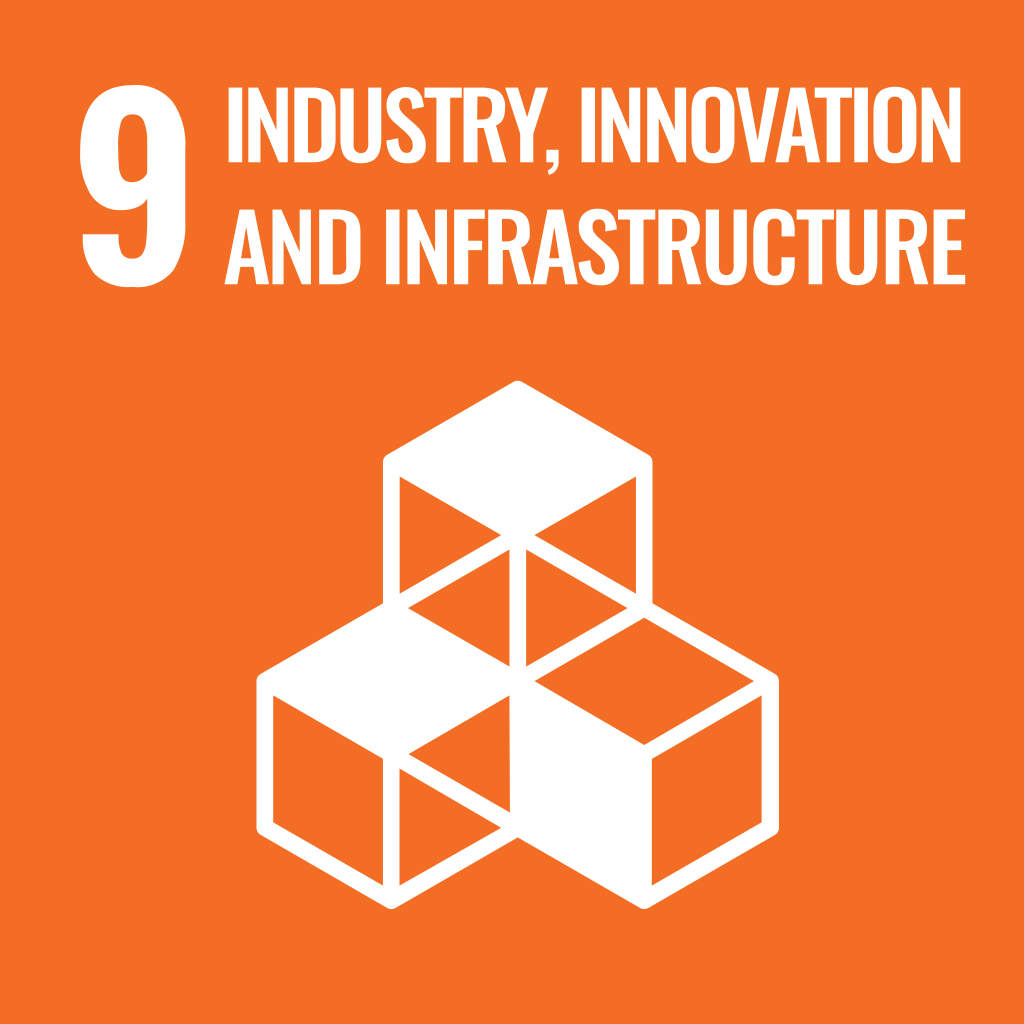
****



**FINAL YEAR PROJECT**

**Advanced Password Strength Assessment Tool**

**By**

**CHEW FAN YEE**

**TP060853**

**APU3FS308CS(CYB)**

**B.Sc. (Hons) Computer Science Specialism in Cyber Security**

**Supervised by Mr. David Tan**

**2nd Marker: Mr. Yogeswaran Nathan**

**2024**

Contents

[ACKNOWLEDGEMENT 4](#_Toc165385809)

[ABSTRACT 5](#_Toc165385810)

[CHAPTER 1: INTRODUCTION 6](#_Toc165385811)

[1.1 Introduction 6](#_Toc165385812)

[1.2 Problem Background 8](#_Toc165385813)

[1.3 Project Aim 10](#_Toc165385814)

[1.4 Objectives 10](#_Toc165385815)

[1.5 Scope 11](#_Toc165385816)

[1.6 Potential Benefit 14](#_Toc165385817)

[1.6.1 Tangible Benefit 14](#_Toc165385818)

[1.6.2 Intangible Benefit 14](#_Toc165385819)

[1.6.3 Target User 14](#_Toc165385820)

[CHAPTER 2: LITERATURE REVIEW 16](#_Toc165385821)

[2.1 Introduction 16](#_Toc165385822)

[2.2 Domain Research 16](#_Toc165385823)

[2.3 Similar Systems 17](#_Toc165385824)

[2.3.1 Microsoft Password Checker 17](#_Toc165385825)

[2.3.2 Dropbox zxcvbn 18](#_Toc165385826)

[2.3.3 Google Password Checkup 18](#_Toc165385827)

[2.4 Comparison 19](#_Toc165385828)

[CHAPTER 3: METHODOLOGY 20](#_Toc165385829)

[3.1 System Development Methodology 20](#_Toc165385830)

[3.2 Data Gathering Design 21](#_Toc165385831)

[3.3 Analysis 25](#_Toc165385832)

[CHAPER 4: DESIGN AND IMPLEMENTATION 33](#_Toc165385833)

[4.1 Introduction 33](#_Toc165385834)

[4.2 System Design Diagram 34](#_Toc165385835)

[4.3 Use-Case Diagram 35](#_Toc165385836)

[4.4 Use-Case Specifications 35](#_Toc165385837)

[4.5 Class Diagram 36](#_Toc165385838)

[4.6 Activity Diagram 36](#_Toc165385839)

[4.7 Sequence Diagram 37](#_Toc165385840)

[4.8 Interface Design 37](#_Toc165385841)

[4.9 Execution 38](#_Toc165385842)

[4.10 System Screenshot 40](#_Toc165385843)

[4.11 Summary 43](#_Toc165385844)

[CHAPTER 5: RESULT AND DISCUSSION 44](#_Toc165385845)

[5.1 Introduction 44](#_Toc165385846)

[5.2 Unit Testing 44](#_Toc165385847)

[5.3 User Acceptance Testing 45](#_Toc165385848)

[5.4 System Testing and Discussion 49](#_Toc165385849)

[5.6 Summary 49](#_Toc165385850)

[CHAPTER 6: CONCLUSION 51](#_Toc165385851)

[6.1 Critical Evaluation 51](#_Toc165385852)

[6.2 Limitation 51](#_Toc165385853)

[6.3 Recommendation 52](#_Toc165385854)

[REFERENCES 54](#_Toc165385855)

[APPENDIX 56](#_Toc165385856)

[PPF – Title Registration Proposal 56](#_Toc165385857)

[Ethic Forms 67](#_Toc165385858)

[Log Sheets 72](#_Toc165385859)

[Poster 73](#_Toc165385860)

[Gantt Chart 74](#_Toc165385861)

[Sample Code Implementation 75](#_Toc165385862)

[Respondence Demographic Profile 78](#_Toc165385863)

# ACKNOWLEDGEMENT

I extend my heartfelt gratitude to all those who have played a pivotal role in the realization of my Final Year Project. Without the unwavering support and contributions from various individuals, this milestone would not have been attainable.

Foremost, I express my sincere appreciation to my supervisor, Mr. David Tan, of the School of Technology. With the guidance and encouragement from Mr. Tan, it have been instrumental in providing me with invaluable knowledge and support throughout the development of my Final Year Project. Mr. Tan's expertise in the field of Cyber Security not only steered this project in the right direction but also significantly enriched its content. The patience and constructive feedback he gave were constant motivators, pushing me to strive for a higher standard of excellence in my work.

In addition to my supervisor, I want to thank my friends, my family, and those unnamed individuals who have generously offered their help and advice. The words of encouragement, networks, and shared research insights have been indispensable in shaping the content and direction of this investigation report. I am optimistic that the groundwork laid in this initial phase will greatly contribute to the development of the system in the second half of the project, resulting in the creation of a good Final Year Project.

# ABSTRACT

Motivated by the current world’s pervasive threat of cyber breaches resulting from weak passwords, this research employs advanced methods to assess and strengthen password integrity. Through a systematic analysis of existing password-related vulnerabilities, the project investigates and proposes comprehensive solutions to mitigate risks. The methodology involves a meticulous examination of current password security practices, leveraging cutting-edge technologies to design an assessment tool capable of evaluating the strength and vulnerability of passwords. Preliminary results indicate promising advancements in password protection, with ongoing efforts dedicated to refining and optimizing the tool's functionality. The project's significance lies in its potential to fortify digital defences against unauthorized access, contributing to the overarching goal of creating a secure and resilient digital infrastructure. Aligned with Sustainable Development Goal 9, "Industry, Innovation and Infrastructure," the Advanced Password Strength Assessment Tool plays a crucial role in promoting technological innovation and fostering resilient information systems.

Keywords: Cybersecurity, Password Assessment, Digital Security, Innovation, Information Systems, Password Strength Evaluation, Cyber Threats.

# CHAPTER 1: INTRODUCTION

## 1.1 Introduction

It is through our contemporary digital landscape that technology permeates every facet of our lives, the imperative for robust cybersecurity measures has reached unprecedented levels. As passwords stand as the initial line of defence against unauthorized access to sensitive information, making their strength a pivotal factor in ensuring online security. However, the disconcerting reality persists, which is the substantial portion of the global population continues to employ weak or easily decipherable passwords, exposing themselves to potential cyber threats.

This project centres on the swiftly evolving realm of cybersecurity within the business and industry sector. Faced with the escalating frequency and sophistication of cyberattacks, individuals and organizations must proactively adopt measures to safeguard their digital assets. It is within this context that our project assumes significance.

The vulnerability posed by weak passwords extends beyond mere inconvenience; it encompasses substantial risks to personal and organizational security. Incidents of sensitive data breaches can result in financial losses, identity theft, and compromise the privacy and safety of individuals and communities (Groeneveld, 2022). Addressing this issue becomes not just a matter of convenience but a fundamental tenet of digital citizenship in today's interconnected world.

Aligned with the 9th Sustainable Development Goal (SDG), "Industry, Innovation, and Infrastructure," this project epitomizes the potency of innovation in the cybersecurity domain. By introducing a user-friendly, efficient, and accessible approach to assess password strength, our aim is to contribute to creating a safer digital environment. Strengthening passwords, in turn, fortifies the very infrastructure of digital systems, offering protection to individuals and organizations against potential threats.

To realize this project's objectives, a comprehensive exploration of key research areas is imperative. This includes an investigation into best practices and standards for password security, an examination of existing assessment methods, and an exploration of algorithms for password strength. By delving into these research areas, the aim is to craft an innovative solution that not only empowers individuals to safeguard their digital identities but also aligns with the broader objective of constructing a secure, innovative, and resilient digital infrastructure in harmony with the 9th SDG.

## 1.2 Problem Background

In the era of digital advancement, the security of personal and organizational data pivots on the resilience of passwords. Despite an escalating awareness of cybersecurity risks, a substantial number of users grapple with the creation and maintenance of robust passwords. AustralianMutualBank (2023) underscores the crux of the issue, attributing it to users' inadequacies in knowledge, motivation, and effort – factors often insufficient in generating passwords resilient against contemporary cyber threats. Existing research has delineated key challenges within password security:

* User Knowledge Gap: One prominent challenge in password security stems from a knowledge gap among users. Misconceptions persist, equating password strength solely with complexity, leading to convoluted combinations of letters, numbers, and characters. This misunderstanding tends to let the user fosters passwords that are challenging to recall and prone to errors, exacerbated by a lack of clear guidance on crucial factors such as length, uniqueness, and unpredictability.
* User Motivation: User motivation emerges as another obstacle in password security. Convenience frequently takes precedence over security, as the onus of remembering intricate passwords for multiple accounts becomes burdensome. Some users underestimate the risks associated with weak passwords, assuming they are unlikely targets for cyberattacks.
* User Effort: The substantial effort required to create and manage robust passwords poses a significant challenge. Juggling unique, strong passwords across numerous accounts is time-consuming, fostering "password fatigue." Users may succumb to reusing passwords or employing weak ones due to the overwhelming nature of constant password management.
* Password Reuse: Widespread password reuse, driven by simplicity and convenience, constitutes a critical security concern. This practice exponentially magnifies the impact of security breaches, granting unauthorized access to multiple accounts when one password is compromised.
* Cybersecurity Threats: The ever-evolving landscape of cybersecurity threats adds a persistent challenge. Cybercriminals employ new methods, including brute force attacks and advanced cracking software, necessitating continual adaptation of password practices. The prevalence of compromised password databases on the dark web further underscores the need for enhanced password security measures.

This project innovatively extends prior research in password security by providing a tangible tool for comprehensive password strength evaluation. Unlike previous efforts focusing on user education and complex rule recommendations, this project empowers users with real-time feedback tailored to their unique passwords. Bridging the gap between theoretical knowledge and practical implementation, it aims to tackle the root causes of weak password security.

Furthermore, aligning with recent trends in cybersecurity, the project emphasizes user-centric security solutions. Instead of solely relying on users' informed decisions, innovative algorithms will objectively assess password strength, aiding users in making more informed choices.

In conclusion, while prior work laid the foundation for password security awareness, this project represents a substantial leap forward by offering a practical solution that directly addresses user knowledge, motivation, and effort concerns. By enhancing the user experience and promoting strong password practices, it aspires to contribute to a more secure digital environment aligned with contemporary cybersecurity needs.

## 1.3 Project Aim

The aim of this project is to create a user-friendly and impactful tool that enables individuals to generate and sustain robust passwords, ultimately fortifying their digital security. This endeavour seeks to address the disparity in user knowledge, motivation, and effort related to password security. Aligned with the 9th Sustainable Development Goal, the primary objective is to advance innovative solutions fostering digital security, thereby aligning with broader initiatives for a secure and resilient digital infrastructure.

## 1.4 Objectives

In pursuit of fortifying digital security in an era rife with cyber threats, this project delineates clear objectives aimed at developing an advanced Password Strength Assessment Tool. The overarching goal is to create a comprehensive solution that addresses the critical aspects of user-friendly accessibility, robust algorithmic assessment, and user education within the realm of password security.

* To create and deploy a user-friendly application that facilitates the assessment of password strength, incorporating the development of an interface tailored to user-centric principles.
* To formulate resilient password strength assessment algorithms, capable of delivering precise and instantaneous feedback by analysing diverse factors, including password length, character variety, predictability, and resilience against common cracking techniques.
* To enlighten users on effective password security practices by integrating educational and informative resources within the tool, thereby heightening user awareness, and understanding of secure password practices.

## 1.5 Scope

The "Advanced Password Strength Assessment Tool" embarks on a comprehensive journey to empower individual users in fortifying their digital security through a range of targeted functionalities. The scope of this project is intricately designed to address the unique needs and objectives of individual users, fostering an environment where password security is not only assessed but actively improved. Within this framework, several key tasks are outlined to ensure the successful development and deployment of the tool.

Scope of Tasks:

* Password Strength Assessment and Feedback:
  + Users will have the ability to evaluate the strength of their passwords.
  + The tool will provide real-time feedback on the assessed strength of passwords, offering users immediate insights into the security level of their credentials.
* Personalized Recommendations:
  + The tool will analyse weaknesses in users' passwords.
  + Users will receive personalized recommendations tailored to the specific weaknesses identified, guiding them on how to strengthen their passwords effectively.
* Educational Resources:
  + The tool will incorporate a repository of educational materials and resources focused on password security practices.
  + Users can access informative content that enhances their understanding of best practices, fostering a culture of cybersecurity awareness.
* User Monitoring and Enhancement:
  + The tool will provide a user-friendly interface for regular monitoring of password security.
  + Users will be able to actively enhance their password security based on the feedback and recommendations provided by the tool.

While the password strength assessment tool aims to provide a comprehensive utility for individual users, certain limitations exist. Firstly, the focus on empowering individuals to adopt improved personal password habits means enterprise-grade access management capabilities get excluded from scope. Additionally, inherent technology constraints influence the practical size and sophistication possible across the spectrum of desired features. Advanced encryption, decentralized storage protocols and biometric authentication elements will stretch the bounds of existing methods. Though leveraging cutting-edge technologies in design, balancing complex security with system performance and stability remains imperative. Ultimately the tool must deliver a reliable, user-centric experience guiding better passwords; a pragmatism which shapes decisions around balancing functional possibilities against practical achievement.

Centered on enabling individuals to securely manage personal credentials, several core capabilities define the project’s development commitments. Intuitive design topping the priorities manifests through accessible interfaces navigable for tech-novice users. Robust algorithms follow, assessing password constructs across an array of vulnerability factors like length, randomness, hash strengths. Embedding educational resources takes the next spot, equipping users as informed, active cybersecurity stewards. This ethos of nurturing self-efficacy permeates the fourth focal function - continuous monitoring tools that encourage persistent, proactive safety habits versus reactive ones. Through these fused facets spotlighting usability, evaluation, guidance and habit-building, the password tool aims users can conveniently yet securely operate in modern digital ecosystems. While constrained in enterprise scope, individual empowerment persists as the central driver across all planned capabilities to make personal credential management comprehensible, actionable, and most critically, safe.

While seeking an encompassing solution for individuals, certain capabilities get consciously excluded from design scope. Tailoring usability for average consumers implies enterprise-specific adaptations lie outside project boundaries. Functionality like centralized dashboard oversight or group permission protocols cater more to organizational credential management versus personal. Additionally, pragmatic technology restrictions set expectations around feasible features based on encrypted data limits or interface complexity tolerances. Augmenting biometric inputs or decentralizing storage may push implementation capacity. Hence users should not anticipate every conceivable bell or whistle within any initial tool launch. However, possibilities clearly exist to incrementally expand technical boundaries over later iterations once core individual user-centric functionalities get established. For now, the focus remains students and consumers enhancing personal password habits before expanding to more advanced or niche capabilities.

In conclusion, the scope of the "Advanced Password Strength Assessment Tool" is strategically crafted to empower individual users in optimizing their password security. By incorporating assessment, feedback, recommendations, educational resources, and user-friendly interfaces, the project aspires to contribute significantly to the cultivation of a secure digital environment for individual users.

## 1.6 Potential Benefit

### 1.6.1 Tangible Benefit

The primary tangible benefit of the "Advanced Password Strength Assessment Tool" is an immediate enhancement in digital security for individual users. By actively assessing the strength of passwords, providing real-time feedback, and offering personalized recommendations, users can fortify their online accounts against potential cyber threats. This tangible improvement in password security directly translates to a reduced risk of unauthorized access, data breaches, and identity theft. The tool's ability to guide users in creating and maintaining robust passwords contributes tangibly to their overall cybersecurity posture.

### 1.6.2 Intangible Benefit

An intangible benefit of the tool is the cultivation of a heightened sense of cybersecurity awareness among users. Through the integration of educational materials and resources within the tool, users gain a deeper understanding of password security best practices. This increased awareness extends beyond the immediate use of the tool, empowering users to make informed decisions about their digital security in various online contexts. The intangible benefit lies in the long-term impact of creating a user base that is more vigilant, knowledgeable, and proactive in safeguarding their digital identities.

### 1.6.3 Target User

The primary target user for the "Advanced Password Strength Assessment Tool" is individuals across diverse demographics who seek to bolster their password security. This includes but is not limited to:

* General Users: Individuals with varying levels of technical expertise who want to enhance the security of their online accounts.
* Students: Particularly those navigating the digital landscape for educational purposes.
* Professionals: Individuals using digital platforms for work-related activities.
* Elderly Users: Tailoring the tool to be accessible and user-friendly for individuals less familiar with advanced technologies.

The tool's design and functionalities cater to a broad spectrum of users, making it inclusive and applicable to anyone aiming to strengthen their password security. The target user base reflects the tool's versatility and accessibility, ensuring that individuals from different walks of life can benefit from its features.

# CHAPTER 2: LITERATURE REVIEW

## 2.1 Introduction

In the realm of academic research and scholarly pursuits, a literature review stands as a crucial cornerstone, providing the essential framework upon which new knowledge is constructed. This comprehensive survey and synthesis of existing literature on a particular subject serve multifaceted purposes, each contributing to the depth and validity of any research endeavour. The University of EDINBURGH stated as a foundational step in the research process, the literature review serves to contextualize, analyse, and critically evaluate the body of knowledge that precedes the study, offering a roadmap for further exploration. This introduction outlines the indispensable reasons why undertaking a literature review is not merely a procedural step but an integral and invaluable component of academic inquiry.

## 2.2 Domain Research

Passwords are ubiquitous in the digital world, used to protect access to devices, accounts, services, and sensitive data. However, many users still employ weak passwords that are easy for cybercriminals to guess. Recent research analyses the continued prevalence of weak passwords and evaluates updated techniques to bolster password strength and security. Despite frequent security breaches and awareness campaigns, weak passwords like “123456” and “password” remain stubbornly common. Simple numerical and keyboard patterns, names, and dictionary words continue to dominate. The most popular passwords have not changed significantly in the last decade. The authors conclude that guidance encouraging password complexity has failed to motivate the majority to move beyond basic insecure choices.

To analyse aggregated password data sets, security researchers have developed quantitative password strength estimators. Ur et al. (2020) recently introduced an open-source tool called Password Evidence and Strength (PEAS) to provide feedback on portfolio strength for a collection of passwords. Built on neural networks, it considers the full distribution of passwords rather than assessing passwords individually. In tests, PEAS generated strength estimates better aligned with password cracking rates compared to entropy metrics. Creating more accurate models for password strength at scale enables appropriately targeted interventions to improve security.

Many proposed solutions aim to nudge users towards stronger passwords without overly disrupting their experience. Research find that requiring three-character types produces a good level of security while maintaining usability. Custom strength meters providing instant feedback during password creation may also positively influence security. Egelman et al.’s (2020) meter assessing semantics improved resistance to guessing attacks. Explicitly enforcing expiration dates continues to be controversial, as very frequent resets annoy users who then make minimal changes to their old passwords.

Recent research provides updated perspectives on the severity of weak password usage and pathways for improving security. While passwords likely will not disappear any time soon, supplemental authentication schemes provide users options for enhanced convenience without compromising security. More accurate estimating of portfolio password strength can enable policies and targeted interventions to be efficiently tailored towards those most at risk. Individual accounts now may be protected through multiple factors, but improving baseline password strength across platforms remains critical for overall security and data protection.

In 2020 and beyond, weak passwords continue to predominate despite the known risks. Through leveraging emerging strength estimators to target those most vulnerable, researching usability of composition policies and feedback systems, and integrating convenient alternative authentication schemes, cybersecurity experts are making headway in addressing this persistent problem. But much work remains to change ingrained user behaviour and prevent threats enabled through easily guessed passwords permeating so many facets of digital life.

## 2.3 Similar Systems

### 2.3.1 Microsoft Password Checker

Microsoft built a password strength evaluating tool into its Windows 10 and Windows 11 operating systems to assess new account passwords created by users. When creating an account, such as when setting up a new user profile on a Windows computer, it will analyse the password entered and provide real-time feedback. The Microsoft password checker tool looks at multiple components including length, complexity, use of common passwords, and key patterns to determine strength. It provides a coloured coded indicator highlighting passwords as very weak, weak, medium, or strong. The tool offers helpful tips during password creation, guiding the user to lengthen the password, add special symbols and numbers, and avoid reused or common passwords to strengthen it. Since it is baked into the Windows sign-up flow, it provides a frictionless way for users to create stronger passwords from the start, though the criteria and assessments are fixed based on Microsoft's guidelines, so users have limited configuration (Alexander, 2018).

### 2.3.2 Dropbox zxcvbn

Dropbox developed an open-source password analysing tool called zxcvbn that serves as a JavaScript library that developers can integrate into both web and mobile applications. It uses a very advanced set of patterns matching rules and entropy calculations to estimate the strength of passwords. Zxcvbn runs through thousands of common passwords, manipulations of keyboard patterns, repeats of characters, sequences, dates, and names to detect weak passwords. Unlike basic length and complexity checks, it can catch a wide array of password choices that follow predictable patterns generally considered unsafe (Wheeler, 1970). The tool generates an entropy score from 0 to 4 to rate password strength from highly guessable to extremely secure based on computations of all possible password permutations fitting the identified patterns. One of the most useful features of zxcvbn is that goes beyond just assigning a score, and it provides meaningful feedback about the specific weakness that was identified so developers can guide users properly.

### 2.3.3 Google Password Checkup

As one of the most widely adopted password management solutions, Google developed a simple browser extension called Password Checkup. It can be added into Chrome, Edge and other Chromium based browsers and provides a way to check on passwords already stored in users' Google accounts. Once installed, Password Checkup syncs with a user's Google products like Gmail, Drive and Android device passwords and alerts them if any credentials are reused, known to be compromised or otherwise unsafe. It flags the specific weak passwords and prompts users to immediately change them rather than just assigning an arbitrary score. Part of what makes Password Checkup user-friendly is it never asks users to manually enter any current passwords for analysing. It simply taps into passwords Google already manages across products and devices to recommend fixes for compromised ones. However, the tool does not estimate overall password strength or prevent bad passwords when initially created, it merely checks existing passwords and relies on user's own judgement for changing them.

## 2.4 Comparison

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tool | Analysis Method | Interface | Use Cases | Weaknesses |
| Microsoft Checker | Length, complexity, common passwords | During password creation in Windows | Creating new Windows user accounts | Limited scope, fixed criteria |
| Dropbox zxcvbn | Pattern matching, entropy | API/library for integration | Password validation for web/mobile apps | Computation heavy, requires coding |
| Google Checkup | Known compromised passwords | Browser extension | Checking already saved Google passwords | Only for Google products, no strength estimate |

# CHAPTER 3: METHODOLOGY

## 3.1 System Development Methodology

The Waterfall methodology will provide a structured, linear approach to the development of the password strength assessment tool. This methodology involves progressing sequentially through a series of distinct phases, with some overlap and iteration between phases. The Waterfall methodology allows for clear documentation of requirements early on, followed by orderly design, development, testing, and deployment stages. It is an appropriate choice for this project due to the clear product definition and understanding of requirements at the start, as well as the straightforward functionality to be built without need for ongoing revisions.

Atlassian stated that a major advantage of applying the Waterfall model to this project is that the product requirements are already well-defined. The goal is to develop a tool focused specifically on evaluating and scoring password strength. There is no need to spend time on exploratory requirements activities given this narrowly focused tool. The requirements can be fully specified at the very beginning, in terms of detailed security rules, password criteria scoring, interface choices, and performance requirements. These requirements then drive the downstream processes in a structured way.

Another driver for choosing Waterfall is the ability to clearly estimate schedules and costs due to the linear execution. Because each phase, such as design, coding, and testing, is executed one after another, timelines and resource needs can be accurately predicted. This ensures the product roadmap is realistic and helps with planning. Relatedly, the assumed stability of requirements also fits the direct way this password checker will be built. Once documented, the requirements will remain largely static, requiring no complex, iterative development cycles.

The phases begin with gathering all requirements related to capabilities, system interactions, security needs, and platform/language choices. These requirements are documented, reviewed, and approved before design commences. The next phase focuses on technical software and hardware designs that specify component breakdowns, interfaces, data flows, algorithms, architecture, and infrastructure. Code implementation adheres strictly to these completed technical designs. Robust testing verifies all documented requirements and design elements, first through isolated unit testing, then integration testing, and finally system verification. This Waterfall methodology provides an appropriate, structured process for this password assessment project, aligning with understood requirements, and linear development approach suitable for the well-defined product goals.

## 3.2 Data Gathering Design

A screenshot of a computer screen

Description automatically generated

The introduction paragraph explained the purpose of conducting this questionnaire - to gather insights about people's password knowledge, attitudes, and habits to inform the development of a new password strength testing tool. It established the academic nature and voluntary participation aspect of the research.

A screenshot of a phone

Description automatically generated

The demographics section with questions about age and occupation was included to facilitate analysis of any differences in password practices across groups. Understanding variances by factors like age could allow customization of the tool to match different user needs.

A screenshot of a cell phone

Description automatically generated

Five questions comprised the general password knowledge section. This covered self-reported behaviors like password reuse tendencies, typical password length and complexity used, saving passwords in files/documents, and overall concern about weak passwords. The goal was to benchmark general security awareness and gaps that could be addressed with an enhanced strength testing tool.

A screenshot of a survey

Description automatically generated

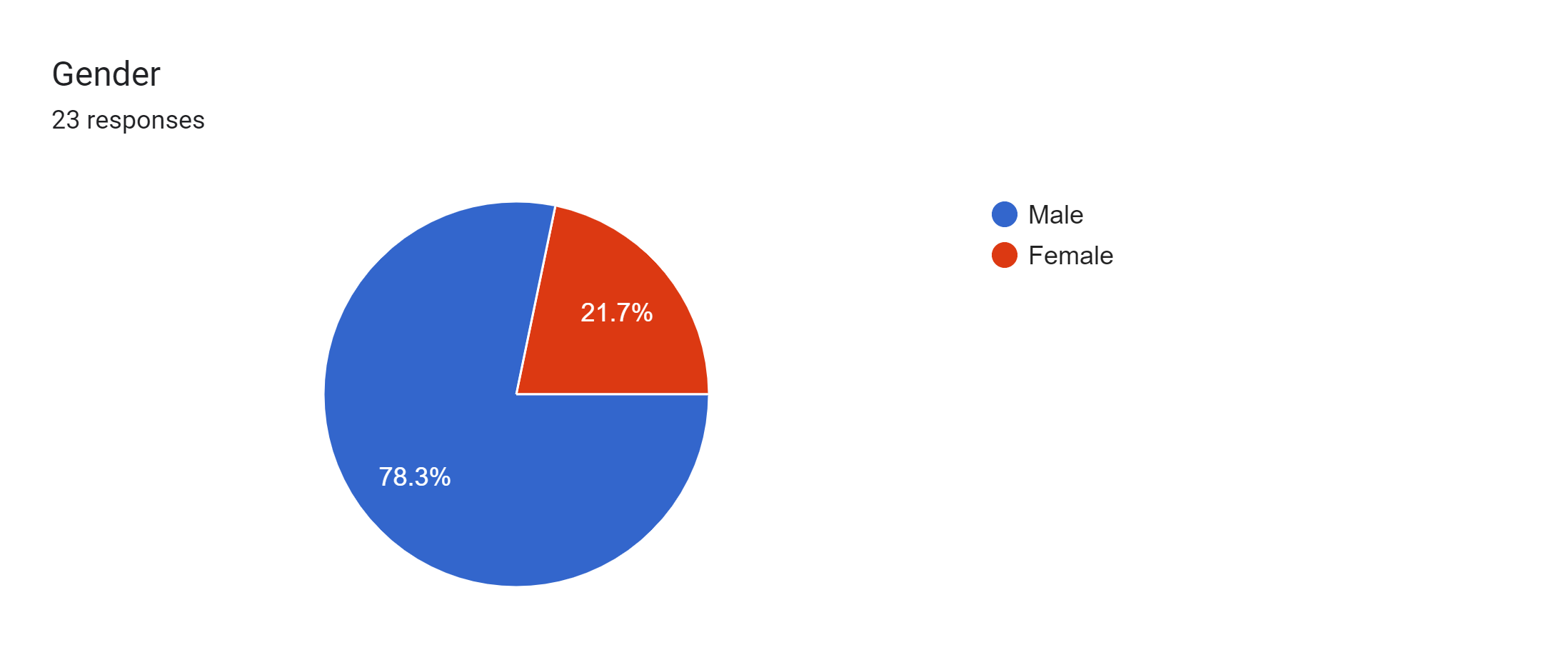
As the most directly relevant section, four questions focused specifically on receptiveness and desired features for a strength testing tool. This provided insights on tool necessity, frequency of potential usage, specific functionality preferences like password generation, and storage/manager.

A screenshot of a computer

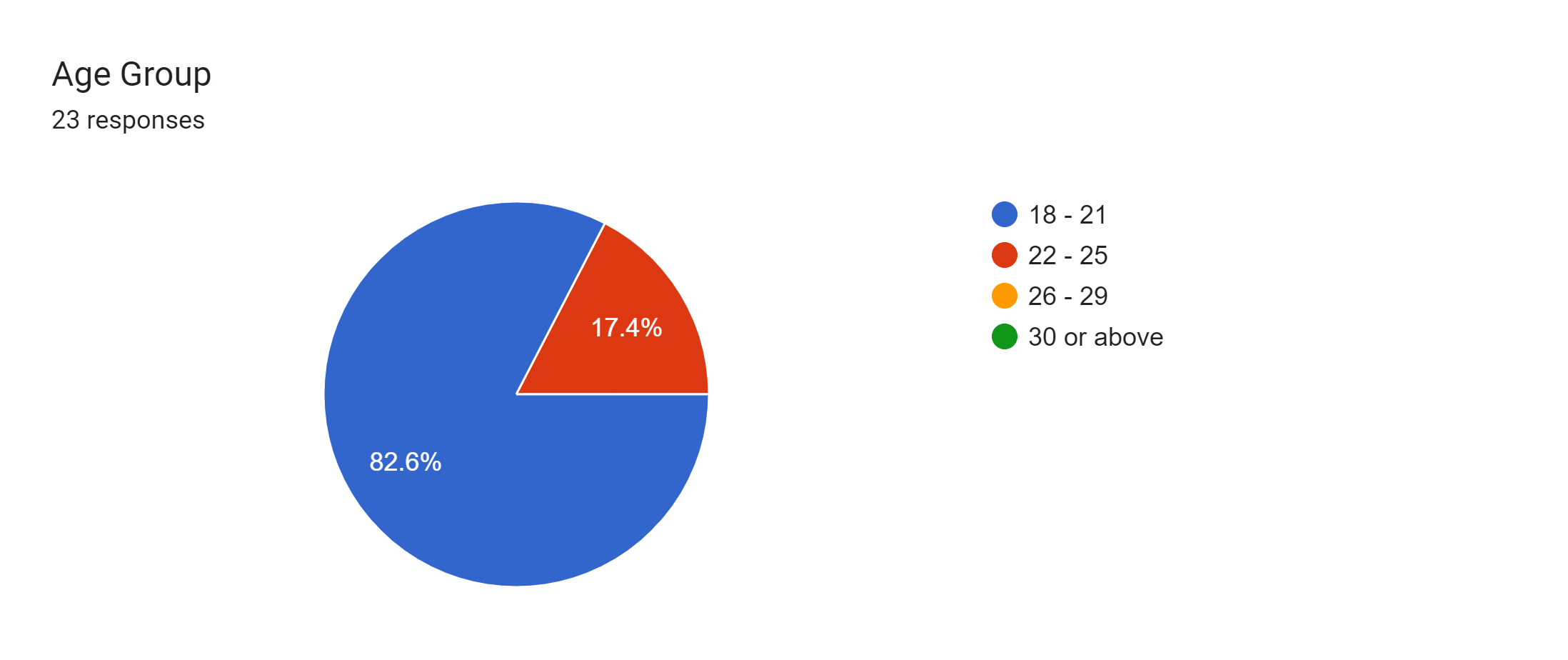
Description automatically generated

The consent section established voluntary participation, anonymity, minimal risks involved, option to withdraw, estimated time commitment and oversight contact details. Obtaining informed consent protects participant rights.

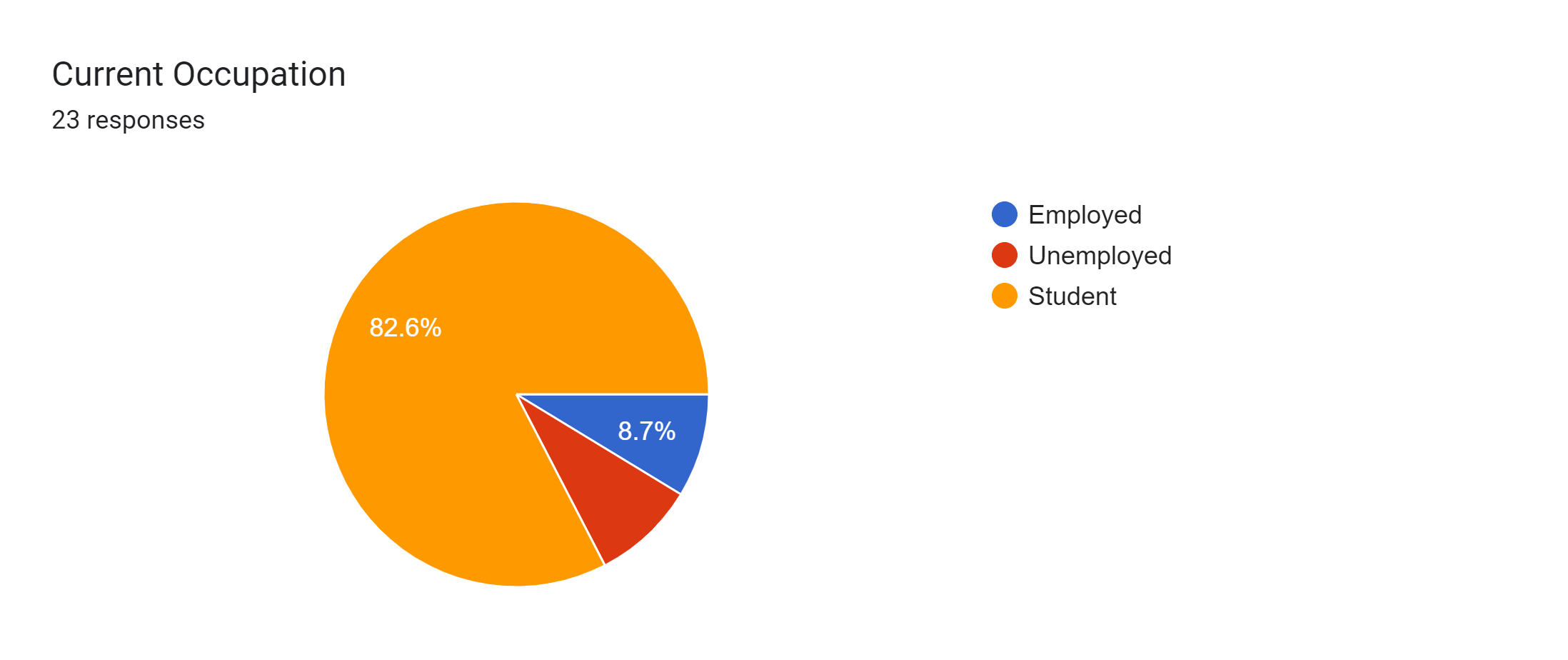
## 3.3 Analysis



The questionnaire responses revealed that 78.3% of respondents were male. The significant overrepresentation of males provides an interesting insight. It could suggest that men more actively seek out tools to evaluate password strength, indicative of comparatively higher digital security awareness. However, the high percentage of male participants also introduces potential gender bias.



The age breakdown reveals the vast majority of responses came from either 18–21-year-olds (82.6%) or 22–25-year-olds (17.4%).



Aligned with the young age, students made up 82.6% of respondents, employed people comprised 8.7%, and the remaining subset was currently unemployed.

Forms response chart. Question title: How often do you create new passwords instead of reusing old ones?
. Number of responses: 23 responses.

The responses reveal considerable concerning rates of password reuse rather than regular password refreshing, especially among such a tech-dependent demographic. The largest subset at 30.4% reported rarely creating new, unique passwords. A further 17.4% disturbingly never create new passwords at all, relying fully on recycling old logins for all accounts. Combined, close to half of participants demonstrated highly insecure tendencies by infrequently or never updating passwords.

On the more positive end, 26.1% sometimes changed passwords and 8.7% claimed to always use new, non-duplicated passwords across the accounts they set up. Interestingly, while less than 10% internally committed to best practices, 17.4% self-assessed as often creating fresh passwords. Overall, the widespread password reuse rates likely stem from perceived inconvenience around managing multiple unique logins. These quantitative results confirm poor practice baselines that urgently need addressing for younger users through remedial guidance.

Forms response chart. Question title: How long do your passwords tend to be?
. Number of responses: 23 responses.

Fortunately, most respondents claimed to use reasonably strong password lengths of 11-15 characters (39.1%) or 16+ characters (30.4%). This aligns positively with the widespread embrace of longer passwords enabling greater complexity among younger demographics. However, 30.4% also acknowledged still sticking with weaker 7–10-character passwords vulnerable to quicker guessing.

Most concerningly, no one admitted to using dangerously short 6 or fewer character passwords. This potentially indicates some positive bias in self-reported behaviour though, especially considering the high password reuse rates simultaneously reported. Some participants may not wish to openly acknowledge clearly poor security habits on the survey even when granted anonymity.

Forms response chart. Question title: Do you include uppercase letters, lowercase letters, numbers, and special characters in your passwords?
. Number of responses: 23 responses.

The data on inclusion of expanded password complexity elements proves concerning, as optimal practice involves combining uppercase, lowercase, numbers, and special characters for heightened security. However, no one claimed implementing the ideal combination. Just 43.5% came close, reporting attempting to incorporate all uppercase, lowercase, numbers, and special characters.

More respondents indicated including lowercase (56.5%) over uppercase letters (52.2%), alluding to the widespread abandonment of initial capitalization in informal messaging. Disconcertingly few leverage numbers (39.1%) often considered the easiest way to lengthen and strengthen passwords. And special characters see 0%, suggesting major perceived usability barriers. Overall, current complexity practices fall substantially short. The intention is there for many participants but not the understanding of vital special characters or tools to easily employ them.

Forms response chart. Question title: Do you write down or store your passwords in a document or file on your devices?
. Number of responses: 23 responses.

With 65.2% answering affirmatively about saving passwords in documents/files, most participants demonstrate dangerous practices jeopardizing account security despite increased reliance on digital access. This ties directly to the previously reported insights around rarely creating unique passwords and low diversity of special characters or case formats used.

Forms response chart. Question title: On a scale of 1-5, how concerned are you about having potential weak passwords that could be guessed?
. Number of responses: 23 responses.

Worryingly, close to 70% of respondents registered only minimal concern about having hackable passwords, rating just 1-2 on a 5-point scale. 26.1% showed complete apathy, not worried whatsoever. This aligns to the previously outlined risky security behaviours including reusing passwords and saving them in plain text documents.

In contrast, only 8.7% expressed heightened concern, ranking a 4. And no participant showed extreme concern, or 5, on the scale. The apparent divide between actual password hygiene practices documented and degree of concern hints at a disconnect for younger digital natives.

Forms response chart. Question title: Have you used an online password strength checking tool before?
. Number of responses: 23 responses.

Promisingly, the vast majority at 91.3% indicate awareness and access of online tools available for checking password strength, saying yes to prior use. Just 8.7% expressed uncertainty, answering maybe. However, no one outright said they have not utilized a strength testing tool before.

Combining these statistics with the numerous unsafe password practices reported an apparent paradox emerges though. Participants claim utilizing checking tools yet simultaneously engage in easily avoidable behaviours like never changing passwords and saving them in plain text files. This implies the current solutions fail at motivating meaningful improvements once users exit the tool interfaces.

Forms response chart. Question title: Would you find value in an app that checks password strength and provides feedback on how to improve?
. Number of responses: 23 responses.

Intriguingly the responses split regarding openness towards a dedicated password strength testing and improvement application. Only 39.1% definitively saw value potential, answering yes, while 26.1% flat out said no. But 34.8% sat on the fence, expressing maybe interest in such an app dependent on specifics.

Forms response chart. Question title: How often would you use a password strength testing app?
. Number of responses: 23 responses.

The largest segment at 30.4% see themselves interacting with a dedicated password testing application just sometimes, followed by 26.1% expecting rare use. However, 21.7% estimate they would use such an app quite consistently, selecting always. Just 8.7% aligned to significant but occasional use by choosing often for likely frequency.

13% appear resistant, asserting they would never actually use a testing app regardless of core purpose or helpful features. These mixed results regarding prospective engagement suggest that both spreading awareness around utility for the resistant minority and ensuring seamless embedding in natural use flows for the intermittently interested majority stand as pivotal adoption factors.

Forms response chart. Question title: What features would be most useful to you in a password strength app? Select all that apply.
. Number of responses: 23 responses.

The clear front runner feature with 87% interest is an integrated password generator to enable hassle-free creation of complex passwords, confirming earlier assumptions around desired convenience. Password storage management ranked substantially lower at 34.8%, likely due to existing reliance on insecure documents.

60.9% felt specific improvement tips would provide decent utility, suggesting receptiveness to coaching. Though based on engagement issues with current tools, proactive education may prove more effective than reactive suggestions only upon assessments. Surprisingly few opted for multifactor authentication or proactive changing reminder capabilities given security gaps identified.

The 4.3% who took initiative to manually specify other useful features championed innovations like cross-site duplicate checking and password change prompts. This shows a subset prioritizes more advanced protections but may still assume to get the baseline offerings in the suggested features without considering their absence.

Forms response chart. Question title: By voluntarily participating in this password questionnaire, you are consenting to allow your responses to be used for academic research purposes only. All data will be analyzed anonymously and no personally identifying information will be collected or linked to your responses without further expressed consent.
. Number of responses: 23 responses.

Fortunately, 100% of respondents digitally signed the consent form to voluntarily participate under common academic research ethical guidelines. This helps validate capturing and reporting insights around potentially sensitive password practices and attitudes. Granting permission to record responses anonymously enables further statistical testing for observational trends.

# CHAPER 4: DESIGN AND IMPLEMENTATION

## 4.1 Introduction

As we know it, the design and implementation of a system are two critical phases in the software development life cycle, and they play quite a crucial role in the successful delivery of a software system. To start off with design, it is the phase in the software development process where the overall architecture and structure of the software system are defined (Kovalenko, 2023). The design phase involves creating some sort of blueprint or a detailed written plan that outlines how to build the system and how the functionality will be presented. The design phase is also generally divided into two main stages, which are the high-level design and the low-level design.

Architectural design, or also known as a high-level design, focuses on defining the overall structure of the system, the components inside, and the interactions between them. This stage usually involves creating diagrams and models that represent the system's architecture, such as class diagrams, use-case diagrams, and system design diagrams (Sharma, 2024). The high-level design also acted as a suggestion towards addressing non-functional requirements, such as performance, security, and scalability of the system.

As for the low-level design, it involves specifying the internal structure and implementation details of each component or module within the system. This stage includes creating algorithms, data structures, and user interface designs (Pandey, 2024). Low-level design also involves defining the interfaces between components and determining how data will flow through the system.

The design phase is crucial because it ensures that the software system is well-planned and organized, making it easier to implement, maintain, and extend in the future. It is said that a well-designed system should be modular, flexible, and scalable, allowing for easier integration of new features or modifications as requirements change over time.

As for the implementation, it is the phase in the software development process where the design on paper is transformed into a proper working software system. It involves programmers writing the actual code that will be implemented in the specified functionality in the design phase. The implementation phase typically follows a structured approach, with developers working on individual components or modules of the system.

During the implementation phase, developers will write the system’s code in a chosen programming language, following coding standards and best practices. They also tend to integrate external libraries, frameworks, and online tools as needed to support the required functionality and enhance the development process. Testing is also an integral part of the implementation phase (Itexus, 2024). Developers often conduct unit tests to verify the correctness of the system’s components or modules, ensuring that they work as expected before integrating them into the larger system. There is also performance testing of integration testing to ensure that the components work together seamlessly.

In the end, effective design and implementation are critical for the success of a software project. A well-designed system that follows best practices and adheres to industry standards is easier to implement and maintain. Proper implementation ensures that the software system meets the specified requirements, functional, and performs as expected. It is together that these two phases lay the foundation for a high-quality, reliable, and maintainable software product.

## 4.2 System Design Diagram

A diagram of a computer program

Description automatically generated

Figure 4.1: System Design Diagram

## 4.3 Use-Case Diagram

A diagram of a password strength

Description automatically generated

Figure 4.2: Use-Case Diagram

## 4.4 Use-Case Specifications

|  |  |
| --- | --- |
| Use-Case Name | Evaluate Password Strength |
| Description | The process where user enter password into provided input field, and the system evaluate the strength of the password based on various criteria. Then display corresponding message indicating level of password strength |
| Actor | System |
| Priority | High |
| Status | Approved |
| Pre-conditions | The system is initiated and ready for user input |
| Post-conditions | The user is informed about the strength of the password through the displayed strength text message |
| Basic Flow | 1. Users enter the password into input field 2. The system calculates the strength score of entered password 3. The system maps the calculated score to the strength’s level based on predefined thresholds 4. The system displays the strength level message to the use |
| Alternative Flow | The system will display a strength text message indicating that the password is too short |
| Exception | None |

Table 4.3: Use-Case Specifications

## 4.5 Class Diagram

A screenshot of a computer

Description automatically generated

Figure 4.4: Class Diagram

## 4.6 Activity Diagram

A diagram of a computer system

Description automatically generated

Figure 4.5: Activity Diagram

## 4.7 Sequence Diagram

A diagram of a software system

Description automatically generated

Figure 4.6: Sequence Diagram

## 4.8 Interface Design

A screenshot of a computer

Description automatically generated

Figure 4.7: Interface Design

The figure above shows the main page of the running Password Strength Assessment Tool system. This page allows user to type in their password to check its strength. This page will also show the revolving links to useful password strength related sites.

## 4.9 Execution

The password strength assessment tool is a web-based application designed to help users create secure and robust passwords by evaluating password strength and providing feedback. The tool aims to raise awareness about password security practices and encourage users to adopt stronger passwords, thereby enhancing the overall security of their online accounts and digital assets.

The application is built using a combination of HTML, CSS, JavaScript, and the jQuery library. The front-end user interface is implemented using HTML and styled with CSS, providing a clean and intuitive layout for the password input field and the strength assessment feedback. The jQuery library is utilized to simplify DOM manipulation and event handling, allowing for efficient and cross-browser compatible code (Kinsta, 2023).

The core functionality of the password strength assessment is encapsulated within a JavaScript file, ‘password.js’. This file defines a jQuery plugin called ‘password’ that can be applied to any password input field on the web page. The plugin creates an instance of the ‘Password’ class, which is responsible for evaluating the strength of the entered password based on various criteria.

There are several implementations of methods in the ‘Password’ class to calculate the password strength score. The ‘calculateScore’ method considers factors such as password length, presence of repetitive characters, inclusion of numbers, symbols, uppercase and lowercase letters, and combinations of different character types. The score is then mapped to a corresponding strength text message using the ‘getScoreText’ method, which relies on predefined thresholds defined in the ‘steps’ object.

In addition to the core password strength assessment functionality, the application also includes an educational component. A revolving set of links to password security best practices and informative resources is displayed below the input field. These links cycle at a predefined interval, allowing users to explore and learn about effective password security practices.

Overall, the password strength assessment tool is a practical and educational application that aims to promote stronger password security practices among users. By providing real-time feedback and educational resources, the tool empowers users to create more robust passwords, thereby enhancing the security of their online accounts and protecting their digital assets from unauthorized access.

## 4.10 System Screenshot

A screen shot of a computer program

Description automatically generated

Figure 4.8: HTML File

A screen shot of a computer program

Description automatically generated

Figure 4.9: 1st Part of JavaScript File

A screen shot of a computer program

Description automatically generated

Figure 4.10: 2nd Part of JavaScript File

## 4.11 Summary

The system I designed is a web-based application that follows a modular and object-oriented design approach, separating concerns between the user interface and the core application logic. The high-level design adopts a client-side architecture, with the user interface implemented using HTML and CSS, and the application functionality encapsulated within JavaScript files.

The low-level design of the user interface comprises an HTML structure with the necessary elements and placeholders for displaying content and user interactions. While the core application logic is implemented in a JavaScript class, which defines methods for handling the essential functionality, such as data processing, calculations, and user input validation. This class is designed to be reusable and extensible, following best practices in object-oriented programming.

The application logic is integrated into the web application through a JavaScript library or framework architecture. This architecture creates instances of the core class and attaches event listeners to the relevant user interface elements. As users interact with the application, the event handlers trigger the corresponding application logic, and the results are dynamically rendered within designated elements in the user interface.

To enhance the user experience and provide additional value, the application incorporates educational and informative components. These components can take the form of rotating content, such as links to relevant resources or best practices, displayed within designated areas of the user interface. These components can cycle at predefined intervals, encouraging users to explore and learn more about the application's subject matter.

# CHAPTER 5: RESULT AND DISCUSSION

## 5.1 Introduction

A test plan is a detailed document that outlines the testing strategy, scope, objectives, and approach for ensuring the quality and correctness of a software system. It serves as a blueprint for the testing activities and helps ensure that all critical aspects of the system are thoroughly tested (BrowserStack, n.d.). In the context of the password strength assessment tool, I will be using 2 test plans of different techniques, a unit testing plan and a user acceptance testing plan. By combining these 2 test plans together, I can ensure that both the individual components and the overall system functionality are thoroughly tested, which will in the end increasing the likelihood of delivering a high-quality and reliable password strength assessment tool.

## 5.2 Unit Testing

A unit testing plan focuses on testing individual units or components of the system in isolation (SmartBear, n.d.). In the case of the password strength assessment tool, I will design the unit test to verify the correctness of the individual methods and functions within the system, such as the score calculations, the password plugin, and the password strength level display. Unit tests help ensure that each component works as expected before integrating it into the larger system.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test Case ID | Test Case Name | Description | Expected Results | Actual Results | Status | Priority |
| 1.1 | Calculate Score (Valid Password) | Test the calculateScore method with a valid password | The method should return strength text message based on the defined thresholds | The method returns the strength text message based on the defined thresholds | Approved | High |
| 1.2 | Calculate Score (Invalid Password) | Test the calculateScore method with an invalid password (e.g., too short) | The method should return a “too short” message | The method returns a “too short” message | Approved | High |
| 1.3 | Initialize Password Plugin (Valid Input) | Test the initialization of the password plugin with a valid input field | The plugin should create an instance of the Password class and attach event listeners to the input field | The plugin creates an instance of the Password class and attach event listeners to the input field | Approved | High |
| 1.4 | Display Strength Text (Valid Password) | Test the display of the strength text with a valid password | The strength text should be displayed in the designated area with the correct message | The strength text is displayed in the designated area with the correct message | Approved | High |
| 1.5 | Display Strength Text (Invalid Password) | Test the display of the strength text with an invalid password | The strength text should be displayed in the designated area with the appropriate error message | The strength text is displayed in the designated area with the appropriate error message | Approved | High |

Table 5.1: Unit Testing Plan

## 5.3 User Acceptance Testing

Another test plan I used is user acceptance testing (UAT), it is a type of testing that involves multiple end-user or representatives of the target audience. The goal of UAT is to validate that the system meets the specified requirements and functions as expected from the user's perspective (Setter, 2023). In the case of the password strength assessment tool, I would involve testing the overall user experience, including the user interface, password strength feedback, and educational resources in the UAT. This will helps ensure that the application is user-friendly, intuitive, and meets the intended goals and objectives.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tester Name: Chan Cay Shen | | Date: 21st April 2024 | | | | |
| Criteria | | Please tick the box for a rating of 1 to 5 (✓) | | | | |
| 1 | 2 | 3 | 4 | 5 |
| 1 | User Interface Validation   * the user interface elements are properly displayed and positioned * design is pleasing to the users |  |  |  |  | ✓ |
| 2 | Password Strength Feedback   * the tool provides accurate feedback on password strength * weak password such as "password" will indicate as a weak or insecure password * strong password such as "Str0ngP@ssw0rd!" will indicate as a strong or secure password |  |  |  |  | ✓ |
| 3 | Educational Resources Display   * the educational resource links are displayed and cycled appropriately * clicking on the educational resource links opens the respective resources |  |  |  |  | ✓ |
| Tester Comments | | - | | | | |

Table 5.2: 1st UAT

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tester Name: Low Kit Xheng | | Date: 21st April 2024 | | | | |
| Criteria | | Please tick the box for a rating of 1 to 5 (✓) | | | | |
| 1 | 2 | 3 | 4 | 5 |
| 1 | User Interface Validation   * the user interface elements are properly displayed and positioned * design is pleasing to the users |  |  |  | ✓ |  |
| 2 | Password Strength Feedback   * the tool provides accurate feedback on password strength * weak password such as "password" will indicate as a weak or insecure password * strong password such as "Str0ngP@ssw0rd!" will indicate as a strong or secure password |  |  |  |  | ✓ |
| 3 | Educational Resources Display   * the educational resource links are displayed and cycled appropriately * clicking on the educational resource links opens the respective resources |  |  |  | ✓ |  |
| Tester Comments | | - | | | | |

Table 5.3: 2nd UAT

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tester Name: Chan Khai Chung | | Date: 23rd April 2024 | | | | |
| Criteria | | Please tick the box for a rating of 1 to 5 (✓) | | | | |
| 1 | 2 | 3 | 4 | 5 |
| 1 | User Interface Validation   * the user interface elements are properly displayed and positioned * design is pleasing to the users |  |  | ✓ |  |  |
| 2 | Password Strength Feedback   * the tool provides accurate feedback on password strength * weak password such as "password" will indicate as a weak or insecure password * strong password such as "Str0ngP@ssw0rd!" will indicate as a strong or secure password |  |  |  |  | ✓ |
| 3 | Educational Resources Display   * the educational resource links are displayed and cycled appropriately * clicking on the educational resource links opens the respective resources |  |  |  | ✓ |  |
| Tester Comments | | The UI is simple. | | | | |

Table 5.2: 3rd UAT

## 5.4 System Testing and Discussion

I had the password strength assessment tool went through rigorous testing using a comprehensive unit testing plan and a user acceptance testing plan. I will be discussing about the execution and results of these test plans, while highlighting the key findings and outcomes.

The first technique, unit testing plan was executed to check on the correctness of each individual components and methods within the system. Overall, the unit testing plan yielded satisfactory results. Most of the test cases passed successfully, indicating that the core functionality of the system was implemented correctly.

However, it is unfortunately that a minor issue was spotted during the unit testing process. It was in the initial of the testing phase, the implementation of the ‘password’ plugin was not able to handle invalid input scenarios correctly, which result in exceptions being thrown. But this issue was quickly resolved by adding proper error handling and validation checks.

The second technique, the user acceptance testing plan aimed to validate the application from the end-user's perspective, the test cover aspects such as user interface, password strength feedback, and educational resources. I was able to receive 3 results, and these test plan yielded positive results with most of the test cases passing with a full mark. The end-users found the user interface intuitive and visually appealing, and the password strength feedback was accurate and helpful in creating secure passwords.

## 5.6 Summary

After implementing both the unit testing plan and the user acceptance testing plan, I can understand that unit testing plan was focused on verifying the correctness of individual components and methods within the system. It covered various aspects of the password strength assessment functionality. The execution of the unit testing plan revealed a minor issue, but the issue was promptly addressed, and the same test case were updated and re-executed to ensure successful completion.

The user acceptance testing plan on the other hand was aimed to validate the application from the end-user's perspective, encompassing aspects such as user interface, password strength feedback, and educational resources. This user acceptance testing phase involved representative end-users, who provided valuable feedback on their experience with the application.

Overall, the execution of both the unit testing plan and the user acceptance testing plan played a crucial role in my project since it helps ensuring the quality and reliability of the password strength assessment tool. The identified issues were deal with, and the necessary improvements were implemented, resulting in a more robust and user-friendly application.

# CHAPTER 6: CONCLUSION

## 6.1 Critical Evaluation

The Advanced Password Strength Assessment Tool project has successfully achieved its primary objective of developing a user-friendly application for individuals to create and maintain robust passwords. It is the implementation of advanced algorithms and using innovative approaches, that this tool can finally provide accurate evaluations of password strength, which will definitely help in addressing a critical aspect of cybersecurity in this cyber age.

I can safely say that one of the project's most notable strengths lies in its comprehensive approach to password assessment. Unlike many existing online solutions or tools that only rely on fundamental length and complexity checks, this tool I employ have sophisticated pattern matching techniques and entropy calculations to identify a wide array of predictable password patterns, making sure of a thorough evaluation of password strength.

Furthermore, the project's emphasis is also a part on heighten user education. By integrating online educational resources and best practices within the tool, users are encouraged to actively learn and adopt secure password habits. Not only does this approach enhance the immediate impact of the tool but it also cultivates a long-term culture of cybersecurity awareness among users of all ages.

## 6.2 Limitation

While this current project of mine has achieved significant milestones, it is still essential to acknowledge its limitations. I found out the one constraint lies in the scope of the target audience, as the tool I designed is primarily utilized by individual users rather than the whole enterprises or organizations. Consequently, advanced features such as account login, centralized credential management or group policy enforcement are not incorporated, limiting its applicability in larger-scale deployments.

I also located another potential limitation revolves around the computational demands of the algorithms inside the advanced password strength assessment overtime. In theory, if the tool gains widespread adoption and is used too many times, the computational overhead associated with processing vast quantities of passwords may become a bottleneck, potentially affecting the system’s performance and scalability.

## 6.3 Recommendation

It is true that the Advanced Password Strength Assessment Tool has immense potential for further enhancement and expansion in future iterations. One key recommendation I obtained is to adapt the tool's functionality to cater to the unique needs of enterprises and organizations, broadening its applicability beyond individual users. This could involve incorporating features such as login, centralized credential management, role-based access controls, and even integration with existing identity and access management systems. By catering to enterprise-level requirements, the tool can become a comprehensive solution for password security across diverse organizational contexts.

Another crucial recommendation is performance optimization. As the tool has been adopted to the world for a long time, the computational demands associated with processing vast quantities of passwords may become a choke point, potentially affecting the system’s performance and scalability. To address this, continually optimizing the computational efficiency of the password strength assessment algorithms is imperative. This could involve exploring parallel processing techniques, leveraging graphics processing units (GPUs), or implementing distributed computing architectures to handle large-scale password processing loads efficiently.

Incorporating contextual awareness into the password strength assessment process could also yield more tailored and relevant feedback. By considering factors such as the intended use case for example, creating financial accounts, social media, and corporate systems, the tool could provide customized recommendations for different users and diverse guidelines that aligned with industry-specific best practices, enhancing its value and relevance across diverse domains.

Furthermore, I can try to implement machine learning techniques to analyze password breach data and study evolving cybersecurity trends to enable the tool to continuously refine its algorithms and adapt to new emerging threats. This approach would ensure that the password strength level assessments remain relevant and effective in the ever-changing landscape of cybersecurity, making it the future-proof tool solution at the forefront of password security.

It is by addressing these recommendations, the Advanced Password Strength Assessment Tool can continue to evolve, maintaining its position as a cutting-edge solution in the realm of password security and contributing to the broader goal of fostering a secure and resilient digital infrastructure.

# REFERENCES

Alexander, Z. (2018, October 21). Is Microsoft’s “password checker” a reliable tool to test the

strength of your password?: Alexander’s blog. Alexander’s Blog | Sharing knowledge with the global IT community since November 1, 2004.

https://www.zubairalexander.com/blog/is-microsofts-password-checker-a-reliable-tool-to-test-the-strength-of-your-password/

Atlassian. (n.d.). Waterfall methodology for project management.

https://www.atlassian.com/agile/project-management/waterfall-methodology

BrowserStack. (n.d.). *What is a test plan: Importance, components, how to create test plan*.

<https://www.browserstack.com/test-management/features/test-run-management/what-is->

test-plan

California, S. E. U. of, Egelman, S., California, U. of, California, S. J. U. of, Jain, S., Rebecca S.

Portnoff University of California, Portnoff, R. S., Google, K. L., Liao, K., Google, Google, S. C., Consolvo, S., California, D. W. U. of, Wagner, D., University, A. S., University, G.-C., University, P., & Metrics, O. M. A. (2014, November 1). Are you ready to lock?: Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security. ACM Conferences.

<https://dl.acm.org/doi/abs/10.1145/2660267.2660273>

Groeneveld, R. (2022, October 17). The password problem. Nomios Group.

<https://www.nomios.com/news-blog/password-problem/>

Itexus. (2024, March 19). *Implementation phase in SDLC*.

https://itexus.com/glossary/implementation-phase-in-sdlc/

Kinsta. (2023, September 11). *What is jQuery? A look at the web’s most-used JavaScript library*.

https://kinsta.com/knowledgebase/what-is-jquery/

Kovalenko, O. (2023, February 21). *SDLC design phase: Definition, activities, goals*. IDAP

Blog. https://idapgroup.com/blog/sdlc-design-phase/

Literature review. The University of Edinburgh. (2020, August 23).

<https://www.ed.ac.uk/institute-academic-development/study-hub/learning->

resources/literature-review

Pandey, S. (2024, March 27). *What is Low Level Design?*. Code 360.

https://www.naukri.com/code360/library/what-is-low-level-design

Setter, M. (2023, October 20). *What is user acceptance testing (UAT): Meaning, definition*.

Usersnap Blog. https://usersnap.com/blog/user-acceptance-testing-right/

Sharma, A. (2024, February 19). *What is high level design*. PrepBytes Blog.

https://www.prepbytes.com/blog/system-design/what-is-high-level-design/

SmartBear. (n.d.). *What is unit testing?*

https://smartbear.com/learn/automated-testing/what-is-unit-testing/

The consequences of weak banking passwords. Australian Mutual Bank. (2023, June 6).

Retrieved September 16, 2023, from

<https://australianmutual.bank/news-blog/articles/consequences_of_weak_passwords/>

Ur, B., Kelley, P. G., Komanduri, S., Lee, J., Maass, M., Mazurek, M. L., Passaro, T., Shay, R.,

Vidas, T., Bauer, L., Christin, N., & Cranor, L. F. (n.d.). How does your password measure up? the effect of strength meters on password creation. USENIX.

Retrieved October 29, 2023, from

<https://www.usenix.org/conference/usenixsecurity12/technical-sessions/presentation/ur>

Wheeler, D. L. (2016, August 10). Zxcvbn: Low-Budget Password Strength Estimation.

USENIX.

Retrieved November 12, 2023, from

<https://www.usenix.org/conference/usenixsecurity16/technical-> sessions/presentation/wheeler

# APPENDIX

## PPF – Title Registration Proposal

A close-up of a proposal form

Description automatically generated

A document with text on it

Description automatically generated

A white rectangular frame with black lines

Description automatically generated

A document with text on it

Description automatically generated

A document with text on it

Description automatically generated

A close-up of a document

Description automatically generated

A document with text on it

Description automatically generated

A document with text on it

Description automatically generated

A close-up of a document

Description automatically generated

A document with text on it

Description automatically generated

A screenshot of a document

Description automatically generated

## Ethic Forms

A document with text on it

Description automatically generated

A close-up of a questionnaire

Description automatically generated

A close-up of a questionnaire

Description automatically generated

A close-up of a document

Description automatically generated

A document with a signature

Description automatically generated

## Log Sheets

A document with text and a list

Description automatically generated with medium confidence

## Poster

A screenshot of a web page

Description automatically generated

## Gantt Chart

A screenshot of a graph

Description automatically generated

A grid with blue squares

Description automatically generated

## Sample Code Implementation

A screen shot of a computer program

Description automatically generated

A screen shot of a computer program

Description automatically generated

A screen shot of a computer program

Description automatically generated

## Respondence Demographic Profile

|  |  |  |  |
| --- | --- | --- | --- |
|  | Gender | Age Group | Current Occupation |
| 1 | Male | 18 - 21 | Student |
| 2 | Male | 18 - 21 | Student |
| 3 | Male | 18 - 21 | Student |
| 4 | Female | 18 - 21 | Student |
| 5 | Male | 18 - 21 | Student |
| 6 | Male | 22 - 25 | Employed |
| 7 | Male | 18 - 21 | Student |
| 8 | Female | 18 - 21 | Student |
| 9 | Male | 18 - 21 | Student |
| 10 | Female | 18 - 21 | Student |
| 11 | Male | 22 - 25 | Unemployed |
| 12 | Male | 18 - 21 | Student |
| 13 | Female | 22 - 25 | Employed |
| 14 | Male | 18 - 21 | Student |
| 15 | Female | 22 - 25 | Unemployed |
| 16 | Male | 18 - 21 | Student |
| 17 | Male | 18 - 21 | Student |
| 18 | Male | 18 - 21 | Student |
| 19 | Male | 18 - 21 | Student |
| 20 | Male | 18 - 21 | Student |
| 21 | Male | 18 - 21 | Student |
| 22 | Male | 18 - 21 | Student |
| 23 | Male | 18 - 21 | Student |