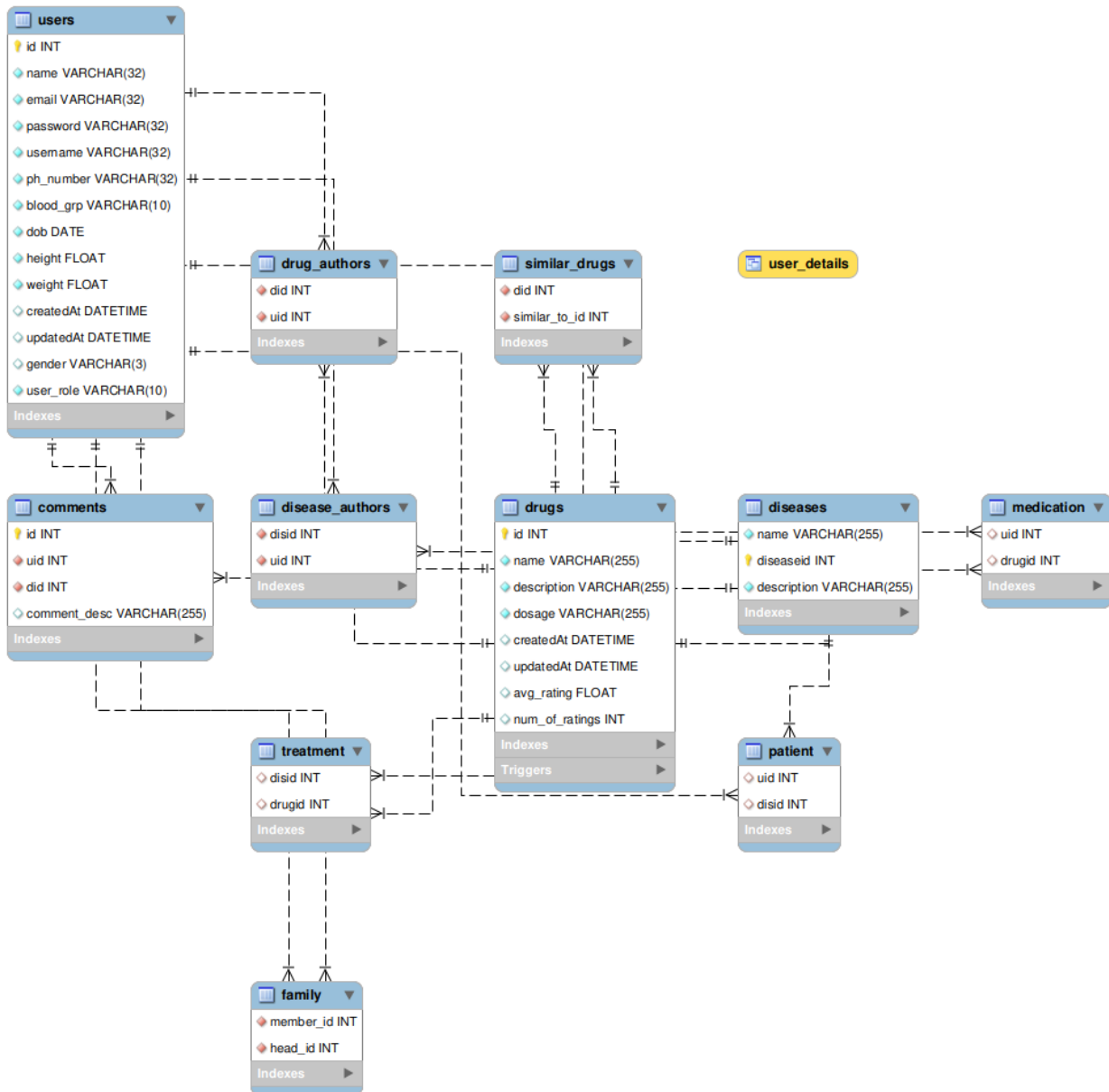


Fig 1. ER diagram of the database

# Mapping to Schema



## Functional Dependencies

Determinant	Dependent
user_id	<ul style="list-style-type: none"><li>• disease_author</li><li>• drug_author</li><li>• comment_user_id</li><li>• patient_user_id</li><li>• member_user_id</li></ul>
drug_id	<ul style="list-style-type: none"><li>• similar_drug_id</li><li>• medication_drug_id</li><li>• treatment_drug_id</li><li>• comment_drug_id</li><li>• author_drug_id</li></ul>
disease_id	<ul style="list-style-type: none"><li>• patient_disease_id</li><li>• author_disease_id</li><li>• treatment_disease_id</li></ul>



## **Normalization**

The database satisfies 3NF Normalization

For satisfying 1 NF Normalization:

- Each table cell should contain a single value.
- Each record needs to be unique.

For satisfying 2 NF Normalization:

- Each table must have Single Column Primary Key

For satisfying 3 NF Normalization:

- None of the table have transitive functional dependencies
- 
1. Multiple similar\_drugs can exist for a single drug so a new table “similar\_drugs” is created.
  2. Multiple authors can exist for a single drug so a new table “drug\_authors” is created which has both the userId and drugId.
  3. Multiple authors can exist for a single disease, so a new table “drug\_authors” is created which has both the userId and diseaseId.
  4. A person can be under the medication of multiple drugs or suffer from multiple diseases, so new tables “medication” and “patient” are created respectively.
  5. A drug can cure multiple diseases so a new table “treatment” is created.
  6. A person can have multiple users part of his or her family so a new table “family” is created.
  7. All tables have a single Primary Key.
  8. All transitive functional dependencies have been avoided.

## Views

The view “user\_details” prevents access to the password of the users ,record creation time and updation time.

```
mysql> desc user_details;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id         | int(11)       | NO   |     | 0        |       |
| name       | varchar(32)   | NO   |     | NULL     |       |
| email      | varchar(32)   | NO   |     | NULL     |       |
| username   | varchar(32)   | NO   |     | NULL     |       |
| ph_number  | varchar(32)   | NO   |     | NULL     |       |
| dob        | date          | NO   |     | NULL     |       |
| height     | float         | NO   |     | NULL     |       |
| weight     | float         | NO   |     | NULL     |       |
| gender     | varchar(3)    | YES  |     | NULL     |       |
| user_role  | varchar(10)   | NO   |     | NULL     |       |
+-----+-----+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

Fig 2 user\_details view

## Trigger

A trigger is a stored program invoked automatically in response to an event such as insert, update, or delete that occurs in the associated table. The trigger “update\_rating” calculated the average rating for the particular drug and stores it in the drugs table

```
mysql> delimiter $$
mysql> create trigger update_rating before update on drugs
-> for each row begin
-> set @old_avg=(select avg_rating from drugs where id=NEW.id);
-> set @old_num=(select num_of_ratings from drugs where id=NEW.id);
-> set NEW.avg_rating=(@old_avg*@old_num+NEW.avg_rating)/(@old_num+1);
-> set NEW.num_of_ratings=@old_num+1;
-> end $$
```

Fig 3 update\_rating trigger

## Index

Indexes are special lookup tables that the database search engine can use to speed up data retrieval.

1. The index `search_drug_index` allows faster retrieval of a drug by its name and it prevents duplicate entries to the table.
2. The index `get_user_index` is used for faster retrieval of user details by username.
3. The index `get_comments_index` is used for faster retrieval of comments .

```
create unique index search_drug_index on drugs (name)
create index get_user_index on users (username)
create index get_comments_index on comments (uid,did)
```

Fig 4 indexes used

## Stored Procedures

All the required queries are placed in stored procedures so that they can be called from the backend.

Db	Name	Type	Definer	Modified	Created	Security_type	Comment	character_set_client	collation_connection	Database Collation
final_medrev	add_comment	PROCEDURE	root@localhost	2020-06-07 14:47:02	2020-06-07 14:47:02	DEFINER		utf8	utf8_general_ci	latin1_swedish_ci
final_medrev	add_disease	PROCEDURE	root@localhost	2020-06-07 14:47:02	2020-06-07 14:47:02	DEFINER		utf8	utf8_general_ci	latin1_swedish_ci
final_medrev	add_drug	PROCEDURE	root@localhost	2020-06-07 14:47:02	2020-06-07 14:47:02	DEFINER		utf8mb4	utf8mb4_general_ci	latin1_swedish_ci
final_medrev	add_family	PROCEDURE	root@localhost	2020-06-07 14:47:02	2020-06-07 14:47:02	DEFINER		utf8	utf8_general_ci	latin1_swedish_ci
final_medrev	add_medication	PROCEDURE	root@localhost	2020-06-07 14:47:02	2020-06-07 14:47:02	DEFINER		utf8mb4	utf8mb4_general_ci	latin1_swedish_ci
final_medrev	add_patient	PROCEDURE	root@localhost	2020-06-07 14:47:02	2020-06-07 14:47:02	DEFINER		utf8	utf8_general_ci	latin1_swedish_ci
final_medrev	add_rating	PROCEDURE	root@localhost	2020-06-07 14:47:02	2020-06-07 14:47:02	DEFINER		utf8	utf8_general_ci	latin1_swedish_ci
final_medrev	getlist	PROCEDURE	root@localhost	2020-06-07 14:47:02	2020-06-07 14:47:02	DEFINER		utf8mb4	utf8mb4_general_ci	latin1_swedish_ci
final_medrev	get_comments	PROCEDURE	root@localhost	2020-06-07 14:47:02	2020-06-07 14:47:02	DEFINER		utf8	utf8_general_ci	latin1_swedish_ci
final_medrev	get_diseases_cured	PROCEDURE	root@localhost	2020-06-07 14:47:02	2020-06-07 14:47:02	DEFINER		utf8	utf8_general_ci	latin1_swedish_ci
final_medrev	get_drug_authors	PROCEDURE	root@localhost	2020-06-07 14:47:02	2020-06-07 14:47:02	DEFINER		utf8	utf8_general_ci	latin1_swedish_ci
final_medrev	get_drug_details	PROCEDURE	root@localhost	2020-06-07 14:47:02	2020-06-07 14:47:02	DEFINER		utf8mb4	utf8mb4_general_ci	latin1_swedish_ci
final_medrev	get_medication	PROCEDURE	root@localhost	2020-06-07 14:47:02	2020-06-07 14:47:02	DEFINER		utf8mb4	utf8mb4_general_ci	latin1_swedish_ci
final_medrev	get_similar_drugs	PROCEDURE	root@localhost	2020-06-07 14:47:02	2020-06-07 14:47:02	DEFINER		utf8	utf8_general_ci	latin1_swedish_ci
final_medrev	get_user	PROCEDURE	root@localhost	2020-06-07 14:47:02	2020-06-07 14:47:02	DEFINER		utf8mb4	utf8mb4_general_ci	latin1_swedish_ci
final_medrev	insert_disease_treated_by	PROCEDURE	root@localhost	2020-06-07 14:47:02	2020-06-07 14:47:02	DEFINER		utf8	utf8_general_ci	latin1_swedish_ci
final_medrev	insert_drug_cures	PROCEDURE	root@localhost	2020-06-07 14:47:02	2020-06-07 14:47:02	DEFINER		utf8	utf8_general_ci	latin1_swedish_ci
final_medrev	insert_similar_drugs	PROCEDURE	root@localhost	2020-06-07 14:47:02	2020-06-07 14:47:02	DEFINER		utf8	utf8_general_ci	latin1_swedish_ci
final_medrev	register_user	PROCEDURE	root@localhost	2020-06-07 14:47:02	2020-06-07 14:47:02	DEFINER		utf8	utf8_general_ci	latin1_swedish_ci
final_medrev	search	PROCEDURE	root@localhost	2020-06-07 14:47:02	2020-06-07 14:47:02	DEFINER		utf8mb4	utf8mb4_general_ci	latin1_swedish_ci
final_medrev	search_disease	PROCEDURE	root@localhost	2020-06-07 14:47:02	2020-06-07 14:47:02	DEFINER		utf8mb4	utf8mb4_general_ci	latin1_swedish_ci

Fig 5 stored procedures

## Transactions

A transaction is a sequential group of database manipulation operations, which is performed as if it were one single work unit. All the queries take place within a transaction.

```
mysql> create procedure add_drug(  
-> name varchar(255),  
-> description varchar(255),  
-> dosage varchar(255),  
-> disease_id_list longtext,  
-> drug_id_list longtext,  
-> user_added int  
-> )  
-> begin  
-> declare drug_id int;  
-> DECLARE EXIT HANDLER FOR SQLEXCEPTION  
-> BEGIN  
->         ROLLBACK;  
->         SELECT 'An error has occurred, operation rolled back and the stored procedure was terminated';  
-> END;  
-> start transaction;  
-> insert into drugs(name,description,dosage) values(name,description,dosage);  
-> set drug_id=last_insert_id();  
-> insert into drug_authors(did,uid) values(drug_id,user_added);  
-> call insert_drug_cures(drug_id,disease_id_list);  
-> call insert_similar_drugs(drug_id,drug_id_list);  
-> commit;  
-> end  
-> $$
```

Fig 6. add\_drug transaction

# Chapter 3: IMPLEMENTATION

## Tools and Technology

- Front-end (Vuejs + HTML + CSS)

Dependencies:

- Axios - Promise based HTTP client for the browser and node.js
- Vue - Vue is a progressive framework for building user interfaces
- Vue-axios - A small wrapper for integrating axios to Vuejs
- Vue-router - the official router for Vue.js.
- Vue-star-rating - A simple, highly customisable star rating component for Vuejs
- Vuelidate - Simple, lightweight model based validation for Vuejs.
- Vuex - Centralized State management for Vuejs

- Server - (Nodejs)

Libraries:

- Express
- Cors
- Mysql.connector
- Json

- Back End - MySql

## Application Architecture

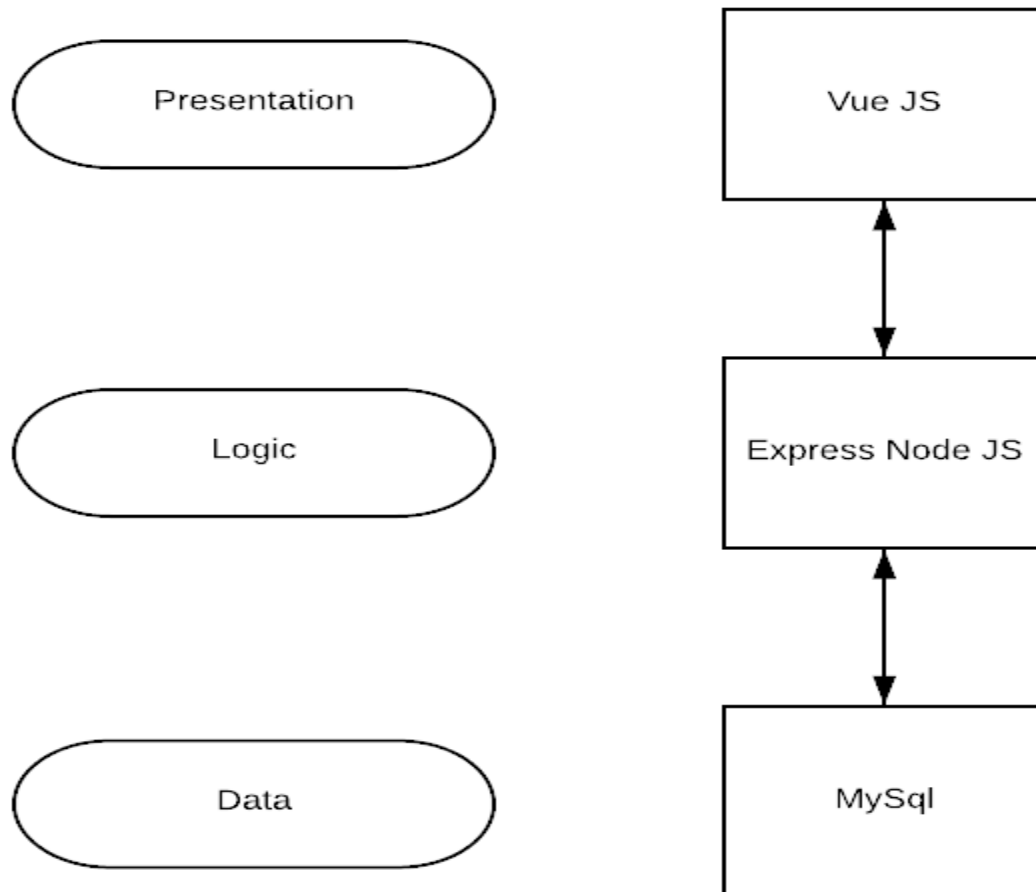
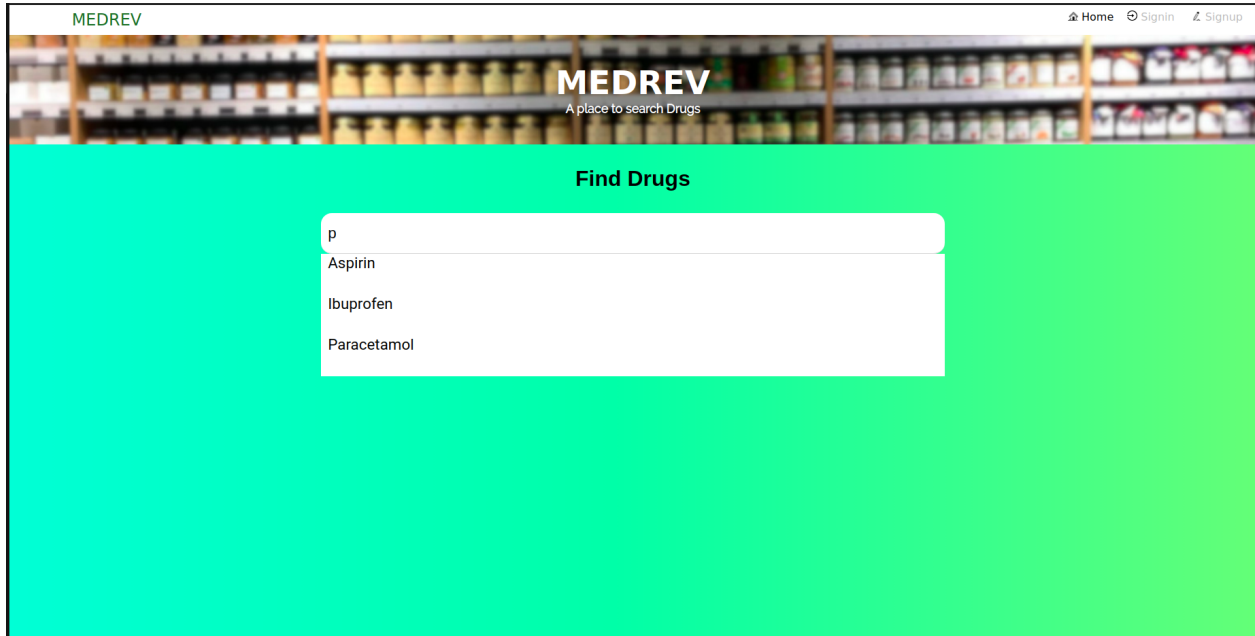


Fig 7 Application Architecture

## Screenshots



**Paracetamol**

3

Added by akshithbellare

**What is Paracetamol?**

Paracetamol is used to treat many conditions such as headache, muscle aches, arthritis, backache, toothaches, colds, and fevers.

**Dosage:** Applies to the following strengths: 160 mg; 80 mg/0.8 mL; 160 mg/5 mL; 500 mg; 650 mg; 80 mg;

**Similar Drugs**

Aspirin

Ibuprofen

**Diseases Cured**

Fever



Submit rating

Enter comment

**A**

Post comment

Add to medication

To add a similar Drug or a disease cured use the search bars to search and click on the disease or drug you want to add

Enter Drug Name

Enter Description

Enter Dosage

Similar Drugs added appear here

Diseases Cured added appear here

Submit

Search Drug

Search Disease



Enter Disease Name

Enter Description

Drugs that treat the diseases appear here

Post Disease

Search Drug

MEDREV A learning project

MEDREV

[Home](#) [@akshithbellare](#) [Add Drug](#) [Add Disease](#) [Logout](#)



akshithbellare

Name Akshith Bellare

Date of Birth 2000-08-14

Blood Group B+

Height 160

Weight 64

Gender M

BMI 64