

Project Proposal

Problem Description

The application involves a system where users submit code to solve algorithmic problems and improve their coding skills. The challenge is to ensure that the test cases, random number generation (rng), and grading process are transparent and fair. Both the problem setters (commissioners) and the users need to trust the system, particularly in how test cases are generated and how the grading is handled, without manipulation from either side.

Proposed Application

The proposed application uses blockchain to secure the process of submitting code, generating test cases, and grading solutions. By storing key data like rng seeds, problem hashes, and grading outcomes on a decentralized ledger, the platform guarantees transparency and immutability, ensuring that all parties can verify the fairness of the process.

Why Blockchain is Needed

1. Grading Fairness:
 - To ensure that user submissions are evaluated fairly, the grading process and results are stored on the blockchain. This prevents manipulation of results by either the platform administrators or the users. The immutable record of submissions and grading outcomes ensures transparency, with both sides being able to audit the process if needed.
2. Problem Reveal and Selection:
 - Blockchain can be used to reveal a portion of the problem (e.g., 1/3 of the test cases) selected by a hash derived from the rng seed. This ensures fair and random disclosure, preventing either party from gaining an unfair advantage.
 - To pass, the user's submitted code must correctly solve at least 2/3 of the revealed test cases. This allows for minor errors without complete rejection.
3. Immutable Data:
 - Storing test sets, user submissions, and grading results on the blockchain creates an immutable, tamper-proof record of each interaction. This ensures that the problem setter cannot change the test set after the user has submitted their code, and the user cannot alter their solution after submission. Both sides can trust the process without relying on a single party to guarantee fairness.

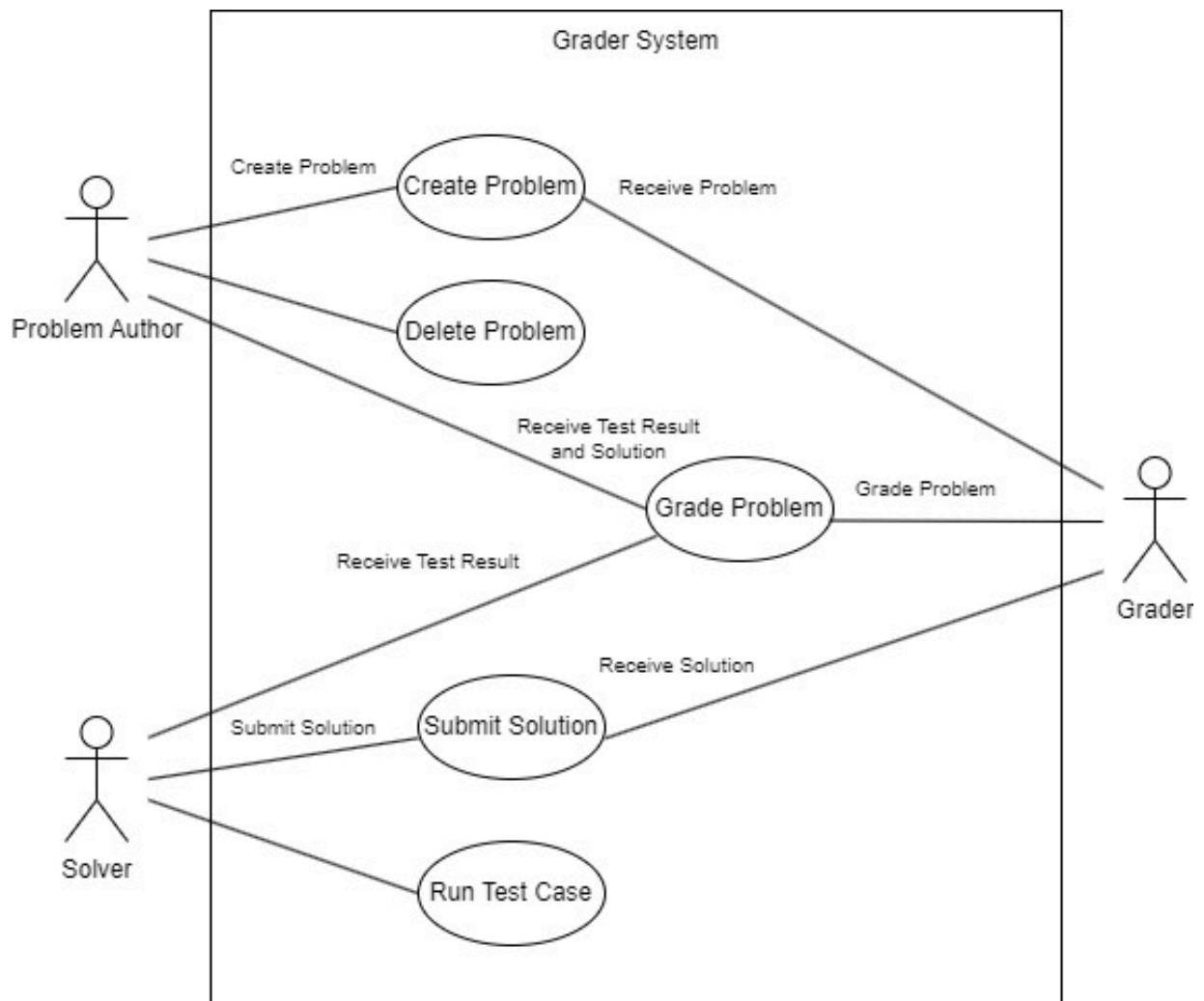
4. Penalties and Incentives:

- Blockchain enforces penalties for failure to comply with the process. For example, if the commissioner refuses to reveal the remaining test cases after the user has passed the initial set, a fine can automatically be paid to the user. Conversely, users submitting incorrect or invalid solutions may face penalties, encouraging adherence to the rules from all sides.

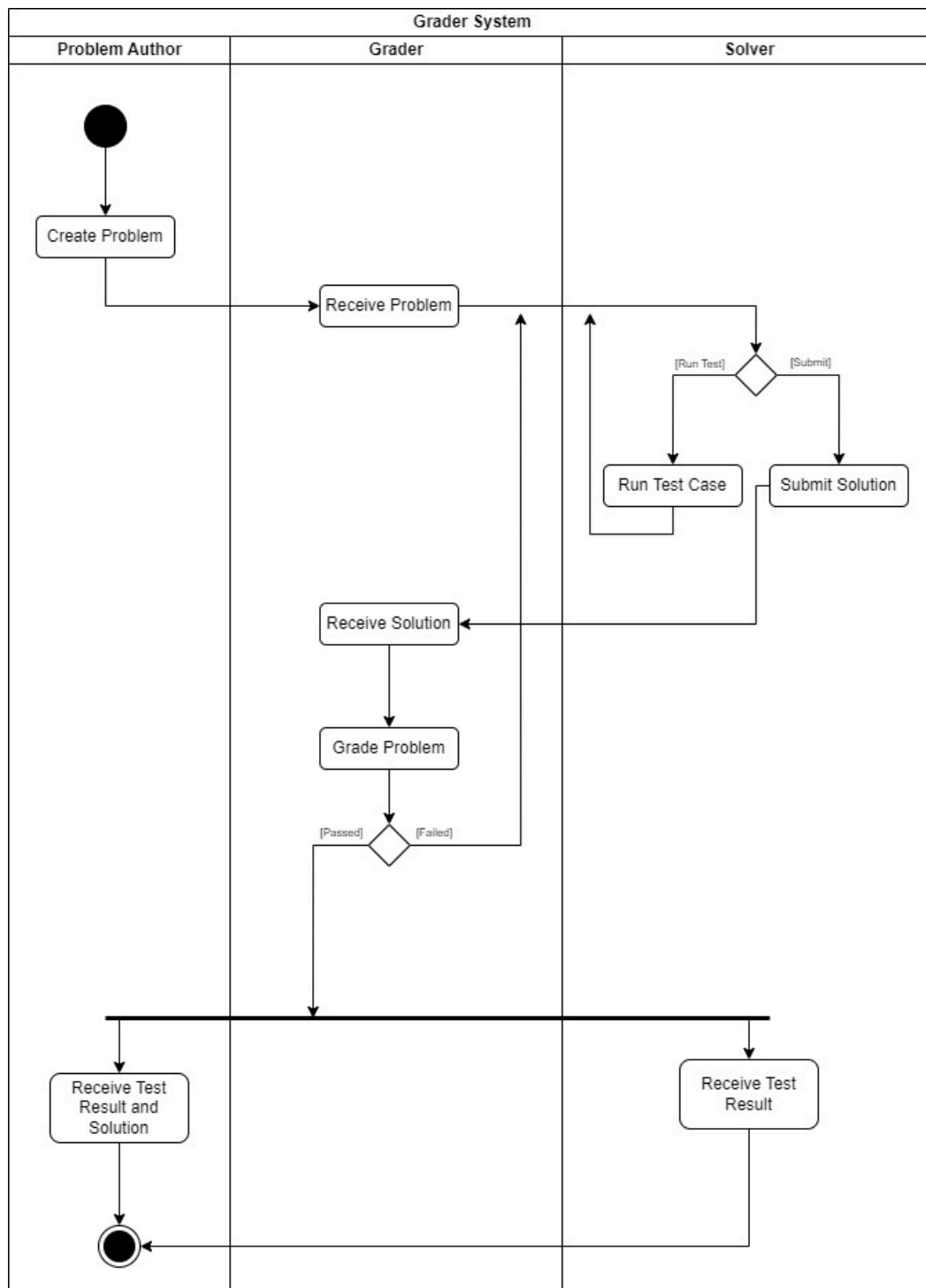
5. Automatic Timeout Handling:

- The blockchain automatically enforces challenge deadlines. If no valid solution is submitted by the deadline, rewards are returned, and the process concludes without dispute. This prevents one party from delaying or stalling the process.

Use Case Diagram



Activity Diagram



DLT

We chose **Ethereum** because it provides secure, decentralized smart contracts that ensure transparency and fairness. Its immutability guarantees that test cases, solutions, and grading outcomes cannot be tampered with. Ethereum's large ecosystem and auditability also make it ideal for managing penalties, incentives, and automated enforcement of rules.