








Task Description:

Our company is focused on event management. Currently, we are looking for a front-end web developer to build our brand new website using React, firebase, react router, etc.

Website Theme:

Your website has to be related to event management. Here, we have given you a list of 7 different types of events. Choose one of these events from the following list to create your website. Your website should not have more than one event type. For example, if you choose Corporate Events, your website should have events related to Corporate Events Only; ✗ you must not add other events ✗ like Social, Entertainment, etc.

- Corporate Events 
 - Conferences and Seminars, Trade Shows and Expos, Team Building Workshops, Product Launches, Corporate Meetings, Award Ceremonies, etc.
- Social Events 
 - Weddings, Birthday Parties, Anniversaries, Engagement Parties, Retirement Parties, Baby shower, etc.
- Entertainment Events 
 - Concerts and Music Festivals, Theater and Performing Arts, Film Screenings and Premieres, Comedy Shows, Sports Events, etc.
- Community and Cultural Events 
 - Festivals, Parades, Carnivals and Fairs, Cultural Celebrations, Fundraisers and Charity Galas, etc.
- Educational and Training Events 
 - Workshops and Webinars, Career Fairs, Educational Conferences, Training Programs, etc.
- Health and Wellness Events 
 - Health Fairs, Yoga and Fitness Retreats, Wellness Expos, etc.
- Tech and Gaming Events 
 - Tech Conferences and Expos, Gaming Conventions, Hackathons, etc.

Main Requirements

1. Make sure your design and website idea is unique. First, finalize your idea (what type of website you want to build). Then google the site design or visit themeforest to get your website idea. [You can visit this blog to collect free resources for your website.](#) However, your website ✗ can not be related to your

previous assignments or any demo project displayed in the course or in our conceptual sessions ✕.

2. Home page will have a navbar, banner/slider, services, and footer. Implement active routes on NavBar.
3. Add two extra sections in the home page in addition to the 4 sections mentioned above.
4. The Services section will have at least 6 event services. You can put one or more services in a row based on your design. Each service should have a relevant name, image, price, short description and a specific button.
5. Clicking on the specific button will take the user to the service detail route. Each route should display detailed information of the service. What you will include in the detailed information is entirely upto you but make sure it is relevant.
6. The service detail route will be a private/protected route. Please make sure that if the user is not logged in, the private route redirects to the login page.
7. You Must implement Email and password based Authentication. This means, you will have to implement the Registration and the login page. Users should be able to toggle between Login and Registration view .Upon successful login or registration, you must show a toast/alert. Do not use the browser alert.`
⚠ Note: Do not enforce forget or reset password feature and the email verification method, as it will inconvenience the examiner. If you want, you can add email verification/forget the password after receiving the assignment result.

On the Registration page, display errors when:

The password

- is less than 6 characters
- don't have a capital letter
- don't have a special character

On the Login page, display errors when:

- password doesn't match
- email doesn't match

You can take the error message from firebase. You can show the error below the input fields or via alert/toast. If you use alert/toast, do not implement the browser alert.

8. Also, implement at least one extra login which could be (facebook, github, google, etc).

9. After reloading the page of a private route, the user should not be redirected to the login page.
10. Once logged in, the user name, profile picture and the logout button should appear on the header. If the user clicks on the logout button, make sure to log him/her out.
11. Add two more routes. Relevant to your website. These two routes will be private. You can put anything relevant to your site on these two routes.
12. Add a 404 page (not found page)

Bonus Requirements:

1. Answer to the questions given in the `questions_and_answers.md` file.
2. Add at least 5 meaningful Github commits. Meaningful `readme.md` file containing your website name and a link to your live site. Add at least five bullet points mentioning different features and functionalities of your website.
3. Make the website responsive. Make sure the site looks different on desktop and mobile responsive. Tablet responsive is optional.
4. Explore the [AOS package](#) and implement it your Homepage.

Additional information:

1. Create your own fake data. You should load the data from a `.json` file.
2. Host image on imgbb or any other hosting platform.
3. You are free to use any css library you want.
4. Environment variable is recommended but optional
5. Try to host your site on Firebase (netlify host will need extra configuration)

What to submit

1. Your github repository
2. Your live website link

Earn your Reward:

- The greatest joys of life aren't gifted; they're earned.

