**Dokumentáció**

**Szoftverfejlesztő és -tesztelő technikus szakma.**

**Vizsgaremek**

**2022**

Foxy-Maxy

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **VERZIÓ** | **DÁTUM** | **KÉSZÍTŐ** | **STÁTUSZ** | **MEGJEGYZÉSEK** |
| v0.1 | 2022. 04.14 | Csiki Akemi | Munka verzió |  |
| v0.2 |  |  | Véglegesítés |  |
| v1.0 |  |  | Végleges verzió |  |

# Dokumentum történet

# Aláírások-Résztvevők

|  |  |  |  |
| --- | --- | --- | --- |
| Szerepkör | név | dátum | aláírás |
| Csapattag | Csiki Akemi |  |  |
| Csapattag | Ladányi Bertold |  |  |
| Csapattag | Varga Dávid |  |  |
| Vizsgabiztos |  |  |  |
| Vizsgabiztos |  |  |  |
| Vizsgabiztos |  |  |  |

**Tartalomjegyzék**

[Dokumentum történet 2](#_Toc101297543)

[Aláírások-Résztvevők 2](#_Toc101297544)

[1. Projekt háttere, körülményei 3](#_Toc101297545)

[2. Projekt célja 4](#_Toc101297546)

[3. A megvalósítás során használt módszertanok meghatározása 4](#_Toc101297547)

[4. Ütemezés 4](#_Toc101297548)

[5. Rendszer érintettségek 4](#_Toc101297549)

[1. Nyeremény játék 4](#_Toc101297550)

[1.1 Fejlesztői dokumentáció 4](#_Toc101297551)

[1.1.1 Felhasznált technológiák 4](#_Toc101297552)

[1.1.2 Adatszerkezet 5](#_Toc101297553)

[1.1.3 Algoritmusok/kódok 6](#_Toc101297554)

[1.1.4 Fejlesztési lehetőségek 14](#_Toc101297555)

[1.2 Felhasználói dokumentáció 15](#_Toc101297556)

[1.2.1 Játék bemutatása 15](#_Toc101297557)

[1.2.2 Előfeltételek 16](#_Toc101297558)

[1.2.3 Letöltés 16](#_Toc101297559)

[1.2.4 Játék szabályok 16](#_Toc101297560)

[1.2.5 Futtatás 16](#_Toc101297561)

[1.2.6 Megnyitás 16](#_Toc101297562)

[1.2.7 Bejelentkezés 16](#_Toc101297563)

[1.2.8 Pontozás 20](#_Toc101297564)

[1.2.9 Nyeremény 20](#_Toc101297565)

[1.3 Tesztelés 20](#_Toc101297566)

[1.3.1 Fejlesztői tesztelés 20](#_Toc101297567)

[1.3.2 Manuális tesztelés: 22](#_Toc101297568)

[1.3.3 Statikus tesztelés 24](#_Toc101297569)

[1.3.4 Terheléses tesztelés 24](#_Toc101297570)

[1.4 Folyamatábra 25](#_Toc101297571)

[2. Összefoglalás 26](#_Toc101297572)

# Projekt háttere, körülményei

# Projekt célja

# A megvalósítás során használt módszertanok meghatározása

# Ütemezés

|  |  |
| --- | --- |
| **Szakaszok** | **Elvárt kezdés (Q1-Q4)** |
| 1. Megalapozás | 2022.Q1 |
| 1. Tervezés | 2022.Q2 |
| 1. Megvalósítás (Fejlesztés, tesztelés, élesítés) | 2023.Q2 - Q3 |
| 1. Zárás | 2023.Q3 |

# Rendszer érintettségek

|  |  |  |
| --- | --- | --- |
|  | **Rendszer érintettség** | **Igen/Nem** |
|  |  |  |
|  |  |  |

# Nyeremény játék

## Fejlesztői dokumentáció

### Felhasznált technológiák

A HardWareShop oldalhoz lefejlesztésre került egy asztali alkalmazás, amely egy akasztófa játék. Az akasztófa játék egy nyeremény játék, amellyel a felhasználok ingyenes házhozszállítást nyerhetnek a weboldalon vásárolt termékekhez. A játékba való bejelentkezés előfeltétele az hogy a felhasználó regisztráljon a weboldalon ugyanis az akasztófa játék az adatbázis user táblájával kommunikál.

#### Eszköz

A fejlesztéshez használt számítógép adatai a következő táblázatban találhatóak:

|  |  |
| --- | --- |
| Számítógép adatai | |
| Eszköz neve | DESKTOP-GMEVQ4B |
| Processzor | AMD FX (tm)-8320 Eight-Core Processor 3.50 GHz |
| Memória mérete | 8,00 GB |
| Rendszer típusa | 64 bites operációs rendszer, x64-alapú processzor |

A fejlesztéshez használt operációs rendszer adatai a következő táblázatban találhatóak:

|  |  |
| --- | --- |
| Operációs rendszer | |
| Kiadás | Windows 10 Pro |
| Verzió | 21H2 |
| Minőség | Windows Feature Experience Pack 120.2212.4170.0 |

#### Fejlesztői környezet

Az általam használt fejlesztői környezet legfőbb adati a következő táblázatban találhatók:

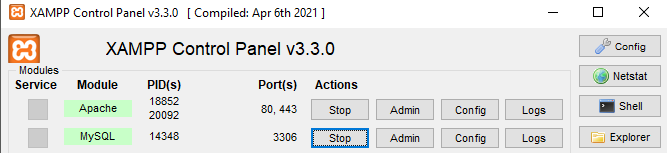
|  |  |
| --- | --- |
| Fejlesztői környezet | |
| Megnevezés | Apache NetBeans IDE 12.2 |
| Java | 11.0.14; Java HotSpot (TM) 64-Bit Server VM 11.0.14+8-LTS-263 |
| Futásidő | Java (TM) SE Runtime Environment 11.0.14+8-LTS-263 |
| Rendszer | Windows 10 version 10.0 running on amd64; Cp1250; hu\_HU (nb) |
| Felhasználói könyvtár | C:\Users\Akemi\AppData\Roaming\NetBeans\12.2 |
| Fogási könyvtár | C:\Users\Akemi\AppData\Local\NetBeans\Cache\12.2 |

### Adatszerkezet

#### Adatbázis kapcsolat

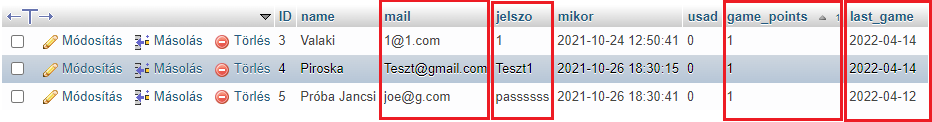
Az általam készített akasztófa játék adatkapcsolatait, Varga Dávid által készített adatbázis biztosítja.

Az adatbázis kapcsolat az XAMPP program segítségével létesül.



Az adatbázis user tábláját használja a játék. Az adatbázis megtekintésére a phpMyadmin felületet került használatra.

A piros színnel jelölt mezőket használja a játék mind olvasásra mind írásra. A user tábla felépítése a következő:



#### Java adatbázis kapcsolat

A fejlesztői környezeten belül a következő két elem került importálásra a libaries-be:

A képen szöveg látható

Automatikusan generált leírás

### Algoritmusok/kódok

#### login.java

A login.java a bejelentkezésre létrehozott JFrame, amelynek fő motívuma az adatbázissal való kommunikáció.

##### Importok:

A bejelentkezés során használt importok a következők:

import java.io.\*;

import java.sql.\*;

import java.text.DateFormat;

import java.text.SimpleDateFormat;

import java.time.LocalDate;

import java.time.format.DateTimeFormatter;

import javax.swing.JOptionPane;

import java.time.LocalDate;

import java.time.LocalDateTime;

import java.time.Period;

##### Adatkapcsolatokhoz szükséges Statementek és Connectek:

Az adatbázis kapcsolódáshoz a következő statementek és connectek kerültek beépítésre.

java.sql.Connection con = null;

PreparedStatement pst = null;

PreparedStatement stmt = null;

ResultSet rs = null;

##### Kapcsolat az adatbázissal:

Az adatbázissal való kapcsolat létrehozása egy try – catch belül kerül sor.

con = DriverManager.getConnection("jdbc:mysql://localhost/webshopdatabase","root", "");

String query = " SELECT \* FROM user WHERE mail=? and jelszo=?";

##### Beviteli mezők aktiválása:

A beviteli mezők aktiválása a PreparedStatement segítségével létesül.

st = con.prepareStatement(query);

pst.setString(1,txtemail.getText());

pst.setString(2,txtjelszo.getText());

rs = pst.executeQuery();

##### Változók létrehozása az adatbázis alapján:

Változókba mentem a beviteli mezők értékét.

String mail =(rs.getString("mail"));

String jelszo =(rs.getString("jelszo"));

##### Globális változó:

A globális változó egy külön class-ban kerül létrehozásra a login.java-ban használt mail okán.

package hangmangame;

public class Mail {

static String mail;

}

A login.java a globális változó segítségével deklarálja a mailt-t. Ellenőrzés képen kiírásra kerül az aktuális felhasználó email címe.

Mail.mail = mail;

System.out.println(mail);

##### Sikeres jelszó és email cím validáció:

Az if függvény segítségével, ha az email cím és jelszó validáció sikeres,egy try blokkban kiolvasásra kerül a felhasználónak a megfelelő mail-hez tartozó last\_game dátuma majd egy date változóba mentődik.

if(jelszo.equals(jelszo) && mail.equals(mail)){

try{

stmt = con.prepareStatement (" SELECT last\_game FROM user WHERE mail=? ");

stmt.setString(1, rs.getString("mail"));

ResultSet result=stmt.executeQuery();

Date myEndDate = null;

while(result.next())

myEndDate = result.getDate("last\_game");

##### Dátum vizsgálat:

A LocalDate import használatával a játék megvizsgálja az aktuális dátum és a last\_game dátum közötti különbséget majd ellenőrzés képpen kiírja az eltelt napok számát.

LocalDate startDate = LocalDate.now();

LocalDate endDate = LocalDate.parse(String.valueOf(myEndDate));

Period period = Period.between(startDate, endDate);

System.out.println(period.getDays());

##### Dátum vizsgálat megfelelő és nem megfelelő eredménnyel:

Egy if függvény segítségével abban az esetben, ha az eltelt napok száma megfelelő létrejön a kapcsolat az adatbázissal Statement-et használva. Hozzáad egy mai dátumot a Last\_game oszlophoz, ahol az mail = belépő mail-el. Ellenőrzés képpen a program kiírja, hogy „5 naptól 5-tel több. Játszhatsz!”. A login.java bezáródik megjelenik a játék. Ha az eltelt napok száma nem megfelelő a játék bezáródik miután tájékoztatta a felhasználót és a fejlesztőt.

f(period.getDays() > +4){

System.out.println("5 naptól 5-el több. Játszhatsz!");

stmt = con.prepareStatement (" UPDATE user SET last\_game=? WHERE mail=? ");

LocalDateTime localDateTime = LocalDateTime.now();

java.sql.Date last\_game = java.sql.Date.valueOf(localDateTime.toLocalDate());

java.sql.Time time = java.sql.Time.valueOf(localDateTime.toLocalTime());

stmt.setDate(1, last\_game);

stmt.setString(2, rs.getString("mail"));

stmt.executeUpdate();

System.out.println(last\_game);

new login().setVisible(false);

TheGame g=new TheGame();

g.setVisible(true);

JOptionPane.showMessageDialog(null, "Sikeres bejelentkezés");

}else if(period.getDays() < -4){

System.out.println("5 naptól 5-el kevesebb. Játszhatsz!");

}

}else{

System.out.println("5 naptól 5-el se több se kevesebb. Nem játszhatsz!");

JOptionPane.showMessageDialog(null, "Kevesebb mint öt napja játszottál!");

##### Sikertelen jelszó és email cím validáció:

Sikertelen bejelentkezés esetén a login felület betöltődik és a set.Text segítségével a beviteli mezőkbe újra megadható adat. Az adatbázissal történő kapcsolat lezárul.

}else{

JOptionPane.showMessageDialog(null, "Sikertelen bejelentkezés, Próbáld újra!");

new login().setVisible(true);

txtemail.setText(" ");

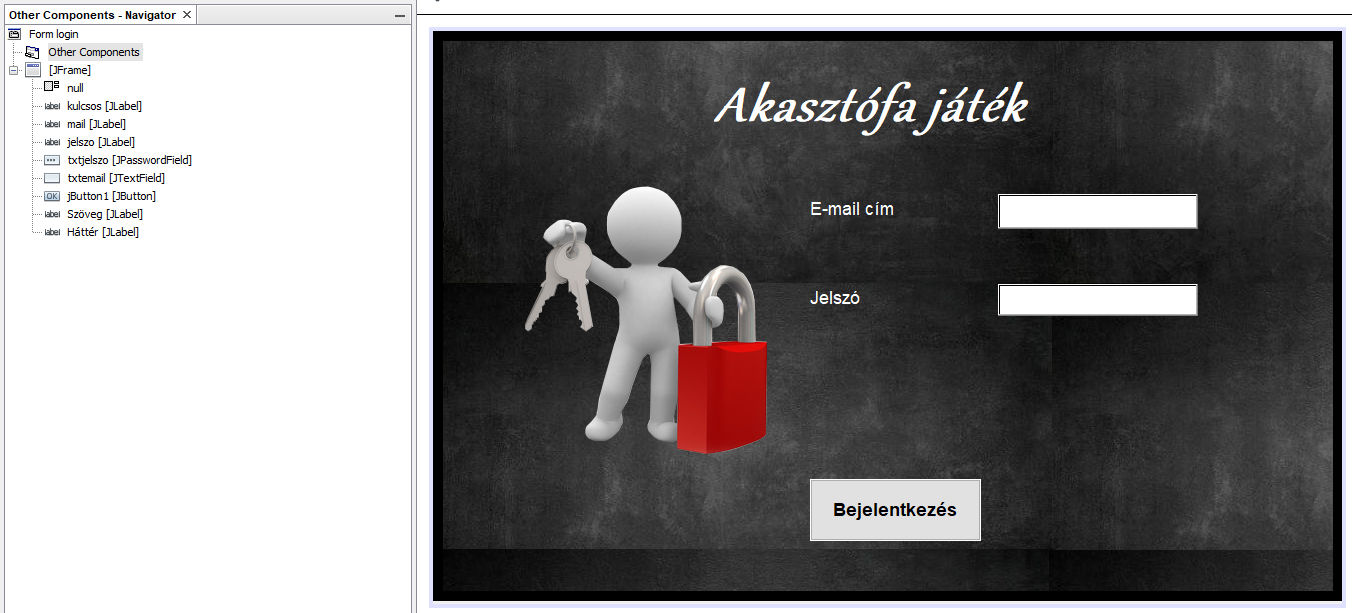
txtjelszo.setText(" ");

}

con.close();

##### Dizájn:

A login.java Dizájn része a képen látható módon került felépítésre.



#### TheGame.java

A TheGame.java a játék felületre létrehozott JFrame, amelynek fő motívumai a kérdés-válasz párok, megjelenítendő képek és az adatfeltöltés.

##### Importok:

A játékfejlesztés során használt importok a következők:

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.ArrayList;

import javax.swing.JFrame;

##### Változó:

Ineteger típusú változó létrehozása a pontok számlálásához:

int w=0;

##### Válasz megadása kérdésre:

A beviteli mezők során actionPerformed kerül használatra. A beviteli mező tartalma mentésre kerül egy string típusú változóba. Az ArrayList tömbben megadásra kerülnek a lehetséges válaszok az adott kérdésre. Egy if függvény segítségével megvizsgálásra kerül, hogy a kérdésre adott válasz helyes-e. A vizsgálat alapján megjelenítésre kerülnek a megfelelő válaszok és iconok. A megjelenítésében a setVisible került használatra. Minden helytelen válasz értéke 1. A rossz válaszok esetén a rossz válasz értéke elmentésre kerül a w változóba.

String V1=valasz1.getText();

###### Lehetséges válaszok ArrayList használata:

ArrayList<String> Jvalasz1 = new ArrayList<String>();

Jvalasz1.add("Biztonsági másolat");

Jvalasz1.add("biztonsági másolat");

Jvalasz1.add("Biztonságimásolat");

Jvalasz1.add("Biztonsagi masolat");

Jvalasz1.add("biztonsagi masolat");

Jvalasz1.add("Biztonsagimasolat");

Jvalasz1.add("BIZTONSAGI MASOLAT");

Jvalasz1.add("BIZTONSÁGI MÁSOLAT");

Jvalasz1.add("BIZTONSAGIMASOLAT");

Jvalasz1.add("Biztonsagimasolat");

Jvalasz1.add("BIZTONSÁGIMÁSOLAT");

Jvalasz1.add("biztonsági mentést");

Jvalasz1.add("Biztonságimentés");

Jvalasz1.add("Biztonsagi mentes");

Jvalasz1.add("biztonsagi mentes");

Jvalasz1.add("Biztonsagimentes");

Jvalasz1.add("BIZTONSAGI MENTES");

Jvalasz1.add("BIZTONSÁGI MENTÉS");

Jvalasz1.add("BIZTONSAGIMENTES");

System.out.println(Jvalasz1);

###### Helyes válasz megadása esetén:

if( Jvalasz1.contains(V1)){

jLab\_W.setText(""+w);

kerdes2.setVisible(true);

valasz2.setVisible(true);

valasz1.setEnabled(false);

###### Helytelen válasz megadása esetén

}else{

w=w+1;

jLab\_W.setText(""+w);

kerdes2.setVisible(true);

valasz2.setVisible(true);

valasz1.setEnabled(false);

alja.setVisible(true);

}

}

##### Megjelenítések:

A játék megkezdésekor definiálásra kerültek az elemek megjelíntése seVisible-t alkalmazva.

aember.setVisible(true);

alja.setVisible(false);

kozepe.setVisible(false);

teteje.setVisible(false);

fember.setVisible(false);

bember.setVisible(false);

kerdes2.setVisible(false);

valasz2.setVisible(false);

kerdes3.setVisible(false);

valasz3.setVisible(false);

kerdes4.setVisible(false);

valasz4.setVisible(fal

##### Kiértékelés:

Abban az esetben, ha a w1 változó értéke 0 a játék megjeleníti a ToWin.java JFrame oldalt, az adatbázisba feltöltésre kerül egy 1-es érték a megfelelő e-mail című felhasználó game\_points oszlopába, amely a weboldallal történő kommunikáció során jelöli, hogy a felhasználó nyert egy ingyenes házhozszállítást. Ellenkező esetben a ToLose.java JFrame oldal kerül megjelenítésre, nem történik adatfeltöltés. A felhasználó minden esetben kap tájékoztatást arról, hogy veszített vagy nyert-e.

int w1=Integer.parseInt(jLab\_W.getText());

###### Minden választ helyesen adott meg a felhasználó:

if(w1 == 0){

System.out.println("NYERTÉL");

ToWin tw=new ToWin();

tw.setVisible(true);

java.sql.Connection con = null;

PreparedStatement pst = null;

PreparedStatement stmt = null;

ResultSet rs = null;

try{

con = DriverManager.getConnection("jdbc:mysql://localhost/webshopdatabase","root", "");

stmt = con.prepareStatement (" UPDATE user SET game\_points=? WHERE mail=? ");

int game\_points;

game\_points= 1;

stmt.setInt(1 , game\_points );

stmt.setString(2, Mail.mail);

System.out.println("e-mail cím valid");

stmt.executeUpdate();

System.out.println(game\_points);

System.out.println("A pontszám feltöltve a megfelelő email című felhasználóhoz");

con.close();

}catch (SQLException e) {

System.out.println(e.getMessage());

}

###### Nem minden választ helyesen adott meg a felhasználó:

}else{

System.out.println("Vesztettél");

TheGame g=new TheGame();

g.setVisible(false);

g.dispose();

ToLose tl=new ToLose();

tl.setVisible(true);

System.out.println(w1);

}

this.dispose();

##### Dizájn:

A TheGame.java Dizájn része a képen látható módon került felépítésre.

A képen szöveg látható

Automatikusan generált leírás

### Fejlesztési lehetőségek

A program, amely lefejlesztésre került egy igen egyszerűnek mondható játék ugyanis 4 kérdés és válasz párból áll. Jelenleg a kérdés és válasz párok nem cserélődnek ezért a nyerési esélyek felhasználói szemszögből is magasak. Egy új weboldal számára kedvező egy olyan nyeremény játék, amellyel jó eséllyel 5 naponta lehetséges nyerni egy ingyenes házhozszállítást, viszont a nyeremény nem felhalmozható így ösztönzi a felhasználót a vásárlásra. Az akasztófa játék marketing szemszögből a weboldal aktiválást követően egy hónapig érdemes üzemeltetni a meglévő formában.

#### Játék fejlesztésének lehetőségei:

A bejelentkezés során a játék nem vizsgálja az e-mail cím során a megfelelő formátumot, azt fogadja el, ami az adatbázisban szerepel így a felhasználó nem megfelelő bevitt adatok esetén egy átfogó popup üzenet kap „Sikertelen bejelentkezés. Próbáld újra!”.

Megoldás:

Formátum vizsgáló beépítése a programba.

Eredmény:

Ezáltal a felhasználó nem megfelelő bevitt adatok esetén konkrét ppup üzenetet kapna „Nem megfelelő e-mail cím formátum”.

A program jelenleg a last\_game dátumának napjait vizsgálja az évre és a hónapra nem terjed ki.

Megoldás:

A dátumvizsgálat során az év-hónap-nap adatok vizsgálatának bővítése.

Eredmény:

Ennek a bővítésnek a hatására számlálható, hogy a felhasználó hányszor játszik az adott nyereményjátékkal a hónapban. Erre épülhet statisztika, illetve jutalmazható a felhasználó, ha a játékok száma meghaladja az x%-ot.

Jelenleg a nyeremény egy ingyenes házhozszállítás, hosszútávon sok felhasználó esetén kedvezőtlen lehet a weboldal bevételének szemszögéből.

Megoldások:

A játék megnyerését követően az ingyenes házhozszállítás helyett 5-10% kedvezmény.

Eredmény:

Sokszínűség fenntartása a felhasználok számára.

Az akasztófa játék 4 kérdés és válasz párból áll. A weboldal aktiválást követően 1 hónap elteltével a játék bővítendő vagy cserélendő a felhasználok érdeklődésének fenntartása okán.

Megoldása:

Akasztófa játék kérdés és válasz párjainak bővítése és randomítása olyan formában, hogy egy adott e-mail című felhasználó ne kaphassa ugyan azon kérdéseket.

Játék kivitelezésének újítása klasszikus akasztófa játék formájába betűk megadásával és egy aktuális kérdés megválaszolásával.

Játék dizájn eleminek lecserélése másik témába, ezáltal is megtartva a felhasználók érdeklődését.

Eredmény:

Felhasználók érdeklődésének fenttartása.

Hosszútávon maga az akasztófa téma elavulttá vállhat a felhasználok számára.

Megoldás:

Egyszerű memória játék elkészítése olyan informatikai eszközökkel, amelyek szerepnek a weboldalon.

Eredmény:

Felhasználók érdeklődésének fenttartása.

## Felhasználói dokumentáció

### Játék bemutatása

A felhasználó a HardWareShop-on történő regisztráció után letöltheti a nyeremény játékot a weboldalról, amely egy akasztófa játék. A játék 4 darab informatikai kérdésből áll a válasz megadások célja a halál által felakasztandó jelölt megmentése. A felhasználó csakis akkor tudja megmenteni a fehér emberkét, ha mind a 4 kérdésre tudja a megfelelő választ. Abban az esetben, ha mind a 4 válasz helyes a fehér emberke elfut a halál elől és a megmentésért cserébe megajándékozza a játékost egy ingyenes házhozszállítással, amit a HardWareShop oldalán lehetséges felhasználni.

### Előfeltételek

A következő táblázat szemlélteti a szükséges előfeltételek adott tevékenységekhez:

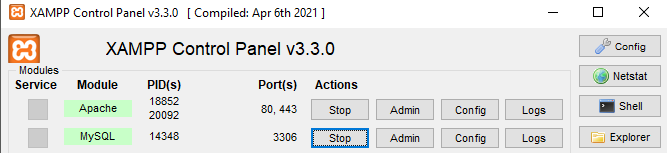
|  |  |
| --- | --- |
| **Tevékenység** | **Előfeltétel** |
| Játék letöltése | Regisztráció a HardWareShop weboldalon |
| Játék futtatása |  |
| Pontozás | Mind a 4 kérdés megválaszolása |
| Nyeremény felhasználása | Vásárlás megkezdése |

### Letöltés

### Játék szabályok

### Futtatás

A jelenlegi működés alapján a játék futtatásának előfeltétele a Xampp program futtatása:



### Megnyitás

A játék elindításához a letöltést követően dupla kattintás szükséges a letöltött .jar fájlon.



### Bejelentkezés

A játék elindul megjelenik a bejelentkezés felület:

A képen szöveg, monitor, képernyő látható

Automatikusan generált leírás

A bejelentkezéshez szükséges megadni azt az email cím és jelszó párost, amellyel a HardWareShop oldalra regisztrált a felhasználó.

Az általam megadott email cím és jelszó páros, amellyel a HardWareShop oldalán regisztráltam a következő:

Email cím: Teszt@gmail.com

Jelszó: Teszt1

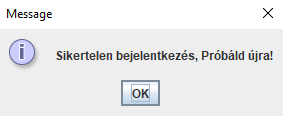
A képen szöveg, monitor, képernyő látható

Automatikusan generált leírás

A megfelelő adatok beírást követően a „Bejelentkezés” gomb megnyomása szükséges.

#### Sikertelen bejelentkezés:

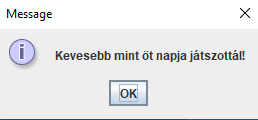
Amennyiben a megadott email cím és jelszó páros nem megfelelő megjelenik a „Sikertelen bejelentkezés. Próbáld újra!” popup üzenet.



Az „OK” gomb megnyomását követően az email cím és a jelszó újra megadható.

#### Sikeres bejelentkezés:

A játék megvizsgálja mikor játszott utoljára felhasználó, amennyiben az utolsó játék dátuma óta nem telt el 5 nap megjelenik a „Kevesebb, mint öt napja játszottál!” popup üzenet.



Mi van akkor ha több mint 5 napja játszott?

Abban az esetben, ha az utolsó játék dátuma óta eltelt 5 nap megjelenik a „Sikeres bejelentkezés” popup üzenet és az akasztófa játék.



Az „Ok” gomb megnyomását követően a felhasználó megkezdheti játékát.

#### Akasztófa játék megkezdése

A válasz megadásához a szöveg mezőbe szükséges kattintani. Majd a válasz megadását követően a billentyűzeten az „Enter” gomb megnyomása szükséges, amelynek hatásására a játék megvizsgálja, hogy a válasz helyes-e. A válasz az „Enter” gomb megnyomása után nem módosítható. Megjelenik a következő kérdés.

A képen szöveg, beltéri látható

Automatikusan generált leírás

#### Helytelen válasz:

Helytelen válasz megadását követően megjelenik az akasztófa alja és a következő kérdés. Az „Elrontott válaszok” kiírja az elrontott válaszok számát.

A képen szöveg látható

Automatikusan generált leírás

#### Helyes válasz:

Helyes válasz megadását követően megjelenik a következő kérdés.

A képen szöveg látható

Automatikusan generált leírás

Mind a 4 kérdés megválaszolása után a „Tovább” gomb megnyomása szükséges

A játék a megadott válaszok alapján kiértékeli az eredmény.

Majd megjelenik a kiértékelés alapján a megfelelő ablak (Nyertél vagy Vesztetél).

### Pontozás

A pontozást a következő táblázat szemlélteti:

|  |  |
| --- | --- |
| **Pontozás** | |
| Kérdések | Pontszám |
| 1.kérdés | 1 pont |
| 2.kérdés | 1 pont |
| 3.kérdés | 1 pont |
| 4.kérdés | 1 pont |
| Össz. Pontszám | 4 pont |

### Nyeremény

Az akasztófa játék nyereménye egy ingyenes házhozszállítás, amelyet a felhasználó a HardWareShop oldalon tud felhasználni. A nyeremény nem felhalmozható, addig amíg nem használja fel a nyereményét a felhasználó nem nyerhet ujjat.

## Tesztelés

### Fejlesztői tesztelés

A program fejlesztésének legfőbb szakaszi:

* beviteli mezők létrehozása
* adatbázis kapcsolat elkészítése
* adat kinyerés az adatbázisból
* bevitt adatok ellenőrzése
* adat feltöltés az adatbázisba
* adat ellenőrzés
* dizájn elemek megjelenése

A szakaszok segítségével a program folyamatos tesztelés alatt ált. Ellenőrző pontok kerültek beépítésre a kódba:

* System.out.printl();
* JOptionPane.showMessageDialog();
* catch (SQLException e) {

e.printStackTrace();

}

### Manuális tesztelés:

A manuális tesztek futtatásának dokumentálására a következő táblázatban található:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Típus** | **Státusz** | **Tárgy** | **Leírás** | **Elvár működés** | **Felelős** | **Kezdés dátuma** | **Befejezés dátuma** | **Funkció/felület** |
| Teszteset | Sikeres | Játék\_letöltés | Akasztófa játék telepítő fájl letöltése a HardWareShop weboldalról. | Az Akasztófa játék telepítő fájla sikeren letöltődött a számítógépre. | Csiki Akemi | 2022.04.08 | 2022.04.18 | HardWareShop weboldal |
| Teszteset | Sikeres | Játék\_indítása | .exe fájl futtatása a számítógépen. | A játék elindul megjelenik a bejelentkezés felület. | Csiki Akemi | 2022.04.08 | 2022.04.18 | Számítógép |
| Teszteset | Sikeres | Játék\_Bejelentekzés | Valid e-mail cím és jelszó megadása. | Popup üzenet "Sikeres bejelentkezés” | Csiki Akemi | 2022.04.12 | 2022.04.18 | Bejelentkezés felület |
| Negatív Teszteset | Sikeres | Játék\_Bejelentekzés | Nem valid e-mail cím és jelszó megadása. | Popup üzenet " Sikertelen bejelentkezés. Próbáld újra." | Csiki Akemi | 2022.04.12 | 2022.04.18 | Bejelentkezés felület |
| Teszteset | Sikeres | Játék\_Bejelentekzés | A játék megvizsgálja a bejelentkező last\_game dátumát és összehasonlítja az aznapi dátummal. | Az eltelt napok száma 5 nap. Megjelenik a játék felülete. | Csiki Akemi | 2022.04.12 | 2022.04.18 | Bejelentkezés felület |
| Teszteset | Sikeres | Játék\_Bejelentekzés | A játék megvizsgálja a bejelentkező last\_game dátumát és összehasonlítja az aznapi dátummal. | Az eltelt napok száma kevesebb mint 5 nap. Popup üzenet "Kevesebb mint 5 napja játszottál" Kilépés. | Csiki Akemi | 2022.04.12 | 2022.04.18 | Bejelentkezés felület |
| Teszteset | Sikeres | Játék\_Játék | Első kérdés:" " Helyes válasz megadása: | Megjelenik a második kérdés. | Csiki Akemi | 2022.04.12 | 2022.04.18 | Akasztófa játék felület |
| Negatív Teszteset | Sikeres | Játék\_Játék | Első kérdés:" " Rossz válasz megadása: | Megjelenik a második kérdés. (Rossz válaszoknál megjelenik +1 érték) | Csiki Akemi | 2022.04.12 | 2022.04.18 | Akasztófa játék felület |
| Teszteset | Sikeres | Játék\_Játék | Második kérdés:" " Helyes válasz megadása: | Megjelenik a Harmadik kérdés. | Csiki Akemi | 2022.04.12 | 2022.04.18 | Akasztófa játék felület |
| **Típus** | Sikeres | **Tárgy** | **Leírás** | **Elvár működés** | **Felelős** | **Kezdés dátuma** | **Befejezés dátuma** | **Modul** |
| Teszteset | Sikeres | Játék\_Játék | Második kérdés:" " Rossz válasz megadása: | Megjelenik a harmadik kérdés. (Rossz válaszoknál megjelenik +1 érték) | Csiki Akemi | 2022.04.12 | 2022.04.18 | Akasztófa játék felület |
| Teszteset | Sikeres | Játék\_Játék | Harmadik kérdés:" " Helyes válasz megadása: | Megjelenik a negyedik kérdés. | Csiki Akemi | 2022.04.12 | 2022.04.18 | Akasztófa játék felület |
| Teszteset | Sikeres | Játék\_Játék | Harmadik kérdés:" " Rossz válasz megadása: | Megjelenik a negyedik kérdés. (Rossz válaszoknál megjelenik 1) | Csiki Akemi | 2022.04.12 | 2022.04.18 | Akasztófa játék felület |
| Teszteset | Sikeres | Játék\_Játék | Negyedik kérdés:" " Helyes válasz megadása: Tovább gomb megnyomása | 4 jó válasz esetén megjelenik a nyertél oldal. | Csiki Akemi | 2022.04.12 | 2022.04.18 | Akasztófa játék felület |
| Teszteset | Sikeres | Játék\_Játék | Negyedik kérdés:" " Rossz válasz megadása: Tovább gomb megnyomása | 4 rossz válasz esetén megjelenik a vesztettél oldal. | Csiki Akemi | 2022.04.12 | 2022.04.18 | Nyertél/Vesztetél felület |

### Statikus tesztelés

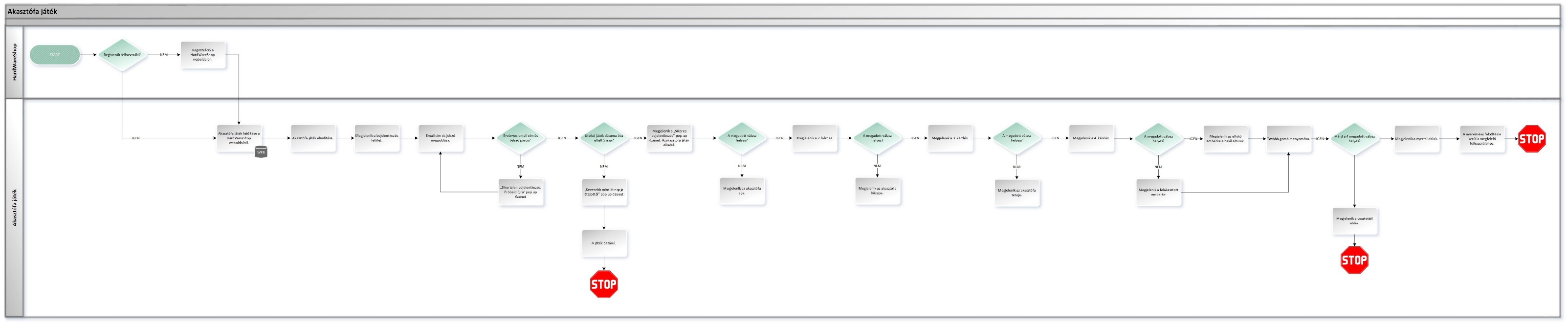
A játék lefejlesztését követően a kód ellenőrzésére került sor, amely a következő módszereken alapul:

* Kód ellenőrzés
* Clear Code elvek betartása
* Tartalmi ellenőrzés
  + - felhasználatlan obijektum
    - felhasználatlan változó
    - érték nélküli változó
    - végtelen ciklus
    - érték nélküli függvény
    - nem változó értékadás

### Terheléses tesztelés

## Folyamatábra

A következő ábrán kerül bemutatásra a játék folyamata:



# Összefoglalás