

## CRUD Application Using ReactJS Hooks

Kanniga Parameshwari M<sup>1\*</sup>, Selvi K<sup>2</sup>, Rekha M<sup>3</sup> and Karthiga R<sup>4</sup>

<sup>1,2,3,4</sup>Department of Information Technology, R.M.K. Engineering College (An Autonomous Institution) Kavaraipettai – 601206

### Abstract

This Project is aimed to implement CRUD operations using the library called React.js which is the advanced version of the language Javascript. The question which pops out everyone's mind now is why React? There's lot of open – source framework/ library/ language that is available in today's internet. The purpose of using react is anyone with the little knowledge of javascript can easily learn react and implementation of this library is user-friendly. Also, there's many more packages react can easily get installed in NPM and run the application whatever changes in the code can do. React also delivers good UI for the developer and to create/develop new apps for future generation. There are more than 2000 developers and over one lakh sites making use of react.js among some of the most powerful are Instagram, Netflix, Whatsapp, Uber. It also integrates with any other javascript libraries or framework and speciality of react can make changes in the web browser without doing subsequent refresh of the pages. React can allow the developer to build any complex web application and more importantly mobile application. The CRUD is an acronym for Create, Read, Update and Delete which is very much essential for implementing any strong application with relational Database. Thus, the project results in Creating the user/users with their details, read the users task, updating if any new user is created through 'Create' operation, and finally delete the user/users list if not wanted for an application. Though the 'crud' functions for creating an application can be done in any language, the author specifically uses Javascript library called react.js which is one of the best platforms to create the single page user application interfaces. The overarching goal of this application is to bring easiness for any product development company in need of crud-user functions can stick to this.

**Keywords:** functions, user interfaces, react.js library, single page application

Received on 05 December 2023, accepted on 26 February 2024, published on 04 March 2024

Copyright © 2024 Kanniga Parameshwari M *et al.*, licensed to EAI. This is an open access article distributed under the terms of the [CC BY-NC-SA 4.0](#), which permits copying, redistributing, remixing, transformation, and building upon the material in any medium so long as the original work is properly cited.

doi: 10.4108/eetiot.5298

\*Corresponding author. Email: [mkp.it@rmkec.ac.in](mailto:mkp.it@rmkec.ac.in)

### 1. Introduction

The day-to-day life the application which are used by the customers must be robust, secure, faster and user-friendly in nature. Though there are plenty of app development library / frameworks available in internet, the reason behind what the language we use for implementing our project must be strong enough comparatively. The topic the author chosen here is CRUD operations/ functions using javascript stuffs, which is the main four components in developing any application is 'crud' and it is overall used in industries like office user list, gaming application, Movie tickets, online competition etc. In this Project, the author uses the

technologies like react.js, html, css, jquery, media query, hooks to make the rendering fast. Javascript used in web development is the world's biggest language to the developer who begin the startup career.

#### A. React.JS

React.js is an open-source powerful a versatile javascript library and reusable components makes UI interactive with user. Jordan Walke, a software engineer at Facebook, developed it. So initially Facebook and Whatsapp maintained for its newsfeed section later many products were used [1]. The user interfaces are created using the Javascript library. interactive and make quicker web applications in an efficient manner in less lines of code and make the non-developer itself to understand and learn easier.

JSX is an html or xml-based syntax used by react.js so that it's easy to learn when someone quite familiar with html and javascript as follows

```
Const x = "world";
```

```
Const word = <h1> Hello {n}</h1>
```

The features of React.js are:

1. Computing – The language can be used to debug the code without requiring the developer to predict an exact method to be followed because react itself will update and render the apt component.

2. Functions and Class based – The functional components are called state-less components because there is no path to track the state or lifecycle of the components unlike the class components which is the only method to track the state and lifecycle on a component. But after the introduction of React 16.8, the concept called Hooks come into existence which made the functional components no way lesser than the class components. The example for some hooks that are used mostly: useState, useEffect, useContext, useRef, useReducer, useHistory, useCallback and many more [2].

3. Document Object Model (DOM), also known as virtual DOM which represents the web page's structure where HTML elements can update their nodes and attributes and so on. The olden days DOM are difficult for the customer to update and switch to any other menus since the accessing of any icon in the page take some time to get rendered the whole page and then update to the user.

4. In react, there is a term called virtual Dom which means the changes or updation made in the web page will not directly affect the real dom, inturn makes the necessary changes in the virtual home that serves as the real home's blueprint. When all the changes are done by the user, the virtual compares all the changes then reflect the last changed updation to the real dom. This makes the rendering in server side easier also makes single page application more effectiveness.

5. Discover Once, Implement Anywhere – The perks of using react is that we don't need to assume what the rest of the stack in building a full stack project, because react itself can render on the server side using NODEJS and for mobile applications using REACT NATIVE. Also React is extended with Redux, Flux to make the web page UI looks clean and good-looking and use full-pledged manner.

B. jQuery and Ajax

jQuery is speed, tiny and high feature javascript. It mainly makes event handling, animations and asynchronous javascript in much efficient way and easy to access in any browser. Also ajax is acronym for asynchronous javascript xmlhttp which in turn used to create interactive web applications [3].

C. Hooks

Hooks is the concept that replace the classes components in react.js but provide the same functionality to functional components. The behavior of the functional components now does the same modularity as class component does also works fantastically to improve the readability of the nature of the code throughout the application. The two most

popular heavy lifters that replace maximum functionality needed by the classes are: useState and useEffect. useState consistently returns an array in two forms, with the first input being the current value of the state and second input is updated value of the first input [4].

```
import React, { useState } from 'react';
const usingState = (property) => {
const [current_state, set_current_State] = useState({fun:
'hope'});
const [input, setInput] = useState("");
...}
```

As the name suggests, Effect Hook plays effectively the role of side effects which may affect components and does not change at the time of rendering. The side effects operations mean the componentDidMount, componentDidUpdate, componentWillUnmount are the methods will get replaced by using Efect Hook.

```
import React, { useState, useEffect } from 'react';
function effectExample () {
const [count, setCount] = useState(0);
// componentDidMount and componentDidUpdate:
useEffect(() => {
// Update the document title using the browser API
document.title = `You are clicking ${count} times`;
});
```

```
return (
<div>
<p>You are clicking {count} times</p>
<button onClick={() => setCount(count + 1)}>
Click me
</button>
</div>
);
```

useEffect unmount here and also re-run the effect. We can use more than single effect in one component

```
useEffect(() => {
return () => {
};
});
```

We also have certain rules for using react hooks in the javascript functions,

(1) At the top level only, hooks can be called. It is not good practice call hooks inside loops, conditions or nested functions.

(2) Hooks can also be call only from React function components. From regular javascript functions, we should not call other valid place.

## 2. Literature Survey

Alok et al. [5] the author developed 'crud' operations in react.js using hooks concept solve a problem of functional changes in the application on big scale usecase. It uses the methods like GET, POST, PUT, DELETE to do operations

for creating the task, updating and deleting it in an effective manner.

Mayelson et al. [6] the author created the rendering of dom from server side and single page application were not much effective few years ago. Sometimes javascript purely can't handle the dependency graph and now advanced javascript make 100% satisfaction to the client and server. This is helpful in creating the very complex UI like blog layouts and submitting forms in banking application, student's database etc.

Arshad et al. [7] the author designed the size of the app scales are very difficult to maintain when building the large scale applications like Facebook, Instagram, Whatsapp are using overall world, it is the responsibility of the developer to not let the server down or issues in the app or problems in storage of data posted in the app, smooth scrolling purpose, the purpose of needs and search of the clients and the privacy for hacking prevention etc..

Xingwei [8] the author investigated on considering the above problems, even though there is many solutions in real world today, choosing this language based operations is fitting best and highly considered/ used all over the IT or any industry. So the developer can build the app without creating the above problems and use the frame work which fits in as a good solution.

From the above literature survey, it is observed that the limitations of developing a single page application depends on the browser compatibility and document object manipulation. In this research, a try has been made to give a clear picture of how the software developer has convenient, compatibility, user-friendly, do not repeating the code all the things are considered into attempt.

### 3. Proposed System

React's primary characteristics should be its component composition. Different writers write components at various workspaces for various application demands., but ultimately all should work well together. In react itself is highly reusable and code readability. There is no tool that can be designed 100% error free, react also has some changeability which may get updated in the future. For example [9], the introduction of state into a component makes changes in the component, that is nothing wrong in do it but for powerful feature they may enable some functional patterns in the future since react is already useful in many ways. The following is the best way to implement mathematical expression or equation using reactjs hooks which helps to resolve the exponential equation easily. But before that, it is

mandatory to install equation parser and equation resolver in node package manager.

`npm install -S react-equation equate-resolve`

```
ReactDOM.render((
  <
    <MathEquaOpt
      vari={defaultVars}
      func={defaultFunc}
      errHandle={defaultErrHandle}
    >
      <MathEqu
        res='10n + 1/2n * sin(π) + (42 m^2) / (8m)'
      />
      <MathEqu
        res='10n + 1/2n * sin(π) + (42 m^2) / (8m)'
      />
    </MathEquaOpt
  >
</>
), document.getElementById("root"));
```

Submit a request to amend our data. The group values API stability a lot. React is used to build more than 50,000 components at Facebook. React is widely used by numerous other businesses, like Twitter and Airbnb. They are typically hesitant to alter public APIs or behaviour for this reason [10]. Apart from this, react has a good developer experience, stability, interoperability, debugging, optimized for tooling, scheduling, runtime configuration. So, the proposed system could be an additional functional requirement or rendering methods or new hooks in use.

For example, `<Equate`

```
Result ='10 + z'
/>
// Renders a 10 + z
```

The maximum number of developers have credentials remote account like github, gitlab, netlify, heroku.com for pushing the code globally for their future reference or team work etc. so when taking the survey, we can analyze that which has number of stars or forks. Since online applications are booming day to day life, we developers are going to be in high dilemma of choosing which libraries to be used rather than how to develop the project. We can also render multiple and interconnected equations, variables and functions and equals are given underscore symbol `_` and the unit.

It is important to remind that, the order matters so we cannot pass the equation-function between one component and others. on the contrary we should use `Equationoptions` which is component name and `getOptionshelper`.

Table 1. Usage of Styles

| Framework      | Git Hub stars | Git Hub forks |
|----------------|---------------|---------------|
| Material UI/UX | 70.2K         | 25.4K         |
| Ant Design     | 75.8K         | 34.5K         |

```

<EqContext render={ (equation) => (
  <
    {equate('x = 2')} Renders x = 2 and defines a
    {equate('y = 5x =')} Renders y = 5x = 10 and
defines y
    {equate('z = 1/y = _ %')} Renders z = 1/y = 10%
and defines z
    {equate('f(a) = a^2')} Renders f(a) = a^2 and
defines f(a)
    {equate('2x + f(x) =')} Renders 2x + f(x) = 8
  </>
)} />

```

The above table 1 will make you better understanding of how far the react libraries is widely globally used. There's a lot of libraries introduced in last ten years using javascript and we can't argue which is better than others. It's a developer choice purely to select the library in which he is comfortable to develop the app.

Table 2. Difference between Angular and React

| Choose Angular.js   | Choose React.js                       |
|---|---------------------------------------|
| When Java, C# and previous version of Angular                   | When you use HTML, CSS and JavaScript |
| When you use large scale feature-rich app                       | When highly customized specific app   |
| When you are ready to use solution and need height productivity | When using App with multiple events   |
| When you regulate the app size                                  | It creates sharable component in app  |

Why choosing react.js over angular.js is shown in the table 2. The need of creating a big app with multiple events and large scale or customizable app will leads to pick react where the flow of writing the code makes flexible for the developer.

### 3.1. System Model and Architecture

The life cycle methods in react is most preferable to learn firstly, because all the components in react go through the

life cycle of events shown in fig1. It is classified into two types: Class based components and function-based components and function-based components. The life cycle method includes three main steps [11]: Mounting, Update, Unmount. Mounting can be compared as Update represents the growth of the component, whereas Unmount represents the death of the component. The render () method in class component is pure functions which means there is no side effects that is the function should return same output when same inputs are passed. The functional component doesn't have these steps instead it has 'hooks' concepts where the useEffect() can do all this three steps.

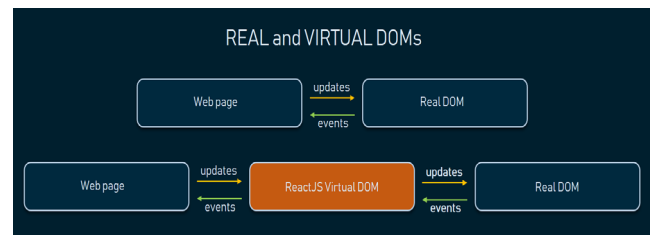


Figure 1. Difference Between Real and Virtual Dom

### 3.2. Lifecycle Methods

componentDidMount() – This method is called when the component is mounted and ready. This is the place where you can initiate API calls and load data from remote endpoint. The state and setState method is used at this point where state is the current state and setState will update the current state But before the browser updates the UI, it will occur.

componentDidUpdate() – The update method is called when we set the setState method updated but make sure that if setState is not updated correctly then it will create infinite loop. So it's essential to compare the present praisers the past praisers as follows – if (this.props.userName != prevProps.userName) then we can fetch the props.userName data. Consider the below diagram for clear understanding fig2.

componentWillUnmount() – the current compliments the prior accolades i.e., where all the clean ups are done here if needed. Also at this stage, we cannot modify the setState, oly we need to clear the cache storage, cancelling APIcalls[12].

Abbreviations and Acronyms

- CRUD – Create, Read, Update, Delete
- React.js – React Javascript
- HTML – Hyper Text Markup Language
- CSS – Cascading Style Sheets
- XML – eXtensible Markup Language
- JSX – Javascript + HTML
- AJAX – Asynchronous Javascript Xml
- DOM – Document Object Model
- UI – User Interface
- SSR – Server Side Rendering
- NPM – Node Package Manager

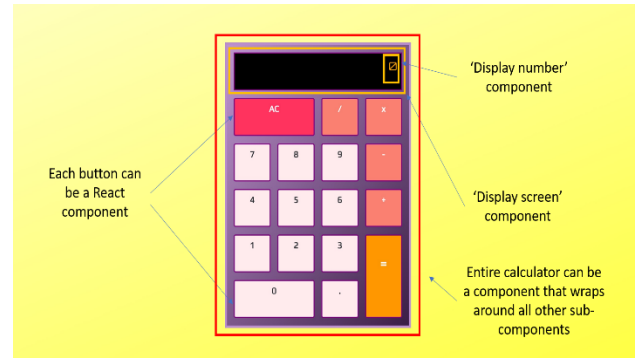


Figure 3. React Components working of Calculator

The reason behind using react.js instead of vanilla javascript is the functionalities are partitioned between javascript and html individually and there is easy way of doing in older days with less complexity code but nowadays the web application and mobile application are having large code complexity and the headache starts for developer i.e., the separation of js file and html file is hard to keep track of where the code starts and ends whereas the react.js have the magic of writing html inside javascript code called JSX and that made easy to follow-ups even when there is code complexity. The one word that explains we switch from vanilla javascript (pure javascript code) to react.js code is simplicity which in turn helps you to build complex apps with much less code comparatively.

Though reactjs can cause issues at compile time rather than run time but client experience keeps updating with new framework react also does get upgraded with its new version like extra tooling requires for finishing development and it is open to all developer world level interest. Apart from that react is picked highly and commonly among developers globally.

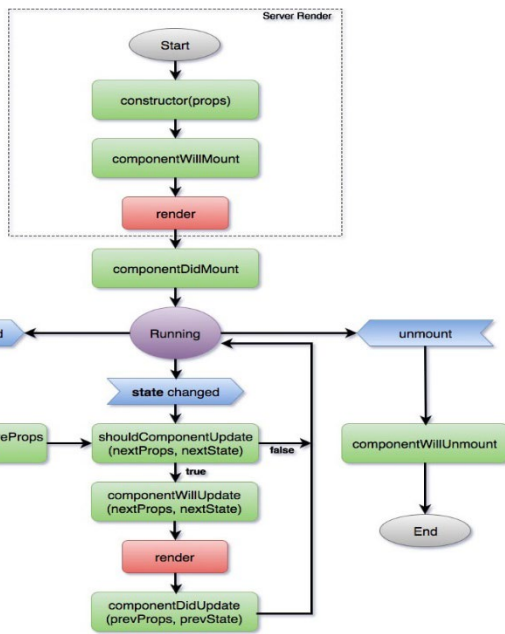


Figure 2. DoM Architecture

3.3. Components

React itself written on javascript, it is the super flexible and efficient for the developer to build the app using breaking the complex code into less and isolated number of codes and they are called “components” and it give more work on front page of a website [13]. Also, it can work on backend using Node.js The basic knowledge beginning with react is App.js where all the components imported and that’s the main root component. Those components are the heart of the library because it is reusable anywhere within the code and purely independent of each other. Following diagram fig3 explains how calculator gets split into react components.

3.4. Rest API

To begin with we need to create react app using the commands: `npx create-react-app crud_app_reactjs` (or) `yarn create react-app crud_app_reactjs`. This will install the react app and the command followed to start the application is `cd crud_app_reactjs` and `npm start` or `yarn start`. There is so much dependency created once the app starts which may have n number of file can ignore the some not in use [14]. The rest api is the mock api used here can be differentiated using `‘/’` the server has localhost where the app runs and `/users`, `/updateUser` redirects to the corresponding links, then `/id` explains the particular id to display or delete only that user and the url can be as follows –



| REST API | URL                         |
|----------|-----------------------------|
| GET      | http://localhost:4000/Users |
| GET      | /students/updateUser/id     |
| POST     | /students/createUser        |
| PUT      | /students/updateUser/id     |
| DELETE   | /students/deleteuser/id     |

```
function App() {
  return (
    <div>
      Crud Operations using React
    </div>
  );
}
```

And we can create the form in separate component called createUser.js and do create operations of the user and most importantly we are supposed to import the package for the form, button and submit which is responsive in mobile and desktop applications as follows

```
import { Link, Route, Switch } from "react-router-dom";
import AppBar from "@material-ui/core/AppBar";
```

```
function App() {
  return (
    <div>
      <AppBar position="static">
        <nav>
          <Link to="/">HOME</Link>
          <Link to="/users">USERS</Link>
        </nav>
      </AppBar>
      <Switch>
        <Route exact path="/">
          <Home />
        </Route>
        <Route path="/users">
          <Users />
        </Route>
      </Switch>
    </div>
  );
}
```

```
function Home() {
  return (
    <div>
      <h2>This page is related to CRUD</h2>
      <p>Go to user page to create and delete the users in the user-list.</p>
    </div>
  );
}
```

To display the createUser.js components in the web browser, we need to import that component in App.js as

this is the root component of all other component in the library[15].

```
const [addusername, setaddusername] =
useState("");
const [adduserpic, setadduserpic] = useState("");

function getUsers(){
  fetch(`https://6100f45a1d56e10017394c85.mockapi.io/users`,{
    method:"GET",
  })
  .then(data =>data.json())
  .then(users =>setuserlist(users))
}
```

```
function createUser(){
  fetch(`https://6100f45a1d56e10017394c85.mockapi.io/users/`,{
    method:"POST",
    headers:{
      "content-Type":"application/json",
    },
    body:JSON.stringify({name : addusername,
pic : adduserpic})
  })
  .then(data =>data.json())
  .then(()=>getUsers());
}
function deleteUser(id){
  fetch(`https://6100f45a1d56e10017394c85.mockapi.io/users/${id}`,{
    method:"DELETE",
  })
  .then(data =>data.json())
  .then((user)=>console.log(user))
  .then(() => getUsers());
}

useEffect(()=>{
  getUsers()
}, []);
```

### 3.5. CRUD Operation

Once we imported the createUser.js component in app.js we are expected to get the UI as shown fig3. The mapping is done using map() function to map the first name, last name and check box conditions also can filter the needed data using filter() functions and reduce() function. We have a table of columns how much we insert but can be deleted or removed whenever needed it depends on which stage additionally the developer can also break down the components and reuse the same components in the app wherever necessary with less number of coding. Import the css file for styling purpose in between or can use

inline styling for small changes [10]. The localhost 3000 is the port for frontend app run and backend runs in localhost 5000.

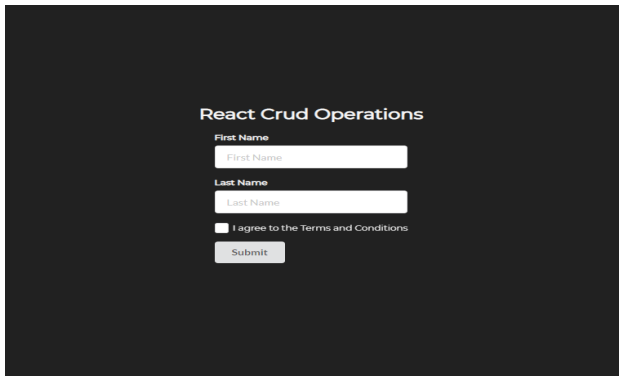


Figure 4. Crud Operations

The importance of using hooks is explained here, useState is the magical hook which makes you keep two states. One is the current state and other state is to update the previous state when the student updates the current state. For example: if the user clicks the like button in the Instagram app, it needs to be incremented for each user like or dislike count button which is updating the state of the previous.

```
*import React, { useState } from 'react';
const [firstName, setFirstName] = useState("");
const [lastName, setLastName] = useState("");
const [checkbox, setCheckbox] = useState(false);
```

Next step is to create the route path for the component, when we enter the user detail and click checkbox and click create button, it has to be redirected to the next page. For that, router-dom is necessary which redirects hazel free path.

```
*import { BrowserRouter as Router, Route } from 'react-router-dom'
<Router>
  <div>
    <Route exact path='/createUser'
component={CreateUser} />
  </div>
</Router>
```

We have set the path of Create to '/createUser'. So, if we go http://localhost:3000/createUser, we will see the create User page. Similarly, we need to import the readUser component and updateUser component in app.js and set the router path using react router dom.

```
import CreateUser from './components/createUser';
import ReadUser from './components/readUser';
import UpdateUser from './components/updateUser';
import { BrowserRouter as Router, Route } from 'react-router-dom'
```

```
function App() {
```

```
return (
  <Router>
  <div className="main">
    <h2 className="main-header">React Crud
Operations</h2>
    <div>
      <Route exact path='/create' component={CreateUser} />
    </div>
    <div style={{ marginTop: 20 }}>
      <Route exact path='/read' component={ReadUser} />
    </div>
    <Route path='/update' component={UpdateUser} />
  </div>
  </Router>
);
}
export default App;
```

From the above code, it is clear that the path to readuser and updateuser is set and can be accessed from the browser if you go to http://localhost:3000/readUser and on http://localhost:3000/updateUser. The next important and much useful hook is useEffect , helps to send the GET request in this project (i.e., used to perform the side effects in our components) Following is the sample output of how we created the user and read the user list and delete the user. So crud operation is the highlight in all the industry which have the list of database.

```
useEffect(() => {
  axios.get('https://60fbca4591156a0017b4c8a7.mockapi.io/fakeData')
  .then((response) => {setAPIData(response.data);})
}, [])
```

### CRUD Database

| ID | First   | Last       | Email        | Phone        | Location           | Hobby       | Actions                            |
|----|---------|------------|--------------|--------------|--------------------|-------------|------------------------------------|
| 35 | Jane    | Smith      | js@gmail.com | 503-555-5555 | Portland, Oregon   | Photography | <span>Edit</span> <span>Del</span> |
| 36 | Tristan | Rodriguez  | tr@gmail.com | 201-555-5555 | New York, New York |             | <span>Edit</span> <span>Del</span> |
| 37 | Andrea  | Jones      | aj@gmail.com | 503-555-5555 | Portland, Oregon   | Programming | <span>Edit</span> <span>Del</span> |
| 38 | Elliott | Richardson | er@gmail.com |              | Boise, Idaho       |             | <span>Edit</span> <span>Del</span> |
| 39 | June    | Bug        | jb@gmail.com |              |                    | Cooking     | <span>Edit</span> <span>Del</span> |

Download CSV Add Item

Figure 5. Database Crud operations

Here, we use axios to fetch the data from the rest API usually the axios package simply installed npm i axios and it is used to send asynchronous HTTP request to rest endpoints. Axios is very useful in performing CRUD operations in react.js. Destructuring is happened like array of objects can be extracted into what exactly we needed. For example,

```

Const gadgets = ['iphone', 'samsung', 'firebolt']
// old way
const mob = gadgets[0];
const tv = gadgets[1];
const smartwatch = gadgets[2];
//destructuring
Const [ mob, tv, smartwatch ] = gadgets;

```

The next step is update api and we need to create update functions as well. That api have different from create api and have the PUT submit a request to amend our data [16].

```

const updateAPIData = () => {
  axios.put(`https://60fbca4591156a06b234c8a7.mockapi.io/fakeData/${id}`, {
    firstName,
    lastName,
    checkbox
  })
}

```

Same can be followed for delete operations. The changes in the fundamentals web page application are not easy task, developers suffer with dom manipulation changes throughout the app and also making the web page or mobile app making good looking with responsive outlook is challenging task for any developer.

```

const onDelete = (id) => {
  axios.delete(`https://60fbca4591156z1127b4c8a7.mockapi.io/fakeData/${id}`)
}
const inputData = () => {
  axios.get(`https://60fbca4591156a0017b4c8a7.mockapi.io/fakeData`)
    .then((inputData) => {
      setAPIData(inputData.data);
    })
}

```

## 4. Conclusion

Finally, this project explains more about why react.js and what is crud operations using react.js. To summarize, react.js has come at the right time or much needed time for developers helping in building highly engaging user interface and mobile apps with less amount of code and faster time. The main advantage of react is it is SEO friendly i.e., google or other search engine can crawl each page of the web page developed by react and interpret the content effective manner and index in database called search engine optimization.

As we all know react learning is easy when you have a little knowledge in javascript hence it has rapidly become popular that leads to today's hot favorite to many developer and business owners. So, it is much needed of the hour for the developers to learn react.js based technologies and use the functionalities in react at the

proper manner to sustain the current world industries, due to its many numbers of pros and very very less number of cons in building it. The react has some major advantage as follows-

**Stability** – The stability of the application developed for API is very much valued by react author. They guide the deprecation caused during the building of an app.

**Interoperability** – React also has combined with any other framework in the same web page built example facebook server side component system is XHP.

**Developer Experience** – Developer can experience the ReactDevTools in firefox and chrome extension using inspection of react component tree. They also give warnings if you ladder the tags which is not understand by the browser.

**Debugging** – When there is something goes wrong it is important to trace the issue to its source code for example if there is any wrong in the state, then the setState() function must have a bug, or if the props are incorrect, you can see them in the react tree component and traverse the errors.

Thus, the author, I conclude by saying even though there are many frameworks that can run crud operations I need for going crud operations based on react.js is explained well in this project. Angular and vue.js are also topmost front end development tool that is equally spread similar to react.js but the major reason for using react.js is it has very easy code readability and code reusability. Doing the crud operations in any app involves important task where all the business and industries are in need of list of users, list of provisions, list of students, list of clients, list of foods, list of movies, list of hospitals, list of colleges etc. so its common that we are using crud operations in our daily life and it must be built with ease of access i.e., hazel free functions, smooth scrolling and so much. That's we need to make the UI looks attractive and interactive with users' application and all this can be achieved by react.js in a short period of time if we have good skills and knowledge in the javascript library.

## 5. Future Work

As discussed from the beginning of this project to till now, all that highlighted is doing crud functions based on react.js is ease of learning comparatively other front-end framework. Also we can strongly say that react.js for sure will not diminish many years in future because according to stack overflow which is a leading software people using tool analyzed that react.js is the number one library used by software developers across the globe. What else to explain further? Developers can use the library at the right time and gain knowledge of how to implement and start building app using react.js, make use of all the operations and functions used by this library for better performance. The report says not only United States has seen the visibility of growth in this technology also developing countries like India also has a drastic change



in the role of front-end development field. Post COVID'19 the post for opening front end engineer and react.js programmers have increased globally and so there is no panic to learn this language presently which helps the developers to lead and dominate many more years in future on a global scale.

The analysis made through the papers published related to this conveys that using reactjs based operations or applications are as easy as developer's daily task than doing vanilla javascript which is pure js code file. The usage of reactjs in fig 6 shown for developing a front-end website development is increased at its peak in a very right time and short time of introduction not only in countries like United States but also developing countries like India. It means that the openings for front end developer in reactjs is wanted more in yearly basis according to the analysis made, so the crud operation done using vanilla javascript is slower and loads in different files whereas reactjs loads in same jsx files and most importantly gets executed and refreshed once saving a file automatically.

The vanilla javascript loading website page takes 1 full second whereas the reactjs operations take 100 milliseconds depends on the project or app we are developing, and it can easily be viewed in DEV-TOOLS i.e., developer tools where each request, response of api can be analyzed.

The leading countries that use reactjs for developing web-based applications are: United States (1,50,975) China (45,291) Japan (42,453). The top three countries use the react library widely and few other countries like Indonesia, Russia, UK, Germany, Spain also rapidly and wisely use around 25+ thousand websites whereas the rest of the world stands 3 lakhs websites nearly.

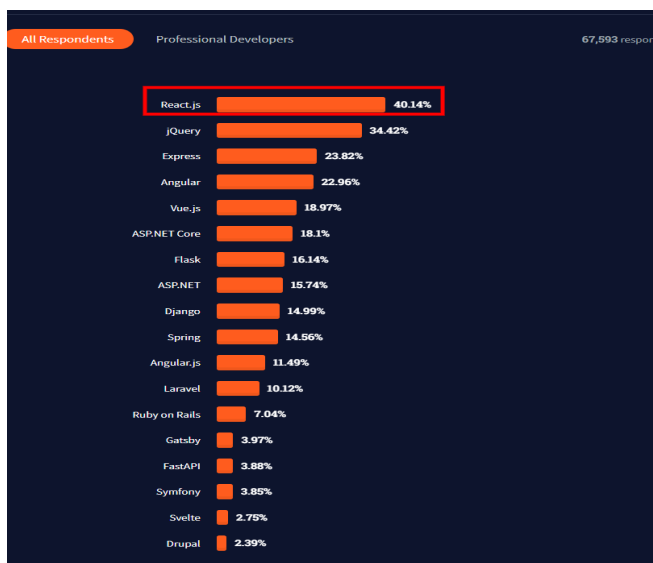


Figure 6 Survey of the languages

So, through the analysis made I'm hereby concluding that for sure can user reactjs as tool for developing front end website because users now can't even sit and wait for 4 seconds if the page gets loading slowly since the world is going faster so as the application developing tool must be.

Table 3. Comparison of Old and Updated JS

| Language           | DOM Rendering | Browser  |
|--------------------|---------------|--|
| Vanilla javascript | One sec       | Internet Explorer, Firefox, chrome, and Opera. |
| Reactjs            | 100 ms        | Edge, Firefox, Chrome, Safari, etc)            |

### References

- [1] Pratik Sharad Maratkar, Protibha Adkar. ReactJS – An Emerging Frontend Javascript Library, issued Iconic Research And Engineering Journals. 2021 ; 4(12) : 99-102.
- [2] Bhupati venkat sai indla, yogeshchandra puranik. Review on ReactJS” International journal of trend in scientific research and development(IJTSRD). 2021 ; 5(4) : 1137-1139.
- [3] Ritwik, Anitha Sandeep.: ReactJS and Front End Development International Research Journal of Engineering and Technology (IRJET) 2020 ; 7(4) : 3676-3679.
- [4] Prateek Rawat, Archana N. Mahajan - ReactJS: A Modern Web Development Framework 2020 ; 5(11) : 698-702.
- [5] Alok Kumar Srivastava1, Vaishnavi Laxmi2 , Payal Singh3 , Km Pratima4 , Vibha Kirti5. React JS (Open Source JavaScript Library). IJIRT. 2022 ; 8(9) : 342-346.
- [6] Mayelson de Sousa, Alexandrino Gonçalves.: Human portal – A reactjs case study 15th Iberian Conference on Information Systems and Technologies. 2020 .
- [7] Arshad Javeed.: Performance Optimization Techniques for Reactjs, IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT) 2019 ; 1067-1073.
- [8] Xingwei Zhou, Wenshan Hu, Guo-Ping Liu.: React-Native based Mobile App for online Experimentation-39th Chinese Control Conference (CCC) 2020.
- [9] Fu Kaifang.: Design and implementation of an instant messaging architecture for mobile collaborative learning” Computing, Communication, Control, and Management, CCCM 2009. ISECS International Colloquium on, 2009 ; 3: 287-290.
- [10] Basques.K.Performance analysis reference <https://developers.google.com/web/tools/chrome-devtools/evaluate-performance/> reference - 2020.
- [11] Hugo Brito; Anabela Gomes; Álvaro Santos; Jorge Bernardino, Javascript in mobile applications: React native vs ionic vs Nativescript vs native development” - 13th Iberian Conference on Information Systems and Technologies (CISTI). 2018.

- [12] Archana Bhalla<sup>1</sup>, Shivangi Garg<sup>2</sup>, Priyangi Singh<sup>3</sup>.: PRESENT DAY WEB-DEVELOPMENT USING REACTJS, International Research Journal of Engineering and Technology (IRJET) 2020 ; 7(5) : 1154-1157.
- [13] Hugo Brito; Alvaro Santos; Jorge Bernardino; Anabela Gomes.: Learning Analysis of Mobile Javascript frameworks - 14th Iberian Conference on Information Systems and Technologies (CISTI) 2019.
- [14] Naimul Islam Naim.: Reactjs : An open source javascript library for front end development Thesis 2017.
- [15] Yongkang xing , Jaipeng Huang, Yong yao lai. Research and analysis of the front end framaworks and libraries in E-business development. 11<sup>th</sup> International Conference. ICCAE. 2019.
- [16] Alluri Harshika.: Employee management system using spring boot and react JS, International Journal of Advance Research, Ideas and Innovations in Technology 8.2; 2022.
- [17] Mukthapuram Praneeth Reddy<sup>1</sup>, Sujata Priyambada Mishra<sup>2</sup>. Analysis of Component Libraries for React JS. IARJSET. International Advanced Research Journal in Science, Engineering and Technology. 2021 ; 8(6) : 43-46.
- [18] Heonsik Joo. A Study on understanding of UI and UX, and Understanding of Design According to User Interface Change. International Journal of Applied Engineering Research ISSN. 2017 ; 12(20) : 9931-9935
- [19] Shpetim Kadrija, Agon Memeti, Shkurte Luma-Osmani, "Development of mobile app through React Native hybrid framework", 2022 11th Mediterranean Conference on Embedded Computing (MECO). 2022 ; 1-6
- [20] Vipul Kaushik, kamali Gupta, Deepali Gupta. React Native Application Development. 4th International Conference on Cyber Security (ICCS) 2018 ; 461-467.
- [21] Kamal Kishore, Shanu Khare, Vaibhav Uniyal, Sahil Verma, "Performance and stability Comparison of React and Flutter: Cross-platform Application Development", 2022 International Conference on Cyber Resilience (ICCR). 2022 ; 1-4.