

Sakk program

Felhasználói kézikönyv:

A program indításakor megjelenik egy gomb, amivel el lehet indítani a játékot.

A játékmód kiválasztása után egyből kezdődik a játék. (A fehér bábuval kell lépni először.) Ha a soron következő játékos kitalálta melyik bábuval szeretne lépni, akkor annak a bábunak a mezőjére kell az egérrel klikkelnie. Ekkor kijelölte az adott bábut. Ezután a bábu célmezőjére kell klikkelnie, hogy az adott lépést megvalósítsa. Ha egy másik bábut akar választani elég, ha a arra a bábura kattint. A játékmű mellett a program tartalmaz egy feladás gombot, amit az éppen soron következő játékos nyomhat meg lépés helyett. Ezzel jelzi, hogy feladta a játszmát. (Ilyenkor a másik játékos győzött.)

A játékmű fölött van egy számláló, ami a játék kezdete óta eltelt másodperceket számolja. (Ez minden új játéknál 0-tól indult.)

A játék vége után megjelenik egy záró oldal, ami megmutatja az eddigi meccsek eredményeit, és 3 gombot. Jobbról az első gomb megnyomásával elmenthető a legutóbbi játék, a középsővel új játék kezdhető és az utolsóval ki lehet lépni a programból.

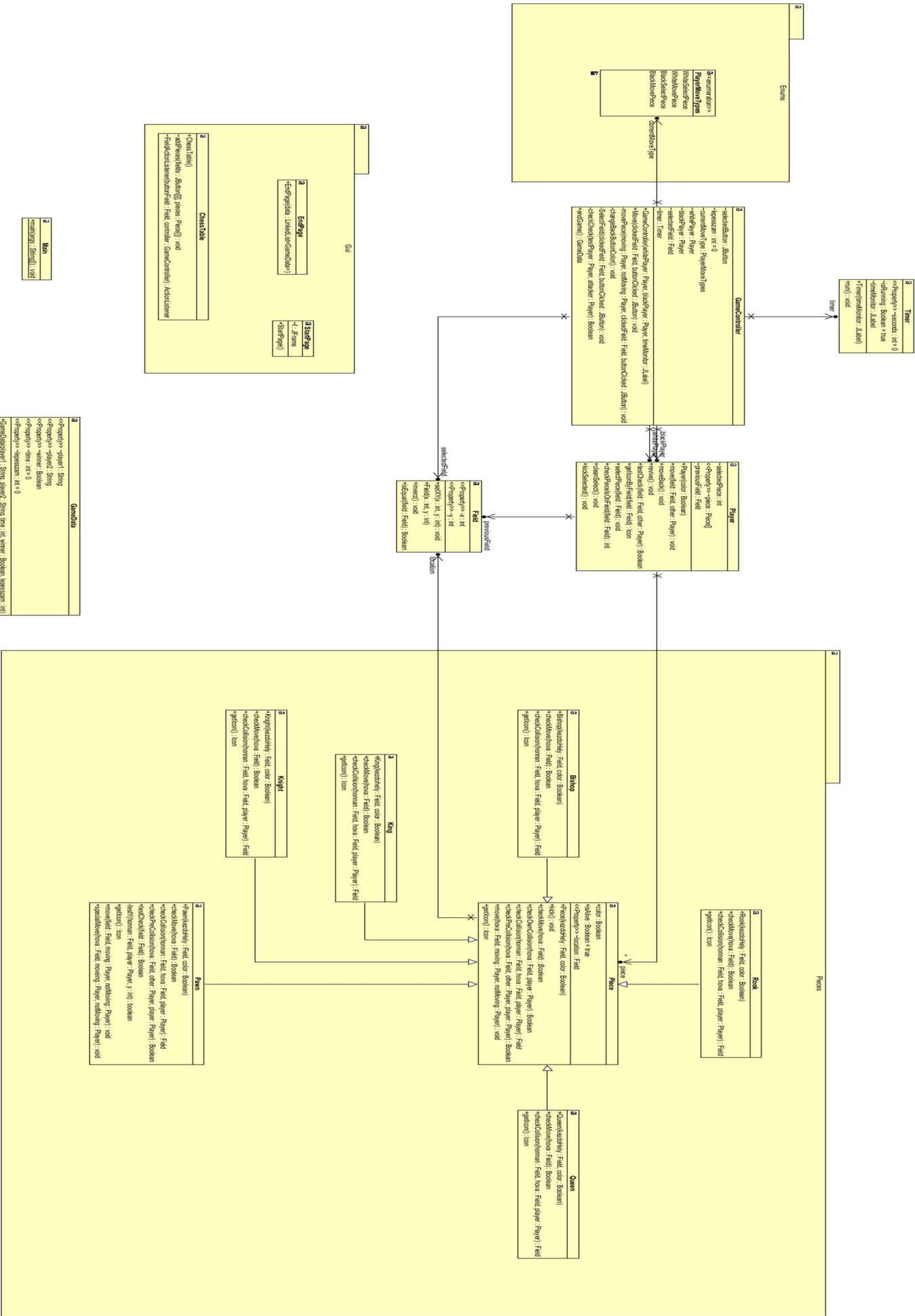
A játék menete

A játéktér egy 8x8-as tábla felváltva fekete fehér mezőkkel. Bizonyos mezőkön bábuk állnak. A bábuk csak adott mezőkre léphetnek. 2 játékos játszhat egy időben. Az egyik a fehér színű bábukot mozgatja, a másik a fekete színűeket. A játékot a fehér bábuk vezérlő játékos kezdi. A bábuk a normál sakk játékszabályai szerint lehet mozgatni. (A <http://www.sakkpalota.hu/index.php/hu/sakk-top/sakkszabalyok> oldalon részletesen is olvashatók.) Ha az egyik játékos a bábuját a másik játékos egyik bábuja mezőjére lépteti, akkor kiütötte a pályáról az ellenséges bábut. (Saját bábut nem lehet kiütni.) A játékosok felváltva lépnek, amíg a játék véget nem ér. Egy király sakkban van, ha az ellenfél egyik bábuja támadja. Ha az egyik játékos királya sakkot kapott, akkor annak a következő lépésében meg kell szüntetnie azt.

A játék a következő okok miatt érhet véget.

- Sakk-matt
 - Ekkor a soron következő játékos királya sakkban van, de a sakkot nem tudja egy lépésben megszüntetni. Az a játékos nyert, amelyik a sakkot adta.
- Patt
 - Az egyik játékos nem tud már lépni, de egyik király sincs sakkban. Az eredmény ekkor döntetlen.
- Feladás
 - Bármely játékos bármikor dönthet úgy, hogy feladja a mérkőzést. Ekkor értelemszerűen a másik játékos győzött.

Osztálydiagram:



Metódusok:

1. ChessTable
 - a. ChessTable
 - i. Az osztály konstruktora.
 - ii. Létrehozza a sakktáblát, és elindítja a játék logikát.
 - iii. A tábla mérete alaphoz 500x500, de ez a felhasználó által átméretezhető.
 - iv. A sakkmező gombokból áll. Felette van egy időjelző szöveg. Mellette pedig egy feladás gomb.
 - b. addPieces
 - i. Hozzáadja a játékmezőhöz a bábuk ikonjait. 16db bábut tud hozzáadni.
 - ii. @param fields A tömb, amiben a játékmező gombjait tároljuk.
 - iii. @param pieces Egy tömb, amiben a hozzáadandó bábuk vannak. Mérete 16 kell, hogy legyen
 - c. FieldActionListener
 - i. Egy mező gombjának a lenyomás figyelőjét hozza létre.
 - ii. @param buttonField az adott gomb által reprezentált mező
 - iii. @param controller a játék controllere (minden mező ugyan azt a referenciát kell, hogy kapja
 - iv. @param controller a játék controllere (minden mező ugyan azt a referenciát kell, hogy kapja
2. PlayerMoveTypes
 - i. enum, ami a lépési lehetőségeket jelöli
 - b. WhiteSelectPiece
 - i. A fehér választja ki a bábiját.
 - c. WhiteMovePiece
 - i. A fehér mozgatja a kiválasztott bábut.
 - d. BlackSelectPiece
 - i. A fekete választja ki a bábiját.
 - e. d. BlackMovePiece
 - i. A fekete mozgatja a kiválasztott bábut.
3. StartPage
 - a. StartPage
 - i. Létrehozza a start menüt. Mérete fix.
4. EndPage
 - a. EndPage
 - i. Létrehozza a játék utáni ablakot.
 - ii. @param data Az eddigi játékok adatai, amit meg fog jeleníteni.

5. Timer

- a. Timer
 - i. A timer konstruktora
 - ii. @param timeMonitor Annak a JLabelnek a referenciaja, ami az idot mutatja
- b. getSeconds
 - i. @return Visszaadja az inditas ota eltelt idot
- c. run
 - i. Szamolja az inditas ota eltelt idot, es kiirja a megadott JLabelere

6. Main

- a. main
 - i. A program kezdőpontja.
 - ii. Megnyitja a start menüt.

7. Player

- a. Player
 - i. A játékos konstruktora. Beállítja a bábukat (szineit/helyzetüket)
 - ii. @param color Ha igaz akkor a játékos fehér színű bábukat tartalmaz. Ha hamis akkor feketéket.
- b. getPiece
 - i. @return Visszaadja a játékos bábuinak referenciáját.
- c. move
 - i. Elmozgatja a kiválasztott bábút egy adott mezőre.
 - ii. Csak akkor mozgat, ha van kiválasztva bábu.
 - iii. Elmenti a bábu mozgatás előtti helyzetét.
 - iv. @param field A mozgatás utáni mező értéke
 - v. @param other A másik játékos referenciája.
 - vi. @throws Exception Ha nincs kijelölve bábu, akkor hibát dob.
- d. moveBack
 - i. Visszaállítja a bábút egy mozgatással korábbra
- e. revive
 - i. Feléleszti a kijelölt bábút.
- f. testCheck
 - i. Megnézi, hogy a játékos tudna-e ütni egy adott mezőt.
 - ii. @param field A mező, amit tesztelni szeretnénk.
 - iii. @param other A másik játékos referenciája.
 - iv. @return Ha tudjuk ütni, akkor igaz, különben hamis.
 - v. @throws Exception
- g. getIconByField
 - i. Visszaadja egy adott mezőn lévő bábu Iconját
 - ii. @param field A kiválasztott mező
 - iii. @return Az icon
 - iv. @throws Exception Ha nincs bábu a mezőn hibát dob.
- h. selectPiece
 - i. Egy bábút választ ki, mező alapján.
 - ii. @param field A mező amiről a bábút akarjuk választani.
- i. checkPiecesOnField
 - i. Megnézi, hogy áll-e saját (aktív) bábu egy adott mezőn.

- ii. @param field A mező, amit nézünk.
 - iii. @return A bábu sorszáma, ha van bábu az adott mezőn, különben
- j. cleanSelect
 - i. Töröljük a bábu kiválasztást
- k. kickSelected
 - i. Töröljük a kiválasztott bábút.
- l. checkMatt
 - i. Ellenőrzi, hogy kaptunk-e mattot.
 - ii. @param other Az ellenséges játékos referenciája
 - iii. @return Igaz, ha mattot adtunk, hamis, ha nem
- 8. GameController
 - a. GameController
 - i. A játék mechanikát indítja el.
 - ii. @param whitePlayer Fehér játékos referenciája
 - iii. @param blackPlayer Fekete játékos referenciája
 - iv. @param timeMonitor Az idő mutató referenciája
 - b. Move
 - i. Ha egy mezőre klikkelt a felhasználó, akkor aktiválódik.
 - ii. Vagy kiválaszt egy bábút, vagy egy már kiválasztott bábút mozgat.
 - iii. @param clickedField A mező értéke amire klikkelt a felhasználó
 - iv. @param buttonClicked A gomb, ami aktiválódott
 - c. movePiece
 - i. Elvégzi a kiválasztott bábu mozgatását, vagy egy új bábút választ ki.
 - ii. Ha egy ellenséges bábu helyére lépünk kiüti azt.
 - iii. Nem lehet úgy lépni, hogy utána a király sakkban legyen.
 - iv. Ellenőrzi a sakk-mattot.
 - v. @param moving A lépést végrehajtó játékos referenciája
 - vi. @param notMoving A lépést csak néző játékos referenciája
 - vii. @param clickedField A kiválasztott mező értéke
 - viii. @param buttonClicked A megnyomott gomb referenciája
 - d. changeBackButtonColor
 - i. Visszaállítja a kiválasztott mező színét az eredeti színre. (Fekete vagy fehér.)
 - e. SelectField
 - i. Kiválasztjuk, hogy melyik bábuval akarunk lépni.
 - ii. @param clickedField A lenyomott mező értéke.
 - iii. @param buttonClicked A lenyomott mező gomja.
 - f. checkCheck
 - i. Ellenőrizzük, hogy a király kapott-e sakkot.
 - ii. @param testPlayer Annak a játékosnak a referenciája, amelyik királyát akarjuk nézni.
 - iii. @param attacker A másik játékos referenciája.
 - iv. @return Ha igaz, akkor a király sakkban áll.
 - v. @throws Exception
 - g. endGame
 - i. Befejezi a játékot, leállítja az órát.

9. Field

- a. getX
 - i. @return Visszaadja az x értékét
- b. getY
 - i. @return Visszaadja az y értékét
- c. setXY
 - i. Atallítja az x, y értéket egy új értékre.
 - ii. @param x Az x új értéke
 - iii. @param y Az y új értéke
- d. Field
 - i. A Field osztály konstruktora
 - ii. Beállítja az x, y értékét.
 - iii. @param x Az x értéke
 - iv. @param y Az y értéke
- e. inverz
 - i. Megforgatja egy mező helyzetét a sakktáblára átlósan tükrözve.
- f. isEqual
 - i. 2 Mező értékét hasonlítja össze
 - ii. @param field A mező, amihez össze akarjuk hasonlítani.
 - iii. @return Igaz, ha a 2 mező helye megegyezik, különben hamis

10. GameData

- i. Ez menti le egy játék adatait. (Lépésszám, játékidő győztes szín.)
- b. GameData
 - i. konstruktor

11. Piece

- a. Piece
 - i. A bábu konstruktora
 - ii. @param kezdóHely A bábu kezdőhelyét állítja be
 - iii. @param color Ha igaz a bábu fehér színű, különben fekete.
- b. kick
 - i. A bábút deaktiválja
- c. checkMove
 - i. Abstrakt metódus. Ellenőrzi, hogy egy adott mezőre léphet-e a bábu egy üres táblán.
 - ii. @param hova A mező amire lépni akarunk.
 - iii. @return igaz, ha lehet az adott mezőre lépni; hamis, ha nem
- d. checkOwnCollision
 - i. Megnézi, hogy van-e az útban bábu.
 - ii. A kiindulási hely a bábu jelenlegi helyzete.
 - iii. @param hova A mező amire lépni akarunk.
 - iv. @param player A játékos akinek a bábuiba nem akarunk ütközni.
 - v. @return Igaz, ha nincs bábu az útban, hamis ha van.
- e. checkCollision
 - i. Megnézi, hogy ha egy adott helyről, egy adott helyre beleütközünk-e egy adott játékos bábujába.
 - ii. @param honnan A kiindulási hely.
 - iii. @param hova A cél hely.

- iv. @param player A játékos akinek a bábuiba nem akarunk ütközni.
- v. @return Ha van az útban bábu, akkor a bábu helyzetét adja vissza, különben null-t.
- f. checkPreCollision
 - i. Ha ki akarunk ütni egy ellenséges bábút, akkor meg kell néznünk, hogy az oda vezető úton a bábu előtti mezőig el tudunk-e jutni ellenséges bábuba ütközés nélkül.
 - ii. @param hova A mező amin az ellenséges bábu helyezkedik.
 - iii. @param other A másik játékos referenciája
 - iv. @return Igaz, ha ki tudjuk ütni, hamis ha nem.
 - v. @throws Exception
- g. setLocation
 - i. Beállít egy új helyet
 - ii. * @param hova Az új hely értéke
- h. move
 - i. Ellenőrzi a lépést, majd lépteti a bábút.
- i. getLocation
 - i. Visszaadja a bábu helyét.
- j. getIcon
 - i. Visszaadja a bábuhoz tartozó ikont. (absztrakt)
- 12. Bishop, King, Knight, Queen, Rook osztályok megvalósítják a Piece osztályok valósítják meg a Piece osztály absztrakt metódusait, úgy, hogy, csak az adott bábu csak a típusának megfelelő lépéseket tudja végrehajtani.
- 13. Pawn
 - i. Speciális lépése van, ezért speciális metódus kell hozzá.
 - ii. A checkPreCollision nem használható nála.
 - iii. A checkCollision módosítva van, mert nem a lépésének megfelelő irányba tud ütni, hanem sréhen.
 - iv. move függvény sem használható a fenti ok miatt.
 - b. testCheck
 - i. Mivel sréhen üt, ezért külön kell vizsgálni, hogy a királyt tudja-e támadni.
 - c. specialMove
 - i. Megvalósítja a paraszt lépéseit