

The Papi Chulos

Christian Kurt Balais: 923366103

Jacob Lazzarini: 923542201

James Dixon: 922440659

Anna Marie Felipe: 915081822

Tony Mendoza Gonzalez: 921676038

Github used for the project: kurteekurt

CSC317 Web Development

Assignment 5 – Final Group Project

Description:

The goal of this assignment was to implement the backend of our website, which included adding a database to store user information using MySQL, communicate to the express endpoints from our client, and to implement Bcrypt, which is a password hashing algorithm designed for safe password storing. Another task of ours was to polish up the website a bit and make it as clean and good looking as possible.

Approach / What I Did:

The first thing that we did was design our database tables, we had a basic structure based on the javascript objects we were using previously. Because we are running this locally we created a markdown guide to make sure we all are working with a database formatted exactly the same. After the database was designed and implemented we worked on making endpoints to connect our client and database through the server. The endpoints would be responsible for storing user inputted data into the database, such as usernames, emails, birthdays, and passwords for users who registered, and usernames, emails, passwords, and locations for organizations wanting to sign up. Additionally we created endpoints to fetch that data (client fetch from server) for use in dynamic page generation for our organization and organizations pages. Next we revamped our FAQ page to allow users to inquire for more information. A form was created to allow the user to input their name, email, phone number, and a message box to allow the user to send an inquiry. When the user presses send, an email is sent to the handy helpers email and we are able to view the message. The home page was also revamped, adding testimonies and a rotating slideshow that showcases photos from different volunteer opportunities.

Issues and Resolutions:

Organization Job post and Dropdown

My first issue was trying to complete the dynamic header with the dropdown menu to complete it but the same issues still came across where the header would appear in a list form when added to the sign in/login page therefore we decided to remove the original code and stuck with the presented version. This is potentially due to the fact that the code used the tag making it a list that was supposed to be erased by the css styling but was overridden by the other dropdown menu styling.

There was a job posting site and we had the form attached to the dropdown menu but there was no posting page to send it to as the project was finished. Due to trying to tackle both challenges, the dropdown and job posting/content.html, at the same time there were various

errors that appeared due to the confusion in pages and mislabeled elements. I watched over 30 videos on youtube where every developer was using php or languages of the fact for the database portion that i mostly spent my time watching sql tutorials only to be more confused than how i started therefore I turned to StackOverFlow and Geeks for Geeks for more simplified versions and worked my way from there but again it was a lot of people who worked in the field with solutions that we couldn't use. On top of all that, due to not thinking ahead, time ran out for adding those features and more the solution was to stick with what we had and continue, even if that portion didn't work entirely, all the html, css, javascript, and sql is there although not able to publish.

FAQ Page:

I did some research on StackOverflow on how to implement an email sending system using the Contact Form in the FAQ page. The issue I ran into was a lot of the people posting solutions and instructions on how to do it were explaining on a high developing level where they were eventually going to push their code to production. This meant that they would need to implement a server (that I didn't feel the need I had to do for a project since no real users would use it) to process accounts from multiple services like Gmail, Hotmail, Outlook, YaHoo etc. So I actually found a solution using a website called ElasticEmail that lets you use a Gmail or whatever accounts service you choose to use and create a server using a flexible SMTP that was perfect for a demo.

Analysis: (Reflect on what you have learned and how decisions made could impact users.)

With this go around of working on the website, I think the big thing we learned was making sure everything stayed uniform and consistent, even while revamping a couple pages of the site. Despite the revampment, the site remained consistent. It's good to keep everything looking consistent because without the design consistency the website will look messy and disorganized. Along with organization, the use of sql to connect a form from a an html inputted into javascript to then be called into an event and sent into a database into another html template set up with css styling then published in the final html site showed us how much work goes into a site every time we used social media and how many processes it has to go through to assure that our assets whether posting on x, facebook, instagram, etc. and that to us it's just a couple of buttons away, something that was certainly overlooked as a push of a couple buttons that people sometimes get mad over it not working. keeping a database also allowed us to refactor a lot of the code to make it cleaner overall.

Screen shots: (next page)

// dropdown menu code added to header code to implement it as a dynamic header

```
@@ -1,14 +1,51 @@
 1 + <link href="css/dropmenu.css" rel="stylesheet" />
 2   <link href="css/header.css" rel="stylesheet">
 3   <header class="header" id="header">
 4     <a href="index.html">
 5       
 7     </a>
 8     <div class="header-right">
 9       <a class="home active" href="index.html">Home</a>
10       <a class="organizations" href="organizations.html">Organizations</a>
11       <a class="aboutus" href="aboutus.html" target>About Us</a>
12       <a class="contactus" href="faq.html" target>FAQ</a>
13       <a class="login" href="login.html" target>Sign In/Sign Up</a>
14     </div>
15   </header>
16 +
17 + <div class="action">
18 +
19 +   <div class="profile" onclick="menuToggle();">
20 +     
21 +   </div>
22 +   <div class="menu">
23 +     <h3>user name<br /></h3>
24 +     <ul>
25 +       <li>
26 +         <a href="userprofile.html">My profile</a>
27 +       </li>
28 +       <li>
29 +         <a href="orgjobpost.html">Upload post</a>
30 +       </li>
31 +       <li>
32 +         <a href="#">Inbox</a>
33 +       </li>
34 +       <li>
35 +         <a href="#">Setting</a>
36 +       </li>
37 +       <li><a href="faq.html">Help</a></li>
38 +     <li>
39 +        <a class="login" href="login.html" target>Sign In/Sign Up</a>
40 +     </li>
41 +   </ul>
42 + </div>
43 + </div>
44 +
45 + <script>
46 +   function menuToggle() {
47 +     const toggleMenu = document.querySelector(".menu");
48 +     toggleMenu.classList.toggle("active");
49 +   }
50 + </script>
51 + </body> Ø
```

//result of said code



```

<link href="css/dropmenu.css" rel="stylesheet" />
<link href="css/header.css" rel="stylesheet">
<header class="header" id="header">
    <a href="index.html">
        
        <a class="home active" href="index.html">
        <a class="organizations" href="organizati
        <a class="aboutus" href="aboutus.html" ta
        <a class="contactus" href="faq.html" targ
        <a class="login" href="login.html" target
    </div>
</header>

<div class="action">

    <div class="profile" onclick="menuToggle();">
        
    </div>
    <div class="menu">
        <h3>User Name<br /></h3>
        <ul>
            <li>
                <a h
            </li>
            <li>
                <a h
            </li>
            <li>
                
            </li>
            <li>
                
            </li>
            <li>
            <li>
                
            </li>
        </ul>
    </div>
</div>

<script>
    function menuToggle() {
        const toggleMenu = document.querySelector

```

```

1 <link href="css/header.css" rel="stylesheet">
2 <header class="header" id="header">
3     <a href="index.html">
4         
8         <a class="home active" href="index.html">
9         <a class="organizations" href="organizati
10        <a class="aboutus" href="aboutus.html" ta
11        <a class="contactus" href="faq.html" targ
12        <a class="login" href="login.html" target
13     </div>
14 </header>

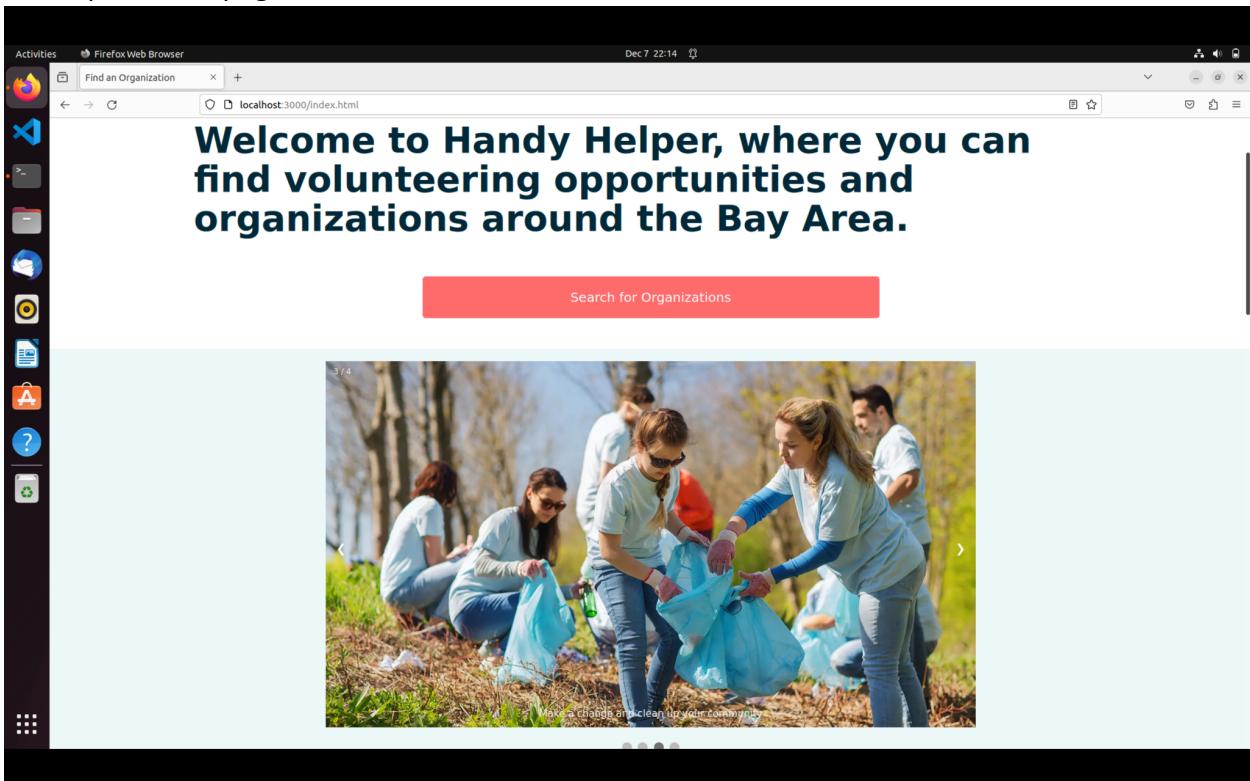
```

//deletion of code along with reference styling

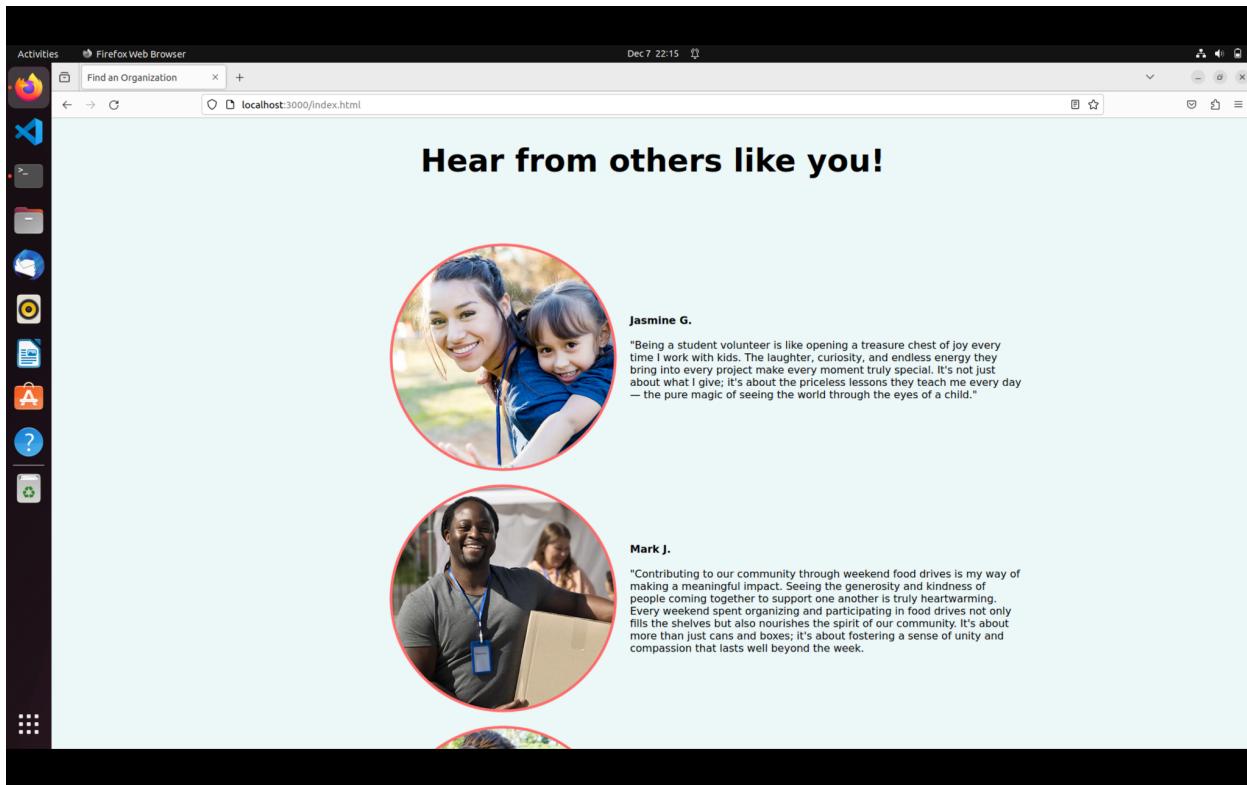
```
// deleted unused pages
```

```
<> dropmenu.html public D  
<> settings.html public D  
<> aboutus.html public/... D  
# content.css public/ess D
```

Revamped Homepage:



Testimonies on Homepage:



Revamped FAQ page:

How does your website work?

- Our website finds the closest volunteer opportunities near you posted by local organizers and groups. All you have to do is find an event you like and ask to volunteer!

How old do you have to be to volunteer?

- You only have to be 13 years old to start volunteering!

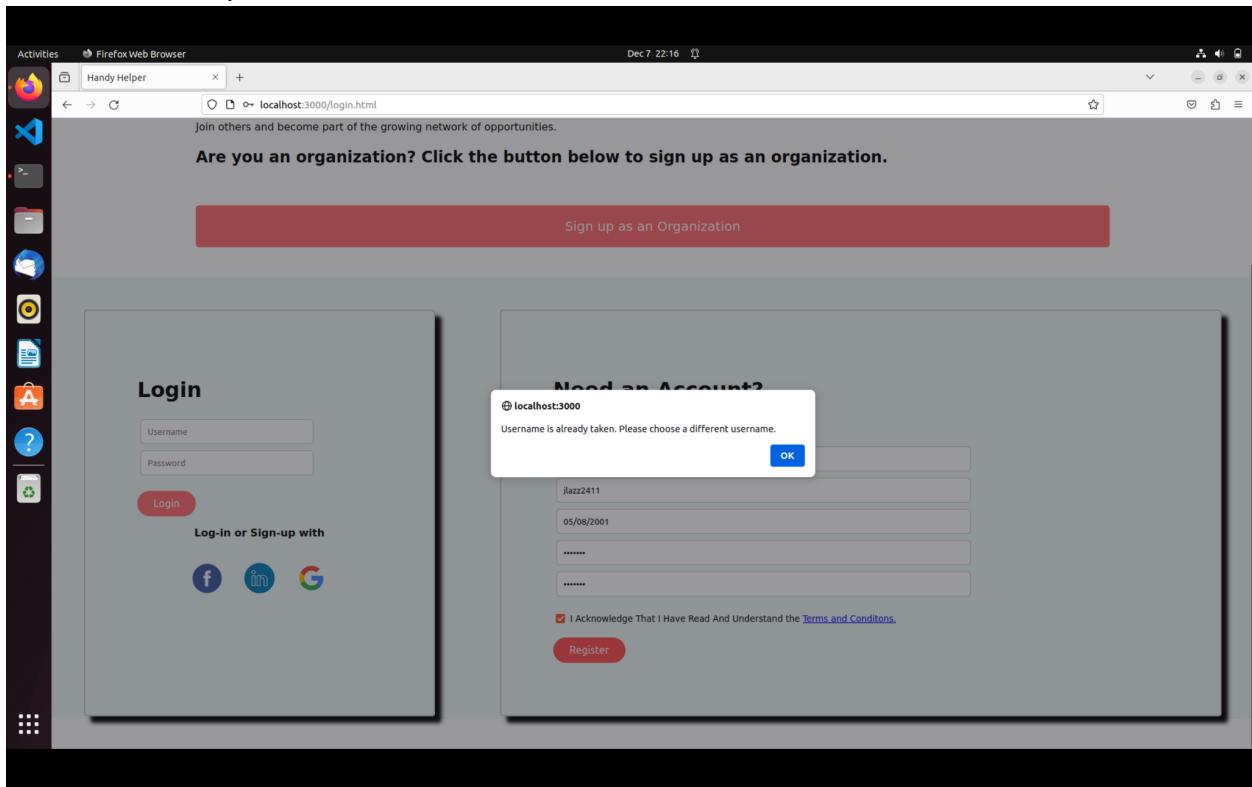
Is your website free to use?

- Yes, of course, it is!

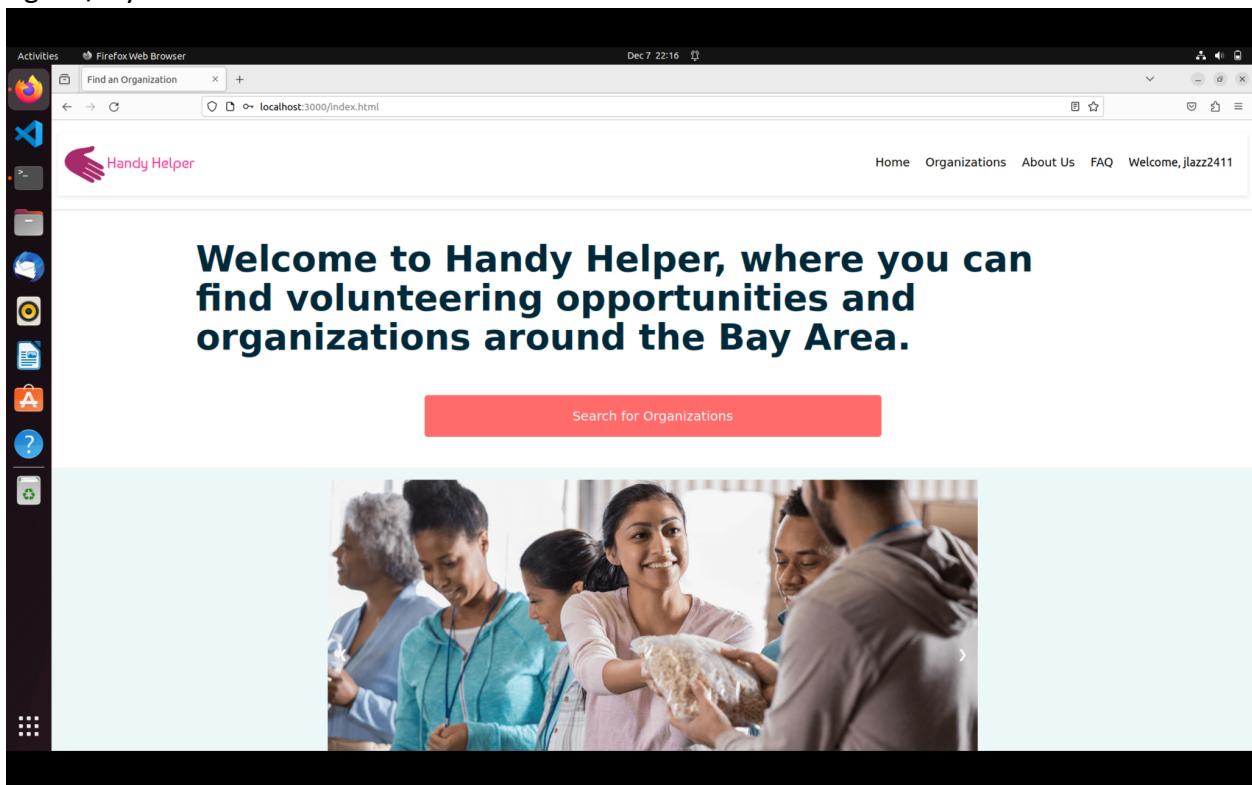
Need to know more? Contact Us

Can't wait to hear from you!

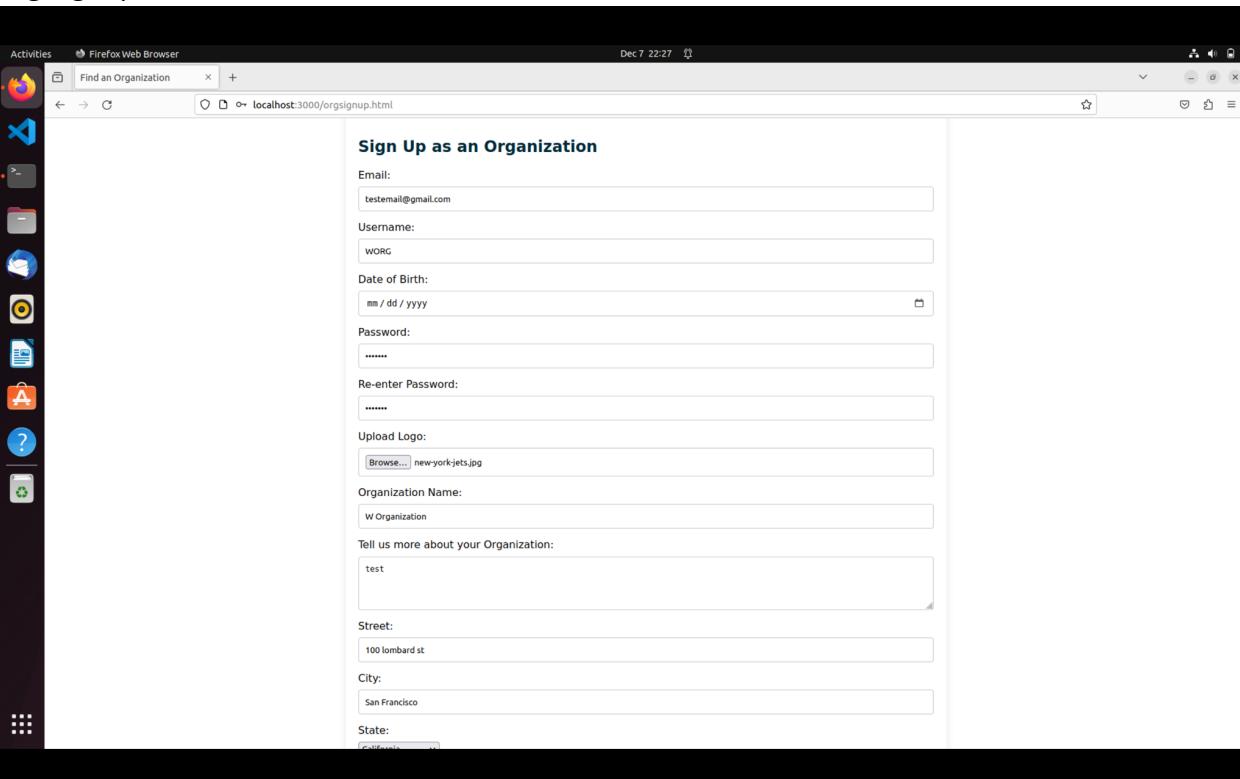
Username already taken alert:



Sign in/ dynamic header:

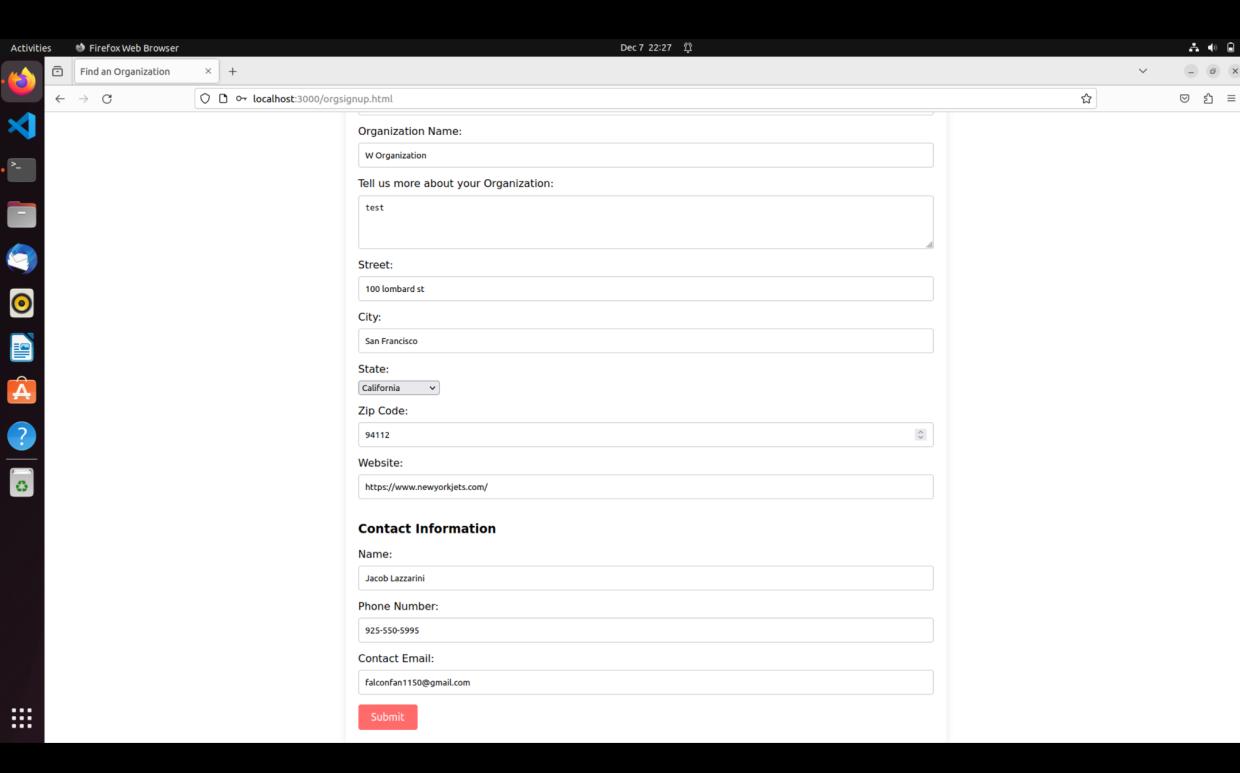


Org Signup:



The screenshot shows the initial step of an organization signup process. The title "Sign Up as an Organization" is at the top. The form fields include:

- Email: testemail@gmail.com
- Username: wORG
- Date of Birth: mm / dd / yyyy
- Password: (redacted)
- Re-enter Password: (redacted)
- Upload Logo: new-york-jets.jpg
- Organization Name: W Organization
- Tell us more about your Organization: test
- Street: 100 lombard st
- City: San Francisco
- State: California



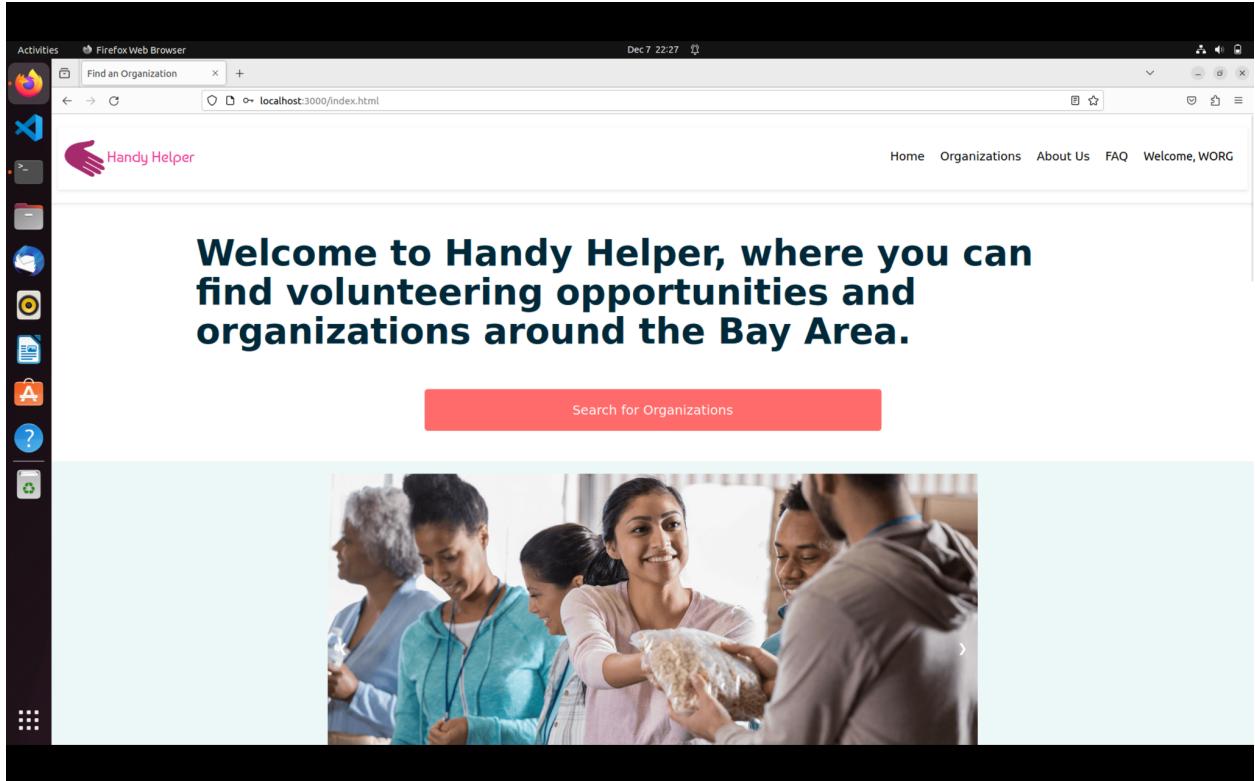
The screenshot shows the continuation of the organization signup process. The form fields from the previous step are still present, along with additional fields:

- Organization Name: W Organization
- Tell us more about your Organization: test
- Street: 100 lombard st
- City: San Francisco
- State: California
- Zip Code: 94112
- Website: <https://www.newyorkjets.com/>

Contact Information

- Name: Jacob Lazzarini
- Phone Number: 925-559-5995
- Contact Email: falconfan1150@gmail.com

Dynamic header after org sign up:



Orgs on search page:

A screenshot of the "Find an Organization" search results page. The top navigation bar includes "Handy Helper", "Home", "Organizations", "About Us", "FAQ", and "Sign In/Sign Up". The main heading is "Find an Organization". Below it is a search bar with dropdowns for "within" (set to 25 miles) and "City/State/Zip/Postal Code", and a "Search" button. Two organization cards are listed: "The Papi Chulos" with a logo featuring the text "Papi Chulo" and the subtext "This is just a test Org"; and "Generic Organization" with a small thumbnail image and the subtext "Based in Santa Clara!".

clicking on a orgs card:



The Papi Chulos

This is just a test Org

Location

Holloway Ave, San Francisco, CA, 94132

Contact

Contact Name: Joe King

Contact Phone: 123-456-7890

Contact Email: dapapichuloz@gmail.com

Organization Website: <https://sfsu.edu>