Relational Database

Record-Keeping System for a Further Education College

Prepared by: Vladislavs Marisevs

For the attention of: Suzanne Kenny

Candidate No: 1479135H

Class: Network and Software Systems

Contents

[Introduction 4](#_Toc355742440)

[System functions 4](#_Toc355742441)

[System requirements 4](#_Toc355742442)

[Entity Relationship Diagram 5](#_Toc355742443)

[Table Teachers 6](#_Toc355742444)

[Table Students 7](#_Toc355742445)

[Table Course 8](#_Toc355742446)

[Table Groups 9](#_Toc355742447)

[Table Subjects 9](#_Toc355742448)

[Table Subject Exam 10](#_Toc355742449)

[Table GroupTeachersSubjects 11](#_Toc355742450)

[Table GroupStudents 11](#_Toc355742451)

[Table Student Results 12](#_Toc355742452)

[Database Tables and Relationships 13](#_Toc355742453)

[Teachers Table 13](#_Toc355742454)

[Course Table 13](#_Toc355742455)

[Group Students 15](#_Toc355742456)

[Group Teachers Subjects Table 16](#_Toc355742457)

[Student Results 17](#_Toc355742458)

[Student Table 18](#_Toc355742459)

[Subject Exam 19](#_Toc355742460)

[Subjects Table 20](#_Toc355742461)

[ER Diagram 21](#_Toc355742462)

[Queries and forms 22](#_Toc355742463)

[Interface design and user’s manual 26](#_Toc355742464)

[Personal details form 29](#_Toc355742465)

[Add edit tutors form 30](#_Toc355742466)

[Add edit student form 31](#_Toc355742467)

[Add edit student to group 32](#_Toc355742468)

[Add edit course 33](#_Toc355742469)

[Add edit group 34](#_Toc355742470)

[Add edit modules 35](#_Toc355742471)

[Add edit exams 36](#_Toc355742472)

[Add edit teachers group modules 37](#_Toc355742473)

[Add edit student results 38](#_Toc355742474)

[Program extras 41](#_Toc355742475)

# Introduction

A Further Education College has increased its student intake steadily over the past ten years. At present it has in excess of 500 students. At the present there is no specific policy with regards to the recording of assessment results. Current system is mainly manual with records being variously kept in hardcopy format or in electronic form, or in mixed form. The location of these records usually knows only particular tutor.

## System functions

* Keep information about
  + Student’s information (name, surname, address, pps number)
  + Teacher’s information (name, surname, address, pps number)
  + College courses information
  + Store student exam results
* Reports about group results
* Do a user friendly interface with the minimum amount of input required

## System requirements

The Record-Keeping System for a Further Education College must have all the information about students in current college. It must keep all student marks for exams, their course and course subjects. The application must have user friendly interface.

For this application I will use MS Access database. And I will keep this database file in \*\GTIExamEntrySystem.0.(version)\db\ExamEntrySystem.mdb

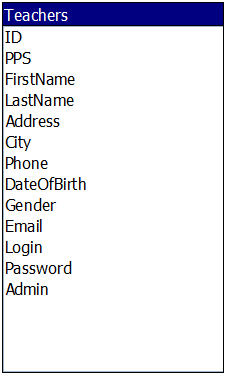
First of all I have to decide how many tables I need:

1. Teachers table – this table should keep information about tutor.
2. Student table – this table should keep information about students.
3. Student Results table – should keep information about student and exam.
4. Exam List table – should keep information about module.
5. Module List table – should keep info about modules.
6. Class group table – should keep information about course.
7. Course List table – should keep information about all college courses.
8. Relational table between Class Group, Module and Teacher.
9. Relational table between Class Group and Student.

Entity Relationship Diagram

This diagram is used by system analyst to identify the entities in the system and describe the relationships between entities. Entities are objects that are important to the system, have at least one attribute, occur more than once and each occurrence must be uniquely identifiable. Because each entity must be uniquely identifiable it is important that every entity has a code or primary key.

### Table Teachers



ID – in order to create stable working database, I was using a primary key in id field. Data type for this field is Auto Number, this means that it is generated automatically. This fixes the bug with unique field errors.

PPS – all citizens of Ireland must have Personal Public Service Number. It should be unique for each person. So it is better to do this field as indexed. Personal Public Service Number contains 7 digits and 2 letters.

FirstName – each tutor have name that stored in teachers table. (as text format)

LastName – each tutor have surname that stored in teachers table (as text format)

Address- each tutor have address that stored in teachers table. (as text format)

City – each tutor have city that stored in teachers table. (as text format)

Phone – each tutor have phone number that stored in teachers table. (as text format)

DateOfBirth – each teacher have date of birth that stored in teachers table (as date time format)

Gender – teachers gender is saved in Boolean format (true = male, false = female)

Email – each tutor have email that stored in teachers table. (as text format)

Login – each tutor have login that is used to enter the system. (as text format)

Password – to enter the system tutor must set a password. I was using text format, but to do a save application it is better to use MD5 format.

Admin – in the system there is two types of users, administrator and user. If user have administrator rights this = true. (Boolean format)

### Table Students

### C:\Users\Evilguy\Desktop\assignment\Doc\ER\008.png

ID – in order to create stable working database, I was using a primary key in id field. Data type for this field is Auto Number, this means that it is generated automatically. This fixes the bug with unique field errors.

PPS – all citizens of Ireland must have Personal Public Service Number. It should be unique for each person. So it is better to do this field as indexed. Personal Public Service Number contains 7 digits and 2 letters.

FirstName – each student have name that stored in student table. (as text format)

LastName – each student have surname that stored in student table (as text format)

Address- each student have address that stored in student table. (as text format)

City – each student have city that stored in student table. (as text format)

Phone – each student have phone number that stored in student table. (as text format)

DateOfBirth – each student have date of birth that stored in student table (as date time format)

Gender – student gender is saved in Boolean format (true = male, false = female)

Email – each student have email that stored in student table. (as text format)

### Table Course

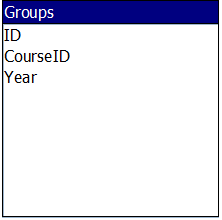
### C:\Users\Evilguy\Desktop\assignment\Doc\ER\001.png

ID – in order to create stable working database, I was using a primary key in id field. Data type for this field is Auto Number, this means that it is generated automatically. This fixes the bug with unique field errors.

CourseName – each course will have a name that will be stored on the course table.

Description – each course will have a short description that will be stored on the course table.

### Table Groups

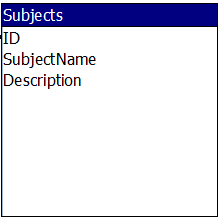


ID – in order to create stable working database, I was using a primary key in id field. Data type for this field is Auto Number, this means that it is generated automatically. This fixes the bug with unique field errors.

CourseID – each class group will have named course, and this field keeps course id.

Year – each class group will have a year that will be stored on the group table.

### Table Subjects

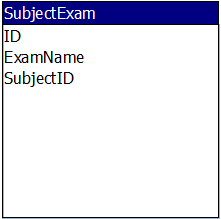


ID – in order to create stable working database, I was using a primary key in id field. Data type for this field is Auto Number, this means that it is generated automatically. This fixes the bug with unique field errors.

SubjectName – each module will have a name that will be stored on the subject table.

Description – each module will have a short description that will be stored on the subject table.

### Table Subject Exam

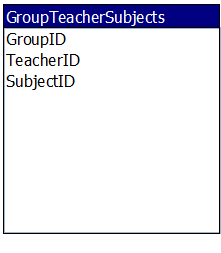


ID – in order to create stable working database, I was using a primary key in id field. Data type for this field is Auto Number, this means that it is generated automatically. This fixes the bug with unique field errors.

ExamName – each module will have an exam that will be stored on the subjectexam table.

SubjectID – this field will keep modules id, because sometimes exams must been relevant to module.

### Table GroupTeachersSubjects



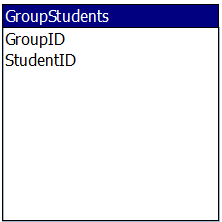
GroupID – this field is used to join Group Table.

TeacherID – this field is used to join Teachers Table.

SubjectID – this field is used to join Subject Table.

All fields in this table are made as combined primary key.

### Table GroupStudents

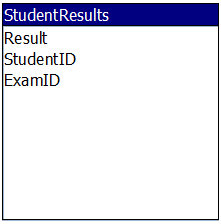


GroupID – this field is used to join GroupID.

StudentID – this field is used to join StudentID.

All fields in this table are made as combined primary key.

### Table Student Results



Result – this field store the exam result for selected student and exam.

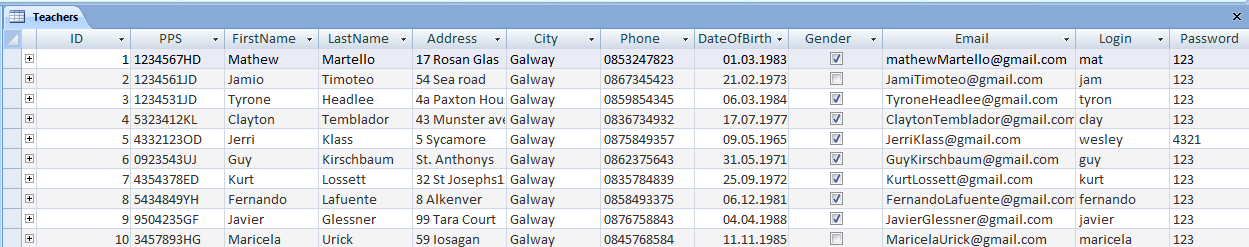
StudentID – this field is used to join StudentID.

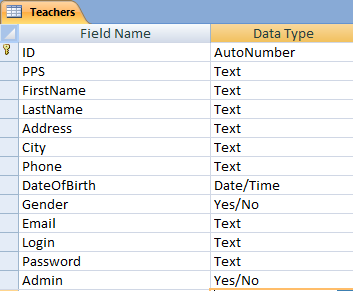
ExamID - this field is used to join ExamID.

## Database Tables and Relationships

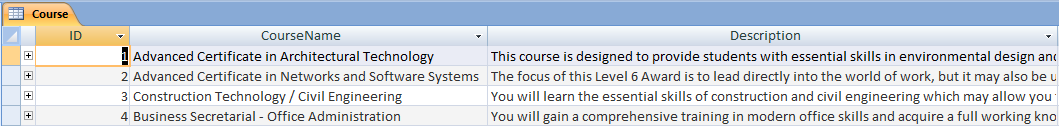
In this section I will provide an overview of database that I developed in Microsoft Access.

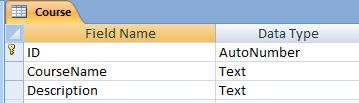
### Teachers Table



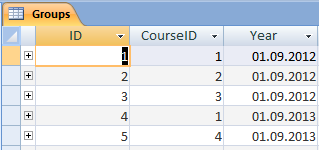


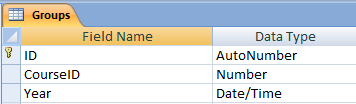
### Course Table





Group Table





### Group Students

### C:\Users\Evilguy\Desktop\assignment\tables\0031.png

### C:\Users\Evilguy\Desktop\assignment\tables\0032.png

### Group Teachers Subjects Table

### C:\Users\Evilguy\Desktop\assignment\tables\0041.png

### C:\Users\Evilguy\Desktop\assignment\tables\0042.png

### Student Results

### C:\Users\Evilguy\Desktop\assignment\tables\0051.png

### C:\Users\Evilguy\Desktop\assignment\tables\0052.png

### Student Table

### C:\Users\Evilguy\Desktop\assignment\tables\0061.png

### C:\Users\Evilguy\Desktop\assignment\tables\0062.png

### Subject Exam

### C:\Users\Evilguy\Desktop\assignment\tables\0071.png

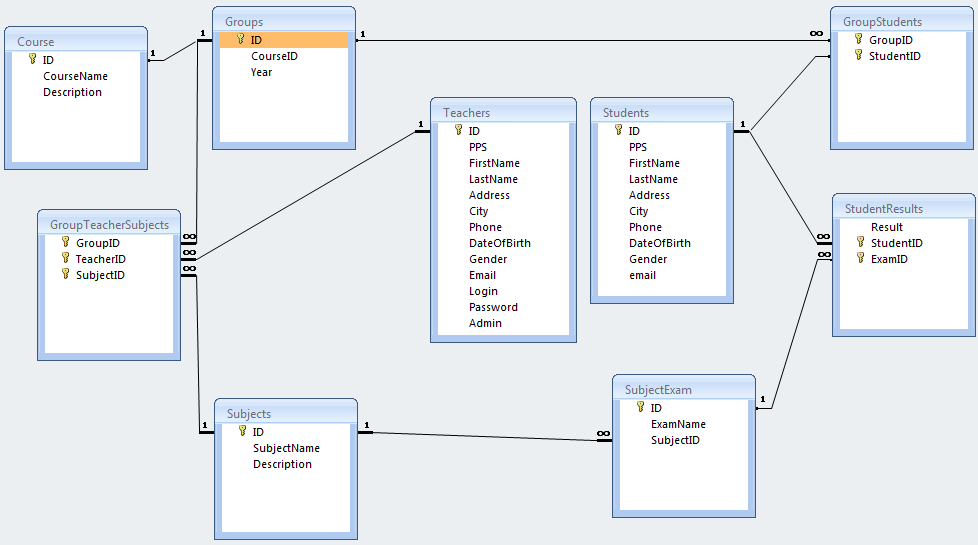
### C:\Users\Evilguy\Desktop\assignment\tables\0072.png

### Subjects Table

### C:\Users\Evilguy\Desktop\assignment\tables\0081.png

### C:\Users\Evilguy\Desktop\assignment\tables\0082.png

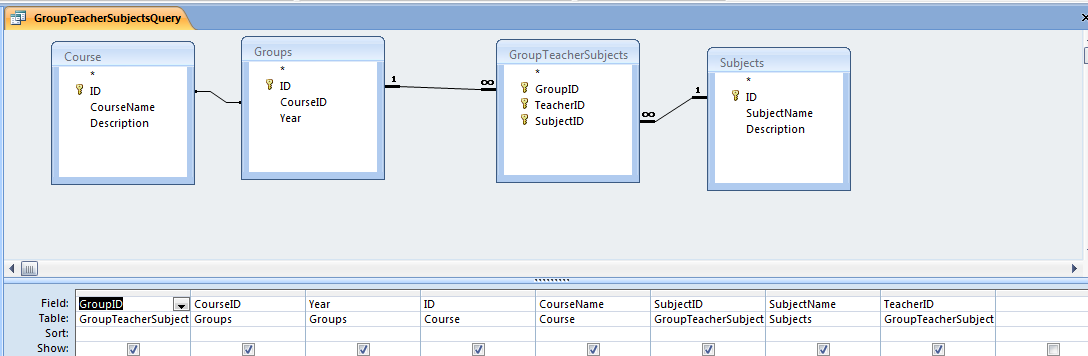
### ER Diagram



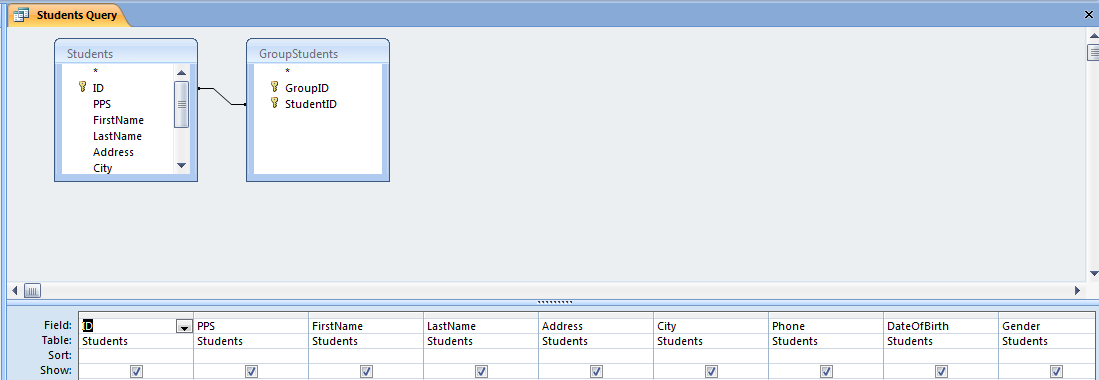
## Queries and forms

Queries and forms that I created in Ms Access Database, I was using to test my developed database workability. And to compare the results with java programmed application.

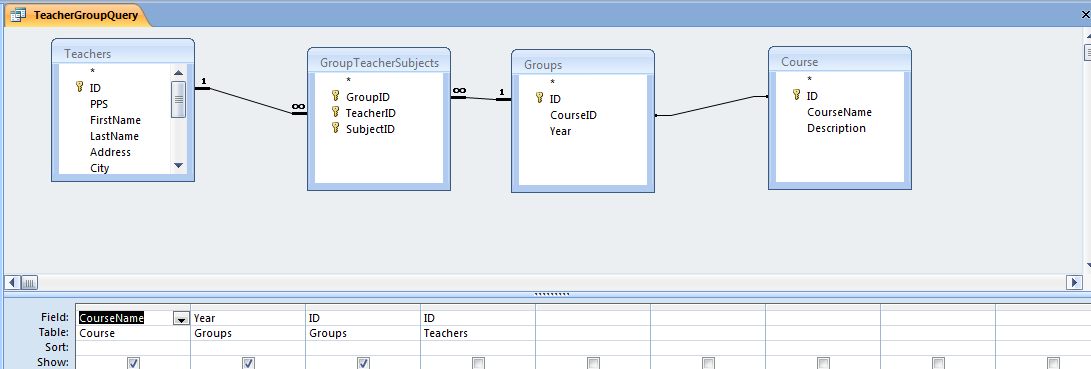
Group Teachers Subject Query that joins 3 tables together



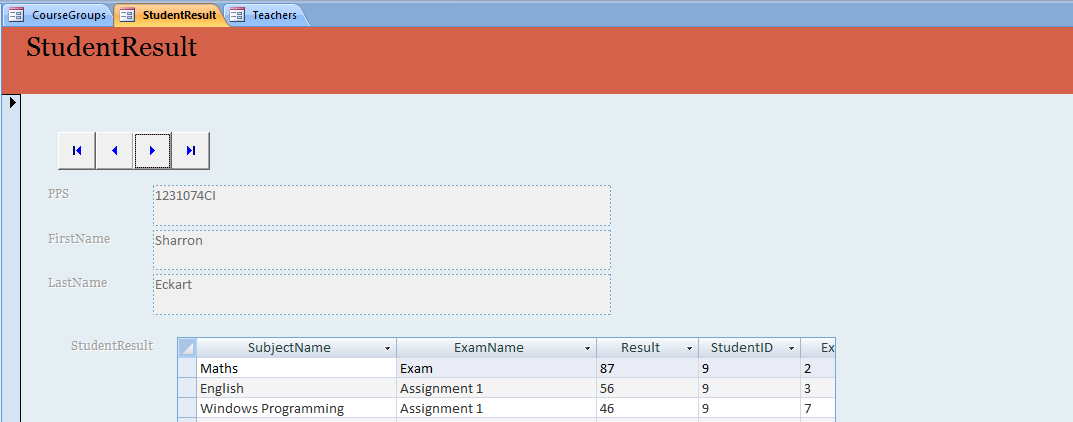
Group Students Query joins 2 tables together.



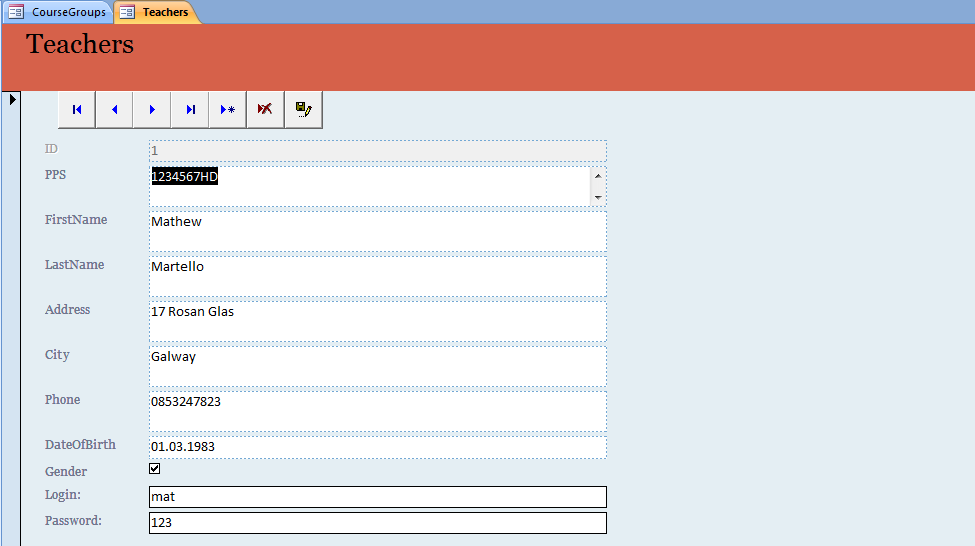
Teachers Group Table returns information about tutors groups



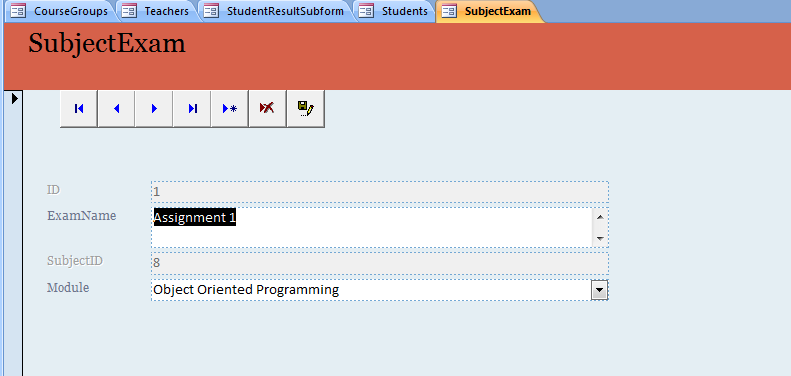
Student Result Form allows us to fill in the information about student results



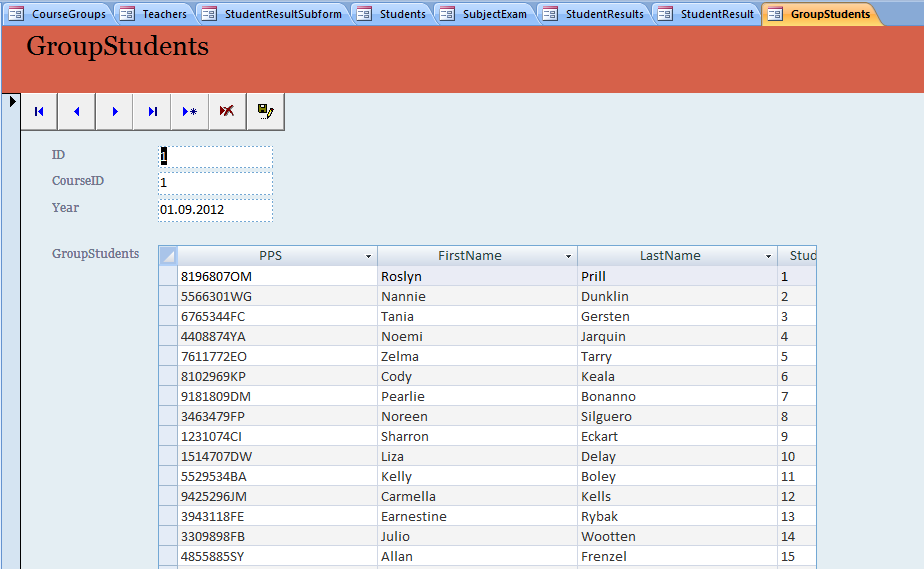
Teachers form allows us to edit tutor information



Module Exam form allows us to change exam information



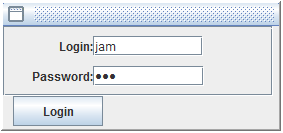
Student Group Form allows us to edit students in groups



# Interface design and user’s manual

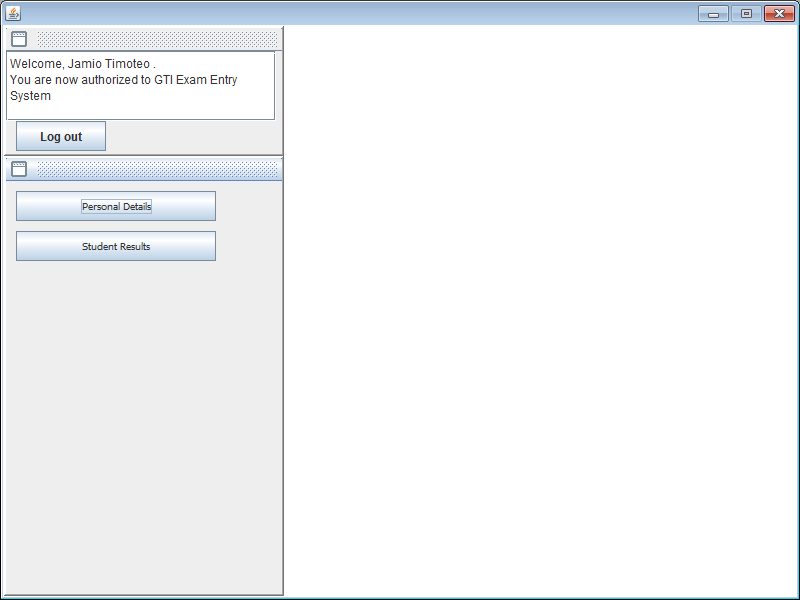
My project is dedicated to Galway Technical Institute record keeping system. I tried to do a user friendly design for this application. The resolution of this application is 800x600 px, so tutors can use this program on old computers as well.

To get into the system tutor must enter the valid password and login.

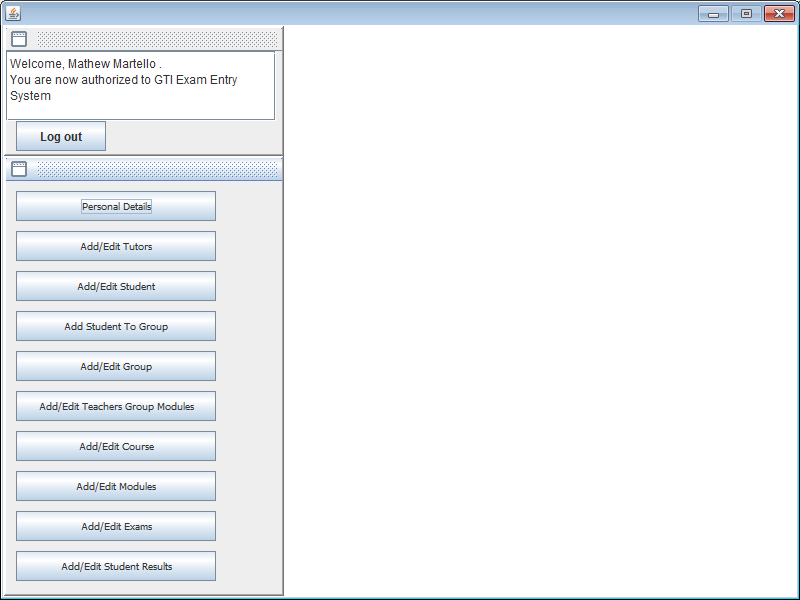


There are two types of users: Administrator and User. To login into the system you can use administrator **login: mat password: 123** and user **login: jam password: 123**

First type is tutor, he could do just simple things add or edit assignment results and edit his own information.

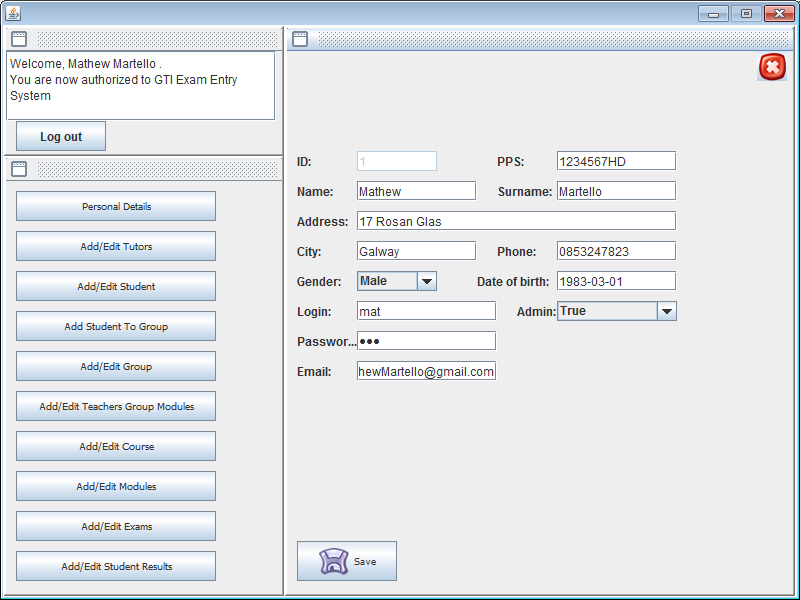


Second is administrator, he can do more than tutor. This user can change his information or add / edit other tutors information. Add or edit student information and set his class group or add one more class group for particular student. Add or edit college courses and add or edit class group. For example “Advanced Certificate in Networks and Software Systems” is a course, but class group needs year of studding. Add or edit modules. And make a relationship between tutor his group and his module (or subject) that he does for particular class group. The administrator can add/remove/ edit exams or assignments for modules. These users also can add/edit/view exam results for any class group and student.



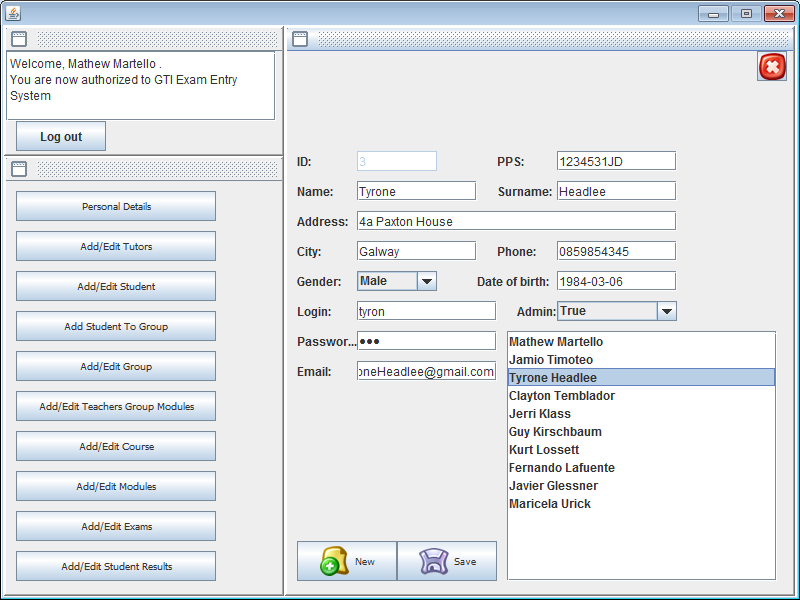
## Personal details form

This form shows the information about user that is logged in. He can change all information except database id, because it is generated automatically. To update information user should click save button and information will be saved in database.



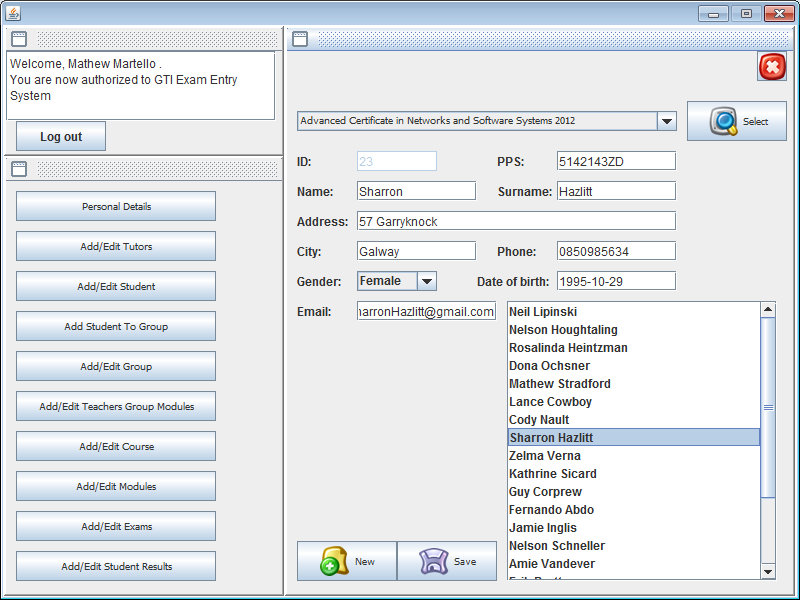
## Add edit tutors form

This form shows the information about current tutor, and user list that can be edited and saved into database. Also, we can add new tutor, to save new information, all text boxes should be filled in.



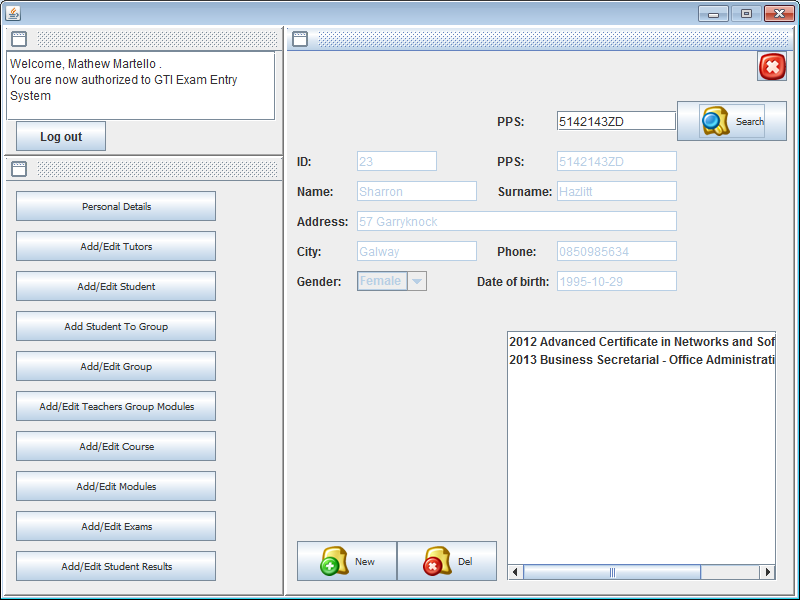
## Add edit student form

To get student information about existing student, administrator should select class group in combo box and click select button. Then student list will be filled in. And choosing the particular student the information about him will be updated on the screen. If administrator wants to add new student, he must select a class group first and then he can add new student to database.



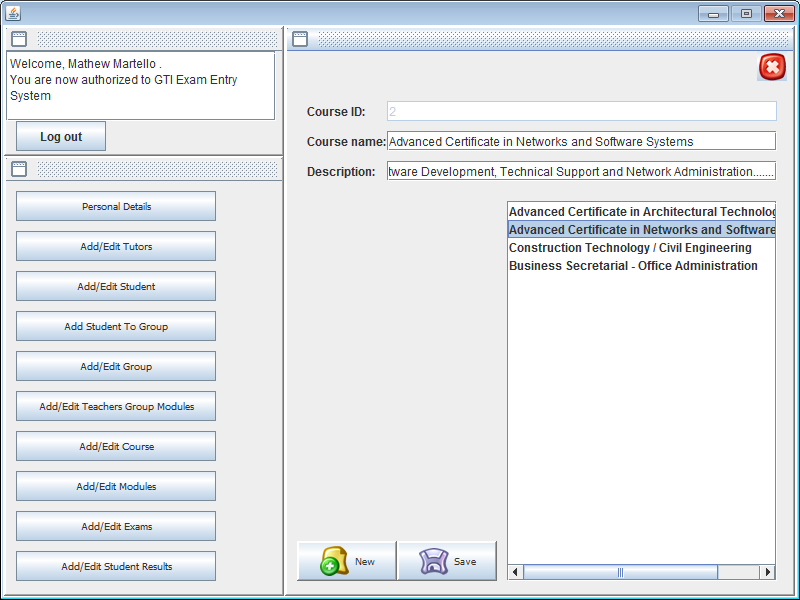
## Add edit student to group

Using this form administrator can add or remove student class group. But first he must choose a student, using his Personal Public Service Number. On this picture we can see that this student did 2 classes in this college.



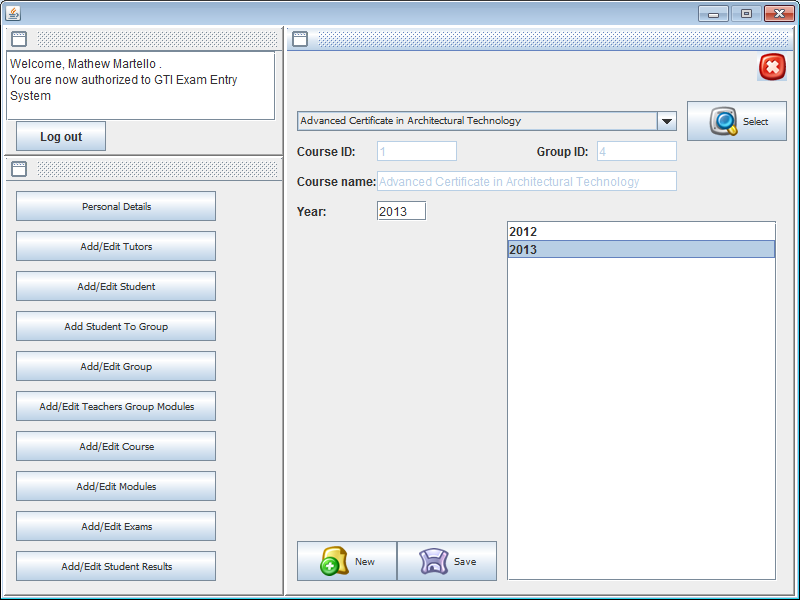
## Add edit course

Using this form administrator can add or edit college courses.



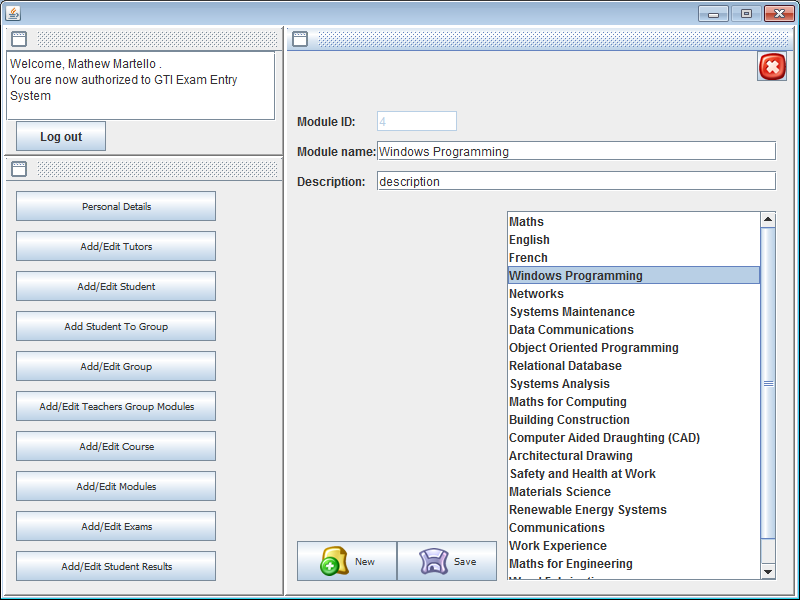
## Add edit group

Using this form administrator can add or edit class group. Class group have two variables, course identification and year of course. To view all class groups for particular course, user should choose it in combo box and press select button.



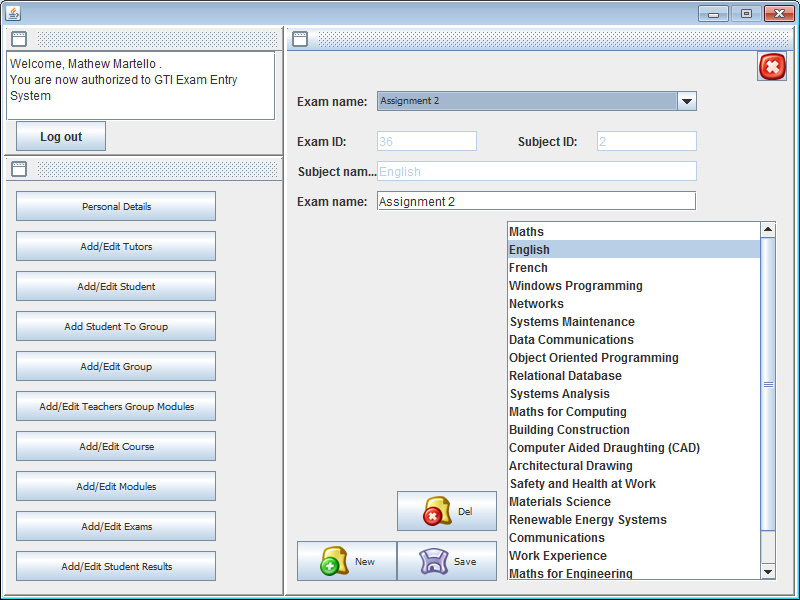
## Add edit modules

Using this form administrator can add or edit new module.



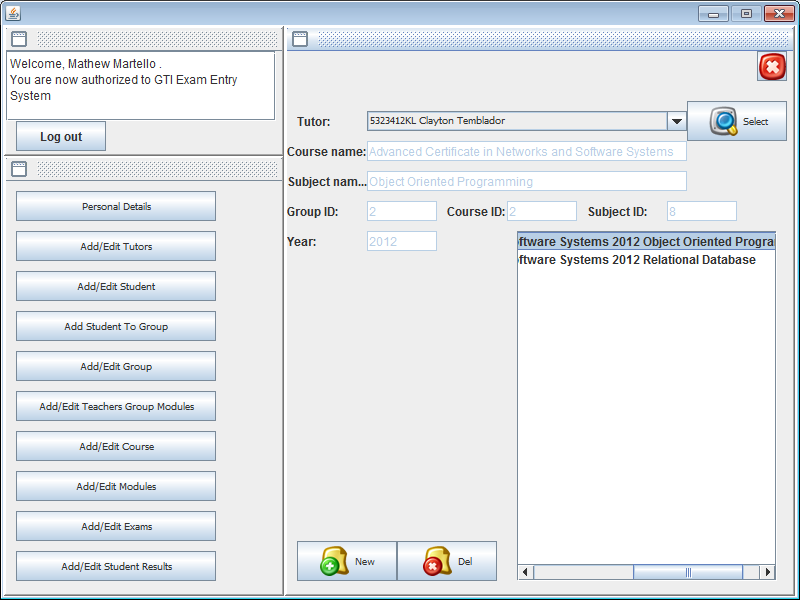
## Add edit exams

Using this form administrator can add, remove or edit exams for particular module. In the list user can see all the modules in college, he should choose one and combo box will be updated with this module exams or assignments.



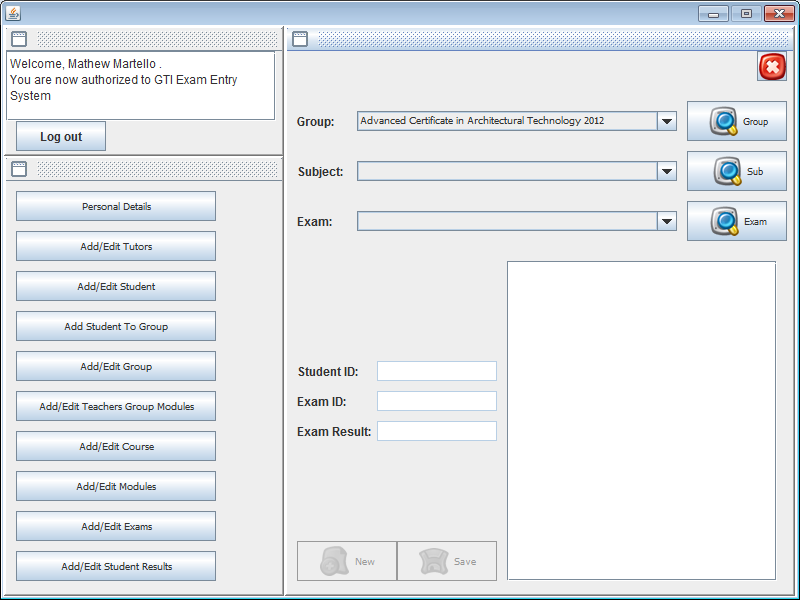
## Add edit teachers group modules

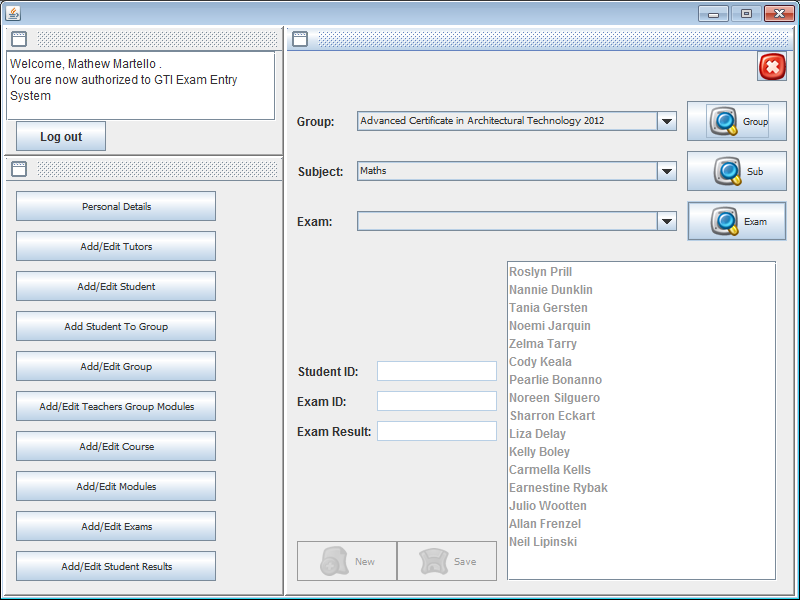
Using this form administrator can add or delete relationship between tutor class group and module. This means that tutor does a module for class group. So we can add to a tutor more groups or remove them. This gives an option to non administrator users add or edit student exam results for selected module.

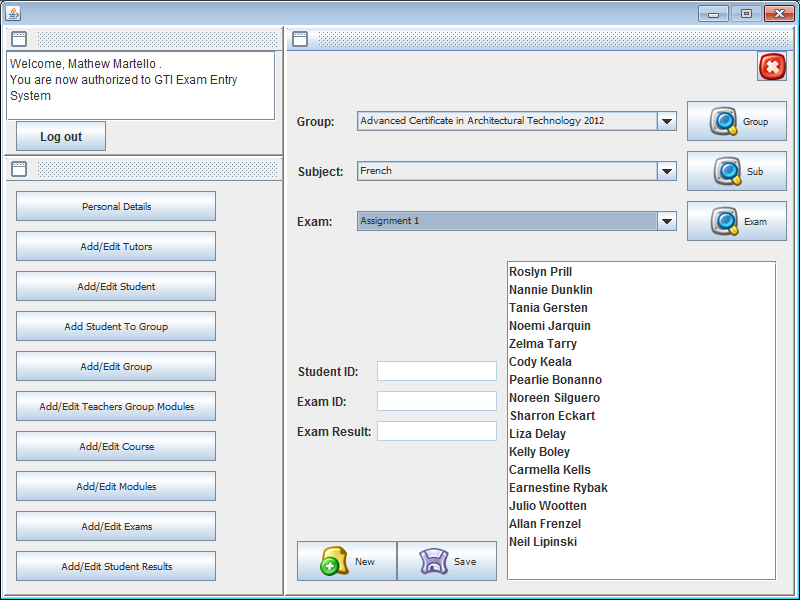


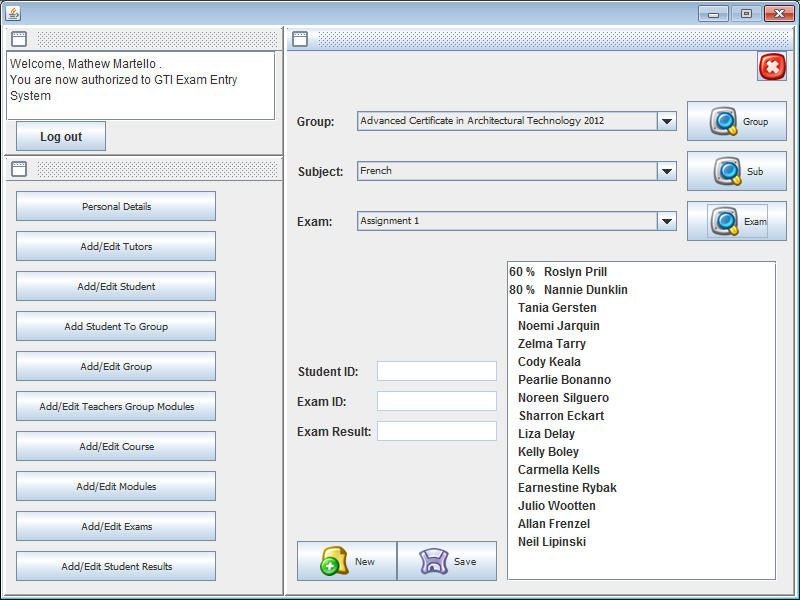
## Add edit student results

Using this form administrator or user can add or edit student results. First of all user should select a class group and click *Group* button, it confirms that the class group was selected. Then module should be selected and to confirm the selection user should click *Sub* button. And to view student results user should select an exam or assignment in combo box and click Exam button to confirm it. Now we can see list of students, if there is a result in percentage before student name and surname. That mean this student was rated. If there is no result before name, then there is no record with this student mark. This relationship makes result table keep only useful information about student.



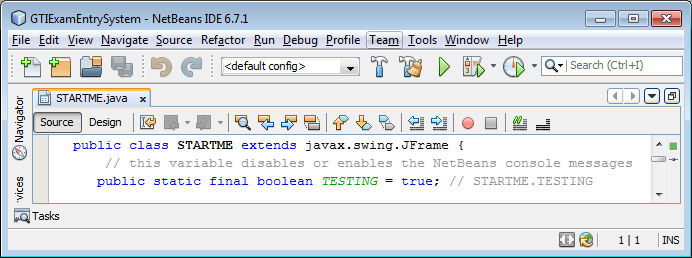






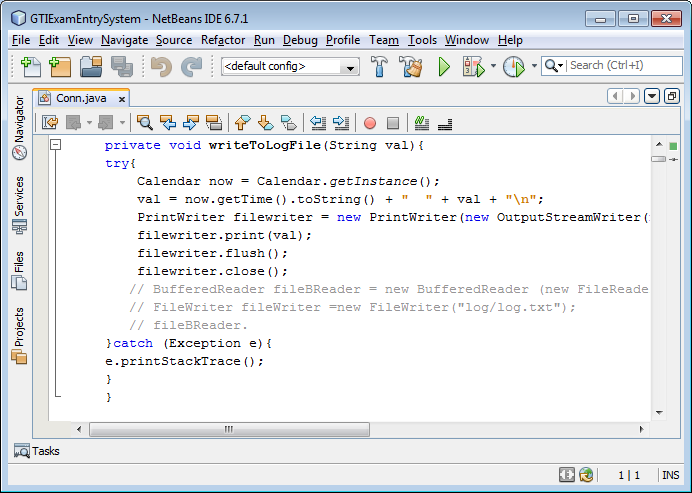
## Program extras

This is quite massive application and to debug an error sometimes took a lot of time. To reduce this time I set up a Boolean constant that enables (true) or disables (false) program messages that show the result set that it gets from database. This constant is in main form (STARTME.java) its name is TESTING.



To debug other errors I did a log file method. This method keeps a track of all login users, or login fails with current date and time. It saves information about wrong connections or connection errors.

Method:



The log file is in \*\GTIExamEntrySystem.0.(version)\log\log.txt

The database is in \*\GTIExamEntrySystem.0.(version)\db\ ExamEntrySystem.mdb