Product Design Specification (Individual Flask Project)

Project Background

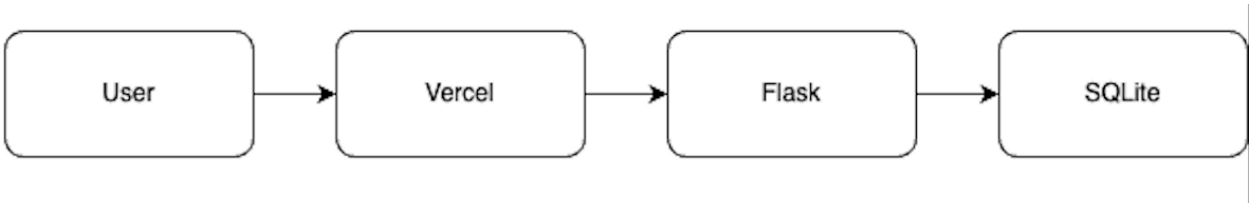
- Project name: Water Carrier
- Created by: Da Thao Trinh
- GitHub URL: https://github.com/dathaotrinh/watercarrier

Project Description

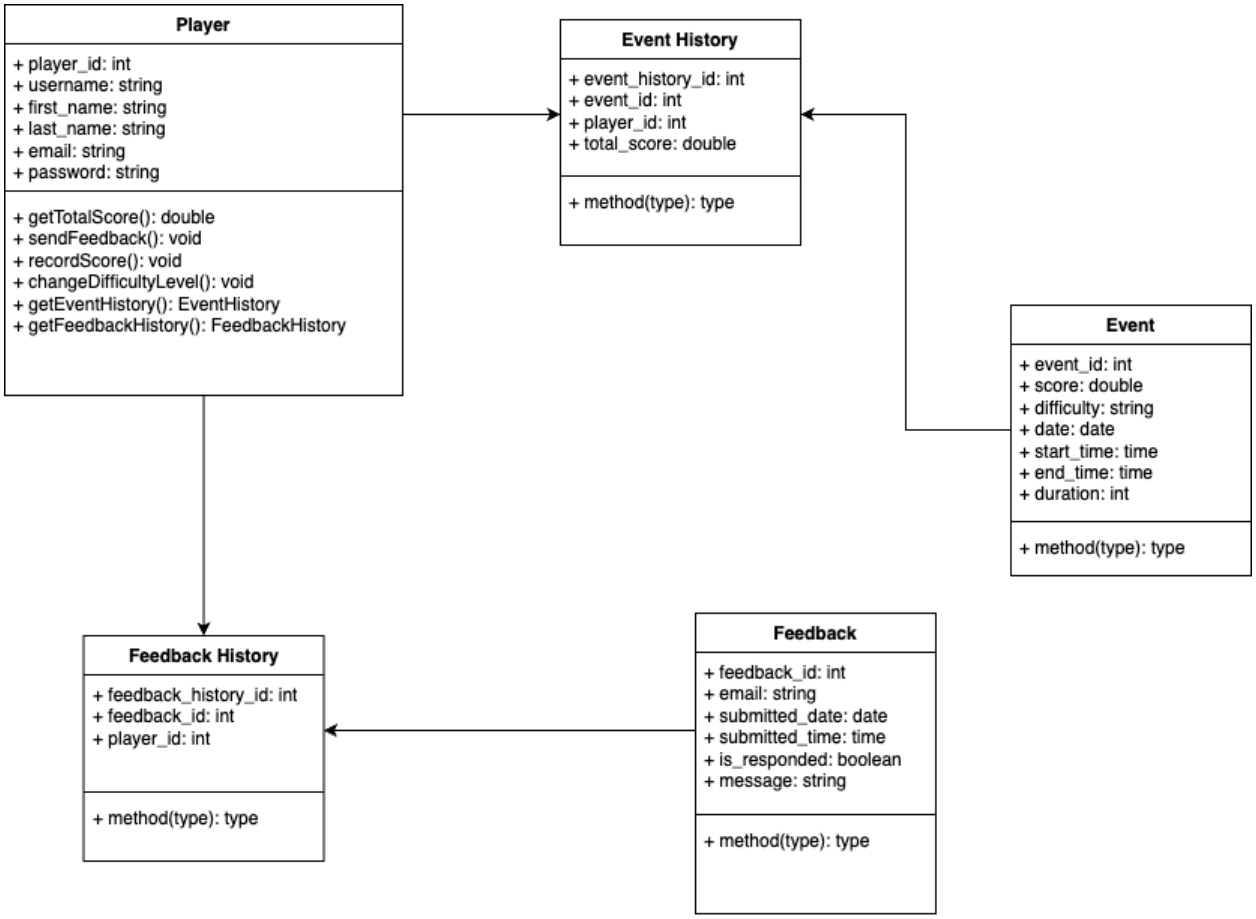
- Imagine you have a bunch of walls of different heights. Each wall can hold water on top of it, and you want to find a pair of walls that, when treated like the sides of a container, can hold the maximum amount of water.
- The score will be determined by how fast the user solves the problem.

Diagrams

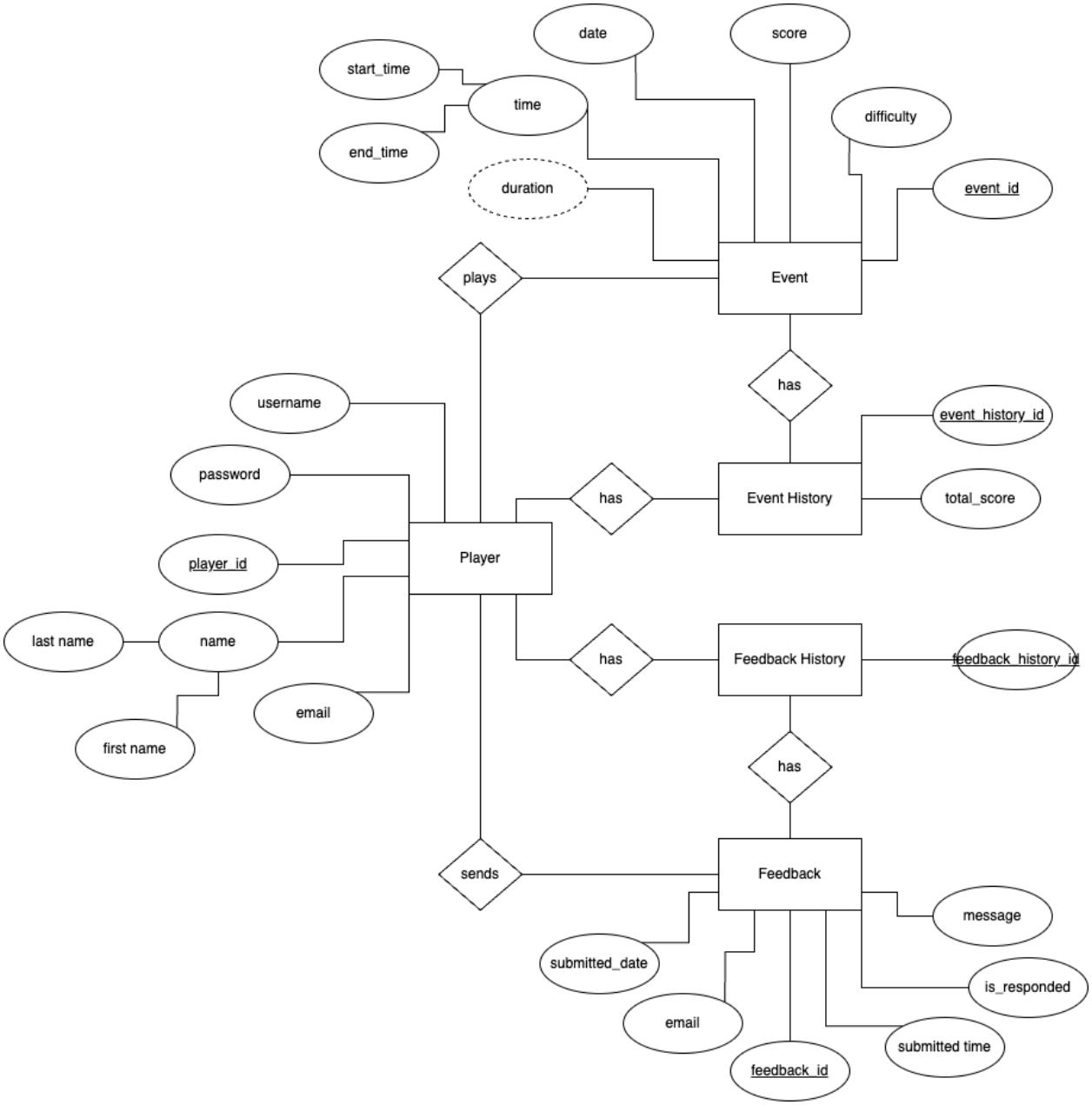
- Flowchart



- UML Diagram


Player
+ player_id: int
+ username: string
+ first_name: string
+ last_name: string
+ email: string
+ password: string
+ getTotalScore(): double
+ sendFeedback(): void
+ recordScore(): void
+ changeDifficultyLevel(): void
+ getEventHistory(): EventHistory
+ getFeedbackHistory(): FeedbackHistory
Event History
+ event_history_id: int
+ event_id: int
+ player_id: int
+ total_score: double
+ method(type): type
Event
+ event_id: int
+ score: double
+ difficulty: string
+ date: date
+ start_time: time
+ end_time: time
+ duration: int
+ method(type): type
Feedback History
+ feedback_history_id: int
+ feedback_id: int
+ player_id: int
+ method(type): type
Feedback
+ feedback_id: int
+ email: string
+ submitted_date: date
+ submitted_time: time
+ is_responded: boolean
+ message: string
+ method(type): type

- ER Diagram



Method used to solve the problem

- One approach to solving this problem is to use a two-pointer technique.
- Start with two pointers, one at the beginning of the array and the other at the end.
- Calculate the area formed by the current two pointers.
- Move the pointer with the shorter height inward, as the area is limited by the shorter height. This is because increasing the distance between the two lines can only increase the area if the height of the shorter line increases.

Market space and selling points

- The game encourages players to think about spatial reasoning and problem-solving, making it suitable for educational purposes.

Features

- User Registration and Authentication
  - Allow users to create new accounts.
  - Enable login via email.
- User Profile Management
  - Profile customization.
  - View/edit account settings.
  - Track earned score.
- Gameplay Features
  - Browser-based game.
  - Show demonstration of containers through bar chart dashboard.
  - Different levels of difficulty.
  - Play game without login
- Content Management and Administration
  - Admin dashboard for managing users.
  - Feedback tracking.
- Score Ranking
  - Ranking the score for users from highest to lowest.
- Support and Bug Report (Feedback)
  - Contact form to send feedback.

Deployment

1. Install Vercel CLI tool. `npm install -g vercel`
2. Navigate to directory for Flask app using the command line. then Add the required dependencies to the `requirements.txt` file: `pip freeze > requirements.txt`
3. Create a new file called `**vercel.json**` in the root directory of Flask app. This file will contain the configuration information for Vercel.

4. Add the following **JSON** code to the **vercel.json** file:

```
{
 "version": 2,
 "builds": [
  { "src": "app.py", "use": "@vercel/python" }
 ],
 "routes": [
  { "src": "/(.*)", "dest": "/app.py" }
 ]
}
```

5. In the above code, replace **app.py** with the name of my Flask app file.
6. Finally, deploy your app to Vercel by running the following command. `vercel deploy`
7. Follow the prompts to configure your deployment, including setting up a Vercel account if you haven't already.

Source:

https://medium.com/@nohanabil/how-to-deploy-flask-app-using-vercel-885ce034624

Milestones

- Milestone 1 (2/5 - 2/16): Initialize technology frontend, and backend development environments.
- Milestone 2 (2/12 - 2/23): Build UI for Login and Registration Forms.
- Milestone 3 (2/26 - 3/8): Implement bar chart and gaming page.
- Milestone 4 (3/11 - 4/12): Implement "Container with Most Water" algorithm, statistics UI & Testing.
- Milestone 5 (4/16 - End of the semester): Deployment & Testing for bugs and making visuals look better.