

SPŠE Ječná

Obor Informační Technologie

Střední průmyslová škola elektrotechnická, Praha 2, Ječná 30

Virtual Piano

Daniel Matoušek

IT

2025

Obsah

1. Cíl práce.....	3
2. Popis programu.....	3
3. Systém requirements.....	4
4. Základní struktura.....	4
5. Testovací data.....	4
6. Uživatelská příručka.....	5
7. Závěr.....	5

1 Cíl práce

Cílem mého projektu bylo vytvořit jednoduchou aplikaci ve formě virtuálního klavíru, která uživateli umožní hrát tóny pomocí běžné počítačové klávesnice. Aplikace simuluje chování hudebního nástroje, kdy se při stisknutí určité klávesy přehraje odpovídající zvukový soubor.

2 Popis práce

Aplikace Virtual Piano je tvořena v jazyce **Java** s využitím knihovny **Swing** pro tvorbu grafického uživatelského rozhraní a balíčku **javax.sound.sampled** k přehrávání zvukových souborů.

2.1....Hlavní prvky GUI

- **Virtuální klávesy:** Bílé a černé klávesy odpovídající skutečnému klavíru. Jsou reprezentovány pomocí **JButton** komponent s přiřazenými obrázky pro stisknutý a uvolněný stav.
- **Zobrazení názvu tónu:** Po stisknutí klávesy se na displeji klavíru zobrazí název hraného tónu a k tomu pomocí jaké klávesy byl tón zahrán. (např. “C2”, “Eb2”)
- **Výběr nástroje:** Pomocí **JComboBox** lze měnit nástroj, kterým jsou tóny přehrávány
- **Regulace hlasitosti:** Pomocí **JSlider** lze nastavit hlasitost mezi 0 % až 100%. Tato hodnota se aplikuje na každý přehrávaný zvuk.

2.2....Funkce Přehrávání

- Každé stisknutí klávesy je zachyceno pomocí **KeyListener** nebo **InputMap/ActionMap**.
- Na základě zvoleného nástroje se sestaví cesta ke zvukovému souboru (např. Sounds/PianoSounds/C2.wav).
- Pomocí třídy **SoundPlayer** se zvuk načte a přehraje.
- Podle nastaveného slideru se přehraje zvuk dle určené hlasitosti.

2.3....Zvukový engine

- Třída **SoundPlayer** obstarává načítání a přehrávání **.wav** souborů.
- Je zde implementována regulace hlasitosti pomocí decibelové škály (**FloatControl**).

2.4....Rozšířitelnost

- Nové nástroje lze přidat jednoduchým vložením složky s názvem nástroje do adresáře **Sounds/** a přidat do **JComboBoxu**.

3 Systém requirements

Program byl napsán v jazyce Java, konkrétně ve verzi **Oracle OpenJDK 21.0.1**. Pro jeho spuštění je tedy potřeba mít nainstalované odpovídající JDK (Java Development Kit). Krom samostatného JDK nejsou potřeba žádné další knihovny.

4 Základní struktura

Aplikace Virtual Piano je implementována v jazyce Java s využitím knihoven pro GUI (`javax.swing`) a zvukové manipulace (`javax.sound.sampled`). Je rozdělena do několika tříd, které spolu komunikují:

1. **Třída Main:**
Spouští aplikaci a inicializuje hlavní okno (JFrame), které obsahuje ovládací panel kláves.
2. **Třída SoundPlayer:**
Zajišťuje načítání, dekodování a přehrávání zvukových souborů (.wav). Implementuje také ovládání hlasitosti přehrávání.
3. **Třída ControlPanelGUI:**
Obsahuje veškerou logiku uživatelského rozhraní, jednotlivé klávesy piana jako JButton komponenty, přidělení zvuků ke klávesám a reakci na klávesové zkratky. Pro každou klávesu je nastaven event handler, který přehraje příslušný zvuk a aktualizuje vizuální stav klávesy.
Také je zde implementována obsluha vstupu z klávesnice, zobrazování stisknutých znaků a ovládání hlasitosti pomocí slideru.

5 Testovací data

Program je možné testovat manuálně opakovaným spouštěním a zkoušením všech dostupných funkcí. Osobně jsem nejčastěji testoval jednotlivé klávesy myši i klávesnicí a ověřoval, zda se správně přehraje odpovídající zvuk a zda se správně zobrazí noty na displeji. Doporučuji hlavně otestovat, zda se na displeji klavíru opravdu zobrazují pouze znaky, které sedí ke klávesám na klavíru.

Celkově tedy doporučuji testovat program způsobem “klikáním na všechno možné”.

6 Uživatelská příručka:

Program se ovládá jednoduše pomocí myši a klávesnice. Po spuštění se zobrazí hlavní okno s grafickým rozhraním virtuálního pianu. Klávesy pianu lze ovládat buď klikáním myši, nebo stiskem přiřazených kláves na běžné klávesnici.

Na pravé straně obrazovky se nachází posuvník hlasitosti sloužící k nastavení hlasitosti výstupního zvuku. Dále se na levé straně obrazovky nachází ComboBox umožňující vybrat, jaký hudební nástroj bude piano napodobovat.

Nakonec se uprostřed obrazovky nachází displej zobrazující noty, které byly právě přehrány a historii stisknutých kláves.

Program je navržen tak, aby byl co nejjednodušší na ovládání a co nejbližší pocitu skutečného hraní na piano. Nevyžaduje žádnou speciální instalaci ani znalosti – stačí ho spustit a můžete začít hrát.

7 Závěr:

S projektem jsem nakonec velmi spokojený. Největší problém mi většinou dělalo designování uživatelského rozhraní. Kdybych věděl, kolika problémům jsem se mohl vyhnout, kdybych namísto Swing Designeru, GUI programoval, nepoužíval bych ho.

Další problém, co mě trápil bylo zapisování kláves, které jsem zahrál na displej. Jakmile jsem totiž někde myši kliknul, KeyListener přestal reagovat na moji klávesnici a nedopisoval nové klávesy na displej. Problém jsem naštěstí vyřešil velmi jednoduše a to tím, že jsem u všeho krom mainPanel vypl funkci focusable. Je to něco s čím jsem se dlouho trápil abych na to přišel, ale zároveň věc, která se mi bude hodit vědět do budoucna.

Jedna věc, kterou jsem přemýšlel že bych implementoval byla možnost nahrát vlastní zvukový soubor který by pak program sám zpracoval, nahradil frekvencí tónů a tím vytvořil “nový nástroj”. Bohužel to bylo nad moje schopnosti ale do budoucna bych se rád k tomuto nápadu vrátil a pokusil se tuto funkci přidat.