

Be prepared to discuss:

1. What decisions did you make to implement your solution?
2. What are the limitations of your implementation due to time constraints?
3. How did you organize the code and why?
4. Given more time, what would you have liked to implement?

Design Thoughts

Language

Python

Framework

Django

Constraints

Time - 60 minutes

Repository

GitHub

Success Metrics

Unknown what will be needed

Error Handling

Unknown where to persist errors or who is handling future issues.

Functionality

- Accept any amount of money in USD as input and return set of coins that add up to that amount.
- Ensure the least amount of coins returned.
- Following coins are allowed:
 - Dollar coin
 - quarter
 - dime
 - nickel

THIS DOCUMENT CONTAINS PROPRIETARY AND CONFIDENTIAL INFORMATION OF CLIFF ROBBINS AND SHALL NOT BE USED, DISCLOSED OR REPRODUCED, IN WHOLE OR PART, FOR ANY PURPOSE OTHER THAN TO EVALUATE THIS DOCUMENT, WITHOUT THE PRIOR WRITTEN CONSENT OF CLIFF ROBBINS.

- penny

Unknowns

1. Who will be using the application?
2. How will this add value?
3. How much revenue will this application (or bigger app) create?
4. Volume of traffic/calls
5. Does it need a front end?
 - a. Loads of other questions...
6. Will it ever need to accept another currency?
7. What if the allowed coins change?
8. How do we know or record volume?
9. What if there is an error?
10. Who will be maintaining the application in the future?
11. Is there a physical component?
12. What type of system does this need deployed to?
13. Any type of association with accounts or users?
 - a. single sign on
 - b. api based
14. Security concerns, need more information about product.
15. Other stuff not thought of yet due to minimal understanding of application