Blockchain based ticket management system

IMPLEMENTATION PROJECT – MEDIUM

1 PROBLEM STATEMENT

Create a blockchain based online ticket management system. The ledger will contain the distribution of seats for a given transportation mode (plane, train or bus). The platform will support booking a transportation mode from a source to a destination, show available vacant seats, filled seats and ticket prices along with date of journey. There are two entities involved, the transportation service provider and the ticket buyer (passenger). The platform must support adding new entries for transportation service provider, new modes of transport, on-boarding new customers (passengers) and deletion of existing booking by the user or mode of transport by the transportation service provider.

2 DELIVERABLES

A github repository containing the code for running and maintaining the company-appraisal system along with a project documentation report that list the details on how to use the platform and the features it has. Your team is welcome to make changes to the scheme mentioned above or adapt your own policy for the ticket-booking platform, mention the same in your project documentation report explaining the changes.

- Create or delete a passenger, mode of transport, transportation provider etc. in the platform.
- Update passenger details and transportation details on the platform. (User has the option to make the profile anonymous or public)
- No changes are possible by the user or transportation provider once a confirmed booking is made on the platform.
- The platform must support dynamic pricing of tickets. (Come up with a scheme to maximize the profits made by the transportation provider yet affordable for the customers.)
- Booking details or passenger information must be **verifiable** on the platform.

3 MINIMUM DELIVERABLES (60 MARKS)

- Create a new customer/travel service provider on the platform.
- Ability to create a ticket for the customer.
- Ability to list transport options along with timings upon choosing the source and the destination.
- Project report mentioning the features and how to use them.
- Successfully demonstrate the use of blockchain based technology in the platform.

4 QUALITY ATTRIBUTES

- System must atleast scale to a 100 active users.
- Atleast 3 options must be available for the customer for each mode of travel listed on the website.
- Tickets are confirmed once the ledger is updated and atleast 2 blocks succeed the confirming block in the blockchain.

Author's address: Implementation Project – Medium.

5 GRADING POLICY

- **80 marks** are awarded if the booking system works without glitch and meets the quality attributes mentioned above.
- **10 marks** are awarded for properly documenting and presenting the features of the system.
- **10 marks** are awarded for showing the system working for 100+ passengers with 50+ different bookings made on the platform and confirmed tickets are verifiable.
- **10 marks** are awarded as **bonus** if tickets can be listed by availability and pricing.
- **10 marks** are awarded as **bonus** for new "useful" or "interesting" features that utilize the underlying blockchain or supports clever dynamic pricing policy. Any other “extra” or “useful” feature can also be worthy of bonus points.

6 DISCLAIMER

Before doing anything “extra” (which might fetch bonus marks), first, complete the basic expectations from your implementation.

Software tools are expected to display their results in a user-friendly manner; a user would never like to use a tool that simply spits out a bunch of numbers. So, display the results from your tool suitably possibly in a good web-based UI or the terminal in verbose user-friendly manner.

Discussion is healthy, copying is not. You are encouraged to discuss the projects with your peers, but you must implement the projects by yourself. If any two groups are found with "similar" pieces of code, both of them will be failed (with no concern as to who was the source). Copying from internet sources or open-source github repositories must be refrained from.

TAs may conduct a code-review after every milestone is reached or 15 days (which ever is earlier) so please be careful about plagiarism.