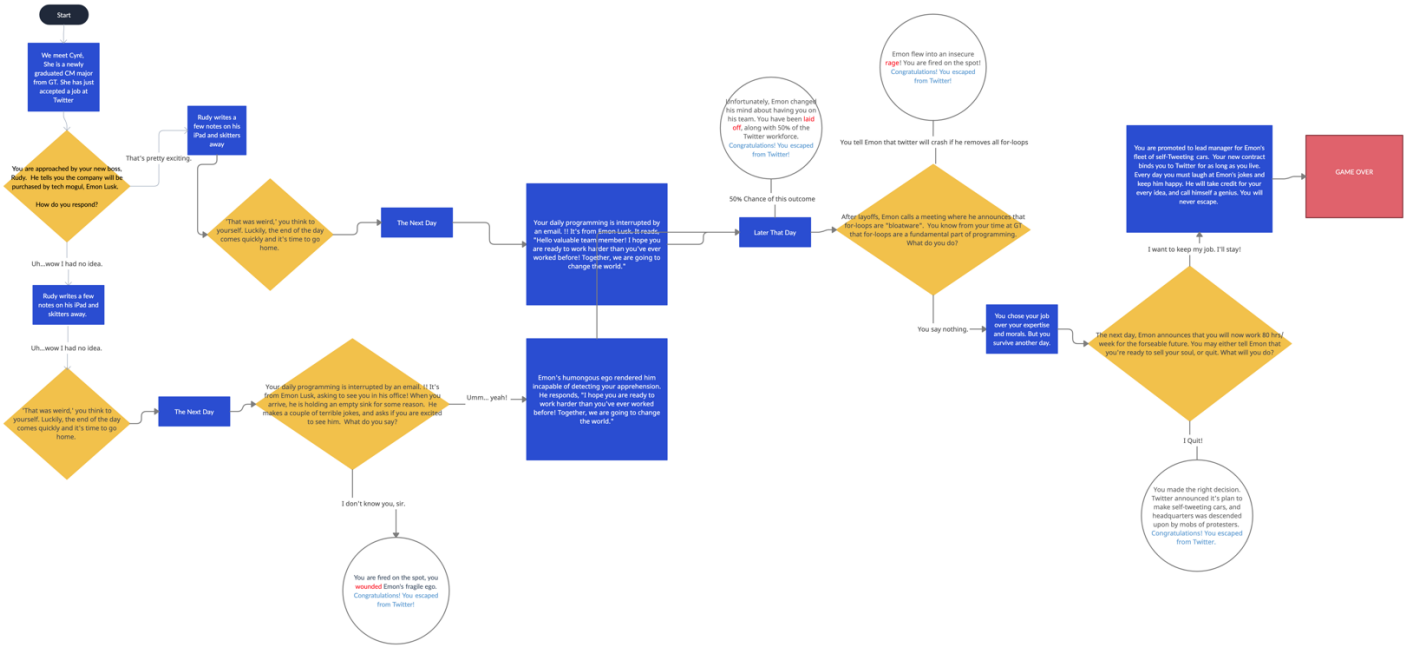
Escape From Twitter

LMC 2700
Final Game Prototype
Game: Escape From Twitter
Group:

Angelica Jones: Group coordinator and software engineer, design lead, HTML/ CSS
Alexandr Shor: Software engineer, HTML/ CSS, Javascript implementation
Nikkolas Glover: Story writer, Twine engineer, software support, HTML/ CSS

Story Development

The group began this project with an idea for a story-based game. We derived this story based on current events, and turned the project into an office-based drama. The first step was creating a layout detailing story structure, characters and game decisions that would propel the player through the game. The initial story skeleton is attached below.



Twine

Based on the story outline, and assessment of the skill levels of teammates and desired outcome for the final product, team members decided to use Twine in combination with Snowman 2.0.2 to implement the game. Snowman provides a unique customizability to the game for people experienced with HTML, CSS, and Javascript. We decided to challenge ourselves to customize the style and the functionality of the game using experiences we gained during the semester with HTML, CSS, and Javascript. We then, created the story layout above within the Twine software. Some small changes had to be made to simplify the story, and we later tweaked the story even more to be compatible with Javascript, which will be discussed further in a later section. After the team was happy with the Twine layout, we discussed stylistic changes we wanted to make.

HTML/CSS

The team used HTML and CSS elements to format the unique design of the game passages. Since the game is designed around a workplace drama set at Twitter headquarters, the story is told through tweets, which were developed using a tweet generator. The generated imaged from the tweet generator were then loaded into each story page using Javascript interaction with HTML DOM elements. The existing text from the Twine file was first finalized so that final versions of tweets could be loaded into each page of the story. CSS also uses link formatting to create specific styling for story headers and links to new passages.

Javascript

The Javascript implemented for this game brings a unique sophistication to story outcomes. A Karma score object is created upon game loading. This karma score is then altered depending on the users positive or negative interactions with other characters, such as the games main antagonist, Emon Lusk. This is where further story editing came into play. Some story elements of the game were simplified to lead to desired outcomes such as making sure each player has a full gaming experience, without accidentally losing too early. The implementation of the Karma score helps keep track of the player's gameplay as they go along, and helps to impact outcomes later in the story. In addition, the Javascript implemented makes use of objects from the Random class, which create variance of interactions within the game! For example, some Javascript functions generate either a random positive response from the antagonist, or a random negative response. We have coded several options for these responses within the code.

Future Avenues

While the game implementation came together for our group in the ways we originally envisioned, there is still a lot of room for growth. One area of improvement remains in the style elements. Even though the tweet generator proved a useful tool for styling the game in a comical yet appropriate way, we feel that the game could develop even more of a clean look, with the inclusion of a toolbar or menu option, even though the game is relatively easy to navigate. We also feel that we could increase the interactivity of the game! For example, the game follows a new software engineer, so we could further gamify this aspect by prompting the user for console

input, effectively turning the game into a coding practice tool as well! The game would ask simple coding related questions. Then, getting questions about Javascript of HTML correct would contribute to the Karma score of players. This would bridge the overall theme of the game, and pay tribute to the programmers behind the project, and audience members who want to practice coding, or just have some fun with their existing knowledge.