

WireWorld dokumentáció

Felhasználói kézikönyv:

A program elindítása után ez az ablak fogad minket:



Itt az első „Magassag” címkével ellátott rubrikában tudjuk megadni a tervezőrács kívánt magasságát.

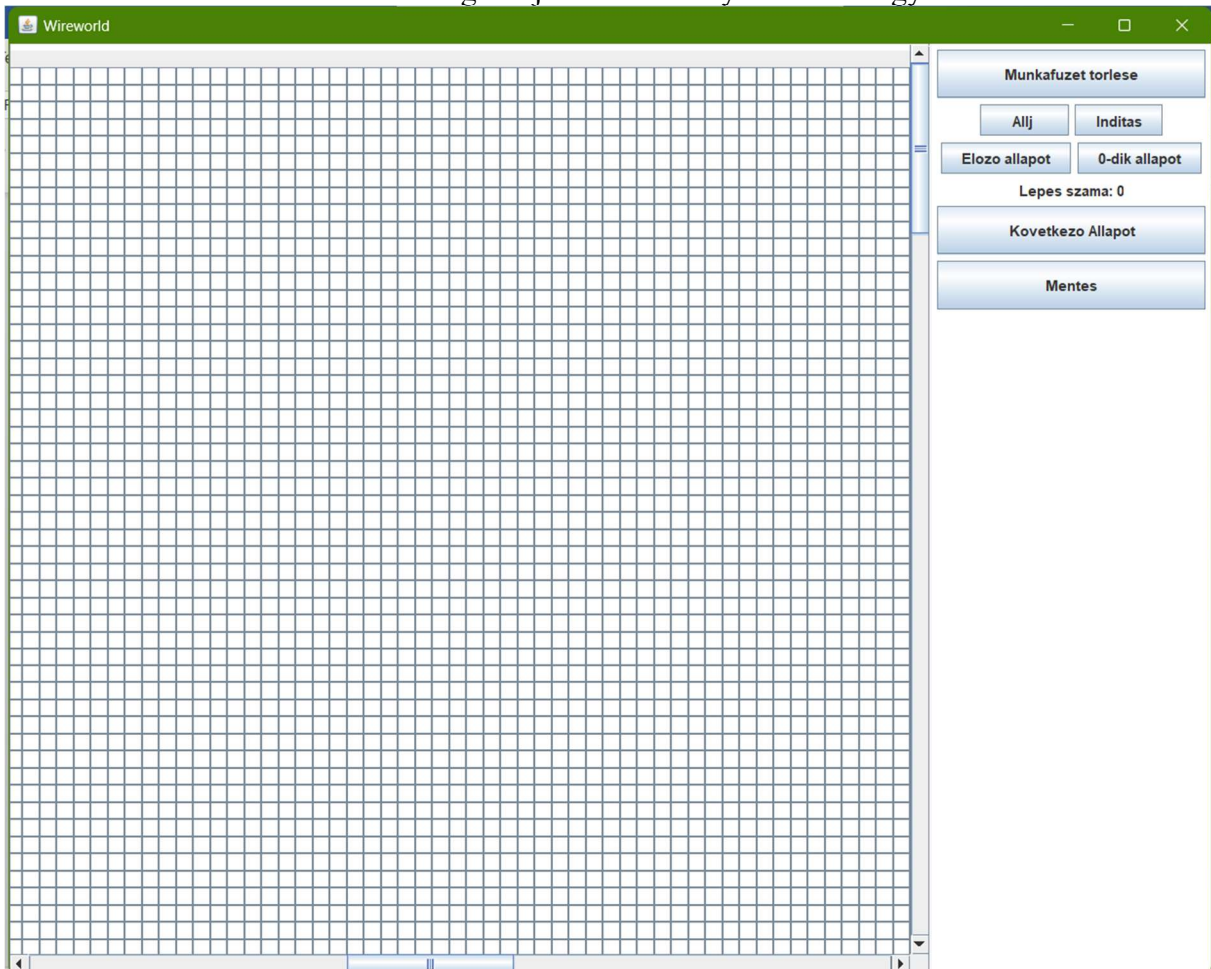
A második „Szelesseg” címkével ellátott rubrikában tudjuk megadni a tervezőrács kívánt szélességét.

A bal alsó sarokban található „Kesz” címkéjű gomb elindítja a tervezőrács létrehozását.

Alul középen a „Betoltes fajlbol” címkéjű gomb elindítja a tervezőrács létrehozását, a korábban mentett fájlból.

A jobb alsó sarokban található „Segitsegi!” címkéjű gomb elindítja a tervezőrács létrehozását, amiben megtalálhatóak az előre elkészített logikai kapuk tervei.

A következő ablak amit az alsó sor gombjainak valamelyike indít el így néz ki:



Itt jobb oldalt találhatóak a vezérlő gombok (fentről lefelé és jobbról balra haladva):

1. „Munkafuzet torlese”: A tervezőrács minden celláját üres (fehér) állapotba állítja
2. „Allj”: Megállítja az elindított szimulációt
3. „Indítás: Elindítja a szimulációt
4. „Elozo állapot”: Visszalép egy állapotot
5. „0-dik állapot”: Az indítás előtti legutolsó állapotba állítja a tervezőrácsot
6. „Kovetkezo állapot”: Egy állapotot lép előre
7. „Mentes”: Elmenti egy fájlba a tábla jelenlegi állapotát, ezt majd az előző pontban említett „Betoltes fajlbol” címkeű gombbal tudjuk megtekinteni (ehhez újra kell indítanunk a programot)

Az ablak jobb oldalán a gombok között található egy számláló, ami az éppen megjelenített állapot számát írja ki.

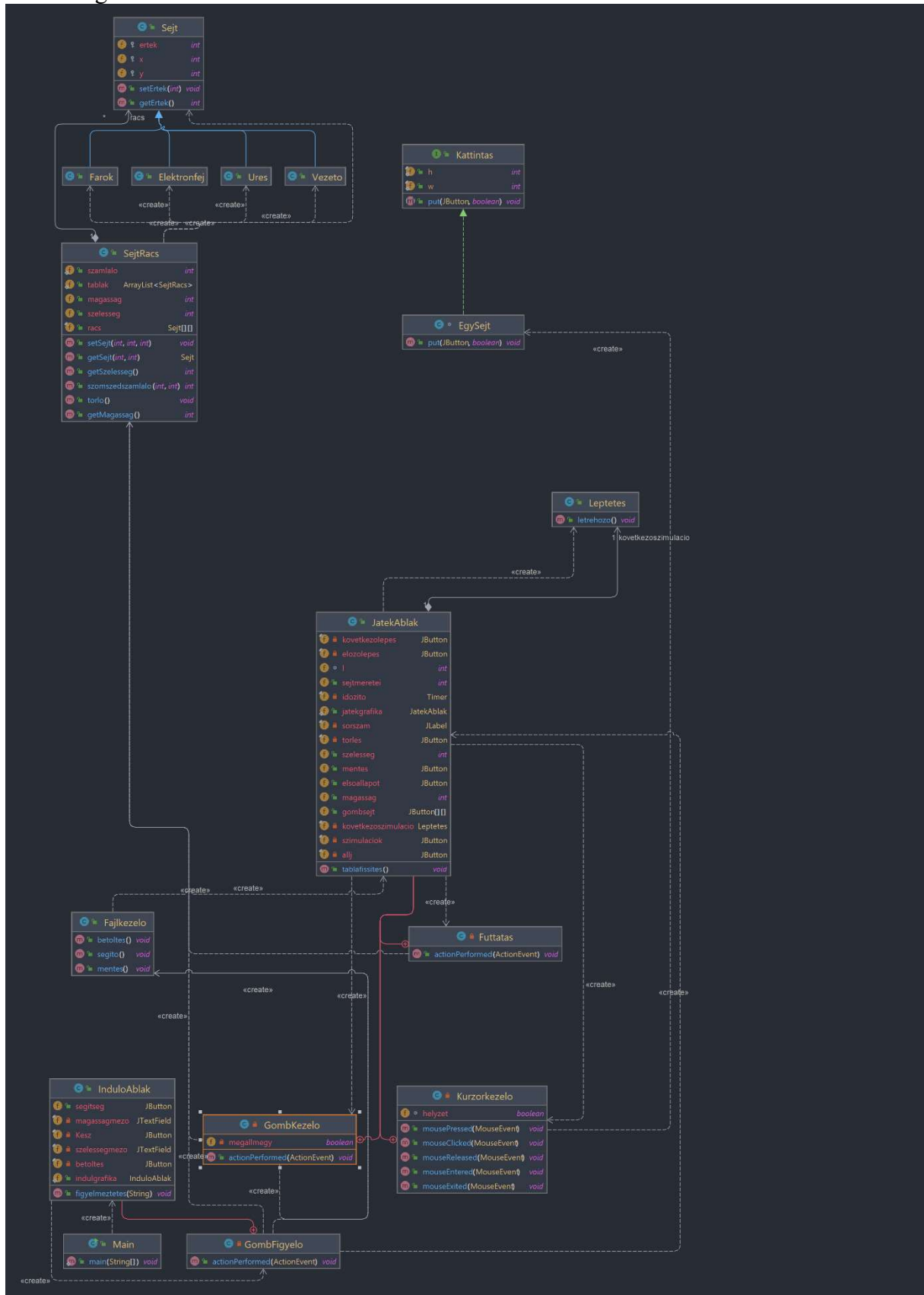
Az ablak bal oldala a tervezőrács, ezen minden rubrika kattintható. Ha nem látszik belőle az egész, akkor az ablak alján, és a tervezőrács jobb oldalán (a gombok bal oldalán) elhelyezett görgőkkel tudjuk mozgatni a tervezőrácsot. Minden rubrikának 4 állapota van, ezek az adott rubrikára való kattintással változtathatóak, minden kattintás eggyel változtatja az állapotát 1-4 között:

1. Fehér: Üres
2. Fekete: Vezető
3. Narancssárga: Elektron farok
4. Piros: Elektron fej

A szabályok:

- 0->0 Az üres üres marad
- 1->2 Az elektron halad a vezetéken és a fejből farok lesz
- 2->3 Az elektron tovább halad és a farok is eltűnik, helyette újra vezetékek lesz
- 3->1 Ha pontosan 1, vagy 2 szomszédos elektronfej van a cella mellett, emiatt tud az elektron haldni a vezetéken

UML diagramm:



A függvények dokumentációi a csatolt JavaDoc nevű mappában a classok nevével ellátott javadocban van.