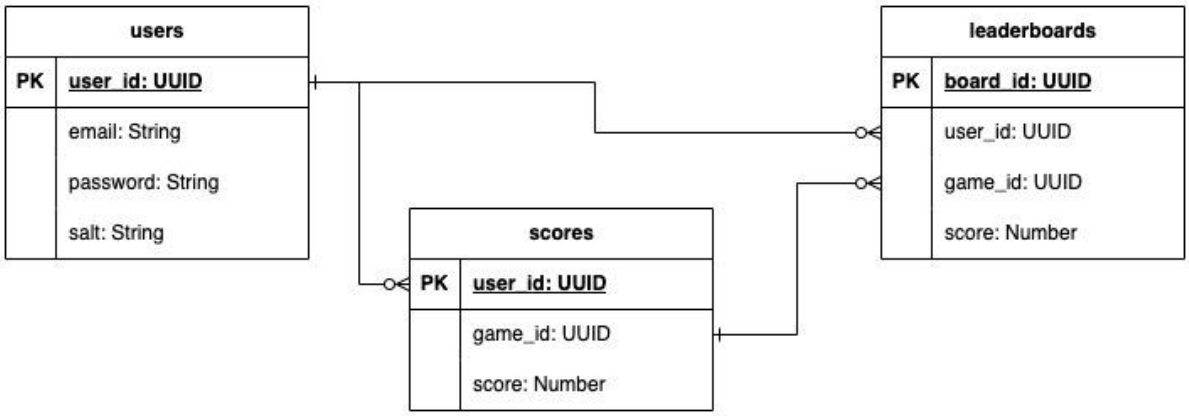
Broken Calculator
Product Design Specification

End Product Description:

The end product, "Broken Calculator," is a challenging puzzle game where users are presented with a desired numerical result and a limited set of functioning buttons on a calculator. Users must strategically use the available buttons to perform arithmetic operations and reach the target result. The game interface will feature a display for the current numerical input, functional buttons, and feedback on the correctness of the calculated result. The game will be implemented as a web application using Flask for the backend and Next.js for the frontend.



Methods Used in the Puzzle Solver:

The puzzle solver in "Broken Calculator" will utilize a combination of algorithms for evaluating arithmetic expressions and heuristic strategies for finding optimal solutions given the constraints of the available buttons. It will analyze the target result and the set of functioning buttons to generate sequences of arithmetic operations that lead to the desired outcome.

Market Space and Selling Points:

"Broken Calculator" appeals to puzzle game enthusiasts and individuals seeking brain-teasing challenges. Its selling points include:

1. Unique gameplay concept that requires strategic thinking and problem-solving skills.
2. Varied difficulty levels, catering to both casual players and hardcore puzzle solvers.
3. Engaging user experience with intuitive controls and interactive feedback.
4. Replay value with randomized puzzles and leaderboard functionality for competitive players.

Functional Specifications

Product Features:

Target Result Input: Users can input the desired numerical result they aim to achieve.
Functional Buttons: Limited set of buttons (e.g., addition, subtraction, multiplication, division) available for performing arithmetic operations.
Numerical Input Display: Visual representation of the current numerical input and calculated result.
Arithmetic Operations: Users can perform arithmetic operations using the available functional buttons.
Feedback Mechanism: Real-time feedback on the correctness of the calculated result compared to the target result.
Hint System: Optional hints available to assist users in finding solutions to challenging puzzles.
Deployment

Deployment

Deploying the Flask Project:

To deploy the Flask project "Broken Calculator" follow these steps:

1. Set up a Heroku account and install the Heroku CLI.
2. Create a new Heroku app from the Heroku dashboard or using the Heroku CLI.
3. Clone the Github Repository locally.
4. Connect your local Git repository to the Heroku app using the Heroku CLI.
5. Push your code to the Heroku remote repository to trigger deployment.
6. Configure any necessary environment variables (Specified in Readme.md).
7. Access your deployed application via the provided Heroku app URL.

Features for Major Milestones:

M1: Basic game functionality with target result input, functional buttons, and numerical input display.
M2: Implementation of arithmetic operations and feedback mechanism for result validation.
M3: Integration of hint system to assist users in solving puzzles.
M4: Optimization of user interface and addition of leaderboard functionality.
M5: Deployment and performance optimization for seamless gaming experience across different devices.