SubIt!

Group 1

Joel Westberg
Mikael Granholm
Simon Stenström
Sofie Björk
Henrik Eriksson Hegardt

5.5 Detailed Design

class Page

Functions
General Note:
All functions below take into account the priveleges of the logged in user.

- *printAssignmentPage(int id)*
  Generates and shows the assignment page for the specified assignment.
  If no id is given, the page for creating a new assignment is shown.

- *printSubmissionPage(int id)*
  Generates and shows the submission page for the specified submission.
  If no id is given, the page for making a new submission is shown.

- *printCoursePage(int id)*
  Generates and shows the course page for the specified course. If no id
  is given, the page for creating a new course is shown.

- *printPGPage(int id)*
  Generates and shows the project group page for the specified project
  group.

- *printStudentList(int course_id)* Generates and shows a list of all students in the specified course.

- *printAssignmentPage(int id)*
  Generates and shows the assignment page for the specified assignment.

- *printInbox()*
  Generates and shows the inbox.

- *printMessage(int id)*
  Generates and shows the page for the specified private message.

- *printCompose(int[] to)*
  Generates and shows the page for composing a new private message.
  Any element in the argument array will be already be filled in as
  receivers of the message.

- *printLoginPage()*
  Generates and shows the page with the login form.

- *printPGList(int course_id)*
  Generates and shows a list of all Project Groups for the specified
  course.

- *printSubmissionList(int type, int id)*
  Generates and shows a list of all submissions. The argument type specifies what type of list to generate, be it for a student, or for an entire course. The id argument allows specification of student or course.

- *printCreateUserPage()*
  Generates and shows the page for adding a user to the system.

Variables
This class has no class variables.

class Assignment

Functions

- *getId()*
  Returns id.

- *getCourse()*
  Returns course.

- *getSoftDeadline()*
  Returns softDeadline.

- *getHardDeadline()* Returns hardDeadline.

- *getStartDate()*
  Returns startDate.

- *getDescription*
  Returns description.

- *setCourse(int id)*
  Sets course.

- *setSoftDeadline(long deadline)*
  Sets softDeadline.

- *setHardDeadline(long deadline)*
  Sets hardDeadline.

- *setStartDate(long deadline)*
  Sets startDate.

- *setDescription(string desc)*
  Sets description.

- *getSubmissions()*
  Returns an array of id's for all submissions for this assignment.

- *save()*
  Saves all data to the database.

Variables

- int id

- int course

- long softDeadline

- long hardDeadline

- long startDate

- string description

class User

Functions

- *getRealName()*
  Returns the real name of a user as a String.

- *setRealName(String realName)*
  Sets the real name of a user.

- *getUserName()*
  Returns the user name of a user as a String.

- *setUserName(String userName)*
  Sets the user name of a user.

- *getPIDN()*
  Returns the personal identification number of a user as a String.

- *setPIDN(String pIDN)*
  Sets the personal identification number of a user.

- *isPassword(String pw)*
  Generates and shows the assignment page for the specified assignment.

- *setPassword(String pw)*
  Sets the password of a user.

- *getCourses()*
  Returns an array with the CourseIDs of the courses the user is participating in.

- *setProjectGroup()*
  Returns the ProjectGroupID of the project group a user has joined.

- *save()*
  Saves the changes to the database.

Variables

- int id

- String userName, realName, pIDN

class Course

Functions

- *getCourseId()*
  Returns the course id.

- *getCourseName()*
  Returns the course name.

- *setCourseName(string s)*
  Set the course name to *s*.

- *getDescription()*
  Returns the course description.

- *setDescription(string s)*
  Set the course description to *s*.

- *getCourseLeader()*
  Returns the user id of the course leader.

- *setCourseLeader(int i)*
  Set the course leader to user with id *i*.

- *getTeachers()*
  Returns an array with user ids of the teachers assigned to the course.

- *addTeacher(int i)*
  Add user with id *i* to the list of teachers in the course.

- *removeTeacher(int i)*
  Remove the user with id *i* from the list of teachers in the course.

- *getAssignments()*
  Returns an array with the assignment ids of the course assignments.

- *addParticipant(int i)*
  Add user with user id *i* to the list of participants in the course.

- *removeParticipant(int i)*
  Remove user with user id *i* from the list of participants in the course.

- *save()*
  Save to database.

Variables

- *int id*
  Course id.

- *int array teachers*
  Array with user id of teachers assigned to course.

- *string description*
  Course description.

- *string name*
  The name of the course.

- *int courseLeader*
  The user id the course leader.

- *int array assignments*
  Array with assignment id of the course assignments.

class ProjectGroup

- *getId()*
  Returns the id as an int.

- *getCourse()*
  Returns the CourseID of as an int.

- *setCourse(int CourseID)*
  Sets the course of a the project group.

- *getName()*
  Returns the name of the project group as a String.

- *setName(String name)*
  Sets the name of the project group.

- *getUsers()*
  Returns the userIDs of the users in the project group.

- *addUser(int userID)*
  Adds a user to the project group.

- *removeUser(int userID)*
  Removes a user from the project group.

Variables

- int id, course

- String name

class Message

Functions

- *getId()*
  Returns the id.

- *getFrom()*
  Returns the senders id.

- *setFrom(int id)*
  Sets sender id.

- *getTo()*
  Returns the recipients.

- *addTo(int id)*
  Adds id to list of recipients.

- *setTo(mixed ids)*
  Sets id's in array to be the recipients. Accepts arrays of id's or comma formated string as argument.

- *getTitle()*
  Returns the title of the message.

- *setTitle(string name)*
  Sets the title.

- *getBody()*
  Returns the body.

- *setBody(string message)*
  Sets message body.

- *save()*
  Saves the message. Equivalent to sending it.

- *getRead()*
  Return read status.

- *setRead(bool read)*
  Sets mesage read status.

Variables

- int[] to

- int from, id

- string body, title

- bool read

class Submission

Functions

- *getId()*
  Returns the id as an int.

- *getAssignment()*
  Returns the assignment of as an int.

- *setAssignment(int assignmentID)*
  Sets the assignment.

- *getCreator()*
  Returns the creator of the submission as an int.

- *setCreator(int userID)*
  Sets the creator of the submission.

- *getProjectGroup()*
  Returns the project group as an int.

- *setProjectGroup(int projectGroup)*
  Sets the project group.

- *getFiles()*
  Returns the files of the submission as a String[].

- *addFile(String file)*
  Adds a file to the files of the submission.

- *getSComment()*
  Returns the sComment as a String.

- *setSComment(String sComment)*
  Sets the sComment.

- *getTComment()*
  Returns the tComment as a String.

- *setTComment(String tComment)*
  Sets the tComment.

- *getGrade()*
  Returns the grade as a String.

- *setGrade(String grade)*
  Sets the grade.

- *getTimestamps()*
  Returns the timestamps as a long[].

- *setTimestamps(long[] timestamps)*
  Sets the timestamps.

- *save()*
  Saves the data to the database.

Variables

- int id, assignment, creator, projectgroup

- String sComment, tComment, grade

- String[] files

- long[] timestamps

5.6

SubIt! consists of two packages. The package *subIt!* represents the system as a whole. In the package *subIt!* are the class Page which generates the pages that is sent to the client webbrowser and the subpackage *objects* on which the class Page depends. The subpackage *objects* contains classes that represents the different functions of the SubIt! system.

subIt!

Page

objects
+Assignment
+User
+Course
+ProjectGroup
+Message
+Submission