Nikomon Pokémon-like game Proposal

Abstract

Pokémon is the most valuable IP in the world and also many people's childhood. We want to create a Pokémon-like game called Nikomon(But in this document we still call our game sprite as Pokémon). The system we mainly want to realize is turn-based battle system including complex effect system, Pokémon development including base status and level/evolution system and plot system including npc, dialogue, choice and even task. We will use Pokémon sprites design and also original character in our game. The feature of our design is that we consider a designer friendly framework to organize all data and game script.

Team

杜昊澄 Haocheng Du 11911124

吴一凡 Yifan Wu 11911223

金冬阳 Dongyang Jin 11911221

赵云龙 Yunlong Zhao 11911309

## Description

Motivation

- We want to create a game designer friendly develop system. In this game, it means a game designer do not need to do any hard core programming, just fill some blank and write lua scripts and then a Pokémon can be added to game.
  - This means we can not use other design pattern to avoid designers to touch true game logic. After some research, we decide to use some
- Game UI Design is not like web development, which have many framework to use and some logic to follow. And also when UI become complex, the game object reference may become much more complex. We need to use some design pattern to solve the complex reference between game object and ui element.

Feature Description

For Game Designer

We want to create a designer friendly game development. Though our team member all know how to program, for more game development teams, the game designer may have no idea how to program. So we will create a designer friendly development mode to solve this problem. The key idea is that game data must be separated with game logic code. We

will explain our idea by some user stories.
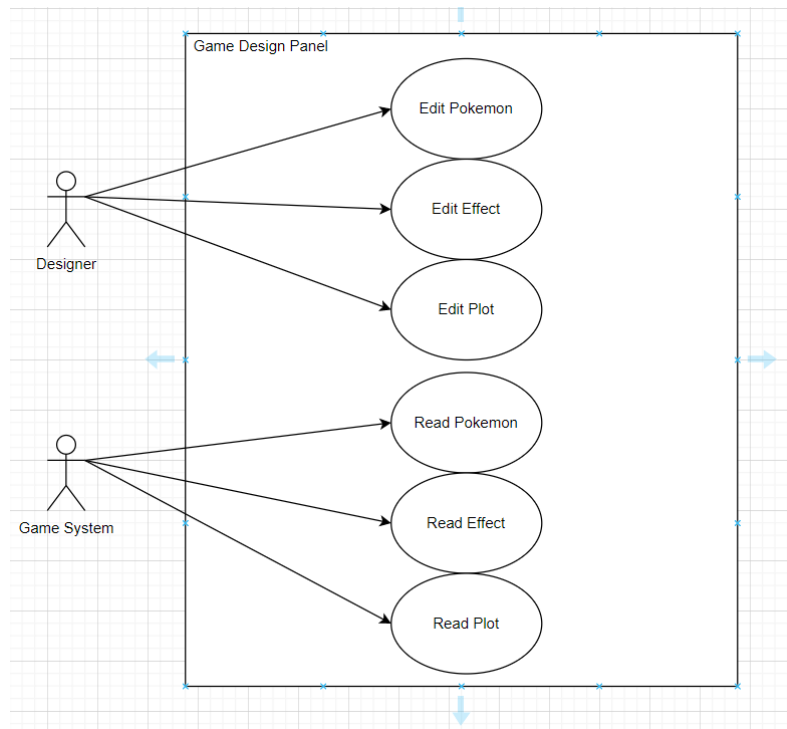
Edit Pokémon

Game designer will fill Pokémon data and save it as a json file. Each Pokémon will have an identical id as well as their moves(招式), rates(种族), evolution chains(进化链). When the game start, this data file will be read into memory and can be shared by the whole game.

Edit Effect

Even though we want to separate game logic and the game data, some data itself has logic. For instance, some Pokémon moves has special effect like making your Pokémon into sleep and in the next battle turn, this sleeping Pokémon can not use any moves. We need to realize this effect, but we do not want game designer to just code this logic into core game logic for both code safety and designer friendly, so we use Lua as scripting language to realize the complex effect. Lua is widely used in American RPG games like Baldur's Gate and World of Warcraft.

Edit Plot（剧情）

We use Yarn Spinner to write plot and embed it into game.

For Player

Main Interaction

Battle Interaction

Battle will start by battle handler and then initialize any essential component related with battle such as battle UI and battle Pokémon model.

Then from UI player choose each Pokémon's move(we want to build a n vs n turn-based system so the Pokémon player have on the battle scene may be more than one). Repeat this work until the battle ends.

Bag & PC Interaction

When players go to check the warehouse, they can choose to check the bag or PC. After selecting one of them, you can choose to view the corresponding pokemon. Then you can choose to view the properties of pokemon, release pokemon, exchange the location of pokemon between bag and bag, bag and PC, PC and PC, and use items.
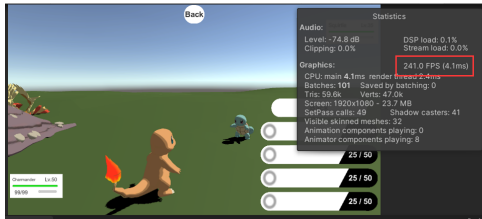
Requirements

Computer configuration:

CPU: Intel(R) Core(TM) i7-8564U CPU @ 1.80GHz   Memory: 8GB

GPU: NVIDIA GeForce MX250   Resolution Ratio:1920*1080
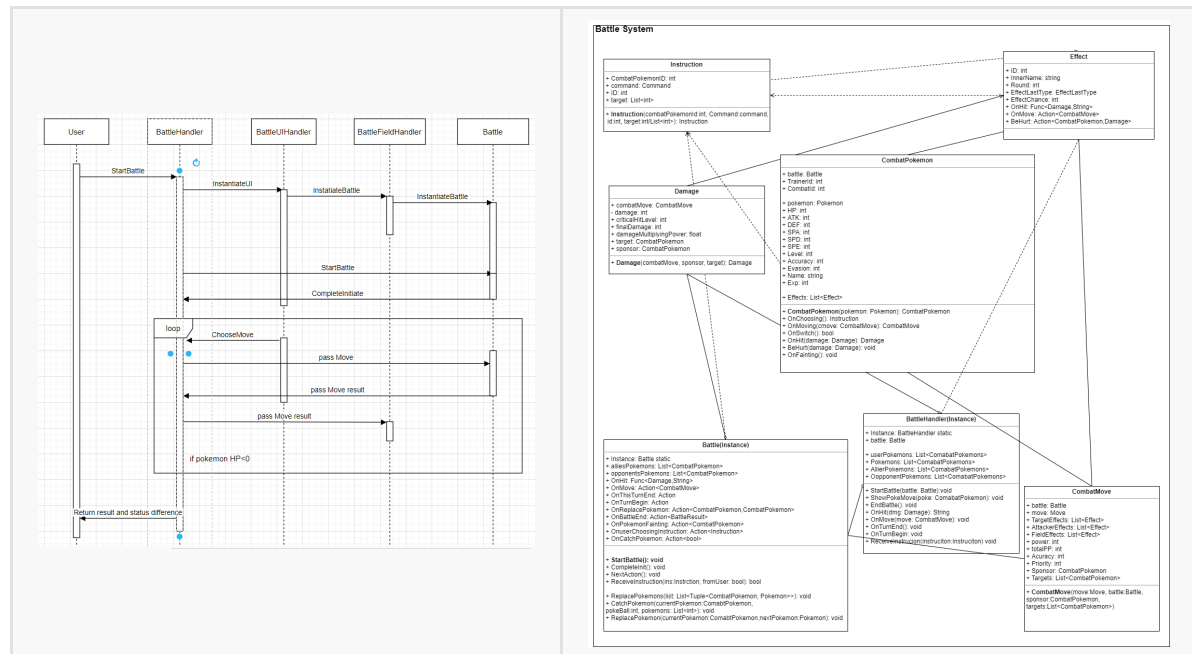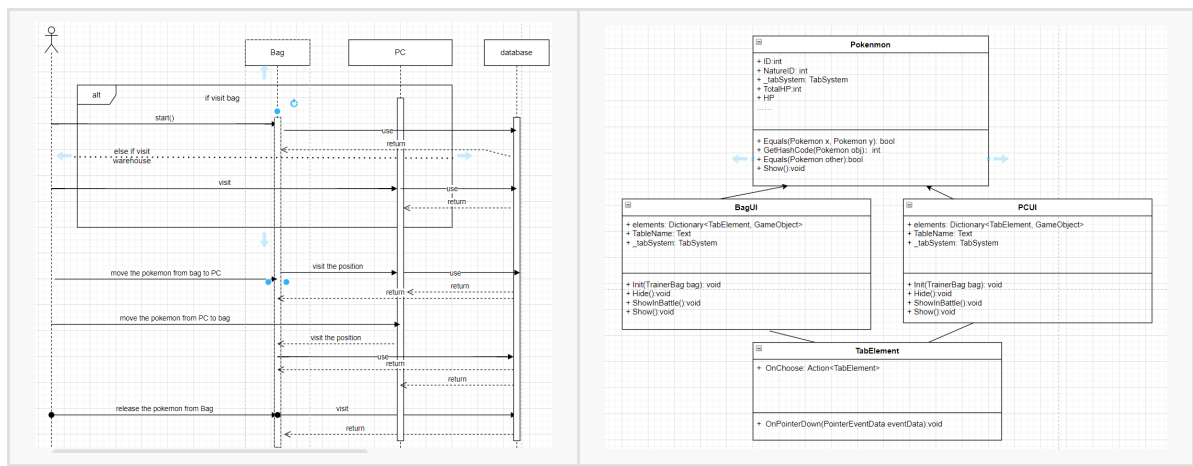
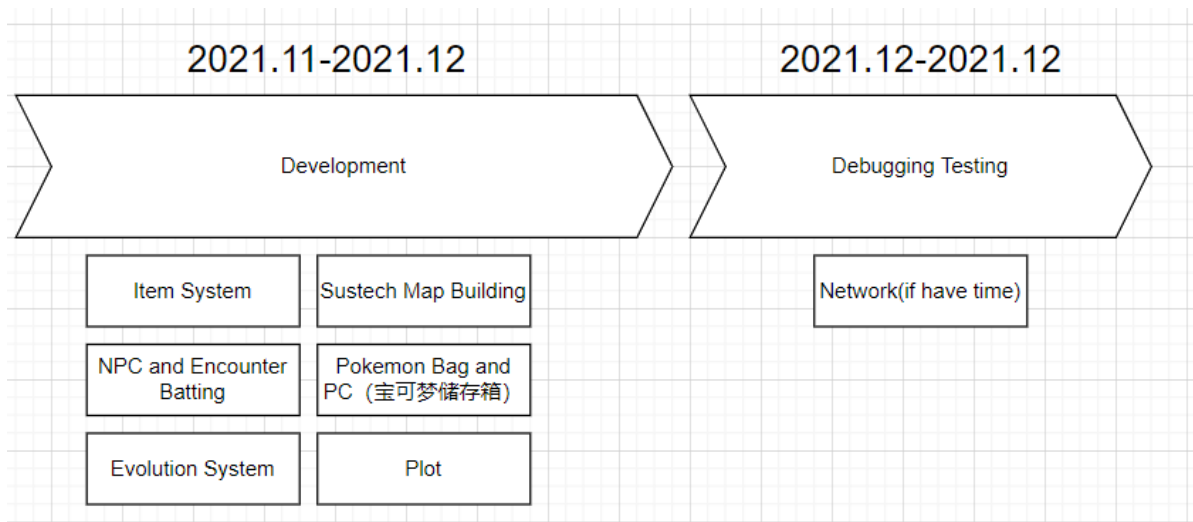| Battle Scene:	Profiler:

Design Document

Architecture

Battle System

Interact with Bag UI

Timeline

Feasibility

In the process of making the game, we are relatively familiar with the game Of Pokémon, so THE basic design of the combat system was relatively smooth. At the same time, we also referred to the User Interface(UI) of *Pokémon: Sword/Shield*

During the battle, since Pokémon has different status, different restraint relationships between status, use of items and different effects caused by Pokémon's skills. so our judgment in battle becomes very complicated. This complicated effect may lead to a failure.

Used APIs and Technology

Unity 2020 LTS

Cinemachine, Used for camera control

Newtownsoft.Json, Used for reading and saving json file

Lua, Used for scripting game effect and other game data containing logic

Yarn Spinner, used for plot editing