MUSHROMIEE: A DIGITAL MUSHROOM BOOKING SYSTEM

A DATABASE-DRIVEN AGRI-ENTREPRENEURSHIP SYSTEM

INTRODUCTION TO MUSHROOMIEE

BUSINESS BACKGROUND

- UiTM staff-owned mushroom business
- Established in 2022
- Partnered with UiTM Machang
- Combines agriculture with digital technology

PROBLEM STATEMENT

- Manual booking caused errors & delays
- Poor inventory control
- Hard to track customers or payments
- Business could not scale

SYSTEM OVERVIEW

- Web-based booking system (PHP + MySQL)
- Central database for orders, products, users
- Real-time inventory & payment updates
- Admin dashboard for management

OBJECTIVES

- Automate booking and inventory
- Centralize customer data
- Enable smooth order tracking
- Improve operational efficiency
- Prepare for future scalability

BUSINESS RULES

📌 **Key Business Rules:**

1. One user can make many orders
→ 1:M relationship between users and orders
→ No order without user

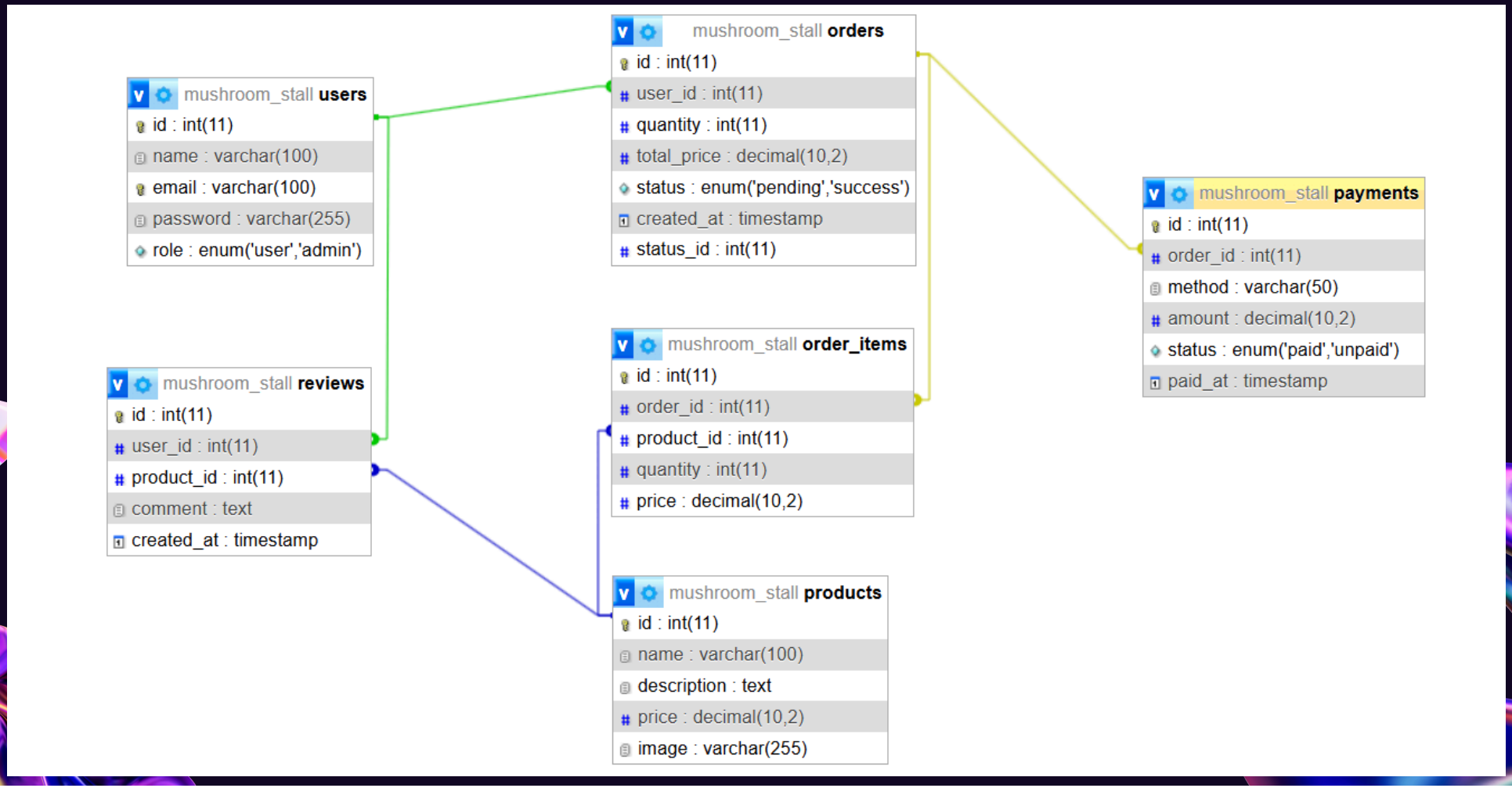
2. One order can have many products
→ M:N via order_items
→ One product can be in many orders

3. One payment belongs to one order
→ Foreign key from payments to orders
→ Payment only used once

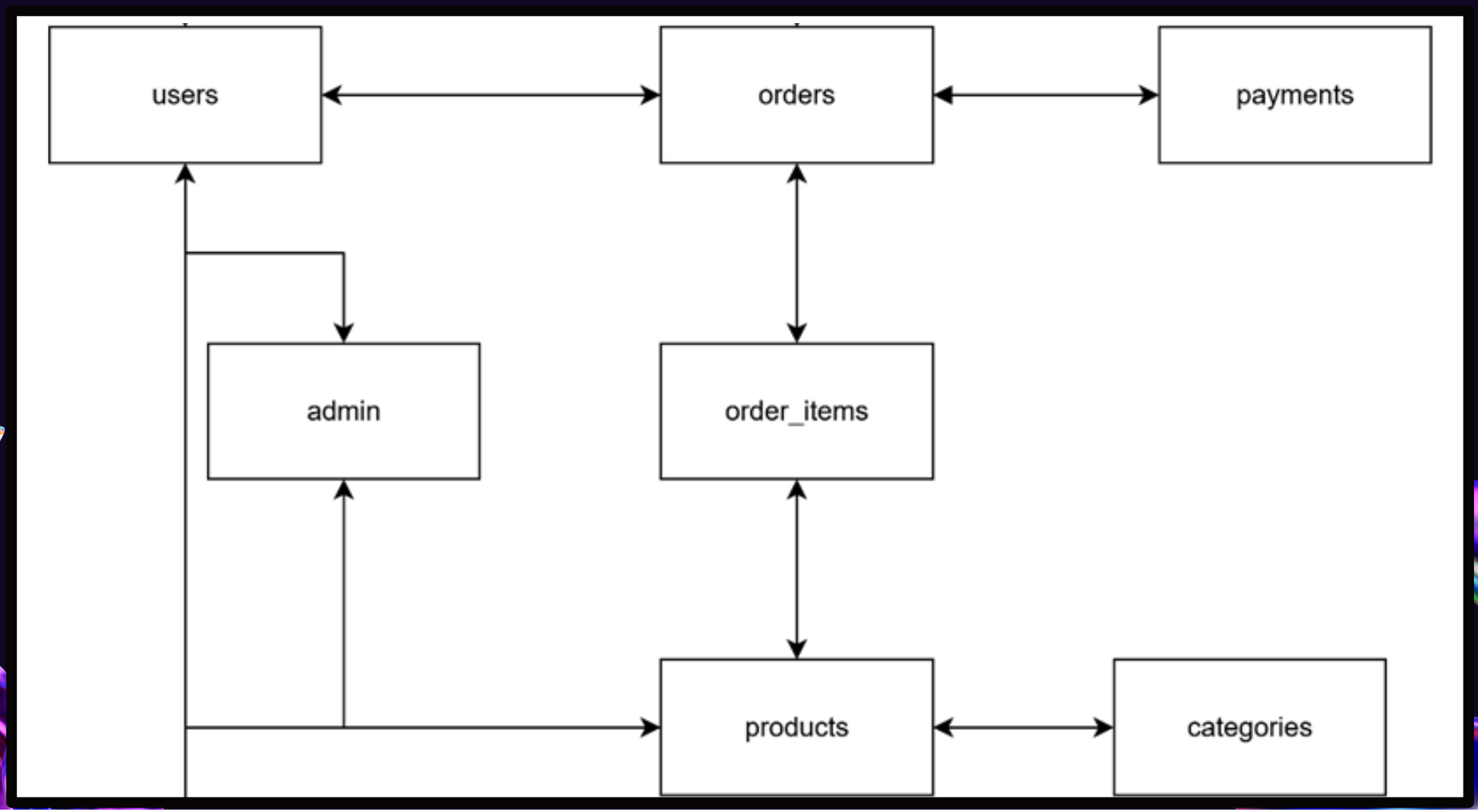
✅ **Why It's Important:**

- Make sure data is correct
- Follow the real business process
- Make system easy to understand
- Avoid wrong data and function problem

ENTITIES RELATIONSHIP DIAGRAM



CONCEPTUAL ENTITIES RELATIONSHIP DIAGRAM



NORMALIZATION

📌 Unnormalized Form (UNF) :

The data is stored in one big table that contain everything together. It not follow any rule of database normalization.

Example:

order_id →
user_id, username, user_email,
product_id, product_name, product_desc,
product_price, product_image,
quantity, total_price,
payment_method, payment_amount,
payment_status, paid_at,
order_status, created_at,
review_comment, review_date

📌 First Normal Form (1NF) :

1NF is the first rule we follow when fixing unnormalized data. In 1NF, we make sure that each column have only atomic values. That means no list or group of data inside one field. Also, each record must be unique and every field should have same type of value.

Example:
users
user_id → username, user_email
products
product_id → product_name, product_desc,
product_price, product_image
orders
order_id → user_id, total_price, order_status,
created_at
order_items

NORMALIZATION

📌 Second Normal Form (2NF) :

(2NF) improve on 1NF by removing partial dependency. Partial dependency happen when a table have a composite key (like order_id and product_id together), and some column only depend on part of the key.

Example: table_items

order_id + product_id → quantity,
product_name, product_price,
product_image

📌 Third Normal Form Form (3NF) :

3NF is the final step of basic normalization. It remove transitive dependency. This is when a non-key column depends on another non-key column. In 3NF, every non-key column must only depend on the primary key.

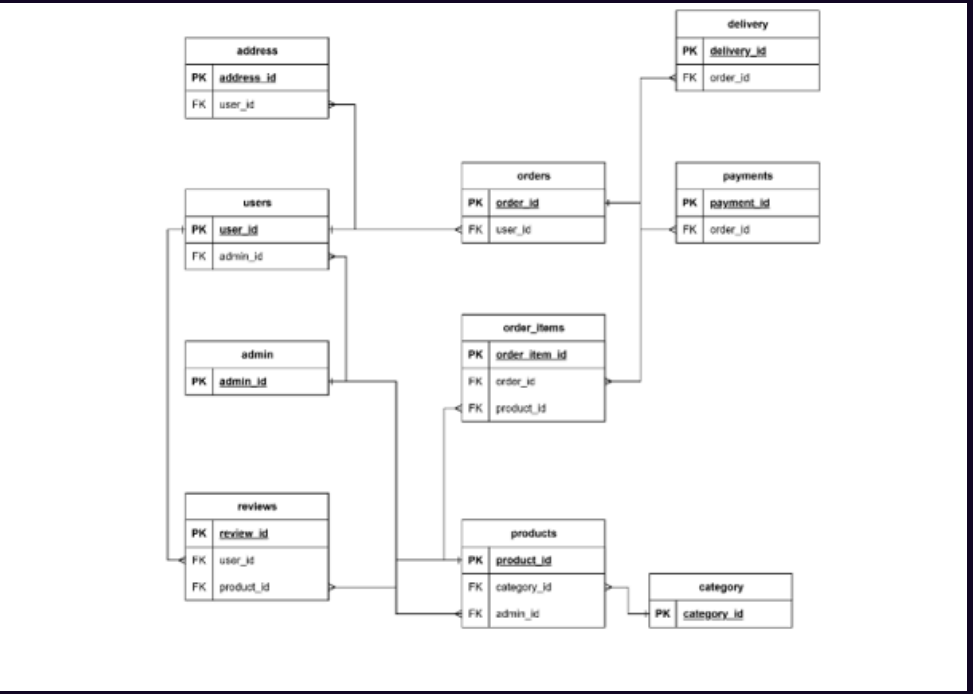
Example:

user_id → name, email, password, role
product_id → name, description, price, image
order_id → user_id, total_price, status, created_at, status_id
order_item_id → order_id, product_id, quantity, price
payment_id → order_id, method, amount, status, paid_at
review_id → user_id, product_id, comment, created_at

RELATIONAL DIAGRAM

Each table in the diagram have their own fields, and it show the primary key (PK) and also the foreign key (FK) clearly.

Relational diagram show 10 table including users, products, orders, reviews, payments, and more. For example, one user can make many orders, so the users table have one-to-many relationship with the orders table. Also, each order can have many items, so order_items table is connect to both orders and products.



DATA DICTIONARY

Data dictionary is one of the important part in database system because it help to explain all the tables and field inside the database

For example, in mushroom stall system, we have many table like users, orders, products and more. Each table have their own column that store different type of data. With data dictionary, we can know easily which one is primary key (PK), which one is foreign key (FK), and what each column is doing in the system.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	int(11)			No	None		AUTO_INCREMENT
2	name	varchar(100)	utf8mb4_general_ci		No	None		
3	email	varchar(100)	utf8mb4_general_ci		No	None		
4	password	varchar(255)	utf8mb4_general_ci		No	None		
5	role	enum('user', 'admin')	utf8mb4_general_ci		Yes	user		

users

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	int(11)			No	None		AUTO_INCREMENT
2	user_id	int(11)			Yes	NULL		
3	quantity	int(11)			Yes	NULL		
4	total_price	decimal(10,2)			No	None		
5	status	enum('pending', 'success')	utf8mb4_general_ci		Yes	pending		
6	created_at	timestamp			No	current_timestamp()		
7	status_id	int(11)			Yes	NULL		

orders

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	int(11)			No	None		AUTO_INCREMENT
2	user_id	int(11)			Yes	NULL		
3	product_id	int(11)			Yes	NULL		
4	comment	text	utf8mb4_general_ci		Yes	NULL		
5	created_at	timestamp			No	current_timestamp()		

reviews

DATA DICTIONARY

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	int(11)			No	None		AUTO_INCREMENT
2	name	varchar(100)	utf8mb4_general_ci		No	None		
3	description	text	utf8mb4_general_ci		No	None		
4	price	decimal(10,2)			No	None		
5	image	varchar(255)	utf8mb4_general_ci		No	None		

Products

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	int(11)			No	None		AUTO_INCREMENT
2	order_id	int(11)			No	None		
3	method	varchar(50)	utf8mb4_general_ci		No	None		
4	amount	decimal(10,2)			No	None		
5	status	enum('paid', 'unpaid')	utf8mb4_general_ci		Yes	unpaid		
6	paid_at	timestamp			No	current_timestamp()		ON UPDATE CURRENT_TIMESTAMP()

payments

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	int(11)			No	None		AUTO_INCREMENT
2	order_id	int(11)			No	None		
3	product_id	int(11)			No	None		
4	quantity	int(11)			No	None		
5	price	decimal(10,2)			No	None		

Order_items

LIMITATION & FUTURE ENHANCEMENT

📌**LIMITATION:**

- The system only focus on basic function
- The review system also not complete

📌**FUTURE ENHANCEMENT :**

- This website can improve by adding better user interface that is mobile friendly
- Live chat or support form will make customer service

CONCLUSION

- This system will be specifically designed to support Mashromie in supporting online sales and related management.