|  |  |  |
| --- | --- | --- |
|  | | |
| **VILNIAUS KOLEGIJA**  **ELEKTRONIKOS IR INFORMATIKOS FAKULTETAS**  **PROGRAMINĖS ĮRANGOS KATEDRA** | | |
|  | | |
| **ANTROJI PROGRAMAVIMO PRAKTIKA** | | |
| PD - 6531BX028 PI19B | | |
| STUDENTAS |  | LUKAS SONGULIJA |
|  | (PARAŠAS) |  |
|  | 2020 lapkritis |  |
| DĖSTYTOJA |  | VAIDAS LIUBINAS |
|  | (PARAŠAS) |  |
|  | 2020 lapkritis |  |
|  |  |  |
|  | Vilnius 2020 |  |
|  |
|  |

**TURINYS**

ILIUSTRACIJŲ SĄRAŠAS 3

ĮVADAS 4

1. DUOMENŲ BAZĖS PROJEKTAVIMAS 5

2. PROGRAMOJE NAUDOTOS KLASĖS 5

3. SQL FUNKCIJOS 8

4. NAUDOTOJO INSTRUKCIJA 10

5. IŠVADOS 20

# ILIUSTRACIJŲ SĄRAŠAS

[Pav. 1.1. Duomenų bazės projektavimas. 5](#_Toc57332220)

[Pav. 2.1. Subject klasė. 5](#_Toc57332221)

[Pav. 2.2. Group klasė. 6](#_Toc57332222)

[Pav. 2.3. Marks klasė. 6](#_Toc57332223)

[Pav. 2.4. Student klasė 7](#_Toc57332224)

[Pav. 2.5. Staff klasė. 7](#_Toc57332225)

[Pav. 2.6. SujbectTeacher klasė. 8](#_Toc57332226)

[Pav. 3.1. LoginStudent funckija. 9](#_Toc57332227)

[Pav. 3.2. DeleteStudent funkcija. 10](#_Toc57332228)

[Pav. 3.3. DisplayStudentMarks funkcija. 11](#_Toc57332229)

[Pav. 4.1. LoginScreen. 12](#_Toc57332230)

[Pav. 4.2. Loggined Student. 13](#_Toc57332231)

[Pav. 4.3. StaffLogin. 13](#_Toc57332232)

[Pav. 4.4. DiplayMarks. 14](#_Toc57332233)

[Pav. 4.5. Add Mark. 15](#_Toc57332234)

[Pav. 4.6. Update Mark. 16](#_Toc57332235)

[Pav. 4.7. Edit buttons. 16](#_Toc57332236)

[Pav. 4.8. Display Students. Display Groups. 17](#_Toc57332237)

[Pav. 4.9. Register Student. 17](#_Toc57332238)

[Pav. 4.10. Register Staff. 18](#_Toc57332239)

[Pav. 4.11. Add Group. 19](#_Toc57332240)

[Pav. 4.12. Delete Group. 20](#_Toc57332241)

[Pav. 4.13. Display Subjects. Display SubjectTeacher. 20](#_Toc57332242)

[Pav. 4.14. Add Subject. 21](#_Toc57332243)

[Pav. 4.15. Delete Subject. 22](#_Toc57332244)

[Pav. 4.16. Add Subject Teacher. 23](#_Toc57332245)

[Pav. 4.17. Delete Subject Teacher. 24](#_Toc57332246)

# ĮVADAS

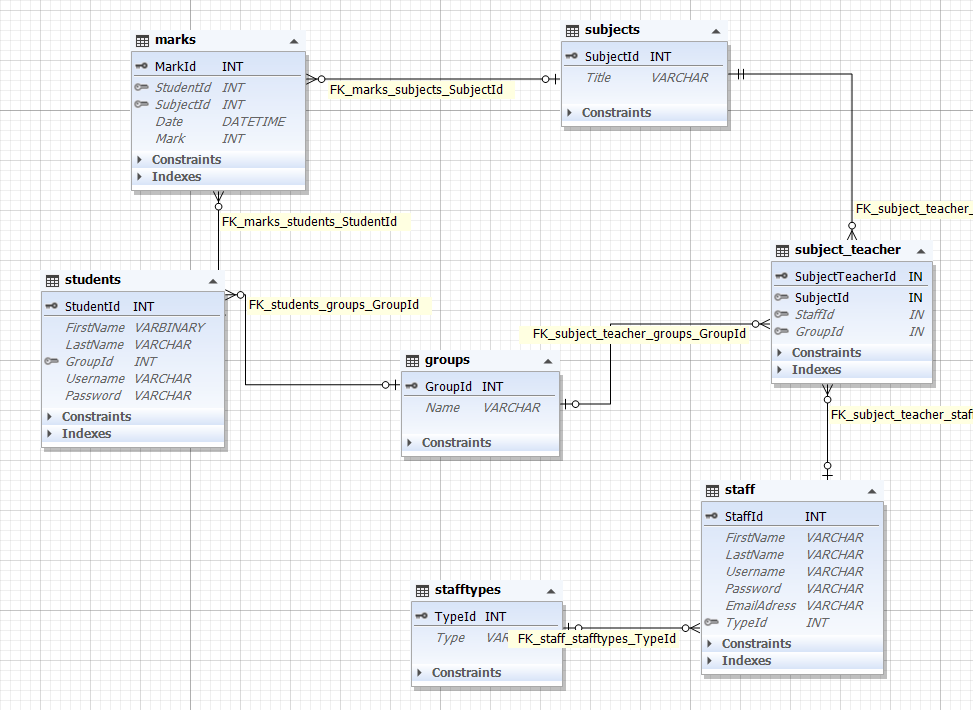
**Darbo tikslas:**

Sukurti akademinę sistemą, kurioje būtų fiksuojami studentų vertinimai už dalykus. Programoje yra trys naudotojų lygiai: administratorius, dėstytojas, studentas.

**Darbo uždaviniai:**

1. Administratorius gali valdyti (kurti ir šalinti) studentų grupes, dėstomus dalykus, dėstytojus ir studentus. Dėstytojus priskirti prie dėstomo dalyko, studentus ir dėstomą dalyką - prie grupės.
2. Dėstytojas gali įvesti ir redaguoti pasirinkto studento dalyko pažymį.
3. Studentas gali tik peržiūrėti savo pažymius.
4. Administratorius registruodamas studentą ar dėstytoją, automatiškai prisijungimo vardu priskiria vardą, o slaptažodžiu – pavardę.

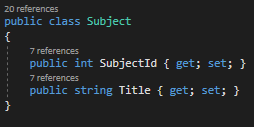
# DUOMENŲ BAZĖS PROJEKTAVIMAS



Pav. 1.1. Duomenų bazės projektavimas.

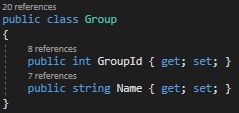
.

# PROGRAMOJE NAUDOTOS KLASĖS



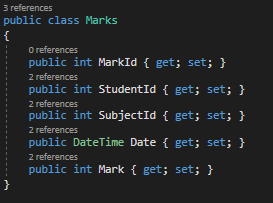
Pav. 2.1. Subject klasė.

Klasė Subject(Dalykas) turi du atributus: SubjectId ir Title. Norėdami įterpti į duomenų bazę naują eilutę su Subject informacija, pirmiausia turėtume susikurti Subject objektą, priskirti jo atributam reikšmes. Tada mes galėtume išsikviesti SQL funkcijas kad įterpti naują Subject(dalyką) į duomenų bazę arba ištrinti iš jos naudojant Subject atributų reikšmes. SubjectId yra nurodytas DB lentelėje kaip PK(pagrindinis raktas).



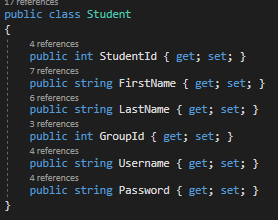
Pav. 2.2. Group klasė.

Klasė Group(Grupė) turi du atributus: GroupId ir Name. Norėdami įterpti į duomenų bazę naują eilutę su Group informacija, pirmiausia turėtume susikurti Group objektą, priskirti jo atributam reikšmes. Tada mes galėtume išsikviesti SQL funkcijas kad įterpti naują Group(grupę) į duomenų bazę arba ištrinti iš jos naudojant Group atributų reikšmes. GroupId yra nurodytas DB lentelėje kaip PK(pagrindinis raktas).



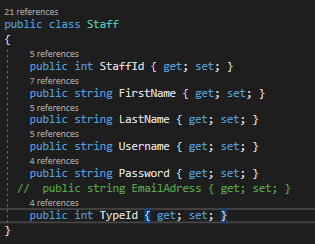
Pav. 2.3. Marks klasė.

Klasė Marks(pažymiai) turi 5 atributus: MarkId,StudentId,SubjectId,Date,Mark. Norėdami įterpti į duomenų bazę naują eilutę su Marks(pažymių) informacija, pirmiausia turėtume susikurti Marks objektą, priskirti jo atributam reikšmes. Tada mes galėtume išsikviesti SQL funkcijas kad įterpti naują Marks objektą į duomenų bazę arba ištrinti iš jos naudojant Marks atributų reikšmes. StudentId ir SubjectId DB lentelėje yra nurodyti kaip FK(išoriniai raktai). MarkId yra nurodytas DB lentelėje kaip PK(pagrindinis raktas).



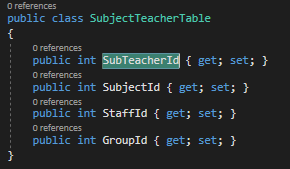
Pav. 2.4. Student klasė

Klasė Student (studentai) turi 6 atributus: StudentId,FirstName,LastName,GroupId,Username,Password. Norėdami įterpti į duomenų bazę naują eilutę su Student (studentai) informacija, pirmiausia turėtume susikurti Student objektą, priskirti jo atributam reikšmes. Tada mes galėtume išsikviesti SQL funkcijas kad įterpti naują Student objektą į duomenų bazę arba ištrinti iš jos naudojant Student atributų reikšmes. GroupId yra DB lentelėje nurodytas kaip FK(išorinis raktas). StudentId yra nurodytas DB lentelėje kaip PK(pagrindinis raktas).



Pav. 2.5. Staff klasė.

Klasė Staff (darbuotojai) turi 6 atributus: StaffId,FirstName,LastName,Username,Password,TypeId. Norėdami įterpti į duomenų bazę naują eilutę su Staff (darbuotojai) informacija, pirmiausia turėtume susikurti Staff objektą, priskirti jo atributam reikšmes. Tada mes galėtume išsikviesti SQL funkcijas kad įterpti naują Staff objektą į duomenų bazę arba ištrinti iš jos naudojant Staff atributų reikšmes. TypeId yra DB lentelėje nurodytas kaip FK(išorinis raktas). StaffId yra nurodytas DB lentelėje kaip PK(pagrindinis raktas).

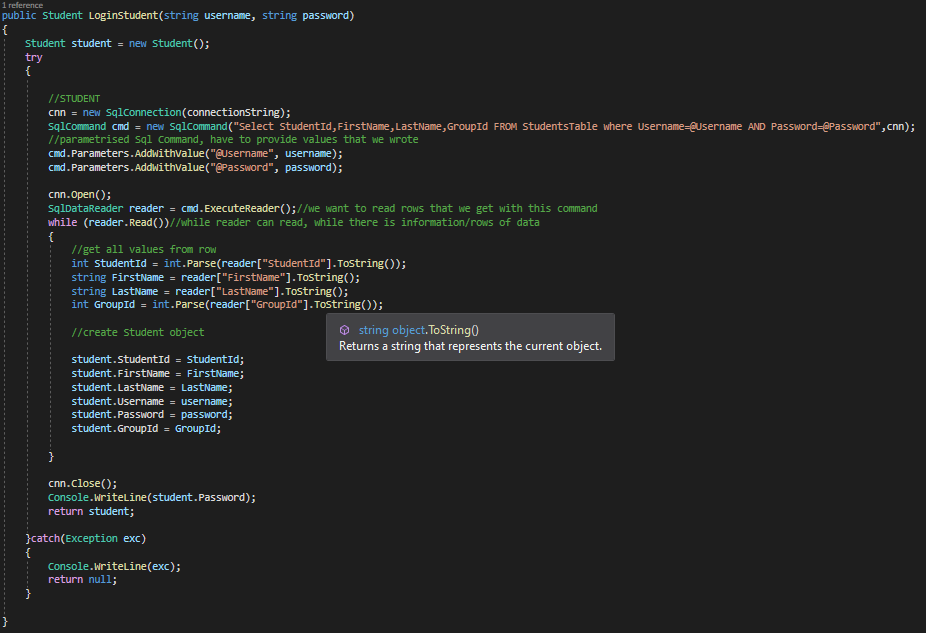


Pav. 2.6. SujbectTeacher klasė.

Klasė SubjectTeacher turi 4 atributus: SubTeacherId,SubjectId,StaffId,GroupId. Norėdami įterpti į duomenų bazę naują eilutę su SubjectTeacher informacija, pirmiausia turėtume susikurti SubjectTeacher objektą, priskirti jo atributam reikšmes. Tada mes galėtume išsikviesti SQL funkcijas kad įterpti naują SubjectTeacher objektą į duomenų bazę arba ištrinti iš jos naudojant SubjectTeacher atributų reikšmes. SubjectId, StaffId ir GroupId yra DB lentelėje nurodyti kaip FK(išoriniai raktai). SubTeacherId yra nurodytas DB lentelėje kaip PK(pagrindinis raktas).

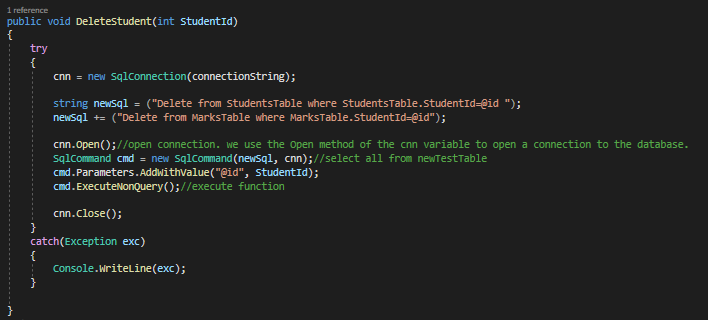
# **SQL FUNKCIJOS**

Funkcija LoginStudent reikalinga tam, kad studentas galėtų prisijungti prie programos. Kviečiant šitą funkciją mes turėtume jai duoti username ir password. Visą kodą rašome į Try and Catch blokus kadangi dirbant su duomenų bazėmis gali atsirasti pakankamai nemažai klaidų, atsiradus klaidai catch blokas ją sugaus ir leis mums žinoti kokia klaida mes padarėme. Pirmiausia atsidarome naują SqlConnection, turime į jį duoti connectionString kuris yra mūsų sujungimo „pavadinimas“. Jis yra būtinas, kad prisijungti prie duomenų bazės. Susikuriame string newSql, kuriame parašome iš kokių lentelių mes norime ištrinti. SQL komandoje parošome gauti lentelės StudentsTable nurodytus duomenis kur Username ir Password sutampa su mūsų ivęstais username ir password. Tai yra parametrizuota užklausa. Nurodome kam lygios Username ir Password reikšmės. SqlDataReader padeda mums nuskaityti funkcijos gautus duomenis/eilutes. Jeigu duomenis gauname įsidedame juos į naujai sukurtą Student objektą. Jeigu mes sėkmingai gauname visus duomenis po Catch bloko naudojant return gražiname atgal užpildytą Student objektą. Kuris parodo, jog sėkmingai gavome visus duomenis, o tai reiškia jog parašyti username ir password sutapo su duomenų bazėje esančiais. Įvykdę komandą ir gavę reikalingas reikšmes galime uždaryti Sql sujungimą.



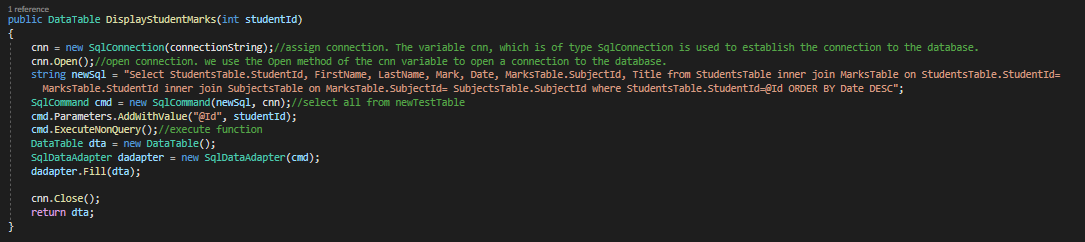
Pav. 3.1. LoginStudent funckija.

Funkcija DeleteStudent reikalinga tam, kad ištrinti Studentą pagal pasirinktą StudentId iš duomenų bazės lentelės. Visą kodą rašome į Try and Catch blokus kadangi dirbant su duomenų bazėmis gali atsirasti pakankamai nemažai klaidų, atsiradus klaidai catch blokas ją sugaus ir leis mums žinoti kokia klaida mes padarėme. Pirmiausia atsidarome naują SqlConnection, turime į duoti connectionString kuris yra mūsų sujungimo „pavadinimas“. Jis yra būtinas, kad prisijungti prie duomenų bazės. Susikuriame string newSql, kuriame parašome iš kokių lentelių mes norime ištrinti. Kadangi mūsų lentelės StudentsTable ir MarksTable yra susietos mes turime ištrinti iš jų abiejų. Ištrinant studentą turi išsitrinti ir visi jo pažymiai. Vėliau atsidarome mūsų sqlConnection ir įvykdome komandą. Įvykdę komandą galime uždaryti Sql sujungimą.



Pav. 3.2. DeleteStudent funkcija.

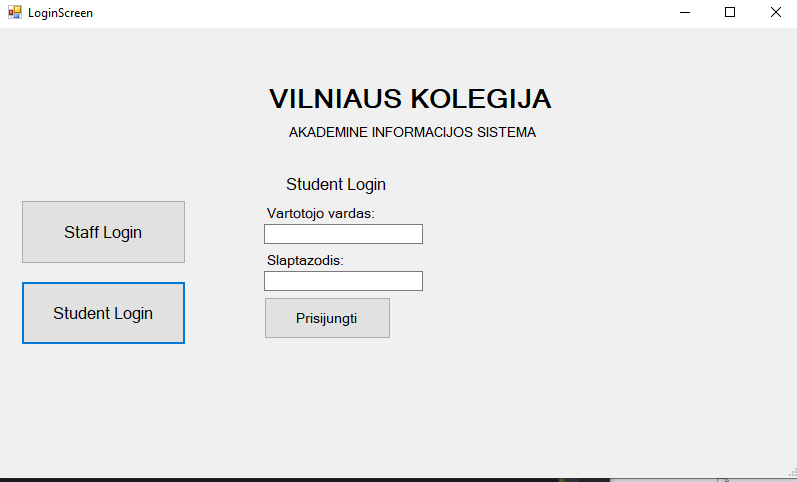
Funkcija DisplayStudentMarks reikalinga tam, kad gauti Studento pažymius iš duomenų bazės pagal studentId. Visą kodą rašome į Try and Catch blokus kadangi dirbant su duomenų bazėmis gali atsirasti pakankamai nemažai klaidų, atsiradus klaidai catch blokas ją sugaus ir leis mums žinoti kokia klaida mes padarėme. Pirmiausia atsidarome naują SqlConnection, turime į duoti connectionString kuris yra mūsų sujungimo „pavadinimas“. Jis yra būtinas, kad prisijungti prie duomenų bazės. Susikuriame string newSql, kuriame parašome iš kokių lentelių mes norime gauti reikšmes. Pirmiausia mes norime gauti iš lentelės StudentsTable, gauti StudentId,FirstName,LastName. Mark,SubjectId ir Date gauname iš lentelės Marks. Title gauname iš lentelės SubjectsTable. Inner join naudojame kad susieti lenteles. Taip gauname visus Marks(pažymius) kažkokiam studentui. Atgal gražiname DataTable kuri bus užpildyta duomenimis kuriuos mes gauvome įvykde Sql kodą. Įvykdę komandą galime uždaryti Sql sujungimą.



Pav. 3.3. DisplayStudentMarks funkcija.

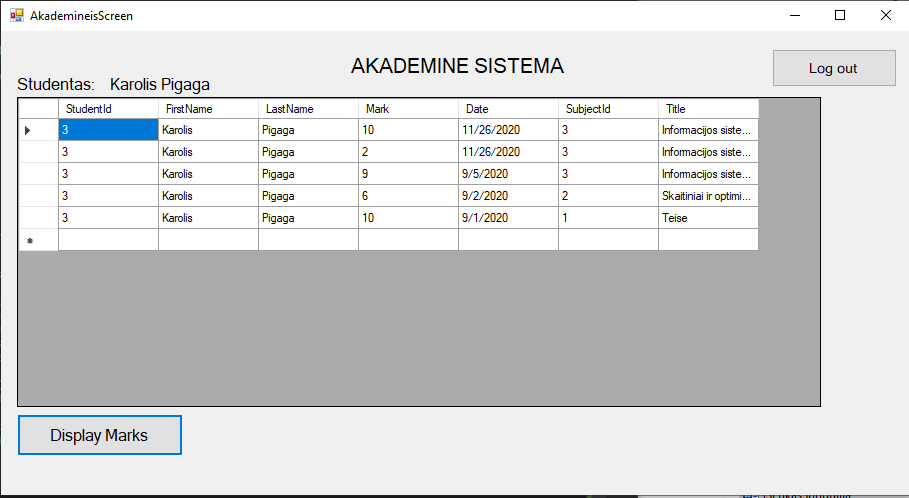
# NAUDOTOJO INSTRUKCIJA

Užėję į programą turite du pasirinkimus: prisijungti kaip Student(studentas) arba prisijungti kaip Staff(darbuotojas). Paspaudę „Student Login“ mygtuką atsiras naujas langas į kurį galėsime įvesti savo prisijungimo duomenis, ir paspausti mygtuką „Prisijungti“.



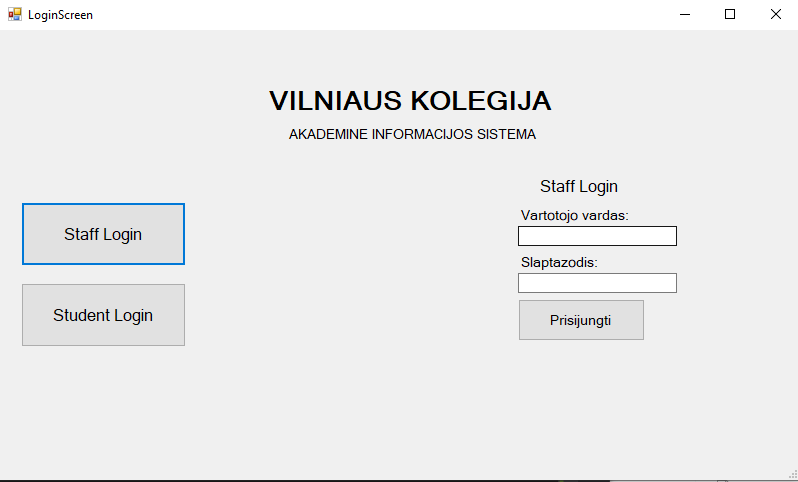
Pav. 4.1. LoginScreen.

Jeigu duomenis Username ir Password sutampa su duomenų bazėje esančiais duomenimis mums atsidaro langas kur parašyti visi studento pažymiai. Surašyti mažėjimo tvarka, eina nuo naujausiai parašytų pažymių iki seniausiai parašytų.



Pav. 4.2. Loggined Student.

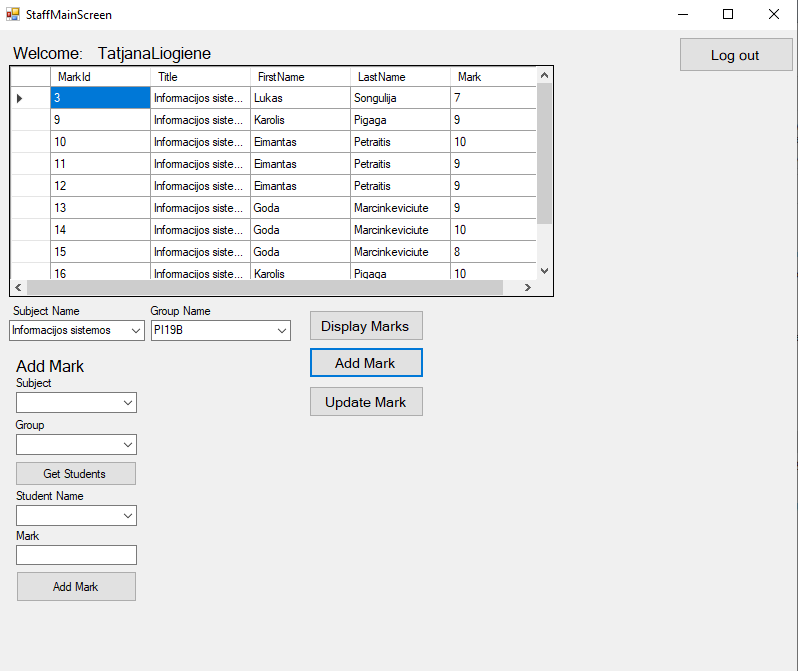
Užėję į programą turite du pasirinkimus: prisijungti kaip Student(studentas) arba prisijungti kaip Staff(darbuotojas). Paspaudę „Staff Login“ mygtuką atsiras naujas langas į kurį galėsime įvesti savo prisijungimo duomenis, ir paspausti mygtuką „Prisijungti“.



Pav. 4.3. StaffLogin.

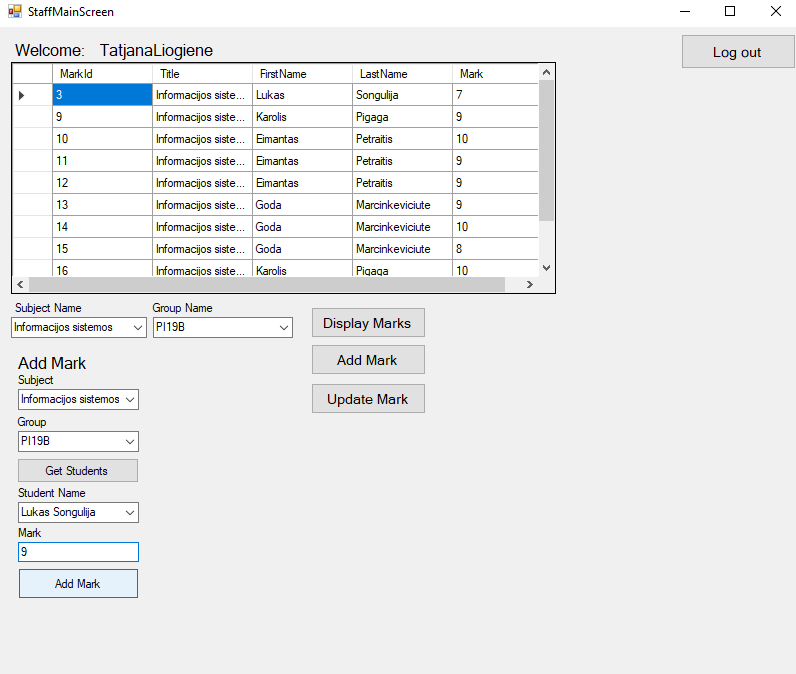
Staff. Staff Type

Jeigu duomenis Username ir Password sutampa su duomenų bazėje esančiais duomenimis mums atsidaro langas kur yra didelis langas kuriame galima pažiūrėti surašytus pažymius pasirinkus Subject Name(dalyko pavadinimą) ir Group Name(grupės pavadinimą). Jie pasikrauna automatiškai kiekvienam dėstytojui. Tai pasirinkę Subject Name ir Group Name ir paspaudę mygtuką „Display“ mums langelyje parodys visus tos grupės ir to dalyko pažymius kuriuos esame surašę.



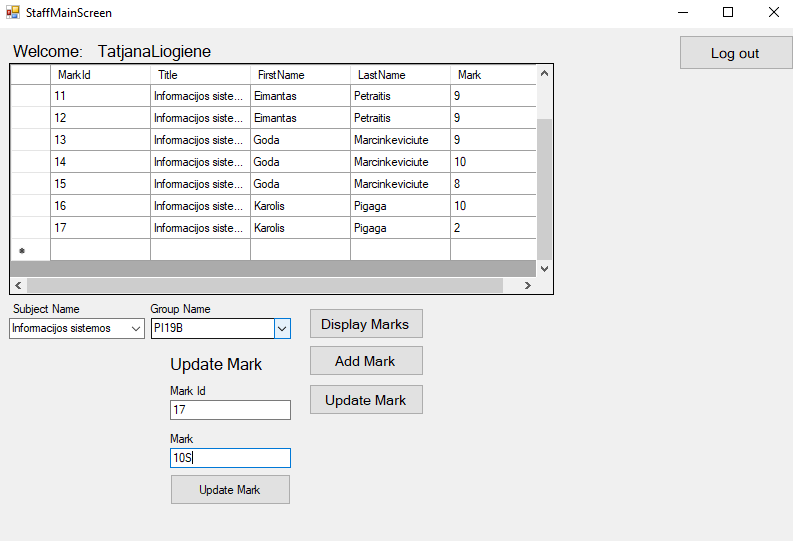
Pav. 4.4. DiplayMarks.

Paspaudę mygtuką dešinėje pusėje „Add Mark“ mums atsidaro naujas langas, kad pridėti naują pažymį. Galime pasirinkti iš mūsų turimų Subject(dalykų), Group(grupių) ir Student(studentų). Pasirenkame kam norime įrašyti pažymį. Subject,Group ir Students pasirinkimai atsiranda automatiškai pagal dėstytoją. Viską pasirinkę ir irašę Mark(pažymį) galime paspausti „Add Mark“ mygtuką apačioje ir pridėti šitą pažymi į duomenų bazę.

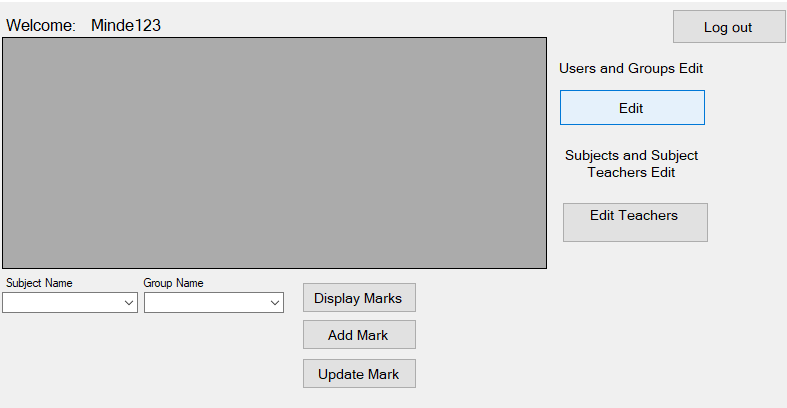


Pav. 4.5. Add Mark.

Paspaudę mygtuką dešinėje pusėje „Update Mark“ mums atsidaro naujas langas, kad atnaujinti esasmą pažymį. Galime įrašyti MarkId ir Mark(pažymį) į kurį norime atnaujinti.Viską įrašę galime paspausti „Update Mark“ mygtuką apačioje ir atnaujinti šitą pažymį duomenų bazėje.



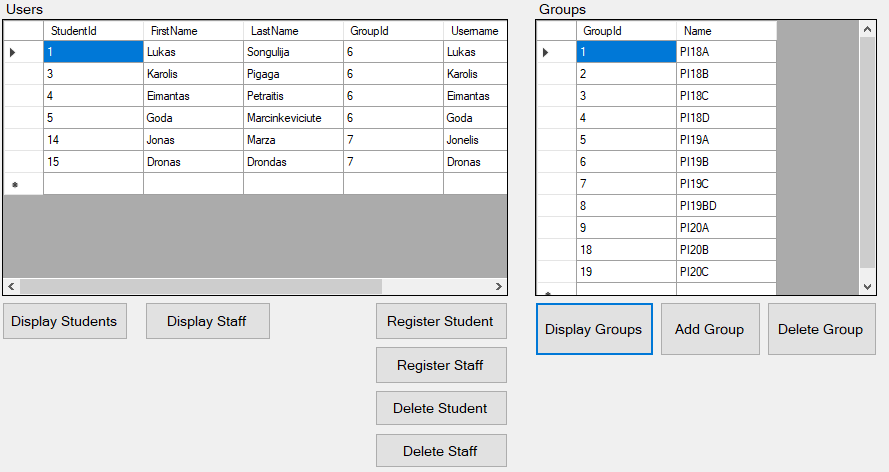
Pav. 4.6. Update Mark.

Jeigu Staff(darbuotojas) yra StaffType „admin“ tada prisijungę prie programos dešinėje pusėje matome papildomus mygtukus kad tvarkyti naudotojus, grupe, darbuotojus ir t.t.

Pav. 4.7. Edit buttons.

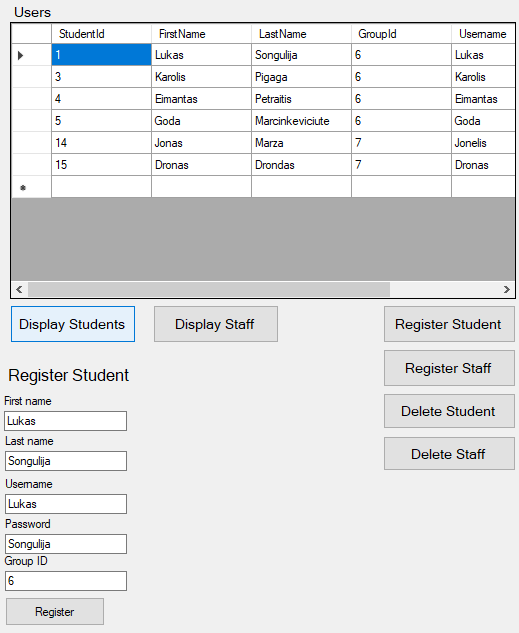
Staff. Staff Type „Admin“

Paspaudę mygtuką „Edit“ mums atsidarys visiškai naujas langas. Šitas langas užsidarys. Paspaudę ant mygtuko „Display Students“ mums lange parodys visus programos studentus. Jeigu paspaudžiame and mygtuko „Display Groups“ mums parodys visas grupes.



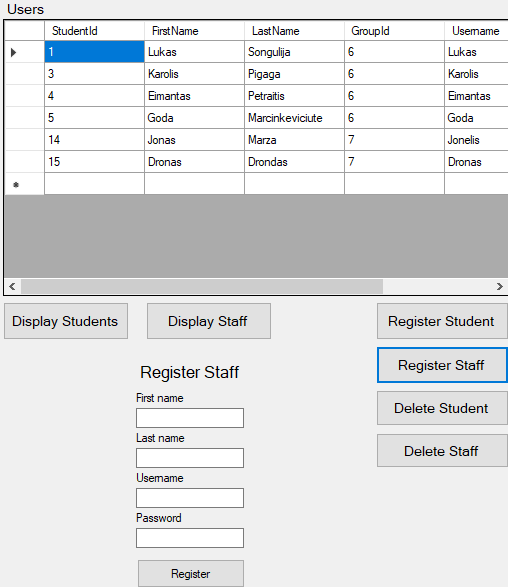
Pav. 4.8. Display Students. Display Groups.

Norėdami įterpti naują studentą galime tiesiog paspausti ant mygtuko „Register Student“. Atsidarys naujas langas ir galėsime įterpti visus naujo naudotojo duomenis. Paspaudę mygtuką „Register“ įterpsime naują naudotoją į duomenų bazę.



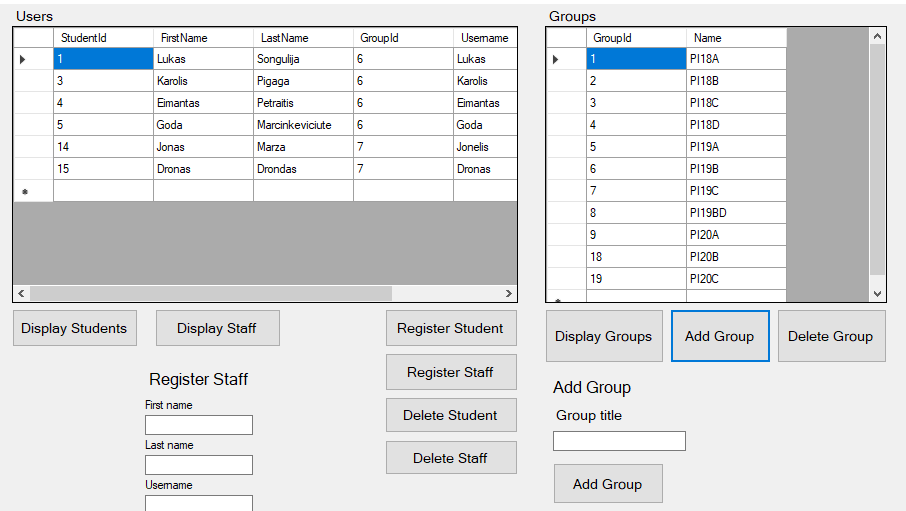
Pav. 4.9. Register Student.

Norėdami įterpti naują Staff(darbuotoją) galime tiesiog paspausti ant mygtuko „Register Staff“. Atsidarys naujas langas ir galėsime įterpti visus naujo darbuotojo duomenis. Paspaudę mygtuką „Register“ įterpsime naują darbuotoją į duomenų bazę.



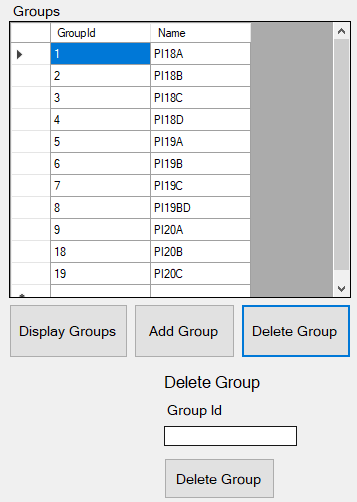
Pav. 4.10. Register Staff.

Norėdami įterpti naują grupę galime tiesiog paspausti ant mygtuko „Add Group“. Atsidarys naujas langas ir galėsime įterpti naujos grupės pavadimą. Paspaudę mygtuką „Add Group“ įterpsime naują grupę į duomenų bazę.



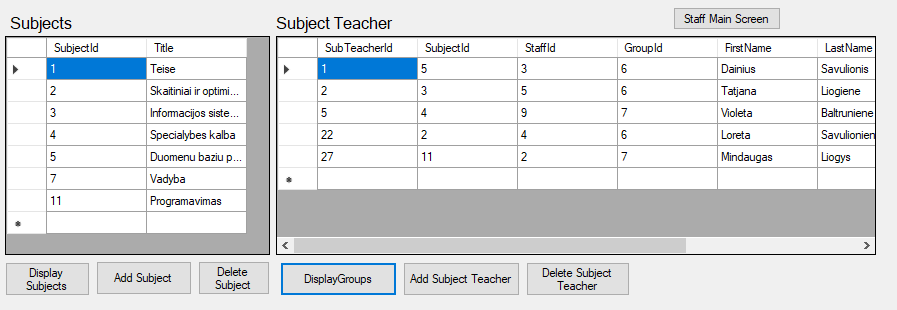
Pav. 4.11. Add Group.

Norėdami ištrinti grupę galime tiesiog paspausti ant mygtuko „Delete Group“. Atsidarys naujas langas ir galėsime įterpti GroupId(grupės Id) kurią norime ištrinti. Paspaudę mygtuką „Delete Group“ ištrinsime grupę iš duomenų bazės.



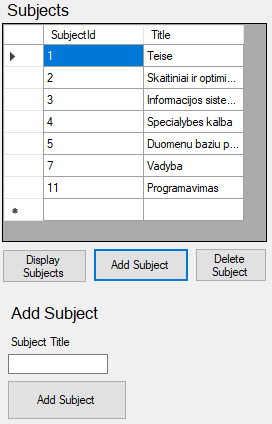
Pav. 4.12. Delete Group.

Paspaudę mygtuką „Display Subjects“ mums kaireme lange parodys visus Subject(dalykų) duomenis. Paspaudę mygtuką „DisplaySubjectTeacher“ mes gausime visus dėstytojus ir jų dėstomus dalykus ir tam dalykui priklausančia grupę.



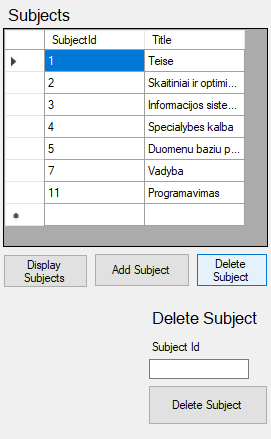
Pav. 4.13. Display Subjects. Display SubjectTeacher.

Norėdami įterpti naują Subject(dalyką) galime tiesiog paspausti ant mygtuko „Add Subject“. Atsidarys naujas langas ir galėsime įterpti naujo Subject(dalyko) pavadimą. Paspaudę mygtuką „Add Subject“ esantį lango apačioje įterpsime naują Subject(dalyką) į duomenų bazę.



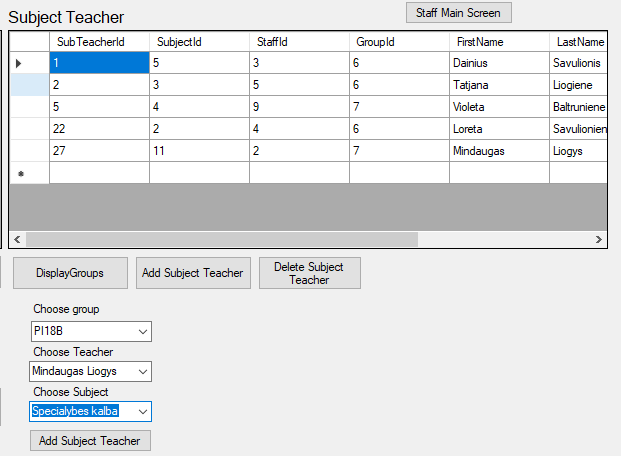
Pav. 4.14. Add Subject.

Norėdami ištrinti Subject(dalyką) galime tiesiog paspausti ant mygtuko „Delete Subject“. Atsidarys naujas langas ir galėsime įterpti SubjectId(dalyko Id) kurį norime ištrinti. Paspaudę mygtuką „Delete Subject“ esantį lango apačioje ištrinsime Subject(dalyką) iš duomenų bazės.



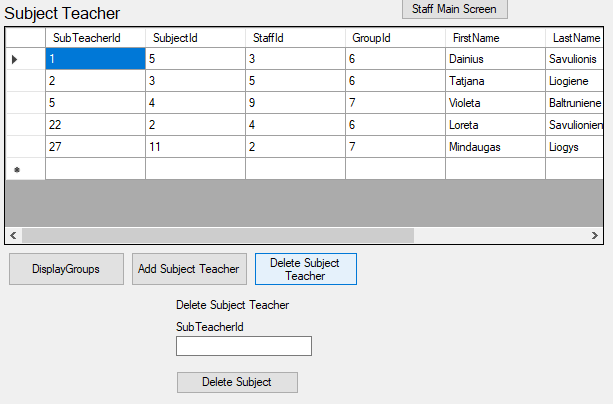
Pav. 4.15. Delete Subject.

Paspaudę mygtuką „Add Subject Teacher“ mums atsidaro naujas langas, kad pridėti naują Subject Teacher(dalyko dėstytoja). Galime pasirinkti iš mūsų turimų Group(Grupių), Staff/Teacher(darbuotojų) ir Subject(dalykų). Pasirenkame darbuotoją, priskiriame jam naują dalyką ir grupę kuriai jis tą dalyką dėstys. Viską pasirinkę galime paspausti „Add Subject Teacher“ mygtuką apačioje ir pridėti pasirinktam darbuotojui naują dalyką ir grupę kuriai jis dėstys šitą dalyką.



Pav. 4.16. Add Subject Teacher.

Norėdami ištrinti SubjectTeacher(dalyko dėstytoja) galime tiesiog paspausti ant mygtuko „Delete Subject Teacher“. Atsidarys naujas langas ir galėsime įterpti SubTeacherId kurį norime ištrinti. Paspaudę mygtuką „Delete Subject“ esantį lango apačioje ištrinsime SubTeacherId dėstytojo dėstomą dalyką ir jam priskirtą grupę iš duomenų bazės.



Pav. 4.17. Delete Subject Teacher.

# IŠVADOS

Tai yra begalo reikalinga sistema tiek studentams tiek darbuotojams. Sukūres sistema pastebėjau, jog ir ji turi pakankamai nemažai trūkumų. Ir beveik visi jie yra susieti su dizainu. Kaip ir kolegijos informacinėje sistemoje dizainas yra pasenęs, galima būtų pasakyti, jog toks dizainas jau yra neaktualus. Pas mane išėjo taip, jog kai kuriose vietose elementai yra išmėtyti po visą ekraną. Tai yra mano supratimo trūkumas kaip reikėtų kurti naudotojams patogu dizainą. Programa atlieka visas jai reikalingas funkcijas.