

Action Based on Feedback

Step 1

- We clarified the problem statement in the overview to emphasize the need for a unified study system.
- We confirmed our database supports scalability, with statistics included in the overview.
- We updated our ERD relationships to reflect M:N connections between Users-Quizzes and Users-StudySets.
- We added an explanation section for naming conventions, to clarify plural entity and singular attribute naming.
- We did not add new entities, as four existing ones already exist and meet the database requirements.

Step 2

- Made a clear schema (apart from ERD)
- Unified naming conventions everywhere (so that ERD, outline, and DDL file all match).
- Changed some data choices from FLOAT to DECIMAL as DECIMAL can be a safer option for grades/percentages (in DDL.sql)
- Addressed the “possible partial dependencies” comment in the Normalization Analysis conclusion.

Step 3

- Change layout so that the tables are at the top of the page rather than the bottom. This will allow users to see the data first before adding more data to the table.

Step 4

- Moved the RESET button to match the same navigation bar as the other links so that it fits the styling of the website.
- After completing the RESET action, show a confirmation popup before resetting (we already do this), but then after RESET, show a short success popup, and then refresh the current page, rather than redirecting the user to a different page.
- Fixed ERD intersection tables, all three of them are labeled “QuizFlashcards” when they should consist of: QuizFlashcards, UserQuizzes, and UserStudySets.
- Improved the User join date formatting. In the User table, we kept the join_date data type as DATE, but we fixed up the formatting so that it got rid of the meaningless 00:00:00.

Study Application

Group 69: Sasan Pourassef, Jeremy Dempsey

Website URL: <http://classwork.engr.oregonstate.edu:6851/>

Overview

Students from various universities and schools can often struggle to organize their study materials through all of their different subjects they have to track. Many students rely on scattered tools, a combination of Google Docs, Quizlet, and even physical flashcards in order to prepare and practice for exams. This results in duplication, lost progress data, and very limited personalization.

Our study app aims to create a database driven study website that allows users to create, review, and track flashcards, quizzes, and topic-filtered vocabulary lists all in one place. Each user will be able to keep multiple study sets for different modules and topics, which allows a streamlined organization and an easy way to get learning materials. This system will support hundreds of users, with each maintaining dozens of study sets across various topics.

The platform will be able to handle up to 10,000 flashcards, 5,000 quiz attempts, and over 1,000 sessions at a time, ensuring scalability for classroom or institutional use. The platform allows users to track their learning progress and performance across quizzes, offering both personalization and analytics based on insights.

In this initial phase, we will focus on implementing core user functionality: flashcard creation, quiz generation, and progress tracking. These features will be built on a robust, scalable relational database structure that can later support advanced features such as analytics, collaborative study groups or adaptive learning recommendations.

Outline

1. Users
 - Purpose: store registered student accounts and their profile details
 - Attributes:
 - user_id (INT, PK, auto_increment, not NULL)
 - username (VARCHAR(50), unique, not NULL)
 - email (VARCHAR(100), unique, not NULL)
 - password_hash (VARCHAR(255), not NULL)
 - join_date (DATE, not NULL)
 - Relationships:
 - 1:M & M:N with StudySets (a user can create and share many sets, and sets may be shared by multiple users. The user doesn't have to share their study sets)
 - 1:M with quizzes (a user can take many quizzes)

2. StudySets

- Purpose: A named collection of flashcards grouped by topic or course
- Attributes:
 - set_id (INT, PK, auto_increment, not NULL)
 - user_id (INT, FK -> users.user_id, not NULL)
 - title (VARCHAR(100), not NULL)
 - subject (VARCHAR(50), NULL)
 - created_at (DATETIME, not NULL)
- Relationships:
 - 1:M with Flashcards (each set contains many flashcards)
 - M:N with Quizzes (a quiz may include flashcards from multiple sets)
 - M:N with Users (allows study set sharing between multiple users)

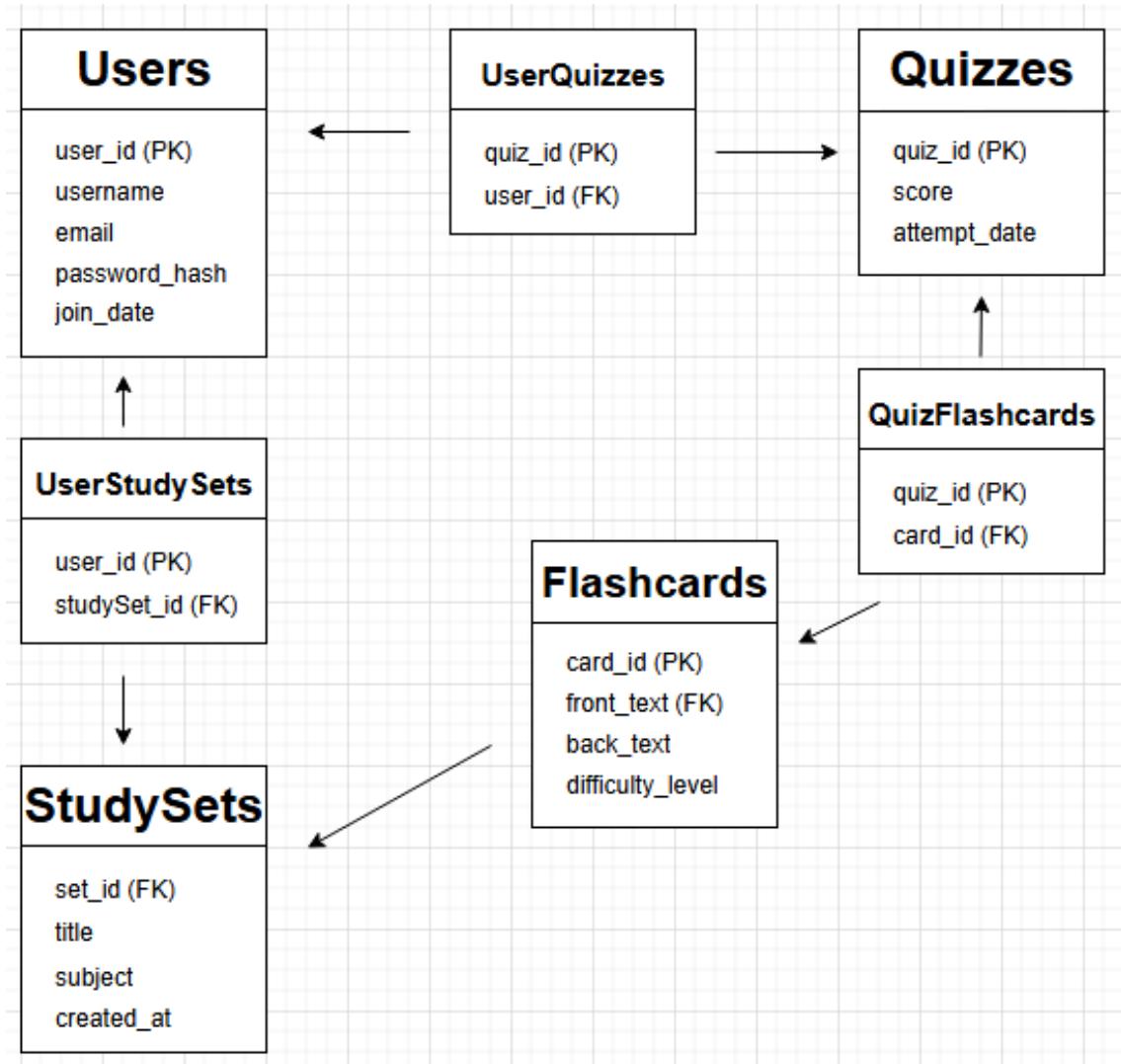
3. Flashcards

- Purpose: Store individual study questions and answers.
- Attributes:
 - card_id (INT, PK, auto_increment, not NULL)
 - set_id (INT, FK -> StudySets.set_id, not NULL)
 - front_text (TEXT, not NULL)
 - back_text (TEXT, not NULL)
 - difficulty_level (ENUM('easy', 'medium', 'hard'))
- Relationships:
 - Belongs to one StudySet
 - M:N with Quizzes (flashcards can appear in multiple quizzes)

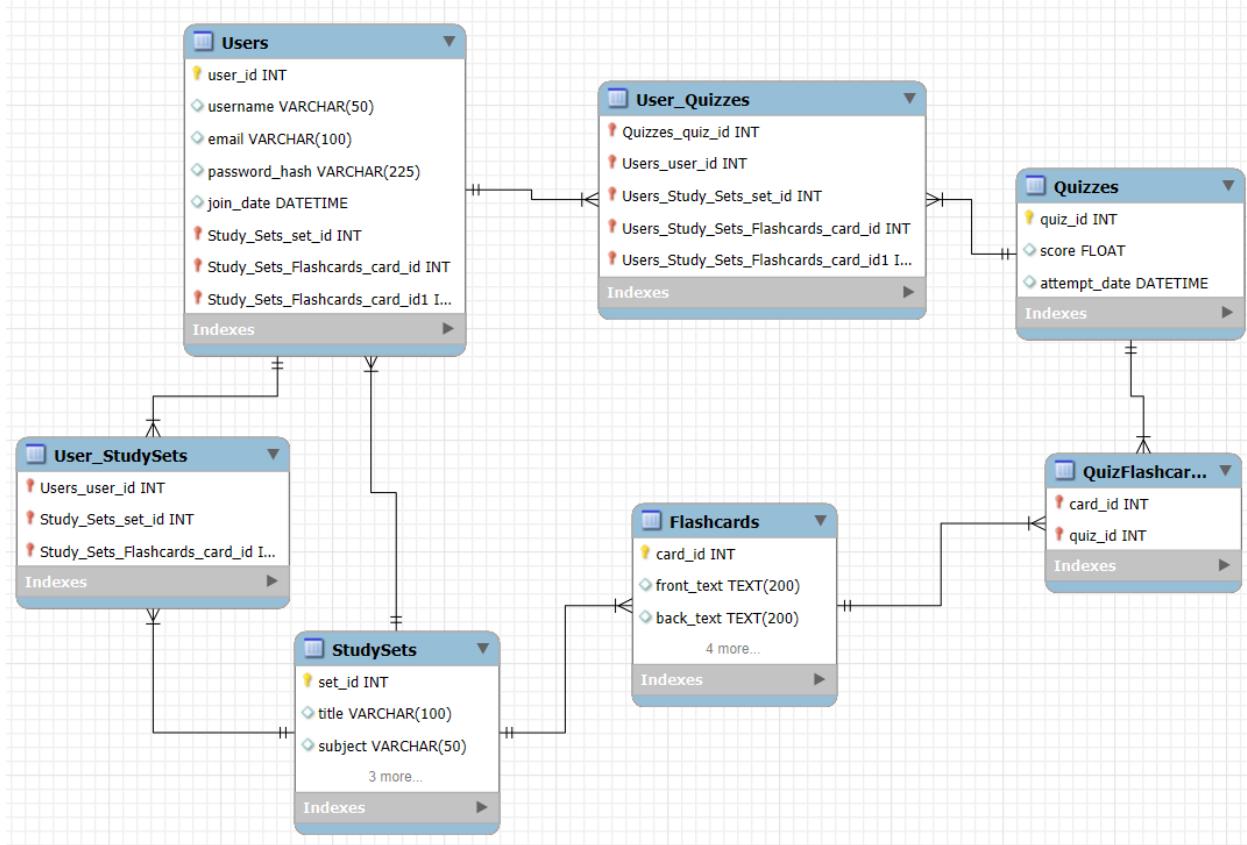
4. Quizzes

- Purpose: Represent generated quizzes, including metadata about when they were taken.
- Attributes:
 - quiz_id (INT, PK, auto_increment, not NULL)
 - user_id (INT, FK -> Users.user_id, not NULL)
 - score (FLOAT(4,2), NULL)
 - attempt_date (DATETIME, not NULL)
- Relationships:
 - M:N with Flashcards (intersection table: QuizFlashcards)
 - M:N with Users (through a linking table: User_Quizzes)

Entity-Relationship Diagram



Schema Diagram



Normalization Analysis

Our database schema has been analyzed using the normalization process:

- 1NF (first normal form)
 - All tables have no repeating groups
 - Each table has a primary key
 - No duplicate rows exist
- 2NF (second normal form)
 - Schema meets 1NF requirements
 - All non-key attributes are functionally dependent on the primary key
 - No partial dependencies exist
 - All tables with composite keys have attributes dependent on the entire key
 - In Quiz_Flashcards, question_order depends on (quiz_id, card_id) composite key
 - In User_StudySets, role depends on (user_id, set_id) composite key
- 3NF (third normal form)
 - Schema meets 2NF requirements

- No transitive dependencies exist
- All non-key attributes depend only on the primary key
- Potential transitive dependencies examined:
 - StudySets.subject could theoretically depend on title, but we allow multiple sets with same title under different users, so no transitive dependency exists
 - Quizzes.score depends only on quiz_id, not on any intermediate attributes
 - All foreign keys represent relationships, not transitive dependencies

Conclusion: Our schema is in 3NF. No denormalization was required as all relationships are properly modeled and data integrity is maintained without introducing redundancy or anomalies. There is no partial dependency in Users, StudySets, or Flashcards as each table non-key attributes depend on its single-column PK; and junctions have composite PKs and their non-key attributes, such as role or question_order, where they depend on the full composite.

Example data (from DDL.sql):

Users Table:

user_id	username	email	password _hash	join_date
1	sasan	sasan@example.com	hash_sasan	2025-09-01
2	jeremy	jeremy@example.com	hash_jeremy	2025-09-03
3	john	john@example.com	hash_john	2025-09-10

StudySets Table:

set_id	user_id	title	subject	created_at
1	1	CS340 - SQL Basics	Databases	2025-09-15 09:30:00
2	1	Biology - Cell Terms	Biology	2025-09-15 14:00:00
3	2	Algorithms - Sorting	CS	2025-09-25 16:45:00

Flashcards Table:

card_id	set_id	front_text	back_text	difficulty_level
1	1	What is a primary key?	A column or set of columns that uniquely identify a row.	easy
2	1	Define foreign key.	A field referencing a PK in another table; enforces referential integrity	easy
3	1	What does ON DELETE CASCADE do?	Deletes child rows when the parent is deleted.	medium

4	2	Mitochondria function?	Powerhouse of the cell; ATP production.	easy
5	3	Average time complexity of merge sort?	$O(n \log n)$	medium

Quizzes table:

quiz_id	user_id	score	attempt_date
1	1	95.00	2025-10-01 10:00:00
2	2	88.50	2025-10-02 13:20:00

User_StudySets table (demonstrates M:N relationship - sharing study sets):

user_id	set_id	role
2	1	viewer
3	1	editor

User_Quizzes table (demonstrates M:N relationship - users and quizzes):

user_id	quiz_id
1	1
2	1
2	2

Quiz_Flashcards table(demonstrates M:N relationship - flashcards in quizzes):

quiz_id	card_id	question_order
1	1	1
1	2	2
1	3	3
2	5	1

Citations

All core work on the project is original and created by Sasan Pourassef and Jeremy Dempsey.
AI was used to generate example data and to generate the base style.css.