CCGC5003

Python Project Phase 2

Suraj Mandal

Venkata Narasimha Vedavyas Muppavarapu

03/20/2023

TO-DO APPLICATION

# Database Design

## Tables and Fields:

|  |  |
| --- | --- |
| **Tables** | **Fields** |
| **Users**: stores user information, such as username, email, password. | id INT(11) ,name VARCHAR(50), email VARCHAR(50), password VARCHAR(255), profile\_picture VARCHAR(255), UNIQUE KEY (email) |
| **Messages**: This table will store direct messages sent between users. It will include the sender and receiver's user ID, the message content, and the timestamp. | id INT(11), sender\_id INT(11), receiver\_id INT(11) , content TEXT, timestamp DATETIME |
| **Groups**: This table will store information about each group, including the group name and description. | id INT(11),name VARCHAR(50), description VARCHAR(255) |
| **Groups Members**: This table stores the user's task groups, including task group ID, user ID, task group name, and description. | id INT(11), group\_id INT(11), user\_id INT(11) |
| **Announcements:** This table will store announcements created by admins. It will include the announcement content and the timestamp. | id INT(11) ,content TEXT , timestamp DATETIME |

## Tables and Usage

|  |  |
| --- | --- |
| **Table** | **Usage** |
| Users | This table will store user information, including their name, email, password and date of birth. |
| Messages | This table will store direct messages sent between users. It will include the sender and receiver's user ID, the message content, and the timestamp. |
| Groups | This table will store information about each group, including the group name and description. |
| GroupMembers | This table will store the relationship between groups and their members. It will include the group ID |
| Announcements | This table will store announcements created by admins. It will include the announcement content and the timestamp. |

An SQL script is created as below for each table with necessary data types as shown below,

-CREATE TABLE Users ( id INT(11) NOT NULL AUTO\_INCREMENT,

name VARCHAR(50) NOT NULL,

email VARCHAR(50) NOT NULL,

password VARCHAR(255) NOT NULL,

profile\_picture VARCHAR(255),

PRIMARY KEY (id),

UNIQUE KEY (email) );

- CREATE TABLE Messages ( id INT(11) NOT NULL AUTO\_INCREMENT,

sender\_id INT(11) NOT NULL,

receiver\_id INT(11) NOT NULL,

content TEXT NOT NULL,

timestamp DATETIME NOT NULL DEFAULT CURRENT\_TIMESTAMP,

PRIMARY KEY (id),

FOREIGN KEY (sender\_id) REFERENCES Users(id),

FOREIGN KEY (receiver\_id) REFERENCES Users(id) );

-CREATE TABLE Groups ( id INT(11) NOT NULL AUTO\_INCREMENT,

name VARCHAR(50) NOT NULL,

description VARCHAR(255) NOT NULL,

PRIMARY KEY (id) );

-CREATE TABLE GroupMembers ( id INT(11) NOT NULL AUTO\_INCREMENT,

group\_id INT(11) NOT NULL,

user\_id INT(11) NOT NULL,

PRIMARY KEY (id),

FOREIGN KEY (group\_id) REFERENCES Groups(id),

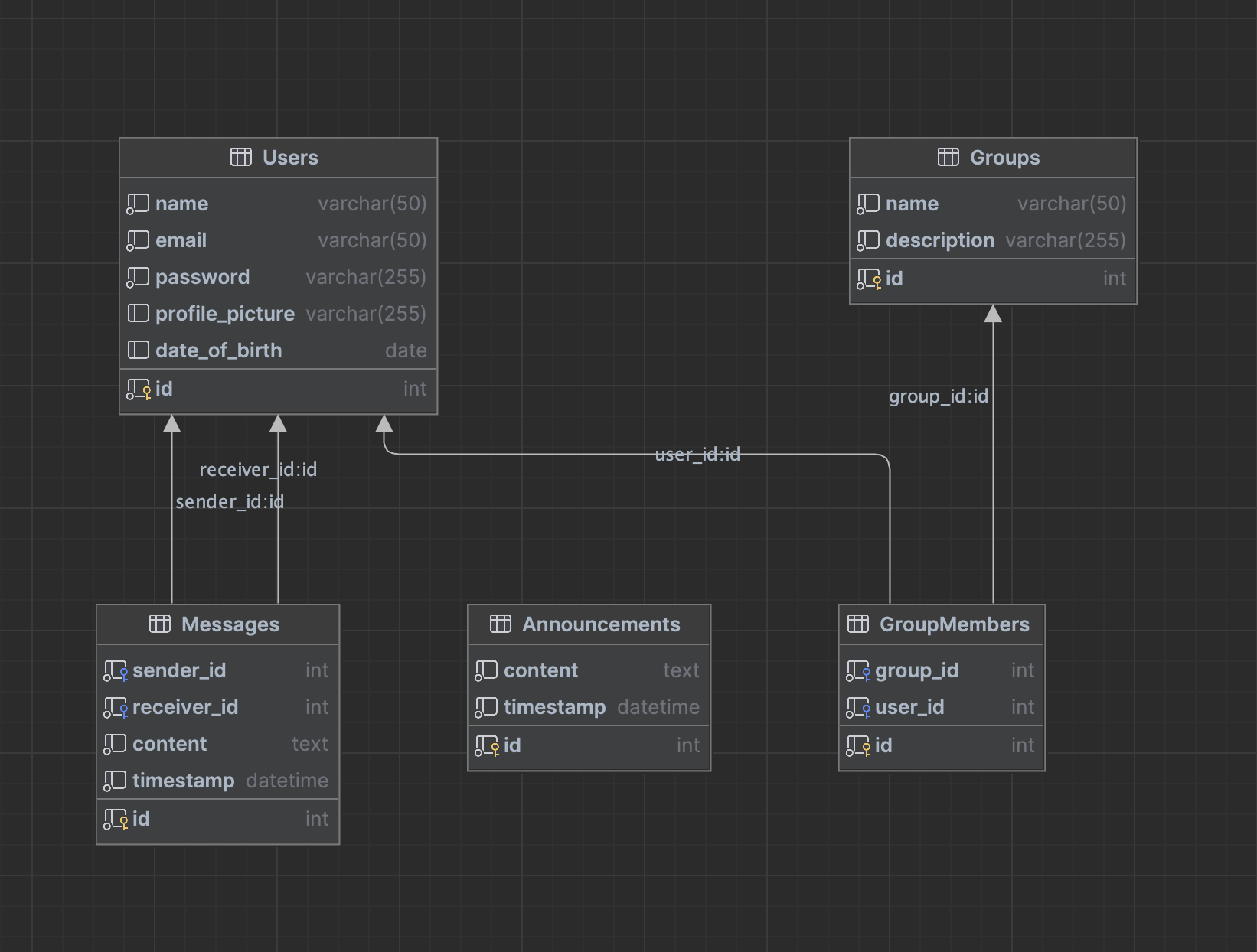
FOREIGN KEY (user\_id) REFERENCES Users(id) );

CREATE TABLE Announcements ( id INT(11) NOT NULL AUTO\_INCREMENT,

content TEXT NOT NULL,

timestamp DATETIME NOT NULL DEFAULT CURRENT\_TIMESTAMP, PRIMARY KEY (id) );

## ER-diagram



## Test Scenarios with Sample data

The below screenshots show the statements for sample data insertion into the above created tables.

A picture containing text

Description automatically generated

Text

Description automatically generated

The below screen captures shows the tables after data insertion.

Graphical user interface, text, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, application, Word

Description automatically generated

## Test cases for various scenarios

1. User Registration and Login:

Graphical user interface, text, application

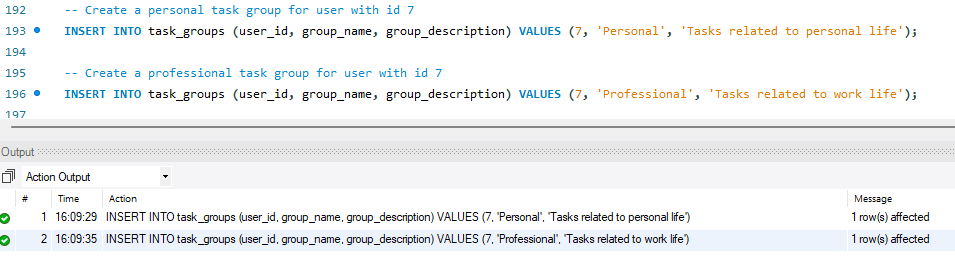
Description automatically generated

1. Token Generation and Verification:

Graphical user interface, text, application

Description automatically generated

1. Creating Task Groups:



1. Creating Tasks and Assigning to Groups:

Graphical user interface, application

Description automatically generated

1. Sharing Tasks with Contacts:

Graphical user interface, text, application

Description automatically generated

1. Deleting a Task:

Graphical user interface, text, application, email

Description automatically generated

1. Updating a Task:

Graphical user interface, text, application, email

Description automatically generated

1. Retrieving all tasks for a user:

Graphical user interface, text, application, email

Description automatically generated

1. Retrieving Tasks by Group:

Graphical user interface, text, application

Description automatically generated

1. Retrieving Shared Tasks:

Graphical user interface, text, application

Description automatically generated

1. Retrieving Contacts:

Graphical user interface, text, application

Description automatically generated

1. Deleting a Contact:

Graphical user interface, text, application, email

Description automatically generated

## Team member contribution

**Jophy**

Worked on 3 tables creation, E-R diagram, inserting data to the tables and test case scenarios.

I will be working on project setup and Django framework integration and implementing the code. Also, will be working on creating API’s and testing all endpoints.

**Savitha**

Worked on 3 tables creation, E-R diagram, inserting data to the tables and test case scenarios.

I will be working on connecting database to the application and testing with multiple scenarios and UI development for login page using HTML. I will be working on creating API’s and testing them using postman.