SystemDomain
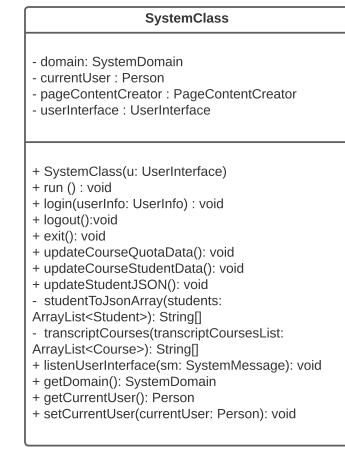
- lecturers: ArrayList<Lecturer>
- advisors: ArrayList<Advisor>
- courses: ArrayList<Course>
- students: ArrayList<Student>

+ SystemDomain()
+ createLecturers(): void
+ createAdvisors(): void
+ createStudents(): void
- setStudentCourses(studentCoursesAr: String[ ]): ArrayList<Course>
- createCourses(): void
+fillStudentListCourse(): void
+ jsonArrToStrArr(jsonArray: JSONArray): String[ ]
- assignStudentsToAdvisor(): void
- assignCoursesToLecturer(): void
+ createPages(currentUser : Person): ArrayList<Page>
+ getter methods

PageContentCreator

+crateMainMenuPageStudentContent(): String
+createMainMenuPageAdvisorContent() :String
+createSelectedStudentsRequesPageContent (student : Student): String
+createEvaluateRequestPageContent (student: ArrayList<Student>): String
+createMyStudentsPageContent (student: ArrayList<Student>) : String
+createSelectableCoursesPageContent (courses: ArrayList<Course>, coursesSelected:ArrayList<Course>):String
+createSelectedCoursesPageContent (courses:ArrayList<Course>):String
+createAllCoursesPageContent (courses:ArrayList<Course>): String
+createApprovedCoursesPageContent (courses:ArrayList<Course>): String
+courseListForContent (courses:ArrayList<Course>):String

UserInterface

- currnetPage : Page
- pages : ArrayList <Page>
- consoleScanner : Scanner

+ UserInterface()
+ displayPage()
+ setCurrendPage(pageType : PageType) :
boolean
+ addPage(page : Page) : boolean
+ selectPage(type : PageType) : Page
+ resetPages() : void

**<<enumaration>>
PageType**

LOGIN_PAGE
MAIN_MENU_PAGE_STUDENT
ALL_cOURSES_PAGE
SELECTABLE_COURSES_PAGE
SELECTED_COURSES_PAGE
APPROVED_COURSES_PAGE
MAIN_MENU_PAGE_ADVISOR
MY_STUDENTS_PAGE
MY_REQUESTS_PAGE
EVALUATE_REQUESTS_PAGE
SELECTED_STUDENT_REQUEST_PAGE

**<>
Page**

- type : PageType
- name : String
- content : Stirng

+ Page(content : String)
+ runPage() : systemMessage : SystemMessage
+ showContent() : void
+ takeInput() : String
+ getType() : PageType
+ setType(pageType : PageType) : void
+ getName() : String
+ setName(name :String) : void
+ getContent() : String
+ setContent(content : String) : void

SystemMessage

- functionType : FunctionType
- pageType : PageType
- input : Object

+ SystemMessage(functionType : FunctionType,
pageType : PageType, input : Object)
+ getFunctionType() : FunctionType
+ setFunctionType(functionType : FunctionType): void
+ getPageType() : PageType
+ setPageType(pageType :PageType) : void
+ getInput() : Object
+ setInput(input : Object) : void

**<<enumaration>>
FunctionType**

NONE
LOGIN
EXIT
CHANGE_PAGE
SELECT_COURSE
DROP_COURSE
SELECET_STUDENT
APPROVE_REQUEST
DISAPPREOVE_REQUEST

SystemClass

- domain: SystemDomain
- currentUser : Person
- pageContentCreator : PageContentCreator
- userInterface : UserInterface

+ SystemClass(u: UserInterface)
+ run () : void
+ login(userInfo: UserInfo) : void
+ logout():void
+ exit():void
+ updateCourseQuotaData(): void
+ updateCourseStudentData(): void
+ updateStudentJSON(): void
- studentToJsonArray(students:
ArrayList<Student>): String[]
- transcriptCourses(transcriptCoursesList:
ArrayList<Course>): String[]
+ listenUserInterface(sm: SystemMessage): void
+ getDomain(): SystemDomain
+ getCurrentUser: Person
+ setCurrentUser(currentUser: Person): void

LoginPage

- userInfo: UserInfo

+ LoginPage(content :String)
+ runPage() : systemMessage : SystemMessage
+ getUserInfo() : UserInfo
+ serUserInfo(userInfo : UserInfo) : void

UserInfo

- username : String
- password : String

+ UserInfo (name : String, password : String)
+ reset() : void
+ getUserName() : Sting
+ setUserName(name : String) : void
+ getPassword() : Sting
+ setPassword(password : String) : void

MainMenuPageStudent

-

+ MainMenuPageStudent(content : String)
+ + runPage() : systemMessage : SystemMessage

MainMenuPageAdvisor

-

+ MainMenuPageAdvisor(content : String)
+ runPage() : systemMessage : SystemMessage

AllCoursesPage

-

+ AllCoursesPage(content : String)
+ runPage() : systemMessage : SystemMessage

MyStudentsPage

-

+ MyStudentsPage(content : String)
+ runPage() : systemMessage : SystemMessage

SelectableCoursesPage

- numberOfSelectableCourses : int

+ SelectableCoursesPage(content : String)
+ runPage() : systemMessage : SystemMessage
+ getNumberOfSelectableCourses() : int
+ setNumberOfSelectableCourses(num : int) : void

SelectedCoursesPage

- numberOfDropableCourses : int

+ SelectedCoursesPage(content : String)
+ runPage() : systemMessage : SystemMessage
+ getNumberOfDropableCourses() : int
+ setNumberOfDropableCourses(num : int) : void

EvaluateRequestPage

- numberOfRequest : int

+ EvaluateRequestPage(content : String)
+ runPage() : systemMessage : SystemMessage
+ getNumberOfRequest() : int
+ setNumberOfRequest(num : int) : void

ApprovedCoursesPage

-

+ ApprovedCoursesPage(content : String)
+ runPage() : systemMessage : SystemMessage

SelectedStudentsRequesPage

-

+ SelectedStudentsRequesPage(content : String)
+ runPage() : systemMessage : SystemMessage

**<<Abstract>>
Person**

- firstName : String
- lastName : String

+ Person(firstName: String,
lastName: String)
+getter setter methods

Student

- studentId : Id
- password: Password
- advisor: Advisor
- transcript: Transcript
- selectableCourse : ArrayList <Course>
- selectedCourses : Arraylist <Course>
- approvedCourses : Arraylist <Course>
- curriculum : Arraylist <Course>
- request : String
- notification : String

+ Student (firstName: String , lastName: String ,
studentID: Id , password: Password, advisor:
Advisor , transcript: Transcript , selectedCourses:
ArrayList<Course> , approvedCourses:
ArrayList<Course> , request: Boolean ,
notification: String , curriculum: ArrayList
<Course>)
+ filterCourses (curriculum: ArrayList <Course>) :
void
+isSelectedCourse(course: Course) : boolean
+isPassedCourse(course: Course) : boolean
+ isPrerequisiteCoursesPassed(course: Course) :
boolean
+ isUnderQuota(course: Course) : boolean
+ isFailedCourse(course: Course) : boolean
+ addSelectableCourse(course: Course) : void
+addApprovedCourse(course: Course) : void
+ dropAllSelectedCourses() :void
+ sendToApproval () : void
+ addSelectedCourse (i : int) :void
+ dropCourse (i : int) : void
+addAllSessions(course: Course) :void
+ removeAllSessions(course: Course):void
+ getter setter methods

Lecturer

- lecturerID : LecturerID
- givenCourses:
ArrayList<Course>

+Lecturer(firstName: String,
lastName: String, lecturerID: Id)
+ getter setter methods

Advisor

- students : ArrayList <Student>
- awaitingStudents: ArrayList
<Student>
- selectStudent: Student
- password : Password

+ Advisor (firstName: String,
lastName: String, lecturerID : Id,
password: Password)
+ findAwaitingStudentList(): void
+ selectStudent (index: int): void
+ Approve(): void
+ Disapprove () : void
+sendNotification (message :
String, type: String): void
+removeAwaitingStudent(student:
Student): void
+ getter setter methods

Id

id: String

setID(id: String) : void
getID() : Id

Password

- password : String

+ Password(password: String)
+ getter setter methods

Transcript

- GPA _100: double
- year : int
- passedCourses : ArrayList
<Course>
- failedCourses : ArrayList
<Course>

+ Transcript(GPA _100: double
, year : int , passedCourses :
ArrayList <Course> ,
failedCourses : ArrayList
<Course>)
+ getter setter methods

Course

- courseId: Id
- courseName: String
- quota : int
- year: int
- day_hour : String
- prerequisiteCourses:
ArrayList<Course>
- studentList : ArrayList<Student>
- lecturer : Lecturer

+Course(courseId: Id,
courseName: String , quota: int ,
year: int , day_hour: String,
lecturer: Lecturer)
+enrollStudent(student: Student):
void
+getter setter methods

CourseSession

- sessionId : Id

+ CourseSession(courseID: Id,
courseName: String, quota: int ,
year: int, day_hour: String,
lecturer: Lecturer, sessionId: Id)
+getter setter methods