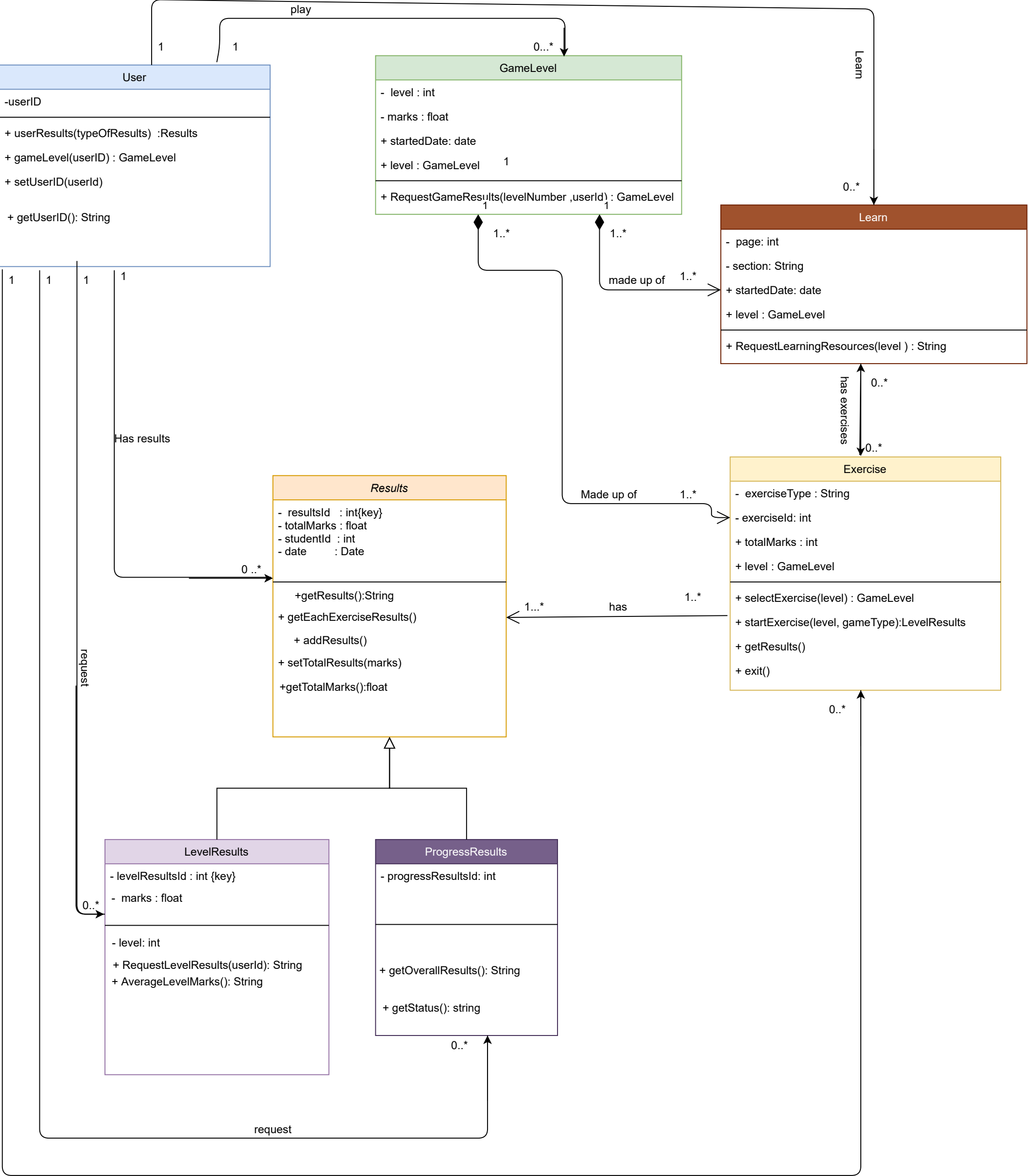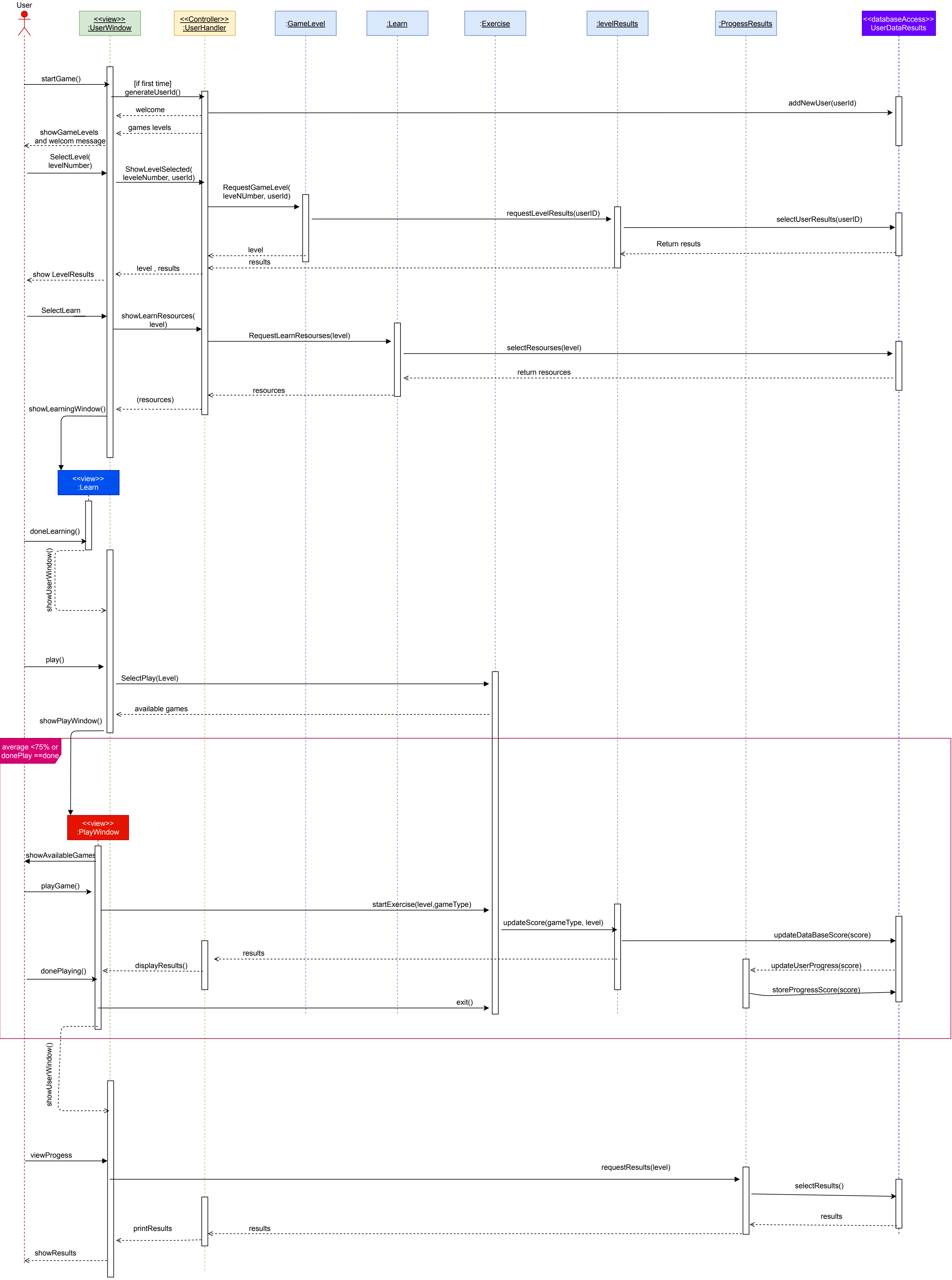Wgame(Word Game)

Actor: User(Language learner)

The user opens the application for learning the language being though by the application. The system welcome the user, shows the list of levels that are playable in the game and describes what the user can expect in each level listed. The system give user recommendation on which level they should start in.

The user select the level(of leaning and playing based on level description of each level listed) they are suitable with. The system present the user with give user choice of playing or learning base on the level user has chose. If user selects to learn the system will give it the resource material needed to play level that was chosen. If the user select playing they will be presented with different kinds of games which test the content assigned to each level, user may select a game they want to play from the ones listed by the system for the level being played. System show user results and check if threshold results required for going next level are met by the use ,then user move to another level of learning and playing when they score above level threshold.
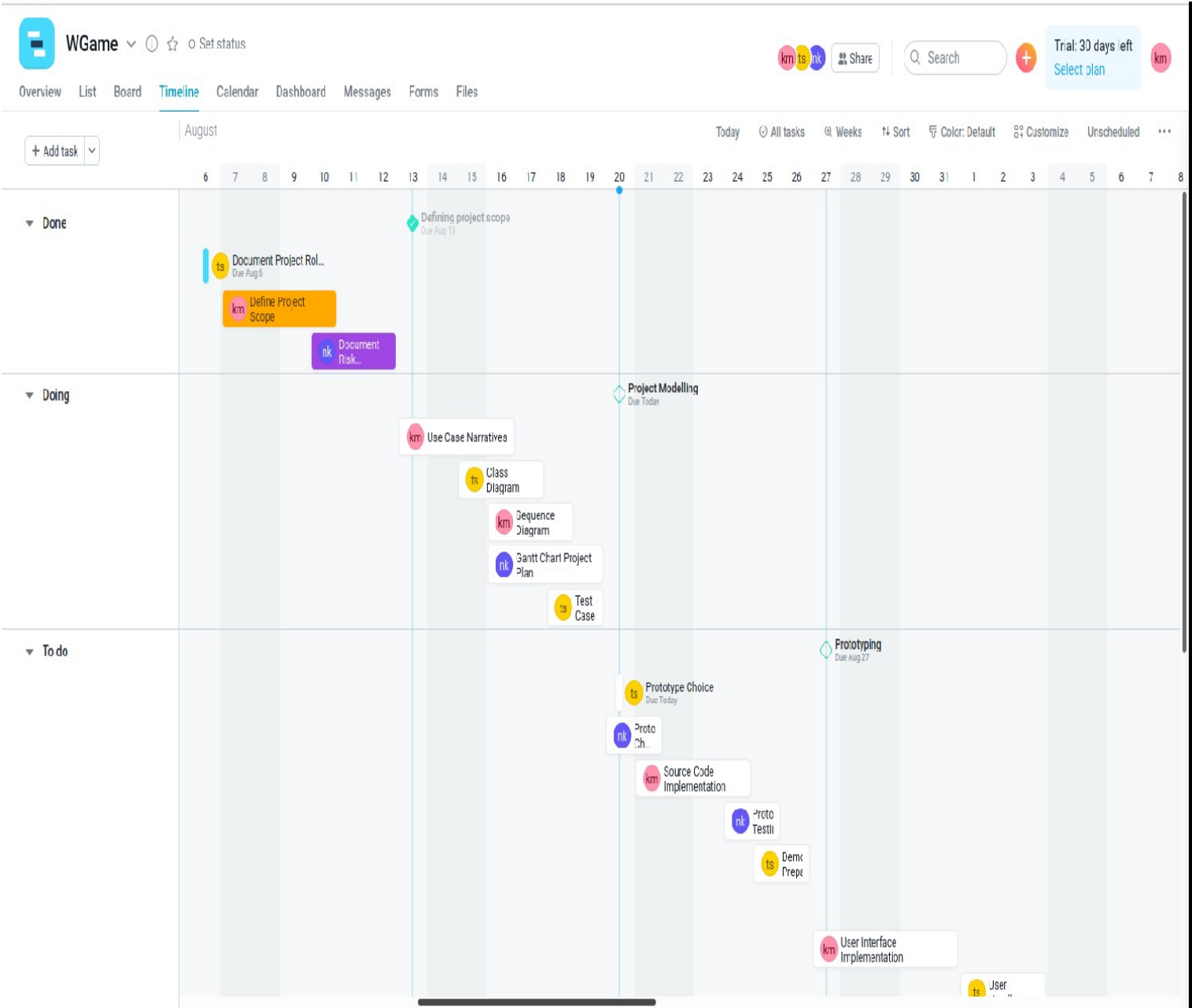
The user check learning progress after finishing each game. The system will give the user overall progress in each level and the high scores that the user has achieved, it also update progress as user keep advancing in the game.

play

User

-userID

+ userResults(typeOfResults) :Results
+ gameLevel(userID) : GameLevel
+ setUserID(userId)

+ getUserID(): String

GameLevel

- level : int
- marks : float
+ startedDate: date
+ level : GameLevel

+ RequestGameResults(levelNumber ,userId) : GameLevel

Learn

- page: int
- section: String
+ startedDate: date
+ level : GameLevel

+ RequestLearningResources(level) : String

Results

- resultsId   : int{key}
- totalMarks : float
- studentId : int
- date   : Date

+getResults():String
+ getEachExerciseResults()
+ addResults()
+ setTotalResults(marks)
+getTotalMarks():float

Exercise

- exerciseType : String
- exerciseId: int
+ totalMarks : int
+ level : GameLevel

+ selectExercise(level) : GameLevel
+ startExercise(level, gameType):LevelResults
+ getResults()
+ exit()

LevelResults

- levelResultsId : int {key}
- marks : float

- level: int
+ RequestLevelResults(userId): String
+ AverageLevelMarks(): String

ProgressResults

- progressResultsId: int

+ getOverallResults(): String

+ getStatus(): string

made up of

Has results

request

request

Learn

has exercises

has

Made up of

1..*   1..*   0...*   0..*   0 ..*   1...*   1..*   0..*   0..*   1..*   1   1   1   1   1   1   1

User

<<view>>
:UserWindow

<<Controller>>
:UserHandler

:GameLevel

:Learn

Exercise

:levelResults

:ProgessResults

<<databaseAccess>>
UserDataResults

startGame()

[if first time]
generateUserId()

addNewUser(userId)

welcome

showGameLevels
and welcom message

games levels

SelectLevel(
levelNumber)

ShowLevelSelected(
leveleNumber, userId)

RequestGameLevel(
leveNUmber, userId)

requestLevelResults(userID)

selectUserResults(userID)

Return resuts

level

results

level , results

show LevelResults

SelectLearn

showLearnResources(
level)

RequestLearnResourses(level)

selectResourses(level)

return resources

resources

(resources)

showLearningWindow()

<<view>>
:Learn

doneLearning()

showUuwUserWindow()

play()

SelectPlay(Level)

available games

showPlayWindow()

average <75% or
donePlay ==done

<<view>>
:PlayWindow

showAvailableGames

playGame()

startExercise(level,gameType)

updateScore(gameType, level)

updateDataBaseScore(score)

results

displayResults()

updateUserProgress(score)

donePlaying()

storeProgressScore(score)

exit()

showUserWindow()

viewProgess

requestResults(level)

selectResults()

results
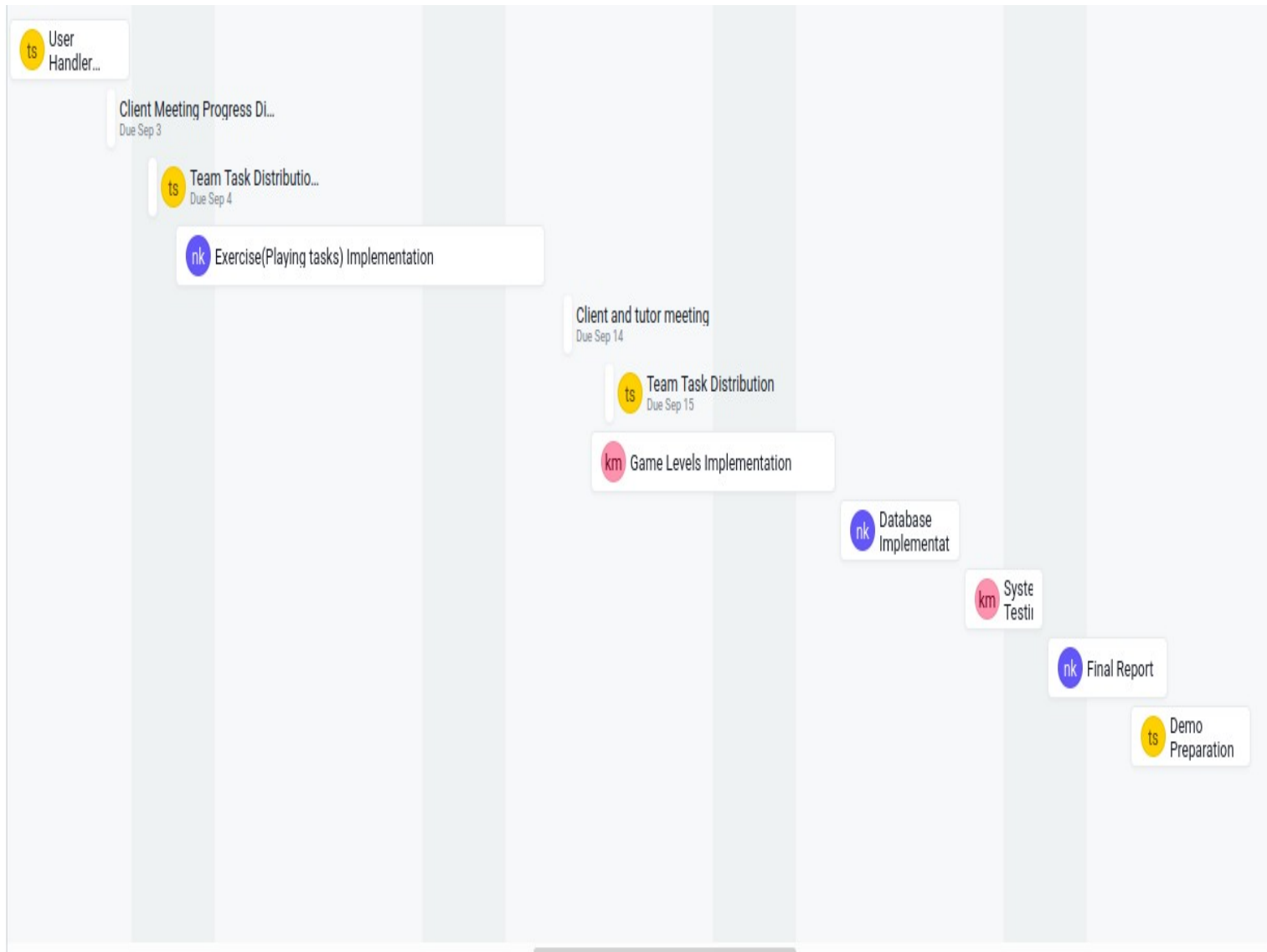
printResults

results

showResults

Gantt chart

Part 1

1 september -> 1 october

Link to original Gantt chart

Project Abbr &Name	WGAME, Word Game	Client/Supervisor +email	Dr Maria Keet maria.keet@uct.ac.za
Date	20 August 2021	Tutor +email	Yash Ramsamy RMSYAS003@myuct.ac.za
Team Members	Lunga Tsewu	Sipho Nkele	Kuhle Mxotwa

Test case number	inputs, behavior and expected outcome
1. User chooses level	• If chosen incorrect level of play • System displays a massage to the user on where to start
2. System checks user Score for level	• If user score meets threshold score • System should allow user to progress to next level
3. User answers during play(exercise)	• System correctly records the user inputs (answers) and progress trackers (average scores) are updated from the correct answers. • System must record answers and progress trackers update to the reflect the correct results.
4. check progress	• System to show user recently updated progress report • Progress report
5. Hit button or hovering over text edit	• System solves portion of a problem for each request hint or show toast that describes the problem and gives the percentage of the solved part of the problem. • Toast with description, solves portion of the problem or gives percentage progress bar of the solved problem portion.