

Front-End Development



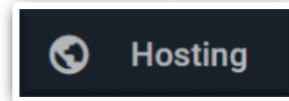
React & Firebase

Deploy React App to Firebase Host

#1 How to Deploy React Project to Firebase Hosting



Firebase



Firebase Hosting provides fast and secure static hosting for your web app.

Firebase Hosting is production-grade web content hosting for developers. With Hosting, you can quickly and easily deploy web apps and static content to a global content-delivery network (CDN) with a single command.

<https://linyoa-37d9d.firebaseio.com/>



Selamat Datang!

Tes deploy React App ke Firebase Hosting!

#1 Create/Prepare Your React App

- Install create-react-app utility globally on your PC (if you have not done yet before)

```
$ npm install -g create-react-app
```

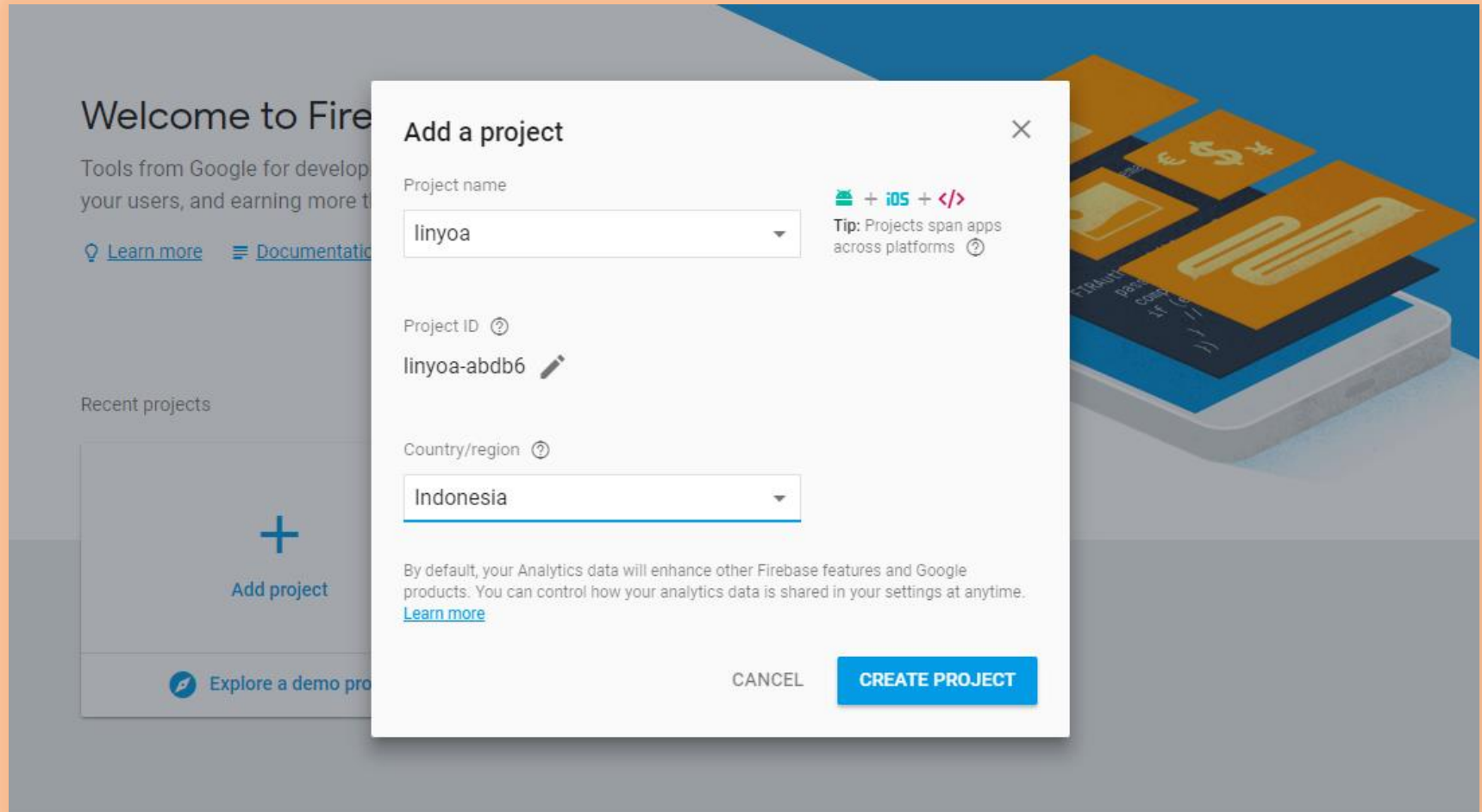
- Create a new react app or prepare yours

```
$ create-react-app linyoa  
$ cd linyoa
```

- Make sure that it runs successfully

```
$ npm start  
$ npm run build
```

#2 Create Firebase Project



#3 Setup Firebase CLI

■ Install Firebase CLI

```
$ npm install -g firebase-tools
```

■ Login to your Firebase account

```
$ firebase login
```

Woohoo!

Firebase CLI Login Successful

You are logged in to the Firebase Command-Line interface. You can immediately close this window and continue using the CLI.

#4 Firebase initialization

■ Firebase initialization

```
$ firebase init
```

- ★ ? Are you ready to proceed? (Y/n) **y**
- ★ ? Which Firebase CLI features do you want to setup for this folder?
Press Space to select features, then Enter to confirm your choices.
(Press <space> to select)
> (*) Database: Deploy Firebase Realtime Database Rules
 () Firestore: Deploy rules and create indexes for Firestore
 () Functions: Configure and deploy Cloud Functions
 (*) Hosting: Configure and deploy Firebase Hosting sites
 () Storage: Deploy Cloud Storage security rules
- ★ ? Select a default Firebase project for this directory:
 [don't setup a default project]
 > linyoa (linyoa-37d9d)
 [create a new project]

- ★ ? What file should be used for Database Rules?
(database.rules.json)
- ★ ? What do you want to use as your public directory?
(public)
build
- ★ ? Configure as a single-page app (rewrite all urls
to /index.html)? (y/N)
y
- ★ ? File build/index.html already exists. Overwrite?
(y/N)
n
- ★ Firebase initialization complete!

#5 Deploy!

■ Deploy to Firebase Hosting!

```
$ firebase deploy
```



```
=== Deploying to 'linyoa-37d9d'...
```

```
i  deploying database, hosting
i  database: checking rules syntax...
+  database: rules syntax for database linyoa-37d9d is valid
i  hosting: preparing build directory for upload...
+  hosting: 10 files uploaded successfully
i  database: releasing rules...
+  database: rules for database linyoa-37d9d released
    successfully
```



```
+  Deploy complete!
```

Project Console:

<https://console.firebase.google.com/project/linyoa-37d9d/overview>

Hosting URL: <https://linyoa-37d9d.firebaseio.com>

<https://linyoa-37d9d.firebaseio.com/>



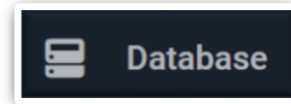
Selamat Datang!

Tes deploy React App ke Firebase Hosting!

#2 How to Connect React Project to Firebase Database



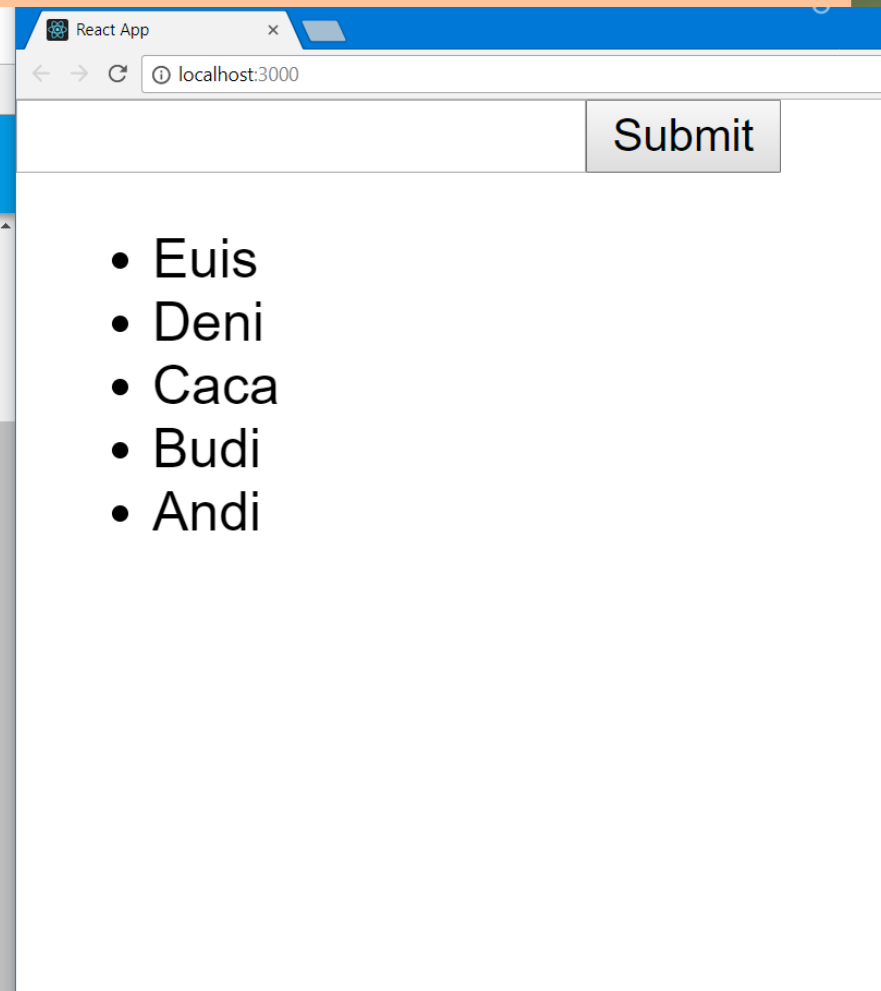
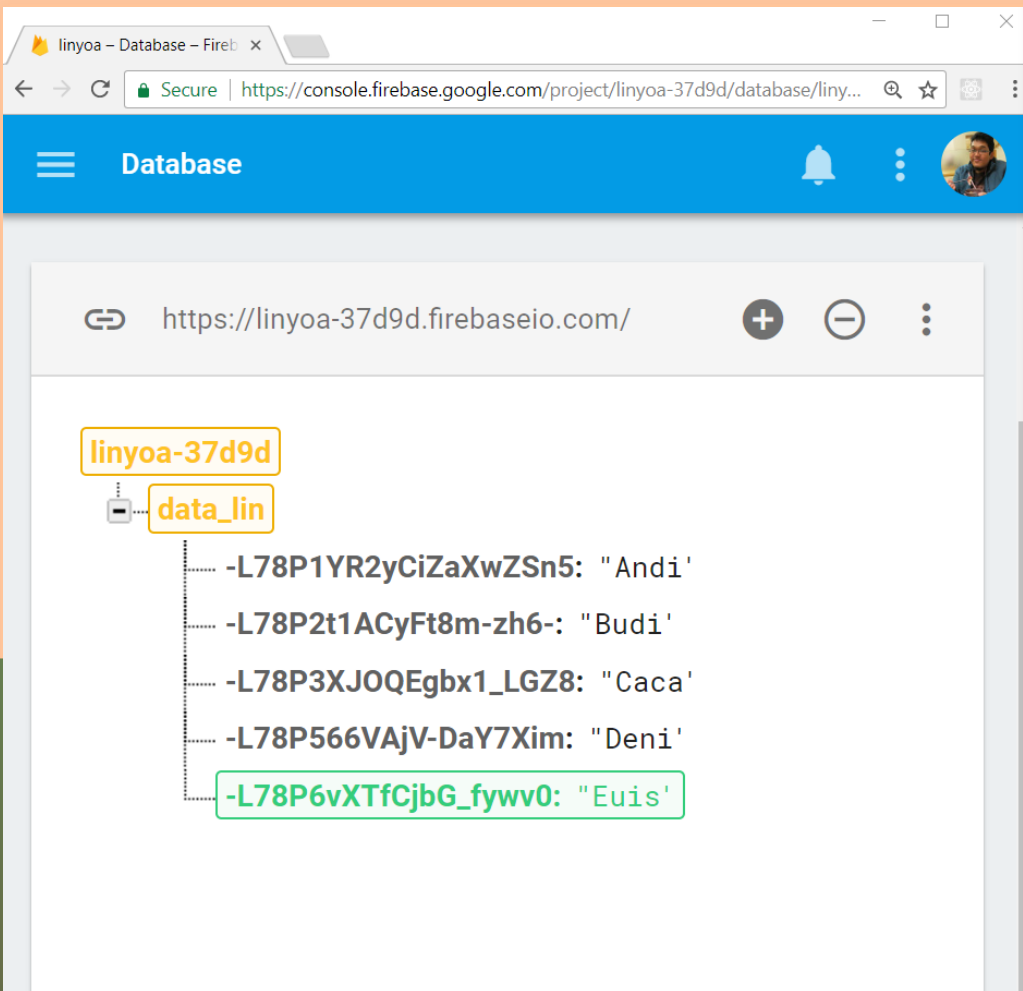
Firebase



Firestore is a cloud-hosted database. Data is stored as JSON and synchronized in realtime to every client.

When you build cross-platform apps with our iOS, Android, and JavaScript SDKs, all of your clients share one Realtime Database instance and automatically receive updates with the newest data.

React & Firebase Database



#1 Setup Firebase Database

- On your project dir, install Firebase Package!

```
$ npm install firebase --save
```

- On Firebase Database project, click **Project Overview** → **Add Firebase to Your Web App** then copy its snippet code.

Add Firebase to your web app

×

Copy and paste the snippet below at the bottom of your HTML, before other script tags.

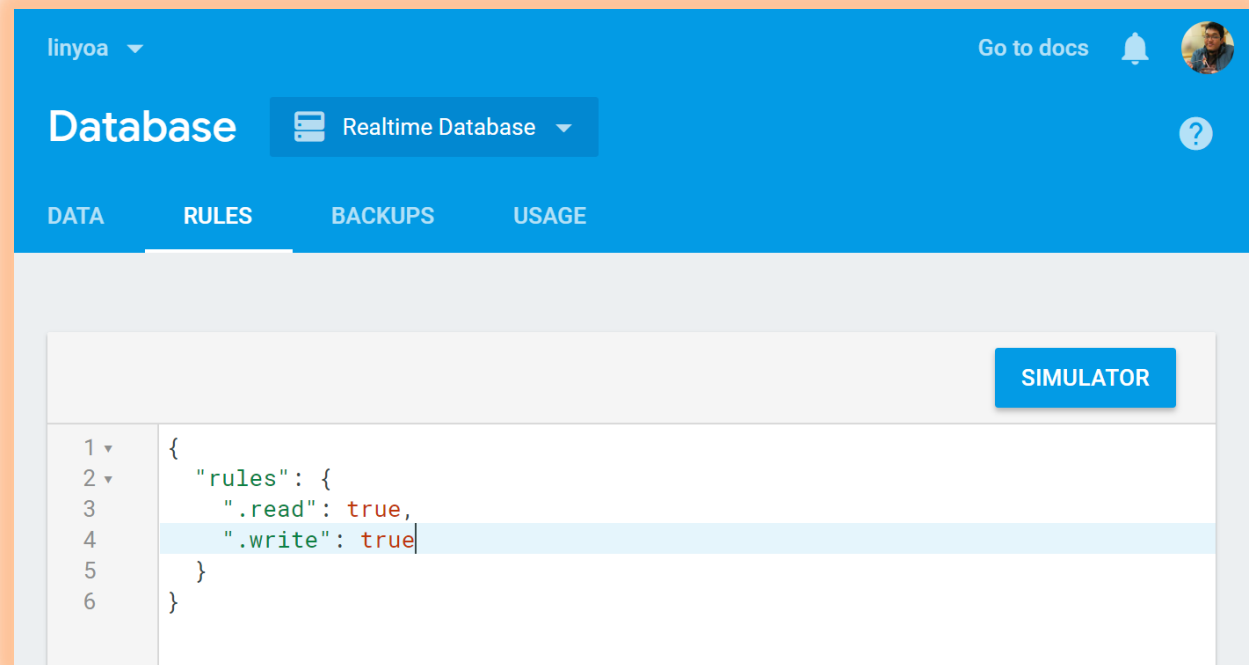
```
<script src="https://www.gstatic.com/firebasejs/4.11.0/firebase.js"></script>
<script>
  // Initialize Firebase
  var config = {
    apiKey: "AIzaSyAtQf_vMPce7JTfYQesg2isg8pZftJwIII",
    authDomain: "linyoa-37d9d.firebaseio.com",
    databaseURL: "https://linyoa-37d9d.firebaseio.com",
    projectId: "linyoa-37d9d",
    storageBucket: "linyoa-37d9d.appspot.com",
    messagingSenderId: "922933748183"
  };
  firebase.initializeApp(config);
</script>
```

COPY

#1 Setup Firebase Database

- On Firebase Database Rules (also on JSON file *database.rules.json* **if any**), set all its value to *true*:

```
{  
  "rules": {  
    ".read": true,  
    ".write": true  
  }  
}
```



#2 Create/Prepare Your React App

- Install create-react-app utility globally on your PC (if you have not done yet before)

```
$ npm install -g create-react-app
```

- Create a new react app or prepare yours

```
$ create-react-app linyoa  
$ cd linyoa
```

- Make sure that it runs successfully

```
$ npm start
```

#2 Create/Prepare Your React App

- Install Firebase packages needed:

```
$ npm install -g firebase-tools
```

```
$ npm install firebase
```

```
import firebase from 'firebase';

var config = {
  apiKey: "a1b2c3d4e5f6g7h8i9j0",
  authDomain: "linyoa-37d9d.firebaseio.com",
  databaseURL: "https://linyoa-37d9d.firebaseio.com",
  projectId: "linyoa-37d9d",
  storageBucket: "linyoa-37d9d.appspot.com",
  messagingSenderId: "1234567890"
};

var fire = firebase.initializeApp(config);

export default fire;
```

#4 Set Reference to Firebase Database

```
import React, { Component } from 'react';
import fire from './fire';

class App extends Component {
  constructor(props) {
    super(props);
    this.state = { messages: [] };
  }

  componentWillMount(){
    let messagesRef =
    fire.database().ref('data_lin').orderByKey().limitToLast(100);
    messagesRef.on('child_added', snapshot => {
      let message = { text: snapshot.val(), id: snapshot.key };
      this.setState({ messages: [message].concat(this.state.messages)
    });
  })
}
```

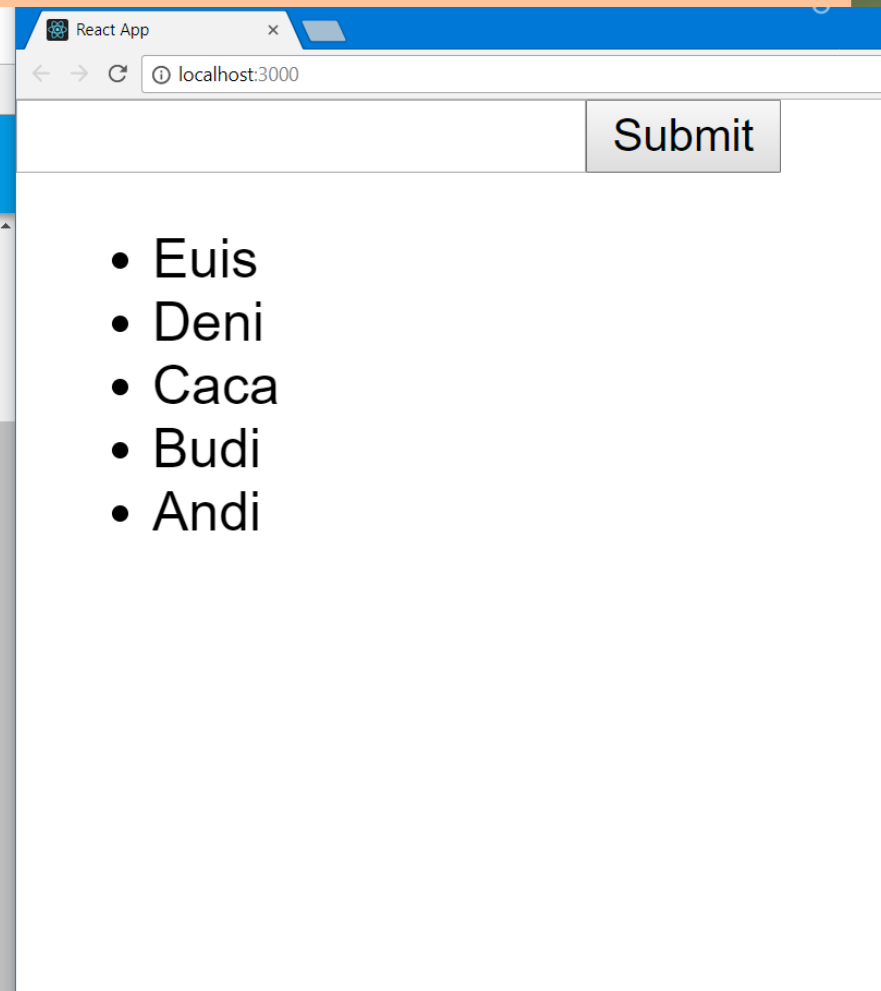
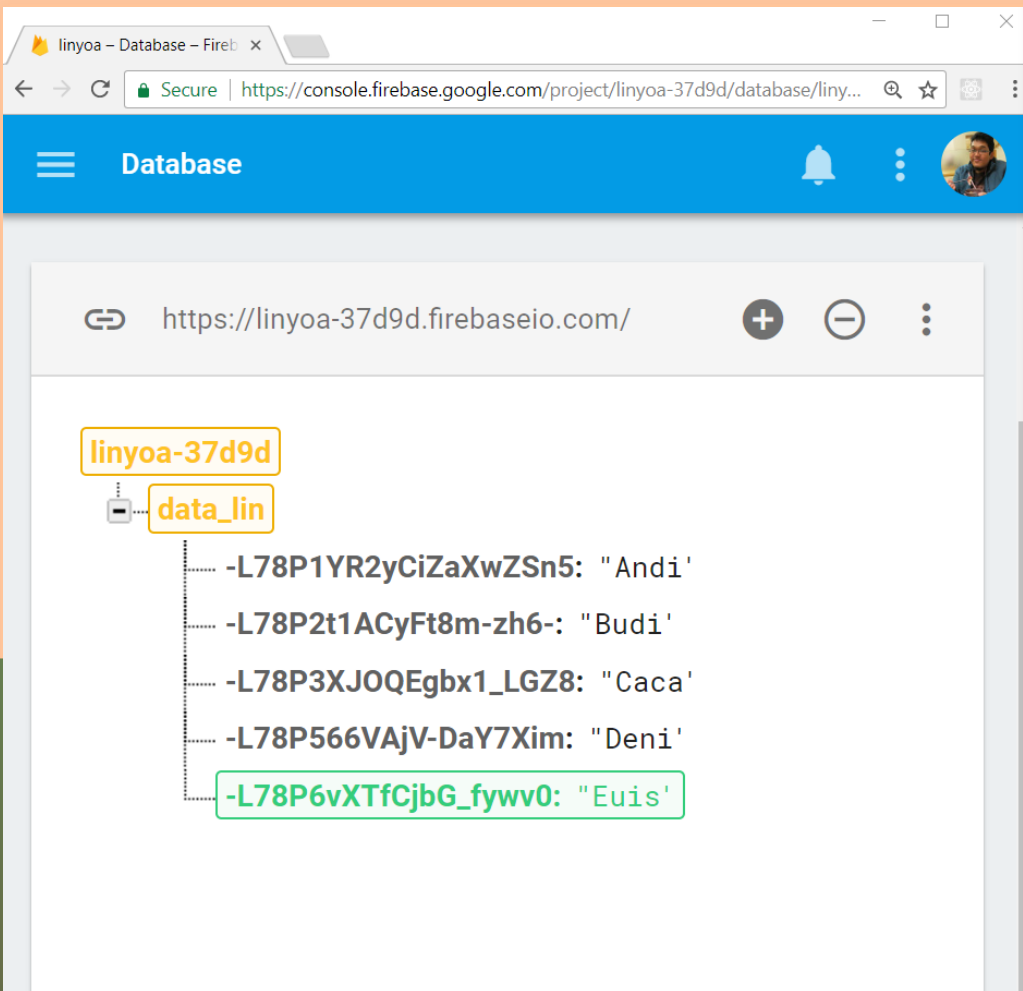
#4 Set Reference to Firebase Database

```
addMessage(e){
  e.preventDefault();
  fire.database().ref('data_lin').push( this.inputEl.value );
  this.inputEl.value = '';
}
```

```
render() {
  return (
    <form onSubmit={this.addMessage.bind(this)}>
      <input type="text" ref={ el => this.inputEl = el }/>
      <input type="submit"/>
      <ul>
        {
          this.state.messages.map( message => <li
            key={message.id}>{message.text}</li> )
        }
      </ul>
    </form>
  );
}
```

```
export default App;
```

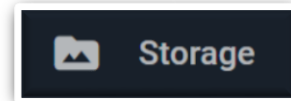
React & Firebase Database



#3 How to Connect React Project to Firebase Storage



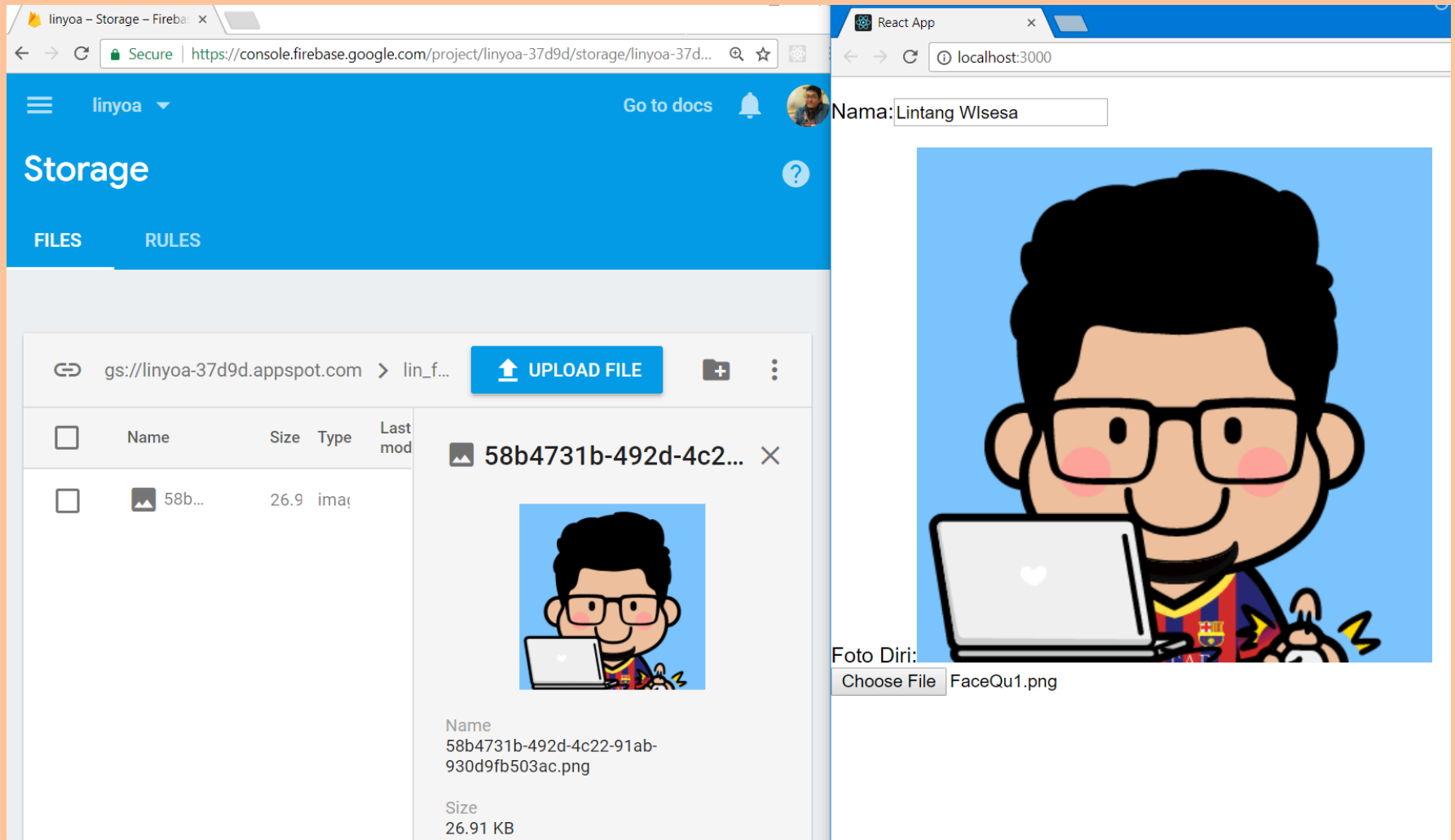
Firebase



Firestore is a powerful, simple, and cost-effective object storage service built for Google scale. The Firebase SDKs for Cloud Storage add Google security to file uploads and downloads for your Firebase apps, regardless of network quality.

You can use our SDKs to store images, audio, video, or other user-generated content. On the server, you can use **Google Cloud Storage**, to access the same files.

React & Firebase Storage



#1 Setup Firebase Storage

- Create a new Firebase project, or simply you can use your old project too.
- On Firebase project, click ***Project Overview*** → ***Add Firebase to Your Web App*** then copy its snippet code.

Add Firebase to your web app

×

Copy and paste the snippet below at the bottom of your HTML, before other script tags.

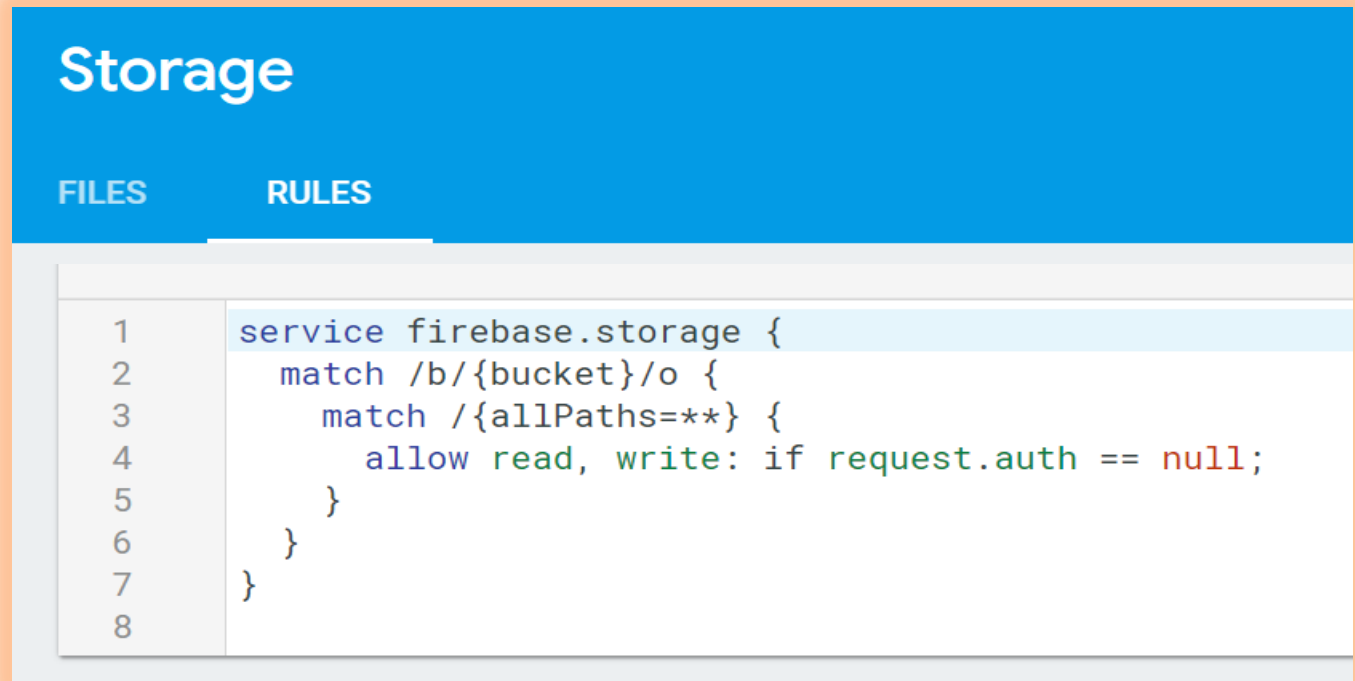
```
<script src="https://www.gstatic.com/firebasejs/4.11.0/firebase.js"></script>
<script>
  // Initialize Firebase
  var config = {
    apiKey: "AIzaSyAtQf_vMPce7JTfYQesg2isg8pZftJwIII",
    authDomain: "linyoa-37d9d.firebaseio.com",
    databaseURL: "https://linyoa-37d9d.firebaseio.com",
    projectId: "linyoa-37d9d",
    storageBucket: "linyoa-37d9d.appspot.com",
    messagingSenderId: "922933748183"
  };
  firebase.initializeApp(config);
</script>
```

COPY

#1 Setup Firebase Storage

- On Firebase Storage Rules, turn off auth request by set its code as following, then publish it:

```
service firebase.storage {  
  match /b/{bucket}/o {  
    match /{allPaths=**} {  
      allow read, write: if request.auth == null;  
    }  
  }  
}
```



#2 Create/Prepare Your React App

- Install create-react-app utility globally on your PC (if you have not done yet before)

```
$ npm install -g create-react-app
```

- Create a new react app or prepare yours

```
$ create-react-app linyoa  
$ cd linyoa
```

- Make sure that it runs successfully

```
$ npm start
```

#2 Create/Prepare Your React App

■ Install Firebase packages needed:

```
$ npm install -g firebase-tools
```

```
$ npm install firebase
```

```
$ npm install react-firebase-file-uploader
```

```
import firebase from 'firebase';

var config = {
  apiKey: "a1b2c3d4e5f6g7h8i9j0",
  authDomain: "linyoa-37d9d.firebaseio.com",
  databaseURL: "https://linyoa-37d9d.firebaseio.com",
  projectId: "linyoa-37d9d",
  storageBucket: "linyoa-37d9d.appspot.com",
  messagingSenderId: "1234567890"
};

var fire = firebase.initializeApp(config);

export default fire;
```

#4 Set Reference to Firebase Storage

```
import React, { Component } from 'react';
import fire from './fire';
import firebase from 'firebase';
import FileUploader from 'react-firebase-file-uploader';

class ProfilePage extends Component {
  state = {
    username: '',
    avatar: '',
    isUploading: false,
    progress: 0,
    avatarURL: ''
  };
}
```

#4 Set Reference to Firebase Storage

```
handleChangeUsername = (event) => this.setState({username: event.target.value});
```

```
handleUploadStart = () => this.setState({isUploading: true, progress: 0});
```

```
handleProgress = (progress) => this.setState({progress});
```

```
handleUploadError = (error) => {  
  this.setState({isUploading: false});  
  console.error(error);  
}
```

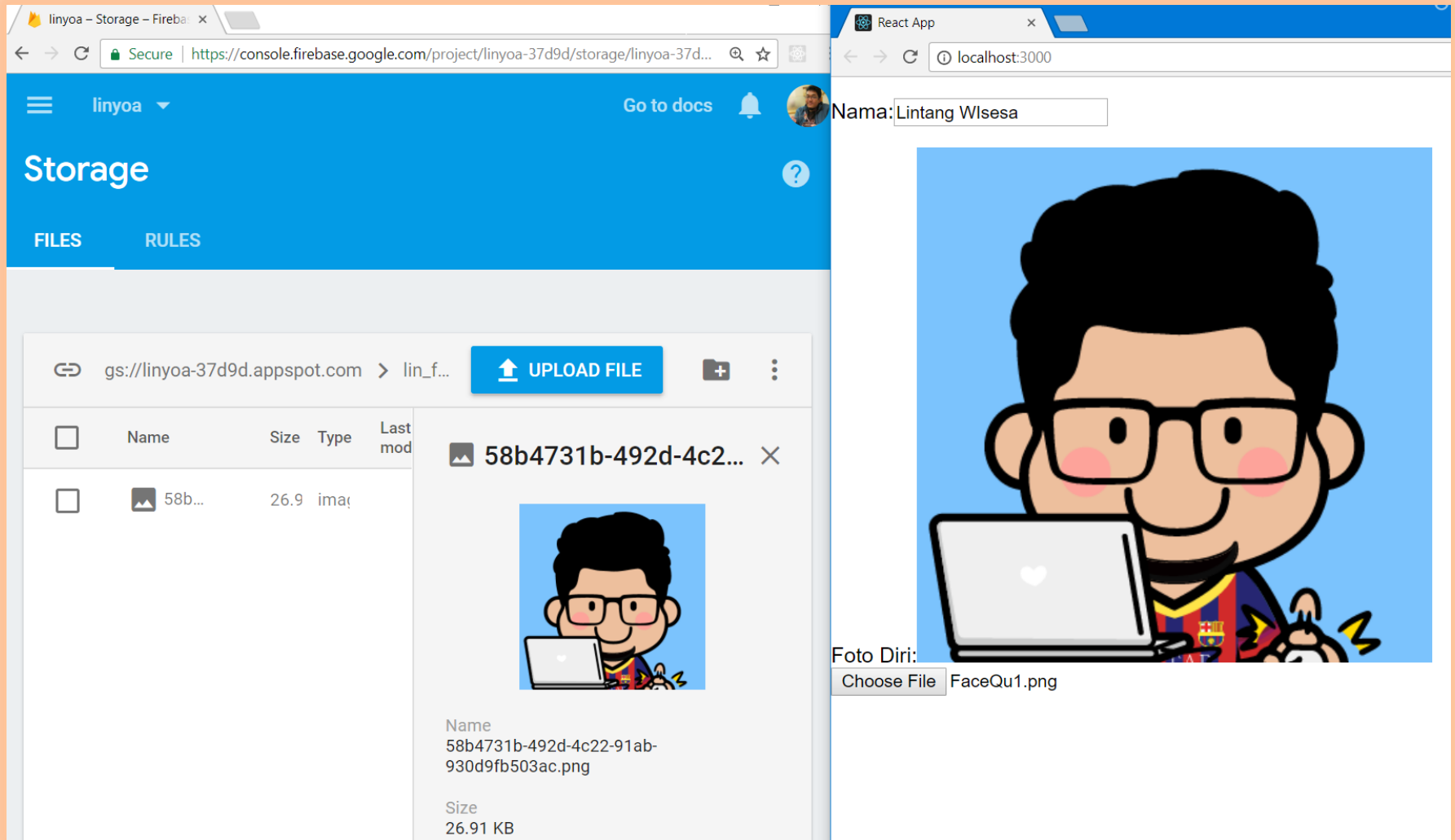
```
handleUploadSuccess = (filename) => {  
  this.setState({avatar: filename, progress: 100, isUploading: false});  
  fire.storage().ref('lin_foto').child(filename).getDownloadURL().  
  then(url => this.setState({avatarURL: url}));  
};
```


#4 Set Reference to Firebase Storage

```
render() {  
  return (  
    <div>  
      <form>  
        <p><label>Nama:</label>  
        <input type="text" value={this.state.username} name="username"  
        onChange={this.handleChangeUsername} />  
      </p><label>Foto Diri:</label>  
      {this.state.isUploading && <p> Process: {this.state.progress}</p>}  
      {this.state.avatarURL &&  
        <img src={this.state.avatarURL} alt="ok" />}  
        <FileUploader accept="image/*" name="avatar" randomizeFilename  
        storageRef={fire.storage().ref('lin_foto')}  
        onUploadStart={this.handleUploadStart}  
        onUploadError={this.handleUploadError}  
        onUploadSuccess={this.handleUploadSuccess}  
        onProgress={this.handleProgress}  
      />  
      </form>  
    </div>  
  );  
}
```

export default ProfilePage;

React & Firebase Storage



Front-End Development



React & Firebase

Deploy React App to Firebase Host