

# Crowdfunding Platform



## Case Study

**Prepared by :**

**Meet Vasita**

# **Introduction to Crowdfunding :**

Crowdfunding is a way of raising money from a large number of people, usually through the internet, to support a project, business, or cause. Instead of getting money from one big investor, people collect small amounts from many individuals who believe in the idea. There are different types of crowdfunding, such as donation-based (where people give money without expecting anything in return), reward-based (where supporters get a small gift or product), and equity-based (where backers receive shares in a company). Crowdfunding helps startups, artists, and charities turn their ideas into reality by gathering support from the public.

## **Objective of the project :**

1. Allow users to create projects and receive funds from supporters through pledges and rewards.
2. Track all transactions and payment statuses to ensure money is processed securely.
3. Allow people to support projects they like and stay updated on progress through comments, updates, and notifications.

## SQL Schema Code :



```
1 CREATE TABLE Users (  
2     user_id INT PRIMARY KEY AUTO_INCREMENT,  
3     name VARCHAR(100) NOT NULL,  
4     email VARCHAR(100) UNIQUE NOT NULL,  
5     password VARCHAR(255) NOT NULL,  
6     role ENUM('backer', 'creator', 'admin') NOT NULL,  
7     created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
8 );
```



```
1 CREATE TABLE Categories (  
2     category_id INT PRIMARY KEY AUTO_INCREMENT,  
3     category_name VARCHAR(100) UNIQUE NOT NULL  
4 );
```



```
1 CREATE TABLE Projects (  
2     project_id INT PRIMARY KEY AUTO_INCREMENT,  
3     creator_id INT NOT NULL,  
4     title VARCHAR(255) NOT NULL,  
5     description TEXT NOT NULL,  
6     category_id INT NOT NULL,  
7     goal_amount DECIMAL(10, 2) NOT NULL,  
8     start_date DATE NOT NULL,  
9     end_date DATE NOT NULL,  
10    status ENUM('ongoing', 'successful', 'failed') NOT NULL,  
11    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
12    FOREIGN KEY (creator_id) REFERENCES Users(user_id),  
13    FOREIGN KEY (category_id) REFERENCES Categories(category_id)  
14 );
```



```
1 CREATE TABLE Pledges (  
2     pledge_id INT PRIMARY KEY AUTO_INCREMENT,  
3     project_id INT NOT NULL,  
4     backer_id INT NOT NULL,  
5     amount DECIMAL(10, 2) NOT NULL,  
6     pledge_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
7     FOREIGN KEY (project_id) REFERENCES Projects(project_id),  
8     FOREIGN KEY (backer_id) REFERENCES Users(user_id)  
9 );
```




```
1 CREATE TABLE Rewards (  
2     reward_id INT PRIMARY KEY AUTO_INCREMENT,  
3     project_id INT NOT NULL,  
4     reward_title VARCHAR(255) NOT NULL,  
5     reward_description TEXT NOT NULL,  
6     min_pledge_amount DECIMAL(10, 2) NOT NULL,  
7     FOREIGN KEY (project_id) REFERENCES Projects(project_id)  
8 );
```




```
1 CREATE TABLE Comments (  
2     comment_id INT PRIMARY KEY AUTO_INCREMENT,  
3     project_id INT NOT NULL,  
4     user_id INT NOT NULL,  
5     comment_text TEXT NOT NULL,  
6     comment_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
7     FOREIGN KEY (project_id) REFERENCES Projects(project_id),  
8     FOREIGN KEY (user_id) REFERENCES Users(user_id)  
9 );
```




```
1 CREATE TABLE Updates (  
2     update_id INT PRIMARY KEY AUTO_INCREMENT,  
3     project_id INT NOT NULL,  
4     update_title VARCHAR(255) NOT NULL,  
5     update_content TEXT NOT NULL,  
6     update_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
7     FOREIGN KEY (project_id) REFERENCES Projects(project_id)  
8 );
```



```
1 CREATE TABLE Transactions (  
2     transaction_id INT PRIMARY KEY AUTO_INCREMENT,  
3     pledge_id INT NOT NULL,  
4     transaction_amount DECIMAL(10, 2) NOT NULL,  
5     transaction_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
6     payment_status ENUM('pending', 'completed', 'failed') NOT NULL,  
7     payment_type ENUM('UPI', 'Credit Card', 'Debit Card', 'Net Banking', 'Crypto') NOT NULL,  
8     FOREIGN KEY (pledge_id) REFERENCES Pledges(pledge_id)  
9 );
```

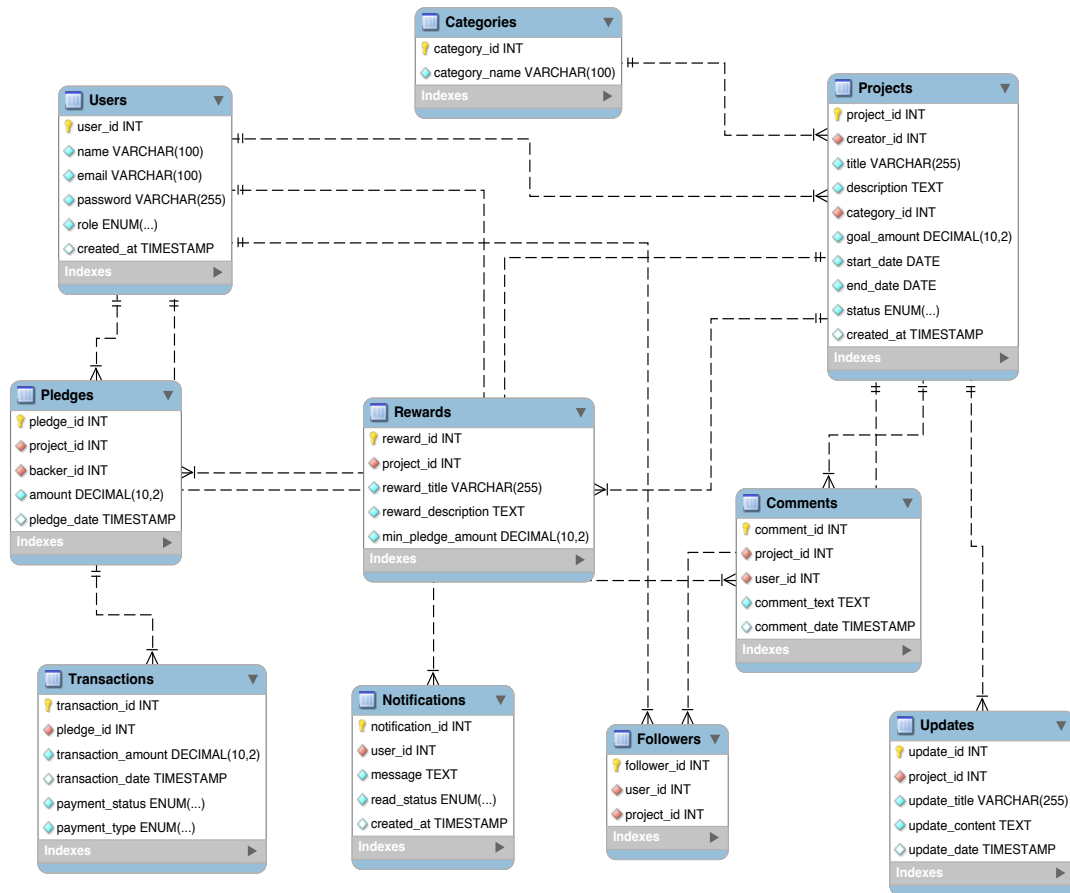


```
1 CREATE TABLE Notifications (  
2     notification_id INT PRIMARY KEY AUTO_INCREMENT,  
3     user_id INT NOT NULL,  
4     message TEXT NOT NULL,  
5     read_status ENUM('unread', 'read') NOT NULL,  
6     created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
7     FOREIGN KEY (user_id) REFERENCES Users(user_id)  
8 );
```



```
1 CREATE TABLE Followers (  
2     follower_id INT PRIMARY KEY AUTO_INCREMENT,  
3     user_id INT NOT NULL,  
4     project_id INT NOT NULL,  
5     FOREIGN KEY (user_id) REFERENCES Users(user_id),  
6     FOREIGN KEY (project_id) REFERENCES Projects(project_id)  
7 );
```

# ER Diagram :



## **Business Insights :**

1. **User Insights** – Track user growth by checking when they joined, see who engages more (backers, creators, admins), and find top creators by counting their successful projects.
2. **Project Insights** – Check how many projects succeed, find the most popular categories, calculate average project duration, and compare the highest-funded projects.
3. **Pledge & Funding Insights** – Sum up total funding, find the average pledge amount, identify repeat backers, and see who donates the most.
4. **Reward Insights** – Find the most popular rewards, and analyze how backers choose different reward tiers.
5. **Transaction Insights** – Compare successful and failed payments, find the most used payment method, and calculate total revenue from successful payments.