

GooGrade

Generated by Doxygen 1.5.5

Mon Mar 16 11:30:56 2009

Contents

1	Main Page	1
1.1	Description	1
1.2	Locations	1
1.3	MVC	1
1.4	Note	1
1.5	Markup	2
2	Namespace Documentation	3
2.1	Package blugoo	3
3	Class Documentation	5
3.1	Account Class Reference	5
3.2	AccountController Class Reference	10
3.3	Announcement Class Reference	11
3.4	AnnouncementController Class Reference	14
3.5	Assignment Class Reference	15
3.6	AssignmentController Class Reference	21
3.7	Course Class Reference	22
3.8	CourseController Class Reference	28
3.9	Email Class Reference	30
3.10	EmailAddress Class Reference	32
3.11	EmailController Class Reference	34
3.12	File Class Reference	35
3.13	FileController Class Reference	38
3.14	GradingRules Class Reference	40
3.15	GradingRulesController Class Reference	43
3.16	Login Class Reference	45
3.17	Password Class Reference	47
3.18	PasswordController Class Reference	49

3.19	Permissions Class Reference	50
3.20	PermissionsController Class Reference	58
3.21	Serializable Class Reference	60
3.22	StorageConnection Class Reference	61
3.23	Student Class Reference	62
3.24	StudentController Class Reference	64
3.25	Teacher Class Reference	65
3.26	TeacherAssistant Class Reference	67
3.27	TeacherAssistantController Class Reference	69
3.28	TeacherController Class Reference	70
4	File Documentation	75
4.1	blugoo/WEB-INF/src/java/blugoo/Account.java File Reference	75
4.2	blugoo/WEB-INF/src/java/blugoo/AccountController.java File Reference	76
4.3	blugoo/WEB-INF/src/java/blugoo/Announcement.java File Reference	77
4.4	blugoo/WEB-INF/src/java/blugoo/AnnouncementController.java File Reference	78
4.5	blugoo/WEB-INF/src/java/blugoo/Assignment.java File Reference	79
4.6	blugoo/WEB-INF/src/java/blugoo/AssignmentController.java File Reference	80
4.7	blugoo/WEB-INF/src/java/blugoo/Course.java File Reference	81
4.8	blugoo/WEB-INF/src/java/blugoo/CourseController.java File Reference	82
4.9	blugoo/WEB-INF/src/java/blugoo/Email.java File Reference	83
4.10	blugoo/WEB-INF/src/java/blugoo/EmailAddress.java File Reference	84
4.11	blugoo/WEB-INF/src/java/blugoo/EmailController.java File Reference	85
4.12	blugoo/WEB-INF/src/java/blugoo/File.java File Reference	86
4.13	blugoo/WEB-INF/src/java/blugoo/FileController.java File Reference	87
4.14	blugoo/WEB-INF/src/java/blugoo/GradingRules.java File Reference	88
4.15	blugoo/WEB-INF/src/java/blugoo/GradingRulesController.java File Reference	89
4.16	blugoo/WEB-INF/src/java/blugoo/Login.java File Reference	90
4.17	blugoo/WEB-INF/src/java/blugoo/Password.java File Reference	91
4.18	blugoo/WEB-INF/src/java/blugoo/PasswordController.java File Reference	92
4.19	blugoo/WEB-INF/src/java/blugoo/Permissions.java File Reference	93
4.20	blugoo/WEB-INF/src/java/blugoo/PermissionsController.java File Reference	94
4.21	blugoo/WEB-INF/src/java/blugoo/StorageConnection.java File Reference	95
4.22	blugoo/WEB-INF/src/java/blugoo/Student.java File Reference	96
4.23	blugoo/WEB-INF/src/java/blugoo/StudentController.java File Reference	97
4.24	blugoo/WEB-INF/src/java/blugoo/Teacher.java File Reference	98
4.25	blugoo/WEB-INF/src/java/blugoo/TeacherAssistant.java File Reference	99

4.26 blugoo/WEB-INF/src/java/blugoo/TeacherAssistantController.java File Reference	100
4.27 blugoo/WEB-INF/src/java/blugoo/TeacherController.java File Reference	101
4.28 front.txt File Reference	102

Chapter 1

Main Page

1.1 Description

This is a project for CSC308, Winter 2009 with Clark Savage Turner. This is made by the bluGoo Team whose wiki can be found at <http://wiki.csc.calpoly.edu/blugoo/>

1.2 Locations

Interested in a printable version of this document? If you have the source code run 'make latex' otherwise check out the most recent published versions.

- PDF: <http://users.csc.calpoly.edu/~nwelch/blugoo/blugooarch.pdf>
- Web: <http://users.csc.calpoly.edu/~nwelch/blugoo/arch/>
- JavaDocs: <http://users.csc.calpoly.edu/~nwelch/blugoo/javadocs/>

1.3 MVC

This project follows the Model, View, Controller Philosophy. Views are Java Server Pages. Models are classes containing Get and Set methods and the ability to store data to the storage system. Controllers act with the models and the views to do actual specific work.

1.4 Note

This is not complete enough to be implemented. Most of the controllers need more functionality and how to connect to the storage system is all but undocumented. Also the JSP pages were excluded because they had no documentation what so ever.

1.5 Markup

Doxygen generates [HTML 4.01 Transitional](#) documents with [Cascading Style Sheets \(CSS\)](#) and you can include markup tags that are compatible with those standards.

Chapter 2

Namespace Documentation

2.1 Package blugoo

Classes

- class [Account](#)
This is a class that keeps all of the information of a user.
- class [AccountController](#)
This class does most of the work on the [Account](#).
- class [Announcement](#)
An [Announcement](#) for a [Course](#).
- class [AnnouncementController](#)
This is the controller for [Announcement](#).
- class [Assignment](#)
This class keeps information about Assignments.
- class [AssignmentController](#)
Does all of the work on the [Assignment](#), it is the controller.
- class [Course](#)
This class keeps information about Courses.
- class [CourseController](#)
Methods for modifying a course and it's related actions.
- class [Email](#)
The class stores information about Emails.
- class [EmailAddress](#)
A model for email addresses since they need to be validated and such.

- class [EmailController](#)
This deals with sending emails.
- class [File](#)
This class keeps information about Files.
- class [FileController](#)
This class is to deal with files.
- class [GradingRules](#)
The [GradingRules](#) class determines how the Total Grade is computed from mutiple [Assignment](#) Grades.
- class [GradingRulesController](#)
The [GradingRulesController](#) class is the [GradingRules](#) controller.
- class [Login](#)
This should deal with login I guess.
- class [Password](#)
A model for passwords since they need to be validated and such.
- class [PasswordController](#)
A controller for passwords since they need to be validated and such.
- class [Permissions](#)
This class keeps information about permissions for each action.
- class [PermissionsController](#)
This class keeps the permissions controller.
- class [StorageConnection](#)
This is the class that deals with connecting to the storage device.
- class [Student](#)
This class keeps information about Students.
- class [StudentController](#)
This class deals with controlling Students.
- class [Teacher](#)
A [Teacher](#) is an [Account](#) type that has the most permissions.
- class [TeacherAssistant](#)
This class keeps information about [Teacher](#) Assistants.
- class [TeacherAssistantController](#)
This class controlls [Teacher](#) Assistants.
- class [TeacherController](#)
This is a controller for teacher.

```
classDiagram
    class Account {
        <<abstract>>
    }
    class Serializable
    class boolean
    class String
    class Permissions
    class EmailAddress
    class Password
    Account <|-- Serializable
    Account <|-- boolean
    Account <|-- String
    Account <|-- Permissions
    Account <|-- EmailAddress
    Account <|-- Password
    Account --> Permissions : permission
    Account --> String : username, fullName
    Account --> String : address
    Account --> String : password
    Account --> EmailAddress : emailAddress
    Account --> Password : password
```

The diagram illustrates the Account class and its relationships with other classes. The Account class is the base class for Serializable, boolean, String, Permissions, EmailAddress, and Password. The Account class has attributes: username, fullName, address, password, emailAddress, and permission. The Account class has methods: manageMetric, manageUser, manageGrade, manageCourse, uploadFile, sendEmail, manageRoster, viewOwnGrade, and manageAnnouncement. The Account class has associations with Permissions (permission), String (username, fullName, address, password), EmailAddress (emailAddress), and Password (password).

3.1.1 Detailed Description

This is a class that keeps all of the information of a user.

A user is a synonym for an [Account](#). This will have lots of get and set methods that will handle getting and saving Accounts to the Storage system using the Storage Connection class.

Author:

bluGoo

Version:

0.42

Public Member Functions

- [EmailAddress](#) getEmailAddress ()
gets the [EmailAddress](#) of this account.
- boolean setEmailAddress ()
sets the [EmailAddress](#) of this [Account](#).

Package Functions

- String getUsername ()
get the accounts user name.
- String getFullName ()
get the accounts full name.
- boolean setPassword (String newpass)
Takes new password, hashes it, and saves to database.
- String setUsername ()
set the Account's user name.
- String setFullName ()
set the Account's full name.
- boolean isPassword (String passwd)
Checks the provided password to see if they match.

Private Attributes

- String [userName](#)
Username for logging in.
- String [fullName](#)

Full name of the [Account](#) owner.

- [EmailAddress emailAddress](#)
Email address of the [Account](#) owner.
- [Password password](#)
Password for logging in.
- [Permissions permission](#)
Permissions for performing actions.

3.1.2 Member Function Documentation

3.1.2.1 String getUsername () [package]

get the accounts user name.

Returns:

a string with the user name of the current account

3.1.2.2 String getFullName () [package]

get the accounts full name.

Returns:

a string with the full name of the current account

3.1.2.3 EmailAddress getEmailAddress ()

gets the [EmailAddress](#) of this account.

Returns:

this user's email address

3.1.2.4 boolean setPassword (String *newpass*) [package]

Takes new password, hashes it, and saves to database.

Parameters:

newpass the new password to be saved

Returns:

true if set, false if failure.

3.1.2.5 String setUsername () [package]

set the Account's user name.

Returns:

true if set, false if error.

3.1.2.6 String setFullName () [package]

set the Account's full name.

Returns:

true if set, false if error.

3.1.2.7 boolean setEmailAddress ()

sets the [EmailAddress](#) of this [Account](#).

Returns:

true if set, false if error.

3.1.2.8 boolean isPassword (String passwd) [package]

Checks the provided password to see if they match.

Parameters:

passwd the password entered by the user that will be compared with the stored value.

Returns:

true if the hashed passwd matches the password in the database, false otherwise.

3.1.3 Member Data Documentation**3.1.3.1 String userName** [private]

Username for logging in.

3.1.3.2 String fullName [private]

Full name of the [Account](#) owner.

3.1.3.3 EmailAddress emailAddress [private]

[Email](#) address of the [Account](#) owner.

3.1.3.4 Password password [private]

[Password](#) for logging in.

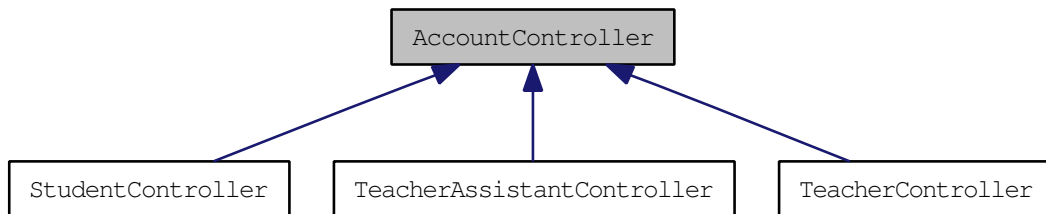
This will not be stored in plain text due to privacy concerns.

3.1.3.5 Permissions permission [private]

[Permissions](#) for performing actions.

3.2 AccountController Class Reference

Inheritance diagram for AccountController:



3.2.1 Detailed Description

This class does most of the work on the [Account](#).

It is the [Account](#) controller. Not really sure what it should be doing, but we'll get to that.

Author:

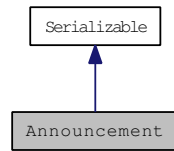
bluGoo

Version:

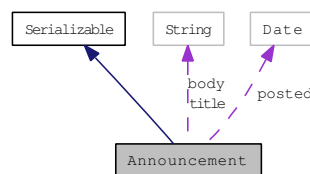
0.42

3.3 Announcement Class Reference

Inheritance diagram for Announcement:



Collaboration diagram for Announcement:



3.3.1 Detailed Description

An [Announcement](#) for a [Course](#).

Quite a simple class for saving and storing Announcements. There should be a method in [Course](#) to obtain an array of these associated with said [Course](#).

Author:

bluGoo

Version:

0.42

Public Member Functions

- String [getBody](#) ()
Returns the body, aka the content, of an [Announcement](#).
- Date [getDate](#) ()
Returns when this [Announcement](#) was posted.
- String [getTitle](#) ()
Returns the title of this [Announcement](#).
- boolean [setBody](#) ()
Sets the body of the [Announcement](#).
- Date [setDate](#) ()
Sets the date of the [Announcement](#).

- String `setTitle ()`
sets the title of the announcement.

Private Attributes

- String `body`
- String `title`
- Date `posted`

3.3.2 Member Function Documentation

3.3.2.1 String `getBody ()`

Returns the body, aka the content, of an [Announcement](#).

Returns:

the content of the post in a single string.

3.3.2.2 Date `getDate ()`

Returns when this [Announcement](#) was posted.

Returns:

the date this was posted.

3.3.2.3 String `getTitle ()`

Returns the title of this [Announcement](#).

Returns:

a string containing the title.

3.3.2.4 boolean `setBody ()`

Sets the body of the [Announcement](#).

Returns:

false if error. True if successful

3.3.2.5 Date setDate ()

Sets the date of the [Announcement](#).

Returns:

false if error. True if successful

3.3.2.6 String setTitle ()

sets the title of the announcement.

Returns:

false if error. True if successful

3.3.3 Member Data Documentation

3.3.3.1 String body [private]

3.3.3.2 String title [private]

3.3.3.3 Date posted [private]

3.4 AnnouncementController Class Reference

3.4.1 Detailed Description

This is the controller for [Announcement](#).

All it does is post the announcement.

Author:

bluGoo

Version:

0.42

Package Functions

- boolean [post](#) (String title, String post)

Posts an announcement.

3.4.2 Member Function Documentation

3.4.2.1 boolean post (String *title*, String *post*) [package]

Posts an announcement.

This will call the functions needed to save to the database.

Parameters:

title The title

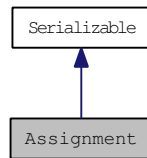
post The post body

Returns:

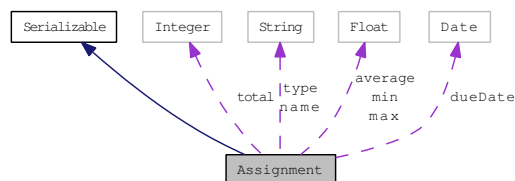
false if fail, true if pass.

3.5 Assignment Class Reference

Inheritance diagram for Assignment:



Collaboration diagram for Assignment:



3.5.1 Detailed Description

This class keeps information about Assignments.

Author:

bluGoo

Version:

0.42

Public Member Functions

- Date [getDueDate \(\)](#)
Gets a Date object for the due date of the assignment.
- String [getName \(\)](#)
Gets a String for the name of the assignment.
- Integer [getTotal \(\)](#)
Gets the total points for the [Assignment](#).
- String [getType \(\)](#)
Gets the type of [Assignment](#).
- Float [getAvg \(\)](#)
Gets the average points of [Assignment](#).
- Float [getMax \(\)](#)

Gets the max point out of all graded assignment of this [Assignment](#).

- Float [getMin](#) ()
Gets the min point out of all graded assignment of this [Assignment](#).
- boolean [setDueDate](#) (Date [dueDate](#))
Sets the dueDate of the [Assignment](#).
- boolean [setName](#) (String [name](#))
Sets the namne of the [Assignment](#).
- boolean [setTotal](#) (Integer [total](#))
Sets the total points of the [Assignment](#).
- boolean [setType](#) (String [type](#))
Sets the type of the [Assignment](#).
- boolean [setAvg](#) (Float [average](#))
Sets the average points of the [Assignment](#).
- boolean [setMax](#) (Float [max](#))
Sets the max points of the [Assignment](#).
- boolean [setMin](#) (Float [min](#))
Sets the min points of the [Assignment](#).

Private Attributes

- Date [dueDate](#)
The due date of the [Assignment](#).
- String [name](#)
The name of the [Assignment](#).
- Integer [total](#)
The maximum possible score of the [Assignment](#).
- String [type](#)
The type of the [Assignment](#); can be Test, Quiz, Homework, Participation, or Other.
- Float [average](#)
The average for all graded assignments for this assignment.
- Float [max](#)
The maximum for all graded assignments for this assignment.
- Float [min](#)
The minimum for all graded assignments for this assignment.

3.5.2 Member Function Documentation

3.5.2.1 Date getDueDate ()

Gets a Date object for the due date of the assignment.

Returns:

the date the assignment is due

3.5.2.2 String getName ()

Gets a String for the name of the assignment.

Returns:

the name of the assignment.

3.5.2.3 Integer getTotal ()

Gets the total points for the [Assignment](#).

Returns:

this Assignment's total points.

3.5.2.4 String getType ()

Gets the type of [Assignment](#).

Returns:

a String of the type [Assignment](#).

3.5.2.5 Float getAvg ()

Gets the average points of [Assignment](#).

Returns:

this Assignment's average points.

3.5.2.6 Float getMax ()

Gets the max point out of all graded assignment of this [Assignment](#).

Returns:

this Assignment's max points.

3.5.2.7 Float getMin ()

Gets the min point out of all graded assignment of this [Assignment](#).

Returns:

this Assignment's min points.

3.5.2.8 boolean setDueDate (Date *dueDate*)

Sets the dueDate of the [Assignment](#).

Parameters:

dueDate the date the [Assignment](#) is due.

Returns:

true if set, false if failure.

3.5.2.9 boolean setName (String *name*)

Sets the name of the [Assignment](#).

Parameters:

name the name of the [Assignment](#).

Returns:

true if set, false if failure.

3.5.2.10 boolean setTotal (Integer *total*)

Sets the total points of the [Assignment](#).

Parameters:

total the total points of the [Assignment](#).

Returns:

true if set, false if failure.

3.5.2.11 boolean setType (String *type*)

Sets the type of the [Assignment](#).

Parameters:

type the type the [Assignment](#) is.

Returns:

true if set, false if failure.

3.5.2.12 boolean setAvg (Float *average*)

Sets the average points of the [Assignment](#).

Parameters:

average the average points of the graded assignment for this [Assignment](#)

Returns:

true if set, false if failure.

3.5.2.13 boolean setMax (Float *max*)

Sets the max points of the [Assignment](#).

Parameters:

max the max points of the graded assignment for this [Assignment](#)

Returns:

true if set, false if failure.

3.5.2.14 boolean setMin (Float *min*)

Sets the min points of the [Assignment](#).

Parameters:

min the min points of the graded assignment for this [Assignment](#)

Returns:

true if set, false if failure.

3.5.3 Member Data Documentation

3.5.3.1 Date dueDate [private]

The due date of the [Assignment](#).

3.5.3.2 String name [private]

The name of the [Assignment](#).

3.5.3.3 Integer total [private]

The maximum possible score of the [Assignment](#).

3.5.3.4 String type [private]

The type of the [Assignment](#); can be Test, Quiz, Homework, Participation, or Other.

3.5.3.5 Float average [private]

The average for all graded assignments for this assignment.

3.5.3.6 Float max [private]

The maximum for all graded assignments for this assignment.

3.5.3.7 Float min [private]

The minimum for all graded assignments for this assignment.

3.6 AssignmentController Class Reference

3.6.1 Detailed Description

Does all of the work on the [Assignment](#), it is the controller.

We haven't really speced out what this is going to do yet.

Author:

bluGoo

Version:

0.42

Public Member Functions

- void [buildMetrics](#) ()
More of a concept than anything.
- boolean [gradeStudent](#) ([Student](#) stu)
Give a student a grade for this assignment.

3.6.2 Member Function Documentation

3.6.2.1 void buildMetrics ()

More of a concept than anything.

3.6.2.2 boolean gradeStudent ([Student](#) stu)

Give a student a grade for this assignment.

Returns:

false if fail, true if pass

Public Member Functions

- boolean `setGradingRules` (`GradingRules` newScale)
setGradingRules saves a new Grading Rules over the old.
- boolean `setCourseTitle` (String newTitle)
setTitle saves a new Course title over the old.
- boolean `setCourseSection` (short newSection)
setSection saves a new section number over the old one
- boolean `setCourseNumber` (int newNumber)
setNumber replaces the course number with a new one
- boolean `setTeacher` (`Teacher` t)
setTeacher sets the `Teacher` attribute for this course
- boolean `setTeacherAssistant` (`TeacherAssistant` ta)
setTeacherAssistant sets the `TeacherAssistant` attribute for this course
- `Teacher` `getTeacher` ()
getTeacher gets the `Teacher` attribute from this course
- `TeacherAssistant` `getTeacherAssistant` ()
getTeacherAssistant gets the `TeacherAssistant` attribute from this course
- Integer `getCourseNumber` ()
getCourseNumber gets the courseNumber
- Integer `getCourseSection` ()
getCourseSection gets the courseSection
- String `getCourseTitle` ()
getCourseTitle gets the courseTitle
- `GradingRules` `getGradingRules` ()
getGradingRules gets the scale attribute for this course
- `ArrayList< Student >` `getStudents` ()
getStudents gets the students attribute for this course

Private Attributes

- String `courseTitle`
The Course's title.
- Integer `courseSection`
The Course's section.

- Integer `courseNumber`
The Course's number.
- `ArrayList< Student > students`
An array of Students who are enrolled in the `Course`.
- `TeacherAssistant assistant`
An instance of `Teacher` Assistant who is associated with the `Course`.
- `Teacher teacher`
An instance of `Teacher` who is teaching the `Course`.
- `GradingRules scale`
The set of grade boundaries for this class.

3.7.2 Member Function Documentation

3.7.2.1 boolean `setGradingRules (GradingRules newScale)`

`setGradingRules` saves a new Grading Rules over the old.

Parameters:

newScale the new grading rules to save

Returns:

true if no errors

3.7.2.2 boolean `setCourseTitle (String newTitle)`

`setTitle` saves a new Course title over the old.

Parameters:

newTitle the new title to save

Returns:

true if no errors

3.7.2.3 boolean `setCourseSection (short newSection)`

`setSection` saves a new section number over the old one

Parameters:

newSection is the number of the new section to save

Returns:

true if no errors occurred

3.7.2.4 boolean setCourseNumber (int *newNumber*)

setNumber replaces the course number with a new one

Parameters:

newNumber is the new course number to save

Returns:

true if no errors occurred.

3.7.2.5 boolean setTeacher (Teacher *t*)

setTeacher sets the [Teacher](#) attribute for this course

Parameters:

t the [Teacher](#) to be assigned to this course

Returns:

true if no error have occurred

3.7.2.6 boolean setTeacherAssistant (TeacherAssistant *ta*)

setTeacherAssistant sets the [TeacherAssistant](#) attribute for this course

Parameters:

ta the [TeacherAssistant](#) to be assigned to this course

Returns:

true if no error have occurred

3.7.2.7 Teacher getTeacher ()

getTeacher gets the [Teacher](#) attribute from this course

Returns:

teacher of this course

3.7.2.8 TeacherAssistant getTeacherAssistant ()

getTeacherAssistant gets the [TeacherAssistant](#) attribute from this course

Returns:

assistant of this course

3.7.2.9 Integer getCourseNumber ()

getCourseNumber gets the courseNumber

Returns:

the courseNumber

3.7.2.10 Integer getCourseSection ()

getCourseSection gets the courseSection

Returns:

the courseSection

3.7.2.11 String getCourseTitle ()

getCourseTitle gets the courseTitle

Returns:

the courseTitle

3.7.2.12 GradingRules getGradingRules ()

getGradingRules gets the scale attribute for this course

Returns:

the scale attribute for this course

3.7.2.13 ArrayList<Student> getStudents ()

getStudents gets the students attribute for this course

Returns:

the students attribute for this course

3.7.3 Member Data Documentation**3.7.3.1 String courseTitle [private]**

The Course's title.

3.7.3.2 Integer courseSection [private]

The Course's section.

3.7.3.3 Integer courseNumber [private]

The Course's number.

3.7.3.4 ArrayList<Student> students [private]

An array of Students who are enrolled in the [Course](#).

3.7.3.5 TeacherAssistant assistant [private]

An instance of [Teacher](#) Assistant who is associated with the [Course](#).

3.7.3.6 Teacher teacher [private]

An instance of [Teacher](#) who is teaching the [Course](#).

3.7.3.7 GradingRules scale [private]

The set of grade boundaries for this class.

3.8 CourseController Class Reference

3.8.1 Detailed Description

Methods for modifying a course and it's related actions.

Author:

bluGoo

Version:

0.42

Public Member Functions

- boolean [addStudent](#) ([Permissions](#) permission, [Student](#) student)
addStudent adds a new student to this course.
- boolean [removeStudent](#) ([Permissions](#) permission, [Student](#) student)
removeStudent removes a particular student from the course.
- boolean [addTA](#) ([Permissions](#) permission, [TeacherAssistant](#) ta)
addTA assigns a new [TeacherAssistant](#) to this course.
- boolean [removeTA](#) ([Permissions](#) permission, [TeacherAssistant](#) ta)
removeTA removes a [TeacherAssistant](#) from this course

3.8.2 Member Function Documentation

3.8.2.1 boolean addStudent ([Permissions](#) *permission*, [Student](#) *student*)

[addStudent](#) adds a new student to this course.

Parameters:

permission the permission of the user who calls this function

student the student that is to be added to the course

Returns:

true if no errors were encountered

3.8.2.2 boolean removeStudent ([Permissions](#) *permission*, [Student](#) *student*)

[removeStudent](#) removes a particular student from the course.

Parameters:

permission the permission of the user who calls this function

student the student to be removed from this course

Returns:

true if no errors were encountered in removal.

3.8.2.3 boolean addTA (Permissions *permission*, TeacherAssistant *ta*)

addTA assigns a new [TeacherAssistant](#) to this course.

Parameters:

permission the permission of the user who calls this function

ta the [TeacherAssistant](#) to be assigned to this course

Returns:

true if no errors have occurred

3.8.2.4 boolean removeTA (Permissions *permission*, TeacherAssistant *ta*)

removeTA removes a [TeacherAssistant](#) from this course

Parameters:

permission the permission of the user who calls this function

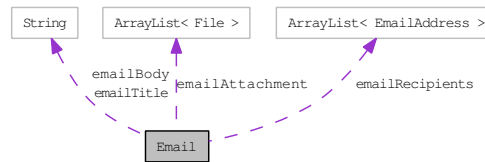
ta the [TeacherAssistant](#) to be removed to this course

Returns:

true if no errors have occurred

3.9 Email Class Reference

Collaboration diagram for Email:



3.9.1 Detailed Description

The class stores information about Emails.

These emails will not be stored in the database. This is purely a way to create a wrapper for validation and sending of emails.

Author:

bluGoo

Version:

0.42

Public Member Functions

- [Email](#) (String title, ArrayList< [File](#) > attachments, ArrayList< [EmailAddress](#) > to, String body)

Constructor for [Email](#) class.

Private Attributes

- String [emailTitle](#)
The title of the [Email](#).
- ArrayList< [File](#) > [emailAttachment](#)
The files to be attached to the [Email](#).
- ArrayList< [EmailAddress](#) > [emailRecipients](#)
An array of email addresses to which the [Email](#) will be sent.
- String [emailBody](#)
The body of the [Email](#).

3.9.2 Constructor & Destructor Documentation

3.9.2.1 Email (String *title*, ArrayList< File > *attachments*, ArrayList< EmailAddress > *to*, String *body*)

Constructor for [Email](#) class.

3.9.3 Member Data Documentation

3.9.3.1 String emailTitle [private]

The title of the [Email](#).

3.9.3.2 ArrayList<File> emailAttachment [private]

The files to be attached to the [Email](#).

3.9.3.3 ArrayList<EmailAddress> emailRecipients [private]

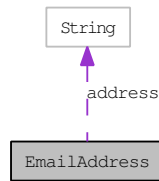
An array of email addresses to which the [Email](#) will be sent.

3.9.3.4 String emailBody [private]

The body of the [Email](#).

3.10 EmailAddress Class Reference

Collaboration diagram for EmailAddress:



3.10.1 Detailed Description

A model for email addresses since they need to be validated and such.

Author:

bluGoo

Version:

0.42

Public Member Functions

- String [toString](#) ()
Obtain a string form of the [Email](#) Address.

Package Functions

- boolean [valid](#) (String in)
Basically a wrapper for validation.

Private Attributes

- String [address](#)

3.10.2 Member Function Documentation

3.10.2.1 boolean valid (String in) [package]

Basically a wrapper for validation.

Parameters:

in [Email](#) address to validate

Returns:

boolean if is a valid email address or not.

3.10.2.2 String toString ()

Obtain a string form of the [Email](#) Address.

Returns:

a String of this [EmailAddress](#)

3.10.3 Member Data Documentation

3.10.3.1 String address [private]

3.11 EmailController Class Reference

3.11.1 Detailed Description

This deals with sending emails.

Author:

bluGoo

Version:

0.42

Public Member Functions

- void [send](#) ()
Sends the email once it has been written.

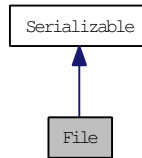
3.11.2 Member Function Documentation

3.11.2.1 void send ()

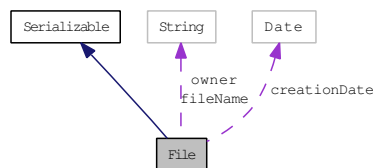
Sends the email once it has been written.

3.12 File Class Reference

Inheritance diagram for File:



Collaboration diagram for File:



3.12.1 Detailed Description

This class keeps information about Files.

Author:

bluGoo

Version:

0.42

Public Member Functions

- String [getOwner](#) ()
Gets the String of the owner for this [File](#).
- Date [getCreationDate](#) ()
Gets the Date of the creationDate for this [File](#).
- String [getFileName](#) ()
Gets the String of the filename for this [File](#).
- [File](#) (String [owner](#), Date [creationDate](#), String [fileName](#))
Constructor for [File](#) class.

Private Attributes

- Date [creationDate](#)

The time and date the [File](#) was created or uploaded.

- String [owner](#)
The owner of the [File](#).
- String [fileName](#)
The name of the [File](#).

3.12.2 Constructor & Destructor Documentation

3.12.2.1 File (String *owner*, Date *creationDate*, String *fileName*)

Constructor for [File](#) class.

3.12.3 Member Function Documentation

3.12.3.1 String `getOwner ()`

Gets the String of the owner for this [File](#).

Returns:

this [File](#)'s owner.

3.12.3.2 Date `getCreationDate ()`

Gets the Date of the `creationDate` for this [File](#).

Returns:

this [File](#)'s creation date and time.

3.12.3.3 String `getFileName ()`

Gets the String of the filename for this [File](#).

Returns:

this [File](#)'s name.

3.12.4 Member Data Documentation

3.12.4.1 Date `creationDate` [private]

The time and date the [File](#) was created or uploaded.

3.12.4.2 String `owner` [private]

The owner of the [File](#).

3.12.4.3 String fileName [private]

The name of the [File](#).

3.13 FileController Class Reference

3.13.1 Detailed Description

This class is to deal with files.

This is not a good implementation.

Author:

bluGoo

Version:

0.42

Public Member Functions

- boolean [upload](#) ([File](#) filename)
Upload to system.
- boolean [download](#) ([File](#) filename)
Download a file.
- boolean [delete](#) ([File](#) filename)
Delete a file.

3.13.2 Member Function Documentation

3.13.2.1 boolean [upload](#) ([File](#) *filename*)

Upload to system.

Parameters:

filename to be uploaded

Returns:

true if successful, false if error.

3.13.2.2 boolean [download](#) ([File](#) *filename*)

Download a file.

Parameters:

filename to be downloaded

Returns:

true if successful, false if error.

3.13.2.3 boolean delete (File *filename*)

Delete a file.

Parameters:

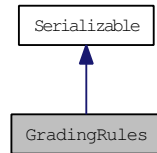
filename to be deleted

Returns:

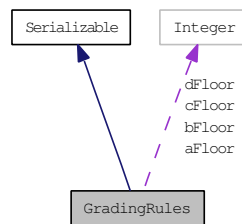
true if successful, false if error.

3.14 GradingRules Class Reference

Inheritance diagram for GradingRules:



Collaboration diagram for GradingRules:



3.14.1 Detailed Description

The [GradingRules](#) class determines how the Total Grade is computed from mutiple [Assignment](#) Grades.

Author:

Blugoo

Version:

0.42

Package Functions

- boolean [setCurve](#) (Character letter, Integer grade)
Sets a the lowest percentage one can have to have a grade.
- Integer [getCurve](#) (Character letter)
Gets a the lowest percentage one can have to have a grade.

Private Attributes

- Integer [aFloor](#)
The lowest percentage a student can have to obtain an A in the course.
- Integer [bFloor](#)
The lowest percentage a student can have to obtain an B in the course.

- Integer [cFloor](#)

The lowest percentage a student can have to obtain an C in the course.

- Integer [dFloor](#)

The lowest percentage a student can have to obtain an D in the course.

3.14.2 Member Function Documentation

3.14.2.1 boolean setCurve (Character *letter*, Integer *grade*) [package]

Sets a the lowest percentage one can have to have a grade.

Parameters:

letter The Grade (A,B,C,D) you wish to change

grade The Percentage you wish to set the floor to.

Returns:

false if fail, true if set.

3.14.2.2 Integer getCurve (Character *letter*) [package]

Gets a the lowest percentage one can have to have a grade.

Parameters:

letter The Grade (A,B,C,D) you wish to get

Returns:

false if fail, true if set.

3.14.3 Member Data Documentation

3.14.3.1 Integer aFloor [private]

The lowest percentage a student can have to obtain an A in the course.

3.14.3.2 Integer bFloor [private]

The lowest percentage a student can have to obtain an B in the course.

3.14.3.3 Integer cFloor [private]

The lowest percentage a student can have to obtain an C in the course.

3.14.3.4 Integer dFloor [private]

The lowest percentage a student can have to obtain an D in the course.

3.15 GradingRulesController Class Reference

3.15.1 Detailed Description

The [GradingRulesController](#) class is the [GradingRules](#) controller.

Author:

Blugoo

Version:

0.42

Public Member Functions

- void [renderPieCharts](#) ()
Renders the Pie Charts.
- boolean [addPercentage](#) (Character letter, Integer amount)
Sets a the lowest percentage one can have to have a grade.
- boolean [subtractPercentage](#) (Character letter, Integer amount)
Sets a the lowest percentage one can have to have a grade.
- boolean [setPercentage](#) (Character letter, Integer grade)
Sets a the lowest percentage one can have to have a grade.

3.15.2 Member Function Documentation

3.15.2.1 void renderPieCharts ()

Renders the Pie Charts.

3.15.2.2 boolean addPercentage (Character *letter*, Integer *amount*)

Sets a the lowest percentage one can have to have a grade.

Parameters:

letter The Grade (A,B,C,D) you wish to change

amount The amount you wish to increment the floor.

Returns:

false if fail, true if set.

3.15.2.3 boolean subtractPercentage (Character *letter*, Integer *amount*)

Sets a the lowest percentage one can have to have a grade.

Parameters:

letter The Grade (A,B,C,D) you wish to change

amount The amount you wish to decrement the floor.

Returns:

false if fail, true if set.

3.15.2.4 boolean setPercentage (Character *letter*, Integer *grade*)

Sets a the lowest percentage one can have to have a grade.

Parameters:

letter The Grade (A,B,C,D) you wish to change

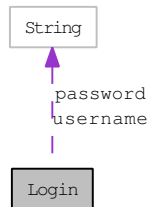
grade The Percentage you wish to set the floor to.

Returns:

false if fail, true if set.

3.16 Login Class Reference

Collaboration diagram for Login:



3.16.1 Detailed Description

This should deal with login I guess.

Taken from the struts example.

Public Member Functions

- String `execute ()` throws Exception
- String `getUsername ()`
- void `setUsername (String username)`
- String `getPassword ()`
- void `setPassword (String password)`

Private Member Functions

- boolean `isInvalid (String value)`

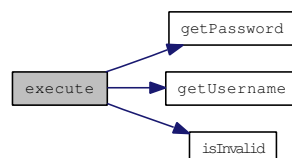
Private Attributes

- String `username`
- String `password`

3.16.2 Member Function Documentation

3.16.2.1 String execute () throws Exception

Here is the call graph for this function:



3.16.2.2 boolean `isInvalid (String value)` [private]

Here is the caller graph for this function:



3.16.2.3 String `getUsername ()`

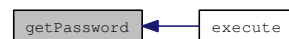
Here is the caller graph for this function:



3.16.2.4 void `setUsername (String username)`

3.16.2.5 String `getPassword ()`

Here is the caller graph for this function:



3.16.2.6 void `setPassword (String password)`

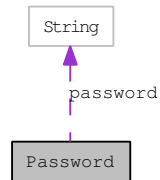
3.16.3 Member Data Documentation

3.16.3.1 String `username` [private]

3.16.3.2 String `password` [private]

3.17 Password Class Reference

Collaboration diagram for Password:



3.17.1 Detailed Description

A model for passwords since they need to be validated and such.

Author:

bluGoo

Version:

0.42

Public Member Functions

- boolean [setPassword](#) (String passwd)

Package Functions

- [Password](#) (String newpass)
Basically a wrapper for hashing.

Private Attributes

- String [password](#)

3.17.2 Constructor & Destructor Documentation

3.17.2.1 Password (String newpass) [package]

Basically a wrapper for hashing.

3.17.3 Member Function Documentation

3.17.3.1 boolean setPassword (String *passwd*)

3.17.4 Member Data Documentation

3.17.4.1 String password [private]

3.18 PasswordController Class Reference

3.18.1 Detailed Description

A controller for passwords since they need to be validated and such.

Author:

bluGoo

Version:

0.42

Public Member Functions

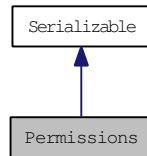
- boolean [validate](#) (String passwd)

3.18.2 Member Function Documentation

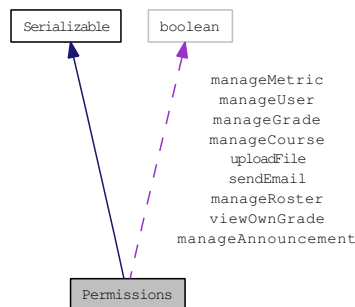
3.18.2.1 boolean validate (String *passwd*)

3.19 Permissions Class Reference

Inheritance diagram for Permissions:



Collaboration diagram for Permissions:



3.19.1 Detailed Description

This class keeps information about permissions for each action.

Author:

bluGoo

Version:

0.42

Public Member Functions

- [Permissions](#) ()
Constructor for [Permissions](#) class.
- boolean [isSendEmail](#) ()
To check if the user has the `sendEmail` permission.
- boolean [isUploadFile](#) ()
To check if the user has the `uploadFile` permission.
- boolean [isViewOwnGrade](#) ()
To check if the user has the `viewOwnGrade` permission.

- boolean [isManageCourse](#) ()
To check if the user has the manageCourse permission.
- boolean [isManageMetric](#) ()
To check if the user has the manageMetric permission.
- boolean [isManageGrade](#) ()
To check if the user has the manageGrade permission.
- boolean [isManageRoster](#) ()
To check if the user has the manageRoster permission.
- boolean [isManageAccouncement](#) ()
To check if the user has the manageAnnouncement permission.
- boolean [isManageUser](#) ()
To check if the user has the manageUser permission.
- boolean [setSendEmail](#) (boolean pSendEmail)
Changes (sets) the sendEmail permission.
- boolean [setUploadFile](#) (boolean pUploadFile)
Changes (sets) the uploadFile permission.
- boolean [setViewOwnGrade](#) (boolean pViewOwnGrade)
Changes (sets) the viewOwnGrade permission.
- boolean [setManageCourse](#) (boolean pManageCourse)
Changes (sets) the manageCourse permission.
- boolean [setManageMetric](#) (boolean pManageMetric)
Changes (sets) the manageMetric permission.
- boolean [setManageGrade](#) (boolean pManageGrade)
Changes (sets) the manageGrade permission.
- boolean [setManageRoster](#) (boolean pManageRoster)
Changes (sets) the manageRoster permission.
- boolean [setManageAnnouncement](#) (boolean pManageAnnouncement)
Changes (sets) the manageAnnouncement permission.
- boolean [setManageUser](#) (boolean pManageUser)
Changes (sets) the manageUser permission.

Private Attributes

- boolean [sendEmail](#)
Permission for sending emails.
- boolean [uploadFile](#)
Permission for uploading files.
- boolean [viewOwnGrade](#)
Permission for viewing personal grades.
- boolean [manageCourse](#)
Permission for managing Courses.
- boolean [manageMetric](#)
Permission for managing metrics.
- boolean [manageGrade](#)
Permission for managing grades.
- boolean [manageRoster](#)
Permission for managing the roster.
- boolean [manageAnnouncement](#)
Permission for managing announcements.
- boolean [manageUser](#)
Permission for managing Users.

3.19.2 Constructor & Destructor Documentation

3.19.2.1 Permissions ()

Constructor for [Permissions](#) class.

3.19.3 Member Function Documentation

3.19.3.1 boolean isSendEmail ()

To check if the user has the sendEmail permission.

Returns:

true if yes, false if no

3.19.3.2 boolean isUploadFile ()

To check if the user has the uploadFile permission.

Returns:

true if yes, false if no

3.19.3.3 boolean isViewOwnGrade ()

To check if the user has the viewOwnGrade permission.

Returns:

true if yes, false if no

3.19.3.4 boolean isManageCourse ()

To check if the user has the manageCourse permission.

Returns:

true if yes, false if no

3.19.3.5 boolean isManageMetric ()

To check if the user has the manageMetric permission.

Returns:

true if yes, false if no

3.19.3.6 boolean isManageGrade ()

To check if the user has the manageGrade permission.

Returns:

true if yes, false if no

3.19.3.7 boolean isManageRoster ()

To check if the user has the manageRoster permission.

Returns:

true if yes, false if no

3.19.3.8 boolean isManageAccouncement ()

To check if the user has the manageAnnouncement permission.

Returns:

true if yes, false if no

3.19.3.9 boolean isManageUser ()

To check if the user has the manageUser permission.

Returns:

true if yes, false if no

3.19.3.10 boolean setSendEmail (boolean *pSendEmail*)

Changes (sets) the sendEmail permission.

Parameters:

pSendEmail A boolean value to which sendEmail is set to.

Returns:

true if set is succesful, false if not

3.19.3.11 boolean setUploadFile (boolean *pUploadFile*)

Changes (sets) the uploadFile permission.

Parameters:

pUploadFile A boolean value to which uploadFile is set to.

Returns:

true if set is succesful, false if not

3.19.3.12 boolean setViewOwnGrade (boolean *pViewOwnGrade*)

Changes (sets) the viewOwnGrade permission.

Parameters:

pViewOwnGrade A boolean value to which viewOwnGrade is set to.

Returns:

true if set is succesful, false if not

3.19.3.13 boolean setManageCourse (boolean *pManageCourse*)

Changes (sets) the manageCourse permission.

Parameters:

pManageCourse A boolean value to which manageCourse is set to.

Returns:

true if set is succesful, false if not

3.19.3.14 boolean setManageMetric (boolean *pManageMetric*)

Changes (sets) the manageMetric permission.

Parameters:

pManageMetric A boolean value to which manageMetric is set to.

Returns:

true if set is succesful, false if not

3.19.3.15 boolean setManageGrade (boolean *pManageGrade*)

Changes (sets) the manageGrade permission.

Parameters:

pManageGrade A boolean value to which manageGrade is set to.

Returns:

true if set is succesful, false if not

3.19.3.16 boolean setManageRoster (boolean *pManageRoster*)

Changes (sets) the manageRoster permission.

Parameters:

pManageRoster A boolean value to which manageRoster is set to.

Returns:

true if set is succesful, false if not

3.19.3.17 boolean setManageAnnouncement (boolean *pManageAnnouncement*)

Changes (sets) the manageAnnouncement permission.

Parameters:

pManageAnnouncement A boolean value to which manageAnnouncement is set to.

Returns:

true if set is succesful, false if not

3.19.3.18 boolean setManageUser (boolean *pManageUser*)

Changes (sets) the manageUser permission.

Parameters:

pManageUser A boolean value to which manageUser is set to.

Returns:

true if set is succesful, false if not

3.19.4 Member Data Documentation**3.19.4.1 boolean sendEmail [private]**

Permission for sending emails.

3.19.4.2 boolean uploadFile [private]

Permission for uploading files.

3.19.4.3 boolean viewOwnGrade [private]

Permission for viewing personal grades.

3.19.4.4 boolean manageCourse [private]

Permission for managing Courses.

3.19.4.5 boolean manageMetric [private]

Permission for managing metrics.

3.19.4.6 boolean manageGrade [private]

Permission for managing grades.

3.19.4.7 boolean manageRoster [private]

Permission for managing the roster.

3.19.4.8 boolean manageAnnouncement [private]

Permission for managing announcements.

3.19.4.9 boolean manageUser [private]

Permission for managing Users.

3.20 PermissionsController Class Reference

3.20.1 Detailed Description

This class keeps the permissions controller.

Author:

bluGoo

Version:

0.42

Public Member Functions

- boolean [VerifyPermission](#) (int inquirer)
VerifyPermission determines if an account has the permission required to perform another method.
- void [AddPermission](#) (int permission)
AddPermission grants a new permission to a user.
- void [RemovePermission](#) (int permission)
Remove Permission prevents a user from using a specific permission.

3.20.2 Member Function Documentation

3.20.2.1 boolean [VerifyPermission](#) (int *inquirer*)

[VerifyPermission](#) determines if an account has the permission required to perform another method.

Parameters:

inquirer indicates the specific permission that is being inquired about.

Returns:

permission is true if the account has permission to perform the inquired action.

3.20.2.2 void [AddPermission](#) (int *permission*)

[AddPermission](#) grants a new permission to a user.

It does nothing to permissions already granted.

Parameters:

permission to determine which permission is to be granted

3.20.2.3 void RemovePermission (int *permission*)

Remove Permission prevents a user from using a specific permission.

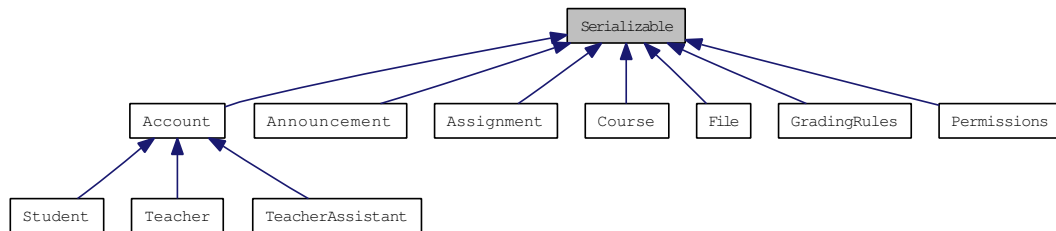
It does nothing to permissions previously not granted.

Parameters:

permission to determine which permission to be removed

3.21 Serializable Class Reference

Inheritance diagram for Serializable:



3.22 StorageConnection Class Reference

3.22.1 Detailed Description

This is the class that deals with connecting to the storage device.

It could also deal with querying the storage system. Look into Hibernate as a way to abstract this more.

Author:

[blugoo](#)

Version:

0.42

Package Functions

- [StorageConnection](#) ()
Connect and be jolly.
- boolean [query](#) (String query)

3.22.2 Constructor & Destructor Documentation

3.22.2.1 StorageConnection () [package]

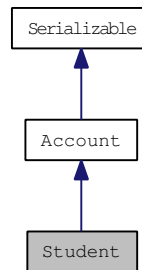
Connect and be jolly.

3.22.3 Member Function Documentation

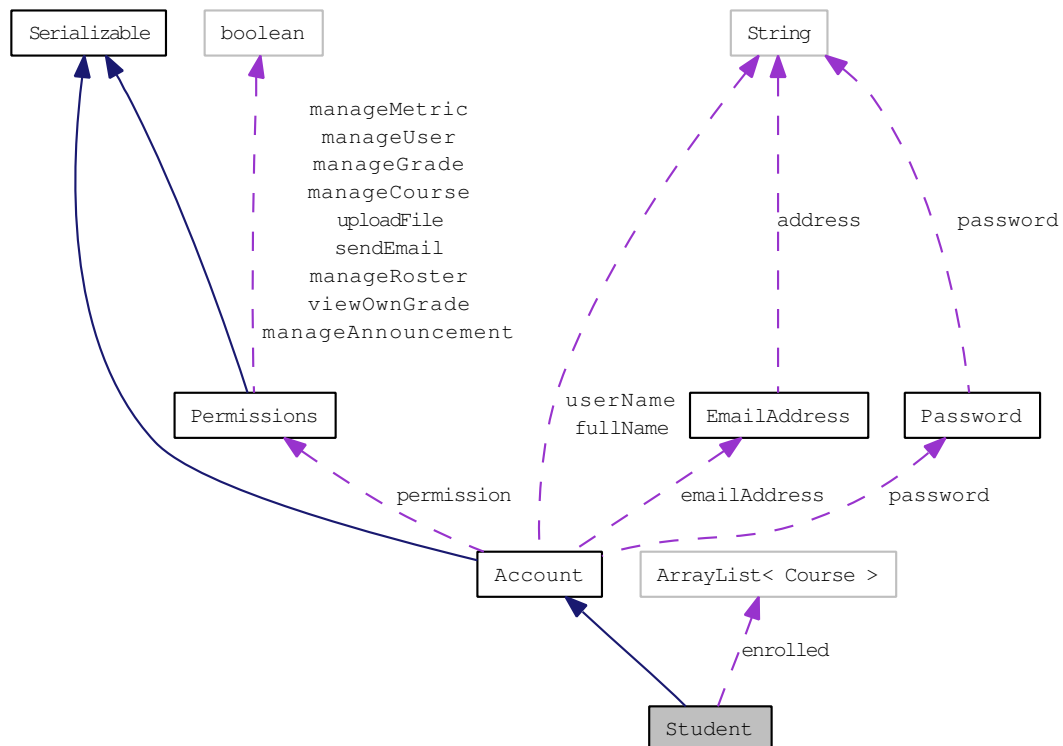
3.22.3.1 boolean query (String *query*) [package]

3.23 Student Class Reference

Inheritance diagram for Student:



Collaboration diagram for Student:



3.23.1 Detailed Description

This class keeps information about Students.

It is an extension of [Account](#). Students have files.

Author:

bluGoo

Version:

0.42

Package Functions

- `int getGrade (String course, Assignment ass)`

*Gets the student grade in the course.***Package Attributes**

- `ArrayList< Course > enrolled`

3.23.2 Member Function Documentation**3.23.2.1 `int getGrade (String course, Assignment ass)` [package]**

Gets the student grade in the course.

Returns:

percentage the student has in the course

Parameters:

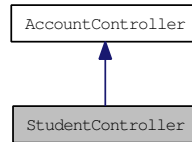
course A string which matches a course name, or other identifier.

ass An assignment to get the grade for.

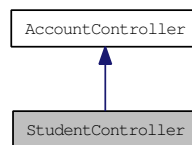
3.23.3 Member Data Documentation**3.23.3.1 `ArrayList<Course> enrolled` [package]**

3.24 StudentController Class Reference

Inheritance diagram for StudentController:



Collaboration diagram for StudentController:



3.24.1 Detailed Description

This class deals with controlling Students.

It is an extension of [AccountController](#).

Author:

bluGoo

Version:

0.42

Public Member Functions

- `ArrayList< Course > viewEnrolledCourses ()`
Get a list of courses a student is enrolled in.

3.24.2 Member Function Documentation

3.24.2.1 `ArrayList<Course> viewEnrolledCourses ()`

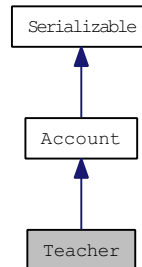
Get a list of courses a student is enrolled in.

Returns:

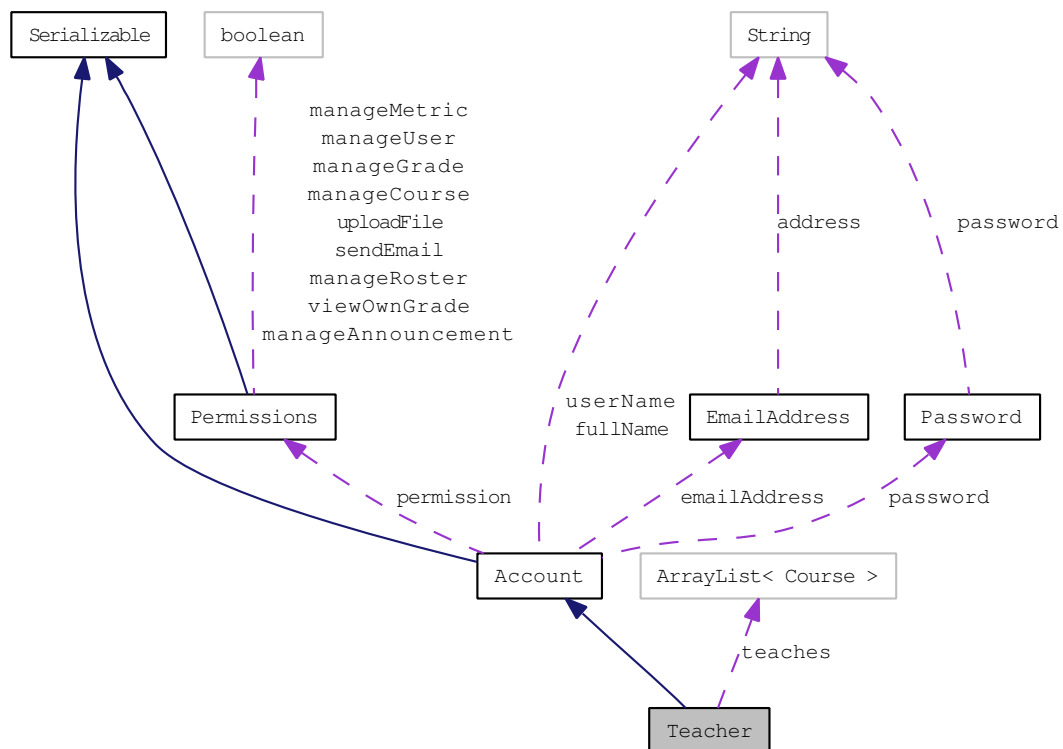
an Array list of Courses.

3.25 Teacher Class Reference

Inheritance diagram for Teacher:



Collaboration diagram for Teacher:



3.25.1 Detailed Description

A [Teacher](#) is an [Account](#) type that has the most permissions.

It acts as the administrator of its installation. There are a lot of methods in here that really should not be in here. I mean seriously. most of the create, edit, and remove methods SHOULD NOT be in here. It makes me cringe everytime I think about it.

Author:

bluGoo

Version:

0.42

Public Member Functions

- [Teacher](#) ()

*Constructor for [Teacher](#).***Private Attributes**

- `ArrayList< Course > teaches`

*A list of courses owned by the [Teacher](#).***3.25.2 Constructor & Destructor Documentation****3.25.2.1 [Teacher](#) ()**

Constructor for [Teacher](#).

All permissions are granted to the [Teacher](#) by default.

3.25.3 Member Data Documentation**3.25.3.1 `ArrayList<Course> teaches` [private]**

A list of courses owned by the [Teacher](#).

Public Member Functions

- [TeacherAssistant](#) ([Permissions](#) granted)
Constructor for [Teacher](#) Assistant.
- [TeacherAssistant](#) ()
Constructor for [Teacher](#) Assistant.

Private Attributes

- `ArrayList< Course > helps`

3.26.2 Constructor & Destructor Documentation

3.26.2.1 [TeacherAssistant](#) ([Permissions](#) *granted*)

Constructor for [Teacher](#) Assistant.

3.26.2.2 [TeacherAssistant](#) ()

Constructor for [Teacher](#) Assistant.

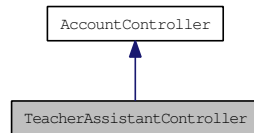
No permissions are passed in, a default, minimal permission set will be initiated.

3.26.3 Member Data Documentation

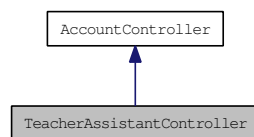
3.26.3.1 `ArrayList<Course> helps` [private]

3.27 TeacherAssistantController Class Reference

Inheritance diagram for TeacherAssistantController:



Collaboration diagram for TeacherAssistantController:



3.27.1 Detailed Description

This class controls [Teacher](#) Assistants.

It is an extension of [AccountController](#).

Author:

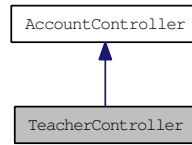
bluGoo

Version:

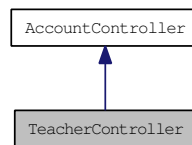
0.42

3.28 TeacherController Class Reference

Inheritance diagram for TeacherController:



Collaboration diagram for TeacherController:



3.28.1 Detailed Description

This is a controller for teacher.

Author:

bluGoo

Version:

0.42

Public Member Functions

- [Course](#) [createCourse](#) ()
createCourse provides the overhead for creating a new class.
- [Course](#) [editCourse](#) ([Course](#) course)
editCourse asks for information from the user and updates the [Course](#) with it
- boolean [removeCourse](#) ([Course](#) course)
removeCourse marks a course for deletion.
- [Announcement](#) [createAnnouncement](#) ([Course](#) which)
createAnnouncement gathers information from the user to make a new [Announcement](#) with
- boolean [gradeStudent](#) ([Student](#) who)
gradeStudent adjusts information in a [Student](#) specified by the user
- [Account](#) [createUser](#) ()
createUser creates a new [Account](#) based on the user's preferences.

- [Account removeUser \(Account user\)](#)
removeUser removes a User from a Course.
- [Account editUser \(\)](#)
editUser provides overhead for Teacher to edit User information
- [Assignment createAssignment \(\)](#)
createAssignment creates a new Assignment with the Teacher's specifications.
- [Assignment editAssignment \(\)](#)
editAssignment changes the information in the Assignment to fit the Teacher's preferences.
- [Assignment deleteAssignment \(\)](#)
deleteAssignment marks an Assignment for removal from the system
- [GradingRules adjustCurve \(\)](#)
adjustCurve modifies the GradingRules to a specific Course
- [boolean takeAttendance \(\)](#)
takeAttendance adjusts the Participation portion of a grade for Students

3.28.2 Member Function Documentation

3.28.2.1 Course createCourse ()

createCourse provides the overhead for creating a new class.

The method prompts the user for information.

Returns:

a new course as specified by the Teacher's input.

3.28.2.2 Course editCourse (Course course)

editCourse asks for information from the user and updates the Course with it

Returns:

the new, edited Course

3.28.2.3 boolean removeCourse (Course course)

removeCourse marks a course for deletion.

Returns:

true if no errors occur

3.28.2.4 Announcement createAnnouncement (Course *which*)

createAnnouncement gathers information from the user to make a new [Announcement](#) with

Parameters:

which The course to add an announcement for.

Returns:

the new, created [Announcement](#).

3.28.2.5 boolean gradeStudent (Student *who*)

gradeStudent adjusts information in a [Student](#) specified by the user

Parameters:

who the [Student](#) to be graded.

3.28.2.6 Account createUser ()

createUser creates a new [Account](#) based on the user's preferences.

If a [Teacher](#) Assistant is being created, their permissions may be given. The new account will be stored in userStorage until they are given a [Course](#), if none specified.

Returns:

the created [Account](#)

3.28.2.7 Account removeUser (Account *user*)

removeUser removes a User from a [Course](#).

If the user is not on any remaining courses, the account is marked for deletion

Returns:

the [Account](#) removed

3.28.2.8 Account editUser ()

editUser provides overhead for [Teacher](#) to edit User information

Returns:

the [Account](#) that was edited

3.28.2.9 Assignment createAssignment ()

createAssignment creates a new [Assignment](#) with the Teacher's specifications.

Returns:

the new [Assignment](#)

3.28.2.10 Assignment editAssignment ()

editAssignment changes the information in the [Assignment](#) to fit the Teacher's preferences.

Returns:

the modified [Assignment](#)

3.28.2.11 Assignment deleteAssignment ()

deleteAssignment marks an [Assignment](#) for removal from the system

Returns:

the [Assignment](#) deleted

3.28.2.12 GradingRules adjustCurve ()

adjustCurve modifies the [GradingRules](#) to a specific [Course](#)

Returns:

the [GradingRules](#) that were modified

3.28.2.13 boolean takeAttendance ()

takeAttendance adjusts the Participation portion of a grade for Students

Returns:

true if no errors occurred

Chapter 4

File Documentation

4.1 blugoo/WEB-INF/src/java/blugoo/Account.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [Account](#)

This is a class that keeps all of the information of a user.

4.2 blugoo/WEB-INF/src/java/blugoo/AccountController.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [AccountController](#)

This class does most of the work on the [Account](#).

4.3 blugoo/WEB-INF/src/java/blugoo/Announcement.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [Announcement](#)
An [Announcement](#) for a [Course](#).

4.4 blugoo/WEB-INF/src/java/blugoo/AnnouncementController.java

File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [AnnouncementController](#)
This is the controller for [Announcement](#).

4.5 blugoo/WEB-INF/src/java/blugoo/Assignment.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [Assignment](#)

This class keeps information about Assignments.

4.6 blugoo/WEB-INF/src/java/blugoo/AssignmentController.java

File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [AssignmentController](#)

Does all of the work on the [Assignment](#), it is the controller.

4.7 blugoo/WEB-INF/src/java/blugoo/Course.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [Course](#)

This class keeps information about Courses.

4.8 blugoo/WEB-INF/src/java/blugoo/CourseController.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [CourseController](#)

Methods for modifying a course and it's related actions.

4.9 blugoo/WEB-INF/src/java/blugoo/Email.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [Email](#)

The class stores information about Emails.

4.10 blugoo/WEB-INF/src/java/blugoo/EmailAddress.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [EmailAddress](#)

A model for email addresses since they need to be validated and such.

4.11 blugoo/WEB-INF/src/java/blugoo/EmailController.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [EmailController](#)
This deals with sending emails.

4.12 blugoo/WEB-INF/src/java/blugoo/File.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [File](#)

This class keeps information about Files.

4.13 blugoo/WEB-INF/src/java/blugoo/FileController.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [FileController](#)

This class is to deal with files.

4.14 blugoo/WEB-INF/src/java/blugoo/GradingRules.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [GradingRules](#)

The [GradingRules](#) class determines how the Total Grade is computed from mutiple [Assignment](#) Grades.

4.15 blugoo/WEB-INF/src/java/blugoo/GradingRulesController.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [GradingRulesController](#)

The [GradingRulesController](#) class is the [GradingRules](#) controller.

4.16 blugoo/WEB-INF/src/java/blugoo/Login.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [Login](#)

This should deal with login I guess.

4.17 blugoo/WEB-INF/src/java/blugoo/Password.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [Password](#)

A model for passwords since they need to be validated and such.

4.18 blugoo/WEB-INF/src/java/blugoo/PasswordController.java

File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [PasswordController](#)

A controller for passwords since they need to be validated and such.

4.19 blugoo/WEB-INF/src/java/blugoo/Permissions.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [Permissions](#)

This class keeps information about permissions for each action.

4.20 blugoo/WEB-INF/src/java/blugoo/PermissionsController.java

File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [PermissionsController](#)

This class keeps the permissions controller.

4.21 blugoo/WEB-INF/src/java/blugoo/StorageConnection.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [StorageConnection](#)

This is the class that deals with connecting to the storage device.

4.22 blugoo/WEB-INF/src/java/blugoo/Student.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [Student](#)

This class keeps information about Students.

4.23 blugoo/WEB-INF/src/java/blugoo/StudentController.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [StudentController](#)

This class deals with controlling Students.

4.24 blugoo/WEB-INF/src/java/blugoo/Teacher.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [Teacher](#)
A [Teacher](#) is an [Account](#) type that has the most permissions.

4.25 blugoo/WEB-INF/src/java/blugoo/TeacherAssistant.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [TeacherAssistant](#)

This class keeps information about [Teacher](#) Assistants.

4.26 blugoo/WEB-INF/src/java/blugoo/TeacherAssistantController.java

File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [TeacherAssistantController](#)
This class controls [Teacher](#) Assistants.

4.27 blugoo/WEB-INF/src/java/blugoo/TeacherController.java File Reference

Namespaces

- namespace [blugoo](#)

Classes

- class [TeacherController](#)

This is a controller for teacher.

4.28 front.txt File Reference

Index

- addPercentage
 - blugoo::GradingRulesController, [43](#)
- AddPermission
 - blugoo::PermissionsController, [58](#)
- address
 - blugoo::EmailAddress, [33](#)
- addStudent
 - blugoo::CourseController, [28](#)
- addTA
 - blugoo::CourseController, [29](#)
- adjustCurve
 - blugoo::TeacherController, [73](#)
- aFloor
 - blugoo::GradingRules, [41](#)
- assistant
 - blugoo::Course, [27](#)
- average
 - blugoo::Assignment, [20](#)
- bFloor
 - blugoo::GradingRules, [41](#)
- blugoo, [3](#)
- blugoo/WEB-INF/src/java/blugoo/Account.java, [75](#)
- blugoo/WEB-INF/src/java/blugoo/AccountController.java,
 - [76](#)
- blugoo/WEB-INF/src/java/blugoo/Announcement.java,
 - [77](#)
- blugoo/WEB-INF/src/java/blugoo/AnnouncementController.java,
 - [78](#)
- blugoo/WEB-INF/src/java/blugoo/Assignment.java,
 - [79](#)
- blugoo/WEB-INF/src/java/blugoo/AssignmentController.java,
 - [80](#)
- blugoo/WEB-INF/src/java/blugoo/Course.java, [81](#)
- blugoo/WEB-INF/src/java/blugoo/CourseController.java,
 - [82](#)
- blugoo/WEB-INF/src/java/blugoo/Email.java, [83](#)
- blugoo/WEB-INF/src/java/blugoo/EmailAddress.java,
 - [84](#)
- blugoo/WEB-INF/src/java/blugoo/EmailController.java,
 - [85](#)
- blugoo/WEB-INF/src/java/blugoo/File.java, [86](#)
- blugoo/WEB-INF/src/java/blugoo/FileController.java,
 - [87](#)
- blugoo/WEB-INF/src/java/blugoo/GradingRules.java,
 - [88](#)
- blugoo/WEB-INF/src/java/blugoo/GradingRulesController.java,
 - [89](#)
- blugoo/WEB-INF/src/java/blugoo/Login.java, [90](#)
- blugoo/WEB-INF/src/java/blugoo/Password.java,
 - [91](#)
- blugoo/WEB-INF/src/java/blugoo/PasswordController.java,
 - [92](#)
- blugoo/WEB-INF/src/java/blugoo/Permissions.java,
 - [93](#)
- blugoo/WEB-INF/src/java/blugoo/PermissionsController.java,
 - [94](#)
- blugoo/WEB-INF/src/java/blugoo/StorageConnection.java,
 - [95](#)
- blugoo/WEB-INF/src/java/blugoo/Student.java, [96](#)
- blugoo/WEB-INF/src/java/blugoo/StudentController.java,
 - [97](#)
- blugoo/WEB-INF/src/java/blugoo/Teacher.java, [98](#)
- blugoo/WEB-INF/src/java/blugoo/TeacherAssistant.java,
 - [99](#)
- blugoo/WEB-INF/src/java/blugoo/TeacherAssistantController.java,
 - [100](#)
- blugoo/WEB-INF/src/java/blugoo/TeacherController.java,
 - [101](#)
- blugoo::Account, [5](#)
- emailAddress, [8](#)
- fullName, [8](#)
- getEmailAddress, [7](#)
- getFullName, [7](#)
- getUserName, [7](#)
- isPassword, [8](#)
- password, [8](#)
- permission, [9](#)
- setEmailAddress, [8](#)
- setFullName, [8](#)
- setPassword, [7](#)
- setUserName, [7](#)
- userName, [8](#)
- blugoo::AccountController, [10](#)
- blugoo::Announcement, [11](#)
- body, [13](#)
- getBody, [12](#)
- getDate, [12](#)
- getTitle, [12](#)

- posted, 13
- setBody, 12
- setDate, 12
- setTitle, 13
- title, 13
- blugoo::AnnouncementController, 14
 - post, 14
- blugoo::Assignment, 15
 - average, 20
 - dueDate, 19
 - getAvg, 17
 - getDueDate, 17
 - getMax, 17
 - getMin, 17
 - getName, 17
 - getTotal, 17
 - getType, 17
 - max, 20
 - min, 20
 - name, 19
 - setAvg, 18
 - setDueDate, 18
 - setMax, 19
 - setMin, 19
 - setName, 18
 - setTotal, 18
 - setType, 18
 - total, 19
 - type, 19
- blugoo::AssignmentController, 21
 - buildMetrics, 21
 - gradeStudent, 21
- blugoo::Course, 22
 - assistant, 27
 - courseNumber, 26
 - courseSection, 26
 - courseTitle, 26
 - getCourseNumber, 25
 - getCourseSection, 26
 - getCourseTitle, 26
 - getGradingRules, 26
 - getStudents, 26
 - getTeacher, 25
 - getTeacherAssistant, 25
 - scale, 27
 - setCourseNumber, 24
 - setCourseSection, 24
 - setCourseTitle, 24
 - setGradingRules, 24
 - setTeacher, 25
 - setTeacherAssistant, 25
 - students, 27
 - teacher, 27
- blugoo::CourseController, 28
 - addStudent, 28
 - addTA, 29
 - removeStudent, 28
 - removeTA, 29
- blugoo::Email, 30
 - Email, 31
 - emailAttachment, 31
 - emailBody, 31
 - emailRecipients, 31
 - emailTitle, 31
- blugoo::EmailAddress, 32
 - address, 33
 - toString, 32
 - valid, 32
- blugoo::EmailController, 34
 - send, 34
- blugoo::File, 35
 - creationDate, 36
 - File, 36
 - fileName, 36
 - getCreationDate, 36
 - getFileName, 36
 - getOwner, 36
 - owner, 36
- blugoo::FileController, 38
 - delete, 38
 - download, 38
 - upload, 38
- blugoo::GradingRules, 40
 - aFloor, 41
 - bFloor, 41
 - cFloor, 41
 - dFloor, 41
 - getCurve, 41
 - setCurve, 41
- blugoo::GradingRulesController, 43
 - addPercentage, 43
 - renderPieCharts, 43
 - setPercentage, 44
 - subtractPercentage, 43
- blugoo::Login, 45
 - execute, 45
 - getPassword, 46
 - getUsername, 46
 - isInvalid, 45
 - password, 46
 - setPassword, 46
 - setUsername, 46
 - username, 46
- blugoo::Password, 47
 - Password, 47
 - password, 48
 - setPassword, 48
- blugoo::PasswordController, 49

- validate, [49](#)
- blugoo::Permissions, [50](#)
 - isManageAccouncement, [53](#)
 - isManageCourse, [53](#)
 - isManageGrade, [53](#)
 - isManageMetric, [53](#)
 - isManageRoster, [53](#)
 - isManageUser, [54](#)
 - isSendEmail, [52](#)
 - isUploadFile, [52](#)
 - isViewOwnGrade, [53](#)
 - manageAnnouncement, [57](#)
 - manageCourse, [56](#)
 - manageGrade, [56](#)
 - manageMetric, [56](#)
 - manageRoster, [56](#)
 - manageUser, [57](#)
 - Permissions, [52](#)
 - sendEmail, [56](#)
 - setManageAnnouncement, [55](#)
 - setManageCourse, [54](#)
 - setManageGrade, [55](#)
 - setManageMetric, [55](#)
 - setManageRoster, [55](#)
 - setManageUser, [56](#)
 - setSendEmail, [54](#)
 - setUploadFile, [54](#)
 - setViewOwnGrade, [54](#)
 - uploadFile, [56](#)
 - viewOwnGrade, [56](#)
- blugoo::PermissionsController, [58](#)
 - AddPermission, [58](#)
 - RemovePermission, [58](#)
 - VerifyPermission, [58](#)
- blugoo::StorageConnection, [61](#)
 - query, [61](#)
 - StorageConnection, [61](#)
- blugoo::Student, [62](#)
 - enrolled, [63](#)
 - getGrade, [63](#)
- blugoo::StudentController, [64](#)
 - viewEnrolledCourses, [64](#)
- blugoo::Teacher, [65](#)
 - Teacher, [66](#)
 - teaches, [66](#)
- blugoo::TeacherAssistant, [67](#)
 - helps, [68](#)
 - TeacherAssistant, [68](#)
- blugoo::TeacherAssistantController, [69](#)
- blugoo::TeacherController, [70](#)
 - adjustCurve, [73](#)
 - createAnnouncement, [71](#)
 - createAssignment, [72](#)
 - createCourse, [71](#)
 - createUser, [72](#)
 - deleteAssignment, [73](#)
 - editAssignment, [73](#)
 - editCourse, [71](#)
 - editUser, [72](#)
 - gradeStudent, [72](#)
 - removeCourse, [71](#)
 - removeUser, [72](#)
 - takeAttendance, [73](#)
- body
 - blugoo::Announcement, [13](#)
- buildMetrics
 - blugoo::AssignmentController, [21](#)
- cFloor
 - blugoo::GradingRules, [41](#)
- courseNumber
 - blugoo::Course, [26](#)
- courseSection
 - blugoo::Course, [26](#)
- courseTitle
 - blugoo::Course, [26](#)
- createAnnouncement
 - blugoo::TeacherController, [71](#)
- createAssignment
 - blugoo::TeacherController, [72](#)
- createCourse
 - blugoo::TeacherController, [71](#)
- createUser
 - blugoo::TeacherController, [72](#)
- creationDate
 - blugoo::File, [36](#)
- delete
 - blugoo::FileController, [38](#)
- deleteAssignment
 - blugoo::TeacherController, [73](#)
- dFloor
 - blugoo::GradingRules, [41](#)
- download
 - blugoo::FileController, [38](#)
- dueDate
 - blugoo::Assignment, [19](#)
- editAssignment
 - blugoo::TeacherController, [73](#)
- editCourse
 - blugoo::TeacherController, [71](#)
- editUser
 - blugoo::TeacherController, [72](#)
- Email
 - blugoo::Email, [31](#)
- emailAddress
 - blugoo::Account, [8](#)

- emailAttachment
 - blugoo::Email, 31
- emailBody
 - blugoo::Email, 31
- emailRecipients
 - blugoo::Email, 31
- emailTitle
 - blugoo::Email, 31
- enrolled
 - blugoo::Student, 63
- execute
 - blugoo::Login, 45
- File
 - blugoo::File, 36
- fileName
 - blugoo::File, 36
- front.txt, 102
- fullName
 - blugoo::Account, 8
- getAvg
 - blugoo::Assignment, 17
- getBody
 - blugoo::Announcement, 12
- getCourseNumber
 - blugoo::Course, 25
- getCourseSection
 - blugoo::Course, 26
- getCourseTitle
 - blugoo::Course, 26
- getCreationDate
 - blugoo::File, 36
- getCurve
 - blugoo::GradingRules, 41
- getDate
 - blugoo::Announcement, 12
- getDueDate
 - blugoo::Assignment, 17
- getEmailAddress
 - blugoo::Account, 7
- getFileName
 - blugoo::File, 36
- getFullName
 - blugoo::Account, 7
- getGrade
 - blugoo::Student, 63
- getGradingRules
 - blugoo::Course, 26
- getMax
 - blugoo::Assignment, 17
- getMin
 - blugoo::Assignment, 17
- getName
 - blugoo::Assignment, 17
- getOwner
 - blugoo::File, 36
- getPassword
 - blugoo::Login, 46
- getStudents
 - blugoo::Course, 26
- getTeacher
 - blugoo::Course, 25
- getTeacherAssistant
 - blugoo::Course, 25
- getTitle
 - blugoo::Announcement, 12
- getTotal
 - blugoo::Assignment, 17
- getType
 - blugoo::Assignment, 17
- getUserName
 - blugoo::Account, 7
- getUsername
 - blugoo::Login, 46
- gradeStudent
 - blugoo::AssignmentController, 21
 - blugoo::TeacherController, 72
- helps
 - blugoo::TeacherAssistant, 68
- isInvalid
 - blugoo::Login, 45
- isManageAccouncement
 - blugoo::Permissions, 53
- isManageCourse
 - blugoo::Permissions, 53
- isManageGrade
 - blugoo::Permissions, 53
- isManageMetric
 - blugoo::Permissions, 53
- isManageRoster
 - blugoo::Permissions, 53
- isManageUser
 - blugoo::Permissions, 54
- isPassword
 - blugoo::Account, 8
- isSendEmail
 - blugoo::Permissions, 52
- isUploadFile
 - blugoo::Permissions, 52
- isViewOwnGrade
 - blugoo::Permissions, 53
- java::io::Serializable, 60
- manageAnnouncement

- blugoo::Permissions, 57
- manageCourse
 - blugoo::Permissions, 56
- manageGrade
 - blugoo::Permissions, 56
- manageMetric
 - blugoo::Permissions, 56
- manageRoster
 - blugoo::Permissions, 56
- manageUser
 - blugoo::Permissions, 57
- max
 - blugoo::Assignment, 20
- min
 - blugoo::Assignment, 20
- name
 - blugoo::Assignment, 19
- owner
 - blugoo::File, 36
- Password
 - blugoo::Password, 47
- password
 - blugoo::Account, 8
 - blugoo::Login, 46
 - blugoo::Password, 48
- permission
 - blugoo::Account, 9
- Permissions
 - blugoo::Permissions, 52
- post
 - blugoo::AnnouncementController, 14
- posted
 - blugoo::Announcement, 13
- query
 - blugoo::StorageConnection, 61
- removeCourse
 - blugoo::TeacherController, 71
- RemovePermission
 - blugoo::PermissionsController, 58
- removeStudent
 - blugoo::CourseController, 28
- removeTA
 - blugoo::CourseController, 29
- removeUser
 - blugoo::TeacherController, 72
- renderPieCharts
 - blugoo::GradingRulesController, 43
- scale
 - blugoo::Course, 27
- send
 - blugoo::EmailController, 34
- sendEmail
 - blugoo::Permissions, 56
- setAvg
 - blugoo::Assignment, 18
- setBody
 - blugoo::Announcement, 12
- setCourseNumber
 - blugoo::Course, 24
- setCourseSection
 - blugoo::Course, 24
- setCourseTitle
 - blugoo::Course, 24
- setCurve
 - blugoo::GradingRules, 41
- setDate
 - blugoo::Announcement, 12
- setDueDate
 - blugoo::Assignment, 18
- setEmailAddress
 - blugoo::Account, 8
- setFullName
 - blugoo::Account, 8
- setGradingRules
 - blugoo::Course, 24
- setManageAnnouncement
 - blugoo::Permissions, 55
- setManageCourse
 - blugoo::Permissions, 54
- setManageGrade
 - blugoo::Permissions, 55
- setManageMetric
 - blugoo::Permissions, 55
- setManageRoster
 - blugoo::Permissions, 55
- setManageUser
 - blugoo::Permissions, 56
- setMax
 - blugoo::Assignment, 19
- setMin
 - blugoo::Assignment, 19
- setName
 - blugoo::Assignment, 18
- setPassword
 - blugoo::Account, 7
 - blugoo::Login, 46
 - blugoo::Password, 48
- setPercentage
 - blugoo::GradingRulesController, 44
- setSendEmail
 - blugoo::Permissions, 54
- setTeacher
 - blugoo::Course, 25

- setTeacherAssistant
 - blugoo::Course, 25
- setTitle
 - blugoo::Announcement, 13
- setTotal
 - blugoo::Assignment, 18
- setType
 - blugoo::Assignment, 18
- setUploadFile
 - blugoo::Permissions, 54
- setUserName
 - blugoo::Account, 7
- setUsername
 - blugoo::Login, 46
- setViewOwnGrade
 - blugoo::Permissions, 54
- StorageConnection
 - blugoo::StorageConnection, 61
- students
 - blugoo::Course, 27
- subtractPercentage
 - blugoo::GradingRulesController, 43
- takeAttendance
 - blugoo::TeacherController, 73
- Teacher
 - blugoo::Teacher, 66
- teacher
 - blugoo::Course, 27
- TeacherAssistant
 - blugoo::TeacherAssistant, 68
- teaches
 - blugoo::Teacher, 66
- title
 - blugoo::Announcement, 13
- toString
 - blugoo::EmailAddress, 32
- total
 - blugoo::Assignment, 19
- type
 - blugoo::Assignment, 19
- upload
 - blugoo::FileController, 38
- uploadFile
 - blugoo::Permissions, 56
- userName
 - blugoo::Account, 8
- username
 - blugoo::Login, 46
- valid
 - blugoo::EmailAddress, 32
- validate
 - blugoo::PasswordController, 49
 - VerifyPermission
 - blugoo::PermissionsController, 58
 - viewEnrolledCourses
 - blugoo::StudentController, 64
 - viewOwnGrade
 - blugoo::Permissions, 56