AP Computer Science Final Project - README Template

Instructions:
The first step in creating an excellent APCS final project is to write up a README. At this stage, this README file acts as your **project proposal**. Once you've filled in all components, Shelby will read through it and suggest edits. Ultimately, you need a document that adequately describes your project idea and **we must agree on this plan**.

Have one member of your group **make a copy of this Google Doc**. Then, they should share it with all other members **and with Mr. Shelby** so that every group member has edit permissions, and Shelby can add comments on your ideas.

There's a lot of parts of this document that you might not have full answers for yet. Because you haven't written the program yet, it's difficult to think about the **instructions** or **which group members will do which parts**. Even though this is hard to think about, you must have something in these sections that acts as your current plan. However, during the course of the project, you'll **continuously update this document**. This means that you will not be *held* to exactly what you put here - components of this document can change (and it's pretty common!).

There is one exception: the **Features List** section. Once Shelby OKs your README, the Features List section **cannot be modified**. For this reason, it is most important that you get a solid idea of what you want to make and the primary features it will have *now*.

Talk with your group. Consider drawing some pictures of what you think your project might look like. Be precise. When you're ready, fill this out together. Each component in brackets below ([these things]) should be replaced with your ideas. Note that there are several sample READMEs posted on this assignment for you to use as guidance.

----------When README is finalized, remove everything above this line----------

Devil's Diner

Authors: Iris Chou, Sohum Phadke, Chloe Poon
Revision: May 13, 2022

Introduction:
Welcome to Devil's Diner, a cooking game set in post-apocalyptic times where the player has three lives. The user manages a person moving around a kitchen with a top-down view of the room using arrow keys. The person's objective is to complete orders that appear on the top of the screen periodically. To complete an order, the user has to put all the ingredients into a hole in the ground. The order in which ingredients are put into the hole doesn't matter (for example, the tomato can be

put in the hole before the bread and cheese). However, if the order is incorrect, the order won't go away, even if the food is put into the hole. If the order shelf fills up with orders (5 or more), the player loses a life, and the order shelf gets cleared. The ingredients are themed with the post-apocalyptic theme of the game such as radioactive onions and roadkill rabbit. Occasional disasters take place, requiring the player to complete certain actions in order to prevent or fix the problem. The user can address the flood by selecting the sandbags with the enter/return key, and the fire by using the fire extinguisher with the enter/return key. For a power outage, the user can select the light switch using the enter/return key. If the user does not properly address the disaster, it will cause them to lose a life. The player can gain more lives by finishing 10 orders, with a maximum of 3 lives at a time.

Instructions:

To move around, the user uses the up, down, left, and right arrow keys. In order to pick up and drop off food, the user can press the spacebar. The user can only pick up ingredients and throw food into the hole if they are at most 100 pixels away from the center of the hole. The user presses the spacebar to drop food into the hole in the ground. To put out the fire, the user must pick up the fire extinguisher with the enter/return key. For sandbags, the user has to pick up the sandbags by holding down the enter/return key. In the case that there is a power outage, the user needs to go to the light switch and press the enter/return key to turn the light back on. If any of the disasters isn't resolved in 10 seconds, the user loses a life and the disaster is resolved and orders are cleared.

Features List (THE ONLY SECTION THAT CANNOT CHANGE LATER):

Must-have Features:

- Floods: The user gets a flood notice and they must put down one sandbag at the door in 10 seconds. If they don't the screen turns blue and the user loses a life or the game ends. To put down sandbags, the user must press the enter key to pick them up, hold onto the enter key as they move to the door, then release the enter key to drop the sandbags
- Cooking: The user uses keys on the keyboard to move ingredients around and make food to fulfill the orders of the customers. If the number of orders waiting to be filled is more than 5, then the user loses a life. To complete an order, the user puts the ingredients into a hole in the ground. If the user gets the order wrong, the order disappears into the hole and they have to redo it.
- Lives: For every 10 orders completed, the user gains a life. They can have a maximum of three lives. A life can be used if the place floods or burns down. If the player loses a life, any disasters and orders are cleared.
- Counter: A table to put down a single dish the player is holding. There is only one table in the entire kitchen.
- Start: The game starts with an instructions screen, and the user has to click the "Click me to start!" button to start the game. The start screen also shows the user how to use the different keys on the keyboard to control their motion.
- Ingredients: The player must pick up the correct ingredients for an order located around the kitchen, including roadkill and rotten tomatoes.

- Fire: The user sees a flame, and if the user doesn't put out the fire with the fire extinguisher in 10 seconds, the place burns down and the user loses a life or the game ends. The user puts out the fire by waving the fire extinguisher in front of the fire while holding down the left mouse button for 2 seconds.
- Power outage: The screen goes dark as if the lights in the kitchen have gone out. The user must make their way to the light switch, while not being able to see. If the user doesn't turn the lights within 10 seconds, they lose a life. To turn back on the light switch, the user must press the enter key.

Want-to-have Features:

- Increase the Difficulty (each time an order is completed, the next order joins the order shelf at a faster rate. The first order must be delivered in 1 minute, the second order must be delivered in 55 seconds, the third order must be delivered in 50 seconds.) As the speed increases, the difficulty of the order will also increase.
- Add more food and recipe options for the user
- Currency: The user gains 10 coins for every order they finish, displayed in the top right corner of the screen. After the game ends, the value is displayed on the screen, acting as the total points the player earned throughout the game.
- Sprint: The player has a bar that indicates their sprint. The user can hold the shift key to move faster, which will deplete their sprint bar. The sprint bar naturally refills over time, so that the total bar fills up within 1 minute. While sprinting, the user can use wasd keys or the arrow keys to move.
- Background music: During the entire game, cheerful and upbeat music will be playing in the background.
- Stoves: Some ingredients should be placed on the stove to cook the item before being added to the dish.

Stretch Features:

- Create moving customers who order their food in the kitchen rather than their paper orders
- More realistic looking disasters (flooding shows water slowly filling the kitchen, fire looks like fire, etc.)
- Multiplayer: 2 players can work together in the same kitchen, where orders come faster and disasters are more frequent.
- Sound effects (just like fire burning, pan frying, water running, orders coming, etc. → typical kitchen sounds)

Class List:

- Player class: runs the player for the game, allows the player to move around the screen, pick up, and drop off objects
- Orders class: The order class will be used to create an order, randomizing between the possible ingredients

- Ingredients class: Ingredients can be picked up and have their own icons on the grid, extends the interactions class, The ingredients are: onion, squirrel, rotten tomato, rabbit, smelly cheese, radioactive slime
- DrawingSurface class: the drawing surface for the game, and uses the ScreenSwitcher
- Disaster class: A disaster in the game that the player has to account for and fix in a set amount of time, Each type of disaster has its own unique fix
- Main class: main class to run the program and open up a new drawing surface, also used to run music
- Interactions class: A superclass for anything that can be interacted with (picked up and dropped/used)
- Counter class: The table that the player can place one dish on
- Hole: The hole in the ground that the player drops food through
- SecondScreen: game screen with the grid
- Screen: screens the player can use
- FirstScreen: the opening screen for the player that shows the controls for the game and a button to start
- Music class: allows the game to play background music
- ScreenSwitcher: interface for the screens
- ThirdScreen: The ending screen for the player that can take the player back to the menu screen

Credits:

[Gives credit for project components. This includes both internal credit (your group members) and external credit (other people, websites, libraries). To do this:

- List the group members and describe how each member contributed to the completion of the final program. This could be classes written, art assets created, leadership/organizational skills exercises, or other tasks. Initially, this is *how you plan on splitting the work*.
- Give credit to all outside resources used. This includes downloaded images or sounds, external java libraries, parent/tutor/student coding help, etc.]

- Sohum: Graphics, SecondScreen, Hole, Player, Music
- Iris: ThirdScreen, SecondScreen, Orders, Ingredients, Hole, Disaster, Counter
- Chloe: SecondScreen, ScreenSwitcher, Screen, FirstScreen, Player, Orders, Interactions, Ingredients, Music, Main, DrawingSurface
- Processing, Mr. Shelby, java libraries, music and images from the internet