

Systém pro správu 3D tisku

**Semestrální projekt Softwarového
inženýrství**

**Bc. Vladimír Zábrodský
Bc. Lukáš Caha
Bc. Mária Běhalová**

Brno 2015

Obsah

1	Úvod a cíl práce	3
1.1	Úvod.....	3
1.2	Cíl práce.....	3
2	Projektová část	4
2.1	Ericsson-Penker diagram.....	4
2.2	Use Case diagram	7
	Seznam modelů	8
	Zobrazení modelu.....	8
	Smazání modelu	8
	Přidání modelu	9
	Úprava modelu	10
	Vytisknou model	11
2.3	Diagram tříd	12
	Analýza Use Case diagramu	12
2.4	Sekvenční diagram.....	14
3	Závěr	16

1 Úvod a cíl práce

1.1 Úvod

Tato práce je zaměřena na fázi zahájení a rozpracování při tvorbě softwaru. Problémovou doménou je v našem případě systém pro správu 3D tisku. V této práci využíváme osvědčená pravidla UML metodiky, díky kterým jsme schopni rozdělit systém na jednotlivé části a zachytit procesy, které se v systému odehrávají. Práce postupně přechází k fázi rozpracování kde dochází k tvorbě objektových tříd a poté k přípravě na fázi konstrukce.

1.2 Cíl práce

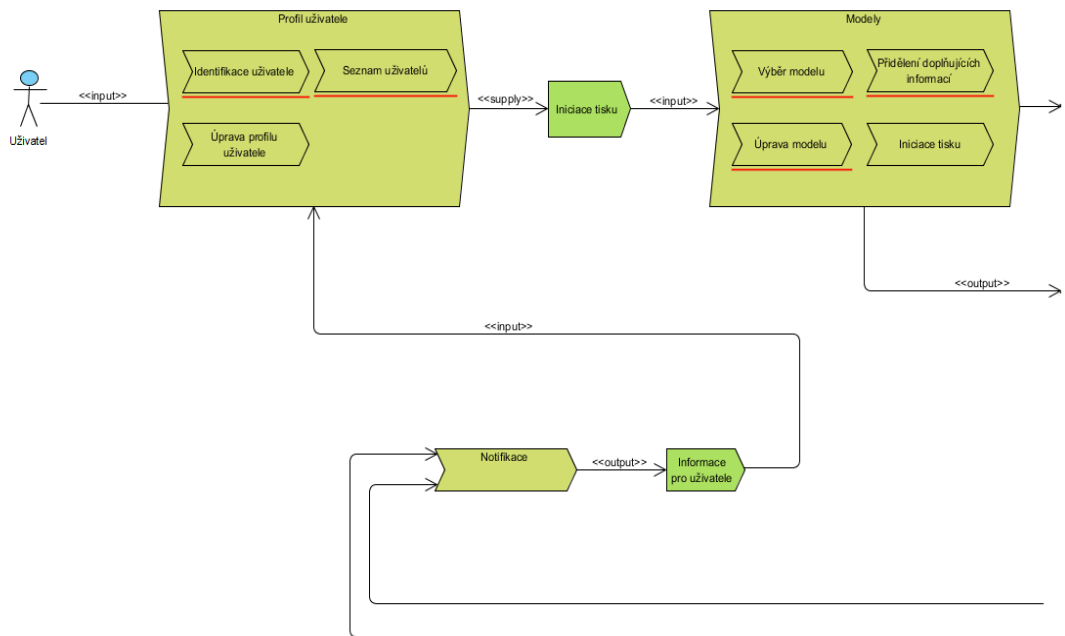
Cílem práce je provést fázi zahájení a rozpracování tvorby softwaru na problémové doméně systém pro správu 3D tisku.

2 Projektová část

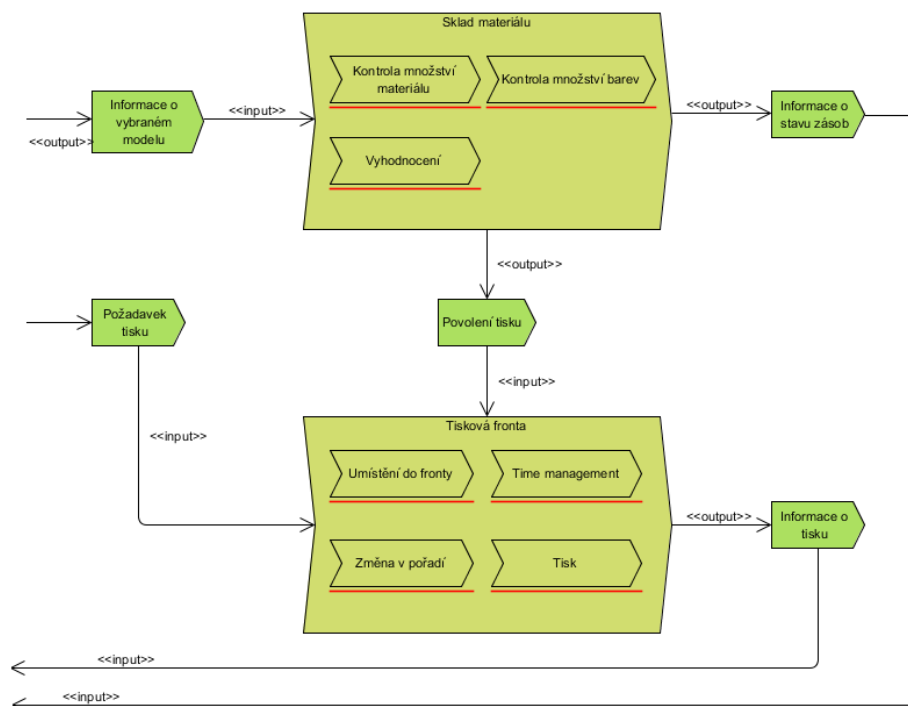
V této části se budeme zabývat návrhem diagramů. Postupně vytvoříme Ericsson-Penker diagram první a druhé vrstvy, Use Case diagram, diagramy aktivit, diagram tříd a diagram aktivit.

2.1 Ericsson-Penker diagram

Nejprve vytvoříme Ericsson-Penker diagram, který zachycuje všechny procesy v aplikaci 3D tisk.

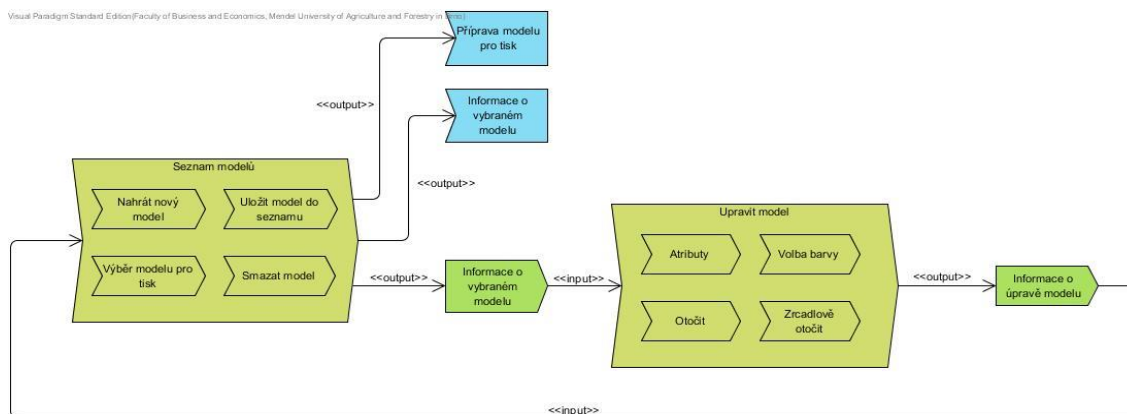


Obrázek 1: E-P diagram 1. vrstvy, zachycení procesů

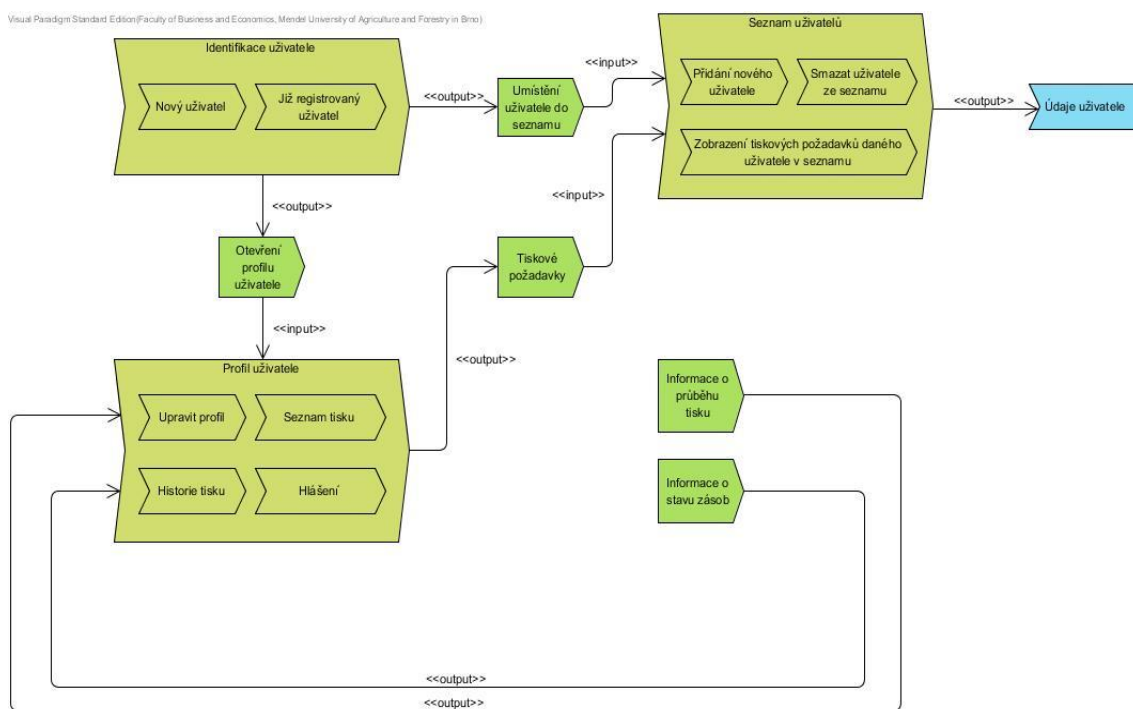


Obrázek 2: E-P diagram 1. vrstvy, 2. část diagramu

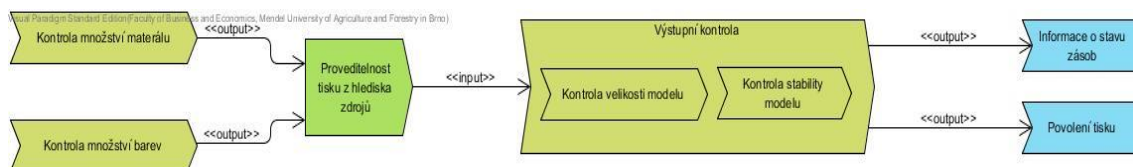
Nyní se podíváme na jednotlivé vrstvy blíže. Následující E-P diagramy zachycují druhou vrstvu procesů v aplikaci.



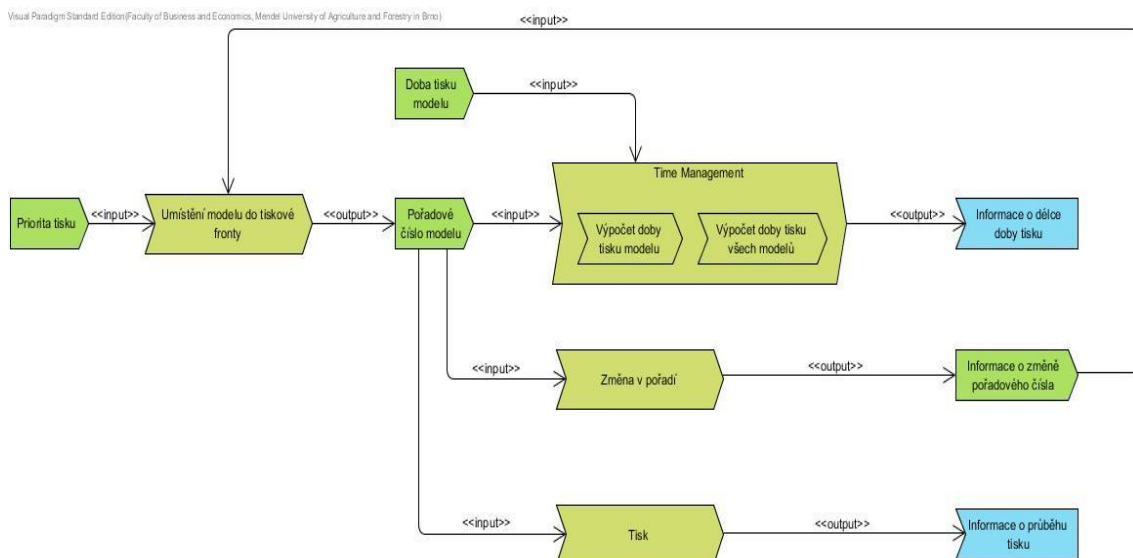
Obrázek 3: E-P diagram 2. vrstvy, Modely



Obrázek 4: E-P diagram 2. vrstvy, Profil uživatele



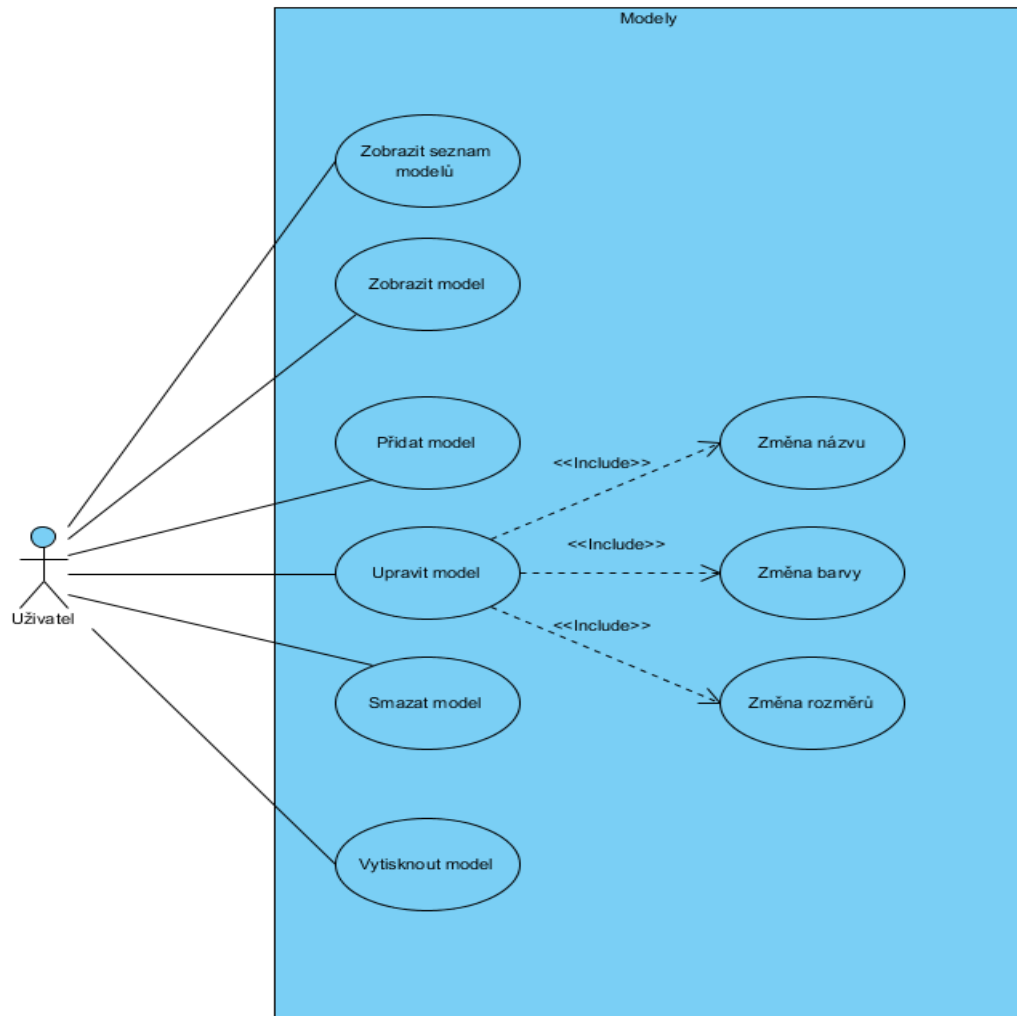
Obrázek 5: E-P diagram 2. vrstvy, Sklad materiálu



Obrázek 6: E-P diagram 2. vrstvy, Tisková fronta

2.2 Use Case diagram

Pomocí Ericsson-Penker diagramu byl vytvořen diagram případů užití (= Use Case diagram) modulu modely. Tento diagram zachycuje případy užití, které spouští uživatel.



Obrázek 7: Případy užití, Modely

Seznam modelů

Primární scénář:

1. Uživatel spustí modul modely.
2. Program načte databázi modelů.
3. Uživateli se zobrazí seznam modelů.

Alternativní scénář (Prázdný seznam):

3. Databáze modelů je prázdná.
4. Program se zeptá uživatele, zda chce přidat model do seznamu.

Vstupní podmínky:

Uživatel musí spustit modul modely.

Následné podmínky:

Uživatel zvolí přidání modelu.

Uživatel zvolí upravit model.

Uživatel zvolí pokračovat v tisku.

Zobrazení modelu

Primární scénář:

1. Uživatel označí model v seznamu modelů.
2. Uživatel zvolí zobrazit náhled modelu.
3. Program otevře nové okno s náhledem modelu.

Vstupní podmínky:

Uživatel označí model v seznamu.

Uživatel zvolí zobrazit náhled modelu.

Následné podmínky: Uživatel zavře náhled modelu.

Smazání modelu

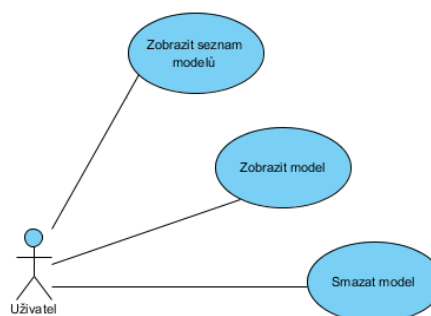
Primární scénář:

1. Uživatel označí model v seznamu modelů.
2. Uživatel zvolí smazat model.
3. Program se zeptá uživatele, zda chce model opravdu smazat.
4. KDYŽ Uživatel zvolí „Ano“
5. Program provede smazání modelu ze seznamu.
6. KDYŽ Uživatel zvolí „Ne“
7. Návrat na seznam modelů.

Vstupní podmínky:

Uživatel označí model v seznamu.

Uživatel zvolí smazat model.



Následné podmínky:

Uživatel zvolí odpověď „Ne“ na otázku, zda chce opravdu smazat model.

Program provede smazání modelu.

Přidání modelu

Primární scénář:

1. Uživatel zvolí přidání modelu.
2. Uživateli se otevře dialogový panel s nastavenou cestou do složky Modely.
3. Uživatel vybere soubor na disku, který chce přidat do seznamu modelů.
4. Uživatel zvolí přidat model.
5. Systém provede kontrolu formátu.
6. Systém provede kontrolu duplicity.
7. Model se uloží do seznamu modelů.

Alternativní (špatný formát) :

6. Program vypíše chybu „Model nelze přidat, protože má neznámý formát.“

Alternativní 2 (model již existuje)

7. Program vypíše chybu „Model nelze přidat, daný model již existuje. Chcete jej přepsat?“
8. KDYŽ „Ano“
9. Program provede přepsání modelu a model uloží do seznamu.
10. KDYŽ „Ne“
11. Program provede návrat na dialogový panel.

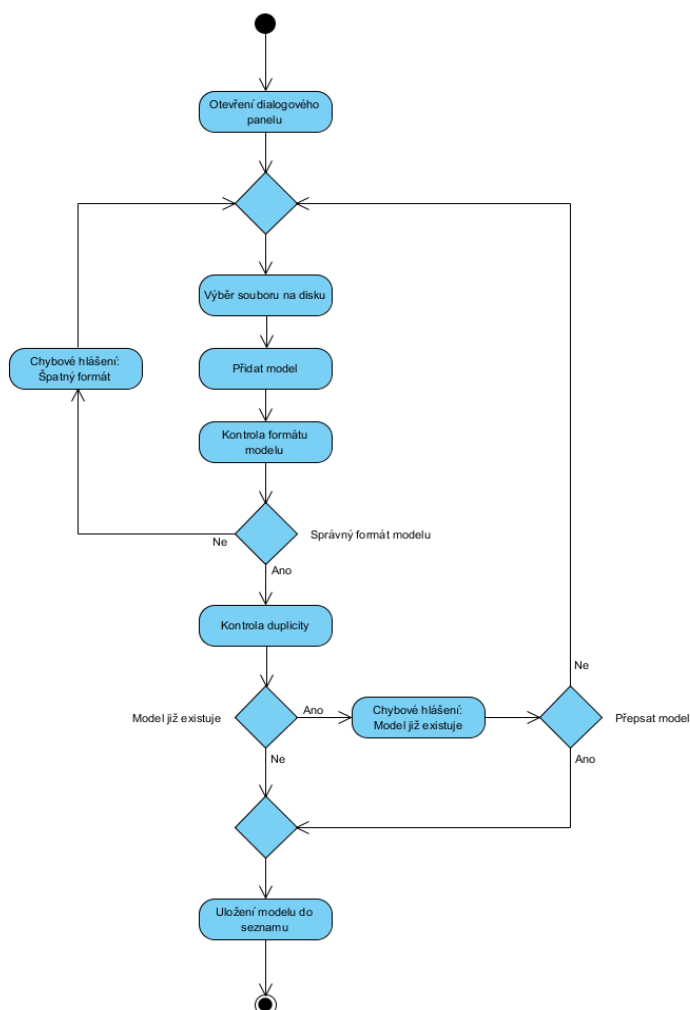
Vstupní podmínky:

Uživatel zvolí přidání modelu.

Následné podmínky:

Uživatel zavře dialogový panel.

Program uloží model do seznamu modelů.



Úprava modelu

Primární scénář:

1. Uživatel zvolí upravit model.
2. Uživateli se zobrazí náhled modelu a jeho parametry, které lze měnit.
3. Uživatel provede úpravy modelu.
4. Uživatel zvolí uložit model zpět do seznamu.
5. Program provede kontrolu duplicity.
6. Program uloží model.
7. Návrat na seznam modelů.

Alternativní (Název modelu již existuje):

6. Program vypíše chybu „Daný model již existuje.“
7. Program se zeptá uživatele zda, chce daný model přepsat.
8. KDYŽ „Ano“
9. Program provede přepsání modelu a uloží jej do seznamu modelů.
8. KDYŽ „Ne“
9. Návrat na úprava modelu.

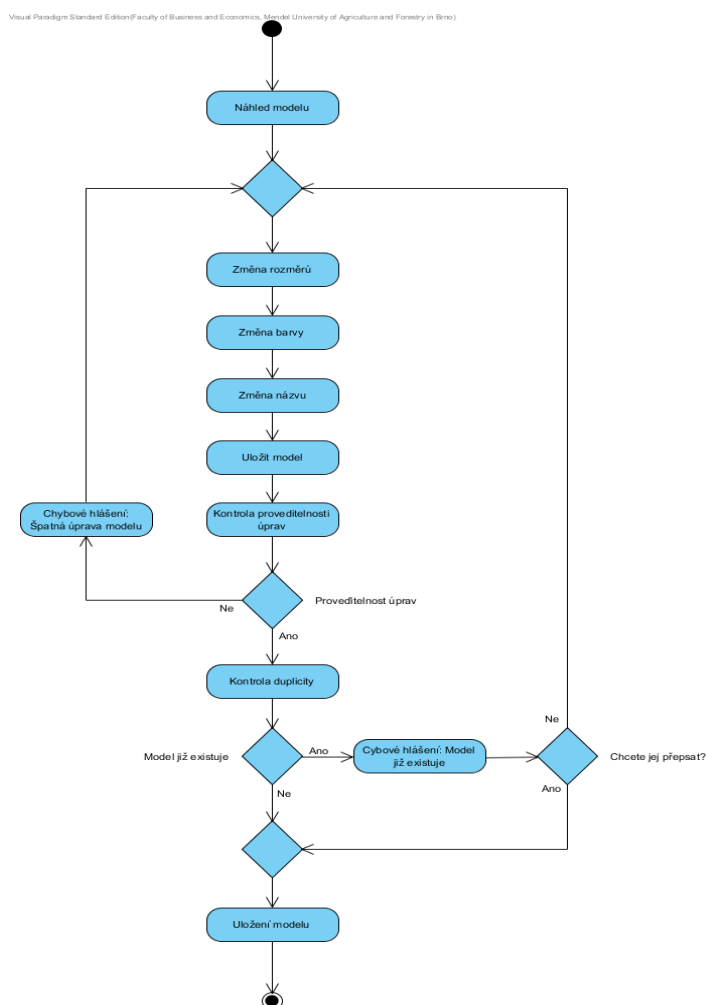
Vstupní podmínky:

Uživatel zvolí upravit model.

Následné podmínky:

Uživatel provede návrat na seznam modelů.

Program uloží model do seznamu modelů.



Vytisknou model

Primární scénář:

1. Uživatel zvolí pokračovat v tisku.
2. Zobrazení okna tisku.
3. Kontrola množství materiálu.
4. Kontrola množství barev.
5. Výpočet stability modelu.
6. Výpočet délky doby tisku.
7. Zobrazení tiskových informací.
8. Uživatel zvolí tisk.
9. Inicie tisku modelu.

Alternativní (Tisk nelze provést):

8. Zobrazení chybového hlášení o neproveditelnosti tisku.

Vstupní podmínky:

Uživatel zvolí pokračovat v tisku.

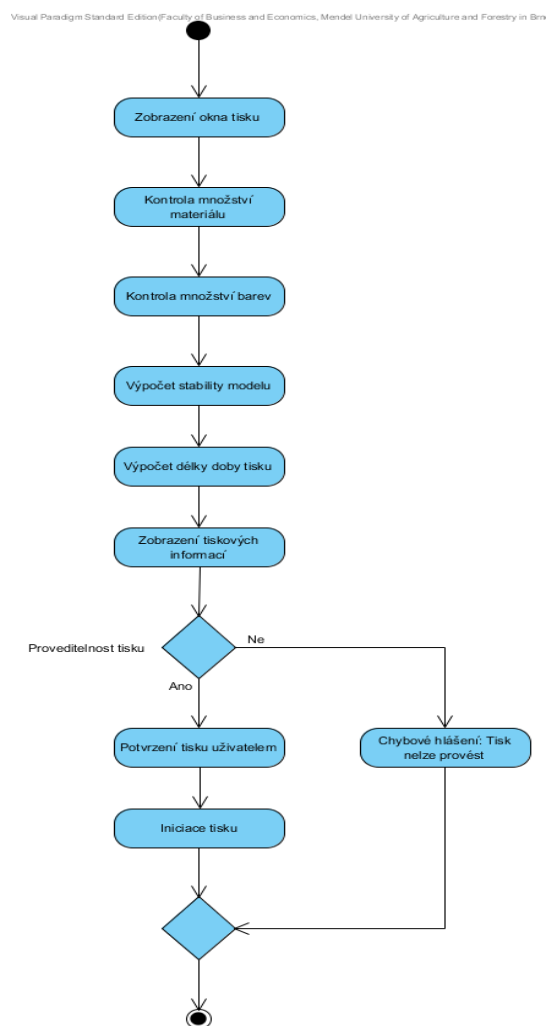
Následné podmínky:

Program ukončí tisk, protože nelze provést.

Uživatel nepotvrdí tisk.

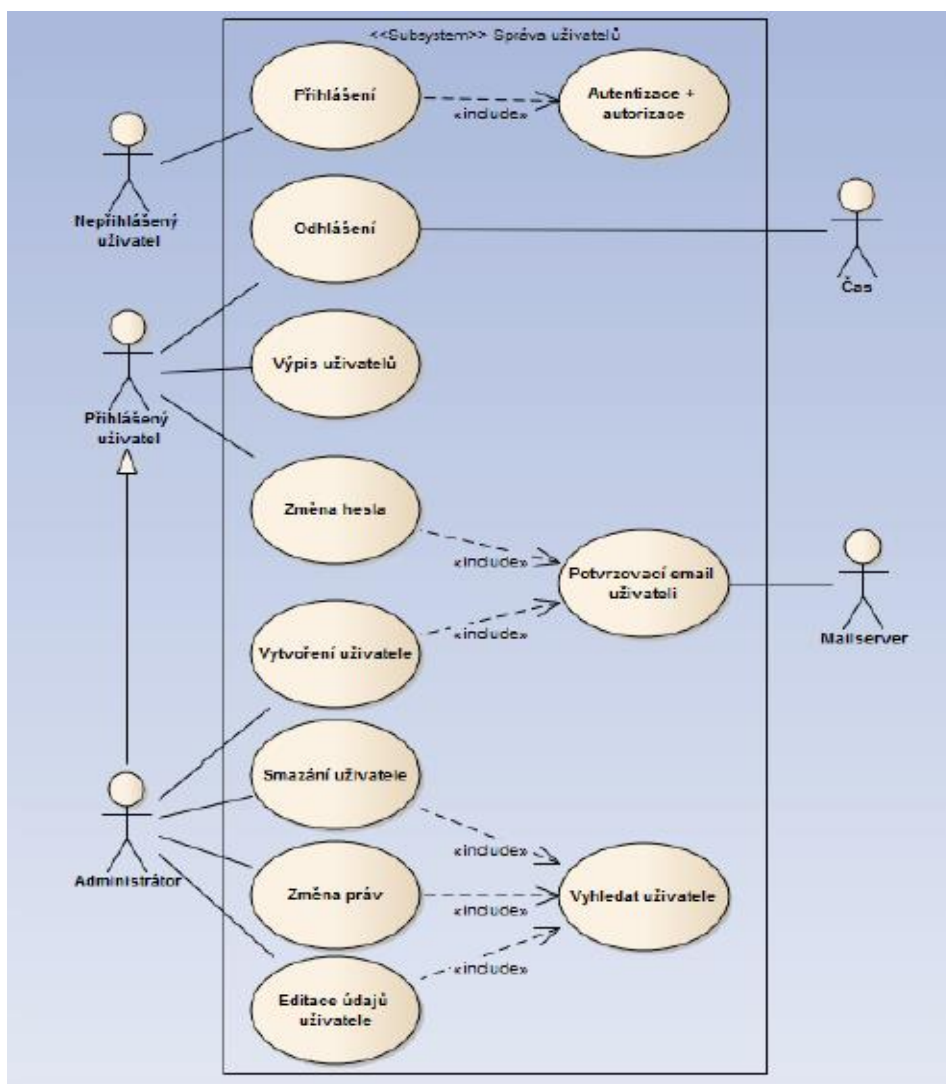
Program provede iniciaci tisku.

2.3



2.3 Diagram tříd

Pomocí případů užití a pečlivě sepsaných scénářů k nim, lze nyní bez problémů sestavit analytický diagram tříd.



Obrázek 8: Získaný Use Case diagram

Analýza Use Case diagramu

Typy kandidátů analytických tříd:

- Uživatel
- Administrátor
- Nový uživatel
- Přihlášený uživatel
- Registr uživatelů
- Přihlašovací formulář

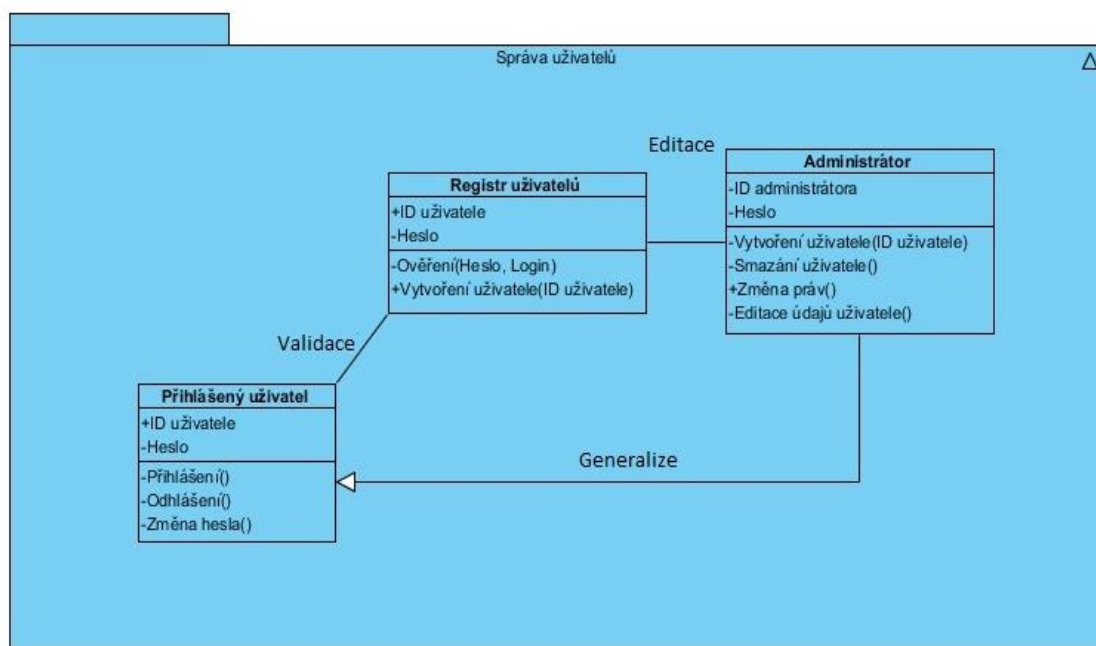
Atributy:

- Přihlašovací údaje

Operace:

- Přihlášení, odhlášení
- Vytvoření, smazání uživatele
- Změna hesla
- Změna práv
- Editace údajů
- Ověření údajů

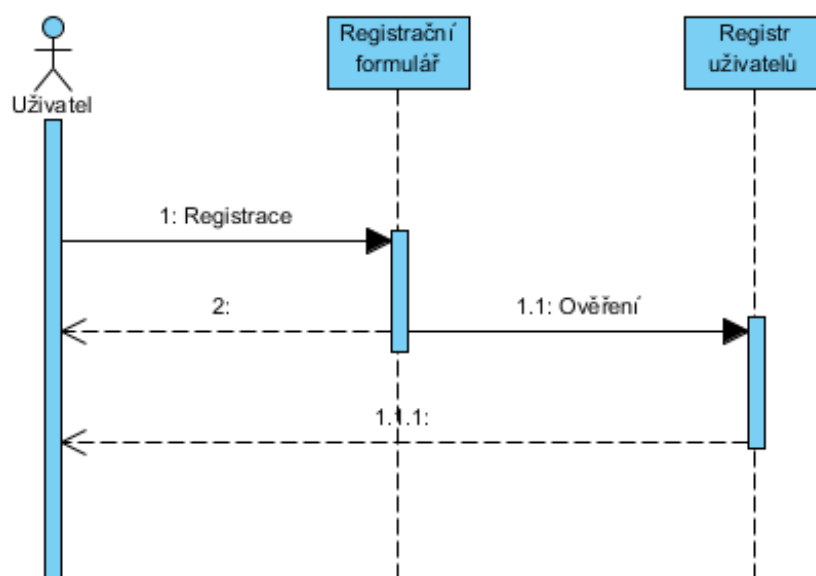
Na základě získaného Use Case diagramu popisujícího případy užití modulu Správa uživatelů, jsme vytvořili následující diagram tříd.



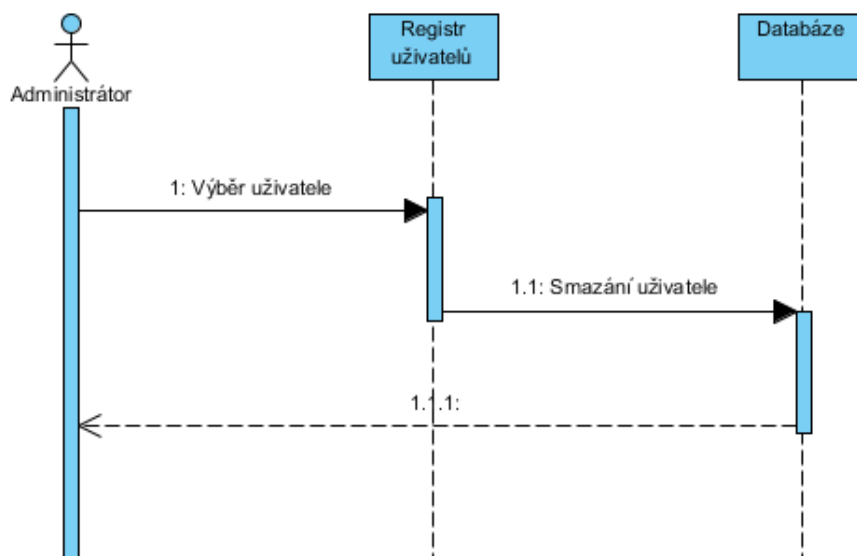
Obrázek 9: Analytický diagram tříd, modul Správa uživatelů

2.4 Sekvenční diagram

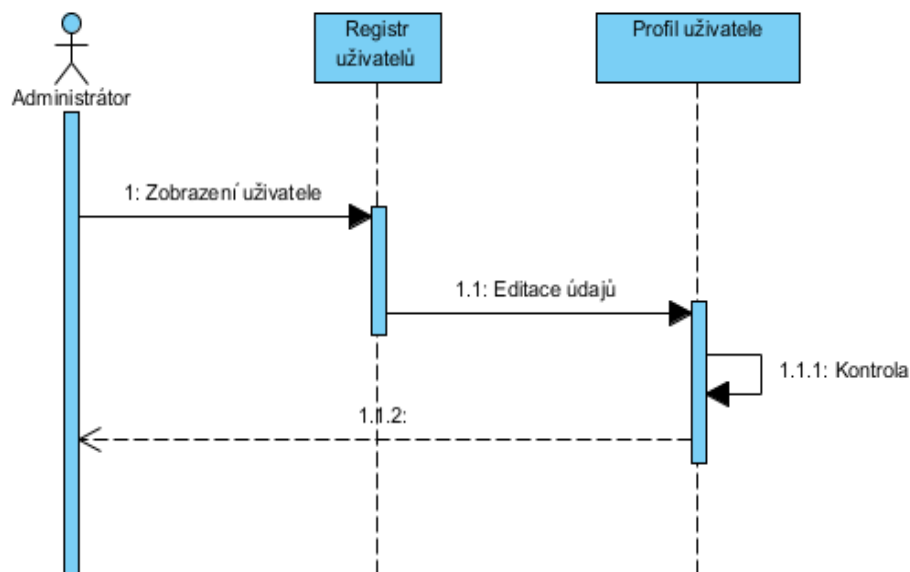
Nyní vytvoříme sekvenční diagram modulu Správa uživatelů z původního Use Case diagramu.



Obrázek 10: Vytvoření uživatele

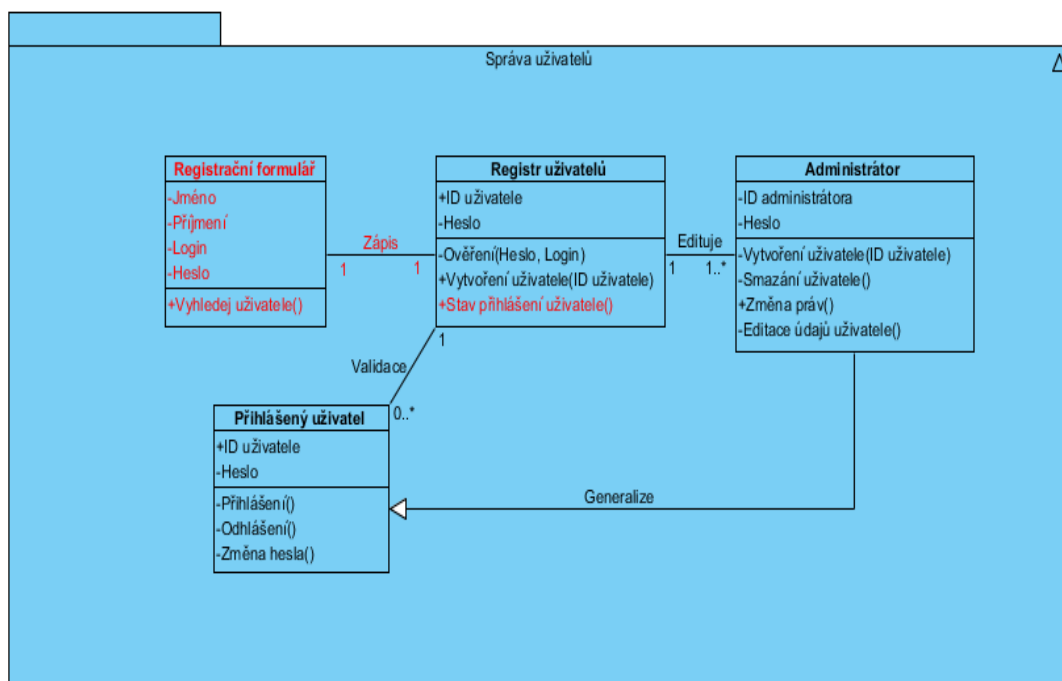


Obrázek 11: Smazání uživatele



Obrázek 12: Editace údajů uživatele

Na základě sekvenčních diagramů byly zjištěny některé nedostatky v diagramu tříd. Tyto nedostatky jsou opraveny na následujícím diagramu:



Obrázek 13: Opravený diagram tříd Správa uživatelů

3 Závěr

Na základě metodiky UML, jsme provedli postup kterým dochází k vývoji softwaru. Výsledný diagram tříd, který jsme v této práci vytvořili je možné použít pro konstrukci funkčního programového kódu a navázat tak na tuto práci dalšími fázemi vývoje softwaru až po vytvoření funkčního produktu, který slouží svému cíli.