

Easy Volunteer

(App to receive or offer volunteering service without an institution behind)

Olivares Valencia, Rafael Isai

Student Id: 300300098

Section: 091

Content

| | |
|------------------------------------|----|
| Content | 0 |
| 1. Statement of the study | 2 |
| 2. Significance of the study | 2 |
| 3. Report | 3 |
| Proof of progress | 3 |
| Setup | 8 |
| Software design architecture..... | 10 |
| Scrip..... | 10 |
| References | 12 |

1. Statement of the study

Volunteering is a good way to get experience and get involved in your community. The experience obtained can make a difference in your resume in the future. Also, it is important to look for the right opportunity that fits your schedule and interests. According to the article “Volunteering in Canada”, people aged 15 and over give their time work and knowledge to volunteering. Also, it highlights that the internet is playing an important role nowadays because one of the ways people find opportunities to work in a volunteering program.

One of the hurdles, at least in Vancouver City is that some voluntaries are made through an association or institution, and the process can be too long. This app, more than a regular volunteering service app, is an easier way to help those who need help right of way. Same as Uber, through the website or the android app, you might pick up a volunteer in your zone, or post your request, if it is not urgent.

With Easy volunteer, you can ask for a volunteering service, such as grocery shopping, daily errand, or general home help, or you can offer your service and time to help those people that need a hand. Easy volunteer does not focus on a specific type of population, it can help anybody. But in particular, those people that may need some help for a specific disability. Don't get confused, this is not an app to find free labor, it is to help faster to those who need it.

About the functionality and features, the main goal of Easy volunteer is to make the process to help someone else much easier. For those who want to offer a service or simple company, they can post their profile in the app to be found by someone that needs that specific skill or tool. On the other hand, those who want some help can post their needs, or look for someone around the area. The post will work as a simple help post that will show to everybody in their network. Also, the person that needs help can chat straight to the person who is giving the service, this person can use our chat, which will help to make easy the way the person who needs help can communicate with the volunteer. As soon as the volunteer is a token from someone that needs assistance, its status will turn as a token. Therefore, this person will no longer be able to assist anyone else, until finishing their task.

At the end of the service, the person that receives the service can add a review about the volunteering, this might help the volunteer for some experience in the future, in work, or for future needs. In the future, we could add more features, such as a donation for volunteers, or integration with a proper institution of volunteering (to make their work easier), even look in real-time how the volunteering is going (in case the benefited involves running some errands)

2. Significance of the study

Easy volunteer is going beyond just a networking app. It helps those who need a service or someone with a specific skill, and also it guides those who are looking for experienced and improves their skills. We could call this app a win-win app because 2 parties are getting a benefit at some point. When a benefited request volunteering to a volunteer, has to make sure the job offered will fit in the skills from the volunteer, or at least in its interest. Otherwise, the volunteer might refuse the volunteering.

Also, the article mentioned previously shows that in 2010 58% of the volunteers were people between 15 to 24 years old, and 54% are between 35 and 44 years old. Therefore, the volunteers are prone to

research volunteering throw the internet or an app, which might make it easier the way to get involved in a volunteering experience.

3. Report

To understand the project better, it is necessary to know the 3 main collections: Profile, Chats, and Post. The profile will save the user information, user connections, profile reviews, and the volunteering the user has achieved. Besides, chats will collect all messages sent from between users. And last, the post will save the info from posts that users have created.

Proof of progress

The progress of this project is based on the collections before mentioned. At this point, Profile has the entire CRUD (Create, read, update, delete) working. This means, a user is able to create a profile, login into the system, logout, look at the profile and update it.

Create profile

The user is allowed to create 2 types of profiles: Benefited and Volunteer. Also, it has to type its password twice to make sure the password is correct.

A screenshot of a mobile application's 'Register' screen. The form has a light blue background. It contains several input fields: 'Name' with the text 'Test Rafael', 'Last name' with 'Olivares', 'Type' with a dropdown menu showing 'BENEFITED', 'Email' with 'r@m.mx', 'Password' with masked characters, and 'Confirm Password' with masked characters. A 'Password match?' label is below the confirm password field. At the bottom is a blue 'REGISTER' button.

Figure 2 Mobil version

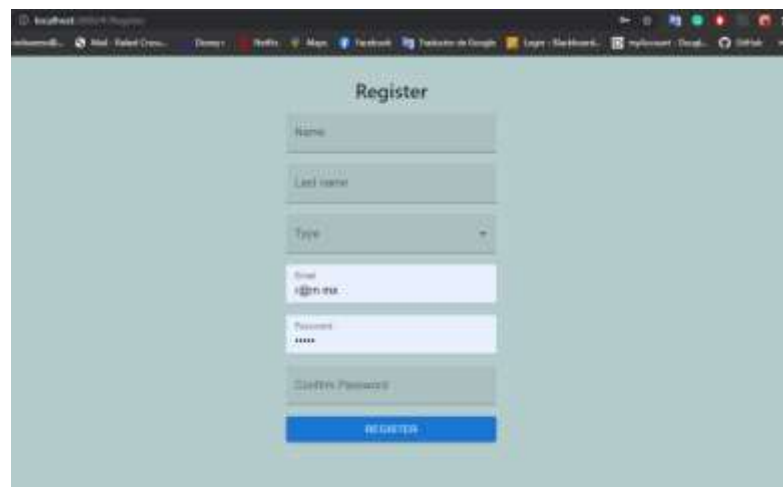
A screenshot of a web browser showing the 'Register' form. The form is centered on a light blue background. It includes fields for 'Name', 'Last name', 'Type' (a dropdown menu), 'Email', 'Password', and 'Confirm Password'. A blue 'REGISTER' button is at the bottom.

Figure 1 Web version

```
5: Object
  name: "Test Rafael"
  lastName: "Olivares"
  profileType: "BENEFITED"
  createdAt: 2021-10-20T03:30:44.927+00:00
  email: "r@m.mx"
  password: "$2a$08$armlceFafgNe.sZ1Z0ngv.ZyNTep09kPHJLQLOsrDt69z9y4RTxf2"
  connection: Object
    _id: ObjectId("61722d445714b3dc10ba40a5")
  reviews: Array
  volunteering: Array
```

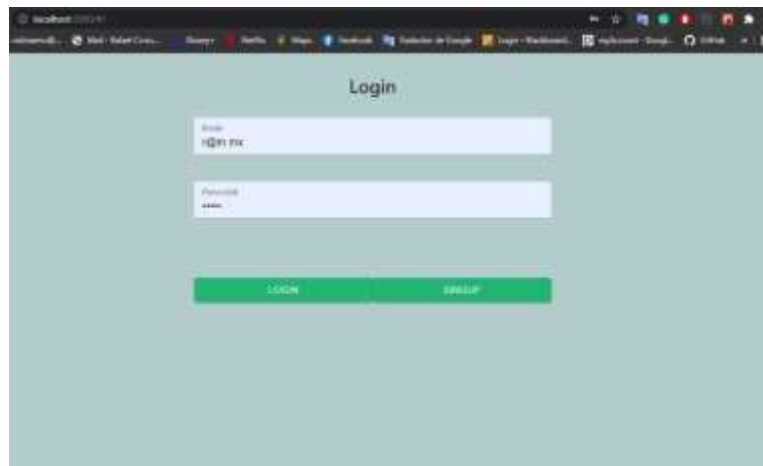
Login

Use the login page to access the app. If the user misses information or is not registered yet, it will show a message.



The mobile version of the login page features a light blue background. At the top, the word "Login" is centered. Below it, there are two input fields: "Email" with the text "ng@mm.mil" and "Password" with masked characters "*****". At the bottom, there are two green buttons labeled "LOGIN" and "SIGNUP".

Figure 4 Mobil version

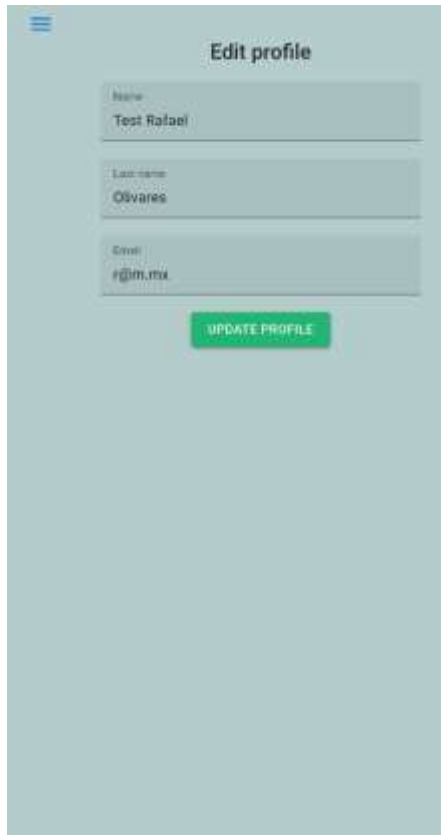


The web version of the login page is displayed within a browser window. It has a light blue background and the title "Login" at the top. The input fields for "Email" (containing "ng@mm.mil") and "Password" (containing "*****") are stacked vertically. Below these fields are two green buttons labeled "LOGIN" and "SIGNUP".

Figure 3 Web version

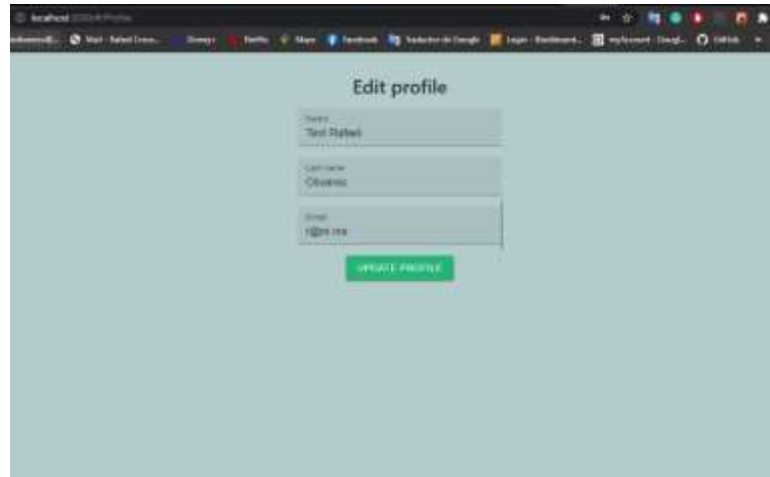
Edit Profile

All users are allowed to look and edit their name, last name, and email. After that, they have to log in again to reload their information.



The mobile version of the 'Edit profile' form is displayed on a light blue background. It features a hamburger menu icon in the top left corner. The form consists of three input fields: 'Name' with the text 'Test Rafael', 'Last name' with the text 'Oliveira', and 'Email' with the text 'r@dm.itx'. Below these fields is a green button labeled 'UPDATE PROFILE'.

Figure 6 Mobil version

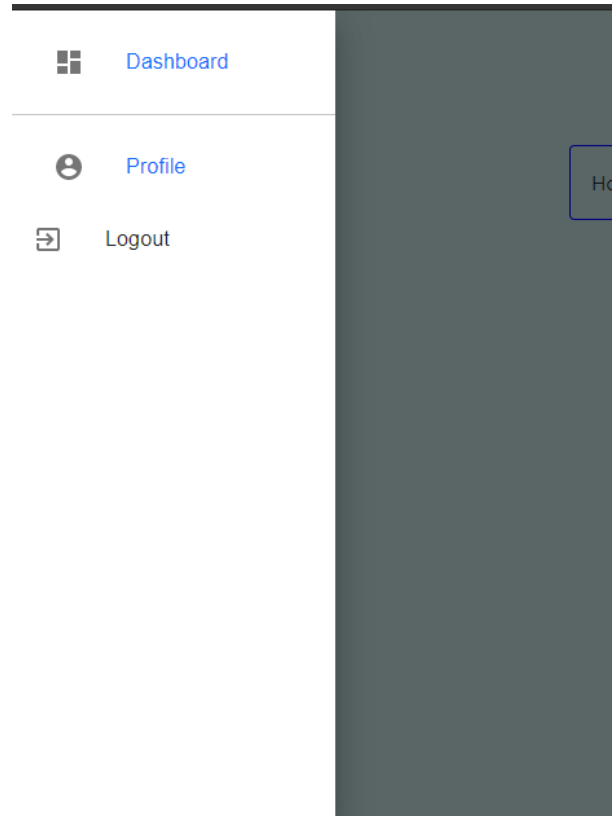


The web version of the 'Edit profile' form is shown within a browser window. The browser's address bar shows 'localhost:3000/#profile'. The form is centered and contains three input fields: 'Name' with 'Test Rafael', 'Last name' with 'Oliveira', and 'Email' with 'r@dm.itx'. A green 'UPDATE PROFILE' button is positioned below the email field. The browser's taskbar at the bottom shows various open applications like Chrome, VS Code, and others.

Figure 5 Web version

Menu

For now, we just have 3 items in our menu: Dashboard, profile, and Logout. Sensitive information is saved in the temp memory in the app, when logout this info is deleted and redirected the page to the login.



Dashboard
It shows the post activated.

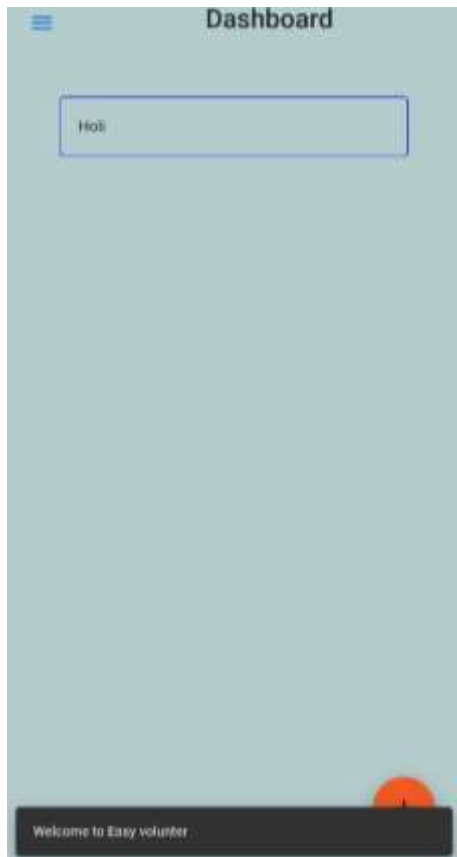


Figure 8 Mobil version

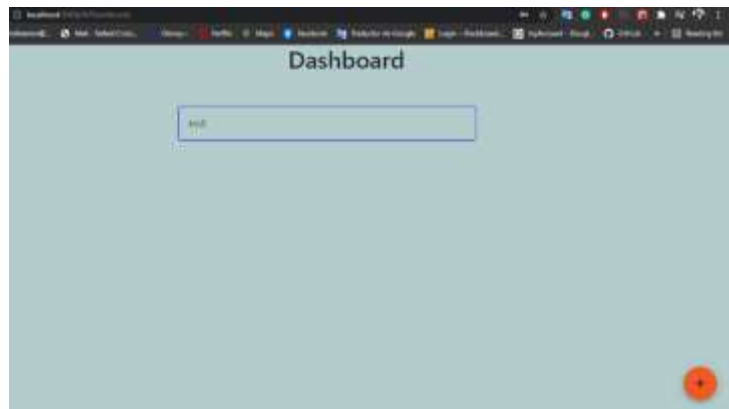


Figure 7 Web version

New post
Create a new post to help or be helped.



Figure 10 Mobil version



Figure 9 Web version

```
2: Object
  > createdBy: Array
    message: "This is a Midterm Test"
    createdAt: 2021-10-22T01:18:08.593+00:00
    status: "Active"
    _id: ObjectId("617211505714b3dc10ba4094")
```

The current work is under Post, a user might post a voluntary under the app and save it on the database. This post record the userId, date, and message.

The dashboard is showing the general box where the post will be set.

Setup

This is a react project, powered by Cordova, which is a framework that allow us to become our web app into android app.

This project is divided into 2 sections:

1. Presentation
2. Webservice

This application was created using the command
create-react-app

To run the project, download or clone the project to your machine.

git clone https://github.com/RIOlivaresV/easy_volunteer

After you download the project, you need to install the dependencies for both projects separately.

Presentation

Go to the folder Presentation and run the command:

npm install to install all dependencies for react.

Web version

To run the app in your browser you need to be under app folder and run the command:

npm start to run the react app locally

This will open the app in a browser from <http://localhost:3000/>

Android version

To run the app on android you need to be under app folder and do follow:

1.- Download Java 1.8 (<https://www.oracle.com/java/technologies/javase/javase-jdk8-downloads.html>)

2.- Download gradle (<https://gradle.org/releases/>)

3.- Set Environment variables (JAVA_HOME, JRW_HOME, ANDROID_HOME, ANDROID_SDK_ROOT, gradle)

4.- Add previews variables to the PATH (restart your pc if needs)

5.- Buil the project running:

npm run build

6.- Delete and Add android platform from project running this:

cordova platform remove android

cordova platform add android

7.- Build project running this:

cordova build android

8.- Run project with this following:

cordova run android

The app will run either an android device connected, or to the emulator.

WebService

Go to the folder WebService and run the command:

npm install to install all dependencies for node js.

*

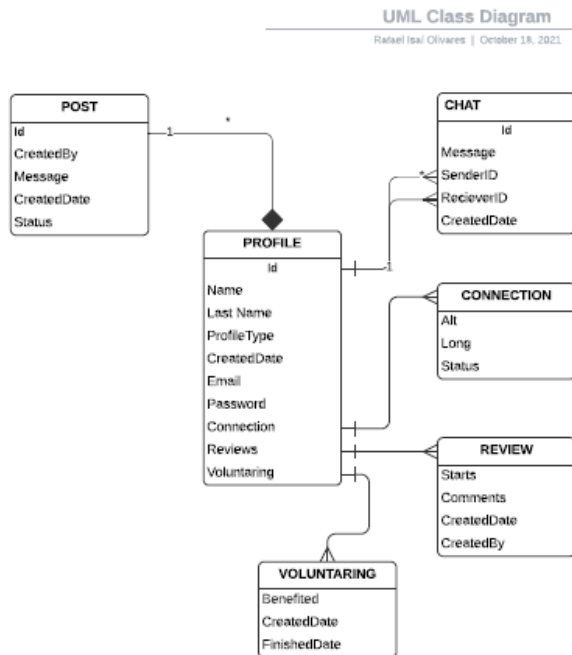
To run the app you need to be under app folder and run the command:

nodemon index.js to tun the node js service locally

This command will run the service. If nothing is changed, you should be able to access the client from <http://localhost:8000/>

Software design architecture

ULM



Builder pattern

The design pattern used in this project is Builder pattern, which is known for separating the construction of a complex object, creating sub tasks to make the process easy and smooth. Also, it is friendly to changes, and offers more control in the progress and development of the project.

In computer science, we well know the quote "Divide and conquer", which means that a task could be split into several pieces, in order to get such a simple task enough to be solved.

As it was mentioned before, easy volunteering is working on 3 main objects, it started with Profile, which 90% of the CRUD is covered and working in the app.

The next step will be Post, the project already records a post in the database, it is working on showing posts in a list under "dashboard".

The follow is Chats, which will involve a communication between users. The last step will be the total design of the app. The main goal for now is to have a functional product, and then it could work on design.

Script

As we already know, the project uses MongoDB to save our data. This project on MongoDB contains 3 main collections. Profiles, Posts, and Chats.

Profile

{

```

profile: [
  {
    name: String,
    lastName: String,
    profileType: String,
    createdAt: Date,
    email: String,
    password: String,
    connection: {
      lat: String,
      long: String,
      status: String,
    },
    reviews: [
      {
        starts: Number,
        comments: String,
        createdBy: String,
        createdAt: Date,
      },
    ],
    volunteering: [
      {
        benefited: String,
        createdAt: Date,
        finishedDate: Date,
      },
    ],
  },
]
]
}

```

Chats

```

{
  chats: [
    {
      message: String,
      senderId: String,
      recieverId: String,
      createdAt: Date,
    },
  ],
}

```

Post

```
{
  post: [
    {
      createdBy: String,
      message: String,
      createdAt: Date,
      status: String,
    },
  ],
}
```

References

- Vezina, M., & Crompton, S. (2012, April 16). *Volunteering in Canada*. Statistics Canada. <https://www150.statcan.gc.ca/n1/pub/11-008-x/2012001/article/11638-eng.pdf>
- Vancouver, C. O. (2021). *Volunteering*. City of Vancouver. <https://vancouver.ca/people-programs/volunteering.aspx>
- Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides (1994). *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison Wesley. pp. 97ff. ISBN 0-201-63361-2.