Web application development

(Introduction to Basic React)

Instructor: Tran Vinh Khiem

September 1st, 2022
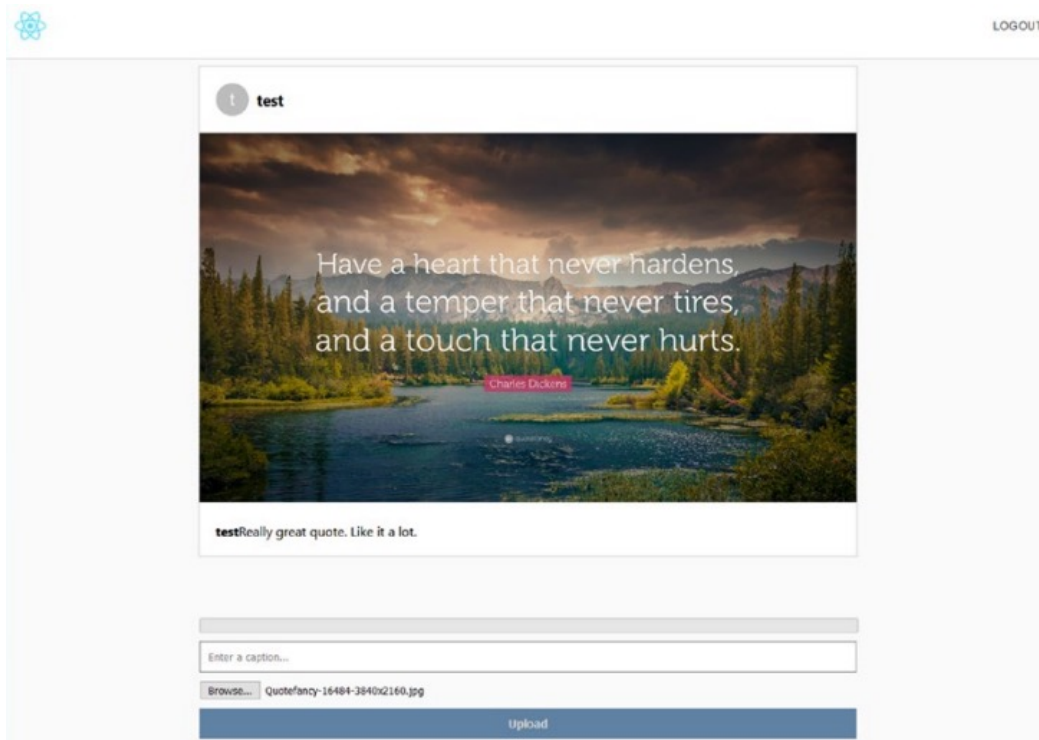
S³T

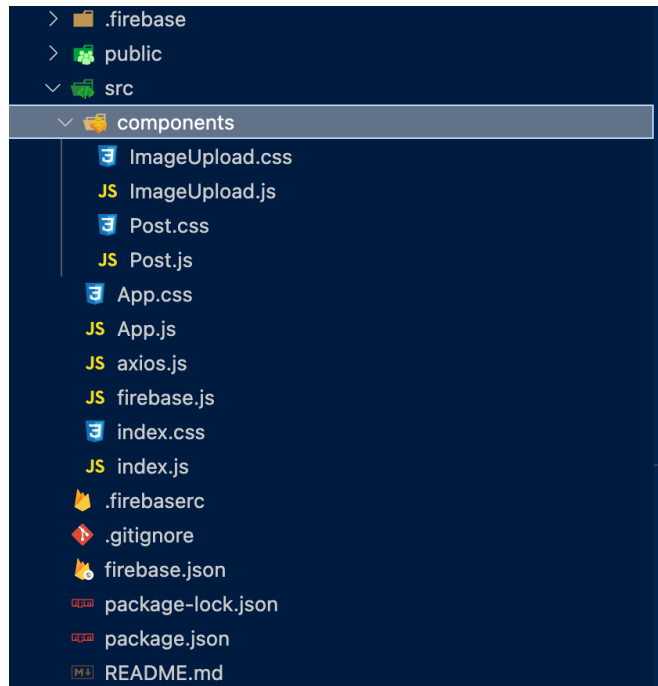*Smart Software System Team*

"We love what we do and we do what our clients love & work with great clients all over the world to create thoughtful and purposeful websites."
— ProWeb365

Basic React – Exercise 1 – Facebook "phake"

Basic React – Exercise 1 – Facebook "phake" - Structure


```
> .firebase
> public
∨ src
  ∨ components
      ImageUpload.css
      ImageUpload.js
      Post.css
      Post.js
  App.css
  App.js
  axios.js
  firebase.js
  index.css
  index.js
  .firebaserc
  .gitignore
  firebase.json
  package-lock.json
  package.json
  README.md
```



```
package-lock.json
package.json
postModel.js
server.js
```

Basic React – Exercise 1 – Facebook "phake"- Set up firebase

```
import './App.css';
import Post from './components/Post';
import React, { useState, useEffect } from 'react';
import { makeStyles } from '@material-ui/core/styles';
import Modal from '@material-ui/core/Modal';
import { Button, Input } from '@material-ui/core';
import { auth } from './firebase';
import ImageUpload from './components/ImageUpload';
import axios from './axios'
import Pusher from 'pusher-js'

const pusher = new Pusher('5601e06d285cef4e6836', {
  cluster: 'ap2'
});

function getModalStyle() {
  const top = 50;
  const left = 50;

  return {
    top: `${top}%`,
    left: `${left}%`,
    transform: `translate(-${top}%, -${left}%)`,
  };
}

const useStyles = makeStyles((theme) => ({
  paper: {
    position: 'absolute',
    width: 400,
    backgroundColor: theme.palette.background.paper,
    border: '2px solid #000',
    boxShadow: theme.shadows[5],
    padding: theme.spacing(2, 4, 3),
```

```
}));

function App() {
  const classes = useStyles();
  const [modalStyle] = React.useState(getModalStyle);
  const [open, setOpen] = useState(false);
  const [username, setUsername] = useState('');
  const [email, setEmail] = useState('');
  const [password, setPassword] = useState('');
  const [user, setUser] = useState(null);
  const [openSignIn, setOpenSignIn] = useState(false);
  const [posts, setPosts] = useState([]);

  useEffect(() => {
    const unsubscribe = auth.onAuthStateChanged(authUser => {
      if(authUser) {
        const setUser: (value: React.SetStateAction<null>) => void
        setUser(authUser)
      } else {
        setUser(null)
      }
    })
    return () => {
      unsubscribe()
    }
  }, [user, username])

  const fetchPosts = async () => {
    await axios.get("/sync").then(response => setPosts(response.data))
  }

  useEffect(() => {
    const channel = pusher.subscribe('posts');
    channel.bind('inserted', (data) => {
```

```
    channel.bind('inserted', (data) => {
      fetchPosts()
    });
  }, [])

  useEffect(() => {
    fetchPosts()
  },[])

  const signUp = (e) => {
    e.preventDefault()
    auth.createUserWithEmailAndPassword(email, password)
      .then(authUser => authUser.user.updateProfile({ displayName: username }))
      .catch(error => alert(error.message))
    setOpen(false)
  }

  const signIn = e => {
    e.preventDefault()
    auth.signInWithEmailAndPassword(email, password)
      .catch(error => alert(error.message))

    setOpenSignIn(false)
  }

  return (
    <div className="app">
      <Modal open={open} onClose={() => setOpen(false)}>
        <div style={modalStyle} className={classes.paper}>
          <form className="app__signup">
            <center>
              <img className="app__headerImage" src="logo192.png"        alt="Header" />
            </center>
            <Input placeholder="username"
```

5

Basic React – Exercise 1 – Facebook "phake"-App

```jsx
return (
  <div className="app">
    <Modal open={open} onClose={() => setOpen(false)}>
      <div style={modalStyle} className={classes.paper}>
        <form className="app__signup">
          <center>
            <img className="app__headerImage" src="logo192.png"        alt="Header" />
          </center>
          <Input placeholder="username"
            type="text"
            value={username}
            onChange={e => setUsername(e.target.value)}
          />
          <Input placeholder="email"
            type="text"
            value={email}
            onChange={e => setEmail(e.target.value)}
          />
          <Input placeholder="password"
            type="password"
            value={password}
            onChange={e => setPassword(e.target.value)}
          />
          <Button type="submit" onClick={signUp}>Sign Up</Button>
        </form>
      </div>
    </Modal>
    <Modal open={openSignIn} onClose={() => setOpenSignIn(false)}>
      <div style={modalStyle} className={classes.paper}>
        <form className="app__signup">
          <center>
            <img className="app__headerImage"
              src="logo192.png"
              alt="Header"
```

```jsx
          </center>
          <Input placeholder="email"
            type="text"
            value={email}
            onChange={e => setEmail(e.target.value)}
          />
          <Input placeholder="password"
            type="password"
            value={password}
            onChange={e => setPassword(e.target.value)}
          />
          <Button type="submit" onClick={signIn}>Sign In</Button>
        </form>
      </div>
    </Modal>
    <div className="app__header">
      <img className="app__headerImage" src="logo192.png" alt="Header" />
      {user ? <Button onClick={() => auth.signOut()}>Logout</Button> : (
        <div className="app__loginContainer">
          <Button onClick={() => setOpenSignIn(true)}>Sign In</Button>
          <Button onClick={() => setOpen(true)}>Sign Up</Button>
        </div>
      )}
    </div>
    <div className="app__posts">
      {posts.map(post => (
        <Post
          key={post._id}
          username={post.user}
          caption={post.caption}
          imageUrl={post.image}
        />
      ))}
    </div>
```

```jsx
      {user?.displayName ? <ImageUpload username={user.displayName} /> : <h3 className="app__notLogin">Need to login to upload</h3>}
    </div>
  );
}

export default App;
```

Basic React – Exercise 1 – Facebook "phake"-App

```css
.app {
  display:grid;
  place-items: center;
  background-color: ■#fafafa;
}

.app__header{
  display: flex;
  justify-content: space-between;
  position: sticky;
  top: 0;
  z-index: 1;
  width: 100%;
  background-color: ■white;
  padding: 20px;
  border-bottom: 1px solid ■lightgray;
  object-fit: contain;
}

.app__notLogin{
  margin-bottom: 20px;
}

.app__loginContainer{
  margin-right: 10px;
}

.app__posts {
  padding: 20px;
}
```

```css
.app__headerImage {
  object-fit: contain;
  margin-left: 10px;
  height: 40px;
}

.app__signup {
  display: flex;
  flex-direction: column;
}
```

Basic React – Exercise 1 – Facebook "phake"-Post

```jsx
import React from 'react'
import './Post.css'
import { Avatar } from '@material-ui/core'

const Post = ({ username, caption, imageUrl }) => {
    return (
        <div className="post">
            <div className="post__header">
                <Avatar
                    className="post__avatar"
                    alt={username}
                    src="/static/images/avatar/1.jpg"
                />
                <h3>{username}</h3>
            </div>
            <img className="post__image" src={imageUrl} alt="React" />
            <h4 className="post__text"><strong>{username}</strong>{caption}</h4>
        </div>
    )
}

export default Post
```

Basic React – Exercise 1 – Facebook "phake"-Post

```css
.post {
    background-color: ▪white;
    max-width: 800px;
    border: 1px solid ▪lightgray;
    margin-bottom: 45px;
}
.post__image {
    width: 100%;
    object-fit: contain;
    border-top: 1px solid ▪lightgray;
    border-bottom: 1px solid ▪lightgray;
}

.post__text {
    font-weight: normal;
    padding: 20px;
}

.post__header {
    display: flex;
    align-items: center;
    padding: 20px;
}
.post__avatar {
    margin-right: 10px;
}
```

Basic React – Exercise 1 – Facebook "phake"- Upload Images

```jsx
import React, { useState } from 'react'
import './ImageUpload.css';
import { storage } from "../firebase";
import axios from '../axios'

const ImageUpload = ({ username }) => {
    const [image, setImage] = useState(null)
    const [progress, setProgress] = useState(0)
    const [caption, setCaption] = useState('')
    const [url, setUrl] = useState("");
    const handleChange = e => {
        if(e.target.files[0]) {
            setImage(e.target.files[0])
        }
    }

    const handleUpload = () => {
        const uploadTask = storage.ref(`images/${image.name}`).put(image);
        uploadTask.on(
            "state_changed",
            (snapshot) => {
                const progress = Math.round(
                    (snapshot.bytesTransferred / snapshot.totalBytes) * 100
                );
                setProgress(progress);
            },
            (error) => {
                console.log(error);
            },
            () => {
                storage
                    .ref("images")
                    .child(image.name)
                    .getDownloadURL()
```

```jsx
                    .getDownloadURL()
                    .then((url) => {
                        setUrl(url);

                        axios.post('/upload', {
                            caption: caption,
                            user: username,
                            image: url
                        })

                        setProgress(0);
                        setCaption("");
                        setImage(null);
                    });
            }
        );
    };

    return (
        <div className="imageUpload">
            <progress className="imageUpload__progress" value={progress} max="100" />
            <input
                type="text"
                placeholder="Enter a caption..."
                className="imageUpload__input"
                value={caption}
                onChange={e => setCaption(e.target.value)}
            />
            <input className="imageUpload__file" type="file" onChange={handleChange} />
            <button className="imageUpload__button" onClick={handleUpload}>Upload</button>
        </div>
    )
}
```

10

Basic React – Exercise 1 – Facebook "phake"- Upload Images

```css
.imageUpload {
    display: flex;
    flex-direction: column;
    max-width: 800px;
    width: 100%;
    margin: 10px auto;
}
.imageUpload__progress{
    width: 100%;
    margin-bottom: 10px;
}
.imageUpload__input{
    padding: 10px;
    margin-bottom: 10px;
}
.imageUpload__file {
    margin-bottom: 10px;
}
.imageUpload__button {
    border: none;
    color: lightgray;
    background-color: #6082a3;
    cursor: pointer;
    padding: 10px;
    font-weight: bolder;
    font-size: 0.9rem;
}
.imageUpload__button:hover {
    color: #6082a3;
    background-color: lightgray;
}
```

Basic React – Exercise 1 – Facebook "phake"- Set up firebase

Basic React – Exercise 1 – Facebook "phake"- Set up firebase

```javascript
import firebase from 'firebase';

const firebaseConfig = {
    apiKey:
    authDoma
    projectI
    storageB
    messagin
    appId: "
};

const firebaseApp = firebase.initializeApp(firebaseConfig)
const db = firebaseApp.firestore()
const auth = firebase.auth()
const storage = firebase.storage()

export { db, auth, storage }
```

Basic React – Exercise 1 – Facebook "phake"-axios

```
import axios from 'axios'

const instance = axios.create({
    baseURL: "https://photo-social-backend.herokuapp.com"
})
export default instance
```

Basic React – Exercise 1 – Facebook "phake"-BE – MongoDB Post model

```javascript
import mongoose from 'mongoose'

const postsModel = mongoose.Schema({
    caption: String,
    user: String,
    image: String
})


export default mongoose.model('posts', postsModel)
```

Basic React – Exercise 1 – Facebook "phake"- BE - server

```
import express from 'express'
import mongoose from 'mongoose'
import Cors from 'cors'
import Pusher from 'pusher'
import Posts from './postModel.js'
import dotenv from 'dotenv';

//App Config
dotenv.config()
const app = express()
const port = process.env.PORT || 9000
const connection_url = process.env.DB_CONN

const pusher = new Pusher({
    appId: process.env.PUSHER_ID,
    key: process.env.PUSHER_KEY,
    secret: process.env.PUSHER_SECRET,
    cluster: "ap2",
    useTLS: true
});

//Middleware
app.use(express.json())
app.use(Cors())

//DB Config
mongoose.connect(connection_url, {
    useNewUrlParser: true,
    useCreateIndex: true,
    useUnifiedTopology: true
})
```

```
mongoose.connection.once('open', () => {
    console.log('DB Connected')
    const changeStream = mongoose.connection.collection('posts').watch()
    changeStream.on('change', change => {
        console.log(change)
        if(change.operationType === "insert") {
            console.log('Trigerring Pusher')
            pusher.trigger('posts','inserted', {
                change: change
            })
        } else {
            console.log('Error trigerring Pusher')
        }
    })
})

//API Endpoints
app.get("/", (req, res) => res.status(200).send("Hello TheWebDev"))

app.post('/upload', (req, res) => {
    const dbPost = req.body
    Posts.create(dbPost, (err, data) => {
        if(err)
            res.status(500).send(err)
        else
            res.status(201).send(data)
    })
})
```

```
app.get('/sync', (req, res) => {
    Posts.find((err, data) => {
        if(err) {
            res.status(500).send(err)
        } else {
            res.status(200).send(data)
        }
    })
})

//Listener
app.listen(port, () => console.log(`Listening on localhost: ${por
```

16

Thank you for cooperating
Gét gô

"*Coming together is a beginning;*
Keeping together is progress;
Working together is success."
- HENRY FORD

Database Management System