Verkefni 2 Team 13

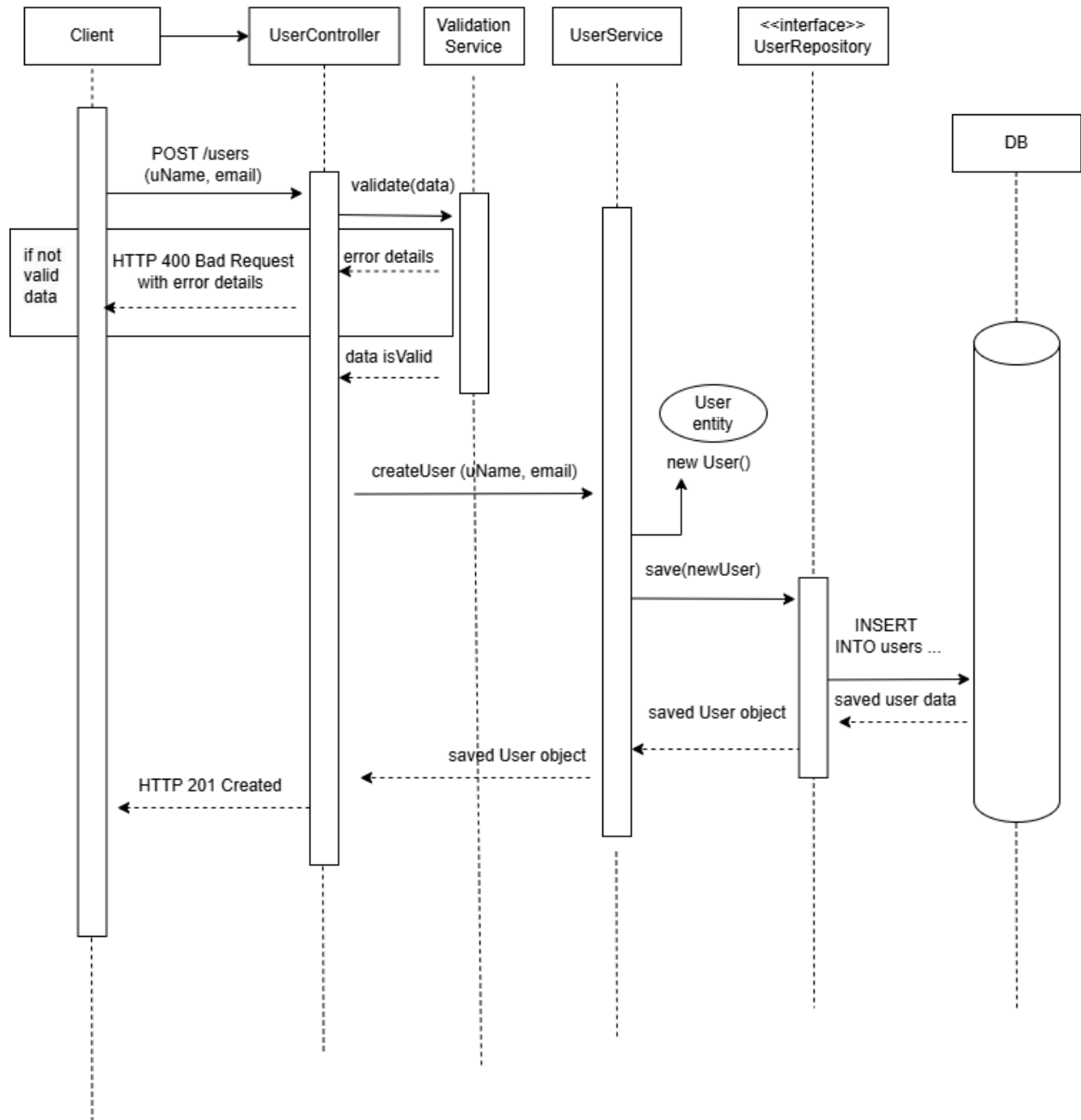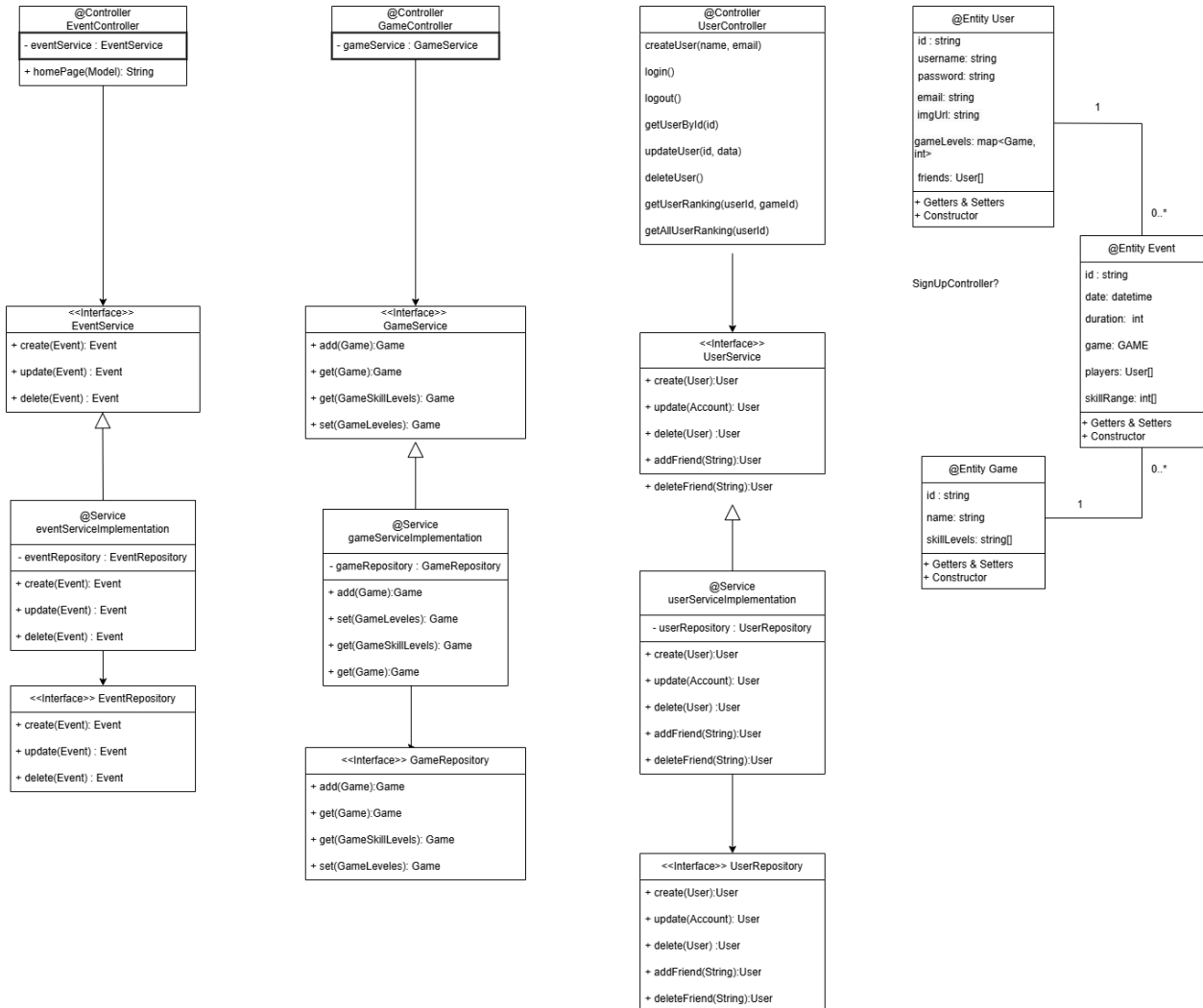Github: https://github.com/KristinnRoach/hugbo_team_13

Sequence Diagram:

A new user of the app presses the Create Account button on the Account Creation page after filling in all required fields

Class diagram:

UML class diagram

@Controller
EventController
- eventService : EventService

+ homePage(Model): String

@Controller
GameController
- gameService : GameService

@Controller
UserController

createUser(name, email)

login()

logout()

getUserById(id)

updateUser(id, data)

deleteUser()

getUserRanking(userId, gameId)

getAllUserRanking(userId)

@Entity User

id : string

username: string

password: string

email: string

imgUrl: string

gameLevels: map<Game, int>

friends: User[]

+ Getters & Setters
+ Constructor

SignUpController?

@Entity Event

id : string

date: datetime

duration: int

game: GAME

players: User[]

skillRange: int[]

+ Getters & Setters
+ Constructor

<<Interface>>
EventService
+ create(Event): Event

+ update(Event) : Event

+ delete(Event) : Event

<<Interface>>
GameService
+ add(Game):Game

+ get(Game):Game

+ get(GameSkillLevels): Game

+ set(GameLeveles): Game

<<Interface>>
UserService

+ create(User):User

+ update(Account): User

+ delete(User) :User

+ addFriend(String):User

+ deleteFriend(String):User

@Entity Game

id : string

name: string

skillLevels: string[]

+ Getters & Setters
+ Constructor

@Service
eventServiceImplementation
- eventRepository : EventRepository

+ create(Event): Event

+ update(Event) : Event

+ delete(Event) : Event

@Service
gameServiceImplementation
- gameRepository : GameRepository

+ add(Game):Game

+ set(GameLeveles): Game

+ get(GameSkillLevels): Game

+ get(Game):Game

@Service
userServiceImplementation
- userRepository : UserRepository

+ create(User):User

+ update(Account): User

+ delete(User) :User

+ addFriend(String):User

+ deleteFriend(String):User

<<Interface>> EventRepository
+ create(Event): Event

+ update(Event) : Event

+ delete(Event) : Event

<<Interface>> GameRepository
+ add(Game):Game

+ get(Game):Game

+ get(GameSkillLevels): Game

+ set(GameLeveles): Game

<<Interface>> UserRepository
+ create(User):User

+ update(Account): User

+ delete(User) :User

+ addFriend(String):User

+ deleteFriend(String):User

1

0..*

0..*

1

State machine Diagram:

Initial pseudo state

Front page

Enters Account info
and clicks login button

Account
creation
fail -
Insufficient
info or
already
exists

Account creation
page

Check credentials

Incorrect
login info /
clear
password
field

Successful
Creation

Successful
login

User Home page
(Always accessible)

Logout

Events page

Friends page

Add, delete, browse friends

Search Events

Create a
new event

Success
or Fail

Event search page

Create Event page

Search
criteria
(date,
game, skill
level)

Go back
to events
page

List of events that
match search criteria

select
event from
list

set
attending

set not
attending