

WORD MASTER 3000

Mastery Over English Words

Word Master 3000 is a mobile based application which aims to help/guide the user achieve mastery over English words. The app helps users learn spelling, word meaning, multiple meanings of words, usage of words in sentences, synonyms, and antonyms. This can be achieved using standard memorization techniques, short training sessions, games and quizzes. The users of our application are people of all ages.

The app will have the following features:

Authentication

Create account: To access all features, the user must create an account. The user needs to provide his/her name, email, unique username, password.

Log-in: The user can log-in using his email and password.

Recover account: If the user forgets his password, it can be recovered by sending a recovery link to his email.

Dictionary

There will be an offline dictionary for the users. It will show word meaning, sub meaning of words, parts of speech, pronunciation, idioms and phrases, and example sentences. There will be a check box with every meaning that the word has. The user can select the specific meanings that interest him. He / she can also select all the meanings. The selection can be saved, which the application will help the user to memorize.

Search

The user will be able to search for different words. To make the process easy and efficient there will be two different ways of searching:

Input in app: The user can directly type or paste any word in the application to search for it. It is basically manual input direct in the app.

Search on copy: The user copies the word in his clipboard from any app in his phone. A floating window will give him an option to search for the word in our app.

Storing

Create, edit, and delete word lists: The user can search for words and store specific meanings in the database. Words can be stored in lists. Each user can create multiple lists. Each list must be named by the user. The user can then add new words or remove existing words for the list; tr, they can delete the entire wordlist. These lists can be shared with other users. When the user saves a list, it is saved as private, but when the user shares his/her word to the marketplace, the list then will then become publicly downloadable.

Memorization

Memorization techniques discussed below will be applied on user's created or downloaded lists.

Memorization and review word meaning by repetition: The user will be shown a word in a sentence. He will have to recall the meaning and check it with the answer. If he gets it right, he will tick mark it or else cross it. When he is done with all words in a list, the round will be repeated but only with the words that he has crossed.

Quiz: Users can take part in quizzes in order to judge their skills. Quizzes composed of MCQ to enhance active recalling for both, word meanings and spelling. Users can participate in new quizzes as many times as he wishes.

Daily training: A short training session will be automatically generated everyday, based on users' previous daily training results.

Smart suggestion: Questions for daily training will be curated specifically for each user. That is, the training sequence will be unique for each user. In addition, the sequence generation algorithm will ensure that the training sessions will include new words; old, but easy words; and old, but difficult words in the right ratio.

Gamification:

The app will allow users to learn through playing games. The app will have the following game:

Spelling master: The player will be shown meaning and pronunciation of a word. He will have to guess the spelling of the word. This will help users learn spelling for everyday use and prepare for international spelling bee competitions.

Daily/weekly progress and leaderboard:

Players will find themselves in the global leaderboard based on the points that they score in daily training.

Market Place

There will be a marketplace for the user where they can download word lists shared by other users. The user can also share his own word list there. All shared word lists will be stored in the server. These word lists will appear by the name “courses”.

Forum

There will be a forum for the users where they will be able to post their queries and share thoughts with others. Users can create posts and view posts made by others. Each user gets to like and comment on any post.

SCENARIO BASED MODELING OF WORD MASTER 3000

USE CASE DIAGRAM

Level 0

Use case Id: 0

Use case Name: Word Master 3000

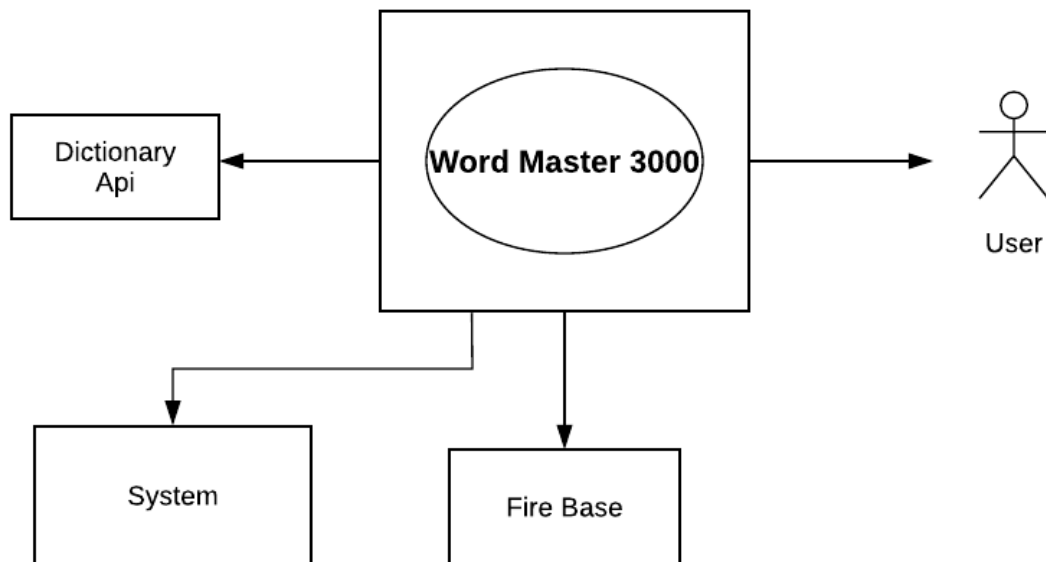


Fig 1: Level 0(use case diagram)

Primary Actor: User

Secondary Actor: Dictionary Api, Fire Base, System

Description: This use case shows the low level interaction between system and actors.

Level 1

Use case Name: WORD MASTER 3000 SUB SYSTEM

Use case Id: 1

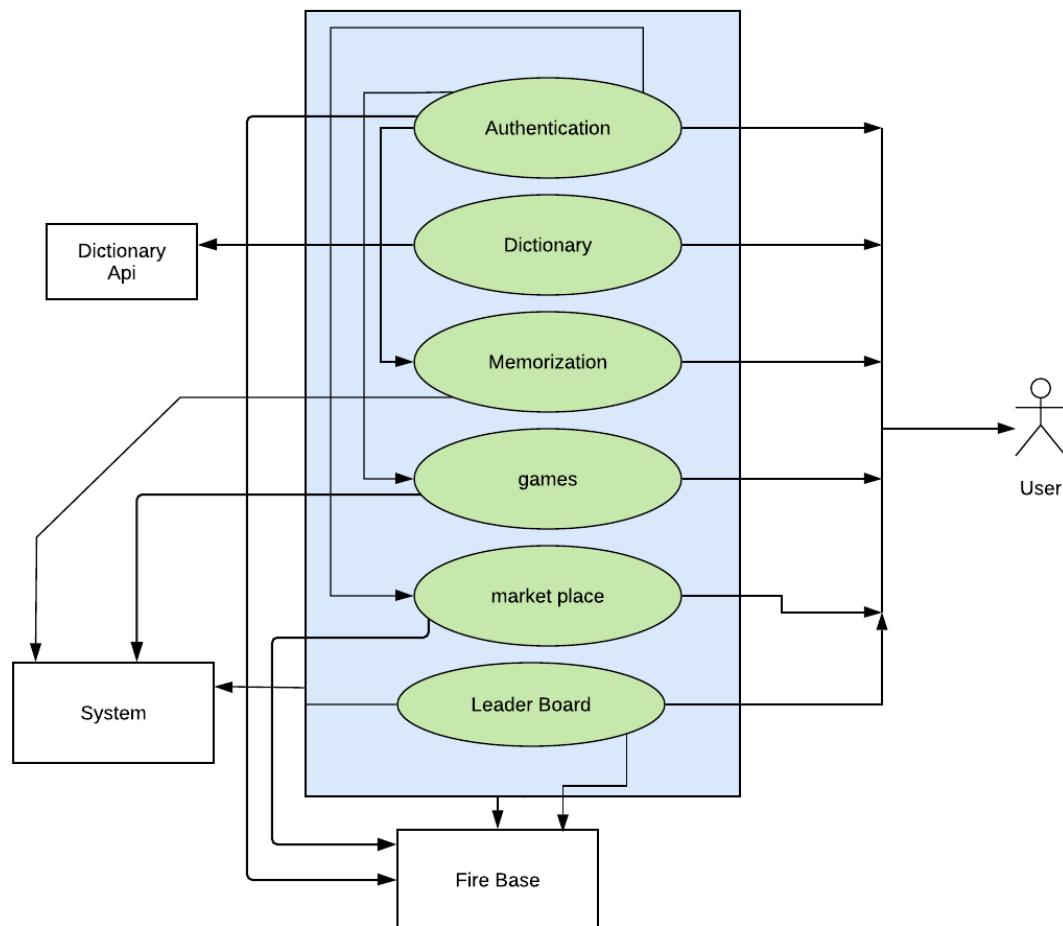


Fig: Level 1 (Use case diagram)

Primary Actor: User

Secondary Actor: Dictionary Api, Fire Base, System

Description: From this level all the subsystems of the proposed main system and connectivity of those subsystems through actors has been explicit. From this level interaction between actors and subsystems will be clearer. Here, the whole system is divided into six subsystem and Fire base is the outside system in this proposed system

Level 1.1

Use case Name: Authentication

Use case Id: 1.1

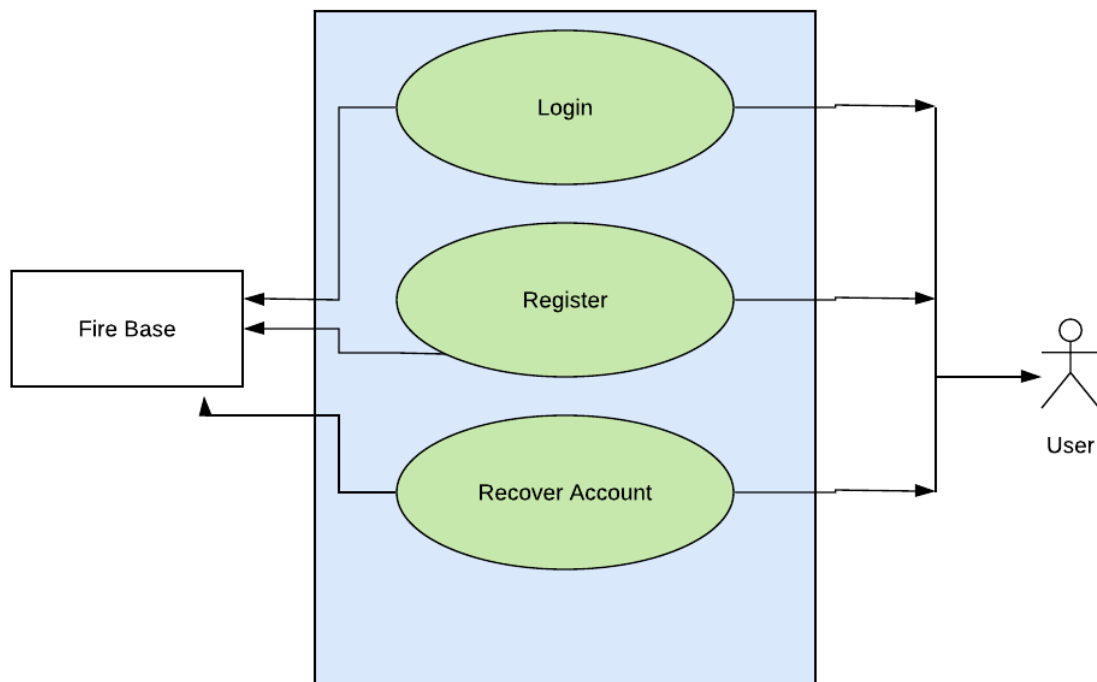


Fig: Level 1.1 (Use case diagram)

Primary Actor: User

Secondary Actor: Fire Base

Description: The whole authentication process has been explicit in this level in four more sub systems. They are Registration, log in & recover info.

Level 1.1.1

Use case Name: Login

Primary Actor: User

Secondary Actor: Fire Base

Description:

Action: The user provides user name and password or asks for anonymous login.

Reply: User is logged in.

Level 1.1.2

Use case Name: Register

Primary Actor: User

Secondary Actor: Fire Base

Description:

Action: The user provides name, email, user name, password.

Reply: Data is saved in the firebase and the user is logged in.

Level 1.1.3

Use case Name: Recover Account

Primary Actor: User

Secondary Actor: Fire Base

Description:

The user asks to recover his account if password or username is forgotten.
Firebase uses its verification process to verify users and recover his account.

Level 1.2

Use case Name: Dictionary

Use case Id: 1.2

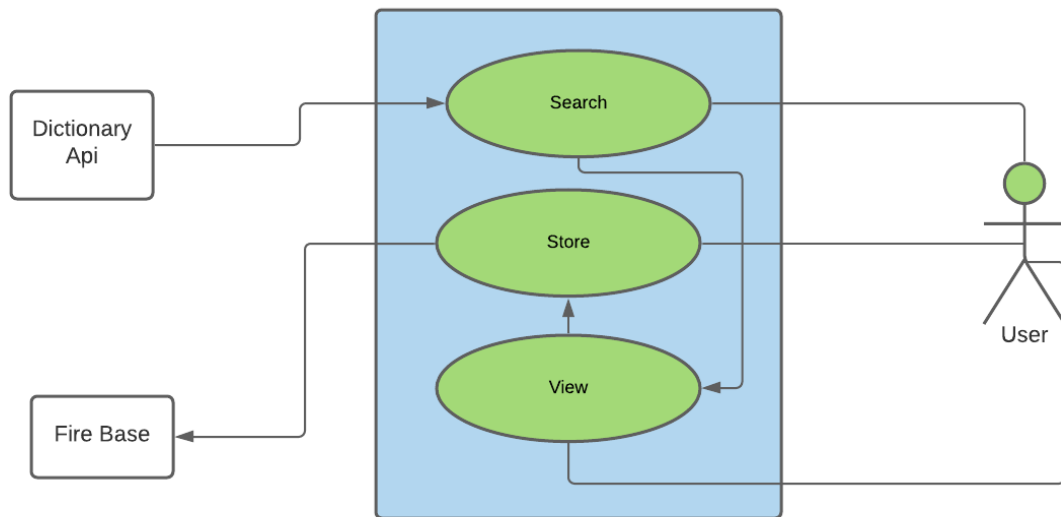


Fig: Level 1.2 (Use case diagram)

Primary Actor: User

Secondary Actor: Dictionary Api, Fire Base

Description:

The sub system is divided into three parts. Search, store and view.

Level 1.2.1

Use case Name: Search

Use case Id: 1.2.1

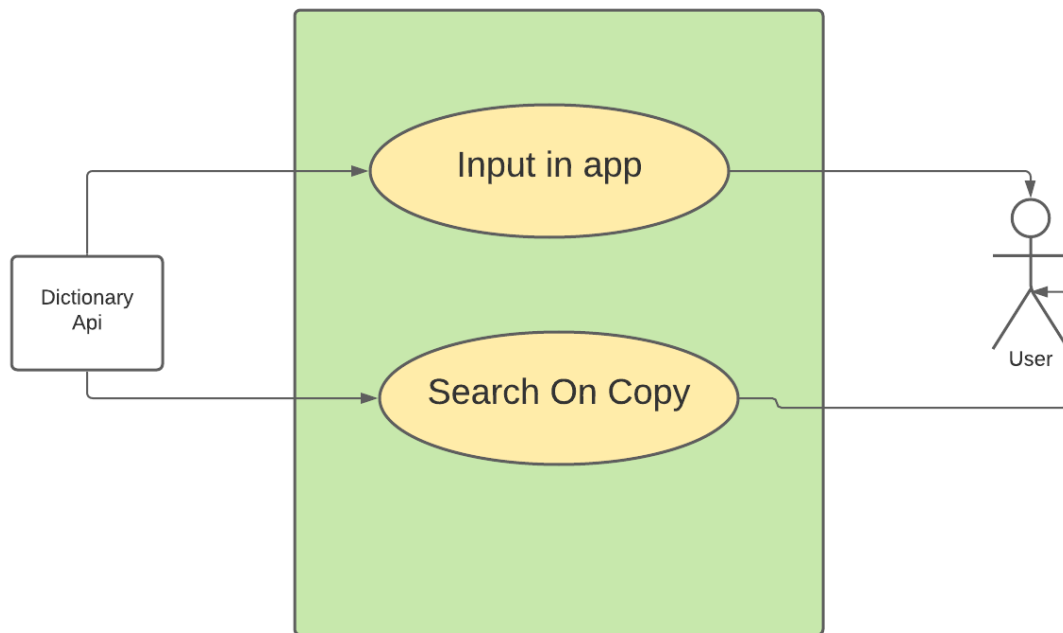


Fig: Level 1.2.1 (Use case diagram)

Primary Actor: User

Secondary Actor: Dictionary Api

Description:

The sub system is divided into three parts. Users can search for a word in three different ways.

Level 1.2.1.1

Use case Name: Input in app

Primary Actor: User

Secondary Actor: Dictionary Api.

Description:

Action: The user provides the desired word directly in the app.

Reply: The word is sent to view and the word is viewed.

Level 1.2.1.2

Use case Name: Search on copy

Primary Actor: User

Secondary Actor: Dictionary Api.

Description:

Action: The user copy's a word to the clipboard.

Reply: The word is sent to view and the word is viewed.

Level 1.2.3

Use case Name: View

Primary Actor: User

Secondary Actor: Dictionary Api.

Description:

Action: searched word is found.

Reply: The word is sent to viewed to the user.

Level 1.2.2

Use case Name: Store

Use case Id: 1.2.2

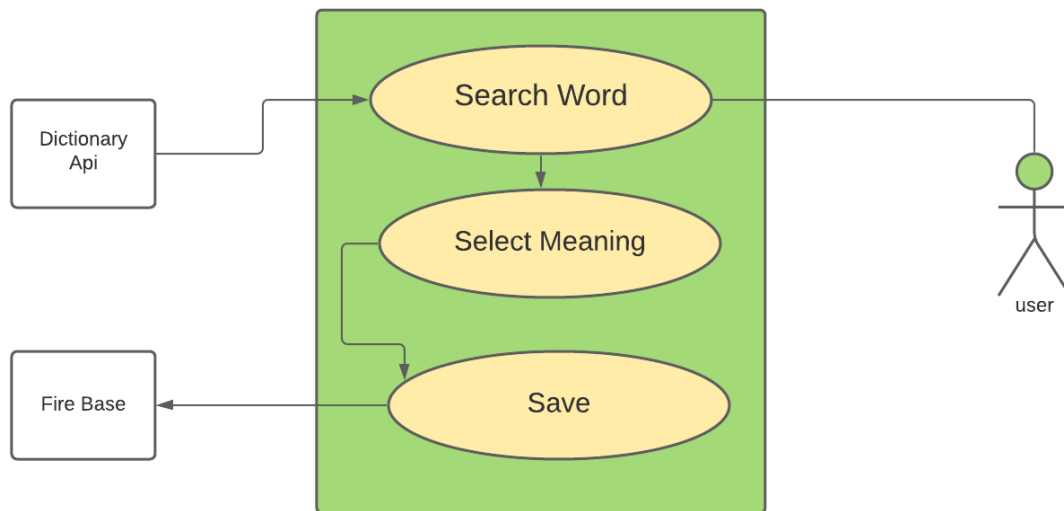


Fig: Level 1.2 .2(Use case diagram)

Primary Actor: User

Secondary Actor: Fire base, Dictionary api

Description:

The sub system is divided into two parts. Users can select a word from the view and store it.

Level 1.2.2.1

Use case Name: Search word

Primary Actor: User

Secondary Actor: Dictionary api

Description:

The user searches for a word from the dictionary api.

Level 1.2.2.2

Use case Name: Select Meaning

Primary Actor: User

Description:

The user selects a meaning of the word he/she searched for.

Level 1.2.2.3

use case id: 1.2.2.3

Use case Name: Save

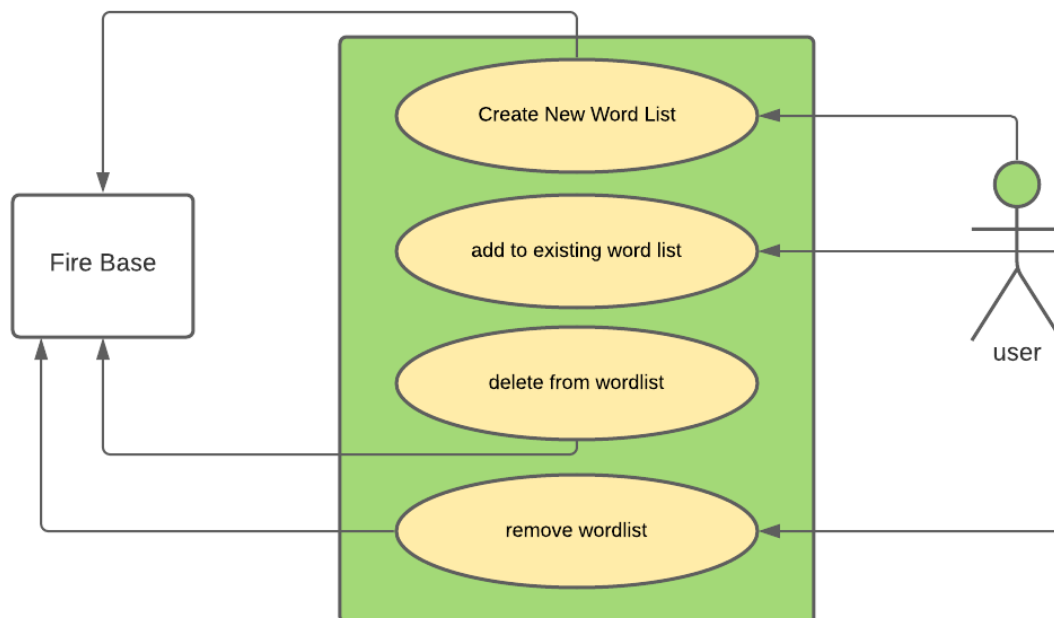


Fig : Level 1.2.2.3 (Use Case Diagram)

Primary Actor: User

Secondary Actor: Fire Base

Description:

The selected word is sent to fire base to save it against the users word list.

Level 1.2.2.3.1

Use case Name: Create New Word List

Primary Actor: User

Secondary Actor: Fire Base

Description:

The user can create a new wordlist for saving the selected meaning.

Level 1.2.2.3.2

Use case Name: Add to existing word list

Primary Actor: User

Secondary Actor: Fire Base

Description:

The user can save the meaning in any existing word list.

Level 1.2.2.3.3

Use case Name: Delete from word list

Primary Actor: User

Secondary Actor: Fire Base

Description:

The user can delete the meaning in any existing word list.

Level 1.2.2.3.4

Use case Name: Remove word list

Primary Actor: User

Secondary Actor: Fire Base

Description:

The user can delete the entire word list.

Level 1.3

Use case Name: Memorization

Use case Id: 1.3

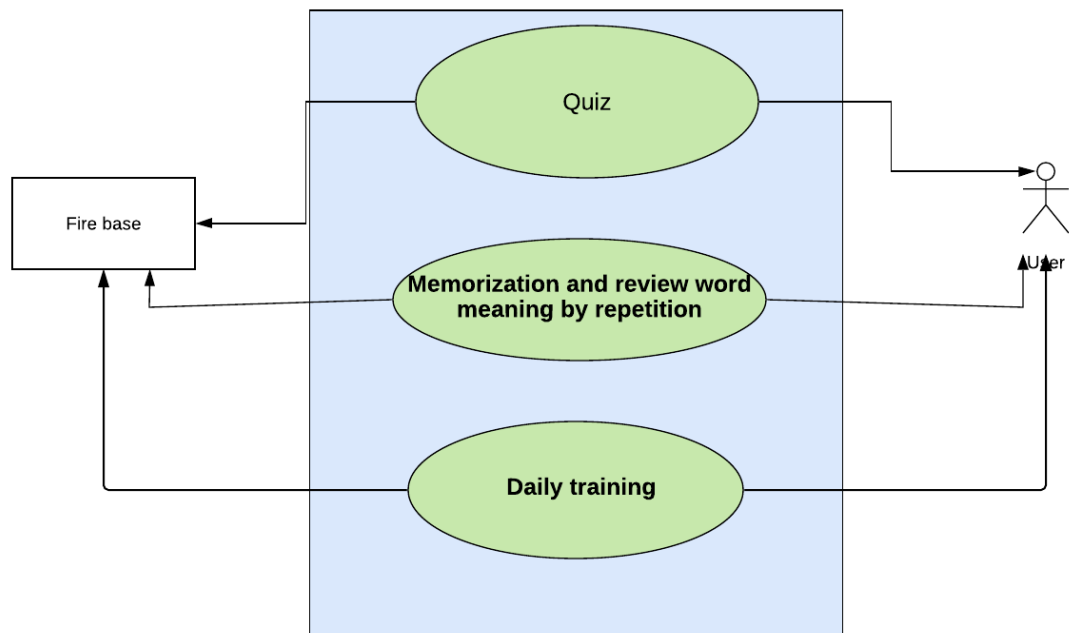


Fig: Level 1.3(Use case diagram)

Primary Actor: User

Secondary Actor: Fire base

Description:

The sub system is divided into three parts. User can take part in a quiz. Besides there is Memorization and review word meaning by repetition and daily training.

Level 1.3.1

Use case Name: Quiz

Use case Id: 1.3.1

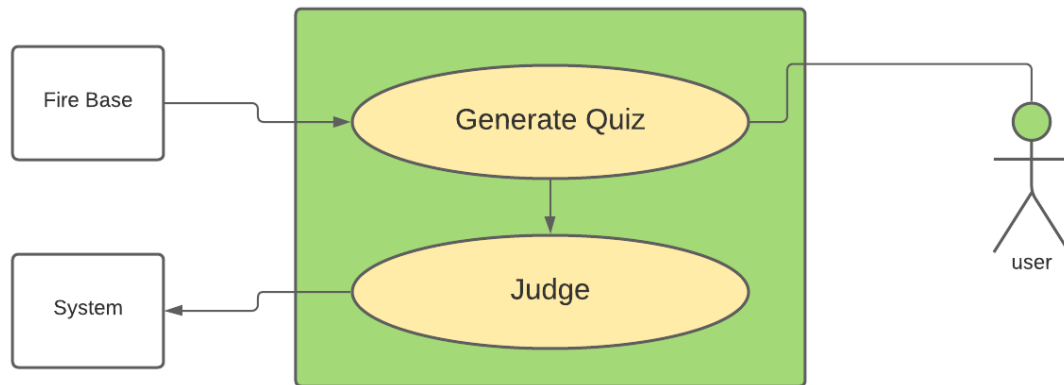


Fig: Level 1.3.1(Use case diagram)

Primary Actor: User

Secondary Actor: Fire base

Description:

The sub system quiz is divided into five parts. Smart Suggestion, generate quiz, store history, judge, analysis.

Level 1.3.1.1

Use case Name: Generate quiz

Primary Actor: User

Secondary Actor: Fire Base, System

Description:

Users can take part in quizzes in order to judge their skills. Quizzes composed of MCQ, active recalling and fill-in-the-blanks for both, word meanings and spelling. Users can participate in new quizzes or continue incomplete quizzes. Questions for the quiz will be automatically generated by the server with help of smart suggestions.

Level 1.3.1.2

Use case Name: judge

Secondary Actor: Fire Base, System

Description:

System will judge the answer given by the user whether the answer is right or wrong. And thus, generate scores.

Level 1.3.2

Use case Name: Memorize by repetition

Use case Id: 1.3.2

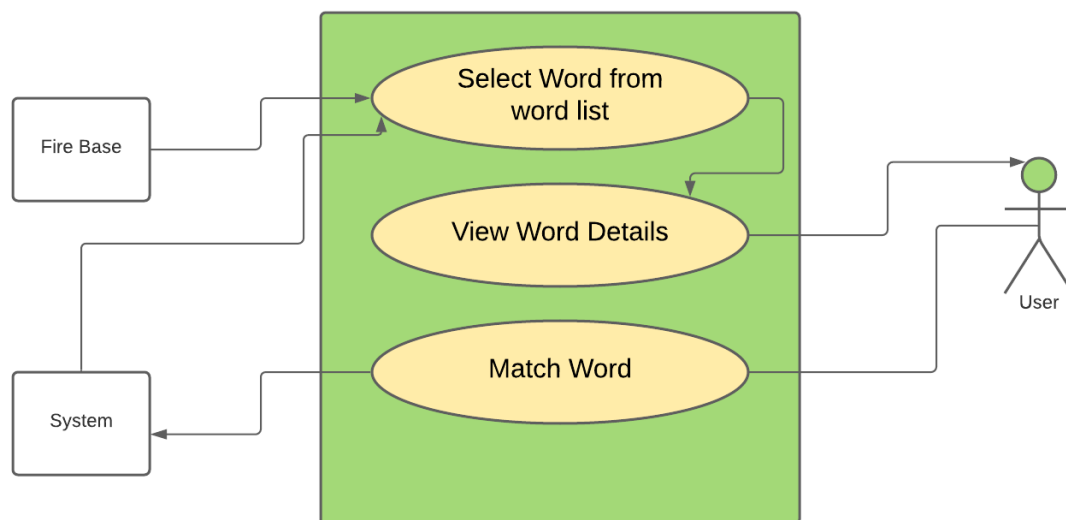


Fig: Level 1.3.2(Use case diagram)

Primary Actor: User

Secondary Actor: Fire base, system

Description:

The sub system quiz is divided into thre parts. Generate sentence for the user, let user review the sentence and finally match the answer

Level 1.3.2.1

Use case Name: Select word from Word list

Secondary Actor: Fire Base, System

Description:

The system will choose a word from the wordlist of the user kept in fire base.

Level 1.3.2.2

Use case Name: view word details

Secondary Actor: Fire Base

Description:

The selected word details from the previous subsystem (1.3.2.1) will be viewed to the user.

Level 1.3.2.3

Use case Name: match word

Primary Actor: User

Secondary Actor: Fire Base, System

Description:

Action: the user will check or cross the word.

Reply: the user submitted answer will be checked and if it is crossed the word will again appear else it will be considered as memorized.

Level 1.3.3

Use case Name: Daily training

Primary Actor: User

Secondary Actor: Fire Base

Description:

A short training session will be automatically generated everyday based on users' previous quiz and memorization results.

Level 1.4

Use case Name: games

Use case Id: 1.4

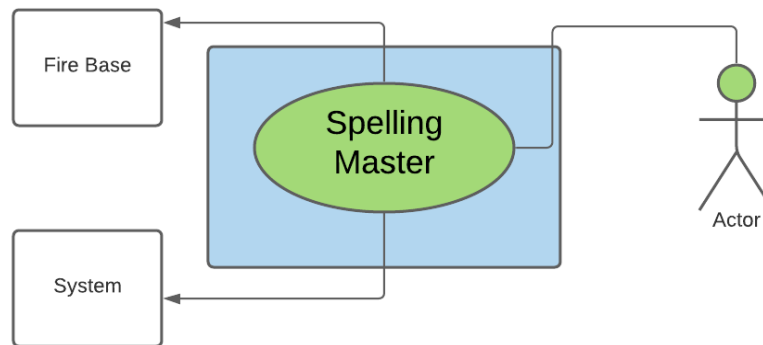


Fig: Level 1.4(Use case diagram)

Primary Actor: User

Secondary Actor: Fire base, System

Description: There are games for the users currently. Spelling Master

Level 1.4.1

Use case Name: Spelling Master

Use case Id: 1.4.1

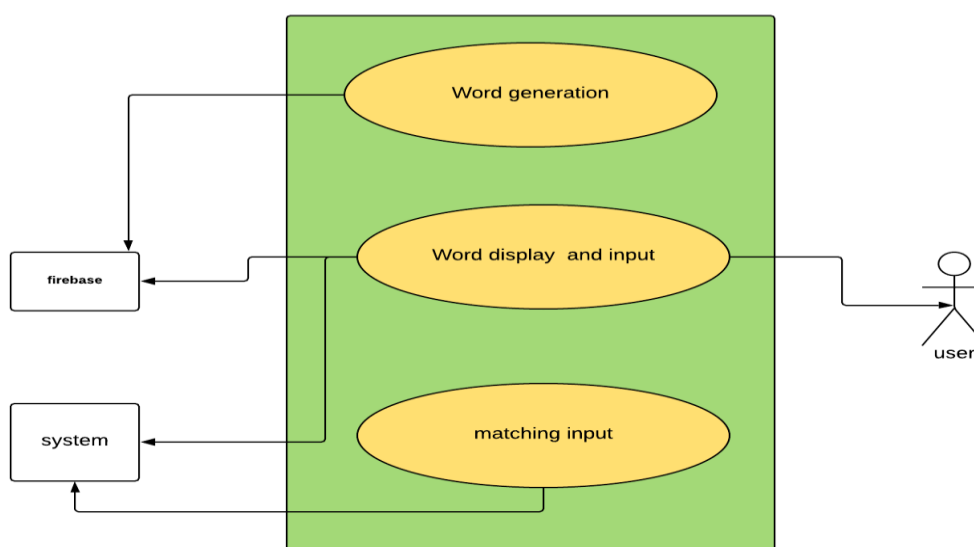


Fig: Level 1.4.1(Use case diagram)

Primary Actor: User

Secondary Actor: Dictionary api, System

Description:

There are three parts in the game spelling master. They are mainly generating a word, display it and then take user input.

Level 1.4.1.1

Use case Name: Word generation

Primary Actor: User

Secondary Actor: Dictionary api, System

Description:

System selects a word from the dictionary to start the game.

Level 1.4.1.2

Use case Name: display and input

Primary Actor: User

Secondary Actor: Dictionary api, System

Description:

The player will be shown meaning and pronunciation. He will have to guess the spelling of the word and give it as input.

Level 1.4.1.3

Use case Name: Match Input

Secondary Actor: Dictionary api, System

Description:

Match the real answer and match the user given answer. If matches issue point.

Level 1.5

Use case Name: Market Place

Use case Id: 1.5

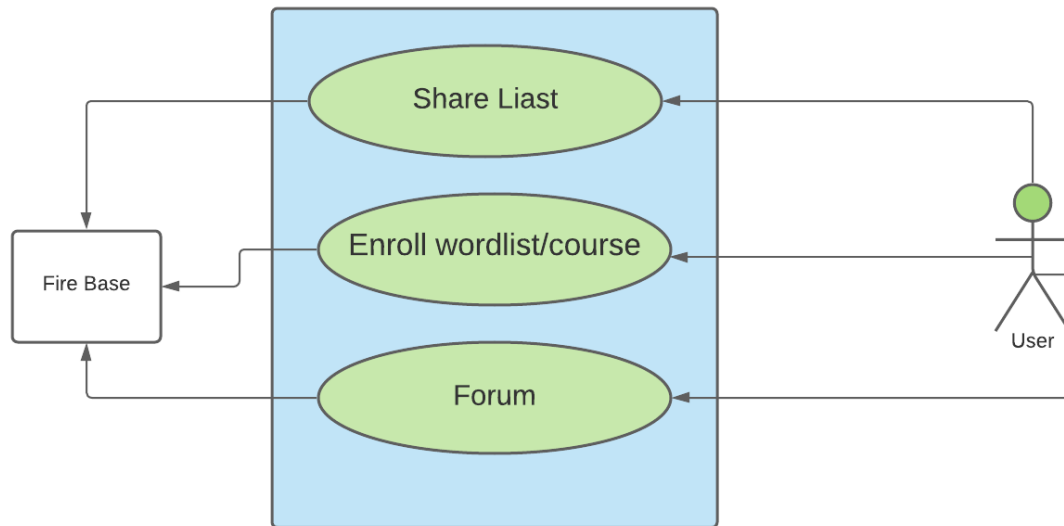


Fig: Level 1.5(Use case diagram)

Primary Actor: User

Secondary Actor: Fire Base

Description:

The whole subsystem has been divided into three parts. Share list, share meme and download list.

Level 1.5.1

Use case Name: Share list

Use case Id: 1.5.1

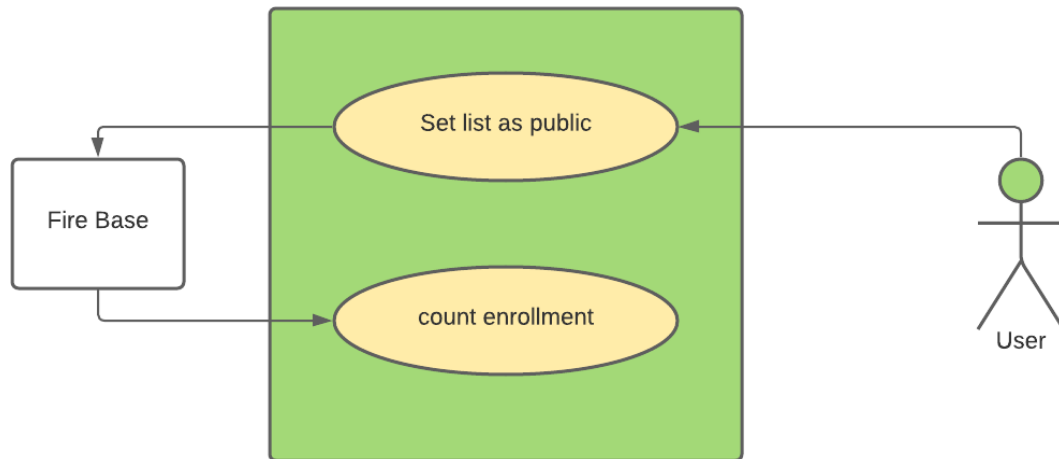


Fig: Level 1.5.1(Use case diagram)

Primary Actor: User

Secondary Actor: Fire Base

Description:

Sharing list has two parts. Set list type as public and add tags.

Level 1.5.1.1

Use case Name: public list

Primary Actor: User

Secondary Actor: fire base

Description:

Action: User changes the word list type from private to public.

Reply: The list is now public and in market place.

Level 1.5.1.2

Use case Name: Count enrollment

Secondary Actor: fire base

Description:

Number of Enrollments in the course is counted

Level 1.5.2

Use case Name: Enroll course/wordlist

Primary Actor: User

Secondary Actor: fire base

Description:

User can enroll the course / word list of his/her choice.

Level 1.5.3

Use case Name: Forum

Use case Id: 1.5.2

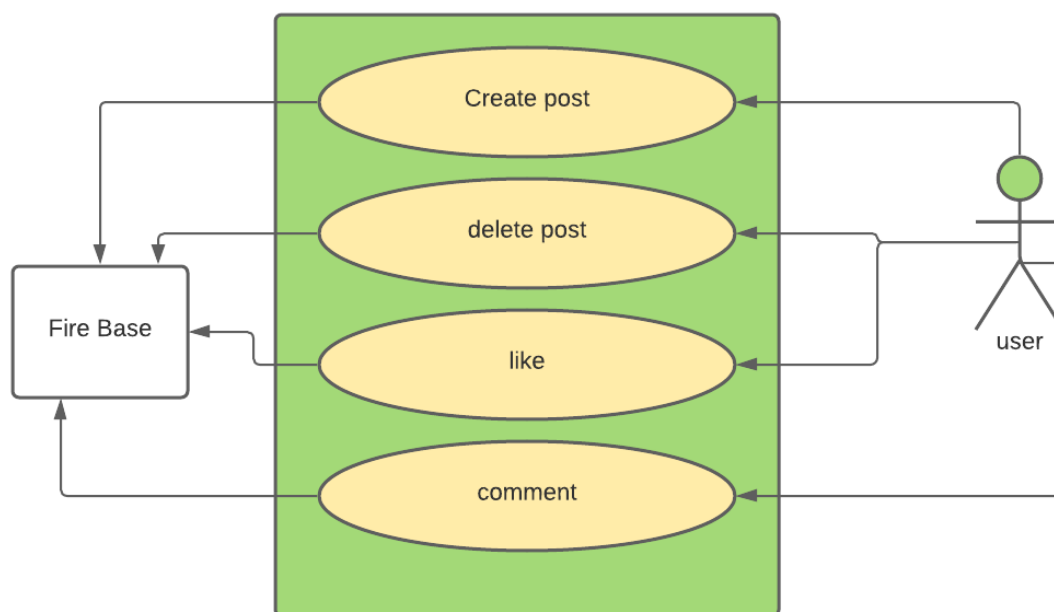


Fig: Level 1.5.2(Use case diagram)

Primary Actor: User

Secondary Actor: Fire Base

Description:

The whole share meme process has been divided into four parts

Level 1.5.2.1

Use case Name: Create post

Primary Actor: User

Secondary Actor: fire base

Description:

Action: User creates a post.

Reply: The post is saved against the user.

Level 1.5.2.2

Use case Name: delete post

Primary Actor: User

Secondary Actor: fire base

Description:

User can delete any post which he/she has posted.

Level 1.5.2.3

Use case Name: like

Primary Actor: User

Secondary Actor: fire base

Description:

Action: User likes a post.

Reply: Like count of the post is increased.

Level 1.5.2.4

Use case Name: comment

Primary Actor: User

Secondary Actor: fire base

Description:

User can comment in a post which will be saved under that post.

Level 1.6

Use case Name: Leader Board

Primary Actor: User

Secondary Actor: fire base

Description:

Players will find themselves in the global leaderboard based on the points that they score in games and quizzes.

Activity Diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency.

3.1 Application

Activity name: Application

Activity id: 3.1

Version 1.0

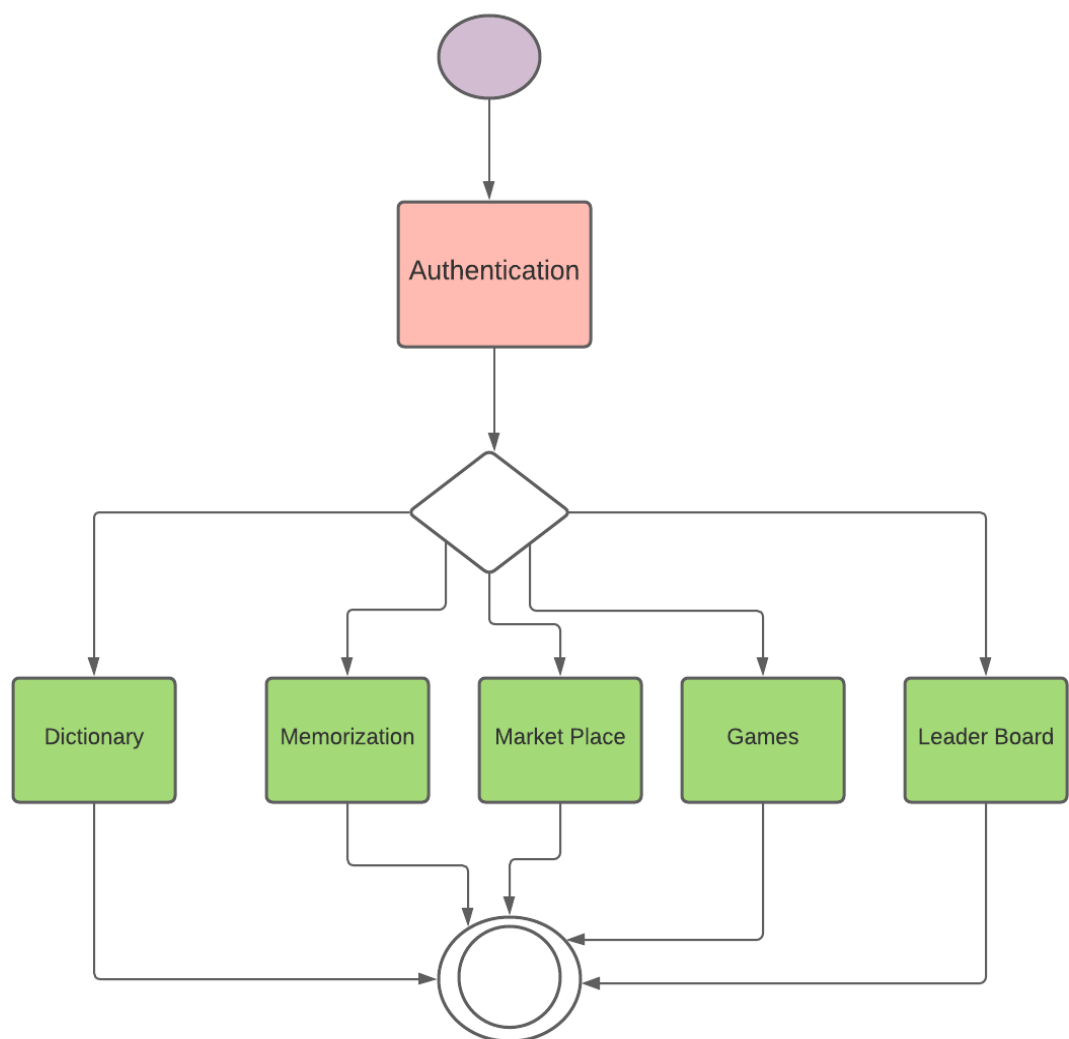


Fig: Application Activity Diagram

3.2 Authentication

Activity name: Authentication

Activity id: 3.1

Version 1.0

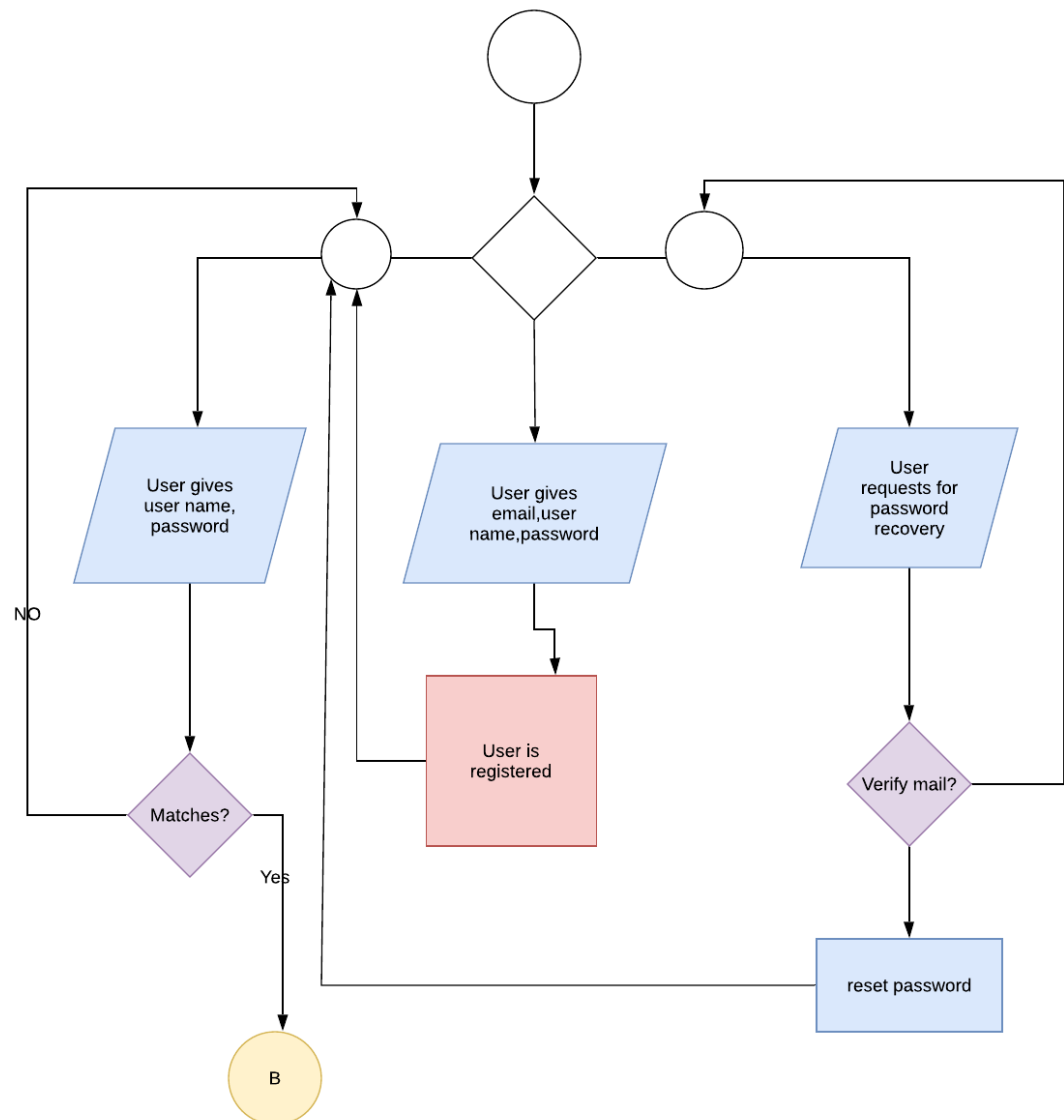


Fig: Authentication Activity Diagram

3.3 Dictionary Words

Activity name: Dictionary Words

Activity id: 3.3

Version 1.0

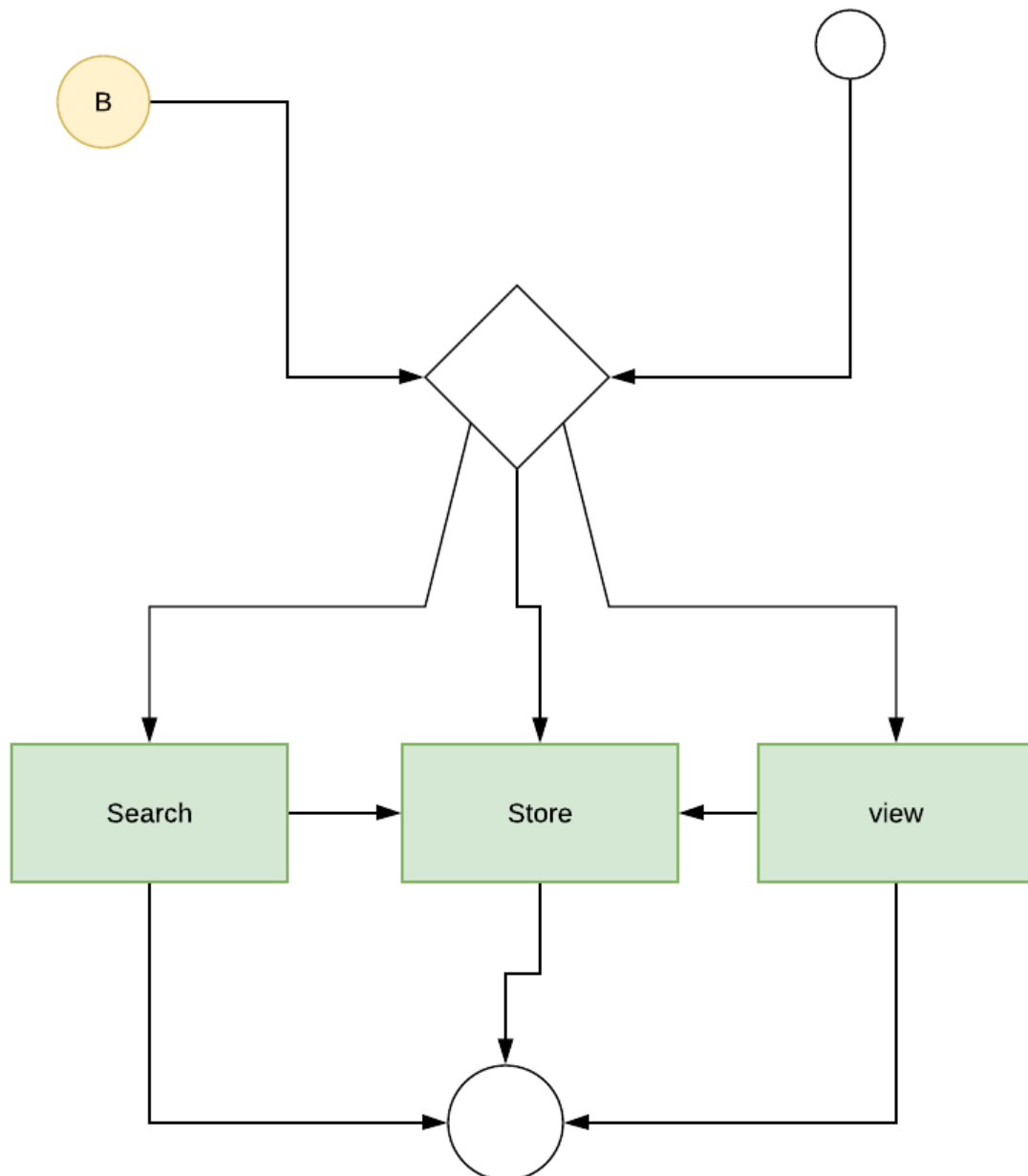


Fig: Dictionary Words Activity Diagram

3.3.1 Search

Activity name: Search

Activity id: 3.3.1

Version: 1.0

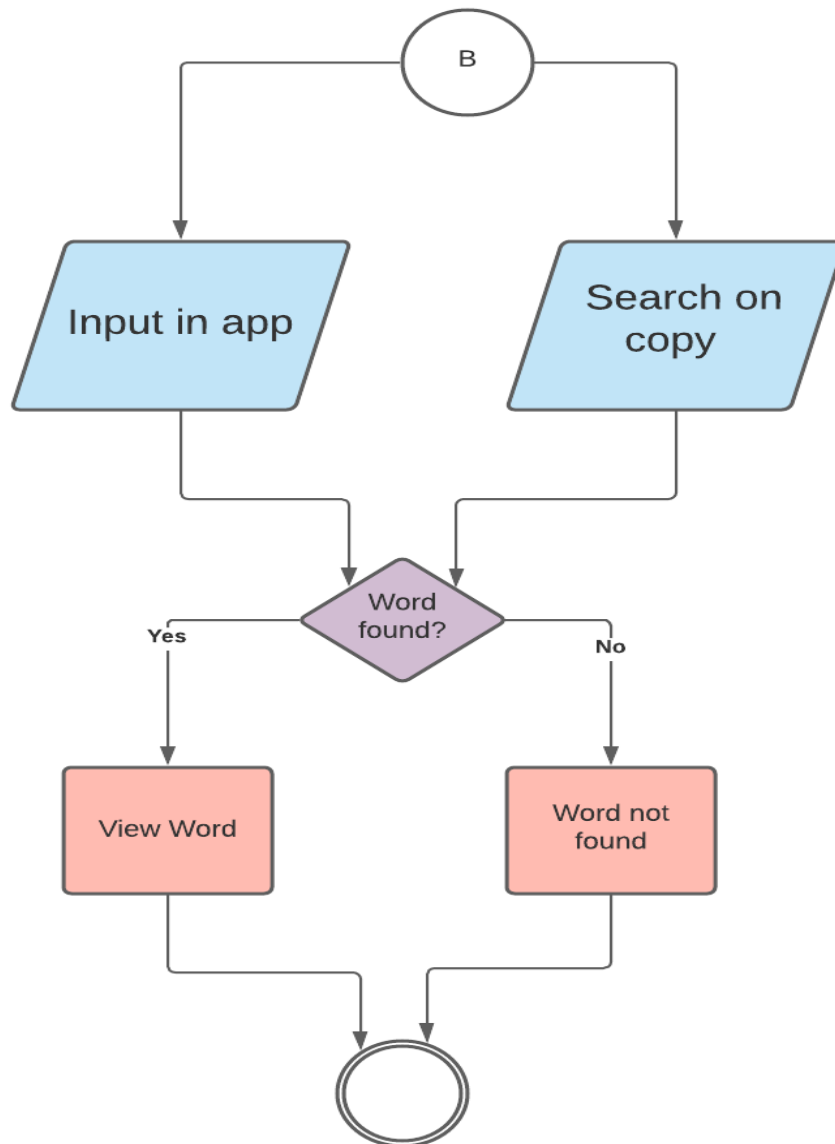


Fig: Search Activity Diagram

3.4 Memorization

Activity name: Memorization

Activity id: 3.4

Version: 1.0

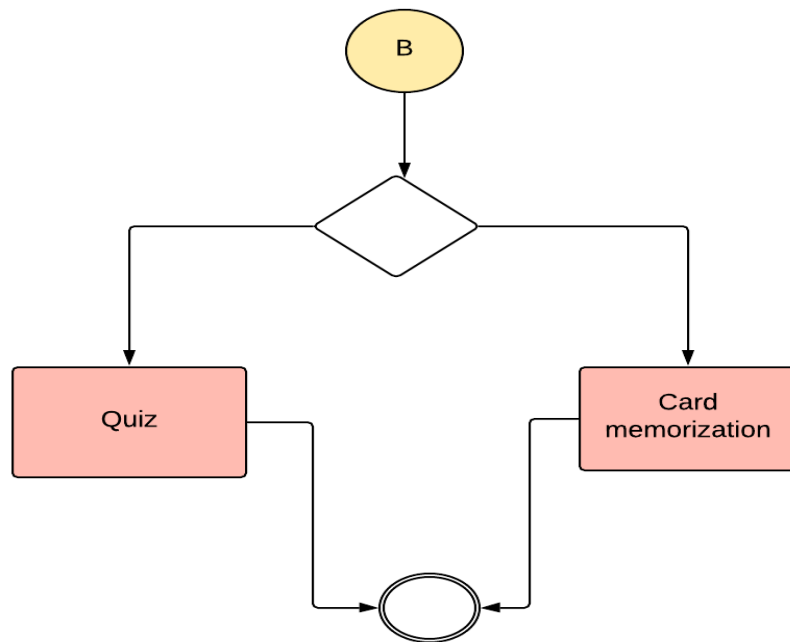


Fig: Memorization Activity Diagram

3.4.1 Quiz

Activity name: Quiz

Activity id: 3.4.1

Version: 1.1

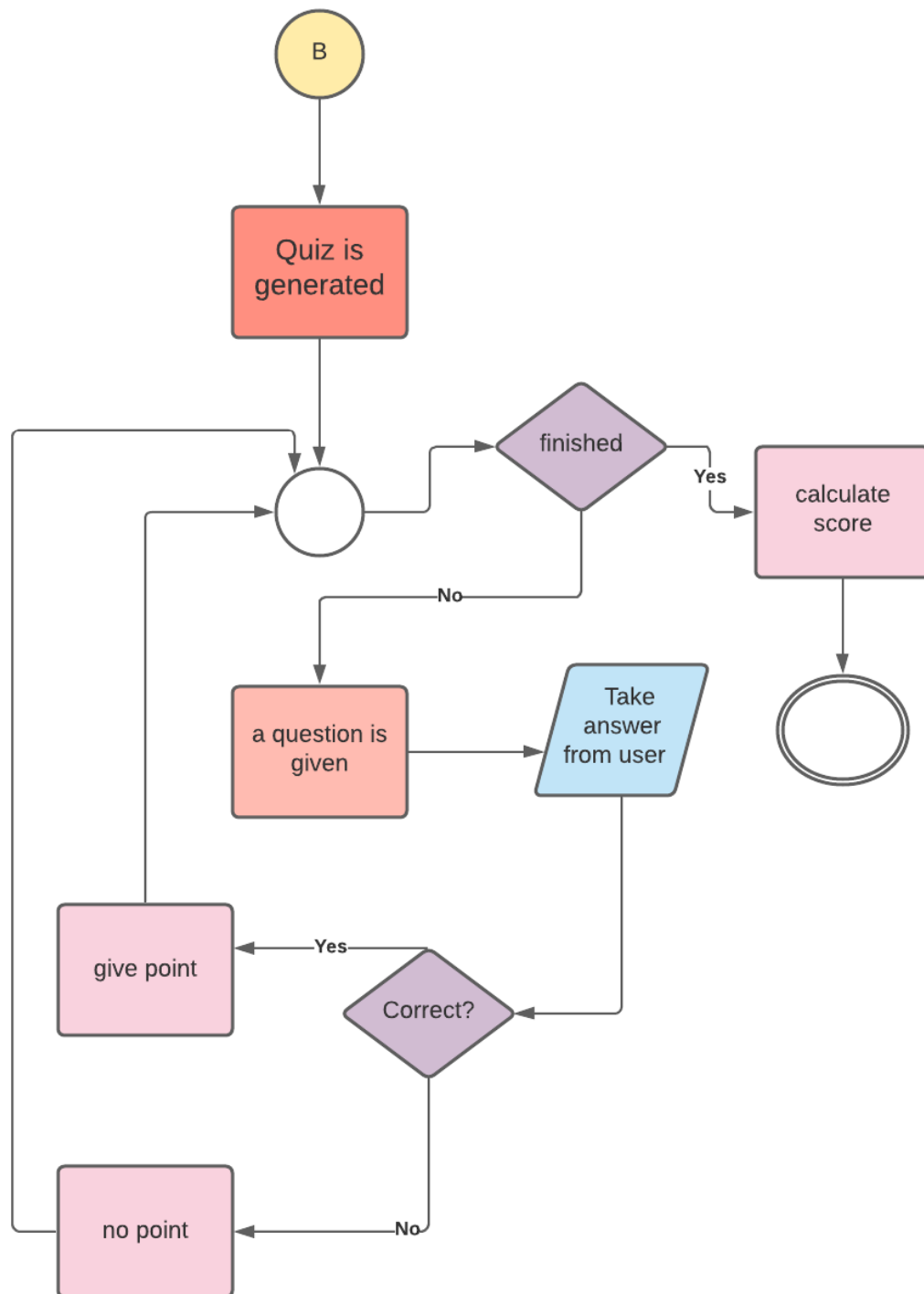


Fig: Quiz Activity Diagram

3.4.2 Memorization by repetition

Activity name: Memorization by repetition

Activity id: 3.4.2

Version: 1.1

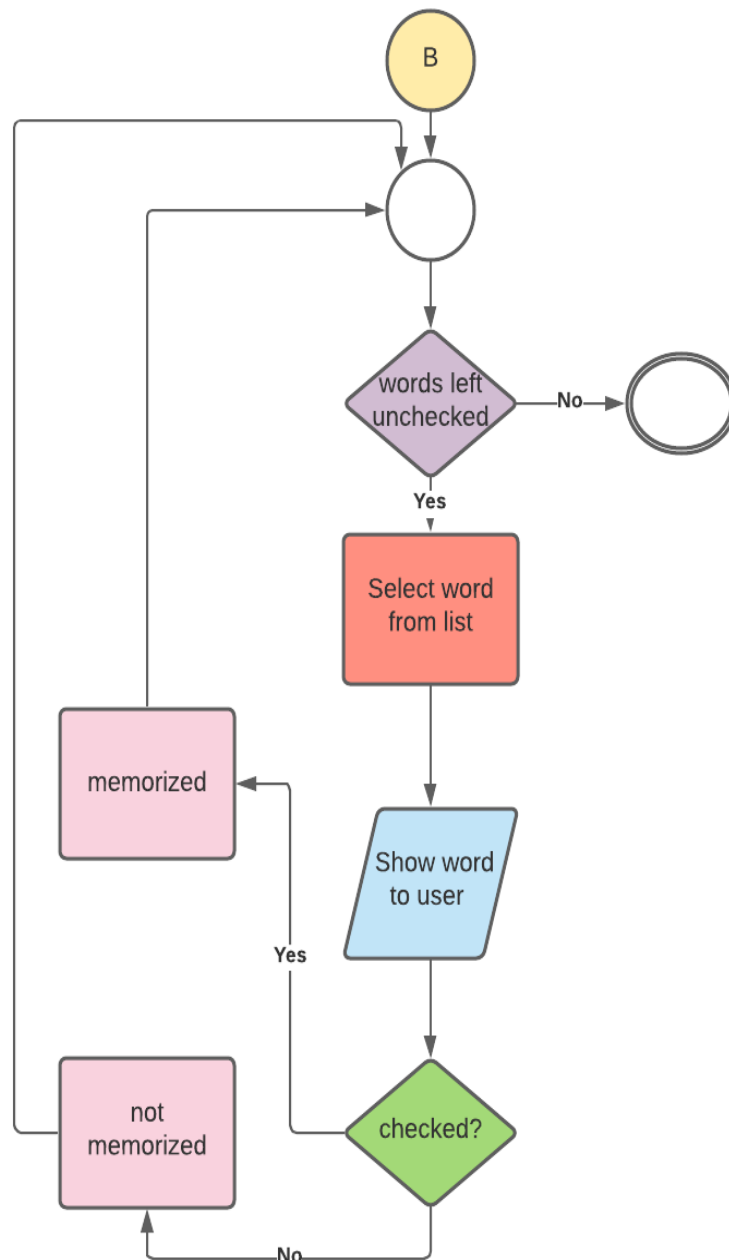


Fig: Memorization by repetition activity diagram

3.5.2 Spelling Master (Game)

Activity name: Spelling Bee

Activity id: 3.5.2

Version: 1.0

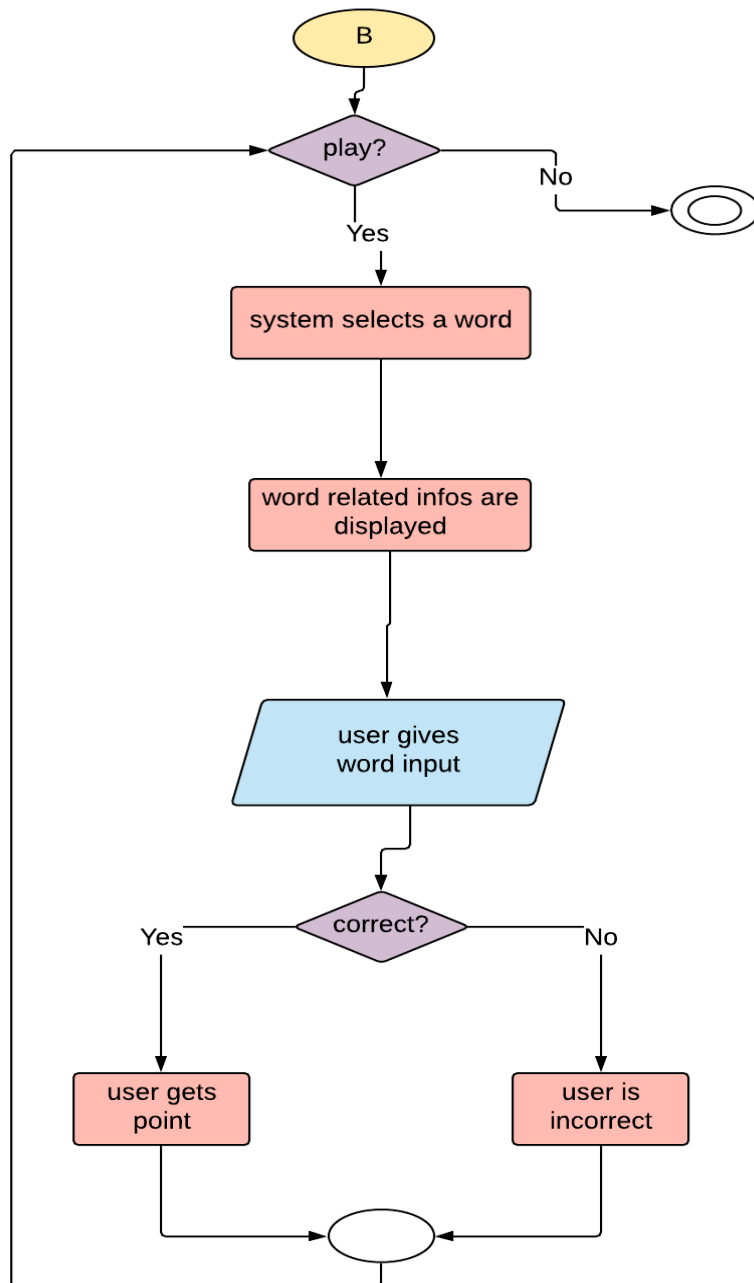


Fig: Spelling Bee Activity Diagram

3.6 Marketplace

Activity name: Market Place

Activity id: 3.6

Version: 1.0

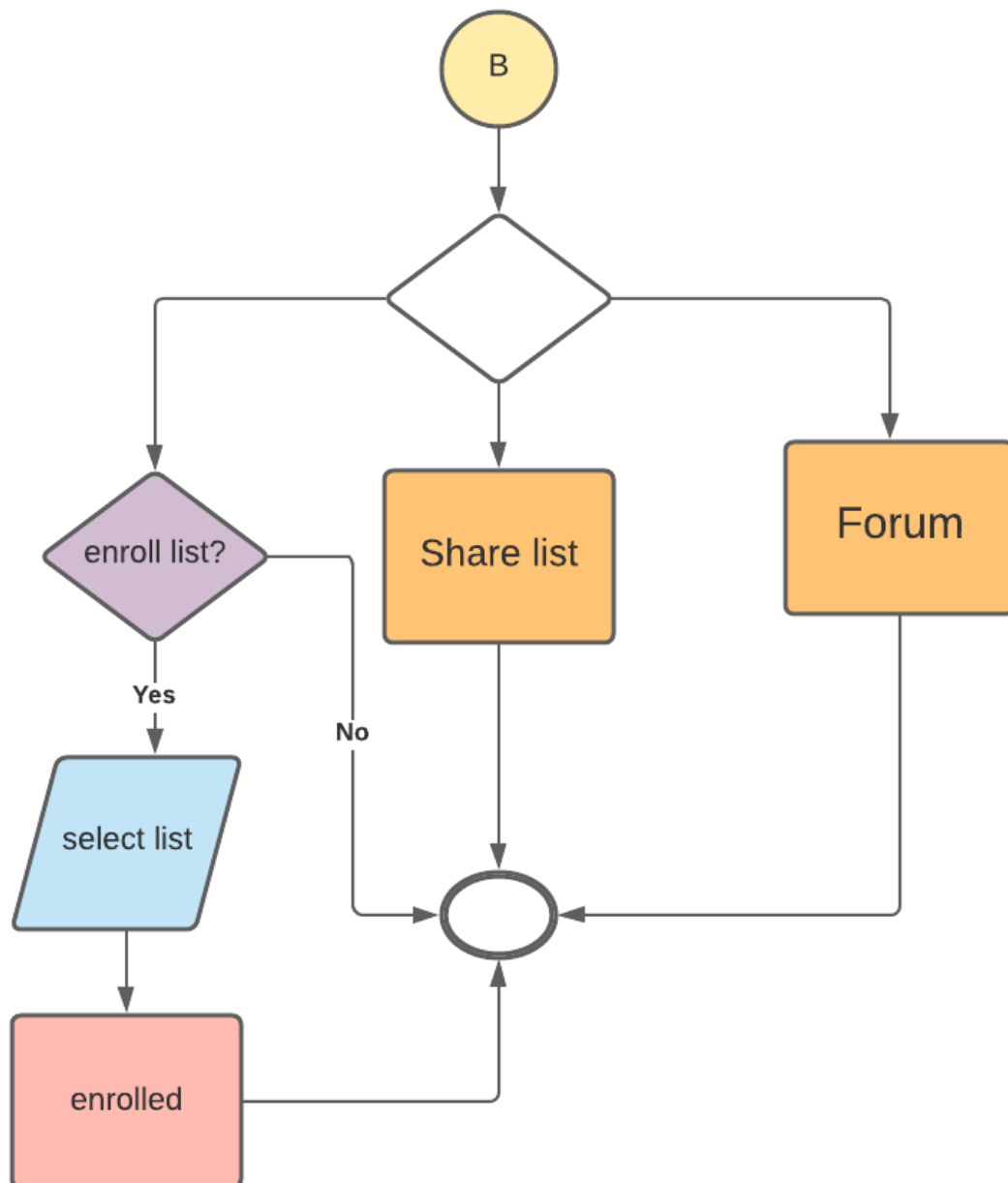


Fig: Market place Activity Diagram

3.6.1 Forum

Activity name: Forum

Activity id: 3.6.1

Version: 1.0

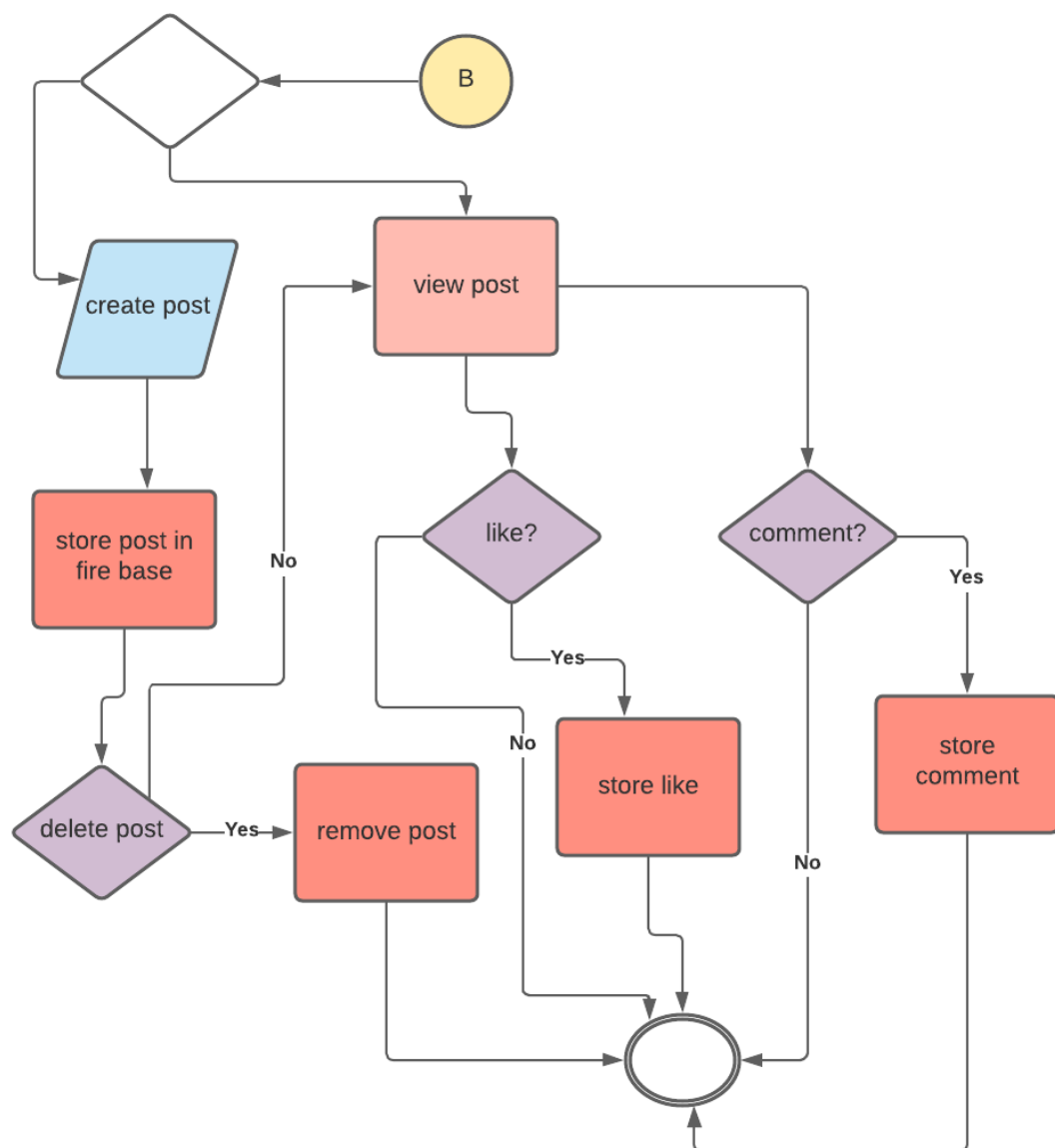


Fig: Share meme activity diagram

3.6.2 Share wordlist

Activity name: Share wordlist

Activity id: 3.6.2

Version: 1.0

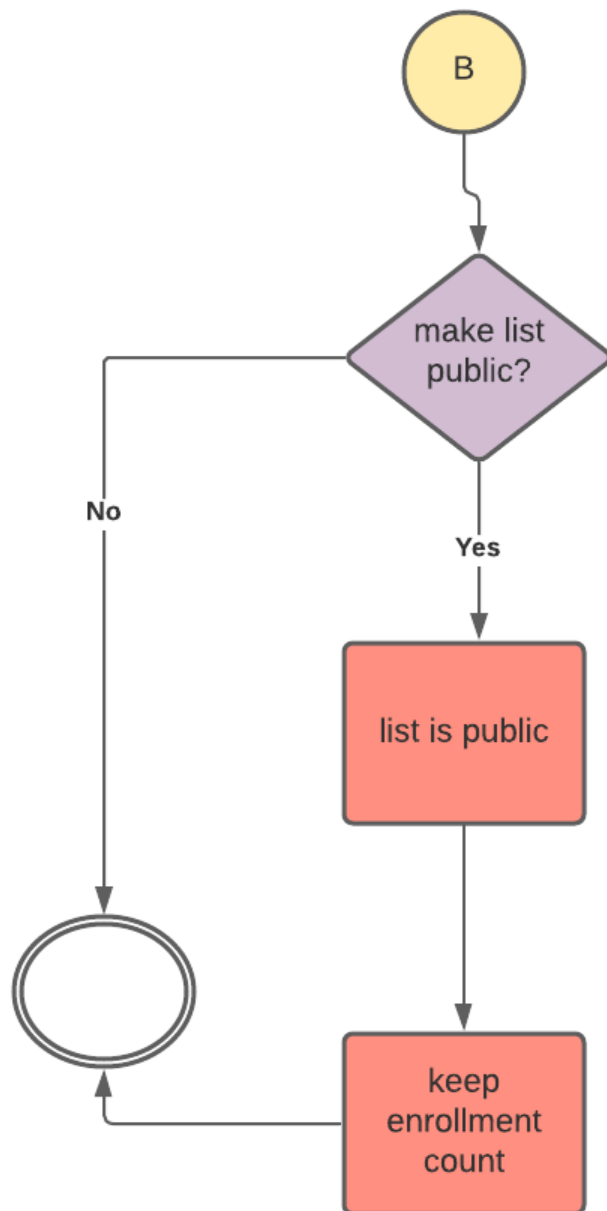


Fig: Share wordlist activity diagram

Data Based Modeling For Word Master 3000

The application depends on two separate databases. One is an offline database for the dictionary, and the other is an online database for storing user and his training related information.

ER DIAGRAM (Dictionary database)

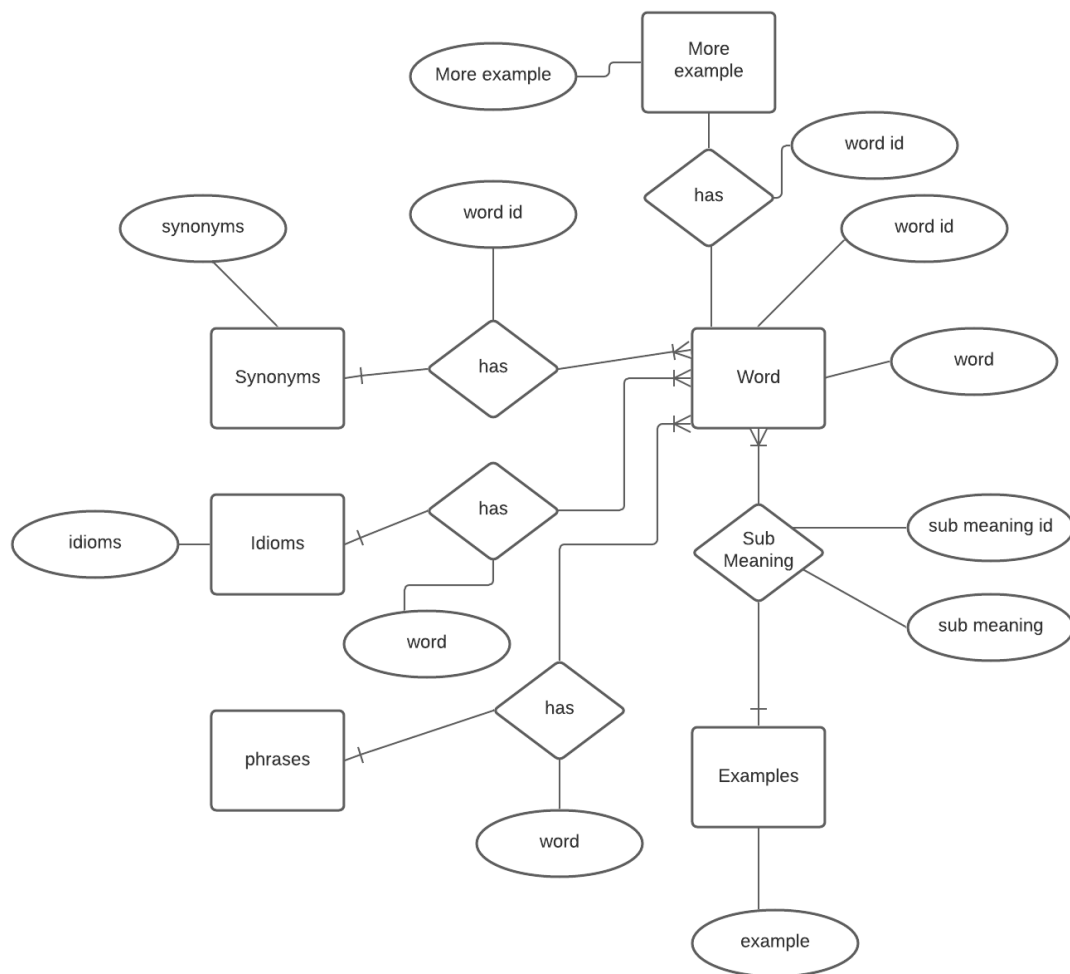


Fig: ER Diagram for word master 3000

Schema Diagram (Dictionary database)

Word		
Attribute	Type	Size
<u>Word ID</u>	Int	8
Word	String	200
Parts of speech	String	200
Meaning	String	200

Sub Meaning		
Attribute	Type	Size
<u>Word ID</u>	Int	8
<u>Sub meaning id</u>	int	8
Sub meaning	String	200

Example		
Attribute	Type	Size
<u>Sub meaning id</u>	int	8
<u>Example</u>	Dtring	200

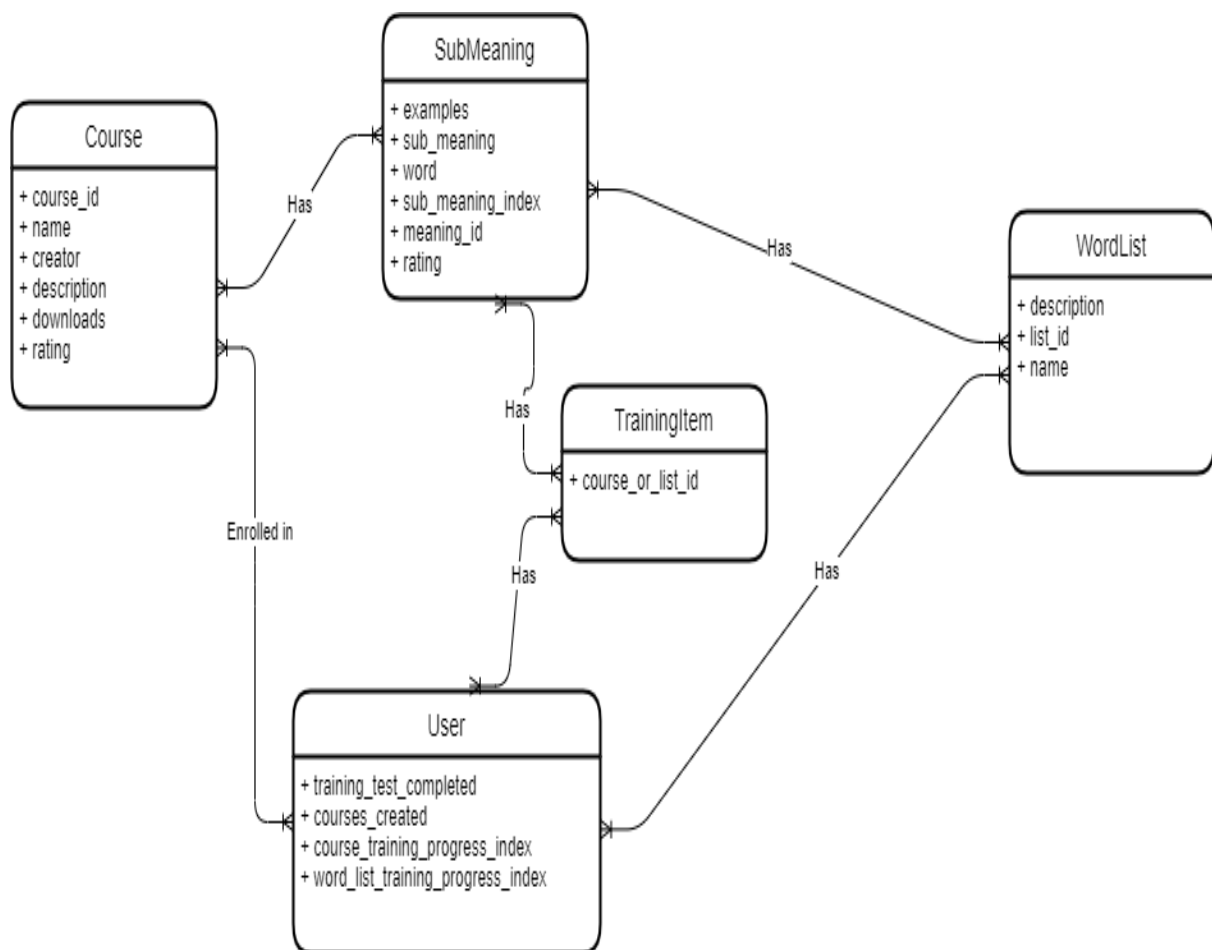
Synonyms		
Attribute	Type	Size
<u>Word ID</u>	Int	8
<u>synonyms</u>	string	200

phrases		
Attribute	Type	Size
<u>word</u>	string	200
phrases	String	200

Idioms		
Attribute	Type	Size
<u>word</u>	string	200
idioms	String	200

More Examples		
Attribute	Type	Size
<u>word ID</u>	Int	8
more example	String	200

Data objects (Cloud storage)





CLASS-BASED MODELING FOR

Word Master 3000

In this part we have discussed about the classes along with their attributes and methods of our application WORD MASTER 3000

CLASS BASED MODELING CONCEPT

Class-based Modeling represents the object. The system manipulates the operations. The elements of the class-based model consist of classes and objects, attributes, operations, class – responsibility - collaborator (CRS) models.

GENERAL CLASSIFICATION

To identify the potential classes, we have first selected the nouns from the solution space of the story. We have used seven general characteristics for this. The seven general characteristics are as follows:

- 1. External entities**
- 2. Things**
- 3. Events**
- 4. Roles**
- 5. Organizational units**
- 6. Places**
- 7. Structures**

General Classification of Word Master 3000

Serial	Noun	
1	Authentication	3
2	MCQ	2,3
3	Fill – in – the blanks	2,3
4	account	2,3,4,7
5	antonyms	
6	check- box	3
7	clipboard	2,7
8	courses	2,7
9	database	1,2,7

10	dictionary	2
11	email	
12	example	
13	game	2,3,7
14	wordlist	2,7
15	leaderboard	2,3,7
16	link	
17	marketplace	2,3,6,7
18	meaning	
19	word	2,7
20	memorization	
21	username	2
22	password	
23	player	
24	point	
25	daily progress	2,3,7
26	pronunciation	
27	question	2
28	quiz	2,3,7
29	result	
30	round	2
31	score	
32	sentence	
33	server	1,2,7
34	spelling	
35	suggestion	2,3
36	synonyms	
37	system	2,3,7
38	Spelling master	2,3,7
39	time	
40	training	
41	user	2,4,5,7
42	forum	2, 7
43	post	2, 7

44	like	2
45	comment	2

SELECTION CRITERIA

The potential classes were then selected as classes by six Selection Criteria. A potential class becomes a class when it fulfills all six characteristics.

- **Retain information**
- **Needed services**
- **Multiple attributes**
- **Common attributes**
- **Common operations**
- **Essential requirements**

Serial	Noun	Selection criteria
1	MCQ	1
2	Fill – in – the blanks	1
3	account	1,2,3,4,5,6
4	clipboard	1
5	courses	1,2,3,4,5,6
6	database	1,2,3,6
7	game	1,2,5,6
8	leaderboard	1,2,5,6
9	marketplace	1,2,3,4,5,6
10	forum	1,2,3,4,5,6
11	player	1,2,3,4,5
12	daily progress	1,6
13	quiz	1,2,3,4,6
14	server	1,2,6
15	system	6
16	user	1,2,3,4,5,6
17	wordlist	1,2,3,6
18	word	2,3,4,6
19	Spelling master	1,2,3,4,5,6

20	Post	1,2,3,4,5,6
21	Comment	1,2,3,4,6

Associate Noun and Verb identification

We will now identify the nouns and verbs associated with the potential classes to find out the attributes and methods of each

Class

Serial	Potential Class	Noun	Verb
1	account	Name, User name, Email, password	Create, Login, recover
2	game	Player, score	Playing Spelling bee, Playing Word battle, Playing Word copter, Pairing, Scoring,
3	leaderboard	User, player	Daily progress, Weekly progress
4	marketplace	Wordlist, Meme, tags	Download wordlist,view, Add tags
5	player	account	Play game
6	quiz	Wordlist, user, answer, score	Generate new quiz, Save quiz, Judge quiz, Quiz analysis
7	user	Account, Frequent,	Daily training, search words

		Infrequent, quiz	
8	Word	Spelling, meaning, multiple meaning, use in sentence, parts of speech, pronunciation	Search, input in app, long press in the dictionary, search on copy
9	wordlist	Name, user, Word, type	Memorize by repetitions, Save to list, training session, Share to market place,
10	Spelling master	Player, Meaning, Pronunciation, spelling	Launch game, Get word from dictionary, Show meaning, Show pronunciation, Check users answer, scoring
12	Post	User, text, timestamp, likes, comments	create post, edit post delete post, write comment, like post, unlike post

Identify Attributes

We have identified our potential classes, So, now we have to Identify the potential attributes of the potential classes

Serial	Potential Class	attributes
1	account	Name, User_name, Email, password
2	game	Player, Rounds, score
3	leaderboard	user, player
4	marketplace	Wordlist, user, Meme, tags
5	player	Account, game
6	quiz	Wordlist, user, answer, score
7	user	account, player, Frequent, Infrequent, Quiz, Wordlist,
8	Word	Spelling, meaning, multiple meaning, use in sentence, idioms, phrase
9	wordlist	ListName,

		user, Word, type
10	Spelling master	Player, Word,
11	post	creator, text, timestamp, likes, comments

Identify Methods

We have identified our potential classes, So, now we have
Identify the potential attributes of the potential methods

Serial	Class	Verb
1	account	createAccount(), Login(), recoverAccount()
3	leaderboard	getScore() displayRanking()
4	marketplace	EnrollInCourse(), PublishCourse(), UnPublishCourse()
5	player	PlayGame(), getScore();
6	quiz	GenerateNewQuiz(), JudgeQuiz(),
7	user	DailyTraining(), getQuizScore(). register(),

		takeQuiz(), Createlist();
8	Word	Savetolist(), selectMeaning(); selectAll(); viewWord() viewSynonyms()
9	wordlist	MemorizeByRepetition() TrainingSession(), publishToMarket(), generateQuiz(),
10	Spelling master	Launch game(), Gettingwordfromdictionary(), Showmeaning(), Showpronunciation(), Check users answer(), Scoring()
11	post	createPost() editPost() postComment() likePost() unlikePost()

Analysis

Here the player class and the user class are the same in this respect. Mainly, the user is a player. So, we can merge the classes. **We can merge player and user classes as user.**

And let us **rename the wordlist class to a course and post to forum post.**

There is no need for a separate Game class and Spelling Master class.

A word might have multiple meanings. And all the meanings will have their own example use etc. So, **we can split words into two classes: word and meaning.** Where words will be having a list of meanings. **Meaning will only have attributes.**

So, our final classes are

1. Account
2. User
3. MarketPlace
4. Quiz
5. Leaderboard
6. Word
7. Meaning
8. SubMeaning
9. WordList
10. SpellingMaster
11. ForumPost
12. DictionaryDatabaseHandler
13. FloatingBubbleHandler
14. Memorize
15. FireBaseBasicUtils
16. FireBaseNormalUtils
17. FireBaseAdvancedUtils
18. ForumComment
19. DailyTraining

Class cards

After identifying our final classes we have generated the following class cards:

Word	
Attributes	Methods
<ul style="list-style-type: none">• Meanings• Idioms• Phrases• Word body	ProcessWordFromDataBase() ProcessIdiomsAndPhrases() ProcessMeanings()
Responsibility	Collaborator
<ul style="list-style-type: none">• Hold information• One of the most valuable data entity• Hold information for the user to learn and practice	DictionaryDatabaseController Meaning

Meaning	
Attributes	Methods
<ul style="list-style-type: none">• PartsOfSpeech• MeaningBody• Synonyms• MoreExamples• MeaningID• SubMeanings	ProcessMeaning() ProcessSynonyms() ProcessMoreExamples() ProcessSubMeanings()
Responsibility	Collaborator
<ul style="list-style-type: none">• Hold information regarding a section of a word• Help complete the necessary information for a word	DictionaryDatabaseController SubMeaning

SubMeaning	
Attributes	Methods
<ul style="list-style-type: none">• SubMeaningBody• Examples	LoadSubmeanings LoadExamples
Responsibility	Collaborator
<ul style="list-style-type: none">•	DictionaryDatabaseController

DictionaryDatabaseController	
Attributes	Methods
	OpenDatabase LoadSynonymsAsStrings LoadIdioms LoadPhrases LoadExamples LoadMoreExamples LoadAllWords
Responsibility	Collaborator
<ul style="list-style-type: none"> Read raw data from local database Help create sub meaning, meaning, and word. 	

FloatingBubbleHandler	
Attributes	Methods
<ul style="list-style-type: none"> CopiedWord LemmatizedWord 	ListenToNewCopy CheckForValidInput LemmatizeWord SearchWord
Responsibility	Collaborator
<ul style="list-style-type: none"> Wait for user to copy a valid input Query word when user demands Show meaning in floating window 	Word Meaning SubMeaning

WordList	
Attributes	Methods
<ul style="list-style-type: none"> Name Description ListID SubMeanings 	DeleteWordList AddSubMeaning SearchSubMeaning DeleteSubMeaning ShuffleSubMeanings PublishToMarket

Responsibility	Collaborator
<ul style="list-style-type: none"> • Store submeanings that interest the user • Trigger update to the cloud storage 	SubMeanings FireBaseAdvancedUtils FireBaseNormalUtils

Marketplace	
Attributes	Methods
<ul style="list-style-type: none"> • 	EnrollInCourse UnPublishCourse LoadUserContribution
Responsibility	Collaborator
<ul style="list-style-type: none"> • Let user enroll in course • Let user unpublish course • Display courses that the user has created • Trigger update to cloud storage 	FireBaseAdvancedUtils FireBaseNormalUtils

SpellingMaster	
Attributes	Methods
<ul style="list-style-type: none"> • TotalQuestionsAttempted • TotalCorrectAnswers 	LoadPronunciation CheckAnswer UpdateScore
Responsibility	Collaborator
<ul style="list-style-type: none"> • Help user learn and practice spelling • Check user answer • Calculate user score • Launch game, • Show word details 	Word SubMeaning WordList

Memorize	
Attributes	Methods
<ul style="list-style-type: none"> • TotalQuestionsAttempted • MistakenSubMeanings 	CheckAnswer StartNextRound
Responsibility	Collaborator

<ul style="list-style-type: none"> • Help user revise and memorize meaning • Check user answer • Start next round with words left to memorize 	Word Meaning SubMeaning WordList
--	---

Quiz	
Attributes	Methods
<ul style="list-style-type: none"> • TotalQuestionsAttempted • TotalCorrectAnswers 	GenerateOptions CheckAnswer UpdateMarks
Responsibility	Collaborator
<ul style="list-style-type: none"> • Present question to user • Evaluate answer • Calculate score 	Word Meaning SubMeaning WordList

FirebaseBasicUtils	
Attributes	Methods
<ul style="list-style-type: none"> • FirebaseAuth • FirebaseFirestore 	UpdateDataToCloud WriteDataToCloud LoadDataFromCloud DeleteDataFromCloud TriggerCloudFunction
Responsibility	Collaborator
<ul style="list-style-type: none"> • Perform basic CRUD operation to cloud storage 	

FirebaseNormalUtils	
Attributes	Methods
<ul style="list-style-type: none"> • FirebaseBasicUtils 	LoadUserData LoadWordListsRecords LoadCoursesRecords DeleteWordList CreateWordList DeleteSubMeaningFromWordList PublishCourseToMarket EnrollInCourse

	UnPublishCourse updateRatingAndScore ForumPostUpdate ForumPostRead
Responsibility	Collaborator
<ul style="list-style-type: none"> Perform normal CRUD operation to cloud storage 	<ul style="list-style-type: none"> FireBaseBasicUtils

FireBaseAdvancedUtils	
Attributes	Methods
<ul style="list-style-type: none"> FireBaseNormalUtils 	PrepareWordListFromRecords PrepareDailyTrainingSequence PrepareCourseFromRecords PrepareLeaderboardStandingsFromRecords UpdatePost PreparePostFromRecords AddSubMeaningToWordList
Responsibility	Collaborator
<ul style="list-style-type: none"> Perform more complex CRUD operations to cloud 	<ul style="list-style-type: none"> FireBaseNormalUtils

Leaderborad	
Attributes	Methods
<ul style="list-style-type: none"> UserPosition UserScore 	getLeaderBoardStandings getUserPosition
Responsibility	Collaborator
<ul style="list-style-type: none"> Fetch leaderboard positions 	FireBaseAdvancedUtils

User	
Attributes	Methods
<ul style="list-style-type: none"> Account TrainingProgress CoursesCreated CoursesEnrolled WordLists 	UnenrollFromCourse CreateWordList
Responsibility	Collaborator
<ul style="list-style-type: none"> Hold user information Trigger update to cloud storage 	FireBaseAdvancedUtils FireBaseMediumUtils Account

Account	
Attributes	Methods
<ul style="list-style-type: none"> Email Name Password UserName 	LogIn SignUp RecoverPassword
Responsibility	Collaborator
<ul style="list-style-type: none"> Provide authentication services to user 	

ForumPost	
Attributes	methods
<ul style="list-style-type: none"> creator, text, timestamp, likes, comments 	createPost() editPost() postComment() likePost() unlikePost()
Responsibility	Collaborator
<ul style="list-style-type: none"> create post edit post delete post post comment like post 	FirebaseAdvancedUtils

ForumComment	
Attributes	Methods
<ul style="list-style-type: none"> creator, text, timestamp 	
Responsibility	Collaborator
<ul style="list-style-type: none"> Store information about comments 	ForumPost

DailyTraining	
Attributes	Methods
<ul style="list-style-type: none"> TrainingSequence 	LoadTrainingSequence UpdateRating

	EvaluateResult TrackTrainingProgress
Responsibility	Collaborator
●	User FireBaseAdvancedutils FireBaseNormalUtils

Class Modeling Diagram

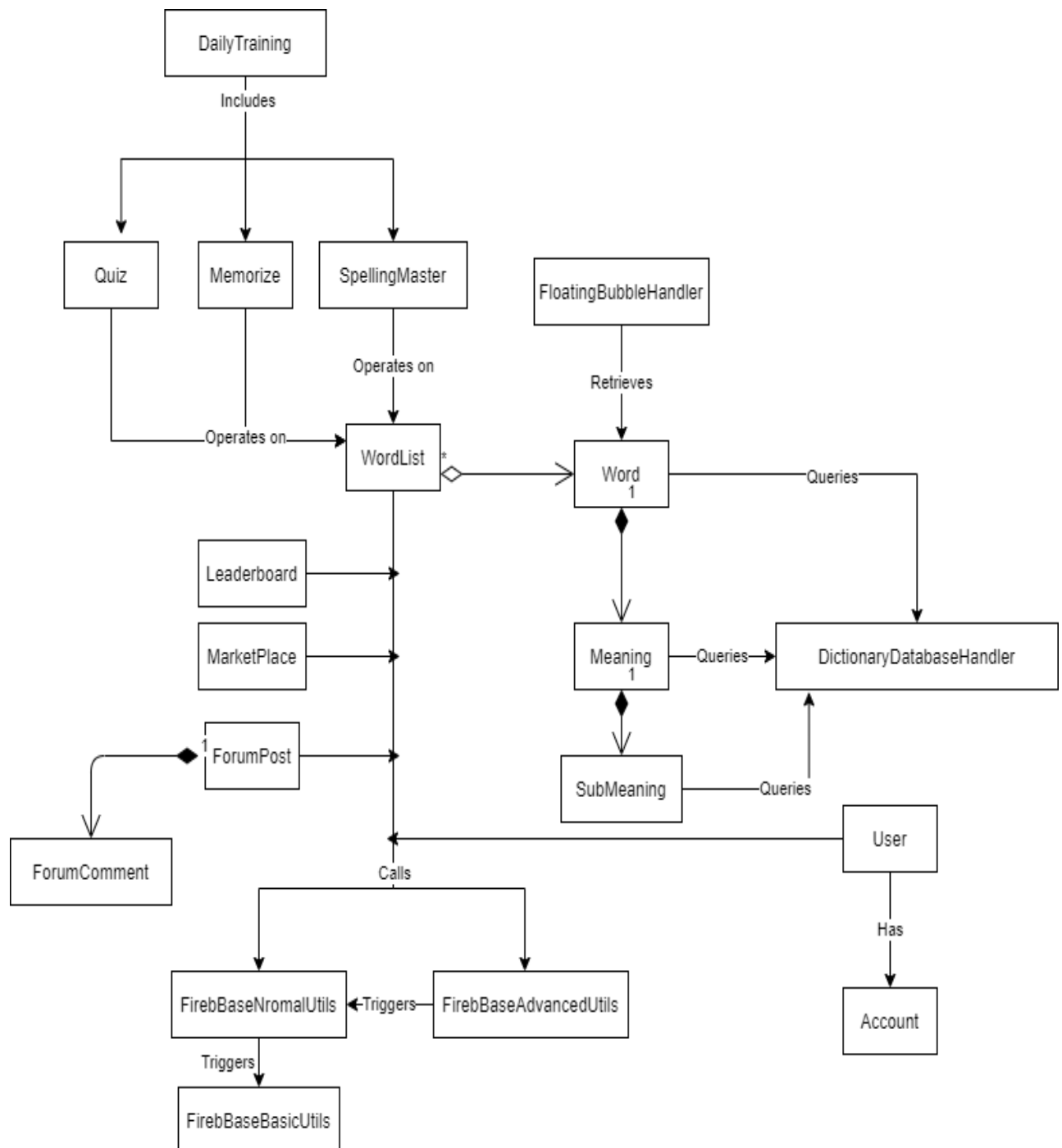


Fig: Class modeling diagram

BEHAVIORAL MODELING OF WORD MASTER 3000

STATE TRANSITION DIAGRAM

State diagram represents active states for each class the events (triggers). For this we identified all the events, their initiators and collaborators.

Serial	Event	initiator	collaborator	State name
1	Aim to help			
2	Spending time	User		
3	Create account	User	User	create Account
4	Access all feature	User	User	
5	To provide info	user	Account	Provide info
6	Login	user	Account	Login
7	Recover id	User	Account	Recover
8	Show word details	User	Word, meaning	Show word details
9	Select specific meaning	User	Word, meaning	Select specific meaning
10	Select all meanings	User	Word, meaning	Select All
11	Group all meanings to get all	System	Word, system meaning	Group all meanings
12	Search words	user	Word	search
13	Input in app	user	System	Input In App,
14	Search on copy	user	System	Search on Copy
15	Create list	user	Word, Course	Create list

16	Store words	User	Course, words, firebase, system	Store word
17	Share list	User,	Marketplace, course, firebase	Share list
18	Make list public	User,	Marketplace, course, firebase	Make list public
19	Memorize words	User,	Course, firebase, meaning	Memorize by repetitions
20	Take part in quizzes	user	Quiz, firebase	Take part in quizzes
21	Create new quiz	User	Quiz, firebase	Create new quiz
22	Smart suggestions	Quiz,	firebase, Course	Smart suggestions
23	Generate short training session	Firebase, course	System,	Generate short training session
24	Play game	User,	Game, Firebase	Play game
25	Launch game	Game	Firebase	Launch game
26	Show meaning and pronunciation	Spelling bee	Game, Firebase, meaning	Show meaning and pronunciation
27	Match the answer of the user	Spelling bee	Game, Firebase, meaning	Match the answer of the user
28	Display question	Quiz,	Firebase, system	Display question
29	Get first one answered			
30	Scoring	Daily Training	Firebase, User	Scoring
31	Generate certain question	Firebase	Course	Generate certain question

32	Show questions to user	Game, quiz	User, firebase	Show questions to user
33	Generate leaderboard	Firebase	User	Generate leaderboard
34	Access market place	User	Marketplace	Access market place
35	Download word list	User	Course	Download word list
36	Delete meme	user	meme	Delete meme
37	Like meme	User	Meme	Like meme
38	Daily training	User	FirestoreAdvancedUtils FirestoreNormalUtils WordList Words	Daily Training
39	Post in forum	ForumPost	User FirestoreAdvancedUtils	Posting in forum
40	Commenting in forum	ForumPost	User FirestoreAdvancedUtils	Commenting in forum
41	Liking posts	ForumPost	User FirestoreAdvancedUtils	Liking posts

Events after analysis:

After some analysis, we can merge some events and states which are of same types. The analyzed data are given below:

serial	event	initiator	collaborator	State name
1.	Create account	User	User	Create account
2.	To provide info	user	Account	To provide info

3.	Login	user	Account	Login
4.	Recover id	User	Account	Recover id
5.	Show word details	User	Word, meaning	Show word details
6.	Select specific meaning	User	Word, meaning	Select specific meaning
7.	Select all meanings	User	Word, meaning	Select all meanings
8.	Group all meanings to get all	System	Word, system meaning	Group all meanings to get all
9.	Search words	user	Word	Search words
10.	Input in app	user	System	Input in app
11.	Search on copy	user	System	Search on copy
12.	Create list	user	Word, Course	Create list
13.	Store words	User	Course, words, firebase, system	Store words
14.	Share list	User,	Marketplace, course, firebase	Share list
15.	Make list public	User,	Marketplace, course, firebase	Make list public
16.	Memorize words	User,	Course, firebase, meaning	Memorize words
17.	Take part in quizzes	user	Quiz, firebase	Take part in quizzes
18.	Create new quiz	User	Quiz, firebase	Create new quiz
19.				
20.	Smart suggestions	Quiz,	firebase, Course	Smart suggestions
21.	Generate short training session	firebase	System,	Generate short training session
22.	Play game	User,	Game, Firebase	Play game
23.	Launch game	Game	Firebase	Launch game

24.	Show meaning and pronunciation	Spelling bee	Game, Firebase, meaning	Show meaning and pronunciation
25.	Match the answer of the user	Spelling bee	Game, Firebase, meaning	Match the answer of the user
26.	Display question	Quiz,	Firebase, system	Display question
27.	Scoring	Game	Firebase, User	Scoring
28.	Generate certain question	Firebase	Course	Generate certain question
29.	Show questions to user	Game, quiz	User, firebase	Show questions to user
30.	Get points from games and quizzes	Game, quiz	user	Get points from games and quizzes
31.	Generate leaderboard	Firebase, leaderBoard	User,	Generate leaderboard
32.	Access market place	User	MarketPlace	Access market place
33.	Download word list	User	Course	Download word list
34	Daily training	User	FirebaseAdvancedUtils FirebaseNormalUtils WordList Words	Daily Training
35	Post in forum	ForumPost	User FireBaseAdvancedUtils	Posting in forum
36	Commenting in forum	ForumPost	User FireBaseAdvancedUtils	Commenting in forum
37.	Liking posts	ForumPost	User FireBaseAdvancedUtils	Liking posts

State Transition Diagram: User

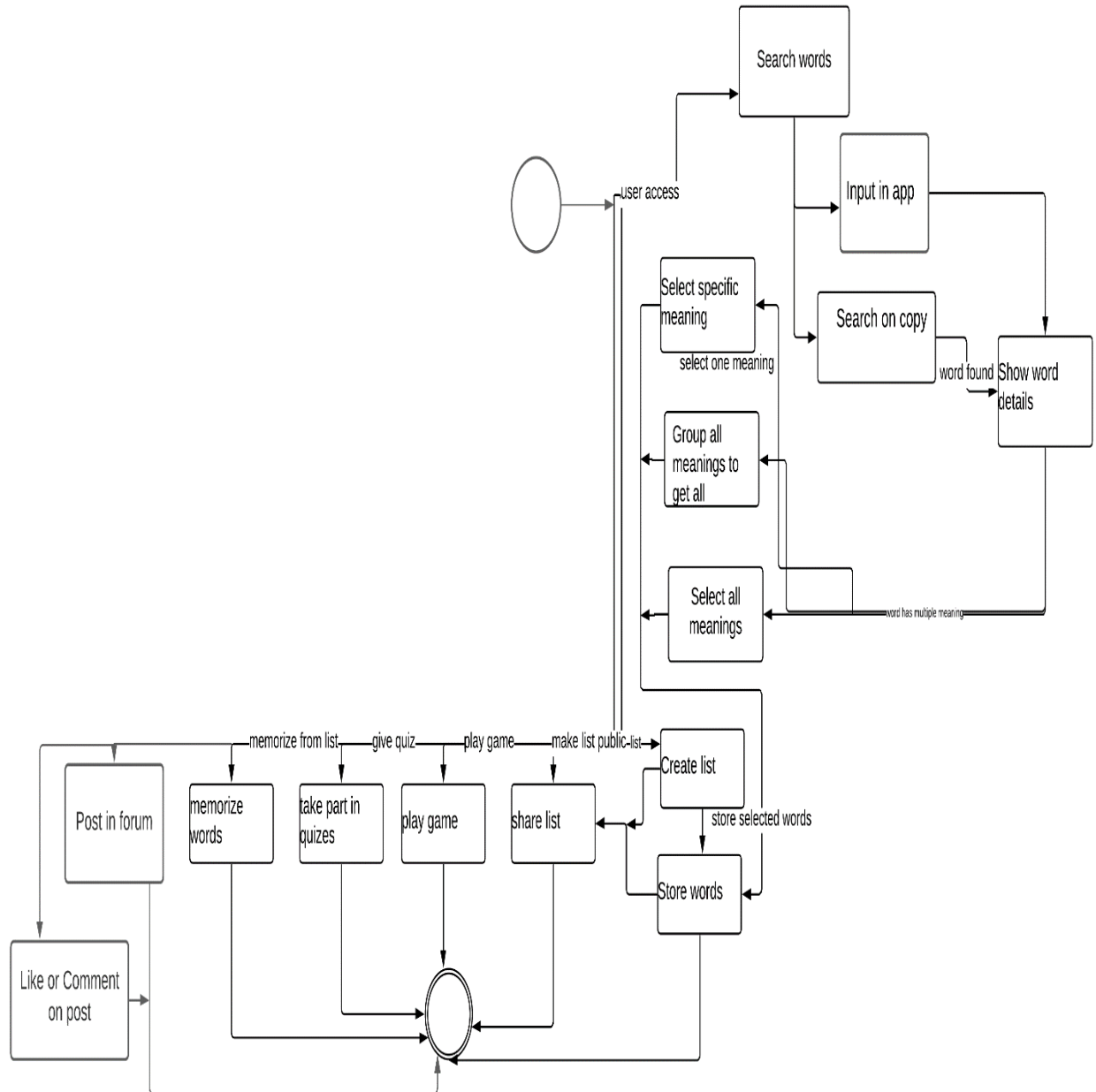


Fig: User State Transition Diagram

Description:

The diagram mainly shows the state of the user class of our application. The account class is also associated with it.

Word

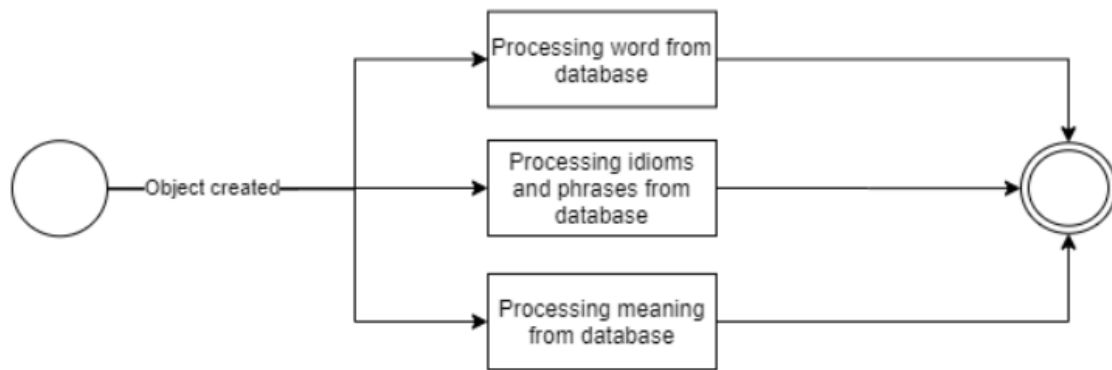


Fig: Word class state transition diagram

Description:

It mainly shows us how a word will be selected and other relating state with word class.

Account

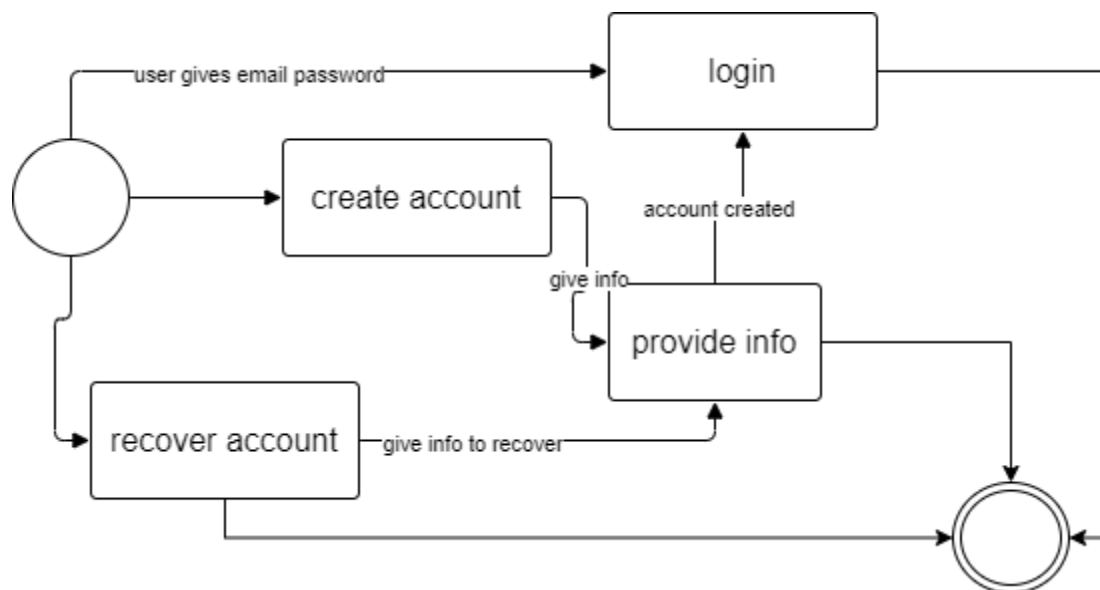
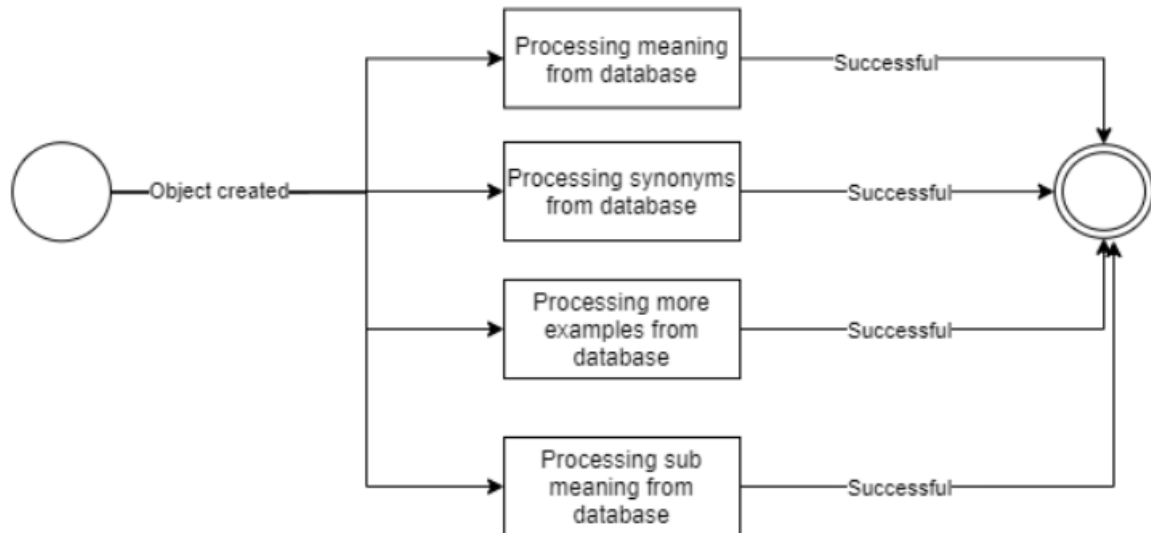
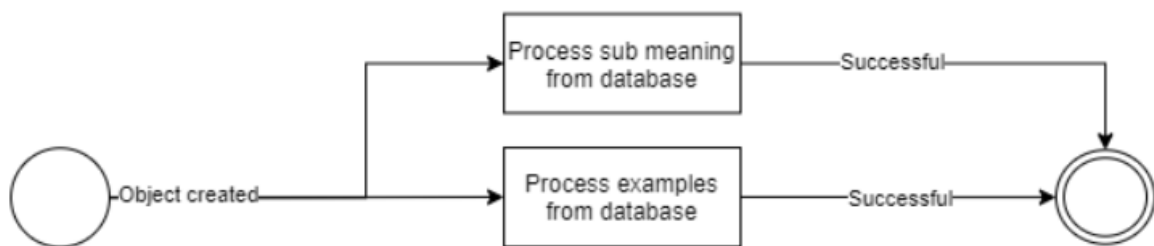


Fig: Account class state transition diagram

Meaning



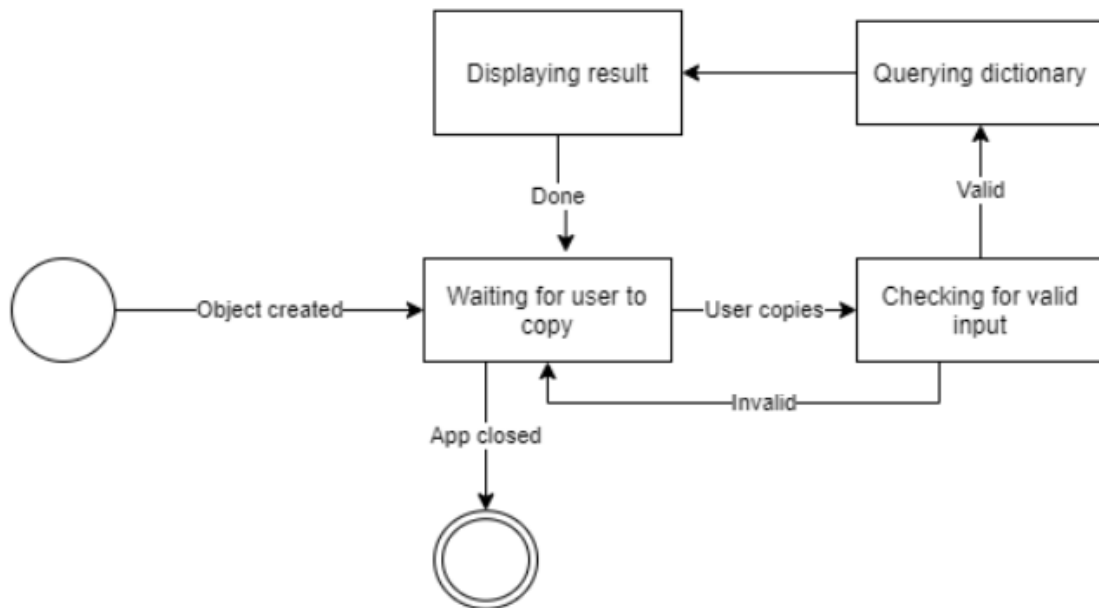
SubMeaning



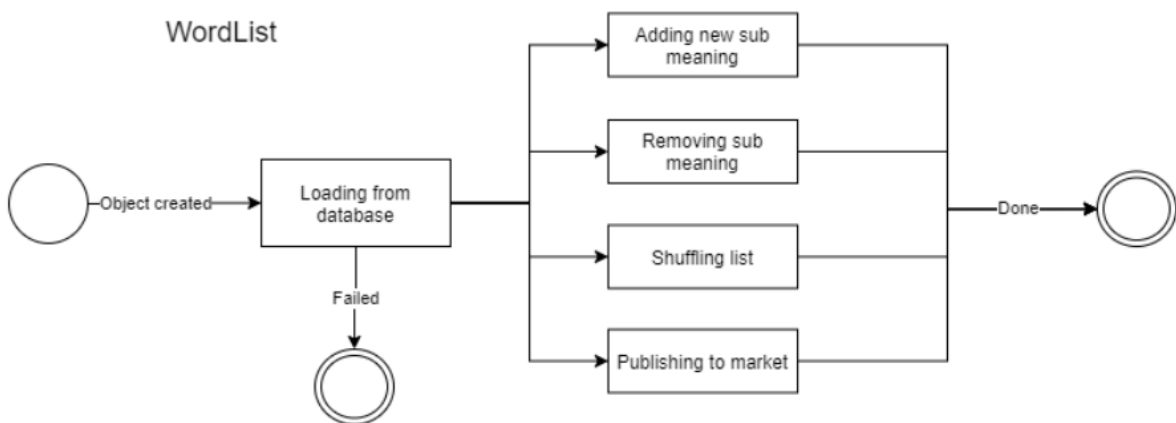
Dictionary database controller



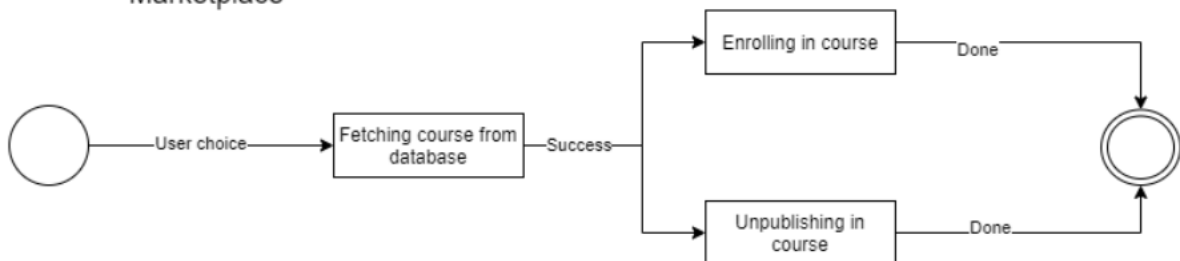
Floating Bubble Handler



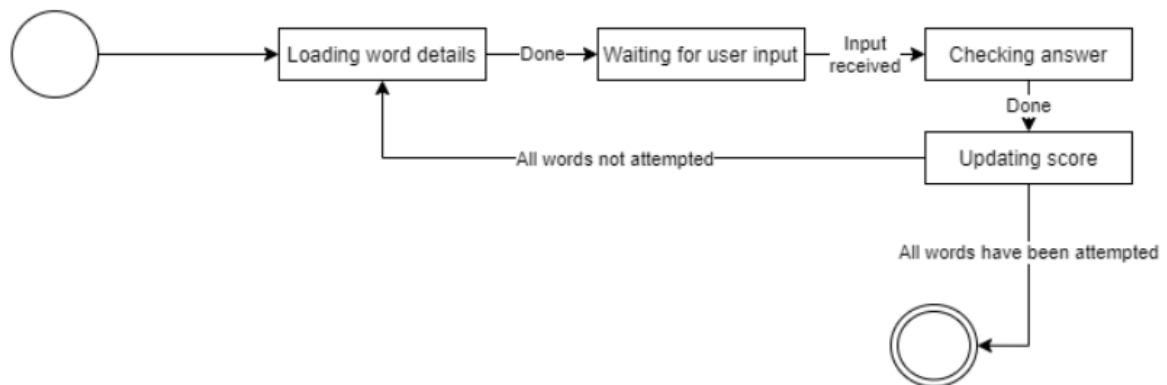
WordList



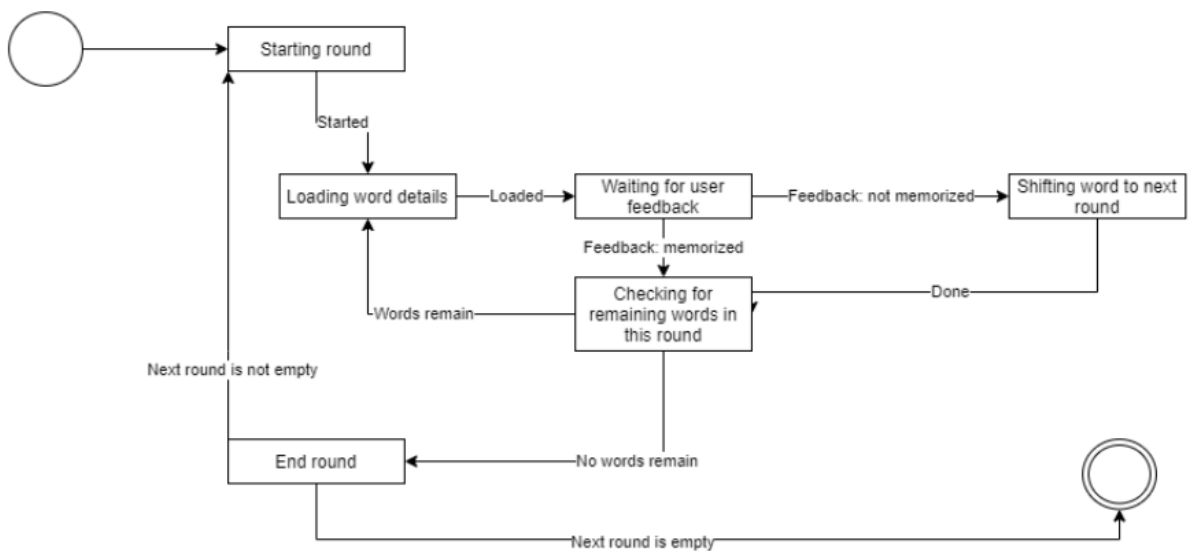
Marketplace



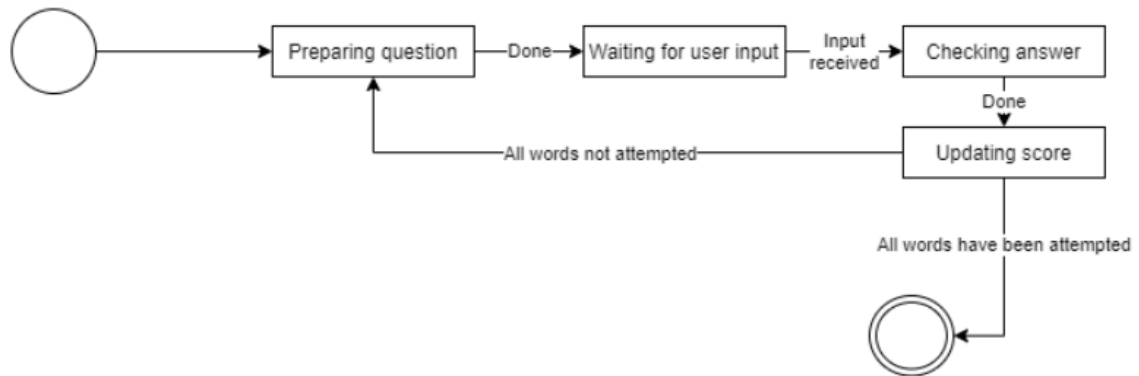
SpellingMaster



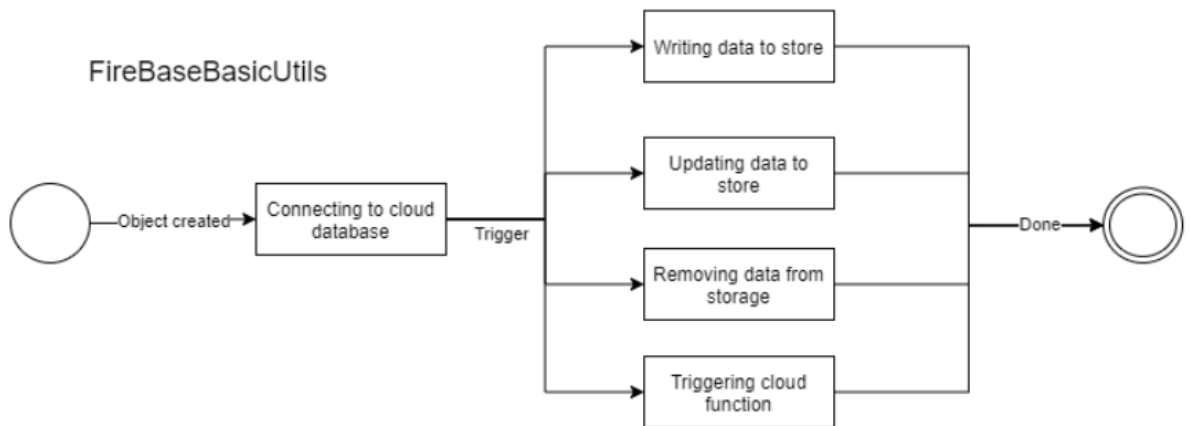
Memorize



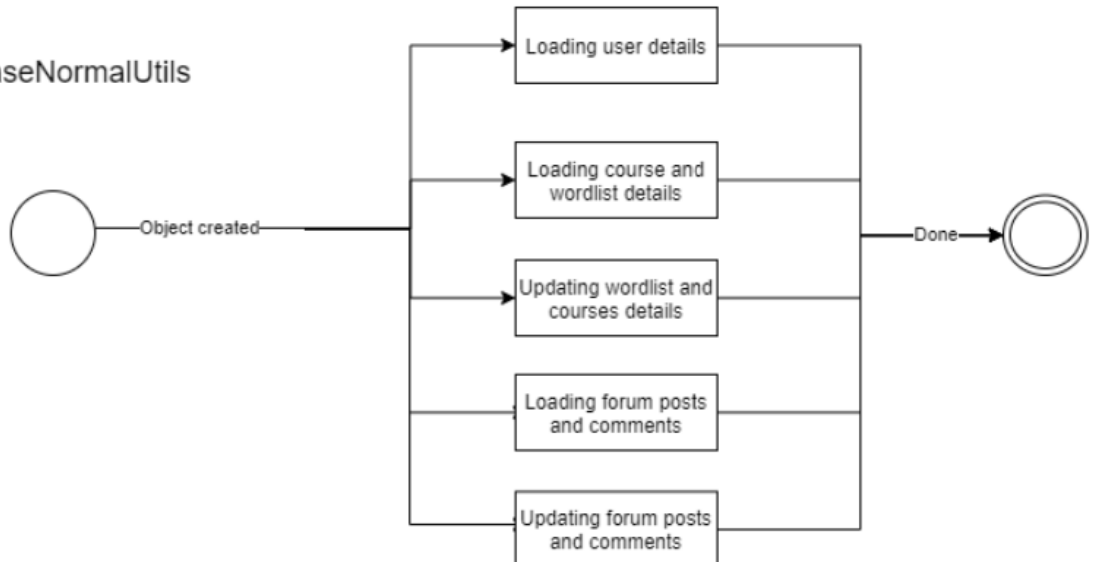
Quiz



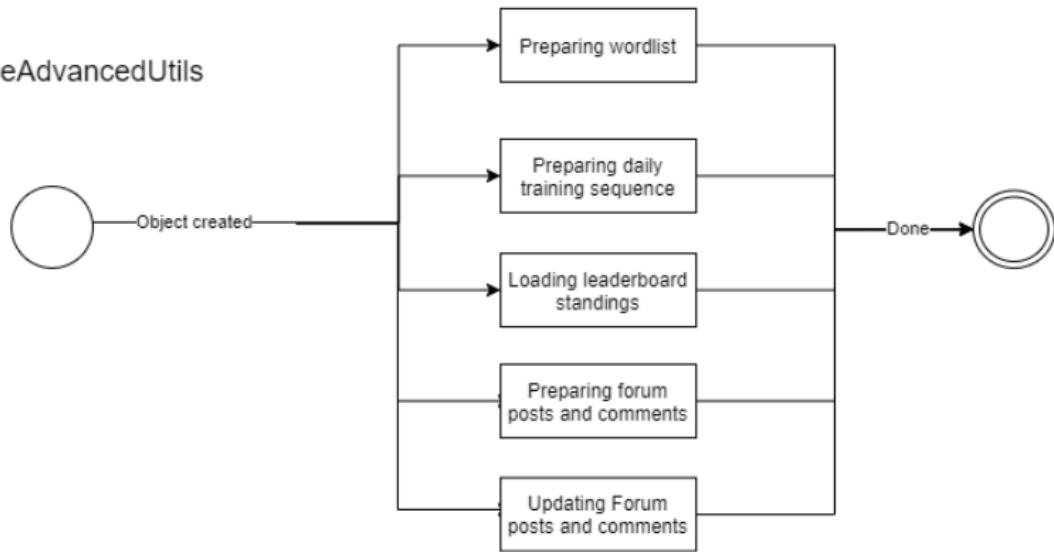
FireBaseBasicUtils



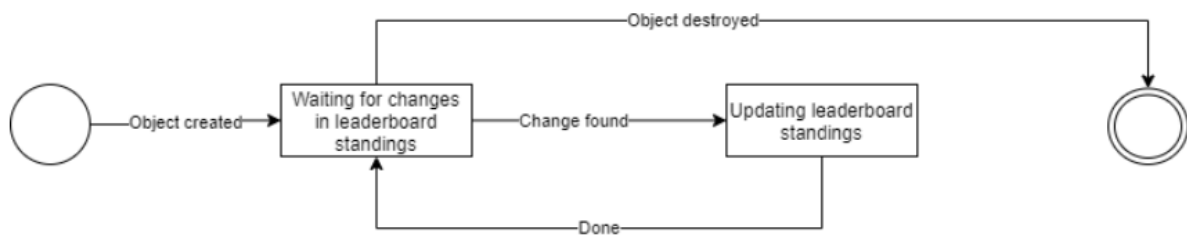
FireBaseNormalUtils



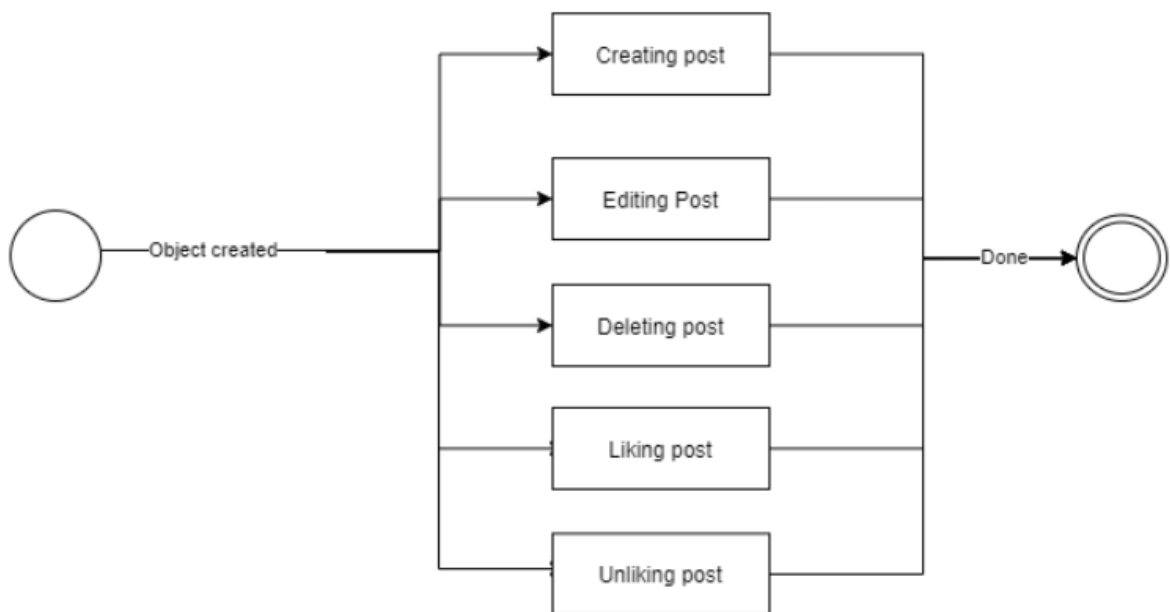
FireBaseAdvancedUtils



Leaderboard



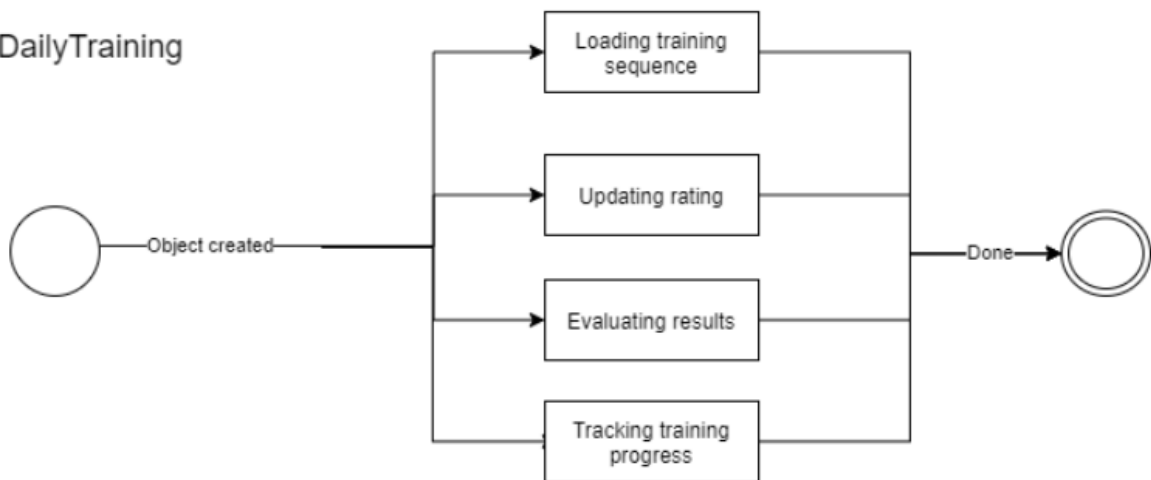
ForumPost



ForumComment



DailyTraining



Sequence Diagram

