

Summer 2019 Internship - Full-Stack Technical Exercise

This exercise consists of few small problems. You should not spend more than 90 minutes on them. If any part seems confusing, skip it and come back to it later - and we do not expect you to finish everything - this is primarily to help us evaluate your strengths.

In this problem, we are trying to create a basic payment management user interface. We have started to build it out using React. There are some minor improvements we are looking to add.

To get started:

- Clone the template React app from https://github.com/NikhilPunwaney/user_payments
- Install the necessary node modules by calling 'yarn install'
- To get the app going, enter 'yarn dev'
- As you make your edits, the app will update automatically

The desired functionality for this process:

- On load, the user is able to see all the users in our database.
- User selects a person. All payments for that user appears in another column.
- User is able to filter payments for a person by the payment status.
- User selects a payment. A form to issue a payment request appears in another column.
- Assuming the payment request is valid, the user can submit it - upon which the payment is updated.

This exercise involves some understanding of endpoints, data structure management, HTML and CSS, the React framework, NodeJS, using third party APIs.

The third party API referenced is Stripe and the documentation for its API can be found at <https://stripe.com/docs/api/errors/handling?lang=node>.

The only files you need to touch include: server.js, client/App.js, client/ClientColumn.css, client/FormColumn.js, client/PaymentColumn.css and client/PaymentColumn.js.

Each file has // TO DOs which should make it easier to follow.

Please create your own **private** GitHub repository with your response and add nikhil@deepbench.io as a collaborator within 90 minutes of receiving this challenge.