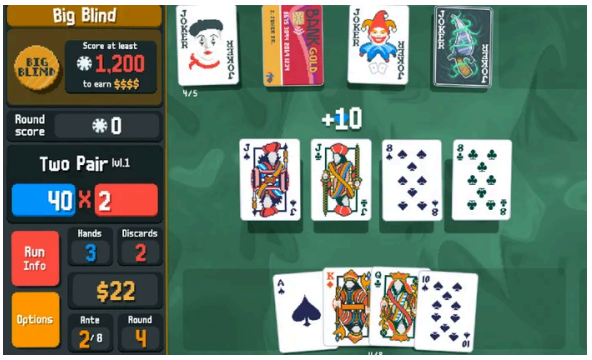
Bologna Kitchen

1. Project Overview

Bologna Kitchen is a Rogue-like game that you play as a chef in an high-stake underground cooking competition, and you have to cook the best meal ever made by combining various ingredients cards and using strong utensils. Become the strongest chef of the universe today.

2. Project Review

The so-called original project is a Rogue-like game called Balatro, Bologna Kitchen is a mere replica of the game, the difference is that we are using Python to create the game bit-by-bit and the theme now changes, implementing that we can make different in-game mechanics, such as; new categories of power-up cards, and such,



3. Programming Development

3.1 Game Concept

In this game, you must draw ingredient cards from the draw pile, you may pick to discard or play those cards, combining those cards will give a score based on the combination. The game also presents spice cards and utensil cards that can make your deck of ingredients stronger, and also chef cards that help you in order to score bigger points.

3.2 Object-Oriented Programming Implementation

- Ingredient Cards (***Instanced***, each of the playing cards)
- Hand (***Singleton***, what cards you currently have on hand)

- Draw Pile (***Singleton***, stores remaining cards)
- Chef Cards (***Instanced***, makes you score more point)
- Chef Cards Storage (***Singleton***, stores chefs; 5 by default)
- Utensils (***Instanced***, acts like single-use power-ups that customize your draw pile)
- Spices (***Instanced***, acts like single-use power-ups that “levels up” one type of dish)
- Bag (***Singleton***, acts like a bag that keeps utensils and spices)
- Statistics Manager (***Singleton***, keeps track of some statistics within the game)

3.3 Algorithms Involved

This game is mainly based on simple math multiplication and addition, the game for its root isn’t complicated, what will be complicated is that each hand draw will be randomized from the draw pile, but the cards within your hand is able to be sorted by rank and suite, similar to the base game, which will probably utilizes sorting algorithms; that will need a lot of optimization in order to make the “sort option” fast enough to feel plausible to play. Another thing is that maybe we need to make a guarantee system in order to have a sort of “pity” system that doesn’t make the game’s randomness too tedious to deal with.

4. Statistical Data (Prob Stats)

4.1 Data Features

- Most Played hand this run
- Consumables Used
- Card Played
- Card Discarded
- Chef Tables (or Rounds played)

Food-Themed Hand Names

Poker Hand	Food-Themed Name	Description
High Card	Appetizer	Just a single dish—not enough for a meal, but it might still be tasty.
One Pair	Matching Sides	Two identical sides, like two orders of fries.
Two Pair	Surf & Turf	A combo of two different but complementary dishes.
Three of a Kind	Triple Stack	Three identical items, like a towering burger or pancakes.
Straight	Full Course Meal	Five dishes arranged in a proper sequence (appetizer to dessert).
Flush	Theme Buffet	All dishes from the same cuisine (e.g., all Italian or all BBQ).
Full House	Chef's Special	A mix of three mains + two sides, like a perfectly planned meal.
Four of a Kind	Family Platter	A big serving of four identical items—great for sharing!
Straight Flush	Gastronomic Symphony	A perfectly balanced, high-end multi-course meal.
Royal Flush	Ultimate Feast	The absolute peak of dining—luxury, balance, and perfection!

"Hands" would look something like this, the description might not make sense to you, but it will in the final product.

	Why is it good to have this data? What can it be used for?	**How will you obtain 50 values of this feature data?**	**Which variable/class will you collect data from?**	**How will you display this feature?**
Most Played hand this run	It can be used to statistically represent how frequently the player plays one singular "hands," which is bluntly their strategy.	Make the game simultaneously track each time the player plays a hand.	There will be a global variable which tracks the number of hands played this run.	Pie Chart
Consumables Used	This set of data is used to analyze the trend of strategies that the player might initialize. This data will be cumulatively collected every game played, not just within 1 run.	The game will take note of every Utensils, and Spices used for every round.	There will be a class that stores utensils and spices used separately, though it will show within the same graph.	Bar Graph / Histogram
Cards Played	This set of data can be	I believe that in one run, a	There will be a class that	Bar Graph for each suite and

	used for players to keep track of what card they've played this round, also for them to know how frequently they play one singular card.	player, if they're not actively trying to lose, will play more than 50 cards.	stores the cards played by duplicating them into a storage within that class.	number, separately shown
Cards Discarded	Similar to Cards Played, this set of data can be used for players to keep track of what card they've played this round, also for them to know how frequently they discard one singular card.	Also similar to Cards Played, I believe that in one run, a player, if they're not actively trying to lose, will discard more than 50 cards.	There will be a class that stores the cards discarded by duplicating them into a storage within that class.	Similar to Cards Played
Chef Tables / Rounds Played	This set of data is good to keep track of a person's record, like how far they've progressed into the game, and their highest score.	After a few runs, the round collected will eventually reach 50, for what I think that the average rounds per run would be 15-20.	There will be a global variable which tracks the number of rounds passed this run.	Bar Graph

3.2 Data Recording Method

The game will store initial data by adding to a temporary value in the python code file, but after each and every round; the score will be added to a CSV File which will keep track of statistics after every round. Data does not continuously update for that the game will not be well optimized.

3.3 Data Analysis Report

You are able to view the recorded data whenever you want, by clicking "P" on your keyboard. The statistics will be shown as a bar graph for each of the 5 data points, data will be as updated as of the last round.

4. Project Timeline

Week	Task
1 (10 March)	Proposal submission / Project initiation
2 (17 March)	Full proposal submission
3 (24 March)	Making the game "Playable," by making scoring possible, and adding in some Chef Cards.
4 (31 March)	Add in Utensils and Spices cards, and make data collecting possible.
5 (7 April)	Finish adding in Chef Cards, and make some visual tweaks
6 (14 April)	Submission week (Draft)

5. Document version

Version: *2.0*
Date: *16 March 2025*

Date	Name	Description of Revision, Feedback, Comments
15/3	Rattapoom	Very good! But don't forget to add a project timeline.
16/3	Parima	The proposal is great in detail and well formatted.

29/3	Rattapoom	For “How will you display this feature?”, you need to specify what type of data representation (Graphs, Charts, Histograms, Descriptive Statistics). Also please put the variable you’re planning to use to keep track of your data features in the UML class diagram.
29/3	Parima	Statistical section should provide more detail on how to display the data.