|  |
| --- |
| Project Group 7 |
| VolunteerNow |
| CEN4010 Principles of Software Engineering, Summer Semester 2023 https://github.com/JonathanG77/CEN-4010-Project Version: 1.0|7/25/2023 |

|  |
| --- |
| Jonathan Garcia jggarcia2020@fau.edu Ernesto Benavides ebenavides2016@fau.edu Jacob Campbell campbellJ2019@fau.edu Tivvon Cruickshank tcruickshank2018@fau.edu Kareen Godder kgodder2020@fau.edu  7-25-2023 |

Table of Contents:

[Product Summary 2](#_Toc141217200)

[Usability Test Plan 3](#_Toc141217201)

[QA Test Plan 7](#_Toc141217202)

[Code Review 11](#_Toc141217203)

[Security Self Check 15](#_Toc141217204)

[Adherence to Original Non-Functional Specs 17](#_Toc141217205)

# Product Summary

Have you grown tired of trudging through multiple other volunteer sites looking for the right opportunity? Finding an opportunity after hours of searching only to discover there’s no way to apply right then and there?  Look no further than VounteerNow, the one stop site for all your needs relating to bettering your community. Both new visitors looking for their first volunteer opportunity and experienced volunteers will enjoy our suite of quality-of-life features that are all built directly into the system; no third-party app download or site redirects required. Our team is committed to making the process convenient so volunteers can spend more time doing what they love. Only at VolunteerNow, where we turn talk into action.

Key Features

* Robust search with various filters to allow for opportunities with certain attributes to be searched for. Allows users to search by location, skill needed, distance from user, etc.
* Account system that will allow users to log in and check various attributes such as opportunity schedule, organization announcements, and messages from organization leaders and other volunteers.
* Ability to allow users with an account to apply for opportunity jobs directly through the system. Application forms are able to have limited customization by organizations as needed for unique opportunities.
* Ability to send and receive messages/email from organization leaders and other volunteers who are volunteering at the same organization.

# Usability Test Plan

The usability test for VolunteerNow will involve an evaluation the system key features and the user experiences on the web page. Our goal will be to identify any issues with the usability, get feedback and make improvement to have a better user-friendly experience, we have a series of steps to follow the usability test plan:

**1.**    **Defines our goals**:This consists of defining our objectives for the usability test. One example will consist of evaluating its use, efficiency, and performance to our users while navigating on VolunteerNow.

**2.**    **Identify User profile**:We need to define the audience we are planning to target for VolunteerNow. A few examples can be new volunteers, experienced volunteers and of course the leaders of the organizations who are seeking volunteers who matches their profiles.

**3.**    **Scenarios:** Creating realistic scenarios that represent the common task a user will have in VolunteerNow. A few examples include creating an account, using the match-search browser, editing your profile among other scenarios.

**4.**    **Enroll Participants:** Enrolling participants whose profile matches the identified user profiles, also try to gather a diverse group in that way we are obtain a diverse comprehensive range of feedback and able to gather the information for future modifications or improvements.

**5.**    **Test Environment:** We will need a location that is suitable for the usability test, making sure the conditions are as close to a real-world experience. We will need the equipment to perform the test on each volunteer, which could be either a computer or a mobile device.

**6.**    **Introduction and Questionnaire:** Introducing the purpose of the test to the participants and explaining how the process will go. We collect some demographic information and ask a few questions regarding their experience with the volunteering platform as it’ll be part of our data.

**7.**    **Usability Test:** Explaining each participant we walk through the scenarios, and we observe their interactions with the VolunteerNow platform. We do our best to encourage them to explore beyond their curiosity, this will be essential to gather feedback.

**8.**    **Data Collection:** We will proceed to collect all the participant’s actions, comments, opinions, and feedback from the usability test, we will gather any information regarding any difficulties they may had, any confusion, their overall satisfaction and what we can improve from it.

**9.**    **Post-test Questionnaire:** After the usability test is completed and we get the feedback and data from our participants we ask them to fill in a post -test questionnaire. Here we can use the Likert-scale questions to explain their experience and add open-ended questions to get more personal and detailed feedback from them.

**10.Data Analysis:** After the test and questionnaires are completed we collected all that data, now is time to analyze and identify any common patterns, pain points , errors , also positive aspects from our platform.

**11.Report & Recommendations: With** all the data collected we prepared a usability test report with a summary of findings, areas that needs to be improved, recommendations to enhance the platform, any errors or bugs, and the user experience.

**12.Iterative process:** After our results we can work with the development team to make all the improvements necessary, correct any errors.

**13.New testing:** We will conduct more usability tests after the changes and improvements are implemented, ensuring the errors have been addressed and repeat the cycle if necessary.

With these proper steps VolunteerNow will gather the feedback needed from actual real volunteers, be able to understand their needs and preferences and their challenges they come across to helps us enhance the platform’s usability and performance. With this we will be able to provide a good experience not only for volunteers but also for the organization leaders..

# QA Test Plan

1) Test Objectives:

Verify the system's usability and functionality.

Look for any issues or problems that can impair the user experience.

Verify that the system satisfies the required specifications for post creation and accurate search functionality. Check the system's compatibility with two distinct browsers. Look for any issues that can affect the user experience: The objective of thorough testing is to find any mistakes that can potentially affect the system's functionality. Confirm that the system satisfies the criteria set out for post creation: The goal is to guarantee that all relevant features and capabilities are successfully executed. By completing these goals, the testing procedure will improve the VolunteerNow system's overall quality and dependability.

2) Hardware and Software Setup:

 Hardware: Standard computer system with internet connection.

 Software: Latest versions of the supported browsers (Chrome, Firefox).

3) Features to be Tested:

Volunteer Search: The ability to search for volunteers based on specified criteria.

Post and Search Functions: Possibility to post volunteer opportunities and look for open opportunities using the post and search features.

Each volunteer opportunity's title, description, and location should all be correctly retrieved by the system and displayed.  We can make sure that the VolunteerNow’s system's post and search tools are dependable, and easy to use by extensively testing these functions. In addition, the system must appropriately display and save the uploaded data, guaranteeing that all information is correctly saved and accessible to users.

4) Test Cases and Results:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test # | Test Title | Test Description | Test Input | Expected Correct Output | Test Results |
| 1 | Create Volunteer Post | Create a new volunteer opportunity post | Title: Fundraising Coordinator | message confirming creation of the post | PASS(Chrome,Firefox) |
| 2 | Search by Location | Search for volunteer opportunities by location | Location: Boca Raton | List of volunteer opportunities at the location | PASS(Chrome,Firefox) |
| 3 | Empty Search | Search without entering information | Empty | Error message indicating the need for search criteria | PASS(Chrome,Firefox) |

5) Test Summary:

All three test cases passed on both browsers. The system properly produces volunteer postings, performs location-based searches, and appropriately manages empty searches.

Test Case 1: Create Volunteer Post - This test case was designed to confirm that posting a new volunteer opportunity was functional. The volunteer opportunity's title, description, location, category, day, and time were all test inputs. A successful message confirming the post's successful creation was the anticipated result. The test was successful on both Chrome and Firefox, demonstrating that users may easily create volunteer postings.

Test Case 2: Search by Location - This test case examined the location-based search feature.

The test input was the location ("Boca Raton"). A list of volunteer opportunities that matched the supplied area was the expected and correct outcome. The test was successful on both Chrome and Firefox browsers, proving that the search feature successfully locates volunteer opportunities.

Test Case 3: Empty Search - This test case examined how the system responded when a search was conducted without any search parameters. There was no test input used in the test. A message highlighting the need for appropriate search criteria was the expected outcome. The test was successful on both Chrome and Firefox, demonstrating that the system encourages users to provide appropriate search terms when using the search system.

Overall, in both Chrome and Firefox browsers, all three test cases were completed successfully, achieving the goals of verifying functionality.

6) Test Execution:

Test #1- Chrome: PASS

Test #1- Firefox: PASS

Test #2- Chrome: PASS

Test #2- Firefox: PASS

Test #3- chrome: PASS

Test #3- Firefox: PASS

# Code Review

'''

The main application file for server

'''

# The code written is very readable and lacks redundancy, very clear and to the point

# Since the code uses many renown packages from the Flask library, security is not much of a concern of the code listed below and another interaction with data or the database is secured

# There could be some more testing for this code as small unit tests should suffice

# The code is separated nicely and practices modularity and encapsulation by having each function do its own respective thing and minimizing the work used throughout all of listed functions. Each function is able to focus solely on what it’s main job because of the allowed reusability from the models

# good using models to minimize work within code

from common.database import Database

from models.user import User

# Good use of imports, can put them together so they are closer and faster to read

from flask import Flask, render\_template, request, session, make\_response, url\_for, redirect

app = Flask(\_\_name\_\_, static\_url\_path='/static')  # '\_\_main\_\_'

# security issue, can be hidden as an environment variable or at least with git.ignore

app.secret\_key = "Nuggets"

@app.route('/')

def home\_template():

return render\_template('home.html')

@app.route('/login')

def login\_template():

return render\_template('login.html')

@app.route('/register')

def register\_template():

return render\_template('register.html')

@app.before\_first\_request

def initialize\_database():

Database.initialize()

@app.route('/login', methods=['POST'])

# clean code, non-redundant, each function has its own purpose, appropriate use of variables and could use more comments to highlight breakpoints

def login\_user():

email = request.form['email']

password = request.form['password']

if User.login\_valid(email, password):

     User.login(email)

else: # update to where we say user is not valid try again

     session['email'] = None

return render\_template("profile.html", email=session['email'])

@app.route('/register', methods=['POST'])

def register\_user():

email = request.form['email']

password = request.form['password']

User.register(email, password)

return render\_template('profile.html', email=session['email'])

if \_\_name\_\_ == '\_\_main\_\_':

app.run(port=4995, debug=True)

Security Self Check  
Major assets we are protecting:

* User data such as username, user passwords, user phone numbers, company numbers, and any form of email or physical addresses, held in the database of the software.
* Volunteer posts: Information shared by volunteers about their skills and availability.
* Organization data: Data related to organizations and their announcements/notifications.

1. Confirm encrypted passwords:   
   Most user information is kept protected but user passwords are kept extremely secure using an encrypted system in the database of the software. The reason that this is done is to protect the users of the website from digital theft due to company database breaches. A small number of hackers are capable of breaking through firewalls and getting into databases; to ensure that our clients will never have all of their information compromised, all user passwords will be heavily protected. The best thing that we can do as a company is put as many blockades in front of the best hackers to provide our system that they enjoy without the worry of losing important assets and we do that through our password encryption.  
   Confirm Input data validation:   
   The reason that we validate the information that is processed is to make sure that the data that is inputted by the user is used in the correct mannerism and follows the correct criteria so it is stored correctly in the database. Input validation is considered in the use-cases that we have provided in the other milestone(s). With that being said the direct inputs of user information can be seen explained below:

* User's names: In the username sections of an account and signing in the system validates that only letters, and potentially numbers are used depending on the influx of population to the website. The user is not allowed to use non-character letters and if they are used the process will return the user to the last step of the process in the use-case entry step in the diagram.
* Phone numbers: In the phone number section of an account, validation is used to provide to the system to ensure that only numbers are being used. If characters or non-character symbols are being entered by the user then the system returns the user back to the last step of the process to correct the issue.
* Addresses: In the address section of the system validation is used to check whether or not an email with the same exact values has already been registered on the website. If the email already exists then the guest user is sent back to the last process of the use-case entry. Otherwise, the user can proceed with the address used.

# Adherence to Original Non-Functional Specs

1. Performance: In the performance spec, we will be making sure that the system is capable of handling a higher population of users to ensure that the system responds quickly to all interactions.   
   Status: [On Track]   
   Explanation: We are working on testing the optimization of the system through testing the system. We believe by the time that we are capable of releasing VolunteerNow that the system will be capable of performing efficiently for users.
2. Security: We are going to implement many security measures to ensure that no one has access to information that should not be authorized to the public. We will also ensure that multitudes of private information is encrypted.   
   Status: [On Track]  
   Explanation: The validation efforts that we take will make it easy for the system to ensure that there are no security breaches through the use-cases but having a strong encryption at the core of the website for sensitive data allows for us to prevent anything bad from happening despite our website being secure. Every website has vulnerabilities, the best way for us to protect our users is by encrypting data that should not be in the hands of anyone else.
3. Scalability: The system is going to be designed to be capable of holding space in the database for new communities to grow. The space created will be created upon new user data entries.  
   Status: [On Track]  
   Explanation: The architecture of the system is designed to be capable of handling the new information and processing it in the correct way. We have also implemented some horizontal scaling methods to be able to work with the user to provide what they need for their user.
4. Reliability: We will ensure that the system is up and running at all times through testing before release as well as keeping track of the system processes efficiently. The database and storage of the website will be updated rigorously throughout.  
   Status: [On Track]  
   Explanation: There will always be downtime on every website, especially ones with high volumes of clients working on the website ritualistically. We plan on working toward finding the most efficient ways to send operations and processes throughout the VolunteerNow system. We will test the system every few days to ensure that we are providing the user with the best performance possible and look to disrupt any excess resource allocation throughout the system. On top of that we are going to work towards making small changes to the website that will increase efficiency as we see fit. To decrease the number of errors that the website may occur we will implement underlying system functionalities that will repeat throughout processes to make the website more stable.
5. Usability: We will create user-friendly interfaces as well as test for higher efficiency interfaces to make users enjoy our website more than other competitors. Each important process will have functional and balanced interfaces to meet their criteria.   
   Status: [On Track]   
   Explanation: The ability for a user to browse throughout VolunteerNow will be catered towards each individual user through processes. We want to focus on making sure that each individual process is usable, efficient, and somewhat flexible. The user will appreciate a clean and functionable process regiment throughout use-cases.
6. Accessibility: We want to make sure that the system is usable throughout all conditions of VolunteerNow. In certain places around the world there are certain blockades that a website must foresee, we will attempt to make the system work despite any blockade.  
   Status: [On Track]  
   Explanation: We will work towards implementing multitudes of interfaces that will work with the user to support them as they work their way around the website. The interface will include many different support features such as allowing the user to turn on the ulterior text for images as well as compatibility for CPUs of different processing capabilities.  During the testing phase we will also be implementing many more features to allow people to interact on the website despite hearing or vision impairments without having to leave the website.
7. Expected Load: We want to ensure that the system will be able to handle all of the volume and extra inputs that the system will have to operate through without having the performance begin to scale downwards.
8. Status: [On Track]  
   Explanation: We are testing the loading time on all of the different interactions that the use-cases bring the user through. The results that we are getting early are a good indicator of whether or not a system works because the volume on the system is not high yet. We have a good idea of what we will need to do to improve the system based on the fluctuation of each variable. The performance of our system should not be an issue
9. Storage: We will be working with the volume of clients and other memory to ensure that as necessary we find ways to store information when the system needs more space to work with.  
   Status: [On Track]  
    Explanation: The planning that we have combined with the knowledge of how efficient the VolunteerNow system is pre-release allows for us to better work with the system to provide the storage to continue growing. The system will continually grow if we work towards an efficient and strong website but that will not work without the proper storage and maintenance of storage.
10. Availability: We will make sure to make the system capable of running for all users, while also ensuring that our platform is open to use for all people.
11. Status: [On Track]   
    Explanation: The amount of work put into the testing, storing, and growing will allow for us to maintain availability for all users despite the system differences through configurations made to the system. We will make sure that multiple users are capable of using the website at all times and reduce lag on inputs for all users as well.
12. System Backup and Recovery: We will implement an operation to allow for the system to restore itself to its previous processor functionality.
13. Status: [On Track]   
    Explanation: With this functionality we will be able to restore information lost as well as restore previous functionality to compromised organizations. We have been working on implementing this backup system to make sure that data is safe as well as restorable for the users of the website for any circumstances that may arise upon further development.

The areas for improvement have been showing themselves as we have been working towards finding a functional state for VolunteerNow. The non-functional requirements of the system will guide our development team towards creating and further developing the areas of the website that we will be working on to improve further. All issues that come from implementation, performance, security, and user experiences will show throughout the testing that we will do, and we will be working extremely hard to ensure that the system works efficiently and as presented in each interface and process.