- O S T A D L I M I T E D -

Flutter Developer Recruitment Question

NOTE: By gaining access and viewing this document, you agree to solve the challenge and do not share this document with anyone else.

Overview

You will build an app with Flutter. Completing all steps is unnecessary, but completing each step will earn you extra points.

Please read the assessment criteria.

Reach out to us if you have trouble understanding the objective or want to validate your idea of the tasks. This also helps us to understand your progress and communication skills.
For any queries :

Fahim Faisal Shantu
Accounts & HR Manager at Ostad
Contact: +8801940-444470 (WhatsApp)

Submit solution to:
Email: career@ostad.app
Subject line: **Flutter Developer Recruitment Test**

Till February 22, 2023, for the submission of this test., 2 PM is the last time.

Assessment Criteria

- The primary assessment criteria are your learning and adapting capabilities. We are looking for people who can learn a new language, framework, etc., in a short time and can implement solutions.
- We are also looking for people who can read plain vanilla instructions in English and build systems, as a lot of the time work will be remote.
- Early submission will rank high.
- Clean & Well Documented Code will rank high.
- The more tasks you complete, the higher you will rank.

Tasks

For each step below, create a git commit with the step name such as “Step 1” if you have coded up multiple steps simultaneously, just commit with the latest step you worked on. Document in the git README.md file the packages you used in each step. If you used any, document why you used them.

- Step 1: Build an Auction App like eBay. Use firebase auth so that users can sign in using their Gmail account.

- Step 2: After login, the user will see the auction item gallery, which shows the items everybody else has put up for auction. There should be a create/plus button on the homepage that allows the user to create an auction item for everybody to see and interact with. If the user hits that button, a form appears, which lets the user input the Product Name, Product Description, Product Photo, Minimum Bid Price, and Auction End DateTime. When an auction item is created using this form, it shows up in the auction gallery of everybody else and the “My posted items” menu for the posting user. Use firebase to create a backend that holds all this data.

- Step 3: Users can place bids on items posted by others within the Auction End DateTime. If they click on any item in the auction item gallery, they will be taken to the auction page of that item, where they can see the Product Name, Product Description, Product Photo, Minimum Bid Price, and Auction End DateTime. It will also show a table of bids placed by other users for that product. Users can input their bid on the auction item page for that product. After inputting a bid, it shows up in the bid list. The user can edit their bid before the auction ends.

- Step 4: The auction will end at Auction End DateTime; entering the auction item page for any item will show the bid winner for that item.

- Step 5: Make everything look pretty based on your judgment.

- Step 6: Whatever you have built, compile to an apk, upload the code with the apk on GitHub, and send us the link in reply to the mail that we sent you the task in.

- Step 7: Put in the README.md file all the challenges you have faced in each step and how you solved them.

- Step 8: Make a dashboard view showing the total number of running bids, the total number of completed bids, and the total value of completed bids(multiply the winning bid price by the quantity of products). Show a time series of these data as well in a line chart. The statistics are for all auctions combined; all the users should be able to see these stats. Place the entry point of the dashboard in the bottom navigation bar. Update the GitHub code, README.md, and apk.

H A P P Y E N D I N G

- w w w . o s t a d . a p p -