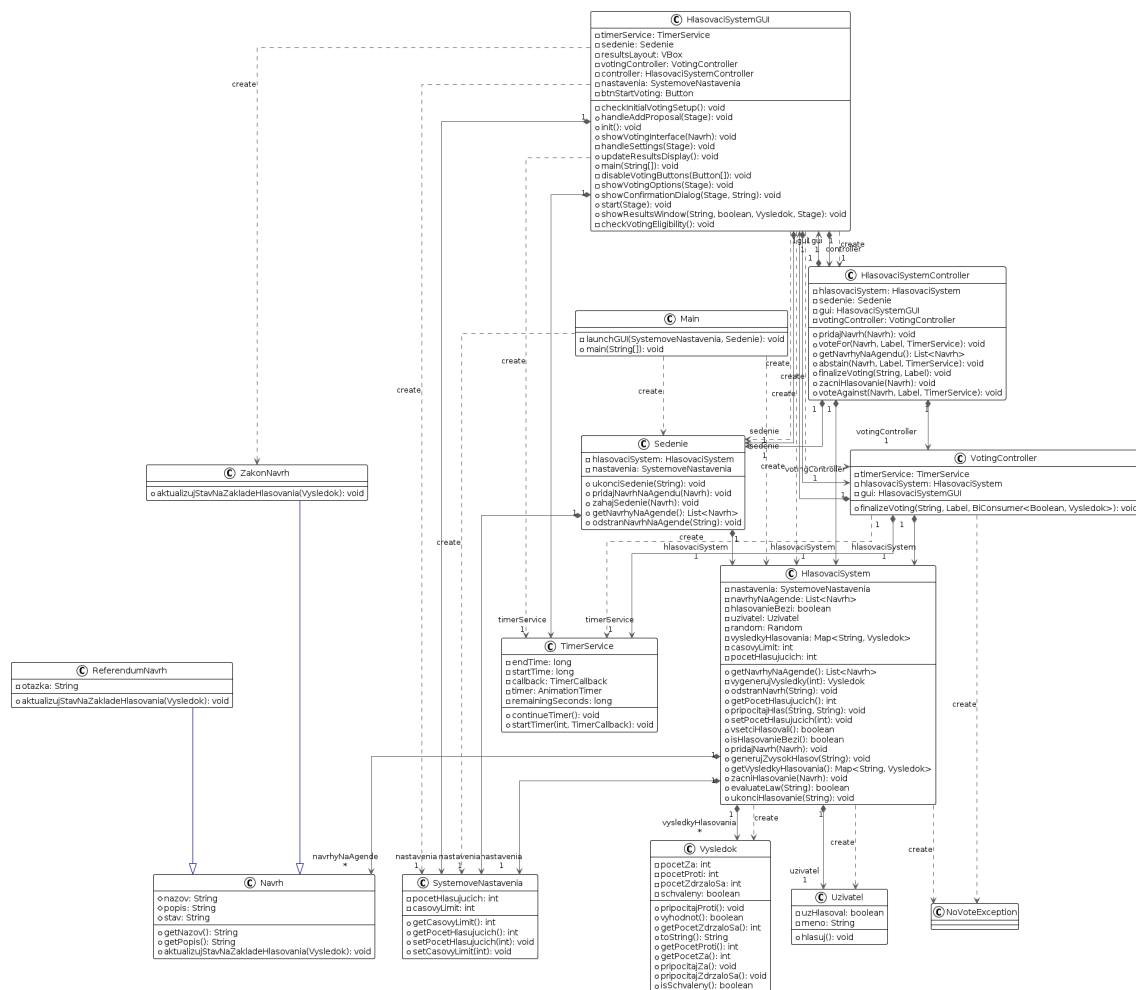


Elektronický Hlasovací Systém Národnej Rady

Cieľom projektu Elektronický Hlasovací Systém Národnej Rady (EHSNR) je vytvorenie jednoduchkej desktopovej aplikácie v jazyku Java, ktorá umožňuje užívateľom hlasovať v Národnej rade Slovenskej republiky. Aplikácia poskytuje rozhranie pre tvorbu návrhov, nastavenie počtu hlasujúcich a čas na hlasovanie, zobrazovanie vytvorených návrhov na hlasovanie a umožní užívateľom voliť medzi možnosťami "za", "proti" alebo "zdržať sa". Po každom hlasovaní systém vyhodnotí a zobrazí výsledky hlasovania vrátane počtu hlasov za, proti a počtu zdržaných hlasov. Akonáhle je dosiahnutá potrebná väčšina, aplikácia oznámi, či bol návrh prijatý alebo zamietnutý a koľko hlasovalo za akú možnosť. Po skončení hlasovania je možné vytvoriť nové návrhy alebo hlasovať o ďalších ak boli vytvorené, čo umožňuje ďalšiu diskusiu. Aplikácia bude obsahovať grafické rozhranie, kde bude vidieť varianty voľby a samotný návrh zákona.



Pre lepšie zobrazenie diagramu je aj v samostatnom súbore diagram.png.

Main Requirements

- meaningful inheritance and method overriding [✓]
 - Trieda ZakonNavrh dedí z triedy Navrh, overriding metodá - aktualizujStavNaZakladeHlasovania
- encapsulation [✓]
 - trieda SystemoveNastavenia využitie privátnych atribútov s využitím verejných metód na získavanie (atribút – casovyLimit, metóda – getCasovyLimit)
- sufficient code comments [✓]
- documentation [✓]
- Javadoc [✓]
- project realization report [✓]
- class diagram [✓]

Additional Requirements

- application of three design patterns (except Singleton) (3/3) [✓]
 - Observer - Time Service
 - Factory - trieda Vysledok na tvorbu Vysledku
 - MVC
- handling exceptional states using own exceptions [✓]
 - Trieda – NoVoteException
- graphical user interface separated from logic [✓]
 - Aplikačná Logika – triedy HlasovaciSystem, VotingController, Sedenie
 - Používateľské rozhranie - trieda HlasovaciSystemGUI
 - Controller – trieda HlasovaciSystemController
- explicit use of multithreading []
- use of generics in own classes []
- explicit use of RTTI []
- nested classes and interfaces []
- lambda expressions or method references [✓]
 - Trieda - HlasovaciSystem – metóda – vsetciHlasovali
- default method implementation in interfaces []
- aspect-oriented programming (AspectJ) []
- serialization []