

Full stack developer test (45 minutes)

Guide

- Comment your code to make it readable
- Please upload your code to Github and share a public URL to the repo.
- Choose an alternate framework using the same base language if you don't have experience with the framework mentioned.

Develop a python-flask based User REST API

1. Supports user retrieval, creation and update
2. User profile includes name, email address and telephone number
3. API follows REST standards and uses verbs like GET, PUT and POST
4. User REST API supports password based authentication
5. Uses a relational database (MySQL..) for storage
6. Passwords are encrypted in the storage
7. REST API should be self-contained and can be run in a Docker container

Develop python-flask decorator methods for authentication and authorisation

We want to secure additional APIs based on user authentication and authorisation. Please develop a python-flask package that can be included in new services and protect the APIs.

1. Develop decorator definitions for protecting API methods
2. Decorators should work for all REST verbs (GET, POST and PUT)

Develop a ReactJS frontend UI for user creation, updates and login

- Using User APIs, create a ReactJS frontend
- Develop components for user registration and login
- Use promises and error handling where required