Toast for Tory the Tortoise: A game of making toast according to certain criteria to be judged by a tortoise

Software Design Final Project Plan
Yuping Huang, Tegan Wilson, and Carolyn Ryan

Goal and Objectives

Our project for the final will be to create a game in which the user makes toast with various toppings (such as butter, avocado, sriracha, etc.) and then is judged based on the quality (or similarity to the criteria) given by the game. The user will be given an image of the ideal toast and description of the toppings by a tortoise named Tory. Tory will then judge the user's toast to see if it meets their toast standards. Depending on the quality of the user's toast, Tory will react to the tastiness or toxicity of the given toast. The toast will be very customizable with the ability to toast to varying degrees and to add toppings of varying amount. The user will not be able to remove toppings once they are on the toast.

The game will be level based and as it continues, the toast criteria will become more vague and harder to achieve the perfect piece of toast. The user can disappoint Tory 3 times, before they lose their toasting privilege. However, as levels get more difficult, toxic ingredients will be added to the selection of toppings and if Tory eats anything toxic, they die and the user loses the game. If the user passes all of the levels, they will win the game.

The goal of our project is to create a fun and simplistic game to entertain our users and provide good humor. It could also provide as a way to help users enhance their response to criteria, love for animals, toast making abilities, and problem solving under pressure.

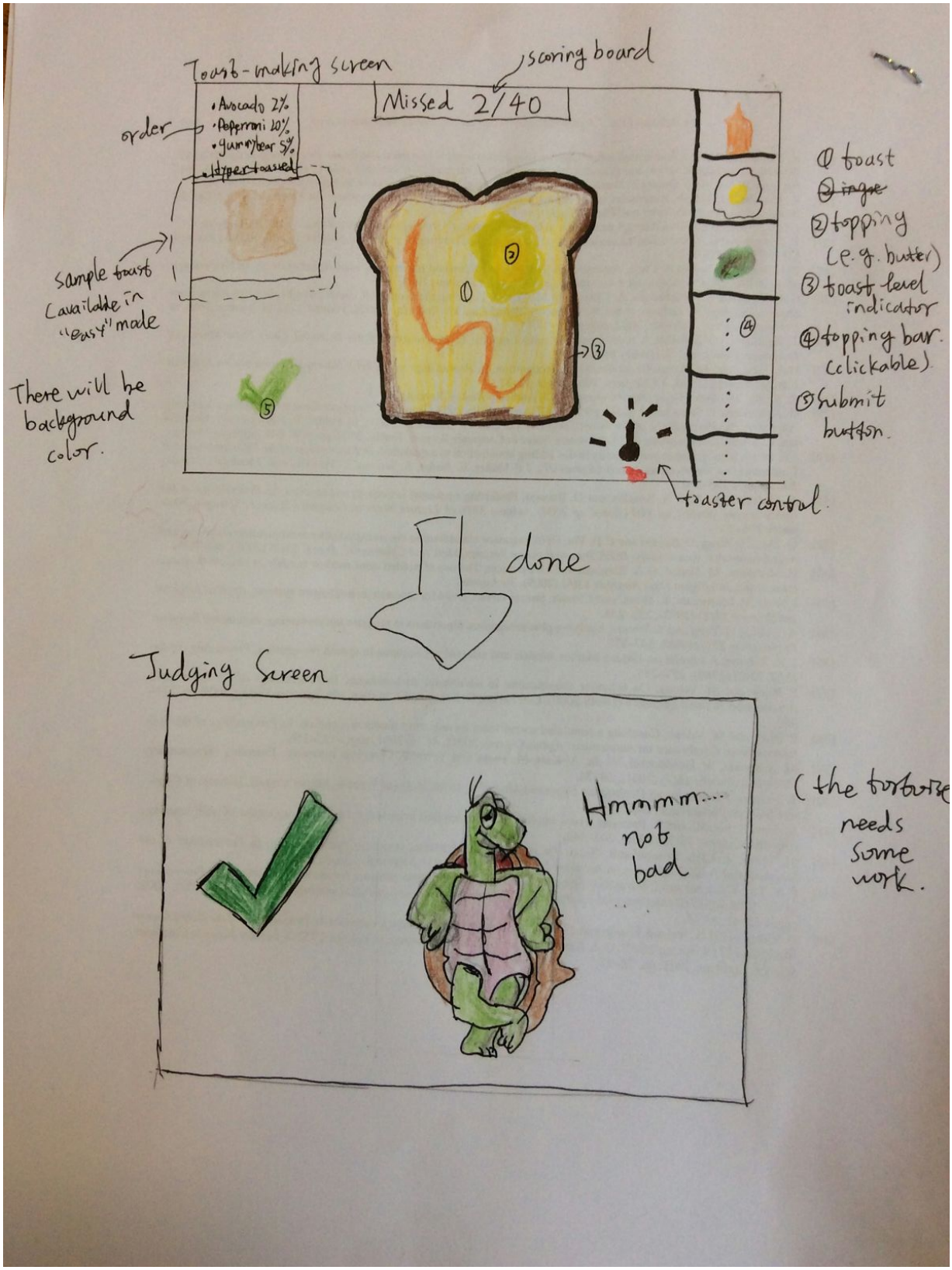
Implementation

- Graphic User Interface
- Toast
    - One instance with the user's current toast
    - One instance with Tory's criteria
- Tory
    - To judge and return varying results
- Toppings
    - Have name, color/graphic characteristics, and amount
- Criteria
    - Randomly generate Tory's toast criteria
    - Make the criteria harder to reach as the levels go up

Design Pattern

We are planning on using the MVC design pattern and a Strategy Pattern to help us deal with the changing level difficulties and judging. MVC will be our general structure and underlying structure of our game.

Sketches


Toast-making screen
scoring board
Missed 2/40
order
• Avocado 2%
• Pepperoni 10%
• yummybear 5%
• ~~Hypertoasted~~
sample toast
Available in "easy" mode
There will be background color.
① toast
② topping (e.g. butter)
③ toast level indicator
④ topping bar (clickable)
⑤ submit button
toaster control
done
Judging Screen
Hmmmm... not bad
(the tortoise needs some work.

Roles/Tasks:
Yuping: View (Graphic Design/GUI)
Tegan: Model (The underlying implementations)
Carolyn: Controller (Communication between GUI and implementations)

We will all be responsible for ensuring functionality and completeness of our various responsibilities. However, this is super vague as all three parts will interact and communicate with one another. As such, we will pair program when we can and when we cannot, we will clearly communicate what our code does, how it functions, and what it requires. All together we will decide and discuss interfaces among parts of the game. Besides, we will discuss requirements for game functions, expectations on what the game will do, tests of the functionality of the game. The roles assigned here show who the team lead for the various stages of the project will be, more than who will be in charge of the development for each project part.