Pokemon Game

Submitted By:

Zariful Islam
Id: **2131593642**
North South University
zarif.zoha@northsouth.edu

Md. Monirul Haque Arnob
Id: **2132218642**
North South University
monirul.arnob@northsouth.edu

Table of Contents

Introduction:

At this point of time, it'd be very hard to find someone who hasn't heard about the game. From 3 years old to senior citizens almost everyone played some game throughout their lives. Games are like a tool to escape reality. A new world with new identity ,new goals,new objectives. Everyone needs a break, a quick match of **PokémonGame** would be very great for them.

Game Overview:

This report shows the development of a 2D Top-Down desktop game, **PokémonGame**. The project is based on Retro Pixel/Java games containing 4 different types of Pokémon for the player to play. It has multiple achievable powers for each Pokémon, computer controlled intelligent opponent where players attack with different skills and also can heal when needed. This project explores a new dimension to the traditional Retro Pixel 1V1 games by mixing the features of different powers and attack strategy where the player has to fight the opponent to win. With the simplicity of this game it simply aims to bring fun and make you look back to your childhood's

Targeted Audience:

We're mainly targeting casual gamers aged from 10-21.

Game Description :

This game involves a single player entering a battlefield with 3 Pokemons after selecting them from the menu of 4 types. Players will deploy their pokemon chronologically based on their selection on the pokemon choosing page. There is an intelligent bot opponent against which the game would be played that will attack the player. If the player successfully defeats the enemy 1st pokemon the enemy will send their next pokemon chronologically and same goes for the player too. If all of the enemy's pokemon die, the player will win the fight and vice versa.

Game Objectives:

- To create a 1v1 game
- To Implement normal power
- To implement special power
- To implement heal
- To design the game that will be simple and pleasant to look at
- To make a User Friendly Interface that will be eye-catchy

Technical Feasibility:

The tools and technology that were used in the making of **PokémonGame** are:

Image Editor Adobe Illustrator CC 2022

Code Editor: Eclipse

Programming Language: Java

Github Link :

Link : https://github.com/Zuana15/Pokemon-game

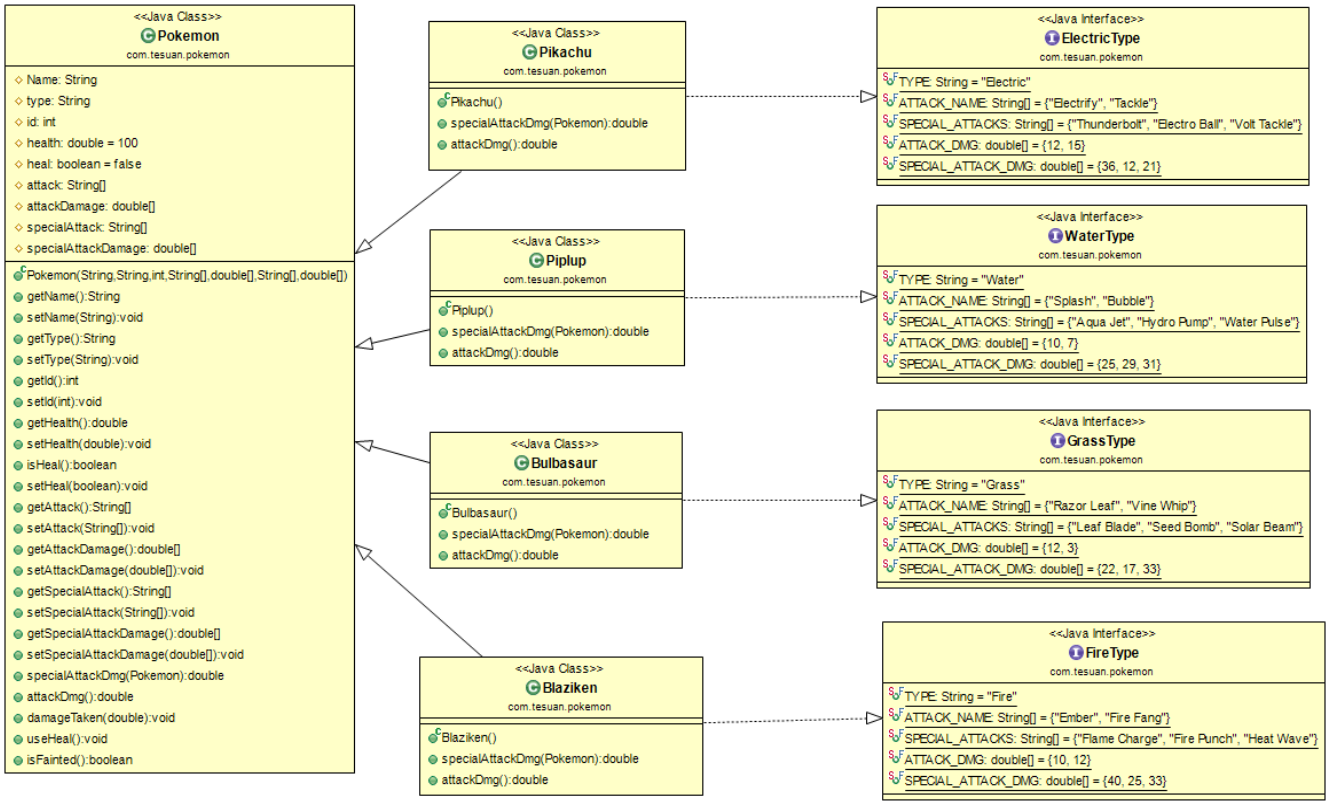
System Design:

Classes Names with description:

Number	Class Name	Responsibility
1	**Pokemon**	Assigns attributes to all the pokemons including their name, type, health, attacks etc.
2	***WaterType***(Interface)	Includes all the attack names and damage values of the Pokemon of Water type
3	***GrassType***(Interface)	Includes all the attack names and damage values of the Pokemon of Grass type
4	***ElectricType***(Interface)	Includes all the attack names and damage values of the Pokemon of Electric type
5	***FireType***(Interface)	Includes all the attack names and damage values of the Pokemon of Fire type
6	Blaziken,Charizard, Magmar, Moltres, Growlith	Includes the properties of the Pokemon class and inherits the FireType interface and its properties
7	Piplup,Squirtle, Froakie, Tentacruel, Mudkip	Includes the properties of the Pokemon class and inherits the WaterType interface and its properties
8	Bulbasaur, Sceptile, Torterra, Snivy, Chikorita	Includes the properties of the Pokemon class and inherits the GrassType interface and its properties
9	Pikachu, Manectric, Electivire	Includes the properties of the Pokemon class and inherits the ElectricType interface and its properties
10	MainMenu	A JFrame class which is the main menu of the game showing two buttons; one to start the game and another to exit the game.
11	ChooseYourAgent	A JFrame class that executes when the Start Game button is pressed in MainMenu. It shows six different buttons. Four of them to select Pokemons

		of four different types. One of them is to reset the selection. And another button,FIght, is to start the battle of the Pokemons. Each Type can be opened once and selected from. The order of selection is used in the battle for Pokemon deployment which happens automatically.
12	WaterTypeGUI	A JFrame class that shows all the five available water type pokemons which checkboxes to choose them as User's Pokemon for the battle.
13	FireTypeGUI	A JFrame class that shows all the five available fire type pokemons which checkboxes to choose them as User's Pokemon for the battle.
14	GrassTypeGUI	A JFrame class that shows all the five available grass type pokemons which checkboxes to choose them as User's Pokemon for the battle.
15	ElectricTypeGUI	A JFrame class that shows all the three available electric type pokemons which checkboxes to choose them as User's Pokemon for the battle.
16	User	Stores the User selected Pokemon in an Arraylist inside it
17	**Fight**	The main JFrame class for Pokemon battle. Three Pokemons are assigned to the Computer and then Users and Computers first Pokemons are deployed. Then the user can use the Attack button for a default attack or the Special Attack button for a special attack based on the opponent's Pokemon Type. Then the Computer also performs a random attack. The health bar shows the current health of both the Pokemon in the battlefield. There is a heal button to heal the User's pokemon too. When a Pokemon faints, another one is sent and the game continues till either the user or the Computer is left with no Pokemons. The last trainer standing wins.
18	Win	A JFrame class that executes when the User wins the game
19	Lose	A JFrame class that executes when the User loses the game

UML Diagram:


<<Java Class>>
Pokemon
com.tesuan.pokemon
Name: String
type: String
id: int
health: double = 100
heal: boolean = false
attack: String[]
attackDamage: double[]
specialAttack: String[]
specialAttackDamage: double[]
Pokemon(String,String,int,String[],double[],String[],double[])
getName():String
setName(String):void
getType():String
setType(String):void
getId():int
setId(int):void
getHealth():double
setHealth(double):void
isHeal():boolean
setHeal(boolean):void
getAttack():String[]
setAttack(String[]):void
getAttackDamage():double[]
setAttackDamage(double[]):void
getSpecialAttack():String[]
setSpecialAttack(String[]):void
getSpecialAttackDamage():double[]
setSpecialAttackDamage(double[]):void
specialAttackDmg(Pokemon):double
attackDmg():double
damageTaken(double):void
useHeal():void
isFainted():boolean
<<Java Class>>
Pikachu
com.tesuan.pokemon
Pikachu()
specialAttackDmg(Pokemon):double
attackDmg():double
<<Java Class>>
Piplup
com.tesuan.pokemon
Piplup()
specialAttackDmg(Pokemon):double
attackDmg():double
<<Java Class>>
Bulbasaur
com.tesuan.pokemon
Bulbasaur()
specialAttackDmg(Pokemon):double
attackDmg():double
<<Java Class>>
Blaziken
com.tesuan.pokemon
Blaziken()
specialAttackDmg(Pokemon):double
attackDmg():double
<<Java Interface>>
ElectricType
com.tesuan.pokemon
TYPE: String = "Electric"
ATTACK_NAME: String[] = {"Electrify", "Tackle"}
SPECIAL_ATTACKS: String[] = {"Thunderbolt", "Electro Ball", "Volt Tackle"}
ATTACK_DMG: double[] = {12, 15}
SPECIAL_ATTACK_DMG: double[] = {36, 12, 21}
<<Java Interface>>
WaterType
com.tesuan.pokemon
TYPE: String = "Water"
ATTACK_NAME: String[] = {"Splash", "Bubble"}
SPECIAL_ATTACKS: String[] = {"Aqua Jet", "Hydro Pump", "Water Pulse"}
ATTACK_DMG: double[] = {10, 7}
SPECIAL_ATTACK_DMG: double[] = {25, 29, 31}
<<Java Interface>>
GrassType
com.tesuan.pokemon
TYPE: String = "Grass"
ATTACK_NAME: String[] = {"Razor Leaf", "Vine Whip"}
SPECIAL_ATTACKS: String[] = {"Leaf Blade", "Seed Bomb", "Solar Beam"}
ATTACK_DMG: double[] = {12, 3}
SPECIAL_ATTACK_DMG: double[] = {22, 17, 33}
<<Java Interface>>
FireType
com.tesuan.pokemon
TYPE: String = "Fire"
ATTACK_NAME: String[] = {"Ember", "Fire Fang"}
SPECIAL_ATTACKS: String[] = {"Flame Charge", "Fire Punch", "Heat Wave"}
ATTACK_DMG: double[] = {10, 12}
SPECIAL_ATTACK_DMG: double[] = {40, 25, 33}

Pokemon Characters and Other Objects:

This is one of the interesting chapters that you may read with enthusiasm, because it contains lots of colorful figures.

- ## Pokemons:

Figure 1 : Pokemons

- ## Buttons:

Figure 2 : Buttons

• Windows:

• First Page(Welcome page):

Figure 3 : Welcome Page

• Second Page (Choosing Pokemon):

Figure 4: Choose

- **4 Third Pages (Types):**

Figure 5 : Types

- **Fourth Pages (Battle Field):**

Figure 6 : Battle Field