Cairo University

Faculty of Computers and Information

CS352 – Software Engineering II

Phase 2 Template

2017

Staff:

Dr Amr Kamel	a.kamel@fci-cu.edu.eg
Dr Khadiga Mohamed	kelbedweihy@fci-cu.edu.eg

TA: Ragia Mohamed Aboulfadl

TAs:	Eng Mohamed Samir	m.samir@fci-cu.edu.eg
	Eng Omar Khaled Ali Ragab	o.khaled@fci-cu.edu.eg
	Eng Ragia Mohamed	r.mohamed@fci-cu.edu.eg
	Eng Ebtehal yahia	ebtehal.yahia@fci-cu.edu.eg
	Eng Ahmed Emad	ahmed.emad@fci-cu.edu.eg
	Eng Amr Tarek	a.tarek@fci.cu.edu.eg

CS352: Phase 2 – Gamers++ <edugame>

Phase 2- document

Project Team

ID	Name	Email	Mobile
20140066	Alaa Atef Badr	alaa.badr.25@gmail.com	01140911255
20140062	Ismail Ahmed Mahmoud	ismail.ahmed2014@stud.fci-cu.edu.eg	01005887976
20140206	Karim Ehab Ahmed	karimehabahmed@stud.fci-cu.edu.eg	01062064070
20140274	Mostafa Mohamed ElMenshawy	mustafa1elmenshawy@gmail.com	01148504856

Contents

Project Team....2

Restful API....4

User Services....4

Login....4

Register Student....4

Register Teacher....5

Course Services....6

Show Courses....6

Show Teacher Courses....8

Show Student Courses....9

Get Course....10

Add Course....11

Enroll Student....12

Get Unenrolled Courses....12

Game Services....13

Get Game....13

Add game....15

Get Games of Course....16

Level Services....18

Get Level....18

Get Levels 19
Add Level 20
Achievement Services 21
Show Achievements 21
Add achievements 24
Class diagram design 25
Subsystems 25
Achievement 26
Course 27
Game 28
Level 29
User 30
Sequence Diagrams 31
Register 31
Login 31
Play Game 32
Show Courses 33
Create Course 33
Create Game 34
Show Games in specific course 35
Entity Relationship Diagram – ERD 35
Trello workspace screenshot 36
Backend 36
Frontend 36
Integration 37
Github repository link 37

Phase 2- document

Restful API

User Services

Login

Service name	Login		
Input type	POST		
Service URL	/edugame/login		
Parameters	NAME	Description	
	User u	User object containing username and password.	
Example request	POST /edugae/login { “username” : “IsmailAhmed” , “password” : “12345678”		
Example response	{ "name": "Ismail Ahmed", "gender": "Male", "age": 21, "email": "ismail@ismail.edu", "username": "Ismail_Ahmed", "password": "12345678", "identity": "Teacher" }		

Register Student

Service name	Register Student		
Input type	POST		
Service URL	/edugame/register/student		
Parameters	NAME	Description	

	Student s	Student object containing all his/her information.	

Example request	POST /edugame/register/student
	{
	"name": "Alaa Atef",
	"gender": "Female",
	"age": 20,
	"email": "alaa@alaa.com",
	"username": "AlaaAtef",
	"password": "12345678",
	"identity": "Student"
	}		

Example response	{
	"name": "Alaa Atef",
	"gender": "Female",
	"age": 20,
	"email": "alaa@alaa.com",
	"username": "AlaaAtef",
	"password": "12345678",
	"identity": "Student"
	}		

Register Teacher

Service name			
Input type	POST		
Service URL	/edugame/register/teacher		
Parameters	NAME	Description	
	Teacher t	Teacher object containing all	

		his/her information.	

Example request	POST /edugame/register/teacher
	{
	"name": "Alaa Atef",
	"gender": "Female",
	"age": 20,
	"email": "alaa@alaa.edu",
	"username": "AlaaAtef",
	"password": "12345678",
	"identity": "Teacher"
	}		

Example response	{
	"name": "Alaa Atef",
	"gender": "Female",
	"age": 20,
	"email": "alaa@alaa.edu",
	"username": "AlaaAtef",
	"password": "12345678",
	"identity": "Teacher"
	}		

Course Services

Show Courses

Service name	Show Courses	
Input type	GET	
Service URL	/edugame/courses	
Parameters	NAME	Description

Example request	GET /edugame/courses
Example response	[{ "courseName": "C++", "courseOwner": { "name": "Alaa Atef", "gender": "Female", "age": 20, "email": "alaa@alaa.edu", "username": "AlaaAtef", "password": "12345678", "identity": "Teacher" }, "description": "C++ Course for dummies", "students": [] }, { "courseName": "Java", "courseOwner": { "name": "Ismail Ahmed", "gender": "Male", "age": 21, "email": "ismail@ismail.edu", "username": "Ismail_Ahmed", "password": "12345678", "identity": "Teacher" }, "description": "Java Course for dummies", "students": []

	}]

Show Teacher Courses

Service name	Show Teacher Courses
Input type	**GET**
Service URL	**/edugame/Courses/{teacherUsername}**
Parameters	**NAME**: **PathVariable: String teacherUsername** — **Description**: **Teacher username to show his/her courses**
Example request	**GET /edugame/Courses/Ismail_Ahmed**

Example response	[
	{
	"courseName": "Java",
	"courseOwner": {
	"name": "Ismail Ahmed",
	"gender": "Male",
	"age": 21,
	"email": "ismail@ismail.edu",
	"username": "Ismail_Ahmed",
	"password": "12345678",
	"identity": "Teacher"
	},
	"description": "Java Course for dummies",
	"students": []
	}

	]

Show Student Courses

Service name	Show Student Courses
Input type	**GET**
Service URL	**/edugame/enrolledCourses/{studentUsername}**
Parameters	**NAME**: **PathVariable: String username** — **Description**: **Student Username to show courses in which he/she enrolled.**
Example request	**GET /edugame/enrolledCourses/KarimEhab**
Example response	**[{ "courseName": "Java", "courseOwner": { "name": "Ismail Ahmed", "gender": "Male", "age": 21, "email": "ismail@ismail.edu", "username": "Ismail_Ahmed", "password": "12345678", "identity": "Teacher" }, "description": "Java Course for dummies", "students": [{ "name": "Karim Ehab",**

	"gender": "Male", "age": 20, "email": "karim@karim.com", "username": "KarimEhab", "password": "12345678", "identity": "Student" }] }]

Get Course

Service name	Get Course
Input type	GET
Service URL	/edugame/courses/{courseName}
Parameters	NAME: PathVariable: String courseName — Description: Course Name to get
Example request	GET /edugame/courses/Java

Example response	{
	"courseName": "Java",
	"courseOwner": {
	"name": "Ismail Ahmed",
	"gender": "Male",
	"age": 21,
	"email": "ismail@ismail.edu",
	"username": "Ismail_Ahmed",
	"password": "12345678",
	"identity": "Teacher"
	},

	"description": "Java Course for dummies", "students": [] }

Add Course

Service name	Add Course
Input type	POST
Service URL	/edugame/courses/{teacherUsername}/addCourse
Parameters	**NAME** / **Description** PathVariable: String teacherUsername / Teacher username: the owner of the course Cource c / The course to be added

Example request	POST /edugame/coursesIsmail_Ahmed/addCourse
	{
	"name": "Java",
	"description": "Java Course for dummies"
	}

Example response	{
	"courseName": "Java",
	"courseOwner": {
	"name": "Ismail Ahmed",
	"gender": "Male",
	"age": 21,
	"email": "ismail@ismail.edu",
	"username": "Ismail_Ahmed",
	"password": "12345678",
	"identity": "Teacher"

	}, "description": "Java Course for dummies", "students": [] }

Enroll Student

Service name	Enroll Student
Input type	GET
Service URL	/edugame/courses/{courseName}/enroll/{studentUsername}
Parameters	NAME: PathVariable: String courseName — Description: Course Name. NAME: PathVariable: String studentUsername — Description: Student Name.
Example request	GET /edugame/courses/Java/enroll/KarimEhab
Example response	true

Get Unenrolled Courses

Service name	Get unenrolled Courses
Input type	GET
Service URL	/edugame/unenrolledCourses/{studentUsername}
Parameters	NAME: PathVariable: String studentUsername — Description: Student Username to show course in which he/she hasn't

		enrolled yet.
Example request	GET /edugame/unenrolledCourses/KarimEhab	
Example response	{ "courseName": "C++", "courseOwner": { "name": "Alaa Atef", "gender": "Female", "age": 20, "email": "alaa@alaa.edu", "username": "AlaaAtef", "password": "12345678", "identity": "Teacher" }, "description": "C++ Course for dummies", "students": [] }	

Game Services

Get Game

Service name	Get Game		
Input type	GET		
Service URL	/edugame/courses/games/{gameName}		
Parameters	NAME	Description	
	PathVariable String gameName	Game Name that we want to get its data.	
Example request	GET /edugame/courses/games/HelloWorldC++		
Example	{		

response	"name": "HelloWorldC++",
	"description": null,
	"type": "TorF",
	"levels": [
	{
	"levelName": "C++ basic",
	"question": "cout<< prints end line?",
	"number": 1,
	"rightAnswer": "True",
	"answers": [
	"True",
	"False"
	]
	},
	{
	"levelName": "C++ operations",
	"question": "x=5; cout<<x++ returns 6?",
	"number": 2,
	"rightAnswer": "False",
	"answers": [
	"True",
	"False"
	]
	}
	],
	"gameOwner": {
	"name": "Alaa Atef",
	"gender": "Female",
	"age": 20,
	"email": "alaa@alaa.edu",
	"username": "AlaaAtef",

```
            "password": "12345678",
            "identity": "Teacher"
          },
          "levelno": 2,
          "course": {
            "courseName": "C++",
            "courseOwner": {
              "name": "Alaa Atef",
              "gender": "Female",
              "age": 20,
              "email": "alaa@alaa.edu",
              "username": "AlaaAtef",
              "password": "12345678",
              "identity": "Teacher"
            },
            "description": "C++ Course for dummies",
            "students": []
          }
}
```


Add game

Service name	Add Game		
Input type	POST		
Service URL	/edugame/courses/{teacherUsername}/{courseName}/games/addGame		
Parameters	NAME	Description	
	PathVariable String teacherUsername	Username of the teacher (owner) of the game.	
	PathVariable	Course Name	

	String courseName	that we wish to add the game in.	
	Body: Game g	The game itself.	

Example request	POST /edugame/courses/AlaaAtef/C++/games/addGame
	Body:
	{
	"name": "HelloWorldC++",
	"type": "TorF",
	"levelno": 2
	}		

Example response	true		

Get Games of Course

Service name	Get Games of Course	
Input type	GET	
Service URL	/edugame/courses/{courseName}/games	
Parameters	NAME	Description
	PathVariable String courseName	The course to get its games.
Example request	GET /edugame/courses/C++/games	
Example response	[{ "name": "HelloWorldC++", "description": null, "type": "TorF", "levels": [	

```
        {
         "levelName": "C++ basic",
         "question": "cout<< prints end line?",
         "number": 1,
         "rightAnswer": "True",
         "answers": [
          "True",
          "False"
         ]
        },
        {
         "levelName": "C++ operations",
         "question": "x=5; cout<<x++ returns
6?",
         "number": 2,
         "rightAnswer": "False",
         "answers": [
          "True",
          "False"
         ]
        }
       ],
       "gameOwner": {
        "name": "Alaa Atef",
        "gender": "Female",
        "age": 20,
        "email": "alaa@alaa.edu",
        "username": "AlaaAtef",
        "password": "12345678",
        "identity": "Teacher"
       },
```

	"levelno": 2,
	"course": {
	"courseName": "C++",
	"courseOwner": {
	"name": "Alaa Atef",
	"gender": "Female",
	"age": 20,
	"email": "alaa@alaa.edu",
	"username": "AlaaAtef",
	"password": "12345678",
	"identity": "Teacher"
	},
	"description": "C++ Course for dummies",
	"students": []
	}
	}
	]

Level Services

Get Level

Service name	Get level
Input type	GET
Service URL	/edugame/courses/games/{gameName}/levels/{levelno}
Parameters	NAME: PathVariable String gameName — Description: Game Name that the level belongs to. NAME: PathVariable — Description: Level number

	int levelNo	in the game.	
Example request	GET /edugame/courses/games/HelloWorldC++/levels/1		
Example response	{ "levelName": "C++ basic", "question": "cout<< prints end line?", "number": 1, "rightAnswer": "True", "answers": ["True", "False"] }		

Get Levels

Service name	Get levels	
Input type	GET	
Service URL	/edugame/courses/games/{gameName}/levels	
Parameters	NAME	Description
	PathVariable String gameName	Game Name that we want to get its levels.
Example request	GET /edugame/courses/games/HelloWorldC++/levels	
Example response	[{ "levelName": "C++ basic", "question": "cout<< prints end line?", "number": 1, "rightAnswer": "True", "answers": [	

```
            "True",
            "False"
          ]
        },
        {
          "levelName": "C++ operations",
          "question": "x=5; cout<<x++ returns 6?",
          "number": 2,
          "rightAnswer": "False",
          "answers": [
            "True",
            "False"
          ]
        }
]
```


Add Level

Service name	Add level
Input type	POST
Service URL	/edugame/courses/games/{gameName}/levels/addLevel
Parameters	

NAME	Description
PathVariable String gameName	Game Name that we want to add a level to.
Body: Level l	The level to be added.

Example request	POST /edugame/courses/games/HelloWorldC++/levels/addLevel
	Body:
	{
	"levelName": "C++ operations",
	"question": "x=5; cout<<x++ returns 6?",
	"number": 2,
	"rightAnswer": "False",
	"answers":[
	"True",
	"False"
	]
	}

Example response	

Achievement Services

Show Achievements

Service name	Show Achievements
Input type	GET
Service URL	/edugame/achievements/{username}
Parameters	NAME: PathVariable String username Description: Username of the user that we want to get his/her achievements.
Example request	/edugame/achievements/KarimEhab
Example	[

response	{
	"score": 100,
	"student": {
	"name": "Karim Ehab",
	"gender": "Male",
	"age": 20,
	"email": "karim@karim.com",
	"username": "KarimEhab",
	"password": "12345678",
	"identity": "Student"
	},
	"gameinAch": {
	"name": "Ta7meer",
	"description": "2atta3 swabe3 aw chipsy",
	"type": "true-false",
	"levels": [
	{
	"levelName": "chipsy",
	"question": "yo7amar fi zobda?",
	"number": 2,
	"rightAnswer": "False",
	"answers": [
	"True",
	"False"
	]
	},
	{
	"levelName": "swabe3",
	"question": "tonqa3 elswabe3 fi ma2 w mal7?",

```
                "number": 1,
                "rightAnswer": "True",
                "answers": [
                 "True",
                 "False"
                ]
               }
              ],
              "gameOwner": {
               "name": "Alaa Atef",
               "gender": "Female",
               "age": 20,
               "email": "alaa@alaa.edu",
               "username": "AlaaAtef",
               "password": "12345678",
               "identity": "Teacher"
              },
              "levelno": 2,
              "course": {
               "courseName": "Btates",
               "courseOwner": {
                "name": "Alaa Atef",
                "gender": "Female",
                "age": 20,
                "email": "alaa@alaa.edu",
                "username": "AlaaAtef",
                "password": "12345678",
                "identity": "Teacher"
               },
               "description": "kayf t7amar elbtates b
          tareeqa e7trafya",
```

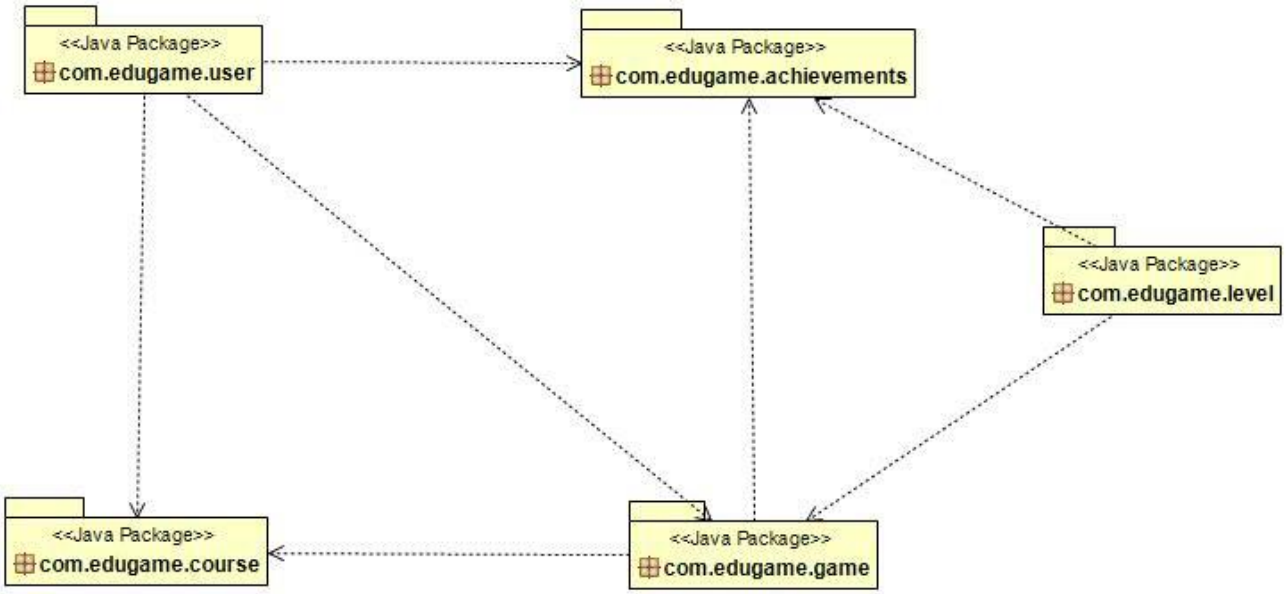
	"students": [] } } }]

Add achievements

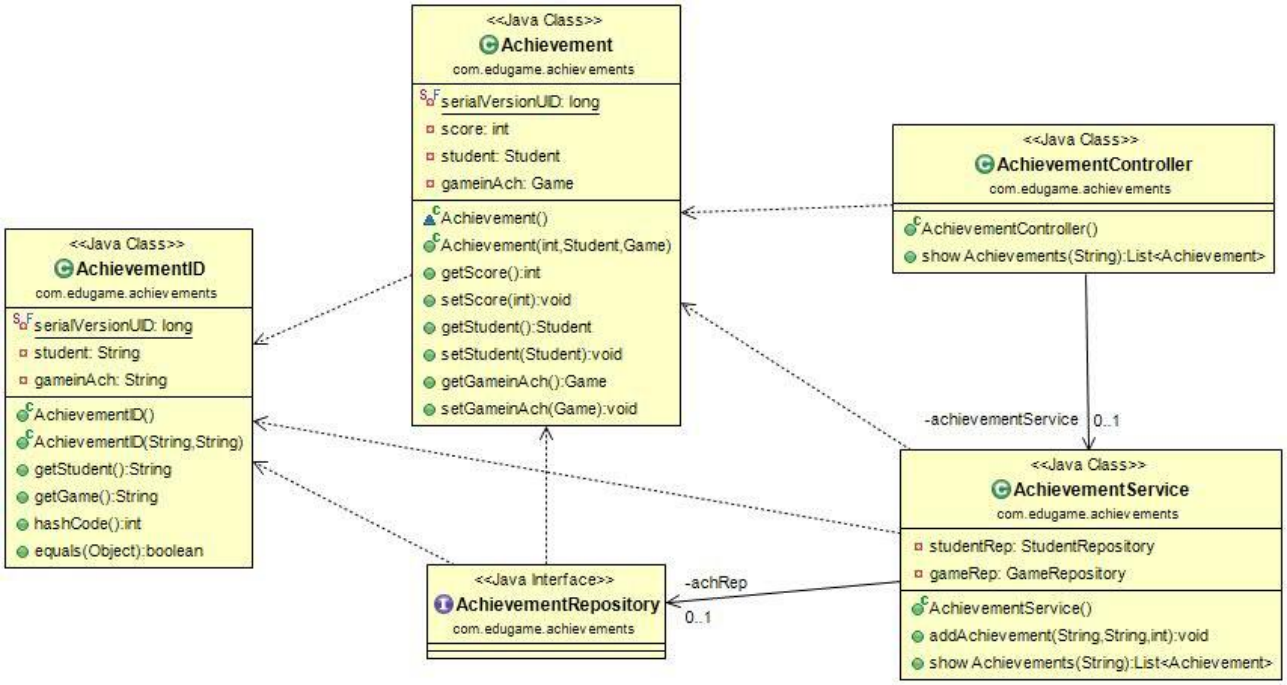
Service name	Add Achievement
Input type	GET
Service URL	/edugame/{username}/courses/games/{gameName}/levels/{levelno}
Parameters	NAME: Description PathVariable String gameName: Game Name that the level belongs to. PathVariable int levelNo: Level number in the game. PathVariable String username: Username of the student playing the game.
Example request	GET /edugame/ElMensh/courses/games/HelloWorldC++/levels/1
Example response	

Class diagram design

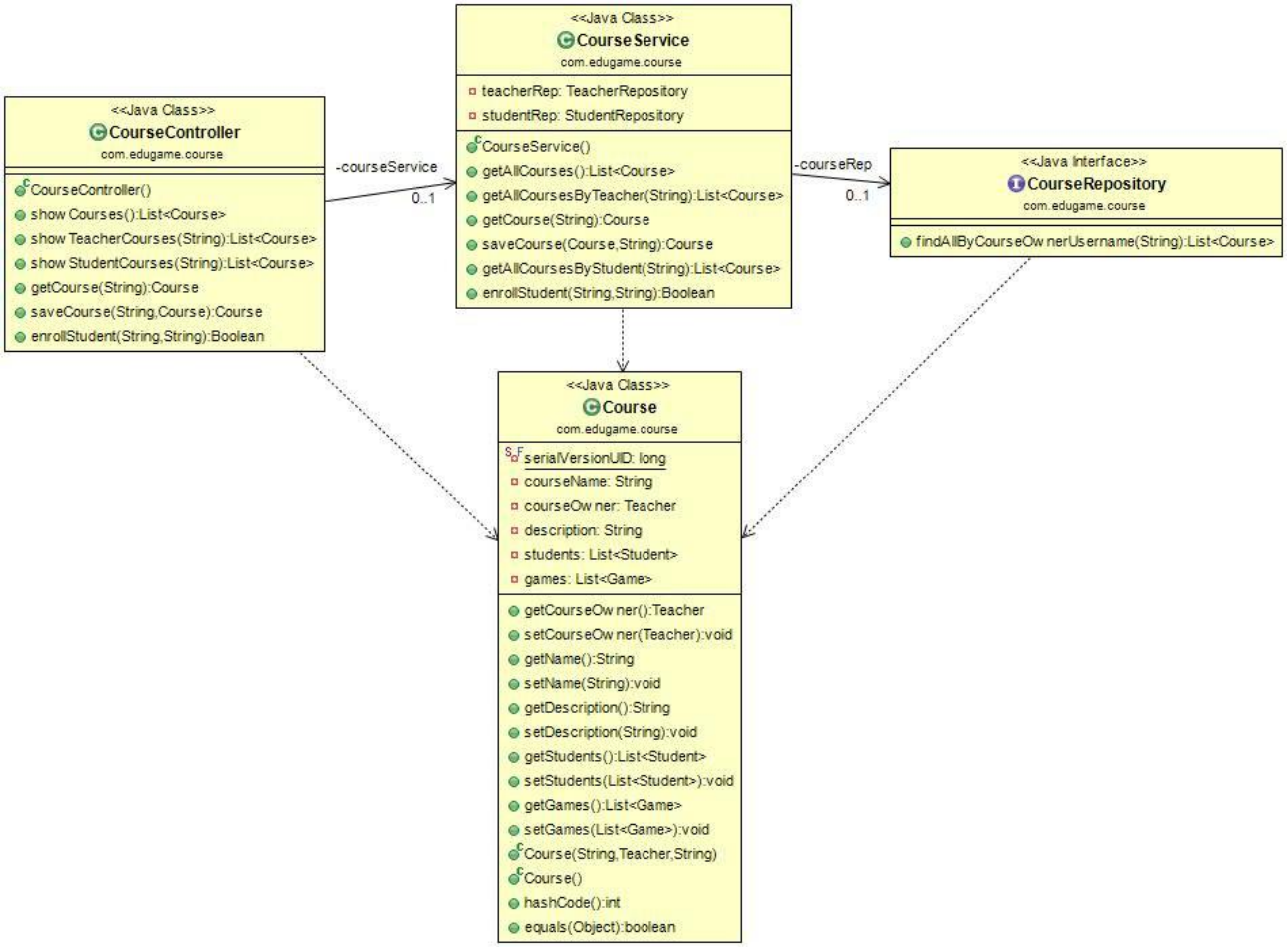
Subsystems



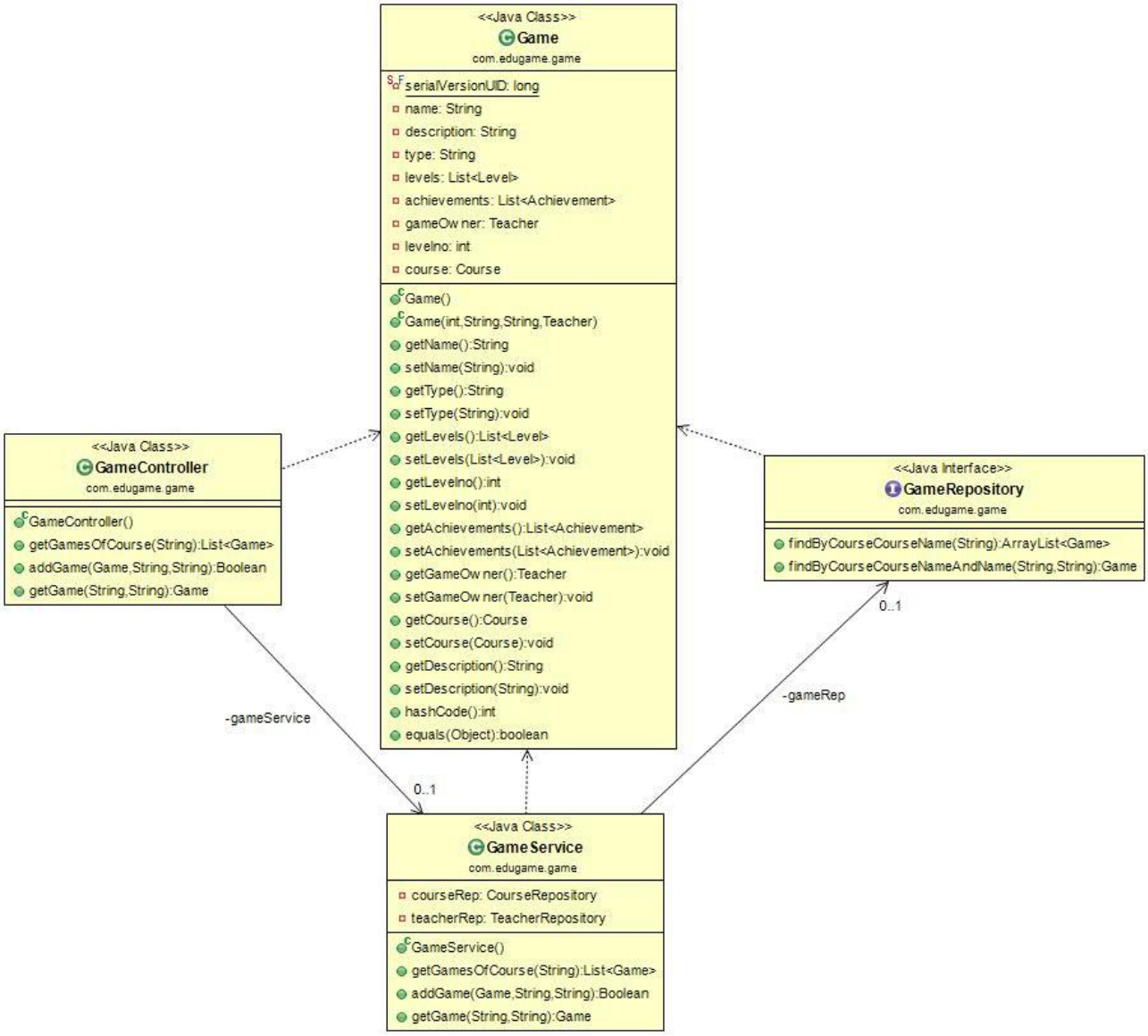
Achievement



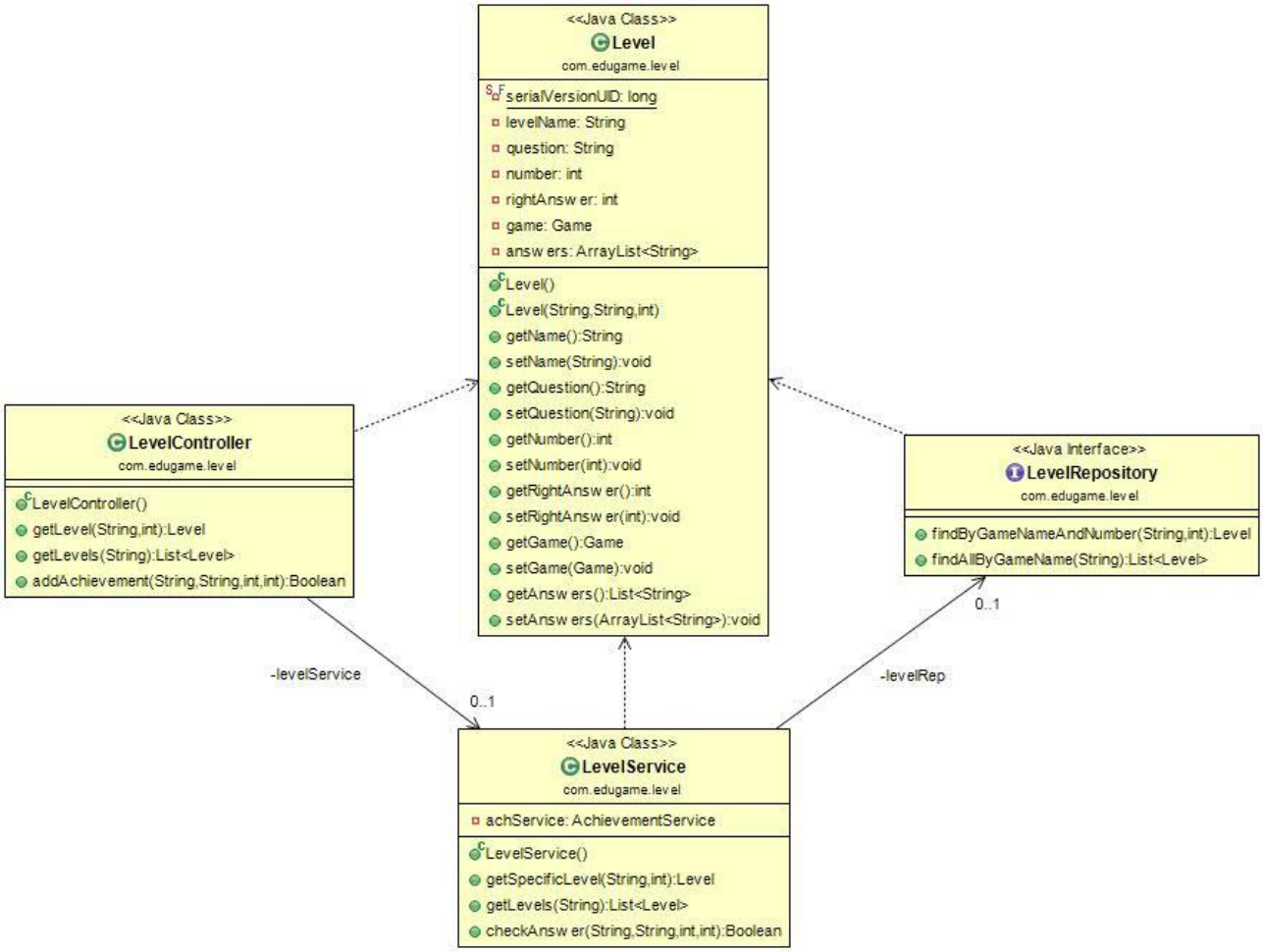
Course


<<Java Class>>
CourseController
com.edugame.course
CourseController()
show Courses():List<Course>
show TeacherCourses(String):List<Course>
show StudentCourses(String):List<Course>
getCourse(String):Course
saveCourse(String,Course):Course
enrollStudent(String,String):Boolean
-courseService
0..1
<<Java Class>>
CourseService
com.edugame.course
teacherRep: TeacherRepository
studentRep: StudentRepository
CourseService()
getAllCourses():List<Course>
getAllCoursesByTeacher(String):List<Course>
getCourse(String):Course
saveCourse(Course,String):Course
getAllCoursesByStudent(String):List<Course>
enrollStudent(String,String):Boolean
-courseRep
0..1
<<Java Interface>>
CourseRepository
com.edugame.course
findAllByCourseOwnerUsername(String):List<Course>
<<Java Class>>
Course
com.edugame.course
serialVersionUID: long
courseName: String
courseOwner: Teacher
description: String
students: List<Student>
games: List<Game>
getCourseOwner():Teacher
setCourseOwner(Teacher):void
getName():String
setName(String):void
getDescription():String
setDescription(String):void
getStudents():List<Student>
setStudents(List<Student>):void
getGames():List<Game>
setGames(List<Game>):void
Course(String,Teacher,String)
Course()
hashCode():int
equals(Object):boolean

Game


<<Java Class>>
Game
com.edugame.game
serialVersionUID: long
name: String
description: String
type: String
levels: List<Level>
achievements: List<Achievement>
gameOwner: Teacher
levelno: int
course: Course
Game()
Game(int,String,String,Teacher)
getName():String
setName(String):void
getType():String
setType(String):void
getLevels():List<Level>
setLevels(List<Level>):void
getLevelno():int
setLevelno(int):void
getAchievements():List<Achievement>
setAchievements(List<Achievement>):void
getGameOwner():Teacher
setGameOwner(Teacher):void
getCourse():Course
setCourse(Course):void
getDescription():String
setDescription(String):void
hashCode():int
equals(Object):boolean
<<Java Class>>
GameController
com.edugame.game
GameController()
getGamesOfCourse(String):List<Game>
addGame(Game,String,String):Boolean
getGame(String,String):Game
<<Java Interface>>
GameRepository
com.edugame.game
findByCourseCourseName(String):ArrayList<Game>
findByCourseCourseNameAndName(String,String):Game
-gameService
0..1
0..1
-gameRep
<<Java Class>>
GameService
com.edugame.game
courseRep: CourseRepository
teacherRep: TeacherRepository
GameService()
getGamesOfCourse(String):List<Game>
addGame(Game,String,String):Boolean
getGame(String,String):Game

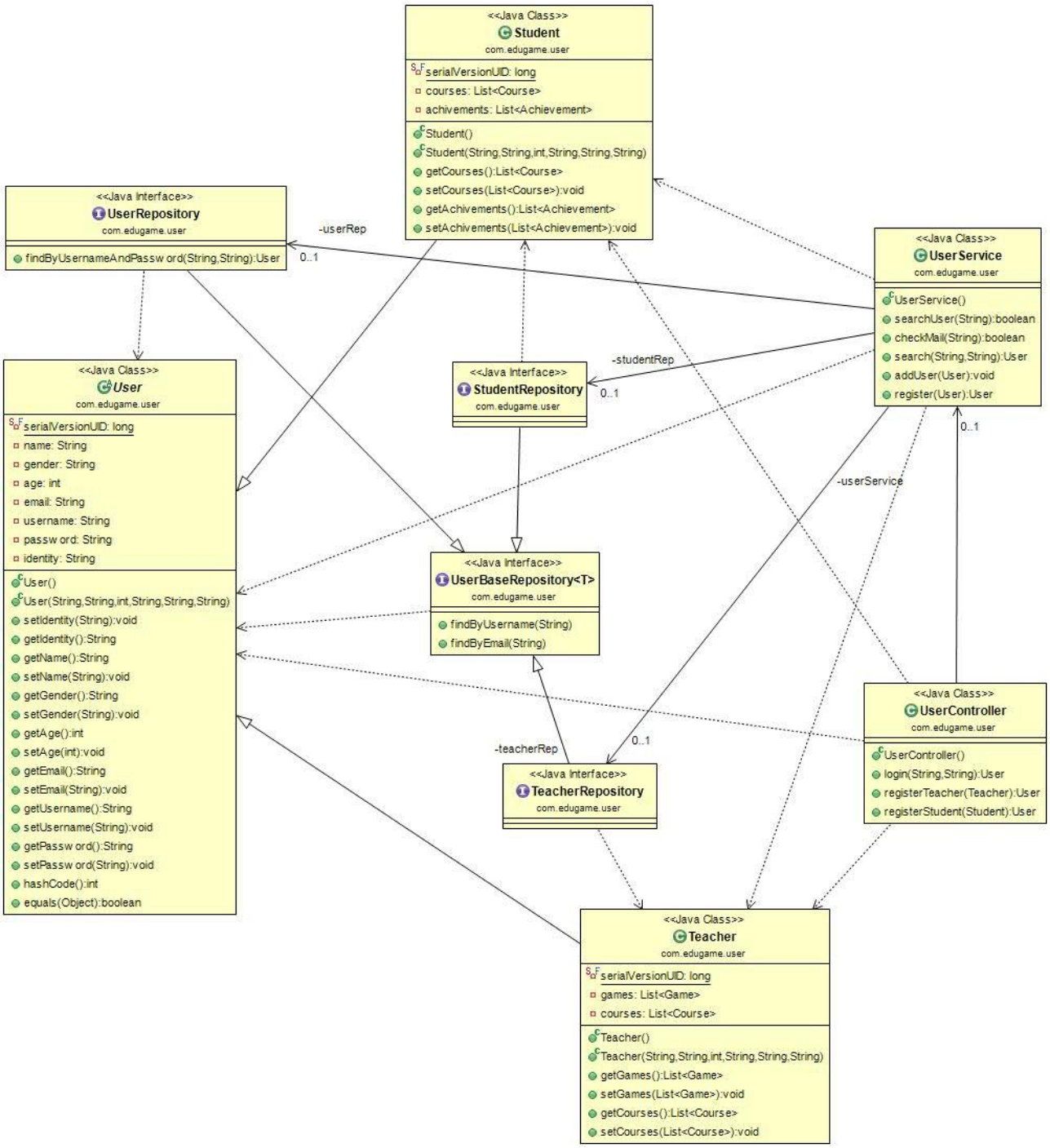
Level


<<Java Class>>
Level
com.edugame.level
serialVersionUID: long
levelName: String
question: String
number: int
rightAnswer: int
game: Game
answers: ArrayList<String>
Level()
Level(String,String,int)
getName():String
setName(String):void
getQuestion():String
setQuestion(String):void
getNumber():int
setNumber(int):void
getRightAnswer():int
setRightAnswer(int):void
getGame():Game
setGame(Game):void
getAnswers():List<String>
setAnswers(ArrayList<String>):void
<<Java Class>>
LevelController
com.edugame.level
LevelController()
getLevel(String,int):Level
getLevels(String):List<Level>
addAchievement(String,String,int,int):Boolean
<<Java Interface>>
LevelRepository
com.edugame.level
findByGameNameAndNumber(String,int):Level
findAllByGameName(String):List<Level>
-levelService
0..1
0..1
-levelRep
<<Java Class>>
LevelService
com.edugame.level
achService: AchievementService
LevelService()
getSpecificLevel(String,int):Level
getLevels(String):List<Level>
checkAnswer(String,String,int,int):Boolean

CS352: Phase 2 – Gamers++ <edugame>

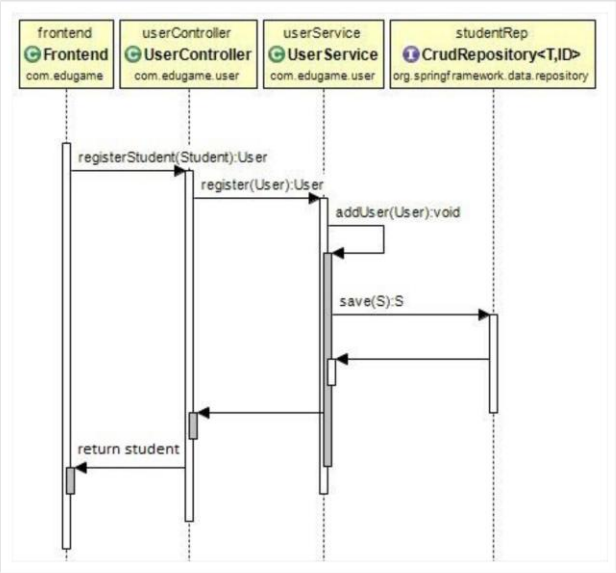
Phase 2- document

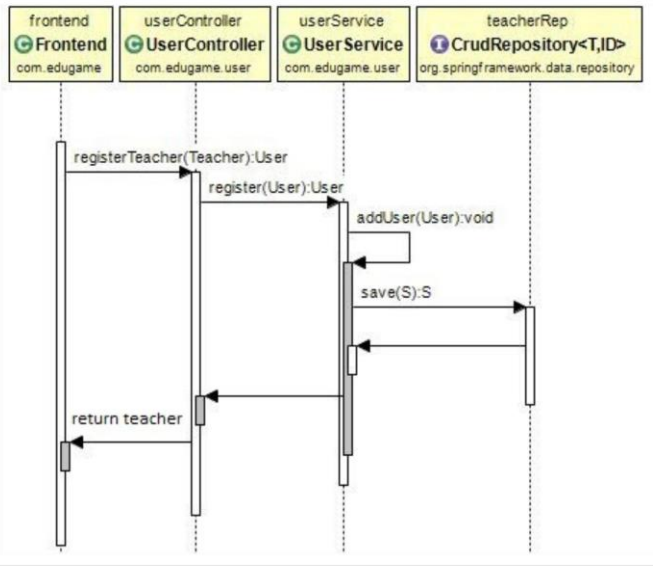
User



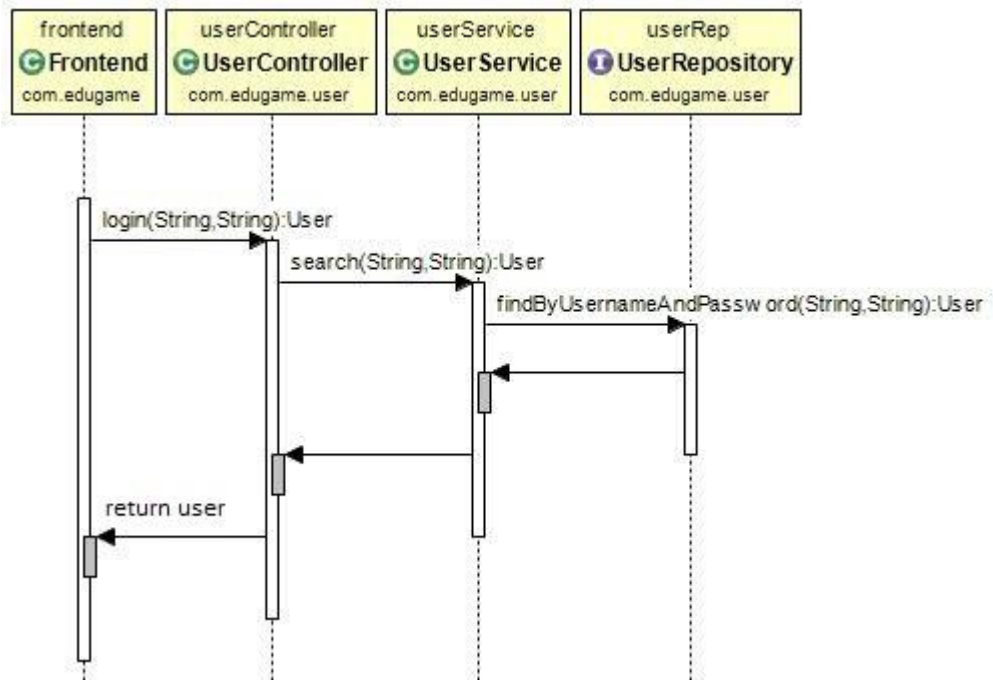
Sequence Diagrams

Register

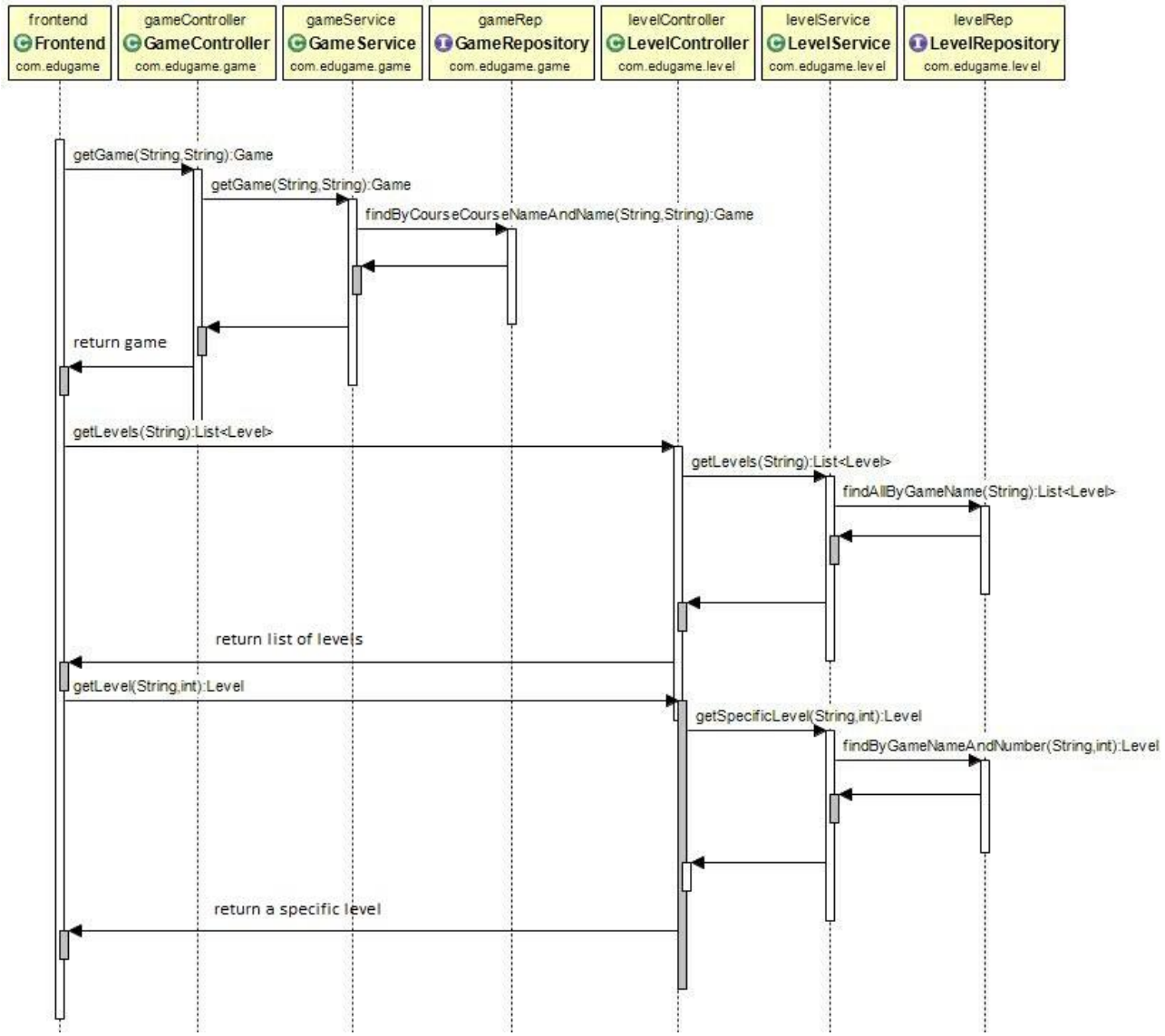




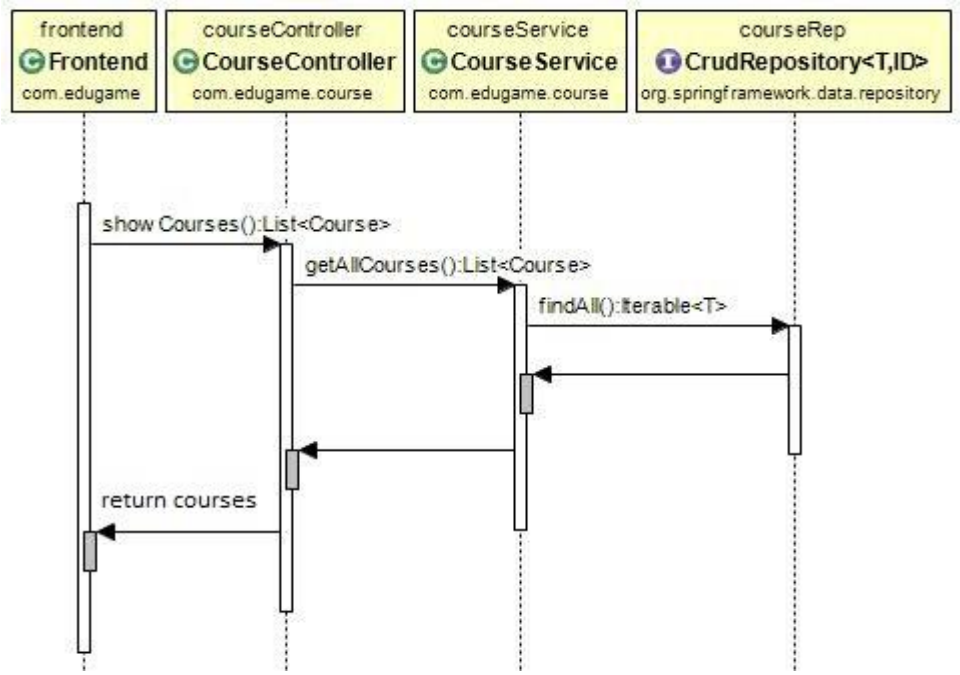
Login



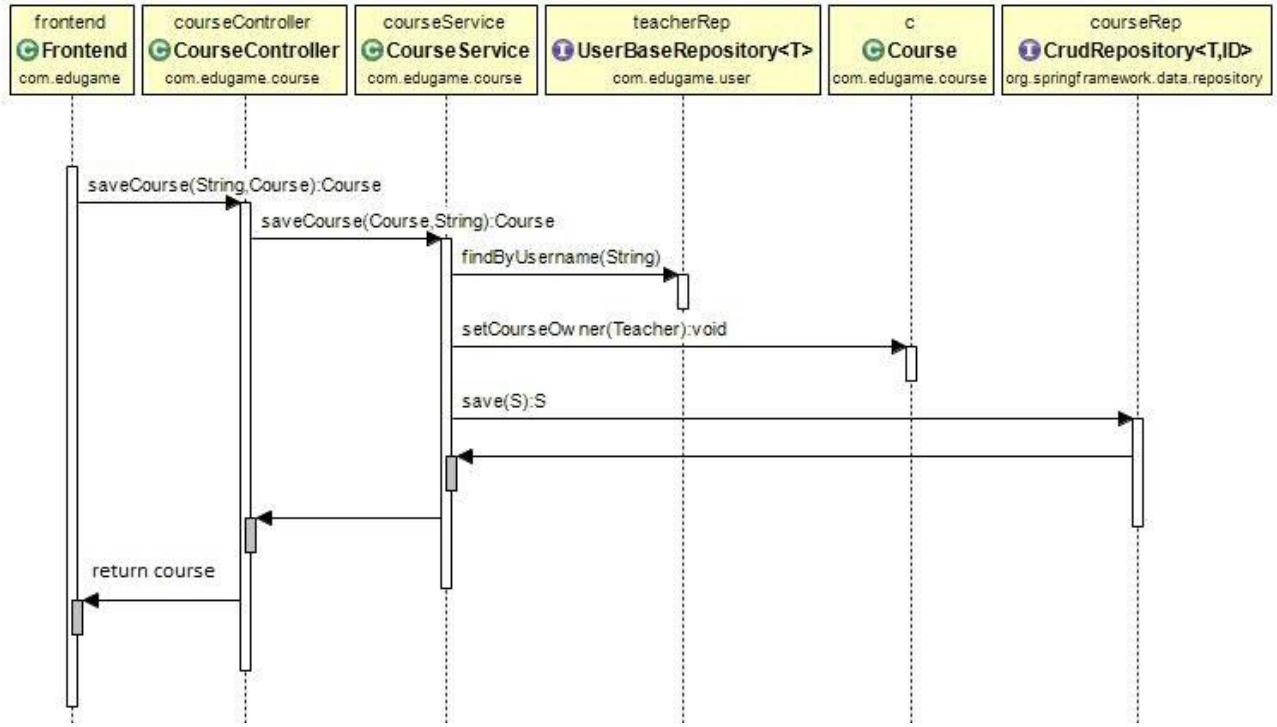
Play Game


frontend
Frontend
com.edugame
gameController
GameController
com.edugame.game
gameService
GameService
com.edugame.game
gameRep
GameRepository
com.edugame.game
levelController
LevelController
com.edugame.level
levelService
LevelService
com.edugame.level
levelRep
LevelRepository
com.edugame.level
getGame(String,String):Game
getGame(String,String):Game
findByCourseCourseNameAndName(String,String):Game
return game
getLevels(String):List<Level>
getLevels(String):List<Level>
findAllByGameName(String):List<Level>
return list of levels
getLevel(String,int):Level
getSpecificLevel(String,int):Level
findByGameNameAndNumber(String,int):Level
return a specific level

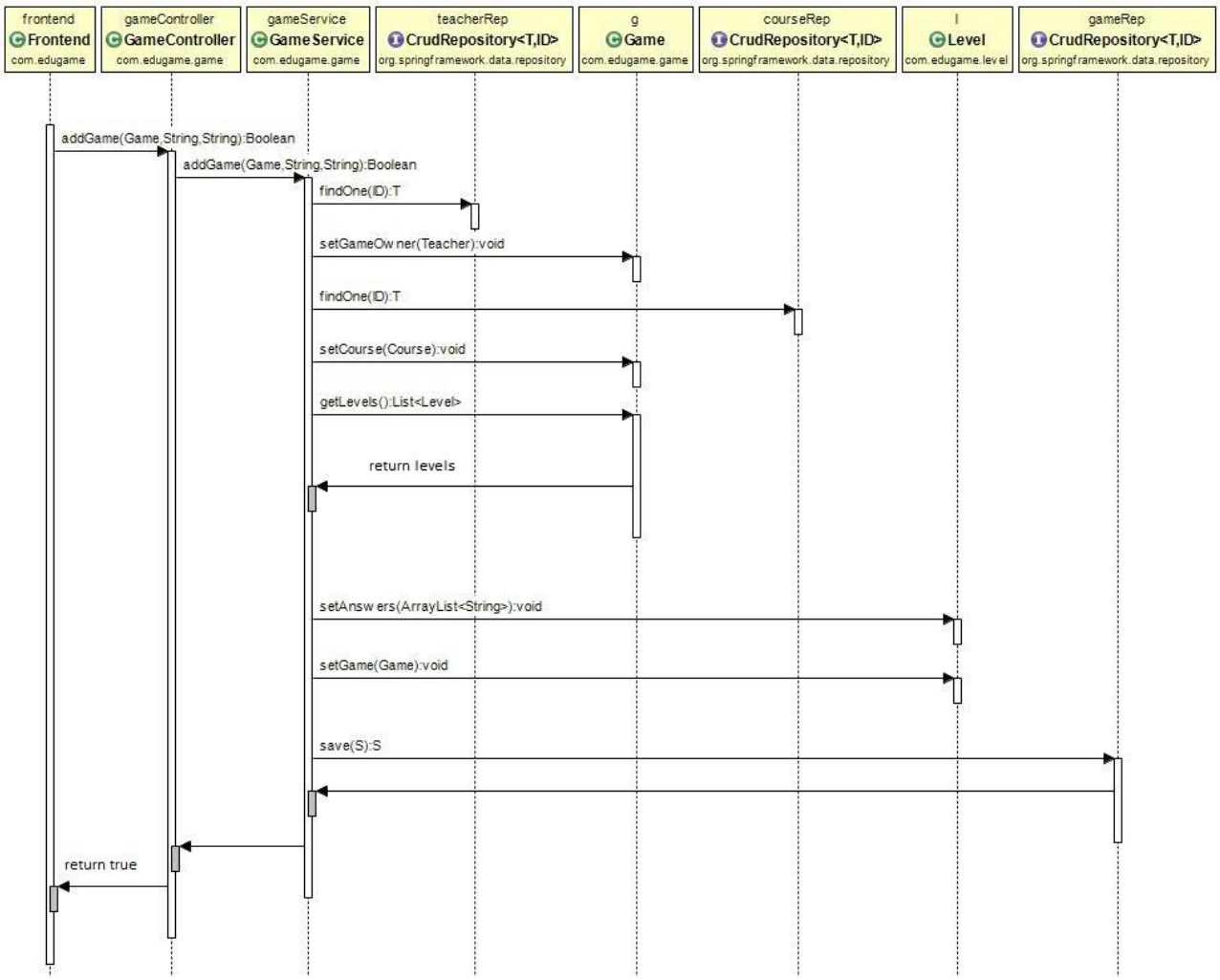
Show Courses



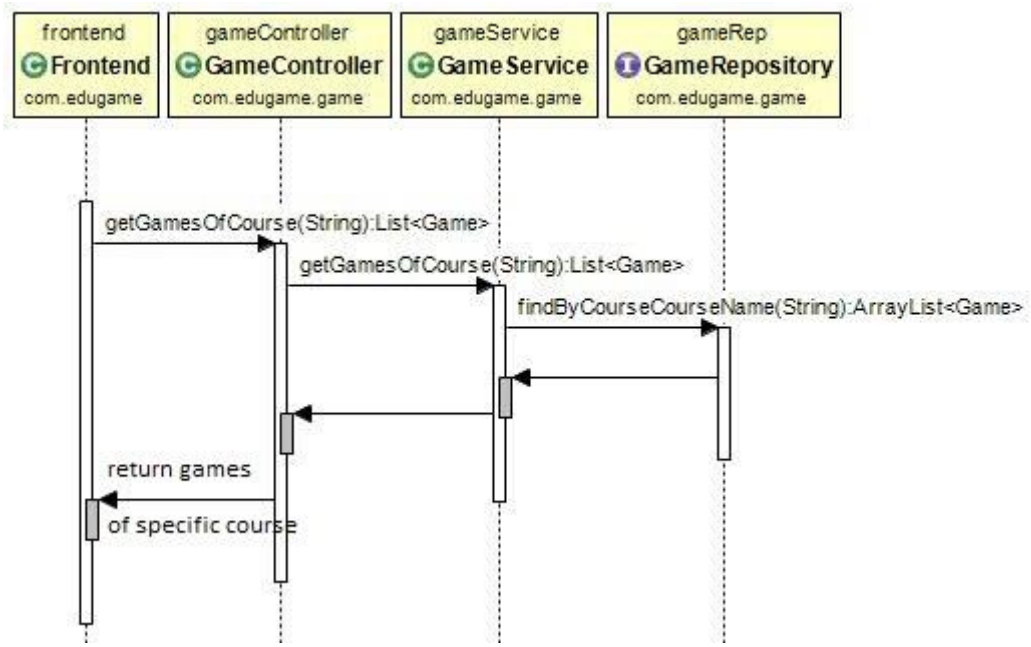
Create Course



Create Game


frontend
Frontend
com.edugame
gameController
GameController
com.edugame.game
gameService
Game Service
com.edugame.game
teacherRep
CrudRepository<T,ID>
org.springframework.data.repository
g
Game
com.edugame.game
courseRep
CrudRepository<T,ID>
org.springframework.data.repository
l
Level
com.edugame.level
gameRep
CrudRepository<T,ID>
org.springframework.data.repository
addGame(Game,String,String):Boolean
addGame(Game,String,String):Boolean
findOne(ID):T
setGameOwner(Teacher):void
findOne(ID):T
setCourse(Course):void
getLevels():List<Level>
return levels
setAnswers(ArrayList<String>):void
setGame(Game):void
save(S):S
return true

Show Games in specific course



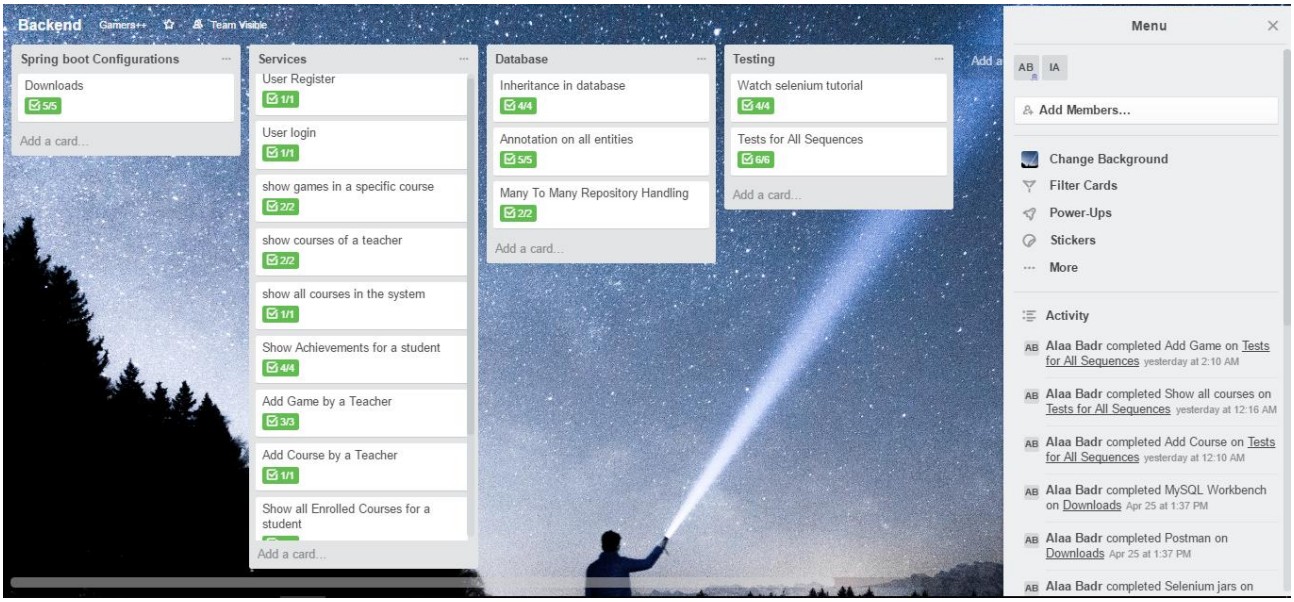
Entity Relationship Diagram – ERD

CS352: Phase 2 – Gamers++ <edugame>

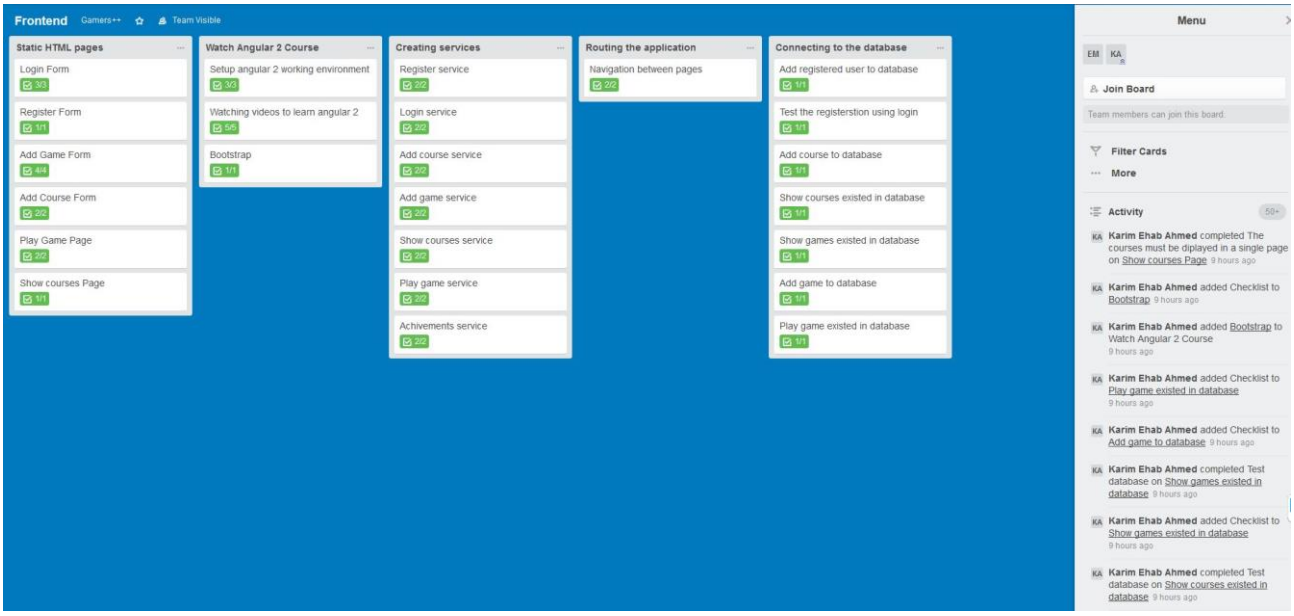
Phase 2- document

Trello workspace screenshot

Backend

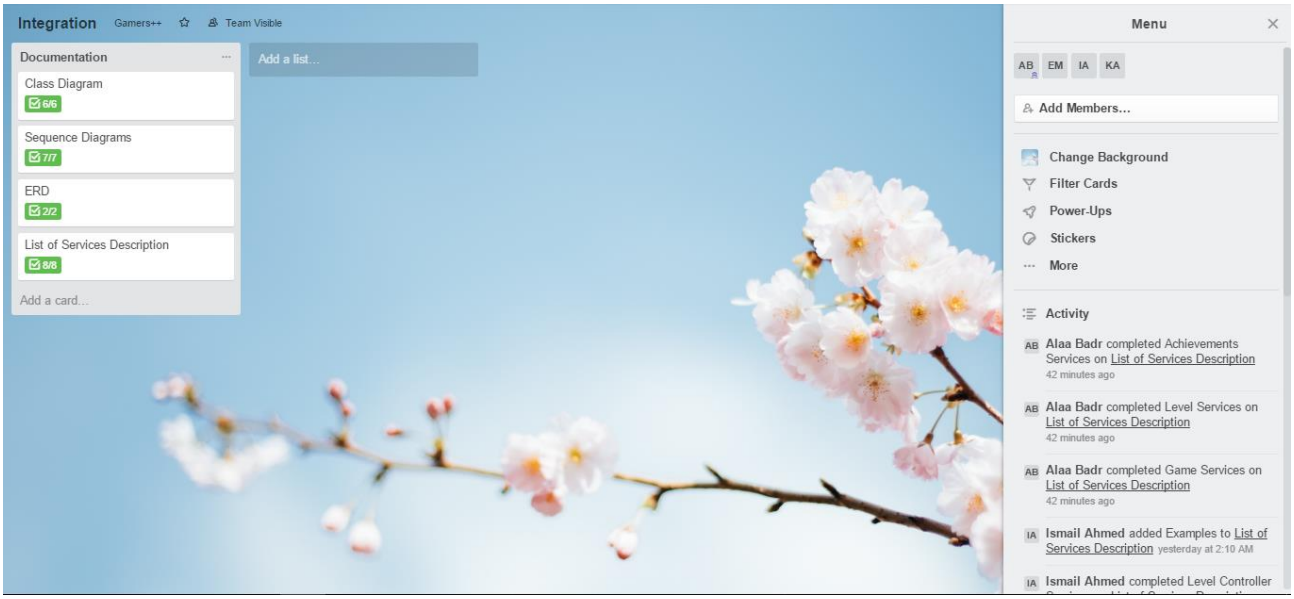


Frontend



Phase 2- document

Integration



Github repository link

https://github.com/AlaaBadr/SWE2-Project