

## ***Online Contact Directory***

A20355499

Ramya Ajjavara Ramesh

[rar@hawk.iit.edu](mailto:rar@hawk.iit.edu)

## Description

This is web application used to store contacts online. In project 1 we had created notepad app which stores information in the indexeddb. This inspired me to come up with this app where users can add, view, edit or delete contacts. This is a very simple app which is developed in ReactJS. There are options to input information such as name, contact number, email address and contact's image. It also facilitates CRUD operation. The data is stored in Firebase Data store.

## Features and Functionality

**Functionality:** This is a **ReactJS web application** which I developed and styled using the Bootstrap CSS Framework. The application reads the input from html form elements and stores it on Firebase data store with unique id. Unique id is used when you want to delete or update a contact. I made use of Gulp for compiling and task management.

I designed the application using the Bootstrap framework which lets us build mobile friendly web application with its built in libraries.

## Accessing the Online Contact directory application

The application is accessible from

<http://www.krishnaenggwire.com/ramyaproject2/mycontacts1/dist/index.html>

OR

You could also try running the index.html page from the dist folder directly. If it doesn't work then follow the installation steps.

### Installation

- 1) Extract the zip file
- 2) Create a different folder
- 3) Browse to the location through command line and run npm init
- 4) Copy the package.json file which will be there inside the extracted zip file.
- 5) From the command line run npm install
- 6) Copy gulpfile.js and src folder to the new created folder.
- 7) Run gulp.
- 8) The web application should now be available inside the dist folder where there will be index.html page.

### Issues faced


React.js does not support few basic features of HTML such as class. Classname is used instead of class. The finding of this error took time.

It was not possible to delete a contact soon after you save it. The information used to disappear from the browser window, however when the page is refreshed the contact used appear again. This is because the AddForm.js used to not receive the file. Created a function to get contact the when the submit button is clicked.


## Working Screenshots

Ramya Ajjavara Ramesh


Add contact




Enter the name\*



Enter the contact number



Enter the E-mail address




Enter the URL for contact image


Save

Ramya Ajjavara Ramesh


Add contact




Enter the name\*



Enter the contact number




Enter the E-mail address



Enter the URL for contact image


Save



**Ramya**

Contact Number: 312-937-4083  
Email Address: ramshaw@lit.edu


EditRemove



**Rashmi**

Contact Number: 988-065-2765  
Email Address: rashmi@gmail.com

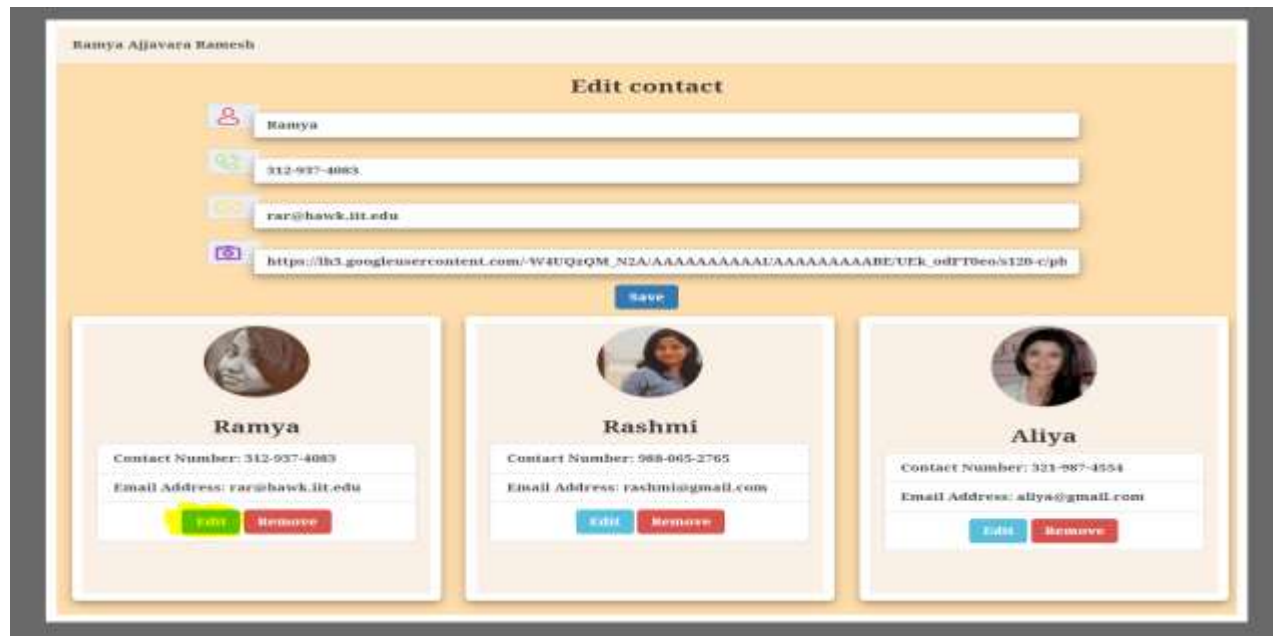
EditRemove



**Aliya**

Contact Number: 321-987-4554  
Email Address: aliya@gmail.com

EditRemove



## Technologies used

***ReactJS.0.14.7***

***React-dom.0.14.7***

***Firebase.2.4.2***

***Browserify.1.0.0***

***Flux.2.1.1***

***Gulp.3.9.1***

***Reactify.1.1.1***

***HTML5***

***CSS3***

***jQuery-1.12.3***

## **Improvements**

The major improvement which I would add would be to have a login module which will differentiate users. Currently any one can access the web page and add their own contacts. This process is not yet secure.

Other than the login page I would like to add more functionalities such as , search for the contact's profile in social media using just a button click.

Also I would like to make use of WebRTC features to directly communicate with the contact using browser with the help of peer connection.

## **What I learned**

ReactJS lets you build scalable web applications. I learned how to separate the functionality of a web application from the view. This allows us to create a proper web application. I also learned to connect the Firebase data store using API calls. It was a great experience building an app in ReactJS.