

Kisil Yulian id:128371

**Objektovo-orientované programovanie**

**Semestrálny projekt**

31.04.2024

**Názov projektu:** “Hlasovací systém pre reštaurácie a ich hodnotenie”

**Zámer projektu**

Zámer môjho projektu je vytvoriť hlasovací systém, ktorý umožní používateľom hlasovať za reštaurácie a hodnotiť ich podľa rôznych parametrov. Pomôže vytvoriť pozitívne prostredie, v ktorom je každý názor dôležitý a každý používateľ sa môže slobodne podeliť o svoj názor na podnik a pomôcť ostatným pri informovanom rozhodovaní o výbere miesta na večeru alebo podujatie. Systém umožní používateľom rýchlo a pohodlne určiť obľúbenosť a kvalitu služieb reštaurácií. Systém tiež umožňuje používateľom zobraziť rôzne aspekty reštaurácie, napríklad všeobecné informácie o reštaurácii, jej menu a bar. Platforma je navrhnutá tak, aby využila silu komunity na zlepšenie kvality služieb v reštauračnom priemysle a na posilnenie vzťahu medzi zákazníkmi a reštaurátormi. Umožňuje každému cítiť sa súčasťou spoločného poslania zlepšiť zážitok z jedla a pôžitok z neho.

Medzi hlavné funkcie systému patria

1. Registrácia používateľov a hlasovacie práva.

2. Zobrazenie zoznamu reštaurácií na hlasovanie.

3. Hlasovanie za reštauráciu a určenie jej hodnotenia na základe takých aspektov, ako sú dojmy zákazníkov z podniku, informácie o podniku, kvalita menu atď.

4. Zhrnutie výsledkov hlasovania a zobrazenie hodnotenia reštaurácií v rôznych kategóriách.

5. Vykonávanie nových hlasovaní a ukladanie a obnovovanie starých hlasovaní.

Cieľom tohto projektu je uľahčiť používateľom výber reštaurácie a zlepšiť kvalitu služieb v reštauráciách prostredníctvom zhromažďovania a analýzy hlasov používateľov.

**Štruktúru systému**

A screenshot of a computer

Description automatically generated

**VotingSystem**- táto trieda je hlavnou triedou pri spustení aplikácie. Ukladajú sa v nej používatelia, ktorí sa zúčastňujú na hlasovaní, reštaurácie a kaviarne, ktoré sa na ňom zúčastňujú. Ukladá aj rôzne údaje, napríklad počet používateľov, podniky, polia, ktoré uchovávajú hodnotenia podnikov a to, či za ne hlasovali. Existujú aj metódy, ktoré poskytujú prístup k týmto parametrom, pretože sú chránené. Existujú metódy, ktoré kontrolujú, či je používateľ v systéme, na počítanie hlasov, na registráciu, na vytváranie štatistík pre administrátora, na ukladanie, mazanie a obnovovanie hlasov, na výpočet priemerného počtu hlasov a na to, či je heslo vhodné na bežnú registráciu.

**ConEvltOfEstabl conForLogin conForVotingPlace ConOvrvOfEstab ConStatisticsEctb** -sú triedy kontrolérov, ktoré tvoria interakciu medzi grafickým rozhraním a modelom (VotingSystem a iné triedy)

**GuiForProgram** - trieda, ktorá vykonáva náš program.

**checkingForSymbols** -trieda, ktorej výnimka sa vyhodí, ak heslo neobsahuje aspoň jedno písmeno

**User(SimpleVoter, ExpertVoter, Host)** - je hierarchia, ktorá implementuje používateľské prostredie v systéme. Má hlasujúcich, ako sú **SimpleVoter** a **ExpertVoter**, ktorí sa líšia silou svojich hlasov. Taktiež trieda **Host** je hlavnou triedou v systéme. Môže vidieť, ako prebieha hlasovanie, a ukladať, mazať a vytvárať nové. tieto triedy komunikujú s triedou VotingSystem. Objekty, ktoré sú vytvorené na základe týchto tried, sú uložené v tejto triede a zúčastňujú sa na vykonávaní programu

**Establishment(Restaurant,** **Cafe)**- Táto hierarchia predstavuje triedy reštaurácií a kaviarní, ktoré implementujú metódy rozhrania. Objekty, ktoré sú vytvorené na základe týchto tried, sú uložené v tejto triede a zúčastňujú sa na vykonávaní programu. Kaviarne majú agregácie **Review** a **Menu**, ktoré ukladajú hodnotenia pre každú kaviareň a menu pre každú kaviareň. reštaurácie majú agregácie **Review,Menu a Bar**, ktoré vykonávajú úlohy ukladania údajov o hodnoteniach menu a barov reštaurácií.

**middleVoiceAgr**- trieda, ktorá uchováva priemerný hlas voliča pre všetky reštaurácie a pre všetky kaviarne. vykonáva metódy rozhrania **middleVoice**

**Review,Menu ,Bar-** agregácie ktoré vykonávajú úlohy ukladania údajov o hodnoteniach menu a barov reštaurácií.

**DishUnit, DishSquadSetup,DishSquad,Dish** - tento systém používa vzor Composite na vytvorenie dátovej štruktúry, ktorá umožňuje reprezentovať jednotlivé jedlá aj skupiny jedál ako jednotlivé objekty. To uľahčuje manipuláciu a interakciu s jedlami a ponukami

**calculateMidleVoiceCafe,calculateMidleVoiceRest,middleVoiceStrategy**- Tieto triedy predstavujú vzor Stratégia v kontexte výpočtu priemerného hlasovania používateľov pre zariadenia (reštaurácie a kaviarne).Obe triedy implementujú rozhranie **middleVoiceStrategy**, takže majú metódu **calculateMidleVoice**, ktorá vypočíta priemerné hlasovanie pre vybraný typ zariadenia na základe vstupných údajov z hlasovacieho systému. Tento prístup umožňuje použiť rôzne stratégie výpočtu priemerného hlasu v závislosti od toho, ktorú stratégiu zvolíme v triede **VotingSystem**

**VotersFollower,ObserverMiddleVoice,ObserverMiddleVoiceAllVoters**(AspectJ sa v grafe nezobrazuje ) - Tento kód implementuje vzor Observer na sledovanie a reagovanie na aktivitu voličov v hlasovacom systéme. Tento kód funguje takto:

1.Po vytvorení nového objektu "VotingSystem" aspekt "**ObserverMiddleVoiceAllVoters**" automaticky inicializuje pozorovateľa "**ObserverMiddleVoice**" a pridá ho do zoznamu pozorovateľov "**VotingSystem**".

2. Keď prejdete na stránku so štatistikami alebo vyhodnotením, zavolajú sa aspekty "pred" a "po", aby sa k pozorovateľovi pridali príslušné ovládače a aktualizovali informácie podľa aktivity voličov.

3.Pozorovateľ reaguje na udalosti a aktualizuje informácie na stránkach štatistík a vyhodnotenia výpočtom priemerných hlasov a aktualizáciou príslušných kontrolérov

**Ako plní [kritériá hodnotenia](http://www2.fiit.stuba.sk/~vranic/oop/projekt/index.html" \l "hodnotenie)**

Aplikácia má dve hierarchie tried, napríklad Používatelia a Establishment. Používajú dedičnosť. Príkladom dvoch vetiev hierarchie, ktoré ukazujú dedičnosť tried, sú:

A screenshot of a computer program

Description automatically generatedA screenshot of a computer program

Description automatically generated

Príklad použitia polymorfizmu v týchto triedach(prekonaná je metoda napr):

A screenshot of a computer program

Description automatically generated

Príkladom agregácie v dvoch vetvách hierarchie je(reštaurácie majú tiež agregačný bar a menu):

A screenshot of a computer screen

Description automatically generated

Tiež príklady použitia rozhrania:  
A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer program

Description automatically generated Toto rozhranie má metódu, ktorá bude implementovaná v triedach, ktoré ho dedia(Cafe, Restaurant)

Každý parameter triedy je zapuzdrený. To je vidieť na predchádzajúcich fotografiách. Tieto triedy používajú metódy set,get na prístup k celej množine parametrov. Všetky triedy sú tiež umiestnené v organizovaný balíky.

**Ďalšie kritériá zahŕňajú:**

1. Použitie Serializacie . Na ukladanie a obnovu hlasovania

**A screenshot of a computer program

Description automatically generatedA screenshot of a computer program

Description automatically generated**

2. používanie RTTI . Používa sa na určenie, či je volič jednoduchým voličom alebo odborníkom na bežné sčítanie hlasov.

A screenshot of a computer program

Description automatically generated

3.Poskytnutie grafického používateľského rozhrania oddelene od aplikačnej logiky.

A screenshot of a computer

Description automatically generated

4. použitie aspektovo-orientovaného programovania (AspectJ). Použitý so vzorom Observer

A screenshot of a computer program

Description automatically generated

5. Ošetrenie mimoriadnych stavov prostredníctvom vlastných výnimiek. Slúži na zistenie, či používateľ počas registrácie zadal aspoň jedno písmeno v hesle.

A screenshot of a computer program

Description automatically generated

6. Explicitné použitie viacniťovosti (multithreading). Slúži na zobrazenie informácií v samostatnom prude

A screen shot of a computer program

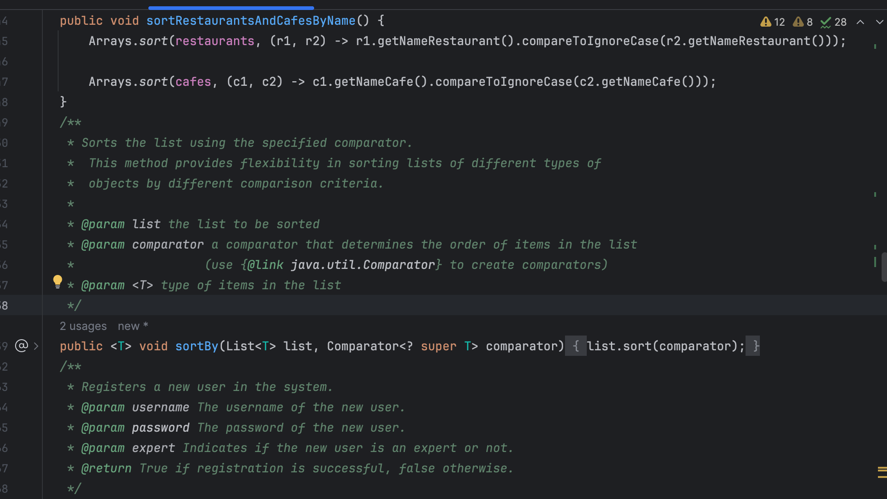
Description automatically generated

7. Použitie lambda výrazov.Slúži na zoradenie array kaviarní a reštaurácií podľa názvu

A computer screen shot of a program

Description automatically generated

8. použitie generickosti vo vlastných triedach. Slúži na triedenie používateľov podľa mena



9. Použitie troch vzorov je vidieť na fotografii vyššie: Composite, Stratégia, Observer. Observer tiež využíva na svoju prácu AspectJ. Zobrazuje, ako sa mení priemerný hlas všetkých hlasujúcich a priemerný hlas pre konkrétneho hlasujúceho. Stratégia sa používa na výpočet priemerného hlasu pre kaviareň alebo reštauráciu. Composite sa používa na vytvorenie menu pre reštaurácie a kaviarne.

**Zoznam hlavných verzií programu odovzdaných do GitHub**

**1. versia**1: Vytvorené hlavné triedy a metódy, ktoré boli plánované v ‘Zámer projektu’

**2. Pracovna verzia programu**. Vytvoril pracovnú časť projektu a opravil chyby, ktoré boli v predchádzajúcej verzii

**3. konečná verzia programu**. Pridané dodatočné kritériá, ktoré boli nastavené