Správa o realizácií projektu „Spaceship Tasker“

Objektovo orientované programovanie, FIIT STU

# Obsah

[Obsah 1](#_Toc419822901)

[Úvod 2](#_Toc419822902)

[Spresnenie rámcového zadania 2](#_Toc419822903)

[Štruktúra programu 3](#_Toc419822904)

[Kritériá hodnotenia 4](#_Toc419822905)

[Ďalšie kritériá 4](#_Toc419822906)

[Zoznam odovzdaných pracovných verzií programu 5](#_Toc419822907)

# Úvod

Toto je správa o realizácií môjho semestrálneho projektu z Objektovo orientovaného programovania s názvom „Spaceship Tasker“. Projekt bol vypracovaný podľa môjho spresnenia rámového zadania.

# Spresnenie rámcového zadania

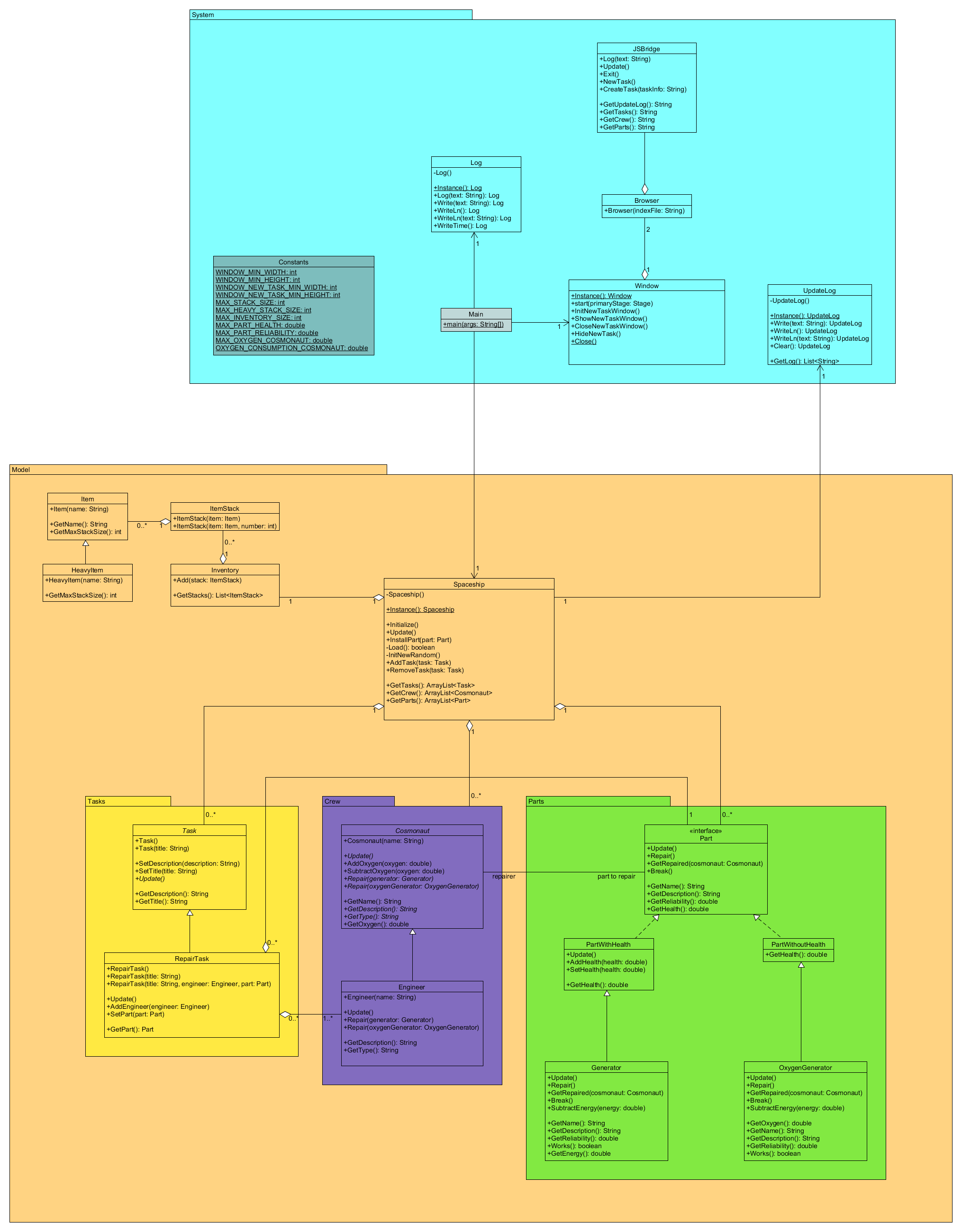
Program bude spravovať vesmírnu loď. Na vesmírnej lodi je veľa súčiastok, ktoré treba pravidelne kontrolovať a ktoré sa takmer pravidelne kazia. Na vesmírnej lodi sa tiež nachádzajú kozmonauti, rôzne suroviny a pomôcky. Problém spočíva v priradení správnych kozmonautov, pomôcok a surovín na kontrolu alebo opravu konkrétnej súčiastky.

Používateľ programu sa prihlasuje ako administrátor systému na vesmírnej lodi. Tento systém automaticky kontroluje súčiastky a hlási chyby alebo potrebné kontroly. Následne dovoľuje používateľovi automaticky alebo manuálne generovať úlohy. Tieto úlohy môžu byť priradené konkrétnym kozmonautom, môžu používať konkrétne pomôcky a suroviny a tiež mať priradený časový limit.

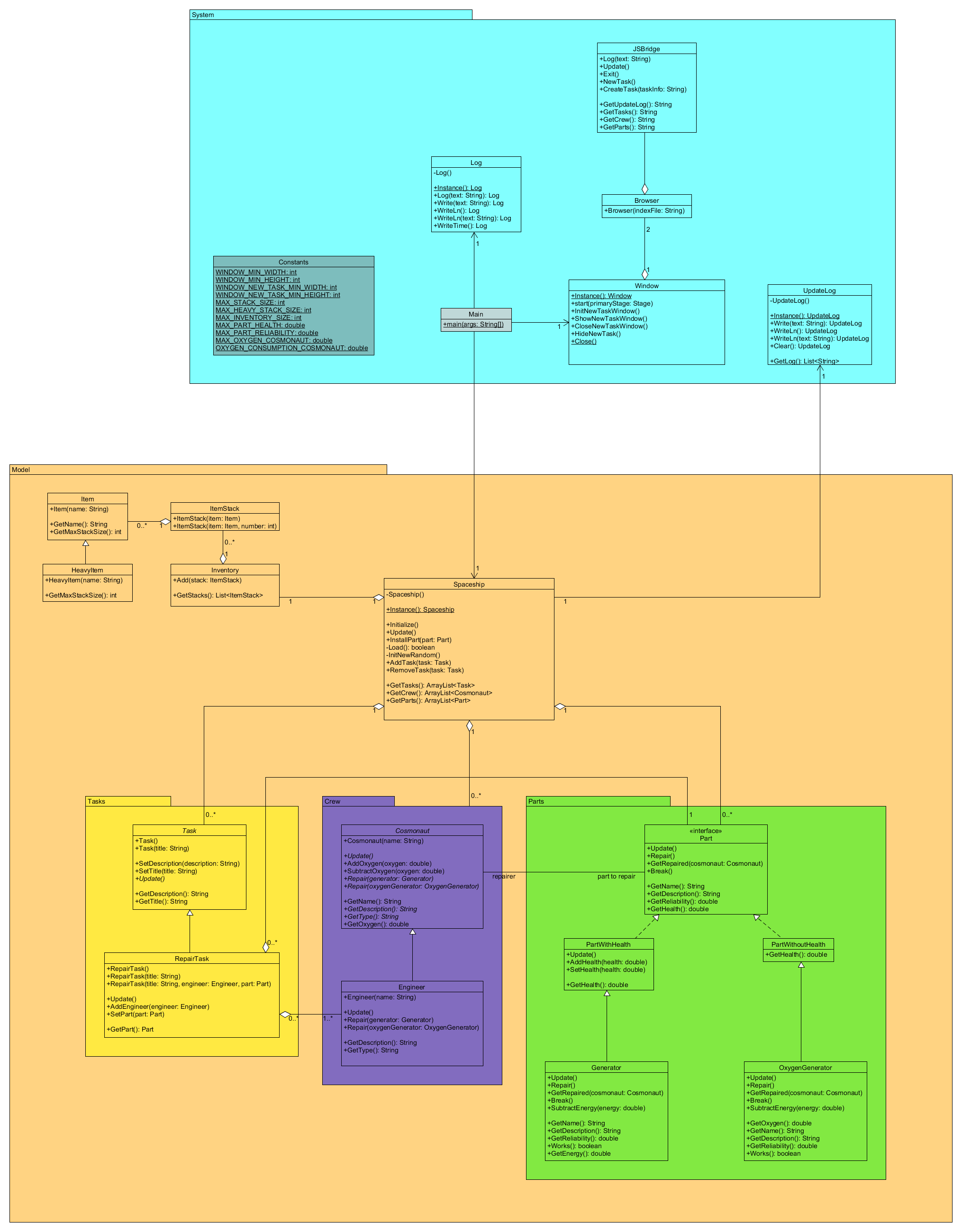
Viac informácií sa nachádza v súbore <Description.pdf>.

# Štruktúra programu

Hlavné triedy, ktoré zabezpečujú primárne chod programu, výpis do konzoly, konštanty a podobne sa nachádzajú v balíku chuckeles.sstasker.system.



Model programu tvoria triedy v balíku chuckeles.stasker.model.

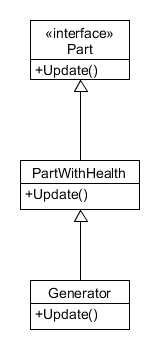


Kód pre grafické rozhranie je uložený v 2 balíkoch. V balíku chuckeles.sstasker.view je uložený kód pre hlavné rozhranie a v balíku chuckeles.sstasker.newtask je uložený kód pre rozhranie, ktoré slúži na vytvorenie novej úlohy. Všetok kód pre grafické rozhranie je napísaný pomocou webových technológií – HTML, CSS, Javascript, [Typescript](http://www.typescriptlang.org/) a používa primárne knižnicu [AngularJS](https://angularjs.org/).

Pre detailnejšie zobrazenie diagramu tried pozri súbor [Class Diagram.png](Class%20Diagram.png).

# Kritériá hodnotenia

Podľa [hodnotenia na stránke predmetu](http://www2.fiit.stuba.sk/oop/projekt/index.html#hodnotenie).

1. Program je funkčný, písaný v IntelliJ IDEA, zodpovedá rámcovému zadaniu a požiadavkám cvičiaceho.
2. Odovzdávaný zdrojový kód zahŕňa všetky potrebné súbory.
3. Program obsahuje zmysluplné dedenie a prekonávanie vlastných metód. Pozri napr. triedy Part, PartWithHealth a Generator.
4. Zapuzdrenie obsahuje takmer každá trieda.
5. Komentáre sú po celok zdrojovom kóde, vrátane JavaDoc komentárov a krátkych situačných komentárov.
6. JavaDoc je v priečinku <javadoc>, diagram tried je v súbore [Class Diagram.png](Class%20Diagram.png).

## Ďalšie kritériá

* V kóde je použité MVC, Singleton (Spaceship, Log) a Visitor(Part a spol. + Cosmonaut a spol.), pozri [diagram tried](Class%20Diagram.png).
* Kód je organizovaný do balíkov.
* V programe sú použité výnimky (pozri Inventory, InventoryFullException a JSBridge).
* Je poskytnuté grafické rozhranie oddelené od aplikačnej logiky.
* Sú použité lambda výrazy, generickosť (trieda Holder), odkazy na metódy.

# Zoznam odovzdaných pracovných verzií programu

* 5. cvičenie – Začal som s implementáciou tried System a Log
* 6. cvičenie – pridal som webview a jednoduchý html kód, ktorý bude moje GUIčko. Momentálne to ale úplne nefunguje a preto to musím trochu prerobiť
* 9. cvičenie – GUIčko už teraz aj niečo robí, ukazuje štatistiky a tak...
* 10. cvičenie – Začal som prácu na druhom okne, v ktorom sa budú vytvárať nové úlohy

Plný zoznam zmien je možné pozrieť [na Githube](https://github.com/chuckeles/sstasker/commits/master).