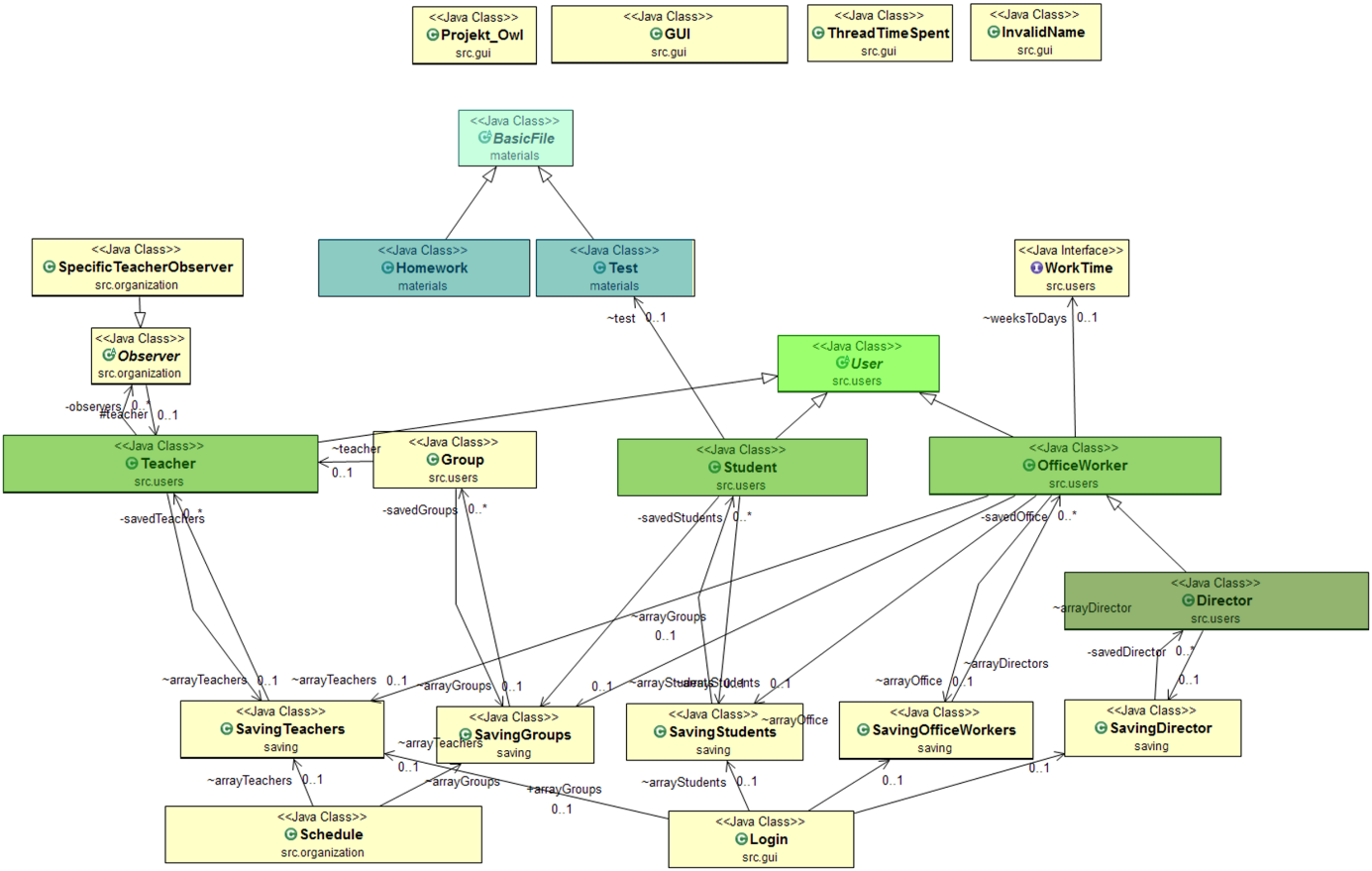
**Názov:** OWL (v preklade sova ako jeden zo symbolov múdrosti)

**Zámer projektu:**

Aplikácia má slúžiť ako podporný systém pri organizovaní doučovania základo a stredoškolského učiva. Pre žiakov resp. účastníkov týchto doučovaní bude poskytovať možnosť prihlásenia sa do systému, po ktorom si budú môcť pozrieť materiály a domáce úlohy, ktoré sú pre nich určené ako aj rozvrh hodín. Učitelia si zas dokážu pozrieť kto v ich skupine je a či si daná osoba pozrela zadanú domácu úlohu, ktorú tam lektor samozrejme aj s materiálmi vie pridať. Lektorovi tiež nebude chýbať schopnosť pozrieť sa na rozvrh hodín, ktoré má ešte odučiť. Nad lektormi je postavený administrátor. V jeho právomoci je pristupovať k zoznamu registrovaných žiakov, zatrieďovať ich do skupín a prideliť túto skupinku vybranému lektorovi. Vytváraním skupín s pridelenými časmi výučby sa generuje rozvrh. Administrátor má aj opačnú právomoc a to rušiť hodiny, skupiny alebo vymazávať žiakov spomedzi registrovaných. Posledný typ používateľa je riaditeľ, ktorý má rovnaké právomoci ako administrátor s tým, že je schopný pristupovať aj k zložke nazvanej „výplaty“. Ako to už z názvu vyplýva, tu bude prehľad kto a akú vysokú výplatu má dostať. Riaditeľ dokáže s výplatami aj pracovať a to pridať k nim bonusy alebo strhnúť, teda dať pokutu (napríklad za neskorý príchod lektora atď.).

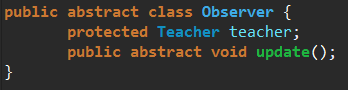
**UML diagram:**



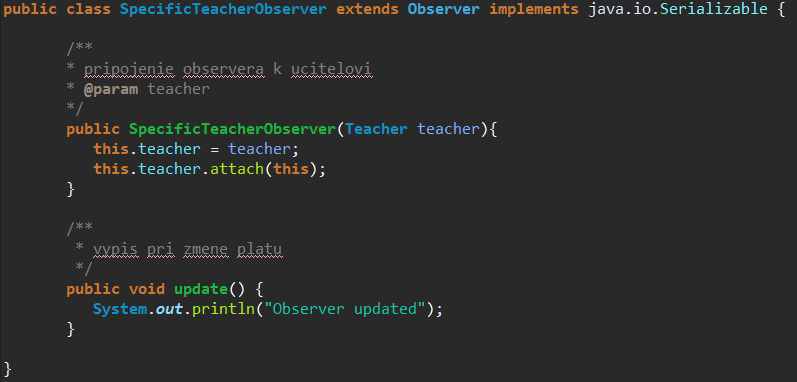
V projekte vystupujú dve hierarchie – používatelia a materiály. Ďalšou, menšou hierarchiou je aj observer, ktorý sleduje zmenu platu učiteľa. Prvá hierarchia používateľov začína klasou User, ktorú extendujú klasy Student, Teacher, OfficeWorker, z ktorého sa následne odvodí Director. Každý typ používateľa má špecifickú klasu, ktorá ho ukladá pomocou serializácie do súboru, resp. ho vie načítať do arrayListu. Student pracuje s materiálmi, no inak si okrem vypísania rozvrhu nevie nič zmeniť ani nastaviť. Na vypisovanie rozvrhov slúži klasa Schedule. Klasa Teacher má metodu pridávanie študentov no inak si tiež len dokáže vypísať rozvrh či prideliť žiakovi úlohu alebo test. Jeho odpracované hodiny sleduje observer a SpecificTeacherObserver, ktorý upovedomuje a uktualizuje veľkosť učiteľovej výplaty pri každom priradení skupiny. OfficeWorker, ako to je z UML diagramu zjavné, má najviac právomocí v rámci systému. Vie prideľovať žiakov a učiteľov do skupín, nastavovať detaily skupine alebo ich aj tvoriť. K nemu sa pripája interface WorkTime, ktorý prislúch ak použitému lambda výrazu. Nad ním, aj v hierarchií keďže ho extenduje, je Director. Jeho poslaním je delegovať prácu OfficeWorker a sledovať alebo meniť platy svojim zamestnancom. Druhou spomenutou hierarchiou sú materiály, ktoré sa oba odvádzajú od BasicFile a sú to Homework alebo Test, ktoré sú pripravené pre daného študenta. Pre rýchle, začiatočné vytváranie užívateľov slúži klasa Project\_OWL, ktorá však pri chode aplikácie nemá žiaden význam. Iné je to pri klase Login, ktorá hneď na úvode kontrolu správnosť prihlasovacích údajov tým, že ich porovnáva s uloženými údajmi o užívateľoch. Keď nájde zmenu, konkrétneho užívateľa prihlási. Rovnako pri spustení aplikácie sa naštartuje nová niť ThreadTimeSpent, ktorá počíta minúty strávené v aplikácií. Pri prihlasovaní sa ešte overuje správnosť zadaného užívateľského mena hľadaním zavináču, ktorý ak sa nenájde, je hodená vlastná výnimka InvalidName.

**Napĺňanie kritérií hodnotenia:**

1. Návrhový vzor observer (Observer a SpecificTeacherObserver)

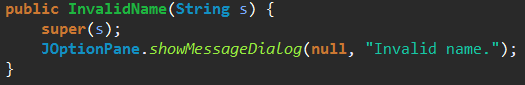
**package** src.organization **public** **abstract** **class** **Observer** 

**package** src.organization; **public** **class** **SpecificTeacherObserver**



1. Vlastná výnimka (InvalidName)

**package** src.gui; **public** **class** **InvalidName** **extends** **Exception**



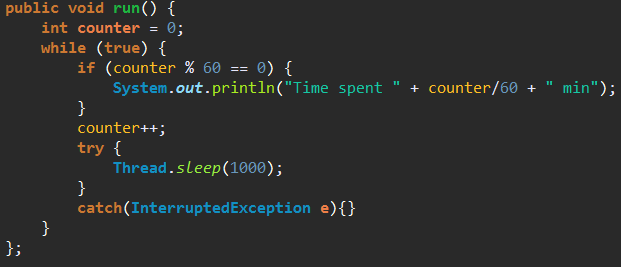


1. GUI

**package** src.gui; **public** **GUI**() **throws** **IOException**, **ClassNotFoundException**

1. Viacniťovosť (ThreadTimeSpent)

**package** src.gui;





1. Lambda výraz (WeeksToDays)

**package** src.users; **public** **class** **OfficeWorker** **extends** **User**





**Odovzdané pracovné verzie projektu:**

|  |  |
| --- | --- |
|  | * Pridané komenty pre Javadoc, finálna verzia * Vygenerovaný UML diagram, začiatok komentovania projektu * Projekt doplnený o druhú hierarchiu dokumentov * Opravená klasa Schedule, správne vypisovanie rozvrhu * Funkcia pridanie posielania taskov * Opravovanie metódy load, kvôli null pointer exception * Nesprávne ukladanie, null pointer exception |

|  |  |
| --- | --- |
|  | * Práca na GUI * Vypisovanie rozvrhov * Vylepšenie klasy Schedule * Pridaný observer * Pridávanie výplat za hodiny |

|  |  |
| --- | --- |
|  | * Zaciatok refaktorovanie kódu, vylepšenie ukladanie používateľov * Opravenie pôvodných chýb v projekte * Prvé commity, skúška práce s GITom |