CPE343 Object-oriented Analysis and Design

Project Design Booklet

Topic: Movie Review Social Network

Team

SunnySideUp

Member

1. Mr.Jarudet Witchit 59070501008

2. Mr.Nawakanok Muangkham 59070501044

Present

Dr. Sally E. Goldin

Semester 2/2018

King Mongkut's University of Technology Thonburi

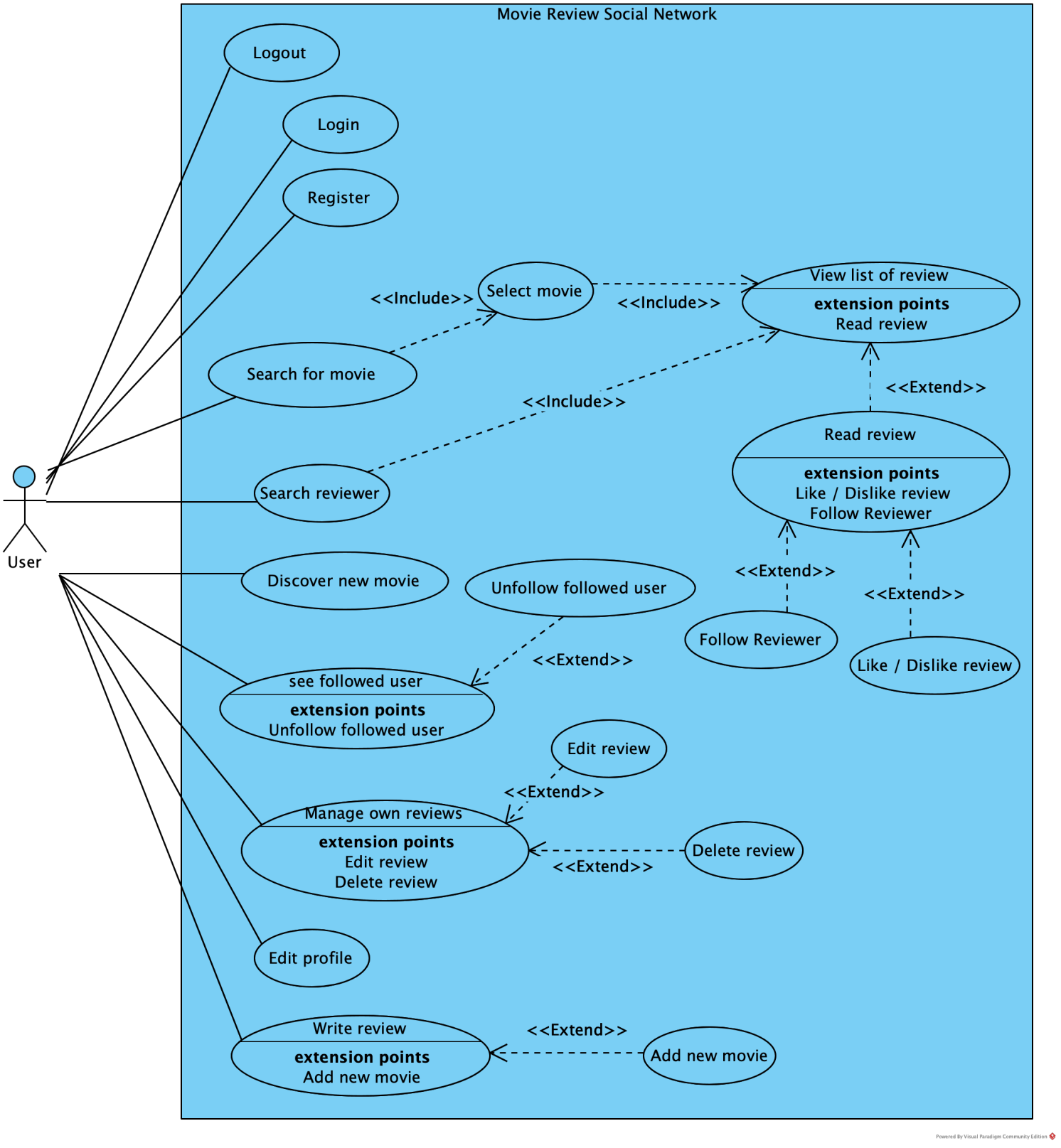
MRSN Description

Movie Review Social Network (MRSN) is a java program that imitated common movie review website you can find on the internet. Though text based, MRSN can be used to read review, search, write review, follow reviewer, manage review, register, save to file, add new movie, and discover new movie you might like through our simple random movie function.

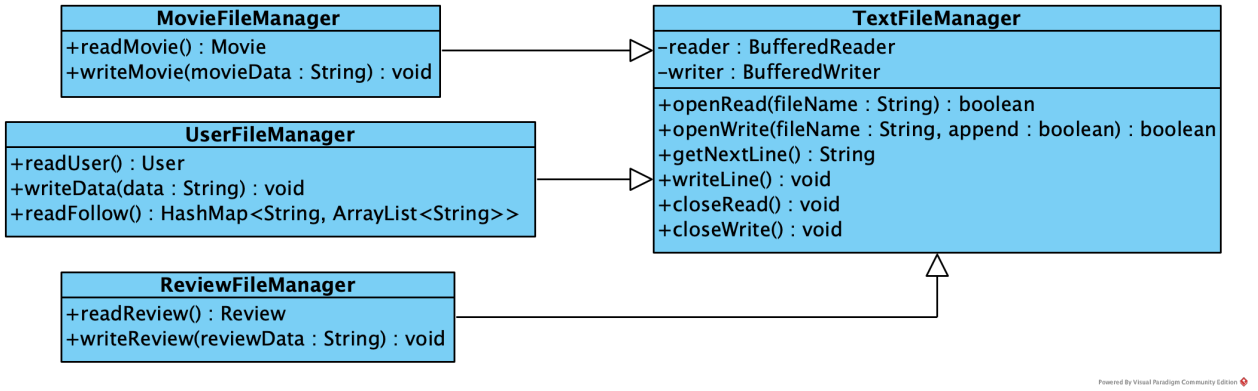
As I said MRSN is text based so it will work on terminal only. When search if there are too many search results MRSN will split search results into 5 results per page. Then, user can navigate through each page with easy command. MRSN is state program; output shown to user will varies for each state. User's action will change MRSN's state according to state diagram. MRSN provides ultra-encapsulation of data each change in data class will be done through each class managers. Each change in data will be later saved by file managers. User can search for movie by category or name but input to search must almost exactly match with its name or category. Also, user can search for review using reviewer name. Most of the data in MRSN can be edited by user. Finally, MRSN has a very big façade class which handle almost every input and output.

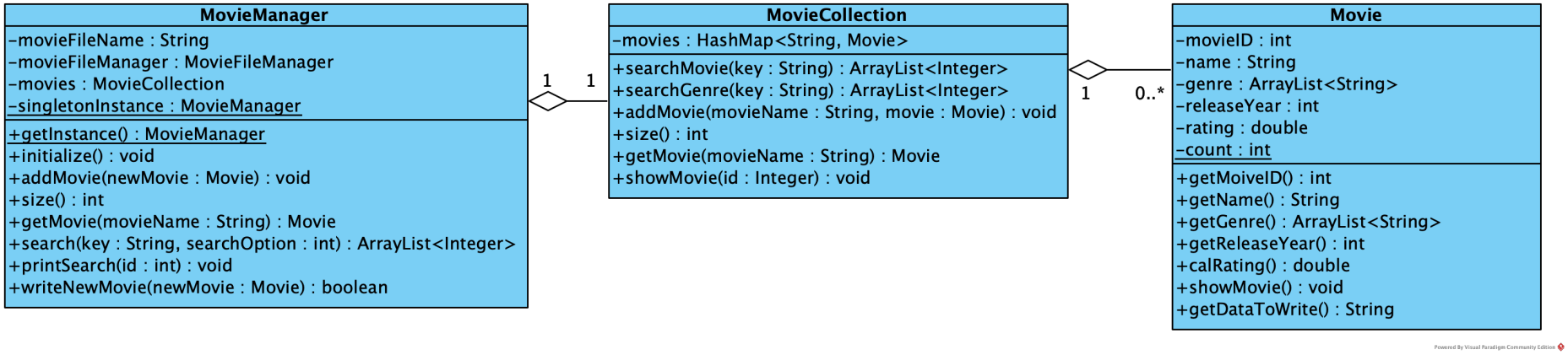
Despite all said MRSN was down scoped from the requirement version. In this version MRSN don't have movie tag, user can't sort movie, only menu will show up at main page, the program doesn't have complex suggestion system, and every part that shown list of something without user asking for it are cut out.

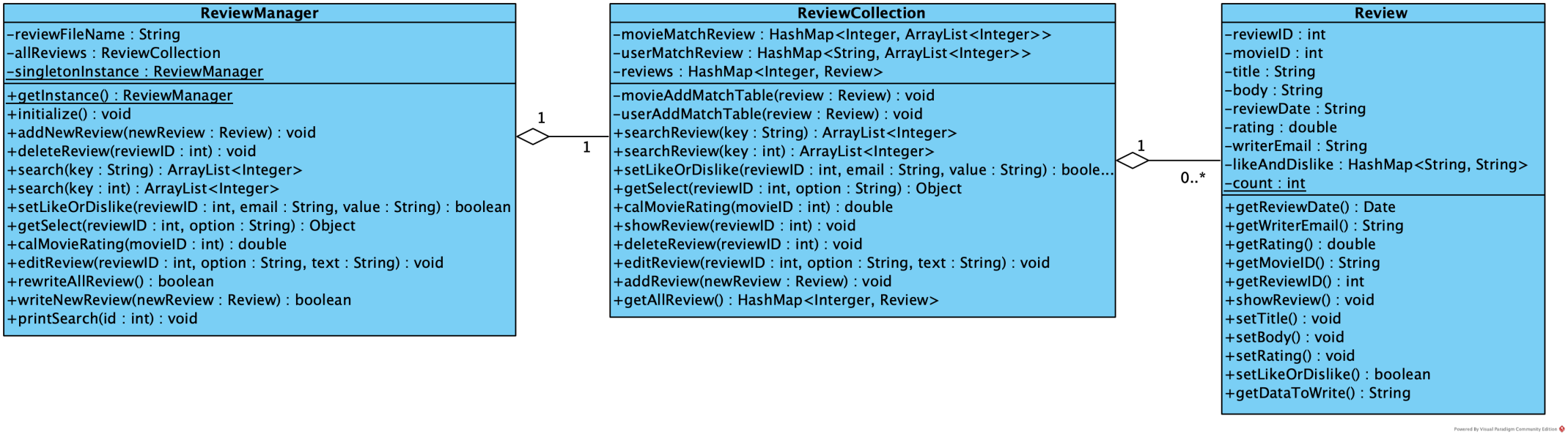
Use case diagram

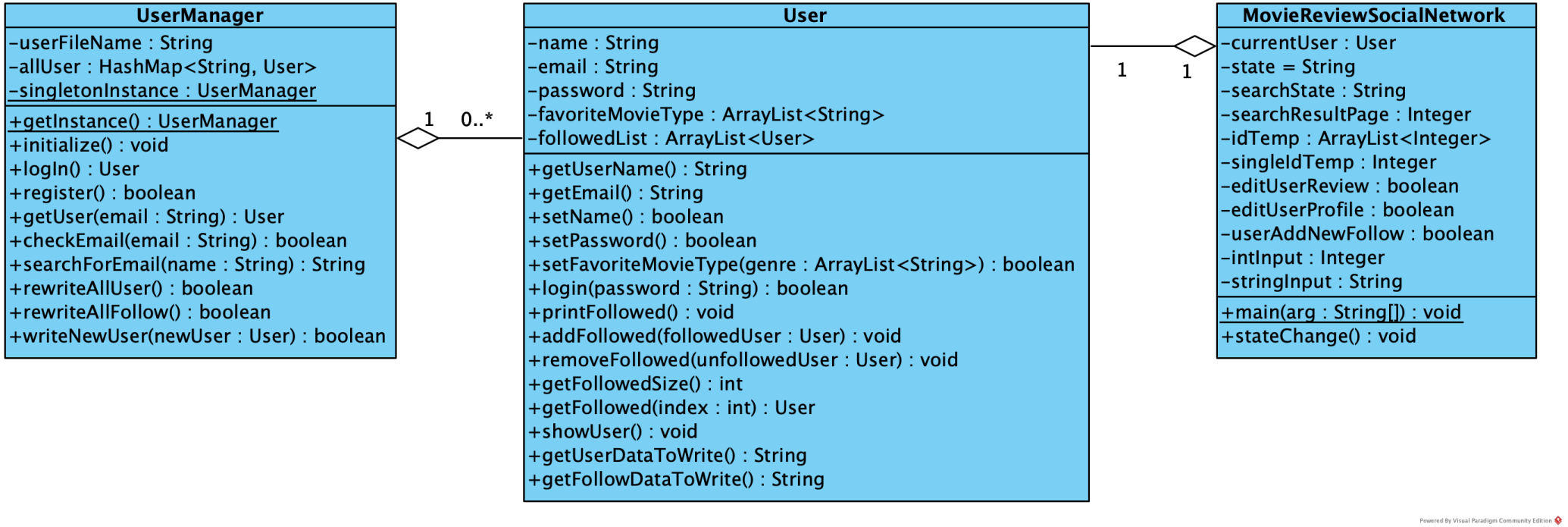


Class diagram






ReviewManager
-reviewFileName : String
-allReviews : ReviewCollection
-singletonInstance : ReviewManager
+getInstance() : ReviewManager
+initialize() : void
+addNewReview(newReview : Review) : void
+deleteReview(reviewID : int) : void
+search(key : String) : ArrayList<Integer>
+search(key : int) : ArrayList<Integer>
+setLikeOrDislike(reviewID : int, email : String, value : String) : boolean
+getSelect(reviewID : int, option : String) : Object
+calMovieRating(movieID : int) : double
+editReview(reviewID : int, option : String, text : String) : void
+rewriteAllReview() : boolean
+writeNewReview(newReview : Review) : boolean
+printSearch(id : int) : void
1
1
ReviewCollection
-movieMatchReview : HashMap<Integer, ArrayList<Integer>>
-userMatchReview : HashMap<String, ArrayList<Integer>>
-reviews : HashMap<Integer, Review>
-movieAddMatchTable(review : Review) : void
-userAddMatchTable(review : Review) : void
+searchReview(key : String) : ArrayList<Integer>
+searchReview(key : int) : ArrayList<Integer>
+setLikeOrDislike(reviewID : int, email : String, value : String) : boole...
+getSelect(reviewID : int, option : String) : Object
+calMovieRating(movieID : int) : double
+showReview(reviewID : int) : void
+deleteReview(reviewID : int) : void
+editReview(reviewID : int, option : String, text : String) : void
+addReview(newReview : Review) : void
+getAllReview() : HashMap<Interger, Review>
1
0..*
Review
-reviewID : int
-movieID : int
-title : String
-body : String
-reviewDate : String
-rating : double
-writerEmail : String
-likeAndDislike : HashMap<String, String>
-count : int
+getReviewDate() : Date
+getWriterEmail() : String
+getRating() : double
+getMovieID() : String
+getReviewID() : int
+showReview() : void
+setTitle() : void
+setBody() : void
+setRating() : void
+setLikeOrDislike() : boolean
+getDataToWrite() : String
Powered By Visual Paradigm Community Edition

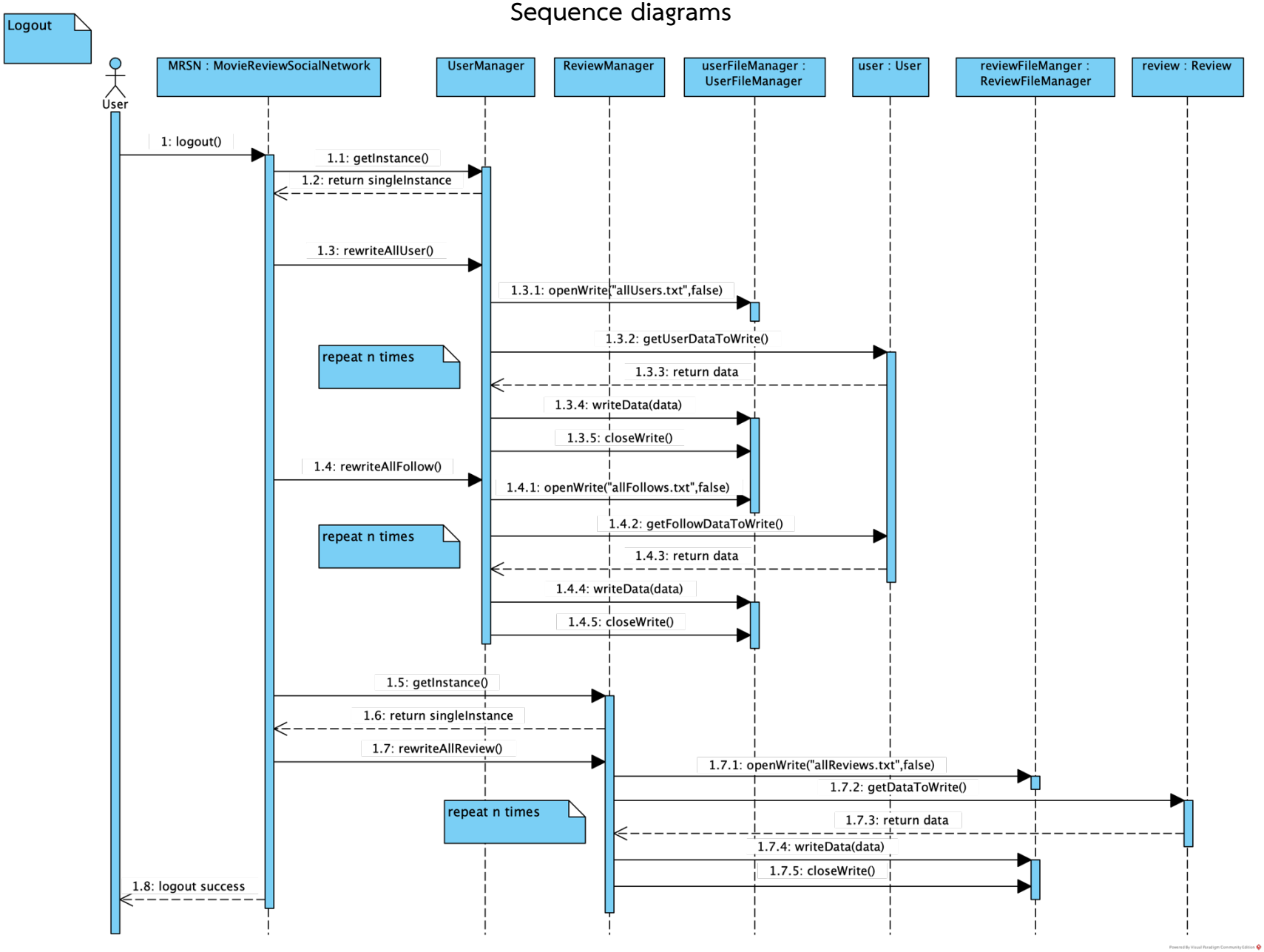

UserManager
-userFileName : String
-allUser : HashMap<String, User>
-singletonInstance : UserManager
+getInstance() : UserManager
+initialize() : void
+logIn() : User
+register() : boolean
+getUser(email : String) : User
+checkEmail(email : String) : boolean
+searchForEmail(name : String) : String
+rewriteAllUser() : boolean
+rewriteAllFollow() : boolean
+writeNewUser(newUser : User) : boolean
1
0..*
User
-name : String
-email : String
-password : String
-favoriteMovieType : ArrayList<String>
-followedList : ArrayList<User>
+getUserName() : String
+getEmail() : String
+setName() : boolean
+setPassword() : boolean
+setFavoriteMovieType(genre : ArrayList<String>) : boolean
+login(password : String) : boolean
+printFollowed() : void
+addFollowed(followedUser : User) : void
+removeFollowed(unfollowedUser : User) : void
+getFollowedSize() : int
+getFollowed(index : int) : User
+showUser() : void
+getUserDataToWrite() : String
+getFollowDataToWrite() : String
1
1
MovieReviewSocialNetwork
-currentUser : User
-state = String
-searchState : String
-searchResultPage : Integer
-idTemp : ArrayList<Integer>
-singleIdTemp : Integer
-editUserReview : boolean
-editUserProfile : boolean
-userAddNewFollow : boolean
-intInput : Integer
-stringInput : String
+main(arg : String[]) : void
+stateChange() : void
Powered By Visual Paradigm Community Edition

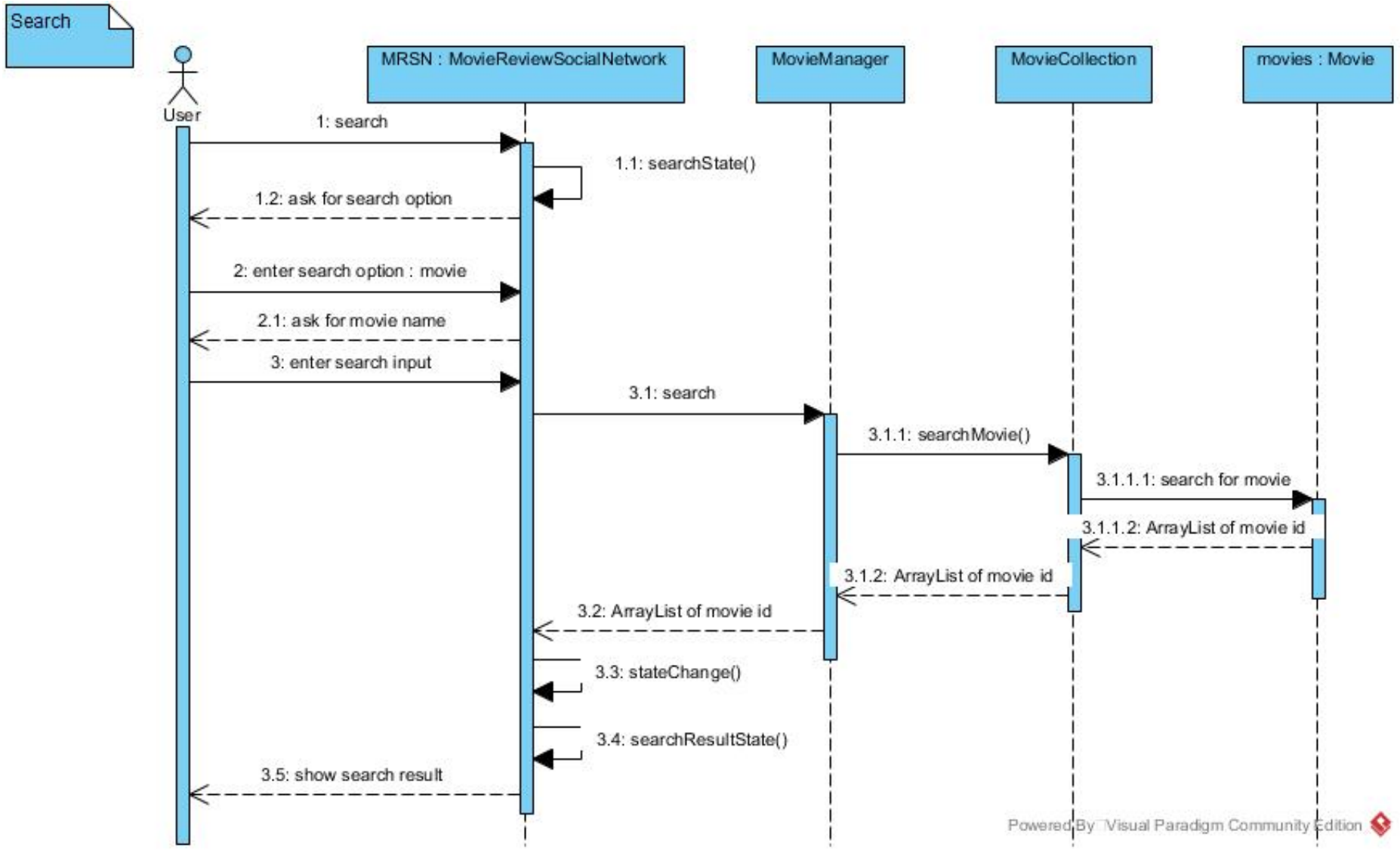
Class description table

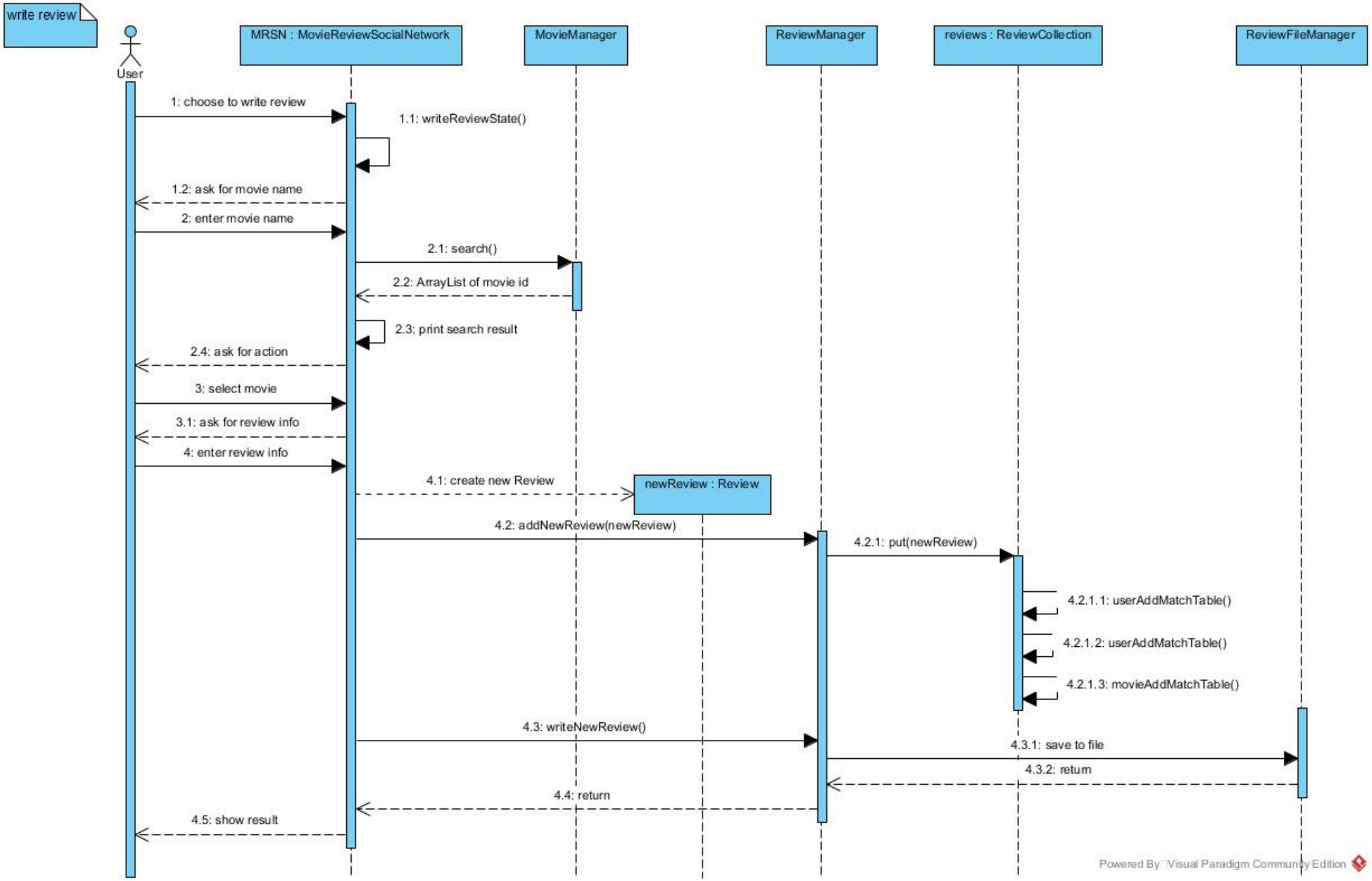
Class	Primary responsibility	Relationship
MovieReviewSocialNetwork	façade	Use every other classes.
TextFileManager	Read/write text file	Is superclass of MovieFileManager, ReviewFileManager, and UserFileManager.
MovieFileManager	Read/Create Movie from file	Subclass of TextFileManager. Used to read/save file by MovieManager.
ReviewFileManager	Read/Create Review from file	Subclass of TextFileManager. Used to read/save file by ReviewManager.
UserFileManager	Read/Create User from file	Subclass of TextFileManager/ Used to read/save file by UserManager.
Movie	Represent movie	Is kept by MovieCollection. Is managed by MovieManager.
MovieCollection	Keep all Movie	Kept all Movie. Is managed by MovieManager.
MovieManager	Manage Movie	Manage Movie and MovieCollection. Use MovieFileManager to get Movie from file.
Review	Represent review of Movie	Is kept by ReviewCollection. Is managed by ReviewManager.
ReviewCollection	Keep all Review	Kept all Review. Is managed by ReviewManager.
ReviewManager	Manage Review	Manage Review and ReviewCollection. Use ReviewFileManager to get Review from file.

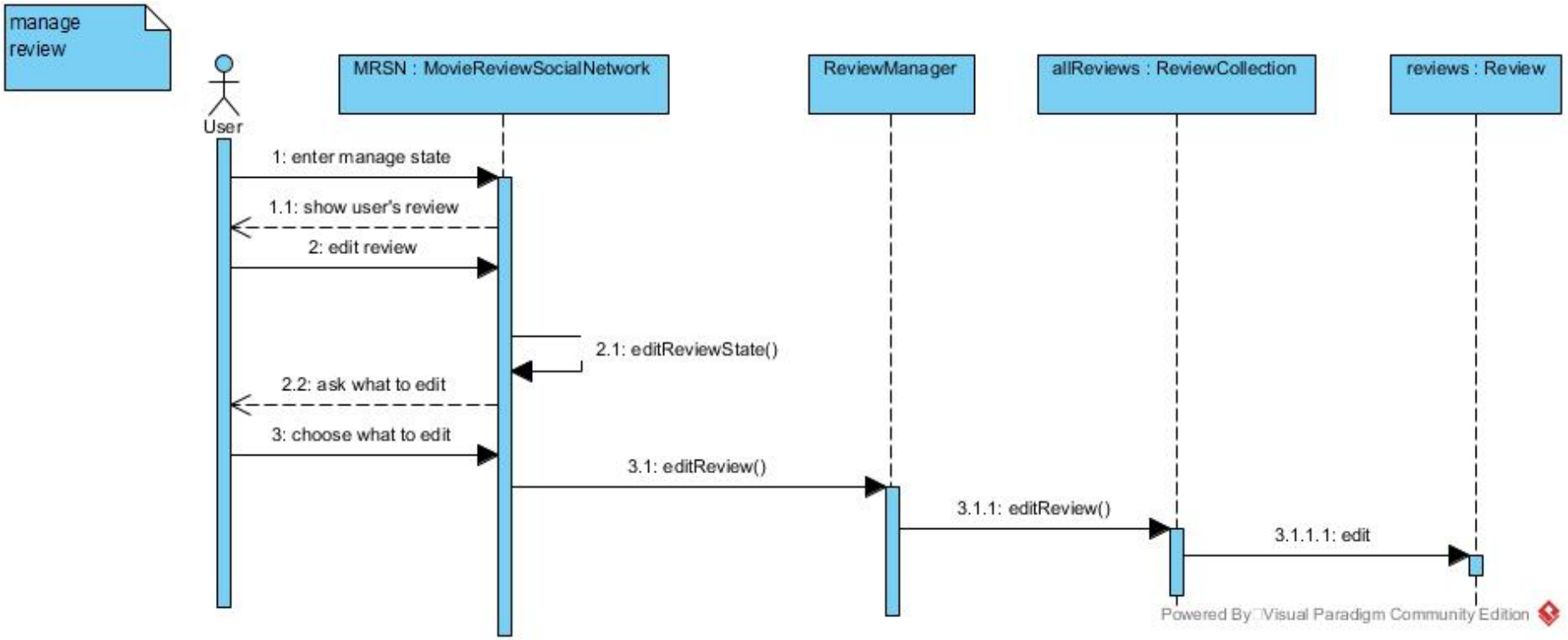
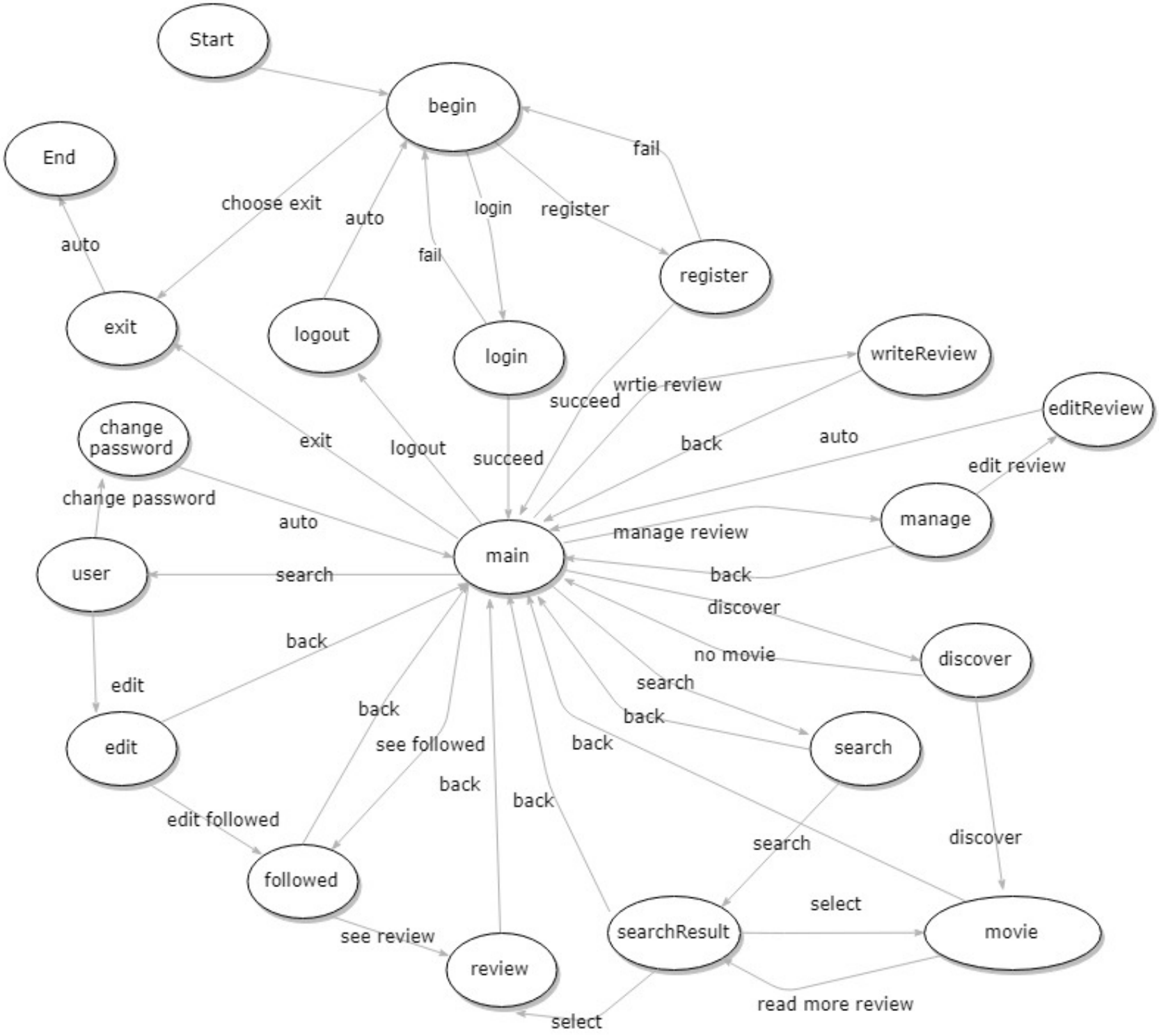
User	Represent user who used movie review website	Is kept in UserManager and MovieReviewSocialNetwork. Is managed by UserManager.
UserManager	Manage User and handle autentication	Manage and keep Users.

Sequence diagrams




Search
User
MRSN : MovieReviewSocialNetwork
MovieManager
MovieCollection
movies : Movie
1: search
1.1: searchState()
1.2: ask for search option
2: enter search option : movie
2.1: ask for movie name
3: enter search input
3.1: search
3.1.1: searchMovie()
3.1.1.1: search for movie
3.1.1.2: ArrayList of movie id
3.1.2: ArrayList of movie id
3.2: ArrayList of movie id
3.3: stateChange()
3.4: searchResultState()
3.5: show search result
Powered By Visual Paradigm Community Edition


write review
User
MRSN : MovieReviewSocialNetwork
MovieManager
ReviewManager
reviews : ReviewCollection
ReviewFileManager
newReview : Review
1: choose to write review
1.1: writeReviewState()
1.2: ask for movie name
2: enter movie name
2.1: search()
2.2: ArrayList of movie id
2.3: print search result
2.4: ask for action
3: select movie
3.1: ask for review info
4: enter review info
4.1: create new Review
4.2: addNewReview(newReview)
4.2.1: put(newReview)
4.2.1.1: userAddMatchTable()
4.2.1.2: userAddMatchTable()
4.2.1.3: movieAddMatchTable()
4.3: writeNewReview()
4.3.1: save to file
4.3.2: return
4.4: return
4.5: show result
Powered By Visual Paradigm Community Edition


manage review
User
MRSN : MovieReviewSocialNetwork
ReviewManager
allReviews : ReviewCollection
reviews : Review
1: enter manage state
1.1: show user's review
2: edit review
2.1: editReviewState()
2.2: ask what to edit
3: choose what to edit
3.1: editReview()
3.1.1: editReview()
3.1.1.1: edit
Powered By Visual Paradigm Community Edition

State Diagram