**Overview**

I developed an interactive game that involves exploring an individual's message history to construct a narrative. I chose this medium because I feel like it allows me to challenge my learning form the document object model. Due to the sheer complexity of interactions required, my primary focus during development was on ensuring that users were safeguarded against unintentional inputs.

As the project progressed, I recognized the scope of the project was broader than I anticipated. Thus, I chose to pare it down and concentrate exclusively on perfecting the interactive chatbot component. Rather than creating a custom user interface, I opted to leverage an existing template and build in all my desired features into the template. This approach enabled me to optimize my time and prioritize the JavaScript component, which was integral to expressing my computational thinking.

**Choice of project**

Interactive games that allow users to explore and interact with their environment have always captivated me. One such game that caught my attention was 'Simulacrum' by Kaigan in 2017, where players can delve into messages, emails, and photo galleries to construct a narrative. I found the layers of interactivity in such games intriguing and wanted to explore them by applying my skills in the Document Object Model.

However, I was aware that the scale of this project was vast, and since this was my first time attempting such a project, I decided to focus on developing a messenger app for my game. Although I had initially planned to include a maze to broaden the scope of the project, I soon realized that creating even a basic level of interactivity required a considerable amount of coding, and so removed that from the final concept.

The plot of my project is not revolutionary, as crafting an engaging narrative with multiple outcomes is a challenging task. Therefore, I chose to create a small scale, intimate story about an individual who used to be close friends with someone but eventually drifted apart. There’s no motivation behind me picking this topic, apart from the fact that I remembered writing a similar story during my exams in grade 10.

**Challenges and Computational thinking process**

Although creating tabs that could switch the content on the screen seemed straightforward at first, I encountered difficulties translating it into code, despite consulting a source from w3schools. To achieve this, I assigned the same class to all three contacts but assigned different ids. When a button was clicked, the event listener would execute a function that set the display of all three contacts to none, except for the clicked contact, which would be set to flex so that only the content from that contact would be visible.

In another section of the game, the user's messages were "encrypted" (in reality, just translated to base64). I wanted the available options to change based on whether the user had decrypted the phone or not. To accomplish this, I wrote code to check the page every second to see if the condition "decrypted" was set to true. When the phone was decrypted, the game would add the next set of event listeners to change the behaviour of the game. I struggled with this aspect of the project because I was not familiar with the set Timeout function.

I also faced a challenge with the dialogue. Initially, my plan was to hardcode a function for every response, but that proved unrealistic. To simplify the code and generalize it, I created an array that included all possible responses and follow-up options.

**Challenges and Lessons learnt**

* DOM manipulation:
  + querySelector to manipulate the DOM
  + Dynamically appending Div elements
  + Modifying the display style of elements
* JavaScript features:
  + setTimeout function
  + setInterval to update the page
  + Implementing base64 encoding in JavaScript
  + Creating conditional event listeners

I also learnt to plan out my game better. Week 10 was wasted because I didn’t get much progress done. When I finally started working on the project, there were multiple instances where I sat down, and even though I knew what the final product was supposed to look like, I was clueless about what to start with. After my tutorial on week 12, I sat down and listed out all the features I wanted to have. After doing so, I worked on each feature 1 by 1, until the game finally got to a stage which I was content with.

**Rip/mash/modify sources.**

Source for messaging UI: <https://codepen.io/mubangadv/pen/rXrOQa>

source for switching tabs: <https://www.w3schools.com/howto/howto_js_tabs.asp>

Initial plan was to create my own encryption book code: <https://www.youtube.com/watch?v=I2O7blSSzpI&t=1s>

I could not figure out to do it computationally, so I opted to simplify by simply converting the text to base64. (I understand this is not real encryption. I just wanted to offer a funny puzzle)

**Self-critique**

While I am actually content with how the final product came out, I will admit that the length of the game is significantly shorter than what I was expecting to create

I do not think my implementation of checking the encryption status is optimal. Ideally, instead of checking it every second, it would have been better to check the encryption status anytime the user changed tabs.

I encountered a few issues with the padding on the options in Jake’s dialogue. I could not figure out a consistent way to dynamically alter the padding depending on the size of the option.

The range of options are quite small, and the possible story paths are quite linear.

**Extensions**

I would love to implement an undo button which would let the user fall back on a certain option. A possible way to do this would be to use a counter which keeps track of the dialogue progress and reverts back to the previous counter.

I would also love to add a separate email section so that the user can explore even more options when looking through the “phone”.