

Verification Specifications

Shelter Connect

Team TBD - Joseph Zhang, Ryker, Schwartzenberger, Simon Bang, Sunwoo Kang

Requirements Changes:

1. The email "Contact" form on the Individual Post page will implement a CAPTCHA component to help prevent spam and other misuse of the built-in email functionality.

Verification Process:

Throughout the development process, we will utilize git and Github to create branches, analyze the changes to our code, and ultimately submit a pull request into the 'master' branch to incorporate the new features into the release candidate. When a pull request into the master branch is created inside of our Github repository, Travis CI will pull that code, attempt to build/transpile it, and kick off automated testing via Mocha. These automated tests are described below. Once complete, we will review the logs generated by this process and will reject the pull request if the build or test processes failed. The individual who creates the pull request will also review a checklist, composed of each manual test specified in this document and initial their name next to each item in the checklist. This step will be completed and posted into the Pull requests message thread before the request is approved and the code merged into master.

Website Shell/Navigation/Footer:

Requirements:

1. The application should be designed and implemented in such a way to be functional only on desktop browsers
2. Nav bar would contain logo of our website on the left, in when clicked, would return user to the homepage
3. The menu on the right would contain the items that link to a registration page, sign in page, and about us page.

Manual Tests:

- Verify that appearance of the navigation bar and footer match design as laid out in requirements specification at 1366x768, 1920x1080, and 2560x1440 screen resolutions.
- Content passes WAVE (<https://wave.webaim.org/>) accessibility testing with 0 errors.
- All links in Navigation to Sign in, Register, and About pages take the user to the expected page.

Automated Testing:

- None.

Database:

Requirements:

1. Donation requests, account information, and user logins should be stored in a secure, highly available database
2. Organizations will be able to register for an account with our site, sign in, and sign out.
3. The application will allow for homeless shelters and non-profit organizations working with homeless people to create "donation requests"

Manual Testing:

- ~~Verify performance and stability statistics in the project's Firebase console. Report on any discrepancies in the Pull request for discussion.~~
- None.

Automated Testing:

- Create new Organizational user by submitting test@test.com and password "test" to Firebase. This test is verified upon a success (the create user function returns a "truthy" value).
- Sign in to user rykerls@uw.edu with password "test". This test is verified upon a success (the create user function returns a "truthy" value), indicating the user is signed in.
- After successful sign in, verify that getting the current user returns the account registered to rykerls@uw.edu.
- Sign out the current user. This test is successful if an error isn't thrown by the signout function.
- Verify that the current user is no longer rykerls@uw.edu, or any other authenticated account. This test will return null when behaving properly.
- Create new donation request populated with dummy data. Success for this test is achieved when an ID for the request is returned, rather than an error.
 - Fake donation request example:
 - Ptr_user_account: A1XZiKPwQBQKXYZ1yYIV1RMU4dr1 (derived from our test account's userid)
 - Donation type: Personal Care
 - Date_added: <today's date>
 - Information: We are requesting any used jackets that we can give to the homeless! We are stockpiling now so that we can be prepared for next winter.
- Close the test request with the ID stored above. Success is achieved when a "truthy" value is returned by the function, and not an error.

- Get request detail for ID stored above. Success is constituted when an error is returned, rather than the information input on the requests creation.

Organization Profile:

Requirements:

1. The organization's information should be automatically populated into the fields for users to view.
2. Verified users of the organization should see an "edit" button and be able to edit organization information and save it.
3. Donation request list should be defaulted to requests made by that specific organization.
 - a. It will be initially organized by the date that the requests were made.
4. The organization's title should be hyperlinked with a link to their website, if they provided one when they registered.

Manual Tests:

- Verify that appearance of the component match design as laid out in requirements specification at 1366x768, 1920x1080, and 2560x1440 screen resolutions.
- Content passes WAVE (<https://wave.webaim.org/>) accessibility testing with 0 errors.
- Test the filter of the list and verify with expected output (the filter should be set to by organization with that specific organization selected as a filter option)
 - Clearing the filters will reorganize the list by the dates that the requests were made.
- Test if the "Edit" button works as intended (The description, email, phone number, and address turns into text boxes containing the current text which they can edit) for verified users of the organization.

Description	"We are a homeless shelter that provides the proper care for those in the street. Come join our cause!"	"We are a homeless shelter that provides food, clothing and other essential resources for those that are stuck in the streets. Please join us in our mission to fight homelessness in Seattle!"
Email	shelterconnectexample@gmail.com	edittest@gmail.com
Phone number	(123)456-7890	(098)765-4321
Address	1111 NE 15th Ave Seattle, WA 98105	1234 SW 51st Ave Seattle, WA 98105

- Test if saving after editing actually saves the information and displays the correctly updated information.
- Test to see if the create new request button brings them to for verified users of the organization.
- Test to see if clicking the organization title opens up a new tab that leads to their website.

Automatic Tests:

- Fetch test organization profile and test its json properties (id, name, contact #, email, address, url) to expected properties to make sure fetches are working
- Check if filter is set to correct option (filter specifying that specific test organization)

About Page:

Requirements:

1. Vision, Problem, and Solution Statement should be displayed in custom styling for users to read.
2. Information about the developers should be located at the bottom of the page.

Manual Tests:

- Verify that appearance of the component match design as laid out in requirements specification at 1366x768, 1920x1080, and 2560x1440 screen resolutions.
- Content passes WAVE (<https://wave.webaim.org/>) accessibility testing with 0 errors.
- Confirm that vision, problem, and solution statement are displayed with correct CSS styling for user experience with eye test
- Verify that the names, roles and pictures of the developing team is located at the bottom.

Automatic Tests:

- Confirm that text stored in html is correct text in each statement by pulling the text stored in the html and comparing it to expected text

Homepage:

Requirements:

1. There will be a banner image with a short message explaining our mission and a button to see the donation requests should be displayed.
 - a. When the button is clicked on, the banner area should be hidden and the list of donation requests below should take up the unused space, and the navbar should be fully displayed.
2. Below this button, there will be text and a link directing new organizations to our Organizational Sign-up page.

3. There should be a link to a login screen for organizations to access their account and settings, which can be found in the navbar.
4. There should be a list of donation requests from the homeless organizations that can be filtered through many different aspects.
5. Each listing on this page should display an icon representing the donation type, the name of the Organization, their location for drop off, and an optional piece of text including more information.
 - a. The 'more information' text snippet should be truncated to ~200 characters if it exceeds that. The full text will be displayed on the Post detail page
6. When a listing is clicked on, the user should be taken to the Individual Post page for that listing.
7. Posts will automatically be taken down after a 30 day period. Organizations can resubmit requests.

Manual Tests:

- Verify that appearance of the component match design as laid out in requirements specification at 1366x768, 1920x1080, and 2560x1440 screen resolutions.
- Content passes WAVE (<https://wave.webaim.org/>) accessibility testing with 0 errors.
- Refresh page to check if list updates.
- Test the filters of the list and verify with expected output.
 - The latest created request posts will be shown at the top of the list initially. When other filters are selected, the posts will be reordered in the specific way that it is meant to be ordered in
 - Ex: Filtering by location will put the nearest requests to the top. In the case that there is a "tie" in terms of how far the organization is, they will be ordered by the latest created post.
- Check that the "clickable" parts of the page direct to the intended pages.
 - Donation requests button
 - Org sign-up page
 - Login screen for organizations
 - Individual posts

Automatic Tests:

- Simulate a 30 day period since the request has been posted to make sure that the post gets automatically taken down.

Individual Post:

Requirements:

1. There should be the correct information about the organization and their request.
 - a. Subject, request type, request donation and request dates.
2. The organization's contact information should be displayed in a separate section.

3. There should be a section for an integrated email section on the post, where the user can directly input their information and send an email body to the organization.
4. There will be a CAPTCHA that users will have to complete in order to send their email.
5. There should be a submit button at the bottom right of the form.
 - a. The button sends an email from the default website email to the organization's email.
 - i. Replies will go to the provided email to begin the email chain.
6. There should be an edit button for organizations so that they have the option to edit their own posts.
 - a. When clicking the edit button, the organization should be able to start editing from the information that is already there.
7. There should be a submit button to finalize any updates.
8. There should be a delete button for organizations so that they can delete their posts once they have received the proper donations or decide that they no longer need the request post.

Manual Tests:

- Verify that appearance of the component match design as laid out in requirements specification at 1366x768, 1920x1080, and 2560x1440 screen resolutions.
- Content passes WAVE (<https://wave.webaim.org/>) accessibility testing with 0 errors.
- Verifying that the contact form properly works:
 - Email sends properly
 - Organization replies to the provided email
 - We will use shelterconnectexample@gmail.com as an organization email initially to check if the email is sent correctly, and then we can "respond" to the user, who should have provided their information.
 - Subject and content send exactly as typed
- Verifying that the edit and submit buttons work:
 - The edit button updates the information properly

Description	"We are requesting some canned goods for our monthly neighborhood BBQ."	"We are requesting canned food, like baked beans or corn, for our monthly neighborhood BBQ! Anything is welcome!"
Start Date	5/22/2018	5/23/2018
End Date	6/22/2018	5/31/2018

Icon	Food can icon	Food icon
------	---------------	-----------

- The submit button updates the page and upon refreshing the changes are still there
- Test the CAPTCHA function to make sure that an email will not go through if it is not correctly completed.
- Check to see if pictures are correctly implemented in the icon section for the request post.
- Check to see if the delete button really deletes the post from the website and Firebase.
- Check to see that only the respective organization can see and use the edit and delete buttons. Use the rykerls@uw.edu test account to verify the edit and delete functions are unavailable for the shelterconnectexample@gmail.com account.
 - Users without organization accounts should not be able to see or use the edit or delete buttons too.
- Fill out the email, subject and body with email "rykerls@uw.edu", Subject "Test", and body "test test test". Fill in the captcha and verify a message is sent fake information to see the results of a correct *and* incorrect email.completeted

Automatic Tests:

- Simulate a 30 day period since the request has been posted to make sure that the post gets automatically taken down.
-

Organization Registration:

Requirements:

1. The correct input lines should be present on the page.
2. Each input line should take in the right kind of information.
3. If the required data is not entered, the user cannot submit their application.
4. The Terms of Agreement should be present and readable.
5. If the Agreement of Terms button is not clicked, the user cannot submit their application.
6. There should be a submit button at the end of the form that will verify all the information is input correctly.

Manual Tests:

- Verify that appearance of the component match design as laid out in requirements specification at 1366x768, 1920x1080, and 2560x1440 screen resolutions.
- Content passes WAVE (<https://wave.webaim.org/>) accessibility testing with 0 errors.
- Verify that the input lines and Terms of Agreement are displayed correctly.
- Test the submit button to make sure that the correct alert message shows up when a required field is missing.

- Test that the password input line shows asterisks instead of the actual password.
- Test to see if clicking the submit button after filling out the fields correctly sends the user to the home page.

Automatic Tests:

- Fill in fake account information to all of the fields to see how it handles correct *and* incorrect data.
- Make sure incorrect or missing data does not go through.

Organization Profile:

Requirements:

1. There should be correct information displayed pertaining to the organization and their mission.
 - a. Organization name, logo, website and/or description.
2. The organization's contact information should be separate and highlighted for emphasis.
3. The organization's current requests will be listed for easier user access, filters still available.
4. Organizations should have an edit button that will allow them to update any information on their account page.
5. Organizations should have a new request button so that they can make new posts.

Manual Tests:

- Verify that appearance of the component match design as laid out in requirements specification at 1366x768, 1920x1080, and 2560x1440 screen resolutions.
- Content passes WAVE (<https://wave.webaim.org/>) accessibility testing with 0 errors.
- Verify that all the information is in the correct locations.
- Verify that the edit and new request buttons are only visible and working for the correct organization.
- Verify that the current request list is correctly organized and that the filters work.
 - This means that the request list is only full of posts that are from the organization itself.
 - The requests would be ordered by the latest created posts initially.
 - Filters include the types of requests and by date (newest/oldest), which will organize the list accordingly. If there are "ties" in the type of request or date, it will be organized alphabetically.
- Check to see that only the respective organization can see and use the edit and new request buttons. Use the rykerls@uw.edu test account to verify the edit and new request functions are unavailable for shelterconnectexample@gmail.com.

- Test that users that do not have an organization account cannot also use or see the edit or new request buttons.

Automatic Tests:

- None.

Sign In Page:

Requirements:

1. There should be an input line for organizations to enter their email and another one for their account password.
2. The password should only show asterisks when entered to ensure some sort of security.
3. There should be a sign in button that users press after they filled in their email and password.
 - a. If the credentials are not correct, the user will be alerted and will have to re-enter their information.
4. There should be a reference to the registration page in case the organization hasn't signed up yet.

Manual Tests:

- Verify that appearance of the component match design as laid out in requirements specification at 1366x768, 1920x1080, and 2560x1440 screen resolutions.
- Content passes WAVE (<https://wave.webaim.org/>) accessibility testing with 0 errors.
- Verify that the correct input fields are visible and in place.
- Verify that the password input line shows only asterisks instead of the actual characters.
- Make sure that the sign in button works correctly and that if the information that is input is incorrect, an alert will show up to notify the user that either their account username or password is incorrect.
 - Specific tests:
 - rykerls@u.edu
 - shelterconnectexampe@gmail.com
 - Random passwords
 - Random usernames
- Make sure that the link to the registration page works correctly.

Automatic Tests:

- Fill in fake information to all of the fields to see how it handles correct *and* incorrect data.
- Make sure incorrect or missing data does not go through.

Organization Request Post Form:

Requirements:

1. The organizational request form should only be accessible to logged in organizations from their profile page.
2. The form will have an end date that the user can specify, combined with the date of the posting, this constitutes how long the request will be shown in the public facing list.
3. Contact information like phone number, email, and address that are provided on account sign up are auto-filled in the form.
4. Organizations will be required to specify one of the donation types on this form.
5. There will be a text box that allows the user to input a short (no more than 300 words) explanation of the specifics of their donation request.
6. There should be a submit button that users press after they filled in the correct information about their request.
 - a. If the required information is not filled in, the user will be alerted and will have to re-enter the information.

Manual Tests:

- Verify that appearance of the component match design as laid out in requirements specification at 1366x768, 1920x1080, and 2560x1440 screen resolutions.
- Content passes WAVE (<https://wave.webaim.org/>) accessibility testing with 0 errors.
- Verify that the correct input fields are available for use.
- Verify that only signed in organizations can access this page.
- Verify that the organization's information is correctly pulled from the database.
- Make sure that the submit button works correctly and that if the information that is input is incorrect, the correct alert will show up to notify the user.

Automatic Tests:

- Fill in fake information to all of the fields to see how it handles correct *and* incorrect data.
- Make sure incorrect or missing data does not go through.