BYTIVE

Bytive Technologies - Mern Stack Assignment

Here we provide the instructions for developing two ReactJS applications, designed by the Bytive Technologies team, to test the proficiency of applicants for the Mern Stack Development position(s):

- Assignment 1 (**Beginner Frontend**): For applicants who have just learned React and have minimal to no experience in developing applications using React.
- Assignment 2 (**Advance Frontend**) : For applicants who have a good working knowledge of React and have worked on / developed at least one React application
- Assignment 3 (**Frontend & Backend**)- For Applicants who have good working knowledge of Both React and Nodejs Express Api Development.

All applicants are required to submit **only one assignment based on their knowledge**, experience and confidence in ReactJS, Nodejs Express Mongodb. We advise applicants to see the demo and instructions for both the assignments before deciding which one to submit.

Prerequisites

- Knowledge of ReactJS (obviously), HTML, CSS, Bootstrap (optional, but will be really helpful for Assignment 1)
- Knowledge of installing npm packages (both local & global)
- Knowledge of creating React application using Create React App
- Knowledge of fetching JSON data from a REST API endpoint
- A Github account in order to share his/her code with us

Submission guidelines

All applicants must share their code in one of the following ways:

- Upload their code on their Github account by creating a public repository with following naming conversion [name_assignment_{no. Of chosen assignment}_] and sharing the link via email.
- Creating a zip file of their project folder (excluding the node_modules directory) and either uploading it to a cloud service (sharing the link with us) or attaching it (if size < 20 MB).

Deploying the code for testing

All applicants are also required to deploy their assignment to a server or to any of the static hosting service providers (link to help with deployment).

Assignment Details

Both the assignments (Beginner & Advanced) are frontend React applications with no backend development required.
The idea of both the assignments is to build a single page that displays the profile of 10 users (the data is obtained from an API endpoint). Each user's profile contains a avatar picture, name, email, phone, address, website and company name.

API endpoint for users data

All 10 users profile data is to be downloaded from the following API endpoint:
Method: GET
URL: https://jsonplaceholder.typicode.com/users

The schema of the data received in the response is:

```
// Array of 10 users
[
  {
    id,    // The user's id
    username,
    name,
    email,
    phone,
    website,
    address: {
            street, // Address line 1
            suite, // Address line 2
            city,
            zipcode
    },
    company: {
            name, // The name of company where the user works
    }
  }
]
```

API endpoint for users' avatar pictures

You will be using Avatars by DiceBear. They provide a free HTTP API to create unique avatar images based on any string we send as query parameter. Each string generates a unique image. You will use the username to generate a unique avatar for each user.
The URL for the GET endpoint is:
https://avatars.dicebear.com/v2/avataaars/{{username}}.svg?options[mood][]=happy

The {{username}} in the URL is the placeholder for the user's username. It should be dynamically replaced by the username received from the users API endpoint. For example, if the username for one of the users is psamd then the URL for the avatar for this user will be: https://avatars.dicebear.com/v2/avataaars/psamd.svg?options[mood][]=happy

Loading Indicator

Upon opening the app a loading indicator is displayed until the data is fetched from the API and is ready to be displayed. The source code for the loading indicator can be obtained from: http://tobiasahlin.com/spinkit/.

Assignment 1 - Frontend Beginner

Deadline for submission: 3 days
DEMO - https://react-basic-assignment.psamd.now.sh/
This assignment is designed for applicants who have learned ReactJS and have minimal hands-on experience in developing react application.

What are we looking for?

With this assignment we would evaluate the following:

- Ability to create new react projects using Create React App
- Understanding of JSX
- Passing props to components
- Understanding of stateful and stateless Components
- Basic understanding of state management and component lifecycle methods
- Fetching data from an API endpoint
- Conditional rendering
- Working with lists

Instructions

We want applicants to create an exact replica of the assignment 1 demo app.
This app is only designed for desktop/laptop and will be tested on Chrome browser.
Bootstrap was used to create the UI design for the demo app. But, applicants are encouraged to use any other CSS library/framework that they are comfortable with.

Assignment 2 - Frontend Advanced

Deadline for submission: 5 days
DEMO - https://react-advanced-assignment.psamd.now.sh/
This assignment is designed for applicants who have a good knowledge and understanding of React and have developed/worked on atleast one React application.

What are we looking for?

Everything mentioned in the Assignment 1 and additionally:

- Ability of the applicant to learn a new React UI library and use its components in their app
- Handling events and working with forms
- Lifting State Up

Instructions

We want applicants to create the closest possible replica of the assignment 2 demo app. This app should be responsive (mobile, tablet and desktop) and will be tested on Chrome browser.
The entire app was designed using the Ant Design library. It is mandatory for applicants to use the library to design the Icons, Buttons, Cards, Grid, Modal and Form in the demo app.

Assignment 3 - Backend and Frontend

Deadline for submission: 7 days
DEMO - https://react-advanced-assignment.psamd.now.sh/
This assignment is designed for applicants who have google knowledge on both frontend and backend api's development.

What are we looking for?

Implement the assignment 2 task but with your own built api's using node js express mongodb/mysql

Instructions

We want applicants to create the closest possible replica of the assignment 2 demo app with self built api's using node, express, mongodb/mysql.

General Tips

- Feel free to use Google, StackOverflow or any other resource
- Examine the demo apps closely to determine all the features
- Open the data API link in your browser and examine the response schema
- For applicants attempting the Assignment 2, please read the documentation of Ant Design library carefully and thoroughly.
- Try to match the UI design of the demos for each assignment as closely as possible.
- Please feel free to get in touch with the Bytive Technologies team to clear any doubts related to the aforementioned instructions.