**A contacts web application (with the feature of OTP sending)**

**> Project overview:**

It is a web-based application that is used to generate OTP and send it to the user by phone number.

**> Methodology:**

This project can generate a random 6-digit OTP and send it to the user by phone number.

1. The user can send OTP to the contact list, one at a time by phone number.

2. The sorted list of contacts who had already sent the OTP will be displayed in descending order of the date and time.

3. The Server is handling the errors and exceptions. eg. if the OTP(6 digits) is not the required length then it will show the error message in Alert Format.

**> Technologies used :**

FrontEnd: **React.js,**

Backend: **Node.js,**

Database : **MongoDB ,**

online-services for sending OTP: **Vonage API**

Deployment: **Heroku,**

Version Control**: Git**

**> Libraries used :**

1)**Axios**: The Axios is the default library for making HTTP requests in React. It is used to make HTTP requests to the server.

2)**Express**: Express is a web application framework for Node.js. It is used to create the server.

3)**Mongoose**: Mongoose is an Object Data Modeling (ODM) library for MongoDB and Node.js. It is used to create the schema for the database.

4)**Nodemon**: The Nodemon is a tool that helps develop Node. js-based applications by automatically restarting the node application when file changes in the directory are detected.

5)**React-Bootstrap**: The React-Bootstrap is a library that provides a set of reusable components that are used to build the user interface.

6**)React-Router-Dom**: The React-Router-Dom is a library that provides a set of components that are used to create the routing in the application.

7)**Bootstrap**: Bootstrap is a library that provides a set of reusable components that are used to build the user interface.

**> Code Structure :**

**BackEnd :**

**1)To install the package:- npm install**

**2) Run the BackEnd:- npm start**

* All the required functions of the server are in the **controllers**. A controller is responsible for handling the request and response.
* The schemas are created in the **model**. A model is responsible for creating the schema for the database.
* Routes are responsible for handling the **request and response**. It is used to create the routes for the application.

**FrontEnd :**

**1)Go to the Client folder and install the package: npm install**

**2) Run the FrontEnd: npm start**

* All the routes are in App.js.
* The components are in the components folder.

**> To start simultaneously frontend & backend:- npm run dev on the server side.**

**> Github Link**: [Github](https://github.com/yashnerkar/otp_app)

**> Application link :** [Otp-web-application](https://otpwebapp.herokuapp.com/)