SZÉCHENYI ISTVÁN EGYETEM
GÉPÉSZMÉRNÖKI, INFORMATIKAI ÉS VILLAMOSMÉRNÖKI KAR
MATEMATIKA ÉS SZÁMÍTÁSTUDOMÁNY TANSZÉK

FÉLÉVES FELADAT

Szemüveg

Mészáros Dániel, Tóth Bence
Programtervező informatikus BSc

Győr, 2024

Tartalomjegyzék

1. Bevezetés

A jelen dokumentum célja, hogy ismertesse az általunk készített *GlassProject* nevű programot, ami a tanult módon egy általunk választott használati tárgyat, esetünkben szemüveg formáját mutatja be.

A program fő feladata egy szemüveg-szerű forma három (elölnézet, oldalnézet, felülnézet) vetületében történő bemutatása, amit a következő lépésekben tesz meg:

1. Egy szemüveg forma **Generálás**a, majd az adatainak ismertetése egy ablakban
2. A generált forma **Rajzolás**a, annak paramétereinek igény szerinti módosíthatóságával, majd a módosított verzió újra rajzoltatása.

Fontos azonban, hogy a generálás után a forma csupán a programon belül értelmezett, ezért az újbóli megjelenítéshez lehetőségünk van **Mentés**re, valamint a későbbi **Betöltés**re, amennyiben szeretnénk korábbi generált objektumainkat viszont látni.

Emellett lehetőségünk van a generált/betöltött objektum **Törlés**ére is a program memóriájából, valamint az aktuális verzió adatainak **Bemutatás**ára is.

Az objektumfájlok speciális, *.glass* formátumot használnak, hogy jól megkülönböztethetők legyenek egyéb fájltípusoktól, és biztosítsa az exkluzivitását a programhoz.

2. Tervezési dokumentáció

2.1. Fejlesztő környezet

A fejlesztés az európai berkekben általánosan használ Microsoft cég **Windows 10 Home/Pro 22H2** -es operációs rendszerén történt, az órákon is használt **IntelliJ IDEA 2024.1.2**-es verziójú Java szoftverfejlesztői környezetében.

2.2. Felhasznált technológiák

A program grafikus megjelenítéséhez az órán tanult **JFrame** megjelenítő helyett a **JavaFX** keretrendszer által használt **FXMLLoader** megjelenítőeszközt használtuk. Ez az eszköz *.fxml* típusú fájlokat használ a grafikus tartalmak kezeléséhez, amiket a hozzájuk tartozó *controller.java* állományokkal tudunk felvértezni Java kódokkal, hasonlóan a **Microsoft Visual Studio** -ban a **WPF** (Windows Presentation Foundation) programoknál.

2.3. Használt hardvereszközök

Tekintve, hogy két különböző eszközön dolgoztunk és teszteltük a program futtathatóságát, így ezek specifikációi eltérnek egymástól, bár a program futási normáit (indítási/mentési/betöltési idő) nem befolyásolták számottevően.

1. Táblázat: Futtatási hardverkörnyezet specifikációi:

Név	Mészáros Dániel	Tóth Bence
Processzor	AMD Ryzen 5 5600H	Intel Core i7-9700KF
Memória (RAM)	16.0GB	32.0 GB
Grafikus kártya	Radeon Graphics Integrált videokártya	Nvidia GeForce GTX 1660Ti
Használt meghajtó	ADATA SU800	Samsung EVO 980 Pro

2.4. A program felépítése

2.5. *Glass* osztály (modellosztály)

2.6. <u>Konstruktorok</u>

- **diameter**: a lencse átmérője, egész szám
- **lensDistance**: a lencsék közti távolság, egész szám
- **sideArmLength**: A szemüveg szárainak hossza, egész szám
- **sideArmThickness**: A szemüveg szárainak vastagsága, egész szám
- **lensThickness**: Lencsék vastagsága, egész
- **red/green/blue**: Három különböző egész számérték a színkombinációk pontosabb megadásához
- **surface**: A szemüveg megközelítő felszínértéke, aminek megállapításához az alábbi képletet használtuk:

$$\left(\left(\frac{d}{2}\right)^2 \cdot \pi \cdot lensT\right) \cdot 2 + 2(sAT \cdot 2 + sAL \cdot 4) + (lensT \cdot lensD \cdot 4 + lensT^2 \cdot 2)$$

- **volume**: A szemüveg megközelítő térfogatértéke, aminek megállapításához az alábbi képletet használtuk:

$$\left(\frac{d}{2}\right)^2 \cdot \pi \cdot 2 \cdot lensT + (sAL^2 \cdot sAT) + (sAT \cdot lensT \cdot lensD)$$

Ahol:

- *lensT, lensD*: lencsevastagság, ill. A közöttük levő távolság
- *sAT, sAL*: szárak vastagsága, ill. hossza

2.7. Főbb metódusok

- **toString** (A objektum adatainak kiirató metódusa)
  Az osztály paramétereit adja meg egy előre formázott String formájában. Visszatérési értékként átadva használatos a **Generálás**, valamint a **Bemutatás** menüpontokban.
- **calculate** (Felszín és Térfogat számoló)
  A **surface** és **volume** *setter* metódusaival használja a fentebb említett képletek szerint. Paraméterként megkapja az adott példány konstruktorait a színértékek kivételével.

2.8. Main_Controller osztály

Az osztály tárgyalása során említést teszünk azt általa kezelt *Main.fxml* fájl funkcióira is, mivel a kettőt szigorúan együtt érdemes kezelni.

2.9. Az FXML fájl elemei

Maga a fájl egy UTF-8 kódolású szövegállomány, ami a már jól ismert HTML formátumához hasonlóan egyedi ún. tag-ek alkalmazásával különbözteti meg a rajta használt elemeket. Az általunk használt elemek közül az alábbiak a legfontosabbak:

- **MenuBar**: Segítségével menücsík alakítható ki az ablak felső részén.
- **Menu/MenuItem**: A menücsík elemei és azokon belüli funkciók kialakításához használtuk. A kontrollerben **konstruktorként definiált osztálypéldányokkal** képesek vagyunk ezekre a menüpontokra **event-szerűen** hivatkozni, így metódusokat hozzárendelni.

2.10. A Main_Controller fájl

Az előbb említett FXML fájl által hivatkozott **metódusokat kezeli** és biztosítja, valamint ez kezeli a **megjelenítését** is. Meghivatkozza továbbá a modellosztályt, és tartalmazza a **mentett fájl** jövőbeli **kiterjesztését** is, ami esetünkben a *glass*.

Itt szerepel az összes kulcsfontosságú metódus, ami a program futtatásához és működéséhez elengedhetetlen, maga a Main fájl is ezt hivatkozza meg indításkor. Ezért a működés áttekintéséhez érdemes ismertetni a menüpontkezelő metódusok mindegyikét.

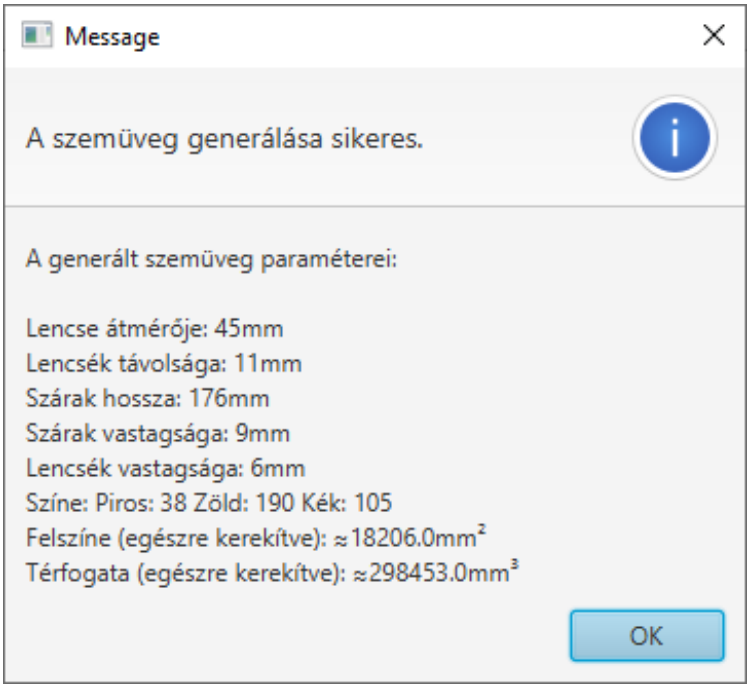
2.11. Szemüveg generálása (generateGlass)

A metódus egy új Glass példányt hoz létre, amit véletlenszerű paraméterekkel lát el. A minimálisan, illetve a maximálisan legenerálható értékek a programkódba vannak beleégetve, ami a szemmel láthatóság, valamint a megfelelő megjeleníthetőség miatt indokolt. Ezek az alábbi értékek:

- Átmérő: 30-70 mm
- Lencsék közti távolság: 10-30 mm
- Lencsék/Szárak vastagsága: 3-10 mm
- Szárak hossza: 120-200 mm

Értelemszerűen a piros, zöld, kék értékeire nincs ilyen megkötés.

Példa egy generált szemüveg adataira:

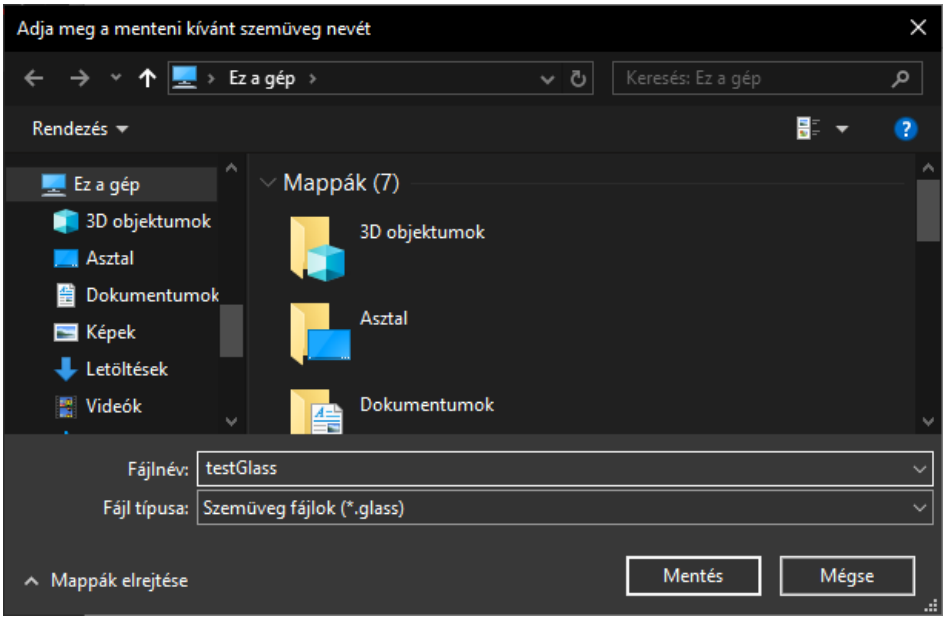


1. Kép: Szemüveg adatai (Generálás után)

2.12. Szemüveg mentése (saveGlass)

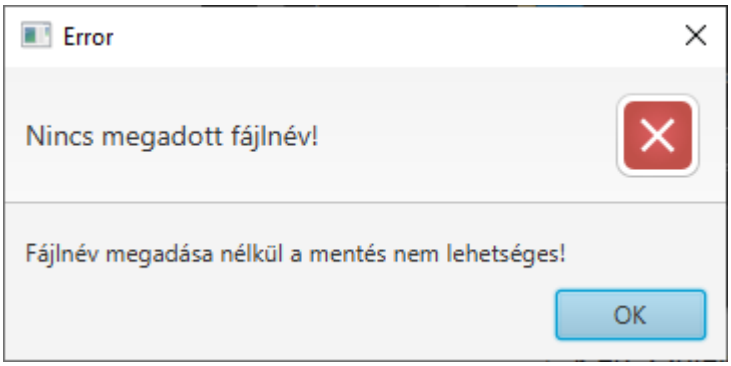
A generált példány mentése ún. *Dialog* menüvel történik, amivel a fájlnév megadása után kiválaszthatjuk a mentés pontos helyét. A kiterjesztés előre beállított, általunk megadott *.glass* formátumban,

Példa a mentés folyamatára:



2. Kép: Objektum mentése fájlba

A **Mégse** gombra kattintva, vagy az ablak bezárásakor a következő hibaüzenetet kapjuk:

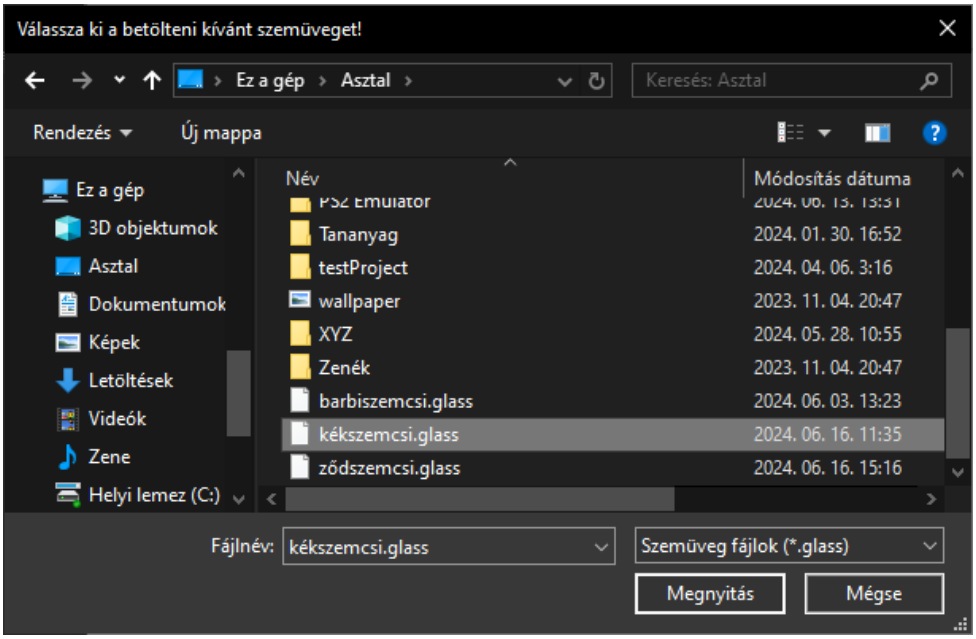


3. Kép: Hibaüzenet fájlnév hiányakor

2.13. Szemüveg betöltése (loadGlass)

Hasonlóan történik, mint a fent említett mentési folyamat, viszont itt mi jelöljük ki a választó ablakban a keresni kívánt .glass fájlunkat. A nem eredményes keresés hibaüzenettel jár.

Példa a betöltés folyamatára:



4. Kép: Objektum betöltése fájlból

A **Mégse** gombra kattintva, vagy az ablak bezárása esetén a következő hibaüzenetet kapjuk:

5. Kép: Hibaüzenet kiválasztatlan fájl esetén

2.14. A Draw_Controller osztály

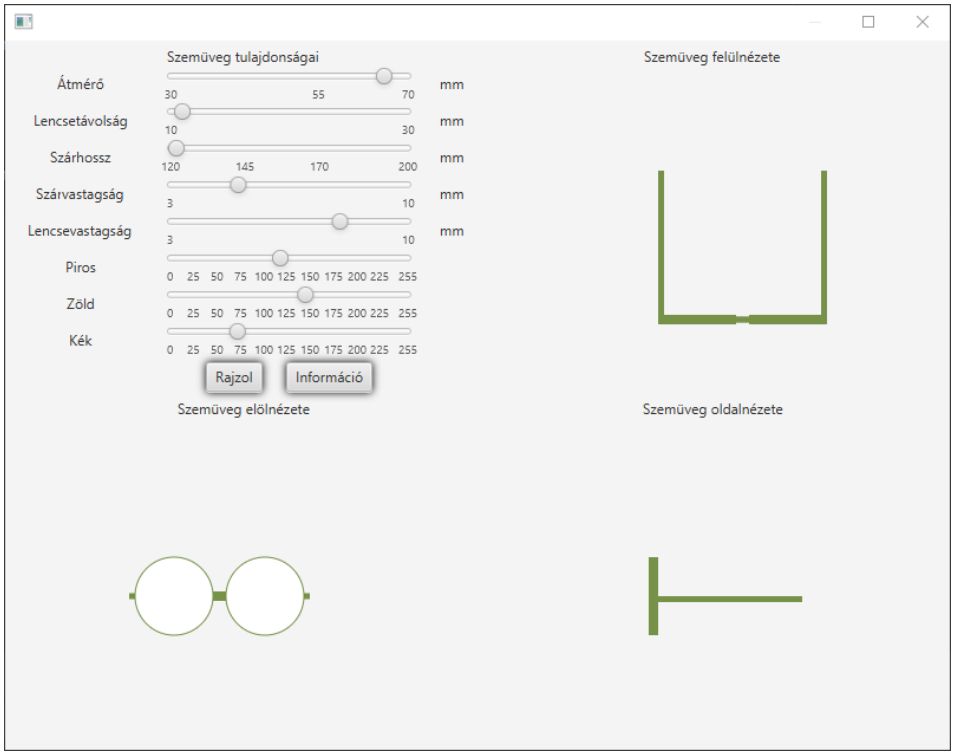
2.15. Az FXML fájl elemei

Ez a fájl valósítja meg a szemüvegünk vizuális megjelenítését és módosíthatóságát, valamint ezen funkciók hivatkozását a *controller*-en belül. A bal felső sarokban találhatóak a csúszkák, amin a szemüveg tulajdonságait tudjuk módosítani, majd ezeket a **Rajzol** gombbal ráfrissítve tudjuk látni a három kirajzolt nézetben. Az **Információ** gomb segítséget nyújt a felhasználónak az ablakon történő eligazodásban.

FONTOS!!

Az ablakot a **Generálás** funkció használata után tudjuk elérni, az ablakon belül csupán a legenerált szemüveg jellemzőinek módosítására van lehetőség!

Példa egy legenerált szemüveg megjelenítésére:



6. Kép: Szemüveg vizuális megjelenítése

2.16. A Draw_Controller fájl

Röviden említve a Java állomány a fent említett csúszkák értékállítását, az **Információ** ablak megjelenítését, valamint az átadott Glass osztály vizuális kirajzolását végzi el az általa megnyitott FXML felületen a **Rajzol** gombbal.

2.17. A szemüveg kirajzolása (drawOver, drawSide, drawFront)

Sorrend szerint először a felülnézetet, majd az oldal és szemből nézetet rajzolja ki az előre definiált ablaknegyedben.

2.18. Help_Controller osztály

Az előző osztályban tárgyalt információs gombbal aktiválható súgót, valamint a főablakban található **Segítség** súgót jeleníti meg, ami egy egyszerű szöveges fájl tölt be, és a hozzá tartozó FXML fájlban jeleníti meg.

3. Felhasználói útmutató

3.1. Hardver- és szoftverkövetelmények

- Minimális hardverkövetelmények:
  - **Processzor**: Bármilyen 2014 után gyártott processzor
  - **Memória**: operációs rendszer szerint változó
  - **Grafikus megjelenítő**: legalább 128MB VRAM -mal rendelkező integrált vagy dedikált grafikus kártya
  - **Tárhely**: legalább 140KB
- Minimális szoftverkövetelmények:
  - **Java verzió**: Java Runtime Enviroment 17, JavaFX bővítmény
  - **Windows verzió**: Windows 7 64-bites verzió
  - **Linux verzió**: bármilyen verzió támogatott

3.2. Könyvtárszerkezet, .glass fájlnév

A könyvtárszerkezet 4 mappából áll, amelyek tartalmazzák a program futásához szükséges fájlokat. Az *application* mappa a főablak és az azzal kapcsolatos elemek kezeléséhez, a *draw* az objektum kirajzolásához, a *help* a súgók megjelenítéséhez szükséges, illetve a *model* az objektum jellemzéséhez és definiálásához szükséges adatokat tárolja.

A program által generált fájlok *.glass* kiterjesztésben kerülnek mentésre, valamint az olvasandó fájlok kiterjesztésének is azonosnak kell lennie az előbb említett kiterjesztéssel.

3.3. Program szolgáltatásai röviden

Ez egy rövid, listaszerű ismertetője a program funkcióinak, bővebben lásd “A program felépítése” fejezetben.

- **Fájl:**
  Ezen menüponton belül a fájlokkal kapcsolatos opciókat érhetjük el.
  - **Kimentés:** Egy már legenerált szemüveg példány lementése fájlba.
  - **Betöltés:** Korábban lementett szemüvegpéldány betöltése a programba.
  - **Törlés:** A programba betöltött példány törlése az aktuális memóriából.
- **Létrehozás**
  Ezen a menüponton belül érhetők el a példány jellemzésére irányuló opciók.
  - **Generálás:** Létrehoz számunkra egy véletlenszerű adatokkal legenerált szemüvegpéldányt.
  - **Rajzolás:** Egy külön ablakba kirajzolja a korábban legenerált vagy betöltött szemüvegünket.
  - **Bemutatás:** A korábban legenerált vagy betöltött példányunk adatait mutatja meg.

- **Információ**
  - **Segítség**
    A főmenü súgója, leírja az egyes menüpontok funkcióit.
  - **Névjegy**
    Megjeleníti a program nevét, valamint a készítők neveit.

4. Irodalomjegyzék

- https://www.javatpoint.com/javafx-tutorial
- Youtube: Bro Code JavaFX GUI Full Course