Project Name: **Farmer Assistance Portal**

Project Members:

PRN No.	Name
210943120012	Amit Singh (Lead)
210943120003	Abhishek Badal
210943120013	Ankit Sharma
210943120048	Mehul Kunte

Technologies Used:

1. **HTML:** Page Layout has been designed in HTML.
2. **CSS:** Used for all designing part**.**
3. **JavaScript:** All Validation task and animations has been developed by JS**.**
4. **JSP:** All front-end logic has been written in JSP**.**
5. **JAVA:** All the business logic has been written in JAVA**.**
6. **MySQL:** MySQL database has been used as database.
7. **Tomcat:** Project will run using Tomcat server**.**

System Requirements:

Software Requirements:

- **Windows (All Versions)**
- **Tomcat 7**
- **VS Code, Eclipse, MySQL, XAMMP**

Hardware Components:

- **Processor – i3**
- **Hard Disk – 5 GB**
- **Memory – 1GB RAM**

Roles and Responsibilities:

Roles And Responsibilities		
1	Role	Handling Frontend and Database
	Member Name	**Amit Singh**
	PRN No	210943120012
	Description	
2	Role	Handling Frontend and Database
	Member Name	**Abhishek Badal**
	PRN No	210943120003
	Description	
3	Role	Handling Backend
	Member Name	**Ankit Sharma**
	PRN No	210943120013
	Description	
4	Role	Documentation & Presentation
	Member Name	**Mehul Kunte**
	PRN No	210943120048
	Description	

Abstract

A Web project to help farmers ensure greater profitability through direct farmer to supplier and farmer to farmer communication. This service boosts business communication and brings transparency in the system.

This innovative site allows for good farmer, retailer and supplier communication. It allows farmers to login and communicate to respective dealers. When dealers publish an advertisement or offer, the respective farmers get notified via SMS message. The farmers may also submit their grievances and complaints to respective dealers or authorities using their farmer login on a separate complaints page and authorities will get access to that page regularly using their login id and passwords.

This web project provides following features:

- Separate login areas with appropriated functionality for farmers, administrators and dealers/ retailers.
- A separate page where only farmers can post complaints and only assigned administrators can read and edit this page.
- Pages where dealers and retailers may post their ads and notifications.
- Farmers are notified of these notifications via SMS whenever new ads are published.
- An effective GUI so that rural people may easily use the service.

Advantages:

- Farmers can directly contact retailers' suppliers by searching online.
- Farmers may submit their grievances online.
- Farmers get notification of any new offers/schemes.

Disadvantages:

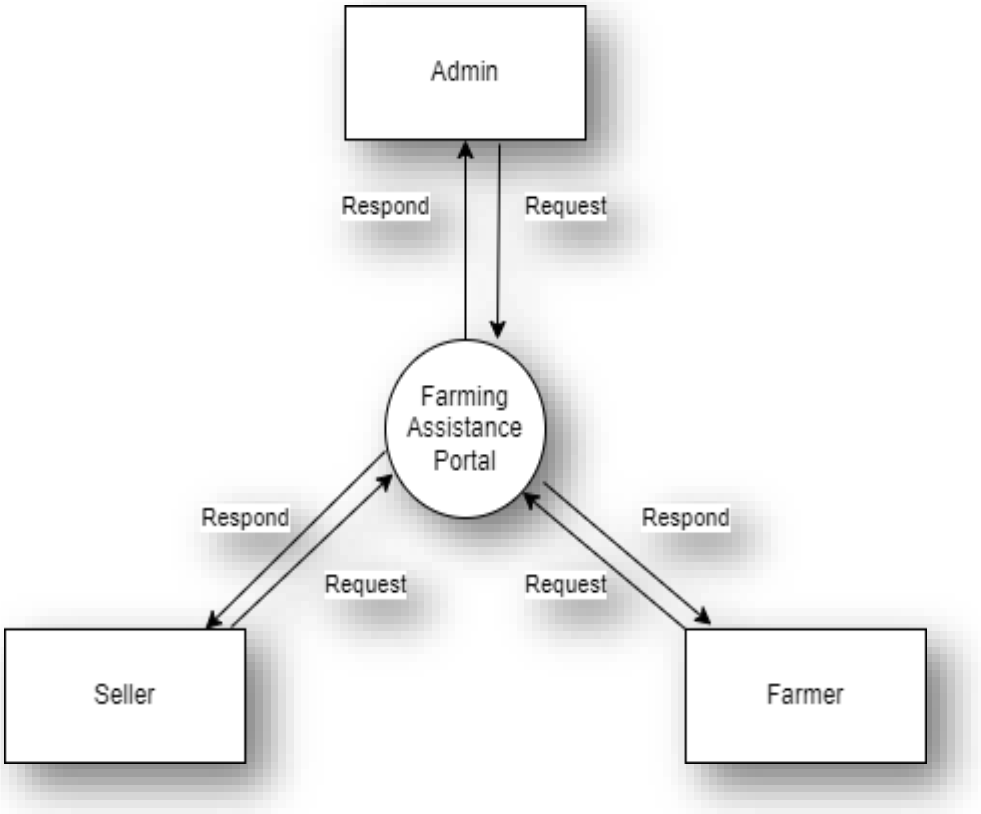
- Verification of the users cannot be done online. They have to first get themselves verified by visiting before using the system.
- It requires a large database and memory.

Applications: This system can be used in developing countries to enhance farmer, manufacturer, retailer communication thus eliminating unnecessary intermediaries.

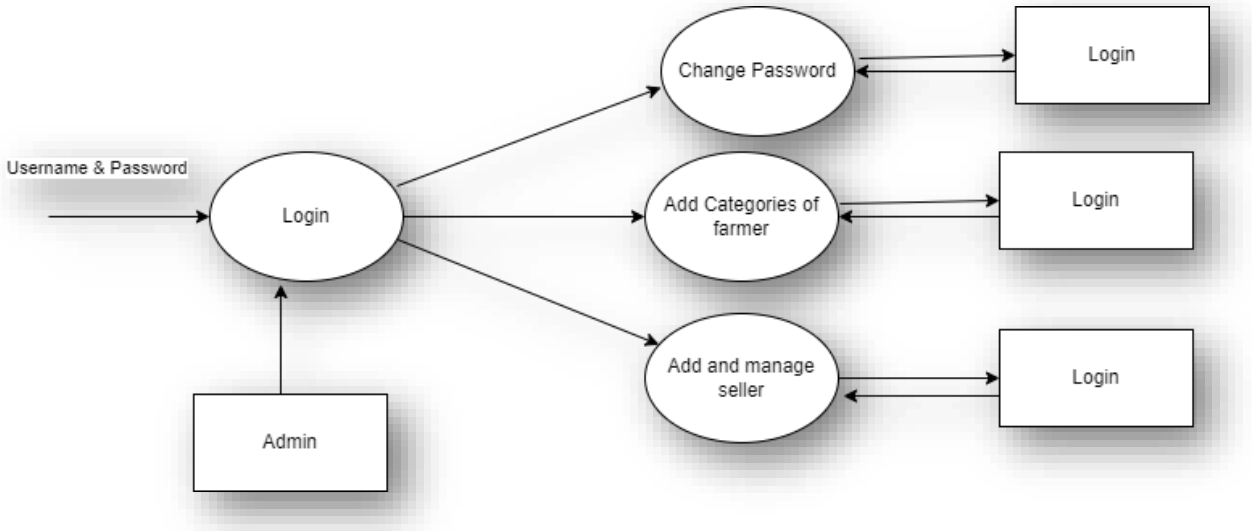
Future Scope

1. Verification of users.
2. Improvement in design.
3. Mobile Application
4. UPI payment
5. Integration of new function like farmers can rent equipment
6. Chat BOT

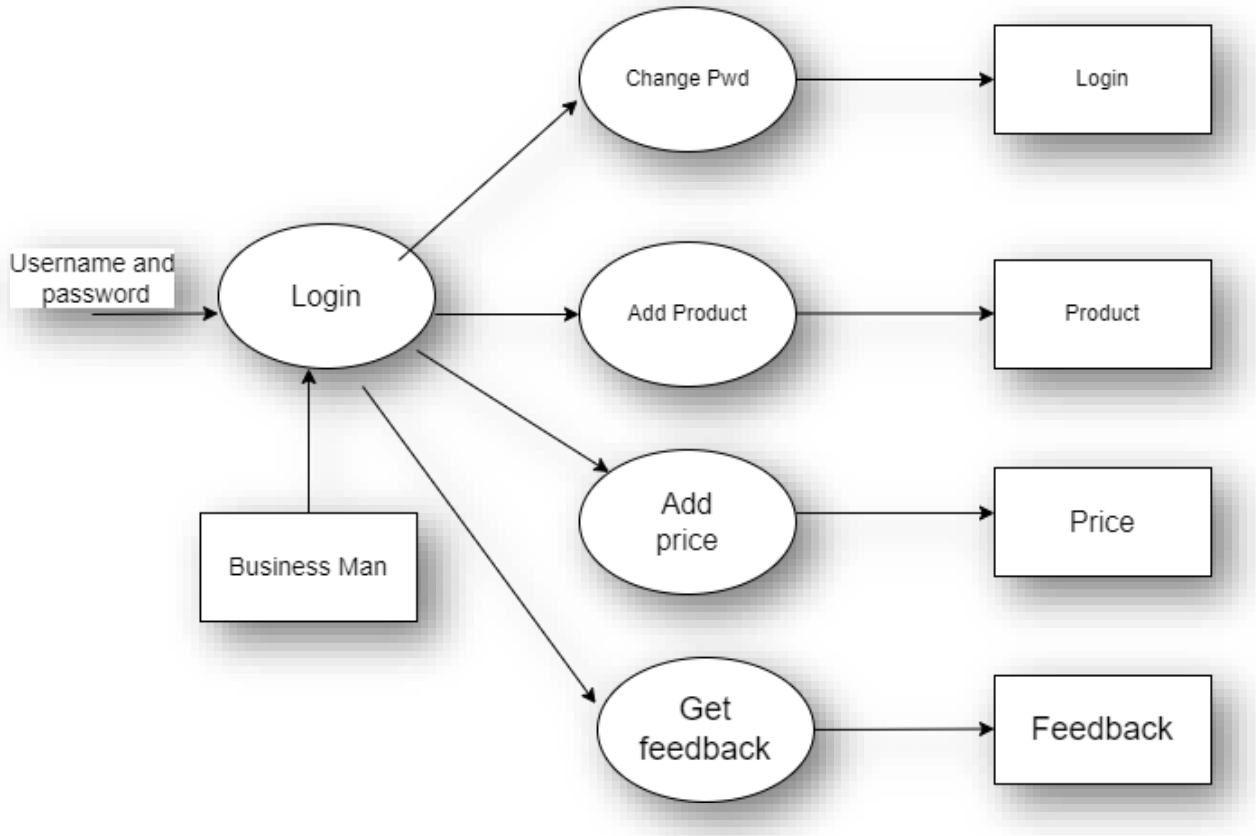
Data Flow Diagram (DFD): Level Zero



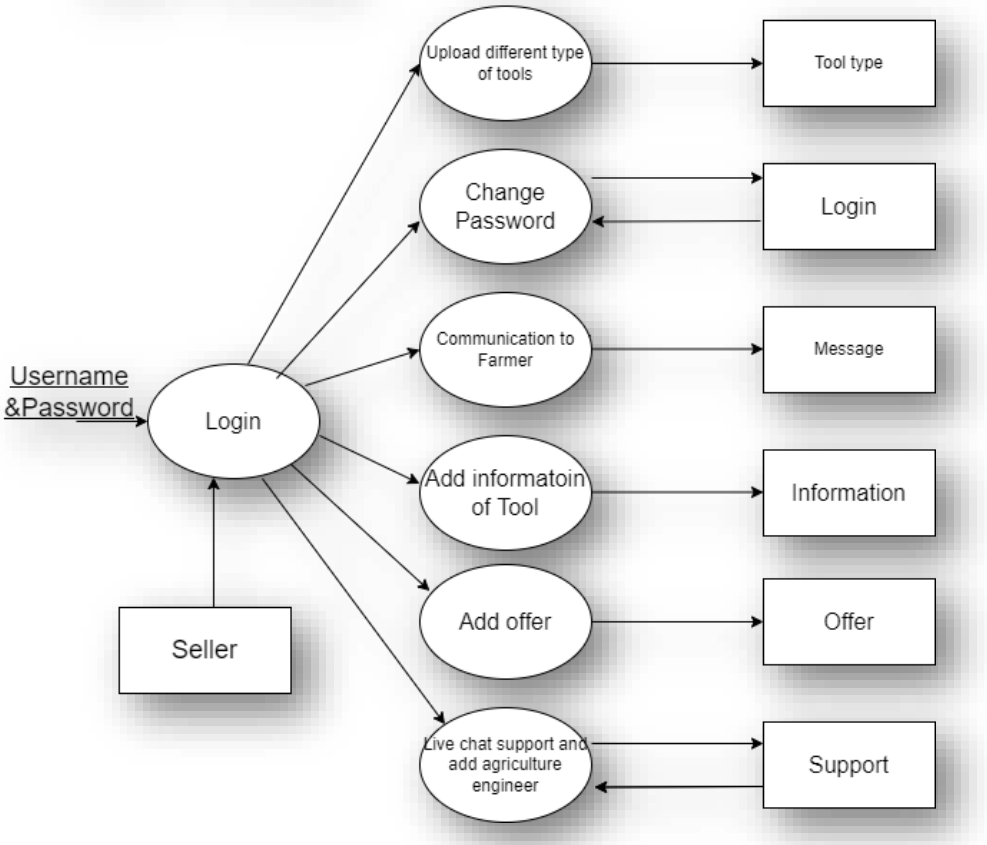
Level 1—



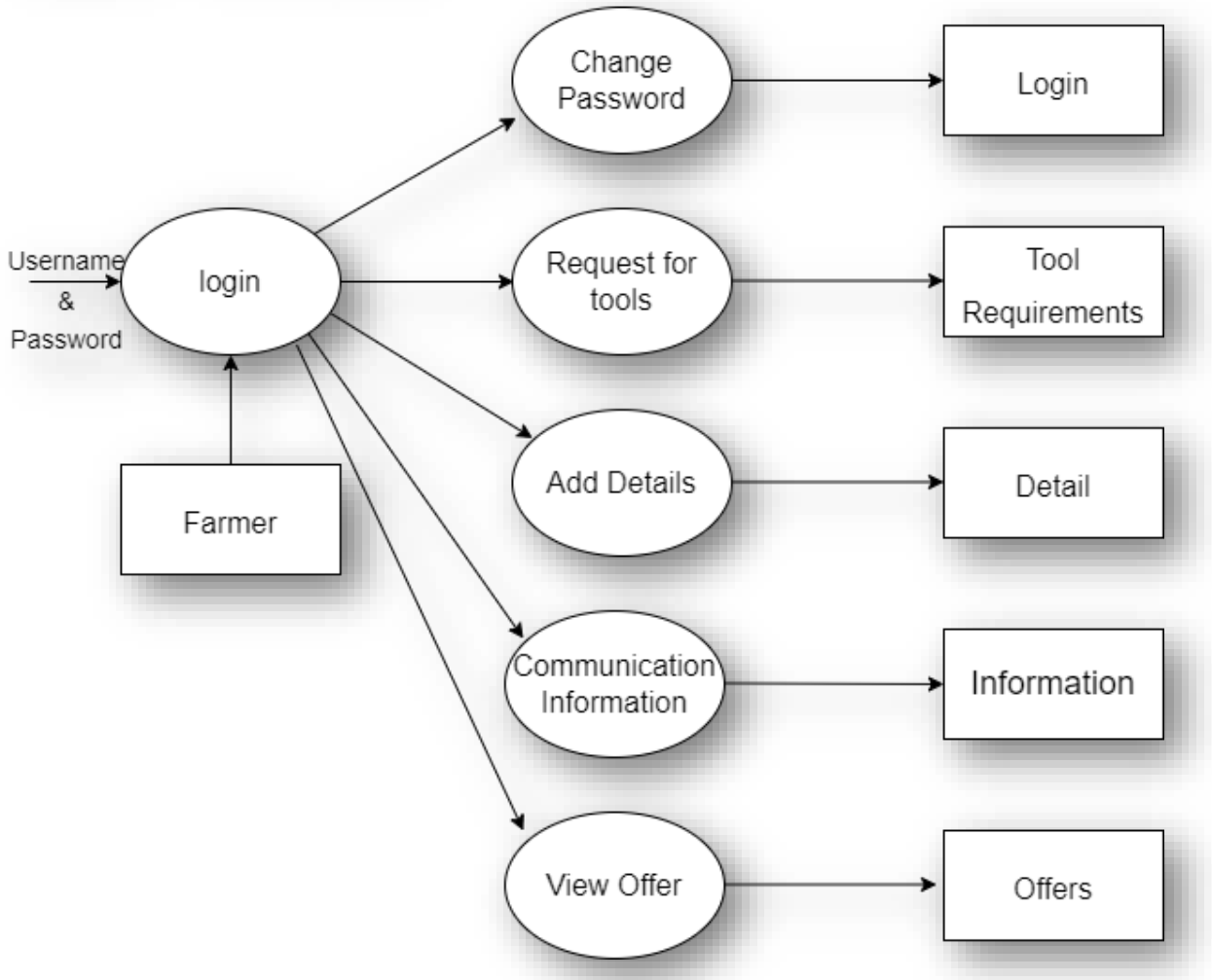
Level 1 Diagram for Business Man



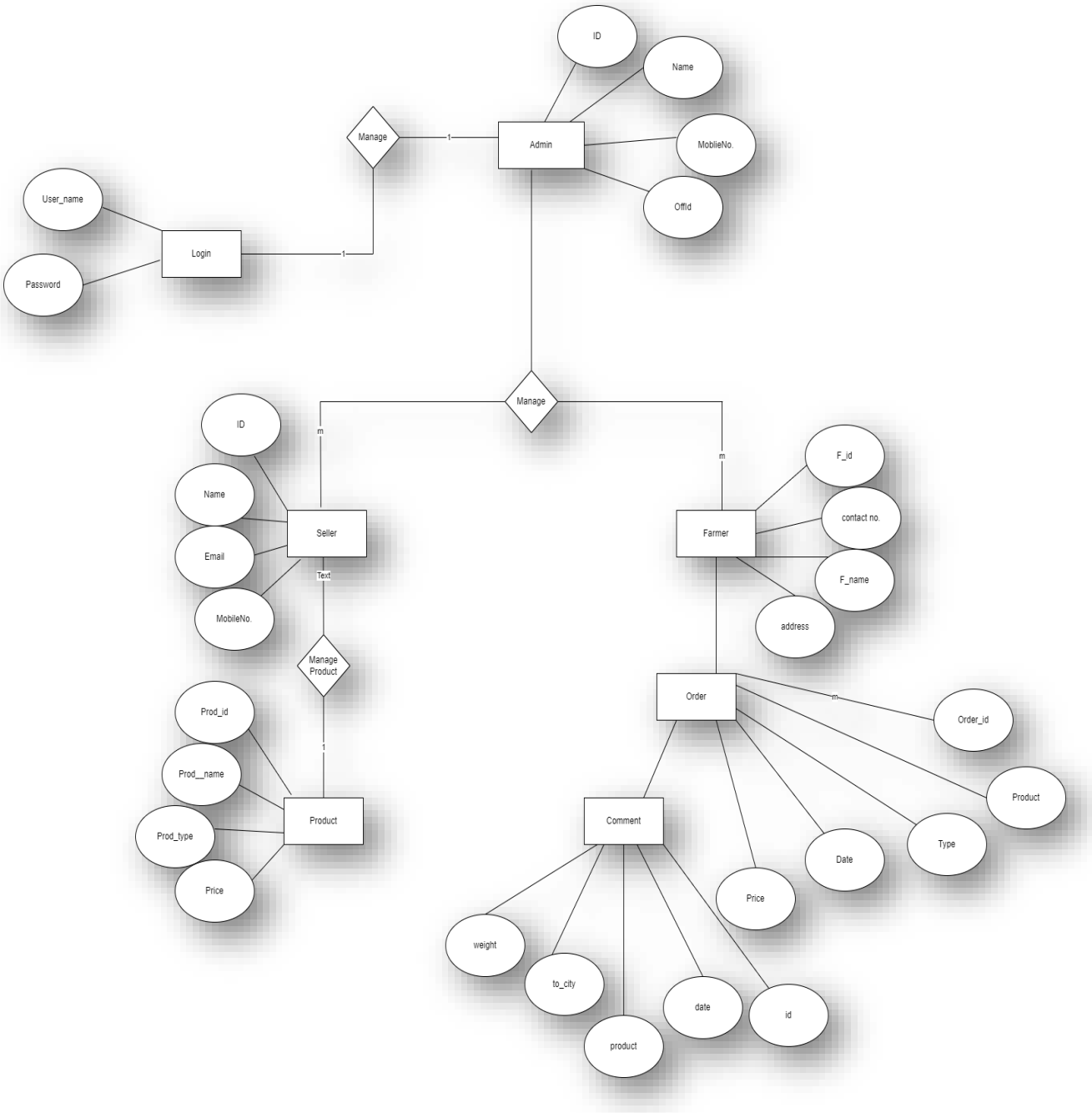
Level 1 For Seller



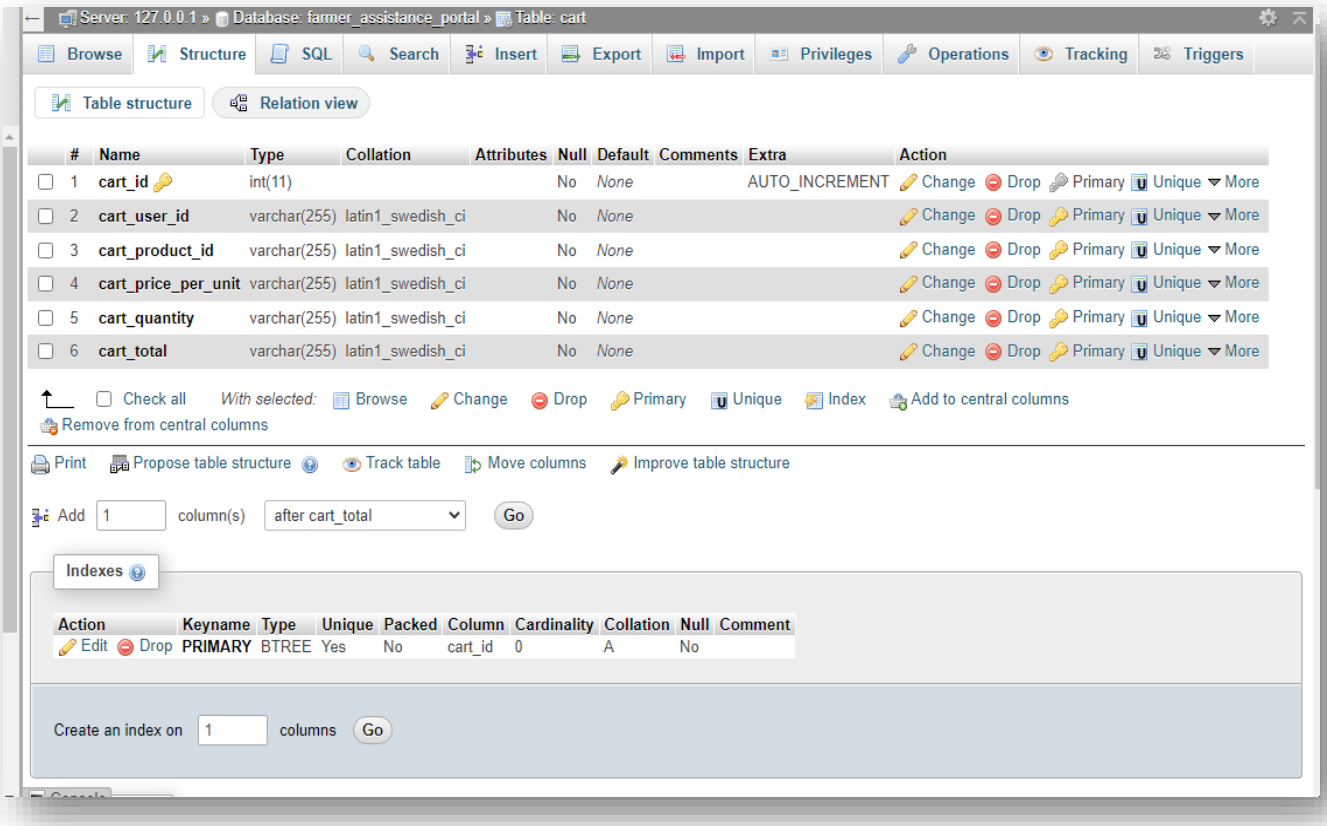
Level1 For Farmer

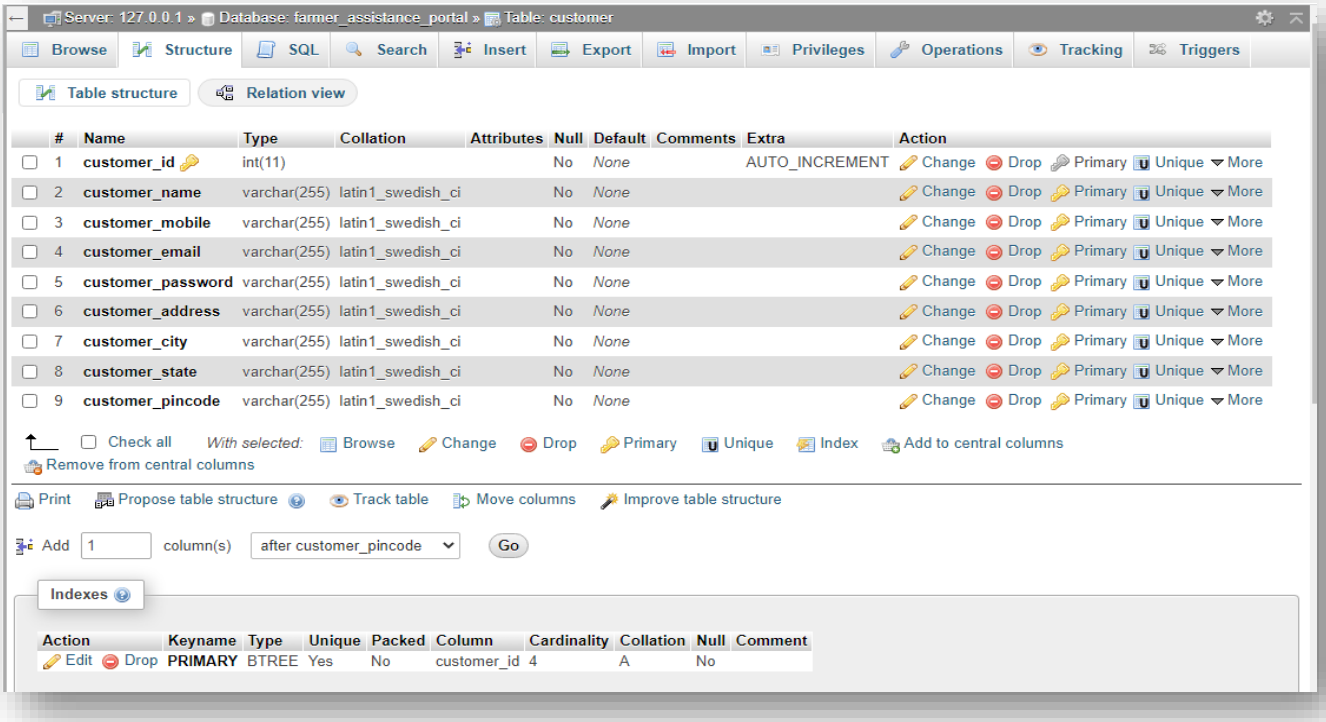


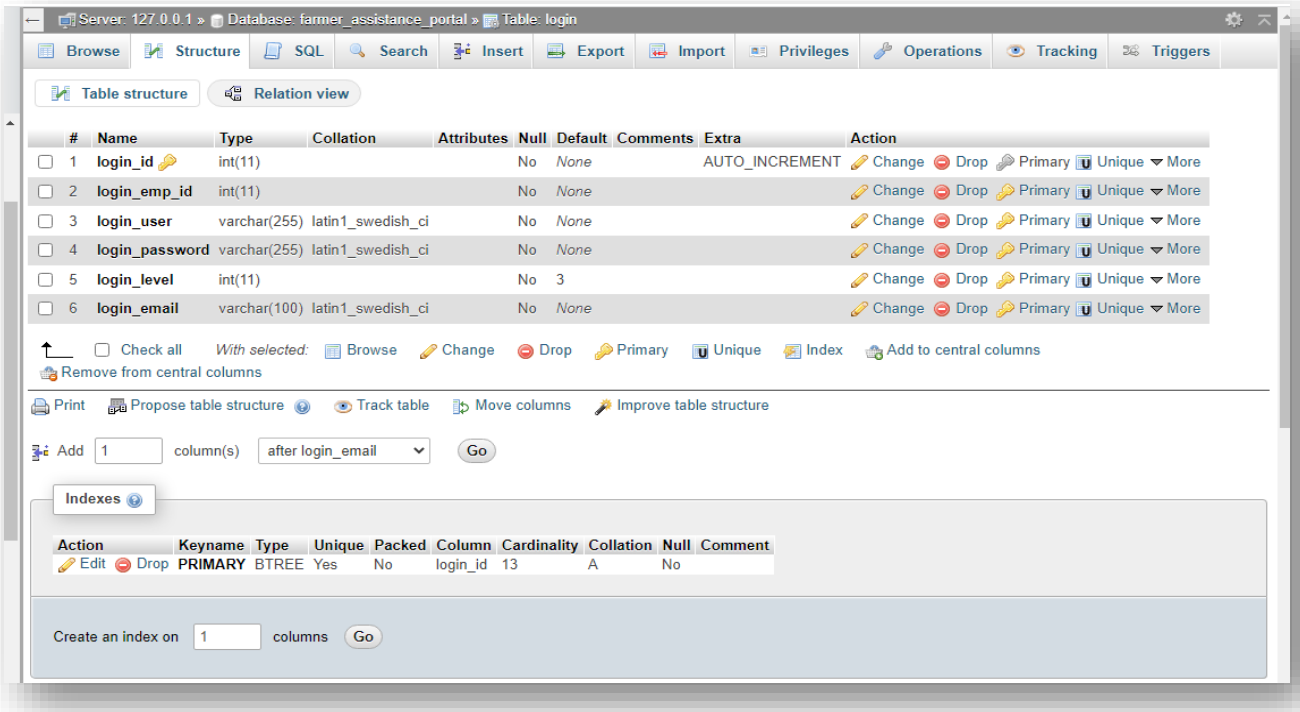
ERD:

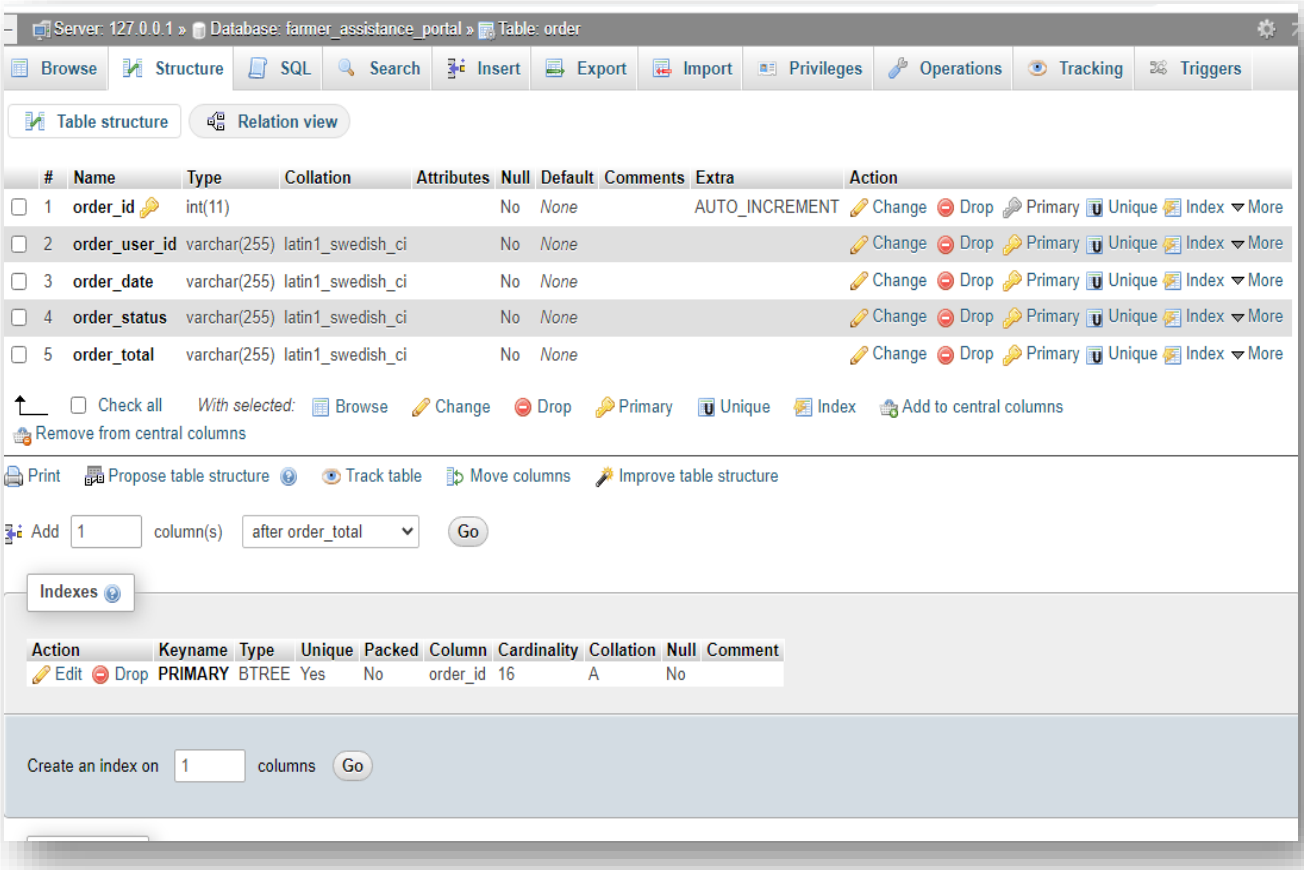

ID
Name
Manage
1
Admin
MoblieNo.
OffId
User_name
Login
1
Password
Manage
ID
m
Name
Seller
Email
Text
MobileNo.
Manage Product
1
Prod_id
Prod__name
Product
Prod_type
Price
m
F_id
contact no.
Farmer
F_name
address
Order
m
Order_id
Product
Comment
Type
Date
Price
weight
to_city
date
id
product

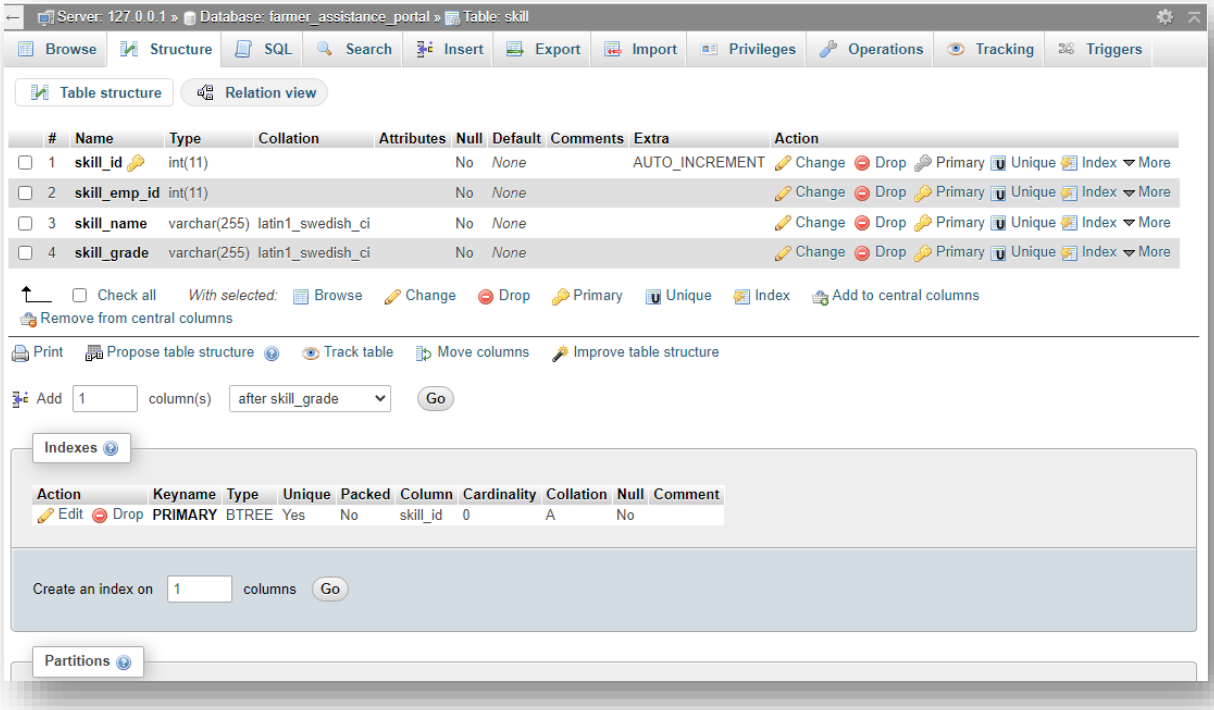
Database Tables:

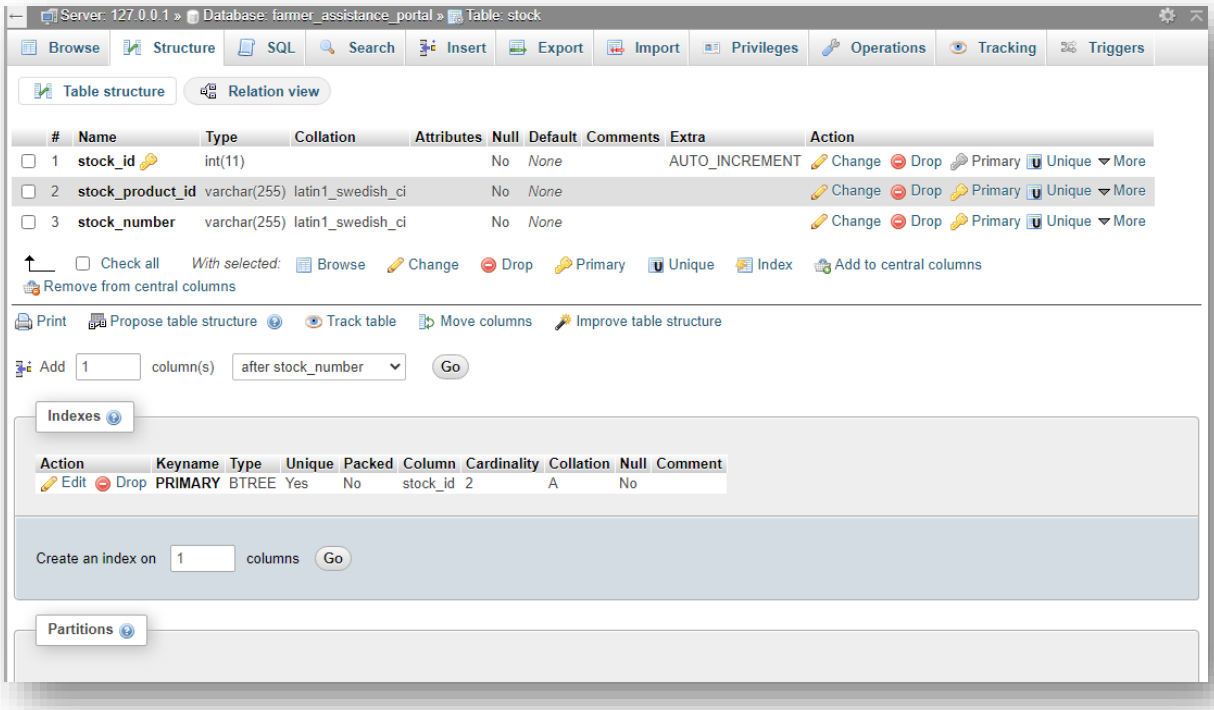


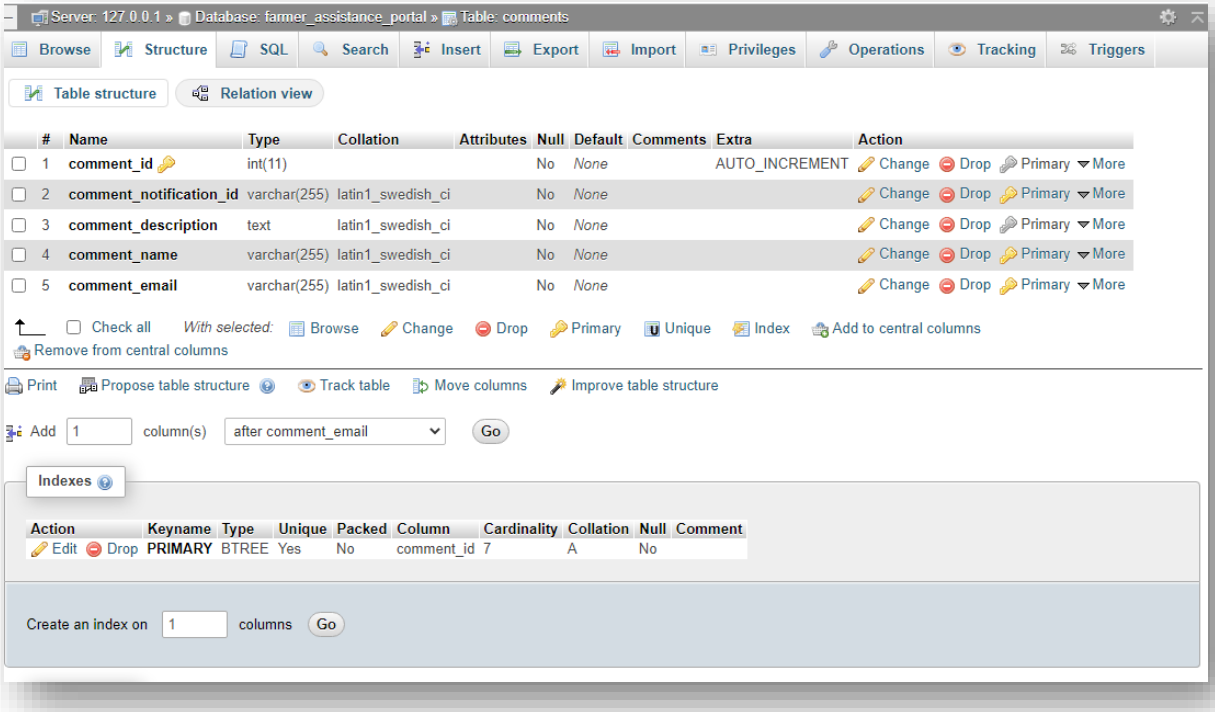


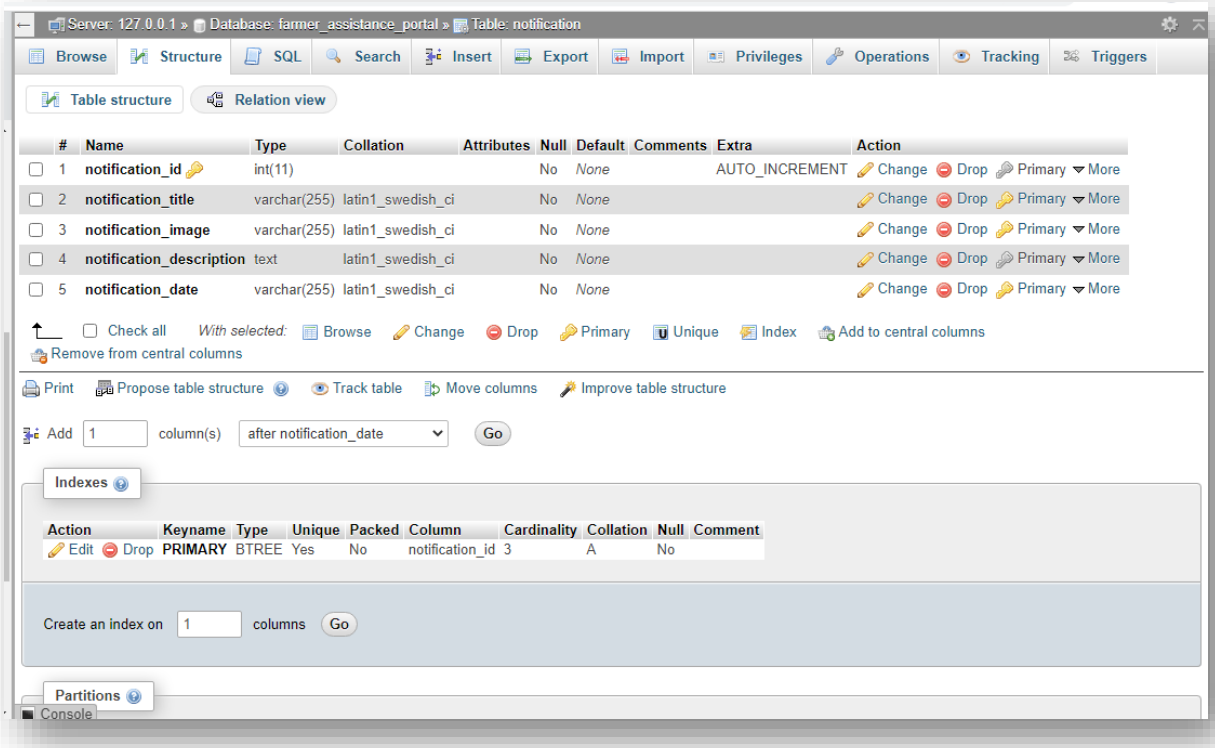

Server: 127.0.0.1 » Database: farmer_assistance_portal » Table: login
Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers
Table structure Relation view
Name Type Collation Attributes Null Default Comments Extra Action
1 login_id int(11) No None AUTO_INCREMENT Change Drop Primary Unique More
2 login_emp_id int(11) No None Change Drop Primary Unique More
3 login_user varchar(255) latin1_swedish_ci No None Change Drop Primary Unique More
4 login_password varchar(255) latin1_swedish_ci No None Change Drop Primary Unique More
5 login_level int(11) No 3 Change Drop Primary Unique More
6 login_email varchar(100) latin1_swedish_ci No None Change Drop Primary Unique More
Check all With selected: Browse Change Drop Primary Unique Index Add to central columns
Remove from central columns
Print Propose table structure Track table Move columns Improve table structure
Add 1 column(s) after login_email Go
Indexes
Action Keyname Type Unique Packed Column Cardinality Collation Null Comment
Edit Drop PRIMARY BTREE Yes No login_id 13 A No
Create an index on 1 columns Go


Server: 127.0.0.1 » Database: farmer_assistance_portal » Table: order
Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers
Table structure Relation view
Name Type Collation Attributes Null Default Comments Extra Action
1 order_id int(11) No None AUTO_INCREMENT Change Drop Primary Unique Index More
2 order_user_id varchar(255) latin1_swedish_ci No None Change Drop Primary Unique Index More
3 order_date varchar(255) latin1_swedish_ci No None Change Drop Primary Unique Index More
4 order_status varchar(255) latin1_swedish_ci No None Change Drop Primary Unique Index More
5 order_total varchar(255) latin1_swedish_ci No None Change Drop Primary Unique Index More
Check all With selected: Browse Change Drop Primary Unique Index Add to central columns
Remove from central columns
Print Propose table structure Track table Move columns Improve table structure
Add 1 column(s) after order_total Go
Indexes
Action Keyname Type Unique Packed Column Cardinality Collation Null Comment
Edit Drop PRIMARY BTREE Yes No order_id 16 A No
Create an index on 1 columns Go


Server: 127.0.0.1 » Database: farmer_assistance_portal » Table: skill
Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers
Table structure Relation view
Name Type Collation Attributes Null Default Comments Extra Action
1 skill_id int(11) No None AUTO_INCREMENT Change Drop Primary Unique Index More
2 skill_emp_id int(11) No None Change Drop Primary Unique Index More
3 skill_name varchar(255) latin1_swedish_ci No None Change Drop Primary Unique Index More
4 skill_grade varchar(255) latin1_swedish_ci No None Change Drop Primary Unique Index More
Check all With selected: Browse Change Drop Primary Unique Index Add to central columns
Remove from central columns
Print Propose table structure Track table Move columns Improve table structure
Add 1 column(s) after skill_grade Go
Indexes
Action Keyname Type Unique Packed Column Cardinality Collation Null Comment
Edit Drop PRIMARY BTREE Yes No skill_id 0 A No
Create an index on 1 columns Go
Partitions


Server: 127.0.0.1 » Database: farmer_assistance_portal » Table: stock
Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers
Table structure Relation view
Name Type Collation Attributes Null Default Comments Extra Action
1 stock_id int(11) No None AUTO_INCREMENT Change Drop Primary Unique More
2 stock_product_id varchar(255) latin1_swedish_ci No None Change Drop Primary Unique More
3 stock_number varchar(255) latin1_swedish_ci No None Change Drop Primary Unique More
Check all With selected: Browse Change Drop Primary Unique Index Add to central columns
Remove from central columns
Print Propose table structure Track table Move columns Improve table structure
Add 1 column(s) after stock_number Go
Indexes
Action Keyname Type Unique Packed Column Cardinality Collation Null Comment
Edit Drop PRIMARY BTREE Yes No stock_id 2 A No
Create an index on 1 columns Go
Partitions


Server: 127.0.0.1 » Database: farmer_assistance_portal » Table: comments
Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers
Table structure Relation view
Name Type Collation Attributes Null Default Comments Extra Action
1 comment_id int(11) No None AUTO_INCREMENT Change Drop Primary More
2 comment_notification_id varchar(255) latin1_swedish_ci No None Change Drop Primary More
3 comment_description text latin1_swedish_ci No None Change Drop Primary More
4 comment_name varchar(255) latin1_swedish_ci No None Change Drop Primary More
5 comment_email varchar(255) latin1_swedish_ci No None Change Drop Primary More
Check all With selected: Browse Change Drop Primary Unique Index Add to central columns
Remove from central columns
Print Propose table structure Track table Move columns Improve table structure
Add 1 column(s) after comment_email Go
Indexes
Action Keyname Type Unique Packed Column Cardinality Collation Null Comment
Edit Drop PRIMARY BTREE Yes No comment_id 7 A No
Create an index on 1 columns Go


Server: 127.0.0.1 » Database: farmer_assistance_portal » Table: notification
Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers
Table structure Relation view
Name Type Collation Attributes Null Default Comments Extra Action
1 notification_id int(11) No None AUTO_INCREMENT Change Drop Primary More
2 notification_title varchar(255) latin1_swedish_ci No None Change Drop Primary More
3 notification_image varchar(255) latin1_swedish_ci No None Change Drop Primary More
4 notification_description text latin1_swedish_ci No None Change Drop Primary More
5 notification_date varchar(255) latin1_swedish_ci No None Change Drop Primary More
Check all With selected: Browse Change Drop Primary Unique Index Add to central columns
Remove from central columns
Print Propose table structure Track table Move columns Improve table structure
Add 1 column(s) after notification_date Go
Indexes
Action Keyname Type Unique Packed Column Cardinality Collation Null Comment
Edit Drop PRIMARY BTREE Yes No notification_id 3 A No
Create an index on 1 columns Go
Partitions
Console

Folder Structure Explanation

In CSS folder we have kept our third-party library files, while in images folder there are images that we have used in the frontend. In include folder we have placed the code which are used in more than one page. In JS folder we have kept the java script files which can also be considered as third-party library. In layout we kept the theme related file which we have used in our project. Our model folder acts as a mediator between JSP files and java class files. At the same time, it will communicate between the front end and the database. In notification and product images folder all the images used in this project is kept. In Web-Inf folder all business logic, public methods and database connection code is written. Rest all JSP files are presentation files which will be displayed when required.

We have integrated tomcat server for smooth run of our project. The product is up and running with all modules.