

**SPŠE Ječná**

**Informační Technologie**

Střední průmyslová škola elektrotechnická, Praha 2, Ječná 30, Ječná  
517, 120 00 Nové Město

**Fakturační systém v Javě**

**Tomáš Bakalík**

Informační Technologie

2025

## Obsah

1. Cíl práce .....	3
2. Popis práce .....	3
3. Požadavky na systém .....	3
4. Základní struktura .....	3-4
5. Testovací data .....	4
6. Uživatelská příručka .....	4-5
7. Závěr .....	5

## 1 Cíl práce:

Cílem mé práce bylo vytvořit jednoduchý fakturační systém, který umožní IČAřům vystavovat faktury.

## 2 Popis práce:

Uživatel si v programu může vytvořit svůj účet tak, že se registruje, při registraci zadá své osobní údaje. Tyto údaje se uloží do souboru. Po registraci se uživatel může přihlásit pomocí e-mailu a hesla, která zadal při registraci. Program se podívá, jestli je uživatel uložený v souboru, pokud ano, přihlásí ho a zobrazí mu uživatelské menu. V menu si uživatel může vybrat, jestli chce vystavit novou fakturu, nebo se podívat na faktury, které už vystavil. Když vystaví novou fakturu, automaticky se mu předvyplní údaje, které zadal při registraci. Jestliže se rozhodne zobrazit starší faktury, program je načte ze souboru, kde jsou faktury uloženy daného uživatele podle jeho e-mailové adresy.

## 3 Požadavky:

K běhu programu je potřeba mít nainstalovanou Javu, a to alespoň ve verzi 17 nebo vyšší. Program byl vytvořen v prostředí IntelliJ IDEA a využívá grafické rozhraní knihovny Swing, která je součástí JDK. Program nevyžaduje žádné další knihovny ani internetové připojení. Faktury a údaje o uživateli se ukládají do textových souborů, které se automaticky vytvářejí ve složce, kde se program spouští. Program je určený pro spuštění na stolním počítači nebo notebooku s operačním systémem Windows, Linux nebo macOS.

## 4. Základní struktura:

Základ programu tvoří třídy pro správu uživatelů, práci s fakturami a zobrazení grafického rozhraní.

O správu uživatelů se stará třída `SpravceUzivatelu`, která umí uložit nového uživatele do souboru a později ho z něj znovu načíst. Každý uživatel je reprezentován třídou `Uzivatel`, která obsahuje jeho osobní údaje jako jméno, e-mail nebo telefon.

O správu faktur se stará třída `SpravceFaktur`. Tato třída dokáže faktury přidávat do seznamu a načítat je ze souboru podle e-mailu přihlášeného uživatele. Třída `Faktura` reprezentuje fakturu, zde se také nachází informace o klientovi (`Klient`), položkách (`PolozkaFaktury`) a o stavu faktury (`StavFaktury`), například jestli je zaplacená nebo ne.

Grafické prostředí je rozdělené do více tříd. HlavníOkno zobrazuje úvodní obrazovku. RegistraceOkno slouží k vytvoření nového účtu (otevřít se, když uživatel klikne na tlačítko „Registrovat“). PřihlaseňOkno slouží k přihlášení. Po přihlášení se otevře UživatelMenuOkno, kde si uživatel může vybrat, co chce dělat, jestli chce například vystavit novou fakturu (pomocí třídy FormularFaktura) nebo si zobrazit své dříve vystavené faktury (pomocí třídy ZobrazitFakturyOkno).

Program funguje tak, že se každému zaregistrovanému uživateli ukládají faktury do jeho vlastního textového souboru podle jeho e-mailu. Díky tomu má každý přístup pouze ke fakturám i po ukončení a následném znovu spuštění programu.

## 5 Testovací data:

Program jsem testoval ručně tím, že jsem zkoušel různé situace, které mohou nastat při běžném používání. Při registraci jsem například zadával neplatné údaje, jako prázdné jméno, špatně zadaný e-mail nebo příliš krátké heslo, abych ověřil, jestli program správně zobrazí chybu. U přihlášení jsem záměrně zkoušel špatné kombinace e-mailu a hesla, abych se ujistil, že se uživatel nepřihlásí, pokud údaje nejsou správné. Testoval jsem také, jestli se po registraci opravdu vytvoří nový uživatel a jestli se po přihlášení načtou správné faktury. Při vystavování faktury jsem zkoušel zadat chybné hodnoty, například text místo čísla u ceny nebo prázdná pole, abych zjistil, jestli program pozná, že není něco správně vyplněné. Vybral jsem tyto funkce, protože jsou nejdůležitější pro správné fungování programu. Kdyby u nich nastala chyba, uživatel by nemohl program používat tak, jak má, proto jsem testoval právě toto.

## 6 Uživatelská příručka:

Po spuštění programu se uživateli zobrazí hlavní okno, ve kterém si může vybrat jednu ze tří možností: Vystavit fakturu, Registrovat, nebo Přihlásit se.

### **Vystavit fakturu:**

Pokud uživatel nechce vytvářet účet a jen si rychle vyzkoušet, jak program funguje, může kliknout na tlačítko Vystavit fakturu hned v hlavním okně. Otevře se jednoduchý formulář, kde uživatel ručně vyplní údaje o vystaviteli, klientovi i položce. Po odeslání si fakturu může stáhnout do počítače.

### **Registrace:**

Pokud se uživateli program líbí, může se registrovat tak, že klikne na tlačítko Registrovat. Otevře se mu formulář, kde vyplní své osobní údaje: jméno, příjmení, adresu, telefon, IČO, e-mail, heslo a ID. Po potvrzení se údaje uloží do souboru a uživatel se může přihlásit.

**Přihlášení:**

Po registraci se uživatel může přihlásit pomocí e-mailu a hesla. Pokud jsou údaje správné, otevře se uživatelské menu.

**Uživatelské menu:**

Po přihlášení má uživatel dvě hlavní možnosti: Vystavit novou fakturu, zde se mu otevře formulář, kde jsou předvyplněné údaje vystavitele podle zaregistrovaného účtu. Uživatel doplní údaje o klientovi a fakturované položce. Po odeslání se faktura uloží do souboru, a zároveň si ji uživatel může stáhnout.

**Zobrazit moje faktury:**

Zobrazí se přehled všech faktur, které uživatel v minulosti vystavil. Tyto faktury se načítají ze souboru, který je pojmenovaný podle e-mailu uživatele.

Program se ovládá klikáním na tlačítka a vyplňováním jednoduchých formulářů. Všechna okna mají přehledné a uživatelsky přívětivé rozhraní.

**7 Závěr:**

Na začátku jsem měl jen základní představu o tom, jak by měl fakturační program fungovat. Postupně jsem však narazil na několik problémů, především při práci s ukládáním a načítáním dat ze souborů. Dlouho jsem řešil, jak správně rozlišit jednotlivé uživatele tak, aby měl každý přístup pouze ke svým vlastním fakturám. Jako výsledné řešení ohledně ukládání dat do souborů jsem zvolil pojmenování jednotlivých souborů podle e-mailu uživatele. To se ukázalo jako jednoduché a za mě pro můj program velmi efektivní řešení. Další výzvou bylo vytvoření přehledného grafického rozhraní pomocí knihovny Swing. Se Swingem jsem totiž nikdy nepracoval, takže jsem se toho trochu bál. Dbal jsem na to, aby bylo prostředí jednoduché a hlavně uživatelsky přívětivé. Naučil jsem se, jak efektivně propojit jednotlivá okna a správně mezi nimi předávat data. Některé části programu jsem musel několikrát přepracovat nebo zcela změnit, protože jsem postupně nacházel lepší a jednodušší řešení. I přes náročnost mě práce na projektu bavila a přinesla mi mnoho nových zkušeností, především s tvorbou graficky v Javě. I když jsem se této části na začátku bál, nakonec jsem rád, že jsem se rozhodl projekt udělat graficky a myslím, že se mi to celkem podařilo. Celkově mám z výsledku dost dobrý pocit a jsem rád, že jsem projekt dokončil zhruba tak, jak jsem si původně představoval.