

TEMA POO

Pirvu Stefania-Daniela

Grupa 322CC

Timp de lucru: 7 zile

Dificultate 8/10

Am implementat jocul in pachetul Game. Clasele cerute sunt implementate conform enuntului, motiv pentru care nu le voi detalia pe toate.

In main apelez clasa Game si metoda gema.run()

Am folosit FactoryPattern , SingletonPattern, VisitorPattern si Builderattern , conform enuntului.

BONUS: Am implementat optiunea de a crea un nou Cont prin intermediul interfetei grafice

CLASA Grid are constructor privat .Grid este o lista de liste , iar in constructor creez o celulele de tip Empty , Enemy, Shop Finish.

Caracterul se afla mereu initial pe prima casuta din coltul stanga sus , iar finish in coltul dreapta jos. Setez celula curenta, initiala ca si vizitata.

Metoda generareMapa intoarce mapa hardcodata, iar metoda generareMapa2 intoarce mapa random, care insa are ppozitia initiala si finish setate ca mai sus.

Metoda de printare mapa parcurge intreaga lista de liste si afiseaza S /P/N/E/F in functie de tipul celulei sau ? daca celula nu a fost vizitata.

Metodele de deplasare sunt asemantoare asa ca voi detalia una singura.

Metoda goNorth primeste caracterul cuent , iar daca acesta se afla pe prima linie, nu se poate deplasa in nord si se afiseaza un mesaj corespunzator.

Daca se poate deplasa, atunci se marcheaza ca vizitata celula in care s-a mutat si se modifica coordonatele celulei curente.

CLASA Game

-game.run()

Citesc datele din fisierul json (am urmat un tutorial ca model

:<https://www.youtube.com/watch?v=HSuVtkdej8Q&t=397s>)

Astept input din linia de comanda pentru a alege modul de joc, email parola si caracterul ales

Daca mailul introdus nu exista in list de conturi sau parola nu corespunde se afiseaza mesaje corespunzatoare

Avem mai departe 3 metode

-1 start (joc CLI hardcodat)

-2 startnehardcodat (joc CLI nehardcodat)startGui (joc GUI)

-3 startGui (joc GUI)

1. In metoda start

Se apeleaza metoda readstories care citeste povestile din fisierul json si le retine in dictionar in functie de tipul acestora.

Se genereaza mapa hardcodedata

Se asteapta input din linia de comanda "P"

Cand tasta P este citita , iar caracterul este in viata, se muta pe urmatoarea casuta se agiseaza povestea (printastory)si o joaca(se apeleaza metoda play)

In print story, se afiseazaprima poveste, dupa care se adauga la final si o sterg de la inceput,(un fel de reciclare a povestilor in caz ca nu sunt destule pentru mapa aleasa)

In metoda play se verifica tipul celulei

Daca este empty atunci folosesc random, existant 20% sanse sa castige monede

Daca este enemy, atunci cat timp si viata caracter si viata inamicului este diferita de 0, se ataca pe rand, primind ca input tasta 'P'

Daca mai exista abilitati si poate sa le foloseasca, caracterul foloseste abilitati, dupa isi foloseste potiunile, iar cand ramane fara ataca normal

Enemy ataca random ori cu abilitati daca mai are, daca nu atac normal

Daca la final enemy are viata 0, atunci caracterul a castigat, experienta sa creste si eventual si nivelul

Daca nu se va afisa mesajul corespunzator si jocul se va termina

Daca este shop, se afiseaza lista cu potiuni , se apasa P si se cumpara cele 2 potiuni

Daca este Finish , se afiseaza mesajul corespunzator si se mai adauga o mapa la mapele terminate de jucator

Se iese din metoda play si se afiseaza harta.

2 In metoda startnehardcodat

Se apeleaza metoda readstories care citeste povestile din fisierul json si le retine in dictionar in functie de tipul acestora.

Se genereaza mapanehardcodata

Cat timp jucatorul nu a ajuns la finish sau nu a murit, se asteapta input de la tastatura : nord/sud/est/vest

Daca se tasteaza altceva nu se intampla nimic si se afiseaza mesajul corespunzator.

Jucatorul se muta in directia aleasa, se afiseaza mapa si povestea si se joaca celula, adica se apeleaza metoda playnehardcodat

In metoda playnehardcodat

Se verifica tipul celulei

Daca este empty exista sanse sa gaseasca monede.

Daca este Shop se afiseaza lista de potiuni si se asteapta input de la tastatura cu potiunea dorita.

Asta se continua cat timp jucatorul mai vrea sa cumpere si cat timp mai exista potiuni in shop
De asemenea, se verifica daca jucatorul are suficiente monede si suficient loc in inventar
Daca este Enemy se afiseaza intai viata mana curenta atat pentru jucator cat si pentru enemy
Cat timp amandoi sunt in viata, perand se ataca. Jucatorul alege din cele 3 optiuni: Atac
normal , abilitate sau potiune.
La enemy este ales random modul de atac
Daca jucatorul pierde se afiseaza mesajul corespunzator, daca nu creste experienta, o poate si
nivelul
Daca ajunge la Finish creste numarul de mape completate de jucator

3- startGui

- Clasa Login

Verific daca emailul exista in lista da conturi si daca parola corespunde. Daca da se apeleaza constructorul clasei Choose player, unde se afiseaza caracterele disponibile.

- Clasa ChoosePlayer

Dupa selectarea unui caracter se apeleaza constructorul clasei start unde am implementat mapa de joc

- Clasa Start

Mapa implementata are dimensiune 5x5, pozitia initiala este in coltul stanga sus, finish in coltul dreapta jos, iar shop , enemy si empty sunt pozitionati random

Jucatorul se deplaseaza prin apasarea unei casute vecine. Daca se apasa o alta casuta nu se va intampla nimic.

-Clasa ShopGui

Daca se ajunge pe o casuta unde este Shop, atunci se va deschide o noua fereastră

Aici se pot alege potiunile dorite dupa care se apasa butonul exit

-Clasa EnemyGUI

Daca se ajunge pe o casuta cu enemy, atunci incepe batalia. Primul atac ul va face jucatorul, dupa care va apasa bunonul "Pregatit pentru atacul inamicului"

Cand unul dintre cei 2 are viata 0 ,batalia se termina si fereastră se inchide

-Clasa FinishGUI

Se deschide o noua fereastră unde sunt mentionate nivelul si experienta curenta a jucatorului