**Slovenská Technická Univerzita v Bratislave, Fakulta Informatiky a Informačných Technológií**

**Správa o realizácii projektu**

**TourMate**

A picture containing font, graphics, typography, text

Description automatically generated

**Objektovo Orientované Programovanie**

Obsah

[Rámcová téma 3](#_Toc134914823)

[Detailnejšie zobrazenie 5](#_Toc134914824)

[Hierarchia Používateľov 5](#_Toc134914825)

[Ubytovanie Hierarchia 6](#_Toc134914826)

[Trieda Hike a HikeManager 7](#_Toc134914827)

[Splnenie Kritérií 8](#_Toc134914828)

[Hlavné Kritériá 8](#_Toc134914829)

[Zapuzdrenie 8](#_Toc134914830)

[Dedenie, Agregácie, Prekonávanie 8](#_Toc134914831)

[Vedľajšie Kritériá 8](#_Toc134914832)

[Návrhový Model 8](#_Toc134914833)

[Vyhodenie Výnimky 9](#_Toc134914834)

[GUI 10](#_Toc134914835)

[Viacniťovosť 11](#_Toc134914836)

[11](#_Toc134914837)

[Generickosť 11](#_Toc134914838)

[RTTI 12](#_Toc134914839)

[Vnorená Trieda 12](#_Toc134914840)

[12](#_Toc134914841)

[Lambda Výraz 13](#_Toc134914842)

[Default Method Implementácia 13](#_Toc134914843)

[Hlavné Verzie Programu 14](#_Toc134914844)

[Spustenie programu 14](#_Toc134914845)

# Rámcová téma

Cesty môžu predstavovať veľmi zložité podujatia, ktoré naozaj môžu mať prospech zo softvérovej podpory. Často zahŕňajú odbočky za účelom navštívenia zaujímavých miest v blízkosti hlavnej trasy. Rôzne služby a udalosti môžu byť ponúkané na týchto miestach. Tieto miesta môžu byť odporúčané na základe potrieb a ohraničení cestujúceho. Počas cesty sa pôvodný plán môže zmeniť. Cestovanie v skupine môže zahŕňať synchronizáciu odbočiek. Či plán cesty vytvoril profesionál alebo samotní cestujúci, môže byť hodný zdieľania s ostatnými.

**TourMate**

Cieľom mojej aplikácie je odporúčanie a plánovanie turistických výprav po Slovensku, keďže mnoho ľudí pozná iba Vysoké Tatry a nevie o mnohých krásach, ktoré slovenská príroda ponúka. Moja aplikácia bude fungovať ako „cestovná kancelária“ pre turistov. Bude obsahovať napríklad odporúčania na základe zvolených preferencií používateľa a na základe jeho predchádzajúcich výletov a tom, ako tieto výlety ohodnotil. Spomínané preferencie budú v podstate súčasťou plánovania trasy. Používateľ si zvolí napríklad rozpočet, vzdialenosť od domova a spôsob dopravy na miesto. Taktiež bude možné vyberať podľa náročnosti a dĺžky trasy, prítomnosti bufetov a horských chát alebo podľa toho, či je túra určená aj pre rodiny s deťmi alebo len pre starších turistov. Jednotlivý turisti budú rozdelení na skúsenejších a amatérskych a podľa toho budú mať prístup k jednotlivým výpravám. Aplikácia bude obsahovať aj predpoveď počasia, prípadné výstrahy pred nebezpečným počasím a v aplikácii bude zakomponovaná aj mapa. Používateľ bude mať možnosť založiť si vlastný účet, kde si bude môcť plánovať cesty, hodnotiť tie, ktoré už prešiel a zdieľať s priateľmi fotografie z nich. Do aplikácie by som rád zakomponoval aj ukazovateľ celkového počtu prejdených kilometrov pre používateľa a odmeny/trofeje za dokončené turistické výpravy.

A screenshot of a computer

Description automatically generated with medium confidence**Diagram**

# Detailnejšie zobrazenie

## A screenshot of a computer Description automatically generated with low confidenceHierarchia Používateľov

Dedenie z abstraktnej triedy **Customer**. Ide o hierarchiu trojitého dedenia. Z Customer abstraktnej triedy, dedí trieda **Child** a **Adult**. Child je trieda dieťaťa a Adult je dospelý, ktorý však nemá prístup ku všetkým chodníkom, keďže nemá profesionálnu licenciu. Z triedy Adult dedí trieda **Professional**, kde ide o dospelého človeka s licenciou, ktorý má prístup ku každému chodníku.

Je tu taktiež použitý interface *access*, ktorý zabezpečí výpis toho, kde má daná trieda prístup.

Zároveň tu je aj default metóda, ktorá poskytuje základné informácie

o užívateľovi.

## Ubytovanie Hierarchia

A screenshot of a computer

Description automatically generated with medium confidence

Toto je hierarchia ubytovania. Opäť trojitá hierarchia, kde pôvodne máme chalupu na prespanie **Cottage**, z nej následne dedením vzniká trieda **CottageWithRestaurant** kde sa užívatelia môžu najesť. Z tej ďalej dedí trieda **MountainHotel**, pričom táto trieda obsahuje aj agregáciu a ide o poslednú triedu tejto hierarchie.

Je tu taktiež interface s metódou **getAccInfo()**,čo je vlastne metóda na výpis informácií ohľadom daného ubytovania.

## A screenshot of a computer Description automatically generated with medium confidenceTrieda Hike a HikeManager

Trieda **Hike** slúži na uchovávanie informácii o jednotlivých turistických tratiach. Je to veľmi rozsiahla trieda, ktorá obsahuje aj agregáciu. Jej agregáciou jej práve **ubytovanie**, ktoré môže a nemusí byť jej súčasťou.

Trieda **HikeManager** slúži na inicializáciu turistík a priradenie ubytovania k nim. Ďalej sa tu nachádzajú dôležité metódy, ako napríklad metóda na filtrovanie turistiky podľa preferencie používateľa.

# Splnenie Kritérií

## Hlavné Kritériá

### Zapuzdrenie

Atribúty v triedach sú zapuzdrené za pomoci *private.* Na prístup k takto zapuzdreným atribútom v iných častiach aplikácie používam getter a setter metódy.

### Dedenie, Agregácie, Prekonávanie

V mojom programe sa nachádza trieda Hike, ktorá nie je v hierarchii, ale obsahuje agregáciu. Následne však môj program obsahuje 2 hierarchie s trojitým dedením. Obidve tieto hierarchie **Customer** a **Cottage**, obsahujú teda trojité dedenie. Obsahujú taktiež obidvoje interface a spolu s ním aj prekonávanie metód. Ďalej bola požiadavka agregácia, ktorá je taktiež prítomná v oboch daných hierarchiách.

Taktiež som sa snažil o správnu organizáciu kódu a celkové správne uplatnenie princípov Objektovo Orientovaného Programovania.

## Vedľajšie Kritériá

### A picture containing screenshot, text, font Description automatically generatedNávrhový Model

V mojom programe, konkrétne v triede HikeManager som využil návrhový model Factory na efektívnejšie vytváranie inštancií Hike pri inicializácii. Vytvoril som takúto triedu HikeFactory, a následne som jej metódu *createHike* volal ďalej v kóde.

### 

### Vyhodenie Výnimky

A screen shot of a computer program

Description automatically generated with low confidence

A screenshot of a computer program

Description automatically generated with low confidence

Takýmto spôsobom som použil vlastnú implementáciu výnimky vo vlastnom programe. V prvom prípade ide o situáciu, ak trieda Child nemá dospelého sprievodcu, tak nemôže ísť na túru, kvôli tomu, že je neplnoleté. Vtedy je vyhodená výnimka.

V ďalšom obrázku sa nachádza overenie toho, či je k turistike potrebný sprievodca. Ak nie, všetko je v poriadku. Ak áno, a sprievodca je typu profesionál, taktiež je to bez problémové. Ak však turista nemá sprievodcu a chce ísť na túru , kde je potrebný, je vyhodený error. Výnimka je chytená, vyhodená, ošetrená.

### GUI

V mojom programe používam grafické rozhranie po celý čas. Obsahuje mnoho dialógových okien, na mnohých miestach používateľ odovzdáva aj vstupné údaje. Keď sú nejaké polia prázdne, program je ošetrený a pošle upozornenie. Taktiež je tam mnoho informačných správ, ktoré vyskakujú.

Toto je úvodná obrazovka. Užívateľ si vyberá funkcionalitu.

A screenshot of a green screen

Description automatically generated with low confidence

A screenshot of a computer

Description automatically generated with low confidence

### Viacniťovosť

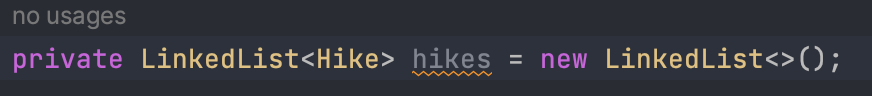
Toto je niť, ktorú som využil v mojej aplikácií. Je v Package Thread, a je využívaná pri Logine. Slúži na vypnutie programu, pri Logine, ak je používateľ neaktívny dlhšie ako 2 minúty, tak sa program sám od seba vypne. Využil som *extend Thread*, keďže som vedel, že táto niť nebude dediť z inej triedy, v opačnom prípade by som využil *implement runnable*.

### 

### A screen shot of a computer program Description automatically generated with low confidence

### Generickosť

Ako generickú triedu som využil spájaný zoznam, do ktorého mám v pláne pridávať turistiky navštívené užívateľom. Po poradí budú pridávané a tým aj časovo sedieť s tým, ako ich užívateľ prechádzal. Žiaľ som to zatiaľ nestihol celé implementovať.



### RTTI

RTTI som v mojom programe viackrát využil na zistenie toho, akého typu je moja trieda. Napríklad vyššie som to využil na zistenie toho, či je sprievodca typu Profesionál. Taktiež som tu mohol využiť *.getClass().getSimpleName()* avšak Instance Of sa mi pripadalo ako logickejšia možnosť.

A screenshot of a computer program

Description automatically generated with medium confidence

### Vnorená Trieda

### A screen shot of a computer program Description automatically generated with low confidence

Vnorenou triedou je napríklad agregácia využitá práve pri triede Adult. Adult má v sebe vnútornú triedu občianskeho preukazu, ktorý obsahuje metódu na kontrolu veku. Ak sa vek na občianskom nezhoduje s vekom v profile používateľa, účet je deaktivovaný.

### Lambda Výraz

Toto je lambda výraz, ktorý využívam v triede HikeManager, práve preto, aby som skontroloval, či boli triedy správne usporiadané a vybrané. Je to výpis do konzoly, ktorú kontrolujem potom s GUI a je to viac menej informácia pre človeka, ktorý spúšťa kód, aby ak nefunguje GUI, zistil, či naozaj dané turistiky neexistujú, alebo je chyba v GUI. **Metóda je mimo GUI.**

A picture containing text, screenshot, font, line

Description automatically generated

### Default Method Implementácia

A screen shot of a computer program

Description automatically generated with medium confidenceToto je default metóda, používaná práve v hierarchii Customer. Je to jednoduchá metóda, slúžiaca na výpis informácií o používateľovi, ako napríklad jeho kontaktné údaje, to či je jeho členstvo aktívne, a tak ďalej.

# Hlavné Verzie Programu

Na GitHub som pushoval veľakrát. Prvýkrát pred prvým odovzdávaním pracovnej verzie, kde som mal spravenú jednu trojitú hierarchiu používateľov a bol tam začiatok GUI, v ktorom bolo spravené prihlasovanie sa a registrácia. Neskôr som pushol program, kde už sa rysovala aj ďalšia hierarchia. Následne som odovzdal plne fungujúcu trojitú hierarchiu, už s celkom fungujúcim GUI. Nakoniec som ešte odovzdával nejaké drobné úpravy. Je tam pred deadlinami veľa odovzdaní, pretože som chcel aby tam všetko bolo a vždy som na niečo zabudol.

# Spustenie programu

Pre začiatok vytvorte účet, poprípade sa prihláste menom: filipzubaj a heslom: filip. Ak však registrovanie a prihlasovanie, kvôli MySQL nebude fungovať, vyskúšajte postupovať podľa komentárov v Main triede. Program sa spúšta v Main triede.