Statement of Purpose

The problem for this project was trying to incorporate everything into a working program. (Have you ever wanted to know if the password you entered was a legit sufficient password?) Well try my password meter website, all you have to do is make sure you have a working computer and click the index file provided. Once the index file is clicked, you will then be prompted to enter some information on the website. Once all the information is filled out, the website will output a score based on the information entered. This password meter will allow you to know if the password you want to use is a low scoring, medium scoring, or high scoring password. There is a break down underneath by each category which will give the user a score beside each one and can help the passwords be improved. The purpose of this Visual Studio program is to help create a safe way to know if the password the user has entered meets the minimum requirements of a password. First the program will ask the user for their personal information. The user will be promoted to enter their name, phone number, and age. It will be excepted that the user doesn't use any of the personal information in their password or else the program will prompt the user to enter another password without that information being contained. The minimum requirements for a password is that it is at least eight characters in length, includes an upper case letter, includes a lower case letter, numbers, and symbols. Also, in my website I included the top 100 passwords and if the user enters a password that is in the top 100 list then they will also be prompted to enter another password.

Research & Background

There was a lot of research done and looked into many password meters online to determine what the pros and cons are of each password meter. The main website used was www.passwordmeter.com. jQuery was something that I had to do a lot of research on because I was not as familiar as I needed to be with jQuery to fulfill the needs of this project. The website https://www.w3schools.com/jquery/jquery_examples.asp was what I used to become familiar with jQuery and the functions and variables that need to be used. I had to go through many tasks to mimic the password meter website but I did include a couple different things to make the project more sufficient. I decided to have the information that the user entered stored on the page upon refresh except for the password entered and the score output. That way if the same user wants to check another password then they do not have to enter their name, age, and phone number all over again. All they will have to do is enter their desired password and the score will output based on the password the user entered. In order to get the password score to actually output properly, I used the code from http://www.todnem.com/. For the top 100 password part, I looked on https://en.wikipedia.org/wiki/Wikipedia:10,000_most_common_passwords and created a function where if the user enters any of the top 100 passwords then they will be prompted to enter another password that is not included in the top 100 passwords list.

Project Language/Software/Hardware

I used a couple different of languages in this website. The languages used in this website are HTML, CSS, JavaScript, and jQuery. In order to incorporate all of those languages into one program, I decided to go with Visual Studio code. Visual studio was a great tool to use to be able to add different languages and get the outcome needed. The hardware used was Intel x86-84 which is the hardware on Mac computers.

Personal Motivation:

This project put my skills to test based on what I have learned over the years. I had to dive deeper into what I was taught and piece everything together. I feel as though, this project will help prepare for what I will be doing in my career field. I chose this project to show students how fun it can be being a cybersecurity major considering cybersecurity has to do with hacking, networks, data breaches, attacks, threats, etc.

Project Requirements

Requirements:

- Create the actual website (how the user will be able to click the index file and view the website)
- Create the name input box (how the user will be able to input their name)
- Access name entered by user (how the website will be able to make sure the name is not contained the password)
- Create the age input box (how the user will be able to input their age)
- Access age entered by user (how the website will be able to make sure the age is not contained the password)
- Create the phone number input box (how the user will be able to input their phone number)
- Access phone number entered by user (how the website will be able to make sure the phone number is not contained the password)
- Set the password requirements (Minimum of 8 characters in length, ¾ with a symbol, uppercase letter, and lowercase letter)
- Display the password score (this will provide the user with a score that way if the password score needs to be improved then they can modify the password)
- Refresh page and information should be saved except for the password and password score (that way the user won't have to keep entering their information)
- Be able to access folders on computer (this will allow the user to run the index file)

- Research the top 100 passwords (that way if the user enters one of the top 100 passwords then they will not be able to proceed)
- Display the different categories and a score by each (how the user will know what section of their password they should improve upon.

Project Implementation Description & Explanation

The project I chose to do is a personal password meter website using Visual Studio Code. I created a website to give users the ability to enter their information such as their name, phone number, and age. Figure 1 is how the website looks once it loads and shows all of the information needed for the user to enter. The requirements for the password is that it must be a minimum of eight characters in length, and must contain ¾ of uppercase letters, lowercase letters, and symbols. Figure 2 shows each category which will display a score beside it once the user enters their desired password. Figure 3 shows how the website will look once the personal information is entered in each box. Once the user enters their name, age, and phone number then the password box will have a message under it displaying “use a few words, avoid common phrases.” Common phrases should not be included in the password that the user enters because it may be easy for others to guess. If the user includes their name in the password then the website will display like figure 4 stating “name cannot be in the password.” Also, if their age or phone number is included in the password then the same message will display but instead it will show “age cannot be in the password” or “phone number cannot be in the password.” Once none of that information is included then the website will look like figure 5 and will display a score. If the user scrolls down then they will be able to see the different categories. The categories will have a score beside each based on the password that the user entered. If they want to improve the password then they will need to look by the different categories and see which category has the lower scores.

Figure 1: Loading Screen

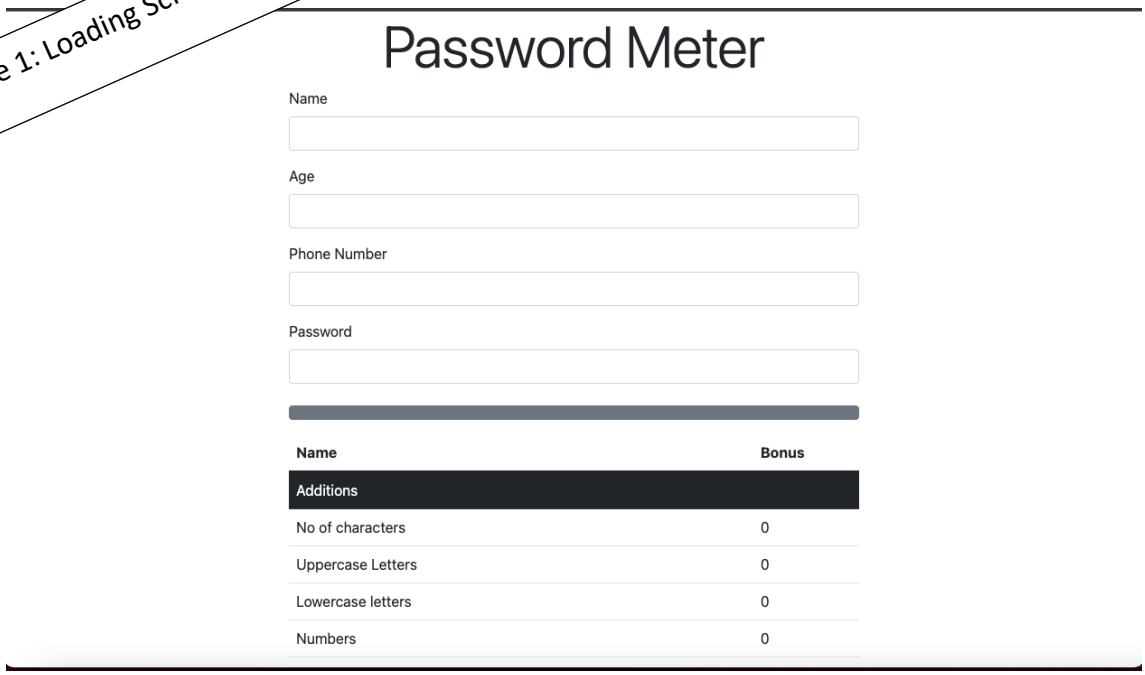


Figure 2: The categories that will display a score by each once the password is entered

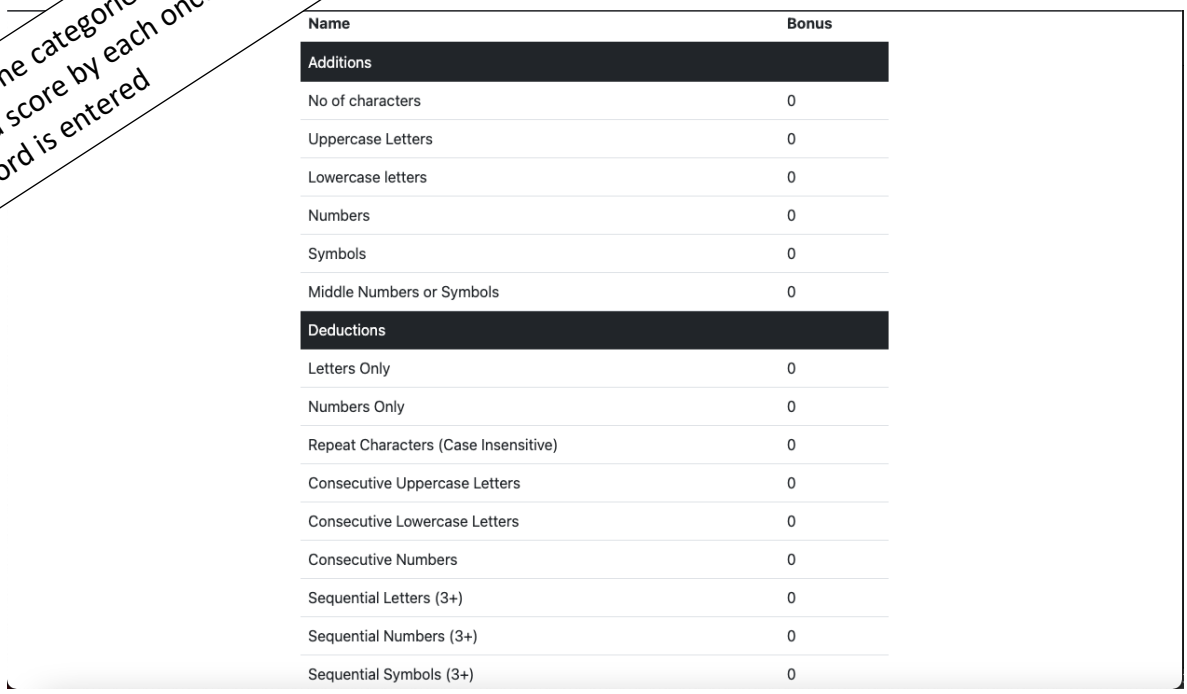


Figure 3: How the screen will look once the personal information is entered

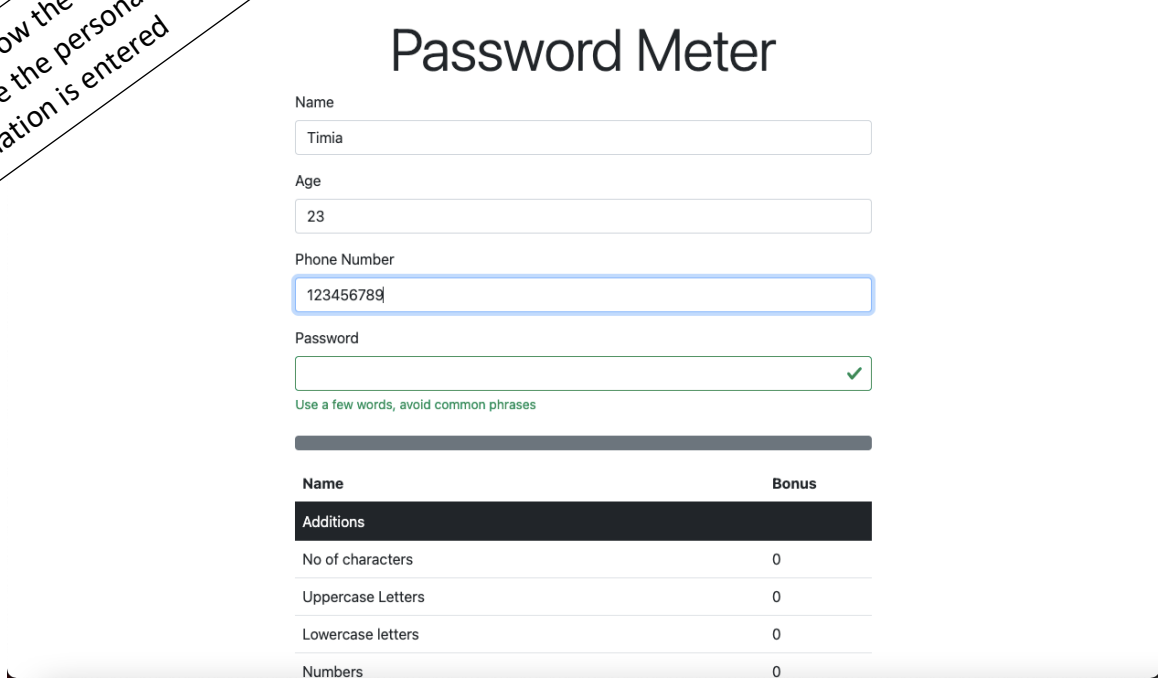


Figure 4: How the website will look if the name, age or phone number is contained in the password.

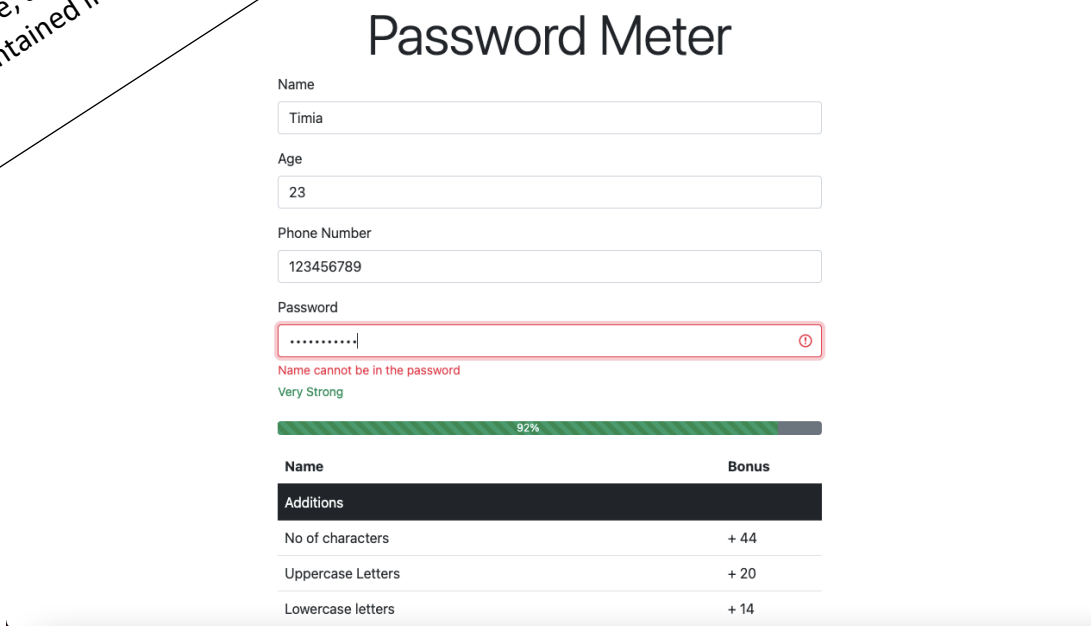
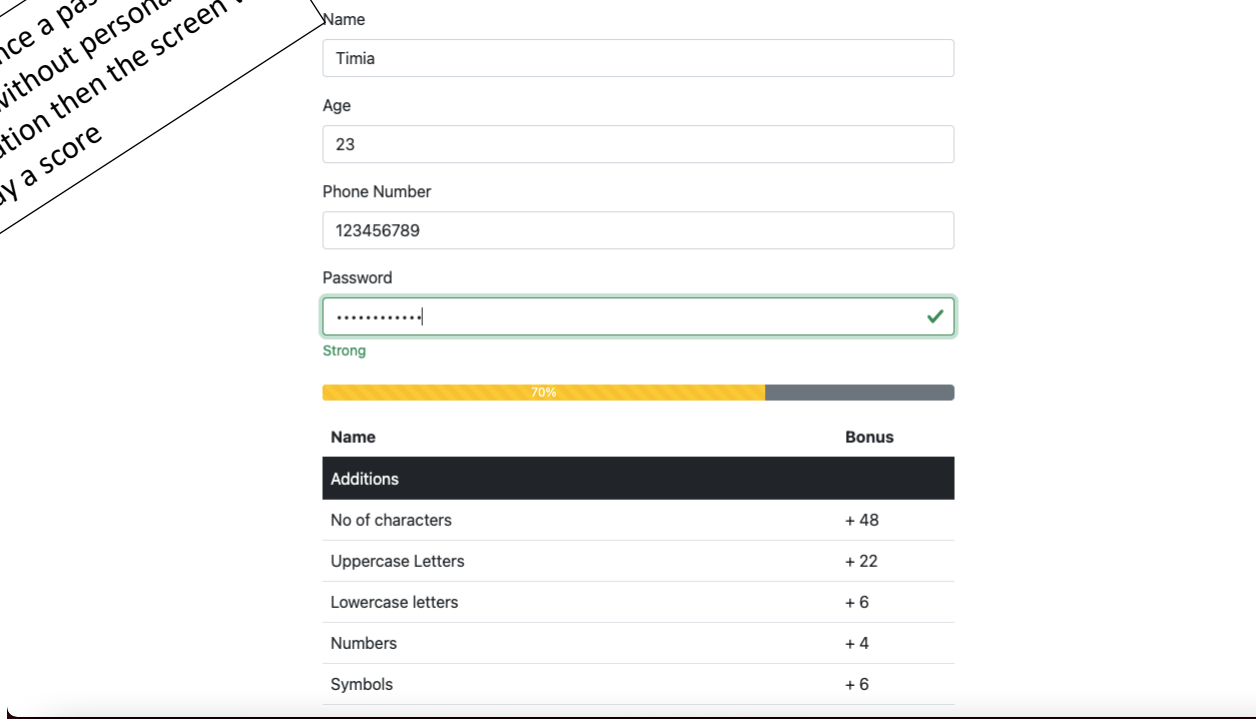


Figure 5: Once a password was entered without personal information then the screen will display a score



Test Plan

Test Plan Identifier: TP1

Introduction:

Overview:

I tested out the website to make sure it works for all computer users. The score output was the most tedious thing to accomplish because of the different categories and making sure the score outputted beside each category. Once the website loads, the user is responsible for entering their name, phone number and age. Once that information is entered then they will need to enter their desired password while making sure to meet the password requirements and also not including their personal information in the password or they will not be able to proceed. Since the website is complete, I had users test on different laptop/computers which have different operating systems and took notes of the outcome.

Usability Testing: I did some usability testing to find out if the website works on both windows computer as well as a Mac computer. Some friends tested the website to make sure it works properly.

Goals/Objective: Got the website working, had users test the website to make sure it works, determine if the information saves upon refresh except for the password and password score.

Constraints: Test the website to make sure it works for all computer users.

References:

- Project Plan (Link)

Test Items:

The items needed for testing are:

- Computer
- Visual Studio Code

Features to be Tested:

- List the features of the software/product to be tested:
- HTML
- Coding features
- Whether the website displays everything in the correct format and font is readable.

Features Not to Be Tested:

- List the features of the software/product which will not be tested.
- Specify the reasons these features won't be tested.

Approach:

- The overall approach of testing was so I knew how to move about completing the final website. After getting the website functionality working then I proceeding in the coding and adding features to enhance the website.

Item Pass/Fail Criteria:

- Specify the criteria that will be used to determine whether each test item (software/product) has passed or failed testing:
- Testing the website on a my Mac computer to see if it works as far as basic functionality.
- Tested the website on a Windows computer to see if it works.
- Got friends to test the website on their laptops/computers
- Got ratings from friends to see if the website was actually a good idea or not. Any rating lower than a 3 out of 5 would result in a failure.

Test Deliverables:

- List test deliverables, and links to them if available, including the following:
  - Test Plan **(Insert link this document itself)**

Test Environment:

- The test environment will include the following systems: Mac OS Big Sur 11.2.1, Windows
- Testing will be performed by the developers regularly and I will have participates test the website after completing it.

Schedule:

- Provide a summary of the schedule, specifying key test milestones, and/or provide a link to the detailed schedule.
- (Project proposal link)
- 10/13/21 – At least have the main functionality of the website working.
- 11/09/21 – Have the information saved upon refresh besides the password and password score.
- 11/05/2021 - Usability testing with participants

Responsibilities:

- Get the basics of the project working and displaying on the website (Done)
- Implementing a design similar to the password meter site (Done)
- Implementing the project to get what the user inputted (done)
- Implementing the different categories which is a breakdown of what the password contains (Done)
- Making sure the website is tested and working (Done)
- Code up any final touches (Done)

Risks:

- Not having anything functioning properly so that the website can properly display everything.
- Possible passwords being saved if the password doesn't go away upon refresh.
- People may be so stuck into making sure the password meets all the requirements and the score being outputted and may use this password for everything and any account they may have.
- Cookies being encrypted

Test Results

Users	Score	Feedback
User1	Could not proceed because their personal information was contained in password.	Enjoyed the website, liked the fact that the website displayed what personal information was contained in the password. (3 out of 5)
User2	70%	Suggested to make the font a bigger size so it is readable for everyone. (3.5 out of 5)
User3	59%	This user felt like the website was better than passwordmeter.com because it was secure and felt the categories was helpful when trying to improve the password. (5 out of 5)
User4	80%	Have the information saved upon refresh except for the password and score so the user won't have to keep entering their information (3 out of 5)

User5	72%	Liked the idea that the information was saved upon refresh so the user just had to enter a password and the score outputted. (4 out of 5)

Challenges Overcome

Honestly, I had to overcome many challenges. One of the first challenges was that I had a partner and we were working on a Scavenger Hunt for CSU where there would be clues given and the person who is working on the scavenger hunt would have to find the items based on the clues. Also, the scavenger hunt had geolocation incorporated so the location would display. However, I found out at the beginning of this semester that my partner was no longer my partner and they had changed their major. After finding out that information, I decided that I didn't want to follow through with the scavenger hunt and wanted to go a different route. I saw the password meter website and decided to create a program similar. Time played a big issue in this because I had to get all of this done quickly and meet all the deadlines set. At first, I started working on the program with Java because I have a Mac computer and it's much different than what can be done on a Windows computer. After working on the program in Java, I then decided that Java was not going to be able to achieve what I was hoping. I wanted to essentially create a mobile app using Java and XCode to make sure the information was protected. Well that didn't go how I wanted because XCode kept freezing on my laptop. I then changed my plans and started working on all of this in Visual Studio that way I would be able to incorporate a website using html, CSS, JavaScript, and jQuery.

Future Enhancements

Some future enhancements would be allowing cookies on the website because in order to use HTTP cookies can help developers with websites. Cookies allow the websites to remember you and some websites do not work as they are intended to work without cookies. On the down side, cookies can be encrypted. Also, every so often, the website should be freshened with up to date content that will attract users to click the website and use it. I also think that social media share buttons should be included on the site. Since social media is a big part of society today then it will make it easier for users to share the website directly through social media which will attract more individuals. In addition, I think implementing this into a app in the future would be very helpful. There are many websites and companies that require passwords to have certain requirements like the ones I put in place on the website. An app would make it secure for users to safely type their password without having to worry about any hackers or attackers trying to get the information that was entered. The app should be created for both Android and iPhone users and should allow the app to be downloaded directly from the App store or Google play store.

Defense Presentation Slides