TABLE OF CONTENTS

TEAM MEMBERS

1. Adigun Boluwatife - 21/2167
2. Adeyemo Ibukunoluwa David - 21/1156
3. Afelogun Moshope - 21/0949
4. Afolabi Tolulope Joanne – 21/2296
5. Adeogun Oluwasemilogo Paul - 21/1427
6. Adeniran Solomon - 21/1151
7. Adeyemi Boluwatife - 21/1900
8. Adeniyi Oreoluwa Elizabeth- 21/2405
9. Ukauwa Mayor Chibuike - 21/2596
10. Adewumi Elisha Adedayo – 21/2062

INTRODUCTION

The **Resource Sharing Platform** is a collaborative web-based application designed to enable users to share, discover, and access a variety of resources, such as documents, tutorials, code snippets, and more. This platform fosters a community-driven environment where users can contribute resources that may be helpful to others. It aims to simplify resource sharing, encourage knowledge exchange, and promote open-source collaboration within various communities, including students, professional's etc. By making resources easily accessible, the platform helps users learn, solve problems, and complete projects more efficiently.

AIM AND OBJECTIVES

Aims:

- To create an accessible, user-friendly platform that allows users to freely share and access resources.
- To build an open-source solution that encourages collaborative learning and community engagement.

Objectives:

- Develop a secure web application to host and manage user-shared resources.
- Enable users to upload, download, and comment on shared resources.
- Facilitate effective categorization and searching to make resources easily discoverable.
- Encourage an open-source culture where users can contribute to and improve the platform.

FEATURES OF THE SYSTEM

User Registration and Authentication: Allows users to sign up and securely log in to access the platform.

Resource Upload and Download: Users can upload files or links as resources, which others can download or view.
Search and Filter Functionality: Includes search bars and filters to help users find specific resources by category, tags, or keywords.
User Profiles: Each user has a profile to track their contributions and manage their resources.
Admin Dashboard: Provides platform administrators with tools to manage content, users, and oversee overall platform activity.
Responsive Design: Ensures the platform is usable across various devices, including mobile, tablet, and desktop.

FUNCTIONAL REQUIREMENTS

1. **User Registration and Authentication**

- The system shall allow students to register using their email or student ID.
- The system shall send a verification link to the student's registered email address to confirm registration.
- The system shall allow students to log in using their email and password.
- The system shall allow students to reset their password through email verification.

2. **User Profile Management**

- The system shall allow students to create and manage their profiles, including adding personal information, profile pictures, and study preferences.
- The system shall allow users to update their account information and preferences.

3. **Upload and Share Resources**

- The system shall allow students to upload different types of files (PDFs, Word documents, PowerPoint, etc.).
- The system shall allow students to categorize the resources (e.g., subject, topic, file type).
- The system shall generate a unique link for each uploaded resource for sharing.
- The system shall allow students to edit or delete their uploaded resources.

4. **Resource Browsing and Searching**

- The system shall provide a search feature that allows students to search for resources using keywords, categories, subjects, or file types.

- The system shall allow filtering search results based on file type, subject, and popularity.
- The system shall provide recommendations based on students' preferences and past searches.

5. **Download and Access Resources**

- The system shall allow students to download any shared resources.
- The system shall ensure that resources are accessible to registered students only.
- The system shall allow students to view resources online without downloading.

6. **Notifications**

- The system shall notify students when new resources are shared in their preferred subjects.
- The system shall notify students when someone comments on or rates their uploaded resources.
- The system shall allow students to turn notifications on or off based on their preferences.

7. **Resource Categorization and Tagging**

- The system shall allow users to categorize resources into predefined categories (e.g., subject, academic level).

NON FUNCTIONAL REQUIREMENTS

1. The system should handle multiple simultaneous uploads and downloads without significant delay, allowing quick and efficient access to resources.

2. The system's user interface should be intuitive and easy to navigate.

3. The system's user interface should accommodatee students with varying levels of technical skills, and making processes like uploading, downloading, and viewing resources straightforward.

4. The system should maintain 99.9% uptime to ensure continuous availability, with uploaded files being backed up to prevent data loss.

5. The system should be scalable, supporting a large number of users and increased traffic over time, as well as accommodating various file types and categories without performance degradation.

6. The system's user profiles and resource access must be secure, with email verification required to validate access, and content restrictions to ensure only authenticated users can access resources.

7. The system should enforce an upload size limit to optimize data storage and prevent excessive memory and disk space usage.

8. Updates and modifications should be easy to implement to maintain the system and respond to new requirements as they arise.

9. The platform should be compatible with various devices, including desktop and mobile, and support major web browsers to ensure accessibility across different user environments.

USER GUIDE FOR THE STUDENT RESOURCE SHARING PLATFORM

1. **Getting Started**

A. Register and Login

i. Sign Up: Enter your email address to register on the platform.
ii. Verification: Check your email for a verification link. Click on the link to verify your account.
iii. Log In: Use your email and password to log into the platform.

2. **Setting up Your Profile**

I. Go to Profile Settings in the dashboard.
II. Fill out your profile information, including study preferences and interests.
III. Save your information to complete your profile setup.

3. **Uploading and Sharing Resources**

A. Uploading Files

i. Navigate to Upload Resource.
ii. Select the file you want to upload (supported formats include documents, images, etc.).
iii. Specify Category: Choose a category (e.g., Music, Tech) that best describes your file.
iv. Set Size Limit: Ensure your file is within the allowed size limit (in MB).
v. Click Upload to share the resource with other students.

B. Viewing Uploaded Files

i. Visit the Resource Library to view all uploaded resources.

ii. Resources can be previewed without downloading.

4. **Downloading Resources**

i. In the Resource Library, browse the available resources.
ii. Click Download next to any file to save it to your device.

5. **Viewing Resources without Downloading**

i. In the Resource Library, you can open any resource to view it directly on the platform.
ii. This feature allows you to quickly check out resources without saving them to your device.

6. **Commenting and Rating Documents**

i. Open any resource to view details.
ii. Scroll to the Comments and Ratings section.
iii. Leave a comment or rate the resource to provide feedback for others.

7. **Notifications**

i. Receive Alerts: You will receive notifications about new uploads, comments, and ratings on resources.
ii. Manage Notifications: Go to Settings to enable or disable specific notifications based on your preferences.

8. **Reporting and Flagging Content**

I. 1. If you come across inappropriate or incorrect content, click on Report next to the resource.
II. 2. This will notify the admin to review the content.
III. 3. Flagged Content: Content flagged by multiple users will be prioritized for admin review.

9. **Admin-Specific Features**

(For Administrators Only)

i. 1. Manage Reports: The admin dashboard includes a Reports section showing all flagged content.
ii. 2. Delete or Block Content: Admins can choose to delete or block any reported posts that violate guidelines.
iii. 3. Receive Email Alerts: Admins will receive an email notification for each reported post.

10. **Troubleshooting & Support**

i. Contact Support: If you encounter any issues, go to Help & Support in the main menu.
ii. FAQ: Refer to the FAQ section for answers to common questions.

DATABASE SCHEMA FOR THE STUDENT RESOURCE SHARING PLATFORM

This is a sample database schema for the student resource sharing platform based on the functional requirements provided. This schema assumes a relational database structure, such as MySQL or PostgreSQL.

---

1. Users Table

Stores information about each user (student or admin).

Field	Type	Description
user_id`	INT (PK)	Unique identifier for each user
`email`	VARCHAR(255)	Users email address
password_hash	VARCHAR(255)	Encrypted password
name	VARCHAR(100)	Full name of the user
is_verified	BOOLEAN	Verification status of the users email
role	ENUM(student, admin)	Role of the user (student or admin)
created_at	TIMESTAMP	Date and time when the user was created
updated_at	TIMESTAMP	Date and time when user information was last updated

---

2. Profiles Table

Stores additional profile information for each user.

Field	Type	Description
profile_id	INT (PK)	Unique identifier for each profile
user_id	INT (FK)	Foreign key referencing `Users.user_id
study_preferences	TEXT	Users study preferences
created_at	TIMESTAMP	Date and time when profile was created
updated_at	TIMESTAMP	Date and time when profile was last updated

3. Resources Table

Stores information about each uploaded resource.

Field	Type	Description
resource_id	INT (PK)	Unique identifier for each resource
user_id	INT (FK)	Foreign key referencing `Users.user_id
title	VARCHAR(255)	Title of the resource
`category	VARCHAR(50)	Category of the resource (e.g., Music, Tech)
file_path	VARCHAR(255)	Path to the uploaded file on the server
file_size	INT	Size of the file in MB
is_public	BOOLEAN	Whether the resource is public
created_at	TIMESTAMP	Date and time when resource was uploaded
updated_at	TIMESTAMP	Date and time when resource was last updated

4. Comments Table

Stores comments on each resource.

Field	Type	Description
comment_id	INT (PK)	Unique identifier for each comment
resource_id	INT (FK)	Foreign key referencing `Resources.resource_id
user_id	INT (FK)	Foreign key referencing `Users.user_id
comment_text	TEXT	Text content of the comment
created_at	TIMESTAMP	Date and time when the comment was made

5. Ratings Table

Stores ratings for each resource.

Field	Type	Description
rating_id	INT (PK)	Unique identifier for each rating
resource_id	INT (FK)	Foreign key referencing Resources.resource_id
user_id	INT (FK)	Foreign key referencing Users.user_id
rating	INT	Rating value (e.g., 1 to 5)
created_at	TIMESTAMP	Date and time when the rating was given

6. Reports Table

Stores reports for flagged resources.

Field	Type	Description
report_id	INT (PK)	Unique identifier for each report
resource_id	INT (FK)	Foreign key referencing `Resources.resource_id
user_id	INT (FK)	Foreign key referencing `Users.user_id
reason	VARCHAR(255)	Reason for reporting the resource
is_resolved	BOOLEAN	Whether the report has been addressed

created_at	TIMESTAMP	Date and time when the report was filed
resolved_at	TIMESTAMP	Date and time when the report was resolved

7. Notifications Table

Stores notifications for each user.

Field	Type	Description
notification_id	INT (PK)	Unique identifier for each notification
user_id`	INT (FK)	Foreign key referencing `Users.user_id
content`	TEXT	Notification message
is_read	BOOLEAN	Whether the notification has been read
created_at	TIMESTAMP	Date and time when the notification was sent

8. Categories Table

Stores predefined categories for resources (e.g., Music, Tech, etc.).

Field	Type	Description
`category_id`	INT (PK)	Unique identifier for each category
`name`	VARCHAR(50)	Name of the category
`description`	TEXT	Description of the category

Relationships Summary

- Users have Profiles, Resources, Comments, Ratings, and Reports.
- Resources belong to Users and have associated Comments, Ratings, and Reports.
- Categories are referenced in Resources to categorize each uploaded file.
- Notifications are linked to Users and alert them about relevant events.

FRONTEND AND BACKEND DEVELOPEMENT TOOLS USED

Technologies Used

Frontend:

- React Js
- Tailwind CSS

Backend:

- Nodejs
- Express.js