I. The Impact of Project on Learning

As mentioned in the applying skills section, I learned a new standard and procedure of selecting information in a research. As the information I looked for was testable and was used as steps of my creation instead of quotes to insert in writings, the credibility of the author and platform providing the information became less significant than I would consider if conducting research for written works of science or social studies, meaning I could include information from more sources with lower depths on my list. In terms of planned research preceding actual creation, this decision increases the amount of time for organization and reduces the efficiency of the testing stage. However, also as mentioned in the applying skills section, the research for programming was dynamic due to uncertainties during the creation of the product, and no beforehand research could be guaranteed used or not used in the product. For the same reason, I had encountered insufficient research in past creative assignments before, where I ended up abandoning plans and searching for guides online arbitrarily with poor organization. Nonetheless, I learned, from the personal project, to modify instead of simply ignoring the schedule.

I kept the research divided up into stages, such as 2D movement testing, 3D movement testing, and rendering, but I combined the time block for the research and product creation. Moreover, I allowed changes in the order of the blocks. For example, I could research on and test for 2D movement before working on rendering, or if I found the process of rendering included the foundation of 2D movements, I could work on scripts for rendering first. These two changes gave more flexibility and reduced chances of redundant preparations, and as I still followed a schedule, despite its seeming disorder, I made steady progress and finished my product creation before the deadline. This experience impacted my attitude towards research plans, which I was obliged to make in past social studies assignments but did not know how to handle without the confidence to strictly follow the plan. Now having learned the way to construct an action plan for dynamic research, I will be able to better organize my timelines, from static research for an essay, dynamic preparation for an art design, to study for an upcoming exam, reducing the constant time pressure I had been facing and improve the quality of my works and learning.

II. Product Evaluation

My product includes the files of the video game and a document explaining the codes without using coding languages.

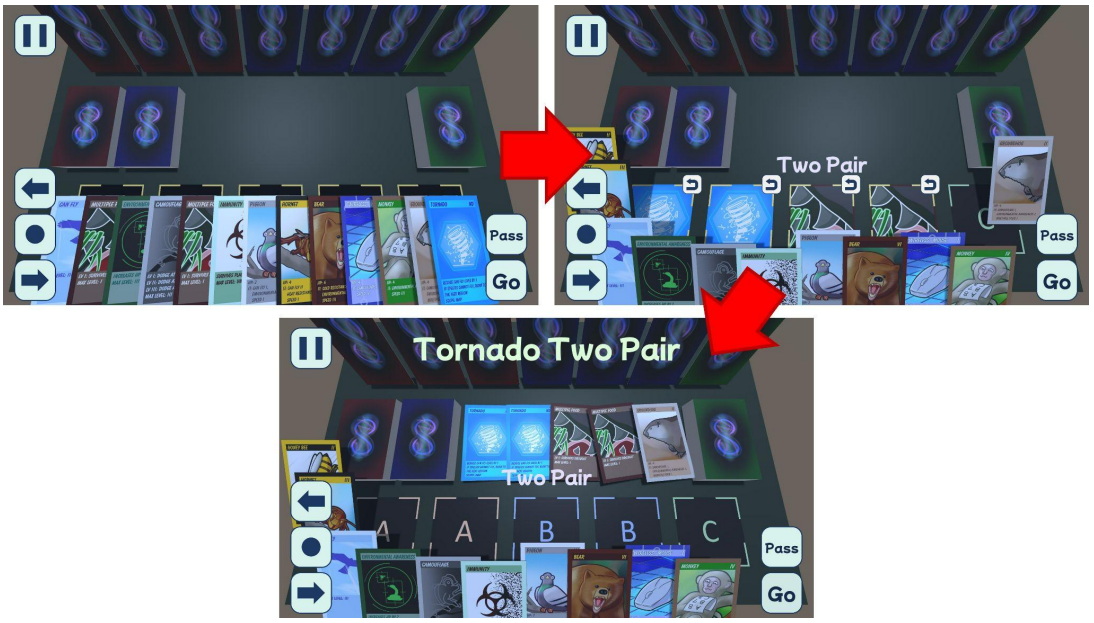
Upon running the execution file and loading the scene, the game automatically draws 13 cards as the opening. Clicking on one card makes the card follow the cursor, and clicking again drops the card. If dropping the card on a slot where the card is legal to be placed, the card would be inserted into the slot, and a button with a return arrow appears on the top right corner of the slot, clicking which returns all cards inserted in the slot. Clicking on one of the 3 card decks, *Species* (blue), *Trait* (red), and *Effect* (green), draws 1 card in the corresponding category.

The top left button pauses the game.

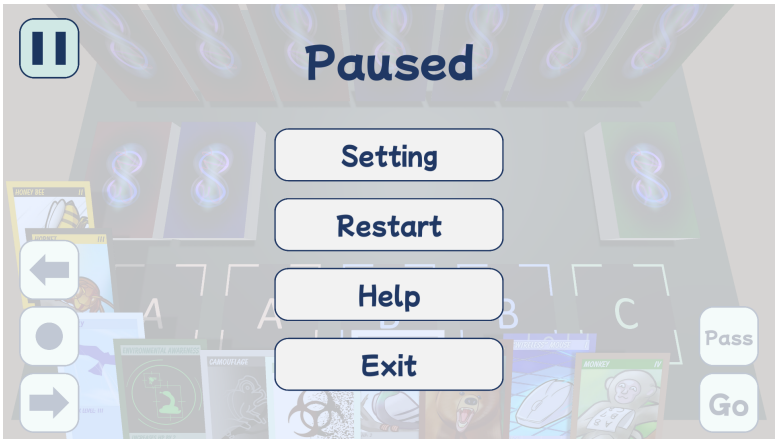
The bottom left buttons changes the hand type, including *four of a kind*, *full house*, *pair*, etc., like in poker; the left arrow changes forward, or to the next bigger hand, and the right arrow changes the opposite way; the circle opens a menu of hands for the player to select by name.

The bottom right button reading *Go* plays the cards in the slots if the cards in slots match the hand type and the hand is legal to play based on the previous hand on the table. The bottom right button reading *Pass* clears the table, and the previous hand on the table becomes empty. Whether a hand can be played after another hand is determined based on the game rules, available from the game.

The following screenshots show views at different steps in the game, starting with the automatic draw of 13 cards, then placing cards into slots according to the selected hand type, in this case *Two Pair*, and then playing the hand, in this case *Tornado Two Pair*.



The following image shows the paused view in the game. *Setting* directs to a page to change the maximum number of cards the player can hold (the default is 16); *Restart* resets and shuffles the card deck and clears the table, but does not make the automatic draw of 13 cards; *Help* directs to a Google Document with the game rules; *Exit* turns off the game and closes the window.



The document explaining the codes, as shown an excerpt in the applying skills section, covers firstly the main elements I created in the scripts, including the self-defined class EVO and the EVO dictionaries, secondly the main checks during the game, including whether a card can be placed in a slot and whether a hand can be played, thirdly the tree diagram determining whether a hand can be played, and finally the life cycle of a card object.

I conducted a survey within school to collect feedback after sharing the game file and the explaining document. The following table is the evidence of my survey, an excerpt of the spreadsheet recording my survey results for the evaluation with success criteria. Each scoring has a range from 1 to 8. *Previous Knowledge of the Game* was for identifying biases in the results; *Game UI Understanding* and *Game Rule Understanding* were used for success criterion 1; *Art Design Identification* and *Art Design Connection* were used for success criterion 2; *Animation Smoothness* was used for success criterion 3; *Code Explanation Clarity* was used for success criterion 4; success criterion 5 was evaluated independently from the survey.

Response Source	Previous Knowledge of the Game	Game UI Understanding	Game Rule Understanding	Animation Smoothness	Art Design Identification	Art Design Connection	Code Explanation Clarity
S11352	Yes	8	8	8	8	7	7
S11262	No	8	8	8	8	8	8
S11419	No	8	8	8	8	8	8
S11229	No	8	7	8	8	7	7
S11293	No	8	7	8	8	8	8
S11349	Yes	8	8	7	8	8	6

The following table lists evaluations based on the survey above and the success criteria mentioned in the planning section of the report, referencing the statements. The detailed requirements and reasons behind can be referred to from the success criteria part in the planning section.

Index	Statement	Succeed	Reason	Actions I did/should do that lead to the success
1	The game rule and user interface can be easily interpreted by players in the expected way.	Yes	Leaving out ones with previous knowledge of the game, the majority and rounded average of respondents' scores for game UI and game rule understanding are 8 out of 8, and the in-person verification afterwards proved their understanding correct.	I introduced the game rules in detail, covering every possible scenario before questions came up; I used clear, common symbols for the buttons, and thus, the respondents could use the interface intuitively.
2	The art design is clear, and the meanings can be easily interpreted by players in the expected way.	Yes	The majority and rounded average of respondents' scores for art design identification and connection are 8 out of 8. Art design identification means if the respondent can tell what is in each picture, and connection means if the respondent can relate the picture to the name of the card, especially for the *Traits*.	I used animals known with specific traits as symbols to convey the abstract traits of organisms. I referred to online sources and several photographs I had taken to draw clear pictures with accurate structures, such as *Human* (*Effect*) the bird in *Can Fly* (*Trait*).
3	The display includes smooth 2D and 3D animations.	Yes	Drawing and moving cards both involved 2D and 3D animations; the majority and the rounded average of the respondents' scores for animation smoothness are 8 out of 8.	The research on the InvokeRepeating function in Unity and choice of simple translations and rotations prevented lags and resulted in smooth animations.
4	The student fully understands and can explain object-oriented programming (OOP) and cross file programming in C#, Unity, and other algorithms.	Yes	The majority of the respondents' scores for code explanation clarity is 8 out of 8, and the rounded average is 7. Success in teaching or explaining concepts to a person without the base knowledge is the best evidence of understanding.	I fully understood the logic in the game rules and made sufficient corresponding research. I also connected the C# coding with past experiences on Sololearn introduced in computer programming class last year.
5	The game, from visual displays to input/output process, functions in the expected way.	Yes	There are no distorted or broken textures in the game; the UI operations functioned as introduced above; all features of the game rules were tested and proved possible to occur, in the	I did thorough research for rendering and wrote complex yet complete logic loops in the scripts. I minimized unnecessary variables, functions, and operations to

			correct way.	avoid lagging from limited resources.