Somesh Basak

Bangalore , India , 560087 | someshbasak2016@gmail.com | LinkedIn
GitHub | Portfolio

Career Objective

 Self-motivated and hardworking fresher seeking for an opportunity to work in a challenging environment to prove my skills and utilize my knowledge & intelligence in the growth of the organization.

Key Skills   *courses yet to be completed

Front End: HTML, CSS, Bootstrap, Figma, JavaScript, jQuery, React JS

Back End: Python, Express JS, Node JS, Command Line, GitHub

Databases: SQLite, MySQL, MongoDB*

Others: Wordpress, Hubspot, Convert PSD to HTML, Microsoft Office

Strengths

* Excellent communication skills.

* Ability to grasp the new skills quickly.

* Hard-working.

* Multitasking.

* Excellent knowledge of Core subjects.

Experience

* Full-Stack Web Development Intern

Solar Secure Solutions
Date: From Nov 9th,2022 - Jan 10th 2023 (2 months)
Task: In this internship built 10 Minor applications and 1 Major project. The main project is to build a **Food Delivery Website** with the help of HTML5, CSS3, JS, Bootstrap, used React js framework in frontend and in backend worked with Node js and Mysql database. Also Implemented the CRUD Operations and the Paypal Payment Gateway Session.

Certifications

~ <u>Nxtwave</u> -: REACT.JS: (LINK: <u>Click Here</u>)

~ <u>Nxtwave</u> -: NODE.JS: (LINK: <u>Click Here</u>)

~ <u>Great Learning</u>-: jQuery: (LINK: <u>Click Here</u>)

~ <u>Great Learning</u>-: WORDPRESS: (LINK: <u>Click Here</u>)

~ <u>Nxtwave</u> -: DEVELOPER FOUNDATIONS: (LINK: <u>Click Here</u>)

~ <u>Nxtwave</u> -: INTRODUCTION TO DATABASES (SQL):   (LINK: <u>Click Here</u>)

~ <u>Nxtwave</u> -: JAVASCRIPT ESSENTIALS: (LINK: <u>Click Here</u>)

~ <u>Nxtwave</u> -: PROGRAMMING FOUNDATIONS WITH PYTHON: (LINK: <u>Click Here</u>)

Education

Nxtwave Disruptive Technologies	**Mar'22 -Ongoing**
Industry Ready Certification in Full-Stack Development	
Maharaja Bir Bikram University	**2021**
Graduation - Bachelor of Commerce (H)	
Hindi Higher Secondary School	**2018**
Intermediate - Commerce	
Umakanta Academy (English Medium)	**2016**
Secondary School Of Education	

Projects

1. <u>**Nxt-Watch (Youtube Clone) :**</u>
   Developed a responsive Youtube Clone website where users have to first login and can see a list of trending videos, gaming videos etc and also can play, like, dislike and can save videos. Search Bar is also provided where users can search videos and also has two types of background themes black and white.
   Username: rahul / **Password:** rahul@2021

   <u>**Technologies used :**</u> React JS   **(Website link-:** <u>Click here</u>**)**

2. Nxt-Trendz (Shopping Website) :

Developed a shopping website in which we have to login by username and password and for authentication used the jwt-token. In the homepage there are three types of protected routes(Home, Product and Cart) so that no unauthenticated users can login. In the Product page there are many products in which we can search for the individual products in the search bar, categorize the products, sort and filter the products according to the prices and ratings, we can also add and remove the products from the cart. There is also an option of logout by removing the jwt token.
Username: rahul / **Password**: rahul@2021

Technologies used : React JS   **(Website link-:** Click here**)**

3. Todos Application :

Developed persistent todo application with CRUD operations to track list of tasks
  ● Displayed list of todos with HTML list elements, styled todo list using CSS, Bootstrap
  ● Implemented todo crud operations by using JavaScript event listeners and updated UI dynamically by using JavaScript DOM operations.
  ● Used Arrays, Objects and their methods during todolist CRUD Operations, Persisted todo list state on page reloads using local storage methods.

Technologies used : HTML, CSS, JS, Bootstrap   **(Website link-:** Click here**)**

4. IPL Dashboard:

When the app is opened, Home Route should be displayed -
  -> When the Home Route is opened :
      ● Make HTTP GET request to the teamsApiUrl
      ● *loader* should be displayed while fetching the data
      ● After fetching the data, the list of teams should be displayed
  -> When a team card in Home Route is clicked,
      ● Page should be navigated to the Team Matches Route with the URL
        `/team-matches/:id`
  -> When the Team Matches Route is opened :
      ★ Make HTTP GET request to the teamMatchesApiUrl with the team id to get the recent matches data of the team ex- `https://apis.ccbp.in/ipl/KKR`
      ★ *loader* should be displayed while fetching the data
      ★ After fetching the data, the team banner, latest match, and list of recent matches should be displayed

Technologies used : React JS   **(Website link-:** Click here**)**

5. <u>Password Manager</u>:

   Initially, the website input, username input, and password input should be empty and No Passwords View should be displayed.

      -> When non-empty values are provided for the website, username, and password and the Add button is clicked :-

- ★ A new password item should be added to the list of passwords
- ★ The passwords count should be incremented by one
- ★ The stars image should be displayed in the password items instead of the passwords
- ★ The value of the input fields for website, username, and password should be updated to their initial values
- ★ When the Show Password is checked, then the password should be displayed instead of the stars image

-> When a non-empty value is provided in the search input field, then password items whose website is matched with the search input value irrespective of the case should be displayed

-> When a non-empty value is provided in the search input field, and if the website of any password item does not match the value given in the search input, then No Passwords View should be displayed

-> When the delete button of a password item is clicked :

- ★ The respective password item should be deleted from the list of passwords
- ★ The passwords count should be decremented by one

-> When all password items are deleted, then No Passwords View should be displayed

    ***<u>Technologies used</u>* :** React JS   **(Website link-:** Click here**)**

6. <u>Food Munch</u> :

    Developed a responsive website for Food Store where users can see a list of food items, detailed information
    about a food item, offers.

- ● Designed page using following HTML structure elements like li, header, article, footer elements, and
  different bootstrap components to show different sections in the website and different bootstrap classes for responsiveness through the mobile-first approach.
- ● Implemented product youtube videos by using bootstrap embed and model component.

    <u>Technologies used</u> : HTML, CSS, Bootstrap   **(Website link-:** Click here**)**

7. <u>Match Game</u>:

The app has the following functionalities.

-> Initially :-
- ★ Score should be 0 and time should be 60 sec.
- ★ The image to be matched should have the src attribute value as the value of the key imageUrl from the first object in imagesList provided.
- ★ The Fruits tab should be active and the thumbnails with FRUIT as their category should be displayed.

-> The timer should start running backwards from the 60 sec.

-> When a tab is clicked, then the thumbnails in the corresponding category should be displayed.

-> When a thumbnail is clicked, if that is matched with the image to be matched :
- ★ Score is incremented by one.
- ★ The new image to be matched should be generated randomly among the value of the key imageUrl from imagesList provided.

-> When a thumbnail is clicked, if it is not matched with the image to be matched :-
- ★ The game should end, and the Scorecard view should be displayed.
- ★ When PLAY AGAIN button is clicked, then we should be able to play the game again -
  - ❖ The score and time values should be reset to 0 and 60 sec respectively
  - ❖ The image to be matched should reset to the value of the key imageUrl from the first object in imagesList provided
  - ❖ The active tab should reset to Fruits, and the thumbnails with FRUIT as their category should be displayed

-> When the timer reached 0 sec, then the game should end, and the Scorecard view should be displayed

<u>**Technologies used :**</u> React JS , HTML, CSS, JS   **(Website link-:** Click here**)**