Report

Intranet

Prepared by:

Gumar Oralkhan

Kendybayeva Dana

Khinatolla Nurbergen

18BD

# Class Description

1. User – is an abstract class that implemented by Employee and Student. Passwords are stored as hash. Methods:

* logIn(String userName, String password) – static method which returns object of User
* logOut() – exits the user system

1. Employee – is an abstract class that implemented by all workers. Method:

* sendOrder(String message, Employee sender) – static method which sends order from any employee to IT specialist

1. Admin – class that implements Employee. Its objects can view log files and make changes in user information. Has access to databases. Methods:

* addUser (User user) – creates new user
* removeUser(User user) – removes user
* viewLogFile() – shows the actions that users did, like added course, sent order, etc.
* viewInfo(Employee e) – shows information about employee
* viewInfoAboutEmployees() – show info about all employees

1. Executor – class that implements Employee. Can access database for getting information about orders. Methods:

* viewOrders() – shows the list of orders that were sent to the executor
* acceptOrder(Order order) – changes the status of order from new to accepted
* rejectOrder(Order order) – changes the status of order from new to rejected
* findOrder(Order order) – finds order

1. Order – class for describing orders. There are fields like sender of type Employee(for storing the sender), time the order was sent and the message. Method:

* changeStatus(Executor e, Status newStatus) – changes status of the order

1. Manager - class that implements Employee. There is field typeOfManager that defines which manager(Офис Регестратора или Деканата). Have access to database for getting information about students and workers and for adding the course to registration. Methods:

* addNews(String news) – adds news
* addRegistrateCourses(Course course, HashSet<Teacher> teachers) – creates the list of courses that can be taken in semester
* viewInfoAboutEmployee(Employee employee) – shows information about employee
* viewStudentInfo(Student student) – gives information about student

1. Teacher - class that implements Employee. Can have access to database for getting information about students and putting marks. Can view course files. There is enumerator Degree. Methods:

* viewStudentInfo(Student student, Course course) – shows information about student from teacher's course
* viewStudentsInfo(Course c) – shows information of all students from teacher's course
* putMark(Student student, Course course, Mark mark) – puts mark to student in the course
* putMarks(Course course, ArrayList<Mark> marks) – puts mark to all student at once
* viewCourseFiles() – shows files of teacher of specific course
* viewCourses() – shows the list of courses that teacher teaches
* addCourse(Course course) – adds new course
* removeCourse(Course course) – removes course
* setTime(Course course, DayAndTime dayTime) – set time for course to be held during the week, it will be used in schedule making

1. CourseFile – class for storing course files. There is fields String title, String description, Date date
2. Course – class of courses. Stores object of Teacher, HashSet<Course> prerequisites (for storing prerequisites), HashSet<Faculty> (enumerator Faculty, which shows which faculties can take this course), int semesterOfStudy(which semester it can be taken)
3. Student – class that implements User. Stores object of Transcript(stores current transcript), SortedSet<Transcript> (stores previous transcripts), Schedule (schedule of student), Attendance (attendance of student), Faculty(enumerator), int debt(student’s debt), boolean hasAccess(shows whether there is access to intranet). Methods:

* registrateToCourse(Course course) – register to course
* dropCourse(Course course) – drop course
* viewTranscript() – shows the transcript
* registrateToCourse(HashSet<Course> courses) – register to several courses at a time
* chooseTeacher(Course course, Teacher teacher) – chooses the course and teacher, used for making schedule

1. Mark – class of marks, stores marks of two attestations and final. Methods:

* getTotal() – gives total mark of all attestations
* changeToLetter(double value) – changes the mark to letter mark

1. Transcript – connects course and marks and has field String semester(shows current semester), int year(year of study). Methods:

* addCourse(Course course) – adds course to transcript
* addCourses(HashSet<Course> course) – adds several courses to transcript at once
* dropCorse(Course course) – drops course
* setMark(Course course, Mark m) – sets mark to course

1. Attendance – shows attendance of student, has fields HashMap<Course, Integer> numOfAbsences(number of absences), HashMap<Course, Integer> numOfDelays (number of delays). Methods:

* addAbsence(Course c) – increases the number of absences
* addDelay(Course c) – increases the number of delays

1. Schedule – schedule of student. SortedMap<DayAndTime, Course> schedule stores object of DayAndTime(stores day and time) and course
2. DayAndTIme – class for storing day and time of the course. Has objects of enumerators Day and Time
3. IncorrectPasswordOrLoginException - Exception. Shows whether the login and password correct or not
4. Singletone – class with static methods. Used for accessing database
5. Degree - enum. Shows the degree of teacher
6. Status - enum. Shows status of Order
7. Days - enum. Shows day of the week
8. Time - enum. Shows time of the day
9. TypeOfManager - enum. Shows type of manager
10. Faculty - enum. Shows faculties
11. Attestation - enum. Shows type of attestation (1st attestation, 2nd attestation, final)