

Precious Oziwo
Ayodeji Olagoke
Ruike Yuan

Start Document

Bar-Arcade Game

BAR-ARCADE GAME DEVELOPMENT GROUP

NHL STENDEN UNIVERSITY OF APPLIED SCIENCES
VERSION 3.0

Table of contents:

Version Management.....	2
Background Information.....	3
Mock-ups and detailed information	4
Test Plan:.....	11
Moscow Analysis:	12
UML diagram.....	13
References:.....	14

Version Management

<i>Version nr.</i>	<i>Changes</i>
1.0	Background Information Mock-ups and detailed information Feasibility Moscow Analysis References
2.0	UML diagram && Test Plan added
3.0	Improved UML diagram

Background Information

The basic objective of the game is to order drinks by playing minigames and earning arcade coin tokens. Drinks can only be purchased using tokens. To earn tokens, players need to play the mini-arcade games in the bar. There are two minigames by default, a whack-a-mole game and a quiz game that uses an API.

Players will be able to test their knowledge by responding to questions from the API in the quiz game. On the other side, the whack-a-mole game tests participants to whack as many moles as they can in each amount of time. There will be a game over in each of the two minigames if the player fails to accomplish the task.

The number of tokens the player can earn by winning the mini games are defined by the development team members. There is no rule/limit by default for players to play these mini games, so players can play as many times as they want, but this may be changed/redefined by the development team members.

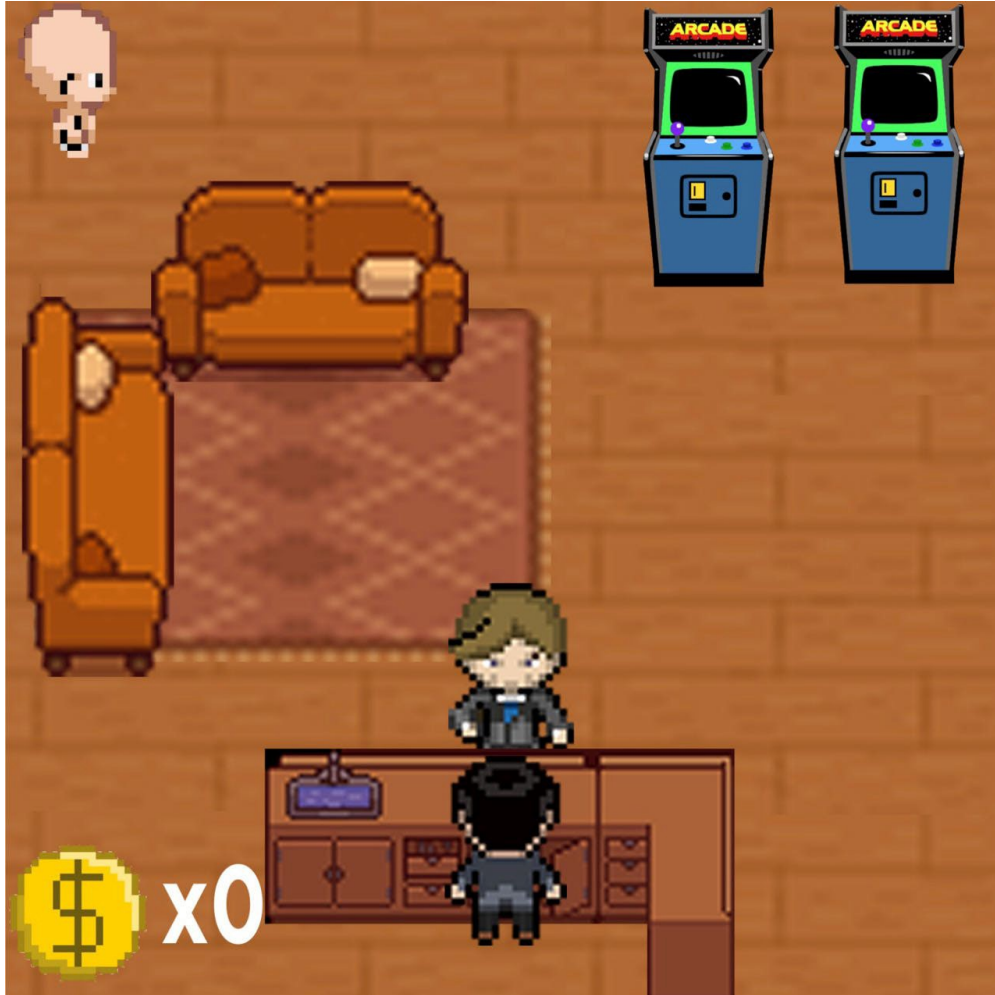
Players will be amused by background music that is hardcoded into the game as they consume their drinks. The mood of the bar may also be further enhanced by the presence of AI NPC characters that interact with players and waiters and other staff members that attend to their needs.

To make it more interactive, the NPC will respond to whatever questions players ask, the player can ask anything they would like to ask but the questions will be filtered before the answer is generated. By doing so the system identifies the invalid questions and respond accordingly. This capability of answering questions will be accomplished by pulling chat GPT answers/or hardcoded answers, giving players a dynamic and interesting experience.

Whether more functionalities or more mini games will be added depends on the decision of the development team.

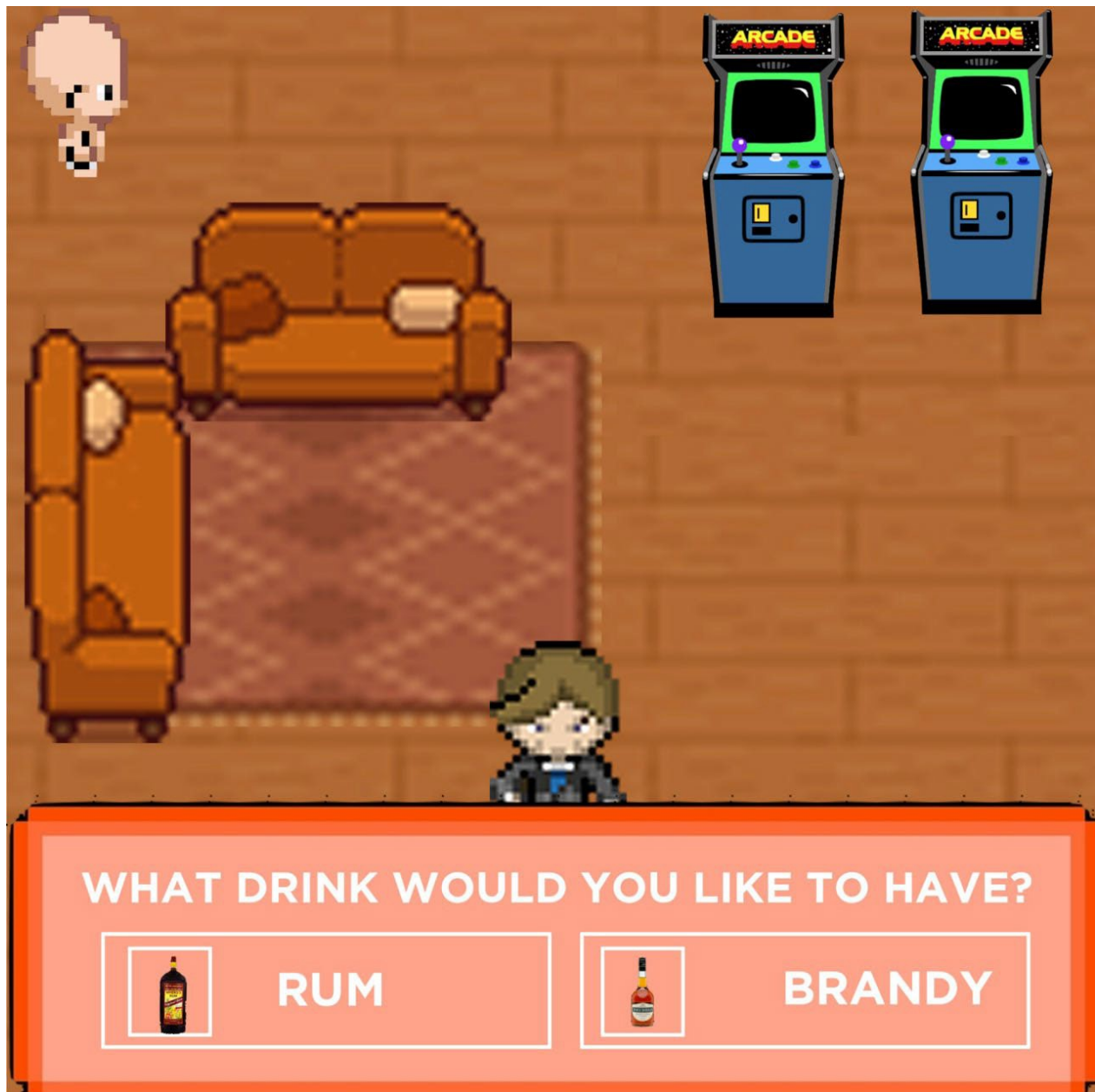
Mock-ups and detailed information

When the player gets into the game, it will be directed to a bar scene where a front desk (NPC) and arcade-game machines are shown.



The player walks to the front desk and talks to the NPC there.

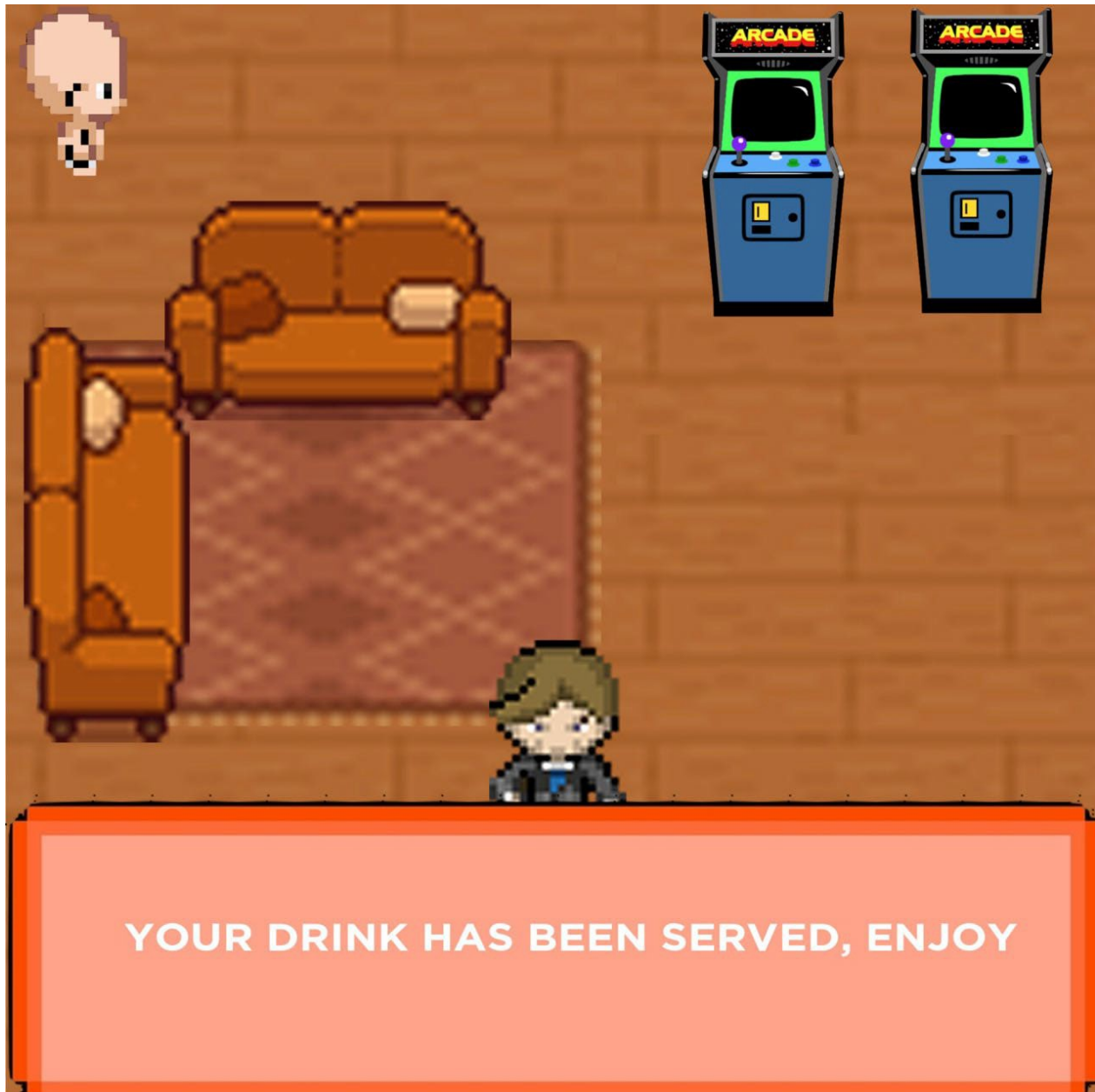
In this case (in the image), the question can be: "I would like to order some wines, can you show me some wines."



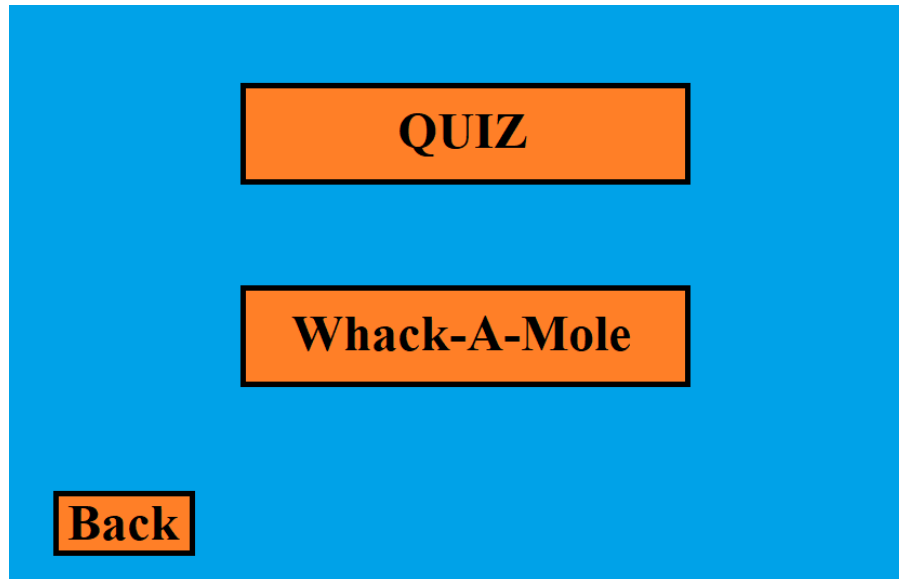
The list of wines is shown to the player.



Here are the replies of the NPC.

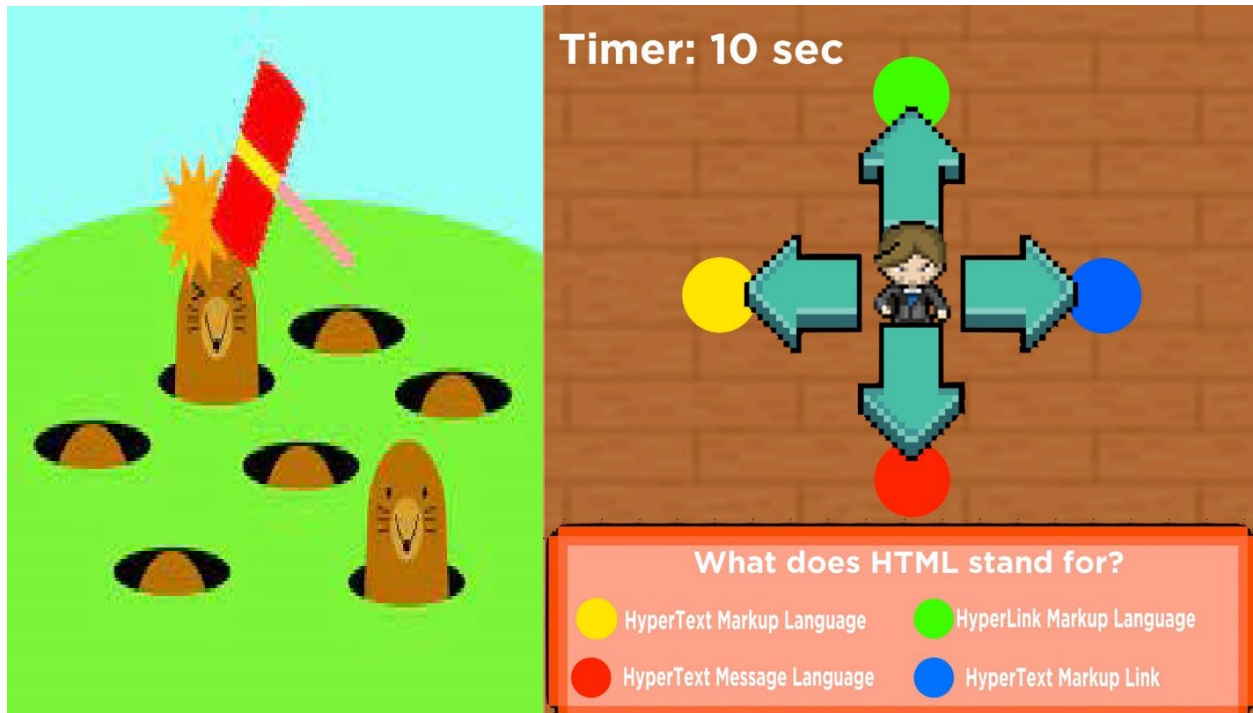


A notification(success) will be displayed if the player has enough tokens to buy the drink. On the other hand, if there are not enough tokens to consume drinks, a different type of notification will be shown.



When the player clicks the button on the arcade machine, a screen will pop up and the players can select which mini game they would like to play.

Also, there is a “Back” button that can direct back to the bar.



Each game features a loading screen. end screen and high score. The score will be saved, and the number of coins(token) earned will be calculated.

Whack a mole can be played with 3 lives where every 10 moles hit equals to a coin.

Quiz is timed where every question answered resets the time. 10 secs per question. If the answer fails or you run out of time queue game over screen.

Feasibility:

Why it is feasible:

1. The team will use Monogame as the main tool (game engine) for developing the game. As it is open-sourced and has a lot of tutorials that the development team members can study. By studying how to use this game engine and related projects like creating NPC and scenes, it is highly possible that the team will complete the project.
2. The team will use APIs for sourcing/retrieving the necessary data like the list of drinks, music and chatting messages in the game. Since there are tons of websites providing free-API access the team can easily get the information needed, and by using C# libraries the team can locate and classify the necessary data and make use of them. As a result, the functionalities like chatting NPC and menu can be achieved easily from an API, guaranteeing that players have access to a variety of beverages from which to choose from and the chat NPC can reply accordingly.
3. The scene in the game will be created using textures. Texture resources can be found easily online and there are mono-game integrated tools that can be used to create scenes with texture resources. As a result, there is no need to worry about whether the scenes can be made.

To conclude, it is highly feasible for the development team to finish the project.

Test Plan:

Version: 1.0

Date: 2023-05-20

Tester: Team

Test Environment:

- Windows 10
- Chrome

Test Cases

Minigames

- Whack-a-mole
 - Verify that the moles appear randomly.
 - Verify that the moles disappear when they are whacked.
 - Verify that the player loses the game if they do not whack enough moles in the allotted time.
- Quiz game
 - Verify that the questions are pulled from the API correctly.
 - Verify that the player receives points for correct answers.
 - Verify that the player loses the game if they run out of lives.

Drinks

- Verify that the list of drinks is pulled from the API correctly.
- Verify that the player can order drinks with the arcade coin tokens they have earned.
- Verify that the player can consume their drinks.

Background Music

- Verify that the background music plays correctly.
- Verify that the volume of the background music can be adjusted.

NPC Characters

- Verify that the NPC characters interact with the player correctly.
- Verify that the NPC characters can answer questions from the player correctly.

Overall Experience

- Verify that the game is fun and engaging.
- Verify that the game is free of bugs.
- Verify that the game is easy to use.

Conclusion

This test plan outlines the steps that will be taken to test the game Bar Games. The test cases are designed to ensure that the game is functional, enjoyable, and free of bugs.

Moscow Analysis:

Must haves:

Quiz game

Wrack a mole game.

Chat NPC

Background music and action sounds

Game over/high score scenes/selecting game scene

Database implementation for the high-score(high score updated each time entering the game)

Bar scene with arcade machine and buttons

Should haves:

Smooth screen transition between all different screens(scenes).

Better graphics (sprites, scenes)

Animations (character movement, background objects)

Could Haves:

More advanced mini games

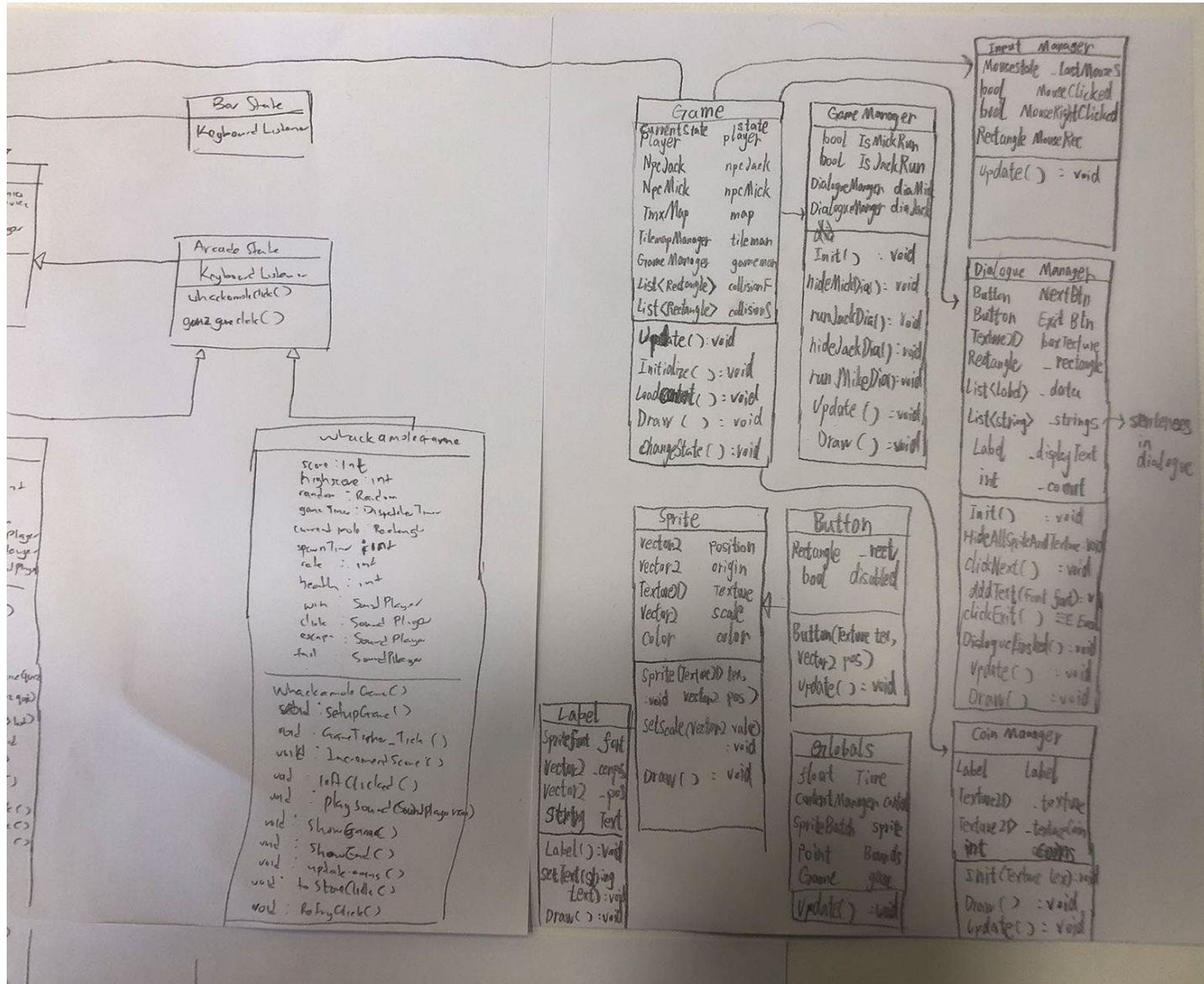
More scenes with more functionalities (like a food shop)

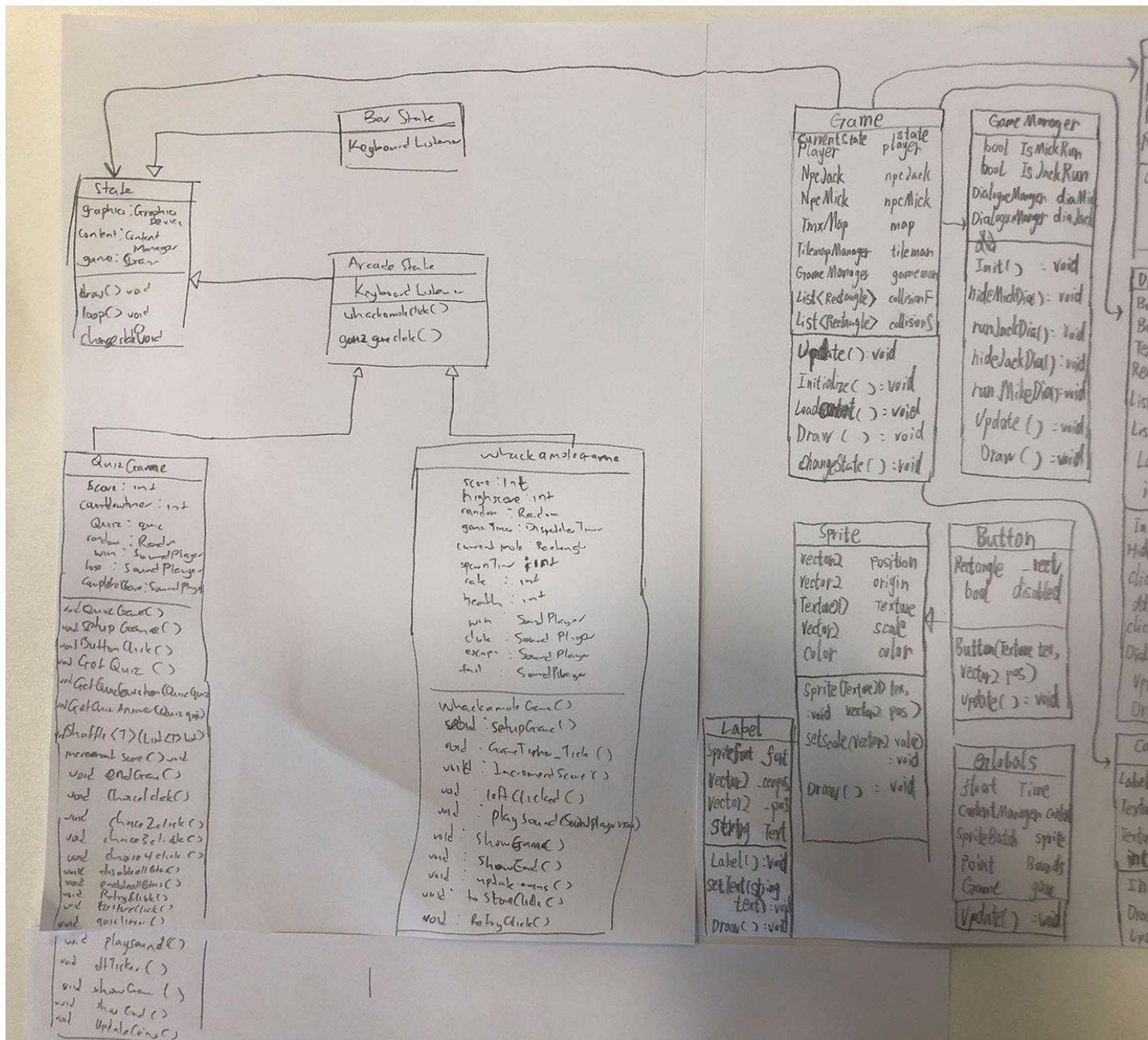
Won't haves:

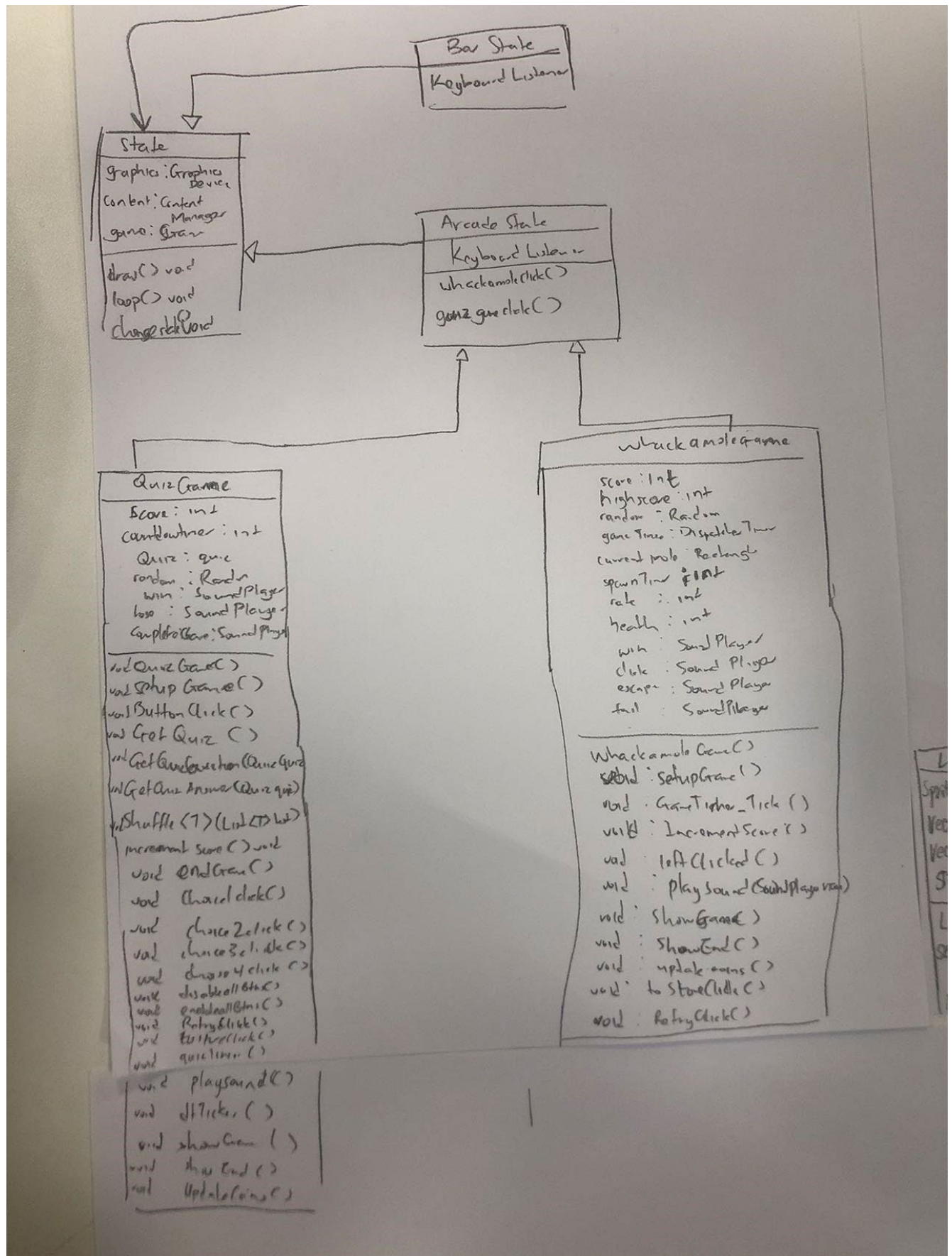
Bilingual (other language than English)

UML diagram:

*This is the version improved







References:

Game Engine

Schaub, K. (n.d.). *Monogame: Introduction to C# game programming*. Udemy. Retrieved April 30, 2023, from <https://www.udemy.com/course/monogame/>

One framework for creating powerful cross-platform games. MonoGame. (n.d.). Retrieved April 30, 2023, from <https://www.monogame.net/>

Assets Websites

Free 2D game assets - craftpix.net. (n.d.). Retrieved April 30, 2023, from <https://craftpix.net/freebies/>

Top free game assets. itch.io. (n.d.). Retrieved April 30, 2023, from <https://itch.io/game-assets/free>

API

Bars apis (free tutorials, SDK Documentation & Pricing). RapidAPI. (n.d.). Retrieved April 30, 2023, from <https://rapidapi.com/search/bars>

Build your bot using chatbot API. ChatBot. (n.d.). Retrieved April 30, 2023, from <https://www.chatbot.com/docs/>