SFU

CMPT 276 - Project Proposal

Deep Phishing

06 - Valleys

Philip Ho - 301472672

Amraj Koonar - 301559468

Kimchhorn Sambath - 301637407

Nathan Huynh - 301608957

Github repository - https://github.com/CMPT-276-SPRING-2025/final-project-06-valleys

Overview

Phishing scams pose a significant cybersecurity threat, deceiving individuals into disclosing sensitive information through fraudulent emails, websites, and messages. Seniors, minors, and those less familiar with technology are particularly vulnerable to these attacks. According to the FBI [1], “Elder fraud is an expensive crime. Scams targeting individuals aged 60 and older caused over $3.4 billion in losses in 2023—an increase of approximately 11% from the year prior. The average victim of elder fraud lost $33,915 due to these crimes in 2023.” One of our group members has witnessed this firsthand when his grandparents fell victim to a phishing scam, though they were fortunate not to suffer significant financial losses.

To combat this issue, This website could offer essential security tools and educational resources. Features would include a **URL, domain, and IP scanner** to detect potential red flags, along with **file scanning** that sends suspicious documents to multiple antivirus services for analysis. Additionally, the platform could provide an **AI-powered email analysis tool**, allowing users to upload emails for automated annotation of potential phishing indicators. A phishing simulation feature could let users generate and edit scam emails via an **npm package**, testing their awareness by sending them to trusted individuals. To reinforce learning, interactive **educational slides** covering topics like password security, phishing techniques, and scam detection strategies would be available. By offering practical security tools and educational materials, this website would empower individuals to recognize and prevent phishing attempts, fostering a safer online experience for all.

Our main target audiences are split between 2 groups of people. The first is the people who are aware of the consequences of being phished and they care about someone who may be not as tech literate as they need to be to know to not be a potential phishing victim. The other target is the person is a potential victim and to provide awareness of phishing directly to them and to give them practice and awareness to not get phished.

Persona

1. Persona: Jimmy John (an average internet user)

   Age: 20

   Job: University student

   Tech literacy: Moderate

   Goals: Plays online video games, uses online social media, and always checks emails for school.

   Pain points: Doesn't recognize phishing threats, clicks links without verifying. Easily tricked by fake giveaways, online friend request links, and online game invitation messages. Also, believes every school-related email without verifying.

   Behaviour: Due to inexperience online at his younger age, he frequently checks and downloads emails, clicking email links commonly. He trusts online strangers too easily and does not inspect emails that claim to be from his school.

2. Persona: Susan Oldei (a retired grandmother)

   Age: 70

   Job: Retired

   Tech literacy: Low to none

   Goals: Stay connected with family through FaceBook, and WhatsApp. Read emails from banks, pension providers, and online shopping sites. Occasionally shops online on basic and easy-to-use websites such as Amazon & eBay.

   Pain points: Doesn't recognize phishing threats, clicks links without verifying. Easily tricked by fake giveaways, online friend request links, and online game invitation messages. Also, believes every school-related email without verifying.

   Behaviour: No education or knowledge of scam emails that may pretend to be the bank, government, or online shopping websites. Frequently forgets passwords to important websites (bank, email). Is the target of being emotionally manipulated by scam emails ("your son is dying, send me money!") and may be easily tricked by tech support scam emails ("We need your banking password urgently!").

3. Persona: Asrap Miguel LIX (a high school computer science teacher)

   Age: 37

   Job: High school computer science teacher (Grades 10-12)

   Tech literacy: High, proficient

   Goals: Educates students about introductory programming, AI ethics, and responsible internet usage. Frequently manages emails via school admins, students emails, and parents emails. Use popular and free learning management systems such as Google Classroom, Github Classroom, Slack, and Microsoft Teams. Always up to date with new technologies and security threats.

   Pain points: Potentially malicious attachments disguised as student assignments. Social engineering scams pretending to be from the principal or IT support within school. Lack of practical teaching material to solidify student's knowledge.

   Behaviour: Encourages students to recognize phishing by teaching it in his classrooms. Uses our website as a resource to educate the younger generation, showing the dangers and effects by example emails generated by our website.

APIS and Features

1. Phishing (VirusTotal)
    a. Scan URL/Domain/IP for any potential red flags, and check hashing.
    b. Scan file to flag any potential red flags, and check hashing, sends to multiple antiviruses to scan the file to determine if its safe or not.
    c. Fetch/Post comments for a specific URL (done through virustotal db).
2. OpenAI
    a. Generate sample email/scam emails, edit them and send via npm package to someone you know to see if they are a potential victim.
    b. Upload email to AI, annotate things where it indicates it might be a scam, red flags or upload an email you have, it edits it to something with a log of red flags and you need to detect them.
    c. Interactive quiz where users have to detect potential phishing emails by providing emails with no red flags, and an email with red flags, the user has to determine which is which.

3. DataLab Surya/Marker
    a. Text detection in Japanese and getting a bounding box for those words making it clickable/highlightable.
    b. Layout Analysis and Element Detection: Surya can identify and analyze various document elements, such as tables, images, and headers (if this is not accepted, I can just add in convert text as Markdown).
    c. Extract text out of the manga panel following the correct reading order of the Japanese.
4. Deepseek
    a. Use the extracted text as the context so the user can ask for information in their language of choice.
    b. Generate a mnemonic from the selected Kanji to be used as a Flash Card for Learning Material.
    c. Create Furigana to be added for difficult Kanji word-aiding reading.

Refer to: Appendix #2

SFU

User Story

1. Generate sample email/scam emails, edit them and send via npm package to a close friend
    a. As someone who cares about my grandparents and them not being phished, I want to be able to generate a fake email and be able to send it to them so that they will gain awareness of what clicking these fake emails might do to them.
    b. Given some information about the grandparents and what they do in a daily life such as play golf, generate a sample email that they may fall for when a user is not focused and just simply reads the base email then clicks on the email because of their lack of awareness of potential phishing scams.
2. Interactive Training for Spotting red flags
    a. As a user who needs to learn about potential phishing scams I want to get the experience of them first hand so that I can see the red flags myself and not fall for these potential phishing attempts.
    b. Given an opportunity to see these red flags myself I want to be able to determine whether the email is fake or not so when the time comes I can correctly be able to not fall for a potential phishing scam. Then I will be able to not fall for these and be educated on phishing and be able to help others not get phished or scammed.
3. Scan file to flag any potential red flags, and check hashing, sends to multiple antiviruses to scan the file to determine if its safe or not
    a. I want the feature to scan file attachments for potential threats by checking for red flags, verifying file integrity through hashing, and using multiple antivirus APIs, So that I can ensure that the file is safe before opening and avoid security risks because a scam email usually attracts with the virus file which can attack my device system to steal information.
4. Scan URL/Domain/IP for any potential red flags, and check hashing.
    a. When I or my grandparent get the email that contains a url, I will want a feature that can scan and the url whether it is safe to open or not, so that I can avoid the risks associated with malicious links
5. Fetch/Post comments for a specific URL (done through virustotal db),
    a. This feature is used when I suspect a URL's integrity even though I use the url scan feature. I want to retrieve replies to a specific comment about a dangerous.. URL so that I can see discussions and insights from others who have encountered the scam email containing the malicious link. I also can post the comment on the specific url that I am sure that the URLis dangerous to the virustotal db, so whoever gets this url will know it is a malicious link.
6. A scam email detection and annotation tool
    a. As a user wanting to identify phishing or scam emails, I want to upload an email and use an AI feature that annotates or highlights suspicious elements. So that I can easily spot red flags and learn how to identify potential threats in the email text.

7. Text detection in Japanese and getting a bounding box for those words making it clickable/highlightable.
    a. As a reader wanting to learn Japanese through immersion, I want to be able to click and highlight Japanese text in a manga panel just like how I would on a regular website so that I can selectively translate only the parts that I need, instead of translating the entire page at once.
8. Layout analysis and element detection, identify and analyze various document elements, such as tables, images, and headers.
    a. While a website or magazine has a horizontal way of writing, that isn't always the case with manga panels as it can change depending on how the author wants to draw the dialogue bubble. I want a central solution that works for both ways of writing while also preserving the correct reading order.
9. A direct conversion from text to beautifully formatted Markdown
    a. Not just manga panels, novels and other documents are also a great way to do immersion and learn Japanese in a quick and effective way. I want to directly convert detected text from those documents into a well-structured Markdown, preserving the readability and proper text hierarchy when saving or sharing the extracted content.
10. Use the text as the context so the user can ask for information in their language of choice.
    a. As a someone who many not understand the context of learning a new language, I want to be able to select a certain word in a phrase or a sentence and understand it in a way I can actually fully comprehend the meaning of their words, so that I can become better at the language I want to learn by understanding it fully.
    b. Given some text input and when I click or highlight a certain part of the sentence, I want to be able to understand it so then I can improve my comprehension of the language I am learning.
11. Generate a mnemonic from the selected Kanji to be used as a Flash Card for Learning.
    a. As someone who needs to learn a new language, especially with something complicated like Kanji, I want to be able to generate ways to better remember the kanji by creating mnemonics and flash cards. So that I can create a fun mnemonic to help associate the kanji with something that I can remember.
    b. Given a certain Kanji and a description I want to remember the Kanji with, so when I need to remember the Kanji I can use my unique mnemonic to then remember the certain Kanji making me, so I can add a Kanji to the list of Kanji's that I remember.
12. Create Furigana to be added for difficult Kanji word-aiding reading.
    a. As someone currently reading a line of text, but I don't currently understand every Kanji I want to be able to understand what the Kanji is, but in a simpler form

(furigana) because I still want to be reading the text in Japanese, so that my Japanese reading comprehension skills will improve.

b. Given a Kanji that I may not fully understand, when I see the Kanji that may be difficult for my level of understanding, then I will be able to see the Furigana above the Kanji so that I can understand the Kanji.

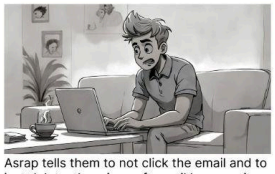
User Storyboard

Asrap sits on a sofa, leaning back enjoying his day and preparing his materials for his next class while drinking coffee with his laptop on his table.

Asrap opens his laptop and is surprised to see a video call request from his grandparents and decides to accept their video call.

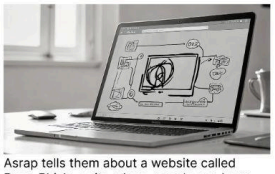
Asrap's grandparents say they recently got an email from Microsoft saying that their windows needs to be updated!

"**ATTENTION** YOUR PC IS OUTDATED!!! PLEASE UPDATE YOUR PC NOW AT www.official-microsoft-update.com"



Asrap tells them to not click the email and to just delete the microsoft email because it seems like a phishing scam and tells them about it.

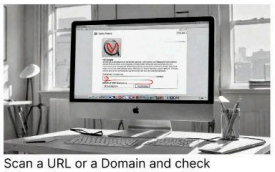
Asrap's grandparents tells Asrap, that they wish they knew more about these to learn more about phishing to not get scammed.



Asrap tells them about a website called Deep Phish, a site where people can learn more, practice and try out common phishing tactics to not get scammed.

Email Annotation Feature to mark where potential red flags are

Upload email feature to scan potential red flags



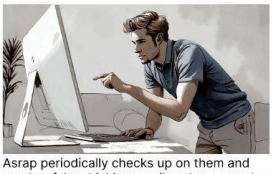
Scan a URL or a Domain and check information, such as HTTP, red flags, javascript injections

Using AI to generate sample emails and compare to real emails

Upload a file and scan it, to see if there is anything wrong with the file, red flags, bad hashing

Interactive Training for Spotting red flags, provide a user with a potential phishing email and a "normal" email

Asrap's grandparents use this site to learn more about phishing and scams and gain practice about prevention of phishing.



Asrap periodically checks up on them and sends a fake phishing email to them to make sure they aren't clicking everything sent to their inbox.

Asrap's grandparents receive the fake phishing email and laughs at it now because she knows this may potentially be a phishing email and doesn't click the email.

[Link For Figma]

SFU

Front-end Technology Stack

We will be using **Next.js**, a full-stack framework for React.js. We are aware that most of the site will be client-sided with the front-end, but we prefer the built-in routing and api routes that we plan to use for the project. The APIs we plan to use will likely need an API route instead of a simple fetch api call and Next.js provides built-in support to do so. We chose this stack because every group member is familiar with React. Additionally, this stack fits best with our project idea and implementation plans. Next.js is the best option because Vite and create-react-app are frameworks that are built towards single page applications which don't fit with our project. Some members may not be completely familiar with Next.js, but they are aware of using React, and have stated they are more than willing to learn if something comes up.

Furthermore, for styling we plan to use Shadcn/ui along with Tailwind CSS, giving us complete freedom over the css. Shadcn provides us with a component library that is easily integrated with Tailwind CSS and can streamline our front-end code to make it consistent.

Appendix

1. Federal Bureau of Investigation, "Elder Fraud in Focus," FBI, Aug. 24, 2023. [Online]. Available: https://www.fbi.gov/news/stories/elder-fraud-in-focus.

2.