

KIV/UPG

Semestrální práce

Hra šachy v Java

Alimzhan Mukanov A22B0388P

26.3.2023

Zadání:

Část 1: Základní vizualizace (až 10 bodů)

Základní funkční požadavky (7 bodů): Po spuštění programu pomocí alespoň jedním z příkazů `Run. cmd` nebo `./run. sh` (případný nefunkční druhý skript při odevzdání z projektu smažte), se zobrazí okno o minimální počáteční velikosti 800×600px. V okně se vykreslí Čtvercová šachovnice, která bude na středu okna a bude zabírat maximální možný prostor tohoto okna. Velikost okna půjde libovolně měnit a po změně velikosti okna program na tuto akci zareaguje (překreslí okno tak, aby bylo vše korektně zachováno). Na šachovnici se dále vykreslí jednotlivé kameny v zahajovací pozici dle pravidel šachu. Kameny budou vykresleny vektorově a bude rozpoznatelné, který kámen je který.

Další požadavky: Kámen půjde pomocí Drag & Drop přesunout na libovolné jiné pole. Pokud dojde k posunu na již obsazené pole, původní kámen bude odstraněn. Součástí odevzdání bude kompletní dokumentace dle formátu poskytnutém vyučujícími.

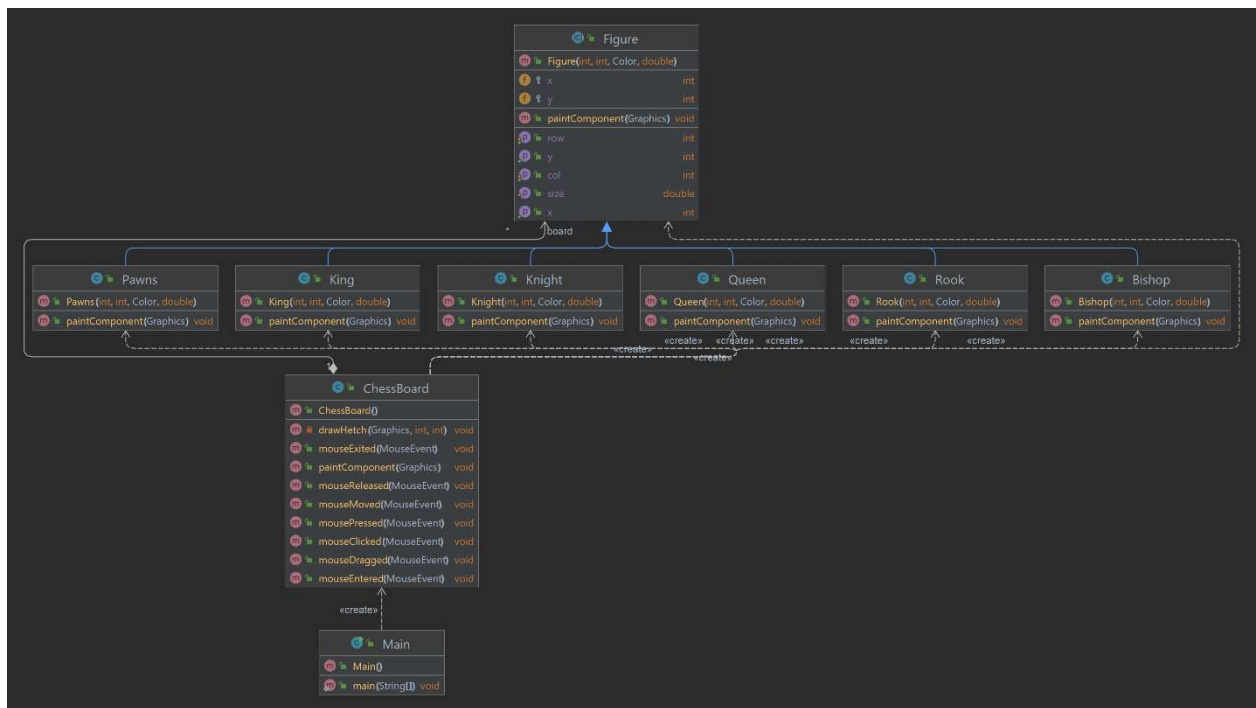
Řešení:

Pro splnění první části práce bylo vytvořeno několik tříd pro různé funkce. První třída, **Main.java**, je určena pro spuštění programu a nastavení velikosti na 800x600px. Druhou třídu, **ChessBoard.java**, jsem vytvořil pro vizualizaci šachovnice, figur a implementaci funkce **Drag & Drop**.

Pro změny velikosti figur a šachovnice se inicializuje **componentListener** s metodou **componentResize()**. Tato inicializace reaguje na změnu rozměru okna a změny velikost kamenů a šachovnice.

Třidy:

- **UML Diagram tříd:**



- **Main.java**

Je hlavní třída projektu, která spustí aplikaci. Metoda **main** nastaví velikost okna, umístí okno do středu obrazovky.

- **ChessBoard.java**

Je hlavní třída projektu. Dědí od **JPanelu** a implementuje **MouseListener** a **MouseMotionListener** pro práci s Drag and Drop funkcionalitou. Třída kreslí šachovnici a figurky. Šachovnice je bílo-šedá a figurky jsou kresleny v jiných třídách.

- `paintComponent(Graphics g)`
vykreslí šachovnici a figurky na ní.

- **Figure.java**

Je abstraktní třída a hlavní třída pro kreslení figur. Dědí od **JPanelu**.

- **Queen.java, King.java, Knight.java, Bishop.java, Rook.java, Pawns.java**

Tyto třídy představují implementaci jednotlivých figurek. Každá třída dědí od **Figure**.

Uživatelská dokumentace:

- **Instalace:**

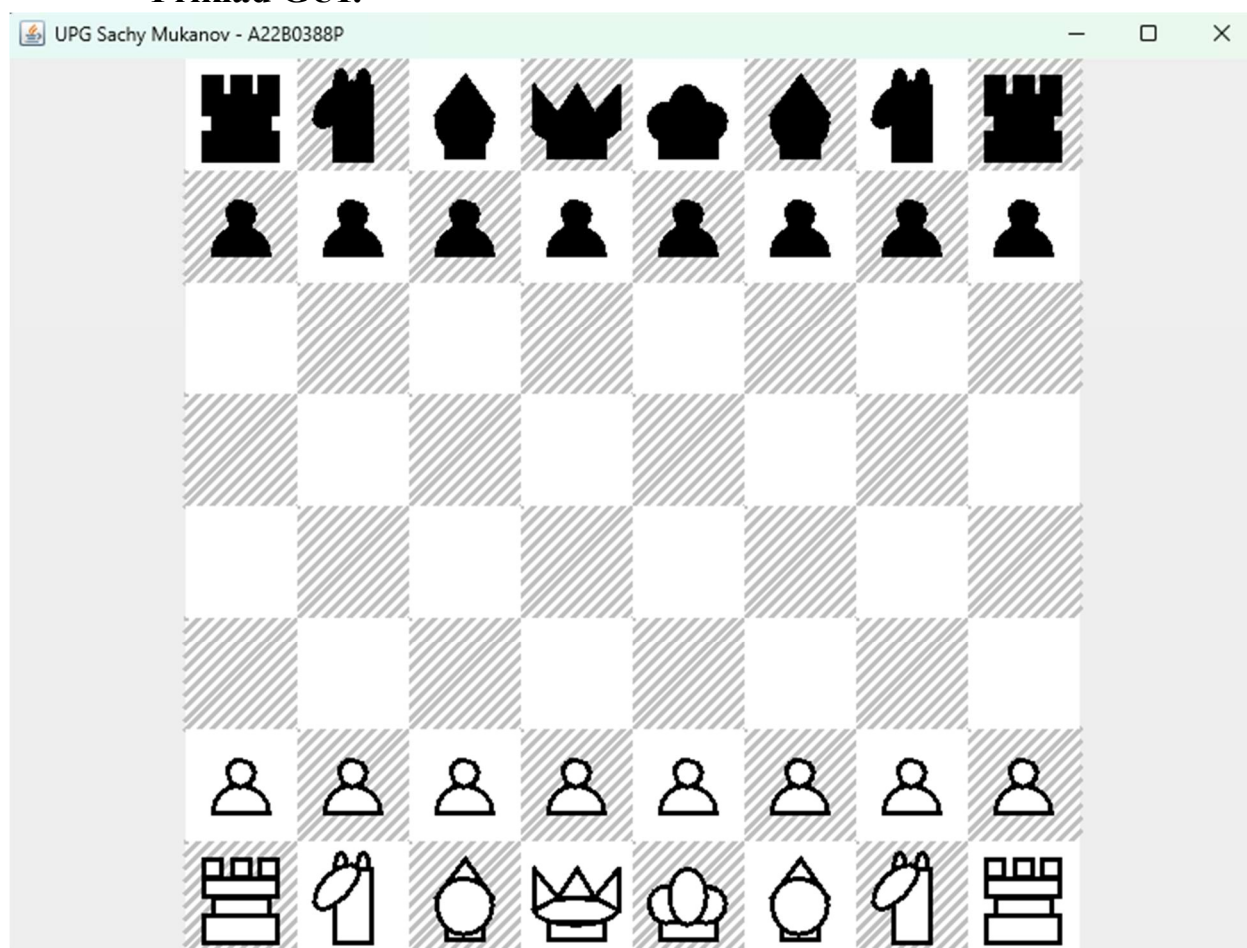
1. Překlad programy pomoci spuštění **Build.cmd** ze složky semestrální práce

bin	✓	26.03.2023 18:18	File folder	
src	✓	26.03.2023 17:56	File folder	
Alimzhan Dokumentace.pdf	✓	26.03.2023 15:16	Firefox PDF Docu...	780 KB
Build.cmd	✓	26.03.2023 18:17	Windows Comma...	1 KB
Run.cmd	✓	26.03.2023 18:13	Windows Comma...	1 KB

2. Spuštění aplikace pomoci spuštění **Run.cmd** ze složky

bin	✓	26.03.2023 18:18	File folder	
src	✓	26.03.2023 17:56	File folder	
Alimzhan Dokumentace.pdf	✓	26.03.2023 15:16	Firefox PDF Docu...	780 KB
Build.cmd	✓	26.03.2023 18:17	Windows Comma...	1 KB
Run.cmd	✓	26.03.2023 18:13	Windows Comma...	1 KB

- **Příklad GUI:**



Závěr:

Tento projekt je program pro hraní šachů s využitím knihovny **Java Swing** a principů **OOP**. Program představuje vizuální simulaci herní desky, na které jsou umístěny figury, které se mohou pohybovat.

Funkce programu zahrnují možnost změny velikosti okna, přičemž šachovnice a figurky se přizpůsobují velikosti okna.

Program je určen pro uživatele, kteří chtějí hrát šachy bez pravidel. V budoucnu budou přidána pravidla hry.

Při práci jsem se naučil používat knihovnu **Jawa Swing**.