

myThoughts

Debate Web Application

Version 1.0

11 March 2017

Team 2

Logan Jahnke

Shaban Samani

Lucy Bradley

April Abbott

Table of Contents

Section 1: Introduction	2
1.1 Purpose	2
1.2 Scope and Limitations	2
1.3 Definitions, Acronyms, and Abbreviations	3
1.4 Overview	3
Section 2: Proposed System	4
2.1 System Overview	4
2.2 Functional Requirements	4
2.3 Non-functional requirements	5
2.4 Constraints	5
2.5 Use Cases	6
2.5.1 Register	6
2.5.2 Login	7
2.5.3 Logout	8
2.5.4 Edit Profile	9
2.5.5 Assign Moderator	10
2.5.6 Create Debate Category	11
2.5.7 Create Debate Topic	12
2.5.8 Comment on Debate Topic	13
2.5.9 Reply to Comment	14
2.5.10 Vote on Debate Topic	15
2.5.11 Vote on Comment	16
2.5.12 Support Side on Debate Topic	17
2.5.13 Support Side on Comment	18
2.5.14 Delete Debate Topic	19
2.5.15 Delete Comment	20
2.5.16 Flag a Debate Topic	21
2.5.17 Flag a Comment	21
2.5.18 View Flags	22
2.6 Class Diagram	23
Section 3: Glossary	24

Section 1: Introduction

1.1 - Purpose

myThoughts is a web application that allows users to discuss their opinions on various topics in a forum-like setting. Posts and threads will be organized in a way that promotes a productive exchange of ideas.

1.2 - Scope

The product to be produced is titled *myThoughts*. The scope of myThoughts includes its features, benefits, and limitations.

1.2.1 - Features:

1. The system will allow an administrator to create *debate categories*. Each category will specify its *name* and *description*.
2. The system will allow an administrator to create *moderators*. Moderators can be chosen from the complete list of registered users (1.2.1.4).
3. A moderator will be able to create *debate topics*. Each topic will specify its *heading* and an *optional description*. Debate topics may also be created by users having a certain (unspecified) karma (1.2.1.8). Debate topics created by users are subject to review by the moderators and can be removed at any time. Debate topics that acquire ten downvotes will be automatically removed (1.2.1.5).
4. A user will be able to *register with the system*. To do that, the user will establish the *username* and *password*. The user must provide a legitimate *email address* to have user-level access.
5. A user will be able to *vote on debate topics*. A user is encouraged to upvote if the topic facilitates meaningful debate. Likewise, a user is encouraged to downvote if the topic is overtly offensive to a specific niche or if the topic is a duplicate. Downvoting will decrease the user's karma by one (1.2.1.8).
6. A user will be able to *comment on a debate topic*. Each comment is subject to an optional review by the moderators. The comment may be removed from the topic by the user who wrote it or by any moderator at any time. The comment will be removed if it acquires ten downvotes (1.2.1.7).
7. A user will be able to *vote on a comment*. A user is encouraged to upvote if the comment facilitates meaningful debate. Likewise, a user is encouraged to downvote if the comment is overtly offensive to a specific niche or if the comment is unrelated. Downvoting will decrease the user's karma by one (1.2.1.8).

8. Each user will have a *karma score*. Karma may be earned by positively contributing to the website. Karma may be revoked by negatively contributing to the website.
9. A user will be able to *agree or disagree with a comment*. This will allow to user to contribute their stance without commenting. Agreeing and disagreeing will not affect karma.
10. A user will be able to *flag a topic or comment* if they deem it inappropriate. It will be sent to the moderators for review.
- ~~11. A user will be able to *earn badges*. Each badge will represent an achievement from the site. Earning badges may increase karma. [FUTURE ENHANCEMENT]~~
12. The system will be *accessible from a common web browser* (such as the Mozilla Firefox, Google Chrome, Safari, and Internet Explorer).
13. The system should provide *multi-user access*, assuring correct concurrent behavior. The system should maintain suitable *authorization* information and *authenticate access*. *User authentication* should be implemented by checking *username* and *password*.
14. The system will have an *easy-to-use user interface* (UI) with screens designed for each part of the system's functionality and suitable for different types of users (customers, administrators, managers).
15. The system will use a *persistent data store* (MySQL RDBMS) for all the relevant data.
16. The system should use *accepted standards* whenever possible (HTML, CGI, Servlet API, JDBC, ODBC, SQL, etc.). The project *will* be coded in Java, possibly including other scripting languages, such as JavaScript, if needed.

1.2.3 - Benefits:

The benefit of myThoughts is to provide organization and facilitation of debates.

1.2.4 - Limitations:

myThoughts will not:

- Offer direct messaging service.

1.3 - Definitions, Acronyms, and Abbreviations

1.4 - Overview

This requirements document for myThoughts contains the introduction, a specification for the proposed system including the requirements, constraints, and system models, and a glossary.

Section 2: Proposed System

2.1 - System Overview

The proposed system, myThoughts, is our solution to facilitate debates. The system, which will use a website and a database, will allow an administrator to facilitate things like creating and managing moderators and creating debate categories. Moderators will be able to create debate topics and moderate user topics and comments. Users will be able to create debate topics, comment on debates, agree/disagree with a comment, upvote/downvote a debate topic, and upvote/downvote a comment. A non-registered user will be able to view debate topics and read comments. All users of the myThoughts system will be able to access the system by registering, logging in, and logging out of the system.

2.2 - Functional Requirements

2.2.1 - Administrator Interactions:

- Assign Moderator
- Create Debate Category
- Logout

2.2.2 - Moderator Interactions:

- Create Debate Topics
- Remove Debate Topics
- Remove Comments
- Logout

2.2.3 – Authenticated User Interactions:

- Create Debate Topics
- Comment
- Upvote and Downvote Topics
- Upvote and Downvote Comments
- Agree and Disagree with Comments
- Logout

2.2.4 - Unauthenticated User Interactions:

- Register
- Login

2.3 - Non-functional Requirements

2.3.1 - Performance:

- myThoughts will provide multi-user access (Administrator, Moderator, User)
- myThoughts will have an easy-to-use software interface
- myThoughts will have persistent data storage using MySQL RDBSM
- myThoughts will be accessible from any common internet browser
 - Safari
 - Google Chrome
 - Firefox
 - Internet Explorer

2.3.2 - Reliability:

- myThoughts will be available 24 hours/day and 7 days/week

2.3.3 - Maintainability:

- The Administrator will have complete control over all moderators and topics.

2.3.4 - Implementation:

- myThoughts will be implemented using the standard programming languages.
 - HTML/CSS
 - SQL
 - JavaScript
 - Java Servlets

2.3.5 - Extensibility:

- Users and moderators will be able to create new debate topics.

2.4 - Constraints

- The implementation language must be standard programming languages
- The user-interface must be accessible from any common internet browser

2.5 – Use Cases

2.5.1 – Register User

Name	Register User
ID	RegisterUser
Version	1.0
Author	Logan Jahnke
Date	03/11/2017
Summary	Use case to create a new User capable of creating debate topics, earning karma, commenting on debates, upvoting and downvoting, and agreeing and disagreeing on comments.
Basic Path	<ol style="list-style-type: none">1. The Unauthenticated User presses the "Register" button.<ol style="list-style-type: none">a. myThoughts responds by presenting a form to the Unauthenticated User.2. The Unauthenticated User completes form that specifies the first name, last name, email address, username, and password.<ol style="list-style-type: none">a. myThoughts receives form, compares it database to check if duplication exists.b. myThoughts adds the User to the database.
Alternative Paths	<ol style="list-style-type: none">1. The Unauthenticated User presses the cancel button, the creation of the User ceases, and the Use Case is terminated.
Exception Paths	<ol style="list-style-type: none">1. If myThoughts in step 2a finds a duplicate, then the action is cancelled, and the Unauthenticated User is notified that he/she made a duplicate.
Extension	
Triggers	The Unauthenticated User presses the "Register" button.
Assumption	
Pre-conditions	
Post-conditions	The User object is added to the database.

2.5.2 – Login

Name	Login
ID	Login
Version	1.0
Author	Logan Jahnke
Date	03/17/2017
Summary	Use case to authenticate an existing User .
Basic Path	<ol style="list-style-type: none">1. The Unauthenticated User presses the "Login" button.<ol style="list-style-type: none">a. myThoughts responds by presenting a form to the Unauthenticated User.2. The Unauthenticated User completes form that specifies the username and password.<ol style="list-style-type: none">a. myThoughts receives form, compares it database to check if the User exists.b. myThoughts authenticates the User.
Alternative Paths	<ol style="list-style-type: none">1. The Unauthenticated User presses the cancel button, the creation of the User ceases, and the Use Case is terminated.
Exception Paths	<ol style="list-style-type: none">1. If myThoughts in step 2a does not find the User in the database, then the action is cancelled, and the Unauthenticated User is notified that he/she made a mistake.
Extension	
Triggers	The Unauthenticated User presses the "Login" button.
Assumption	
Pre-conditions	
Post-conditions	The User object is authenticated.

2.5.3 – Logout

Name	Logout
ID	Logout
Version	1.0
Author	Logan Jahnke
Date	03/17/2017
Summary	Use case to revoke a session from an existing User .
Basic Path	<ol style="list-style-type: none">1. The User presses the "Logout" button.<ol style="list-style-type: none">a. myThoughts revokes the User's session.2. The newly Unauthenticated User is brought to the homepage.
Alternative Paths	
Exception Paths	
Extension	
Triggers	The User presses the "Logout" button.
Assumption	
Pre-conditions	
Post-conditions	The User object is unauthenticated.

2.5.4 – Edit Profile

Name	Edit Profile
ID	EditProfile
Version	1.0
Author	Logan Jahnke
Date	03/17/2017
Summary	Use case to edit information about a User .
Basic Path	<ol style="list-style-type: none">1. The User presses the "Edit Profile" button on their profile page.<ol style="list-style-type: none">a. myThoughts presents a form to the User displaying all things that are editable including <i>username</i>, <i>password</i>, <i>email address</i>, and <i>profile picture</i>.2. The User updates their information and then presses the "Update" button.<ol style="list-style-type: none">a. myThoughts updates the User in the database and returns the User to their profile page.
Alternative Paths	<ol style="list-style-type: none">1. If the User presses "Cancel" during step 2, the User is sent back to their profile page.
Exception Paths	
Extension	
Triggers	The User presses the "Edit Profile" button.
Assumption	
Pre-conditions	
Post-conditions	The User object is updated.

2.5.5 – Assign Moderator

Name	Assign Moderator
ID	AssignModerator
Version	1.0
Author	Logan Jahnke
Date	03/17/2017
Summary	Use case to allow an Administrator to assign a Moderator from a list of Users .
Basic Path	<ol style="list-style-type: none">1. The Administrator presses the "Assign Moderator" button on their unique home page.<ol style="list-style-type: none">a. myThoughts presents a form to the Administrator displaying a list of all eligible Users.2. The Administrator picks a User to assign as a Moderator.<ol style="list-style-type: none">a. myThoughts updates the User in the database to become a Moderator.
Alternative Paths	<ol style="list-style-type: none">1. If the Administrator presses “Cancel” during step 2, the Administrator is sent back to their unique home page.
Exception Paths	
Extension	
Triggers	The Administrator presses the “Assign Moderator” button.
Assumption	
Pre-conditions	
Post-conditions	The User object is made into a Moderator .

2.5.6 – Create Debate Category

Name	Create Debate Category
ID	CreateDebateCategory
Version	1.0
Author	Logan Jahnke
Date	03/17/2017
Summary	Use case to allow an Administrator to create a Debate Category .
Basic Path	<ol style="list-style-type: none">1. The Administrator presses the "Create Debate Category" button on their unique home page.<ol style="list-style-type: none">a. myThoughts presents a form that allows the Administrator to enter in a name.2. The Administrator enters a name for the Debate Category.<ol style="list-style-type: none">a. myThoughts inserts the Debate Category in the database.
Alternative Paths	<ol style="list-style-type: none">1. If the Administrator presses "Cancel" during step 2, the Administrator is sent back to their unique home page.
Exception Paths	
Extension	
Triggers	The Administrator presses the "Create Debate Category" button.
Assumption	
Pre-conditions	
Post-conditions	A new Debate Category is in the system.

2.5.7 – Create Debate Topic

Name	Create Debate Topic
ID	CreateDebateTopic
Version	1.0
Author	Logan Jahnke
Date	03/17/2017
Summary	Use case to allow a Moderator or an eligible User to create a Debate Topic .
Basic Path	<ol style="list-style-type: none">1. The Moderator/User presses the "New Debate Topic" button.<ol style="list-style-type: none">a. myThoughts presents a form that allows the Moderator/User to enter in the title and an optional description.2. The Moderator/User enters the title and optional description for the Debate Topic.<ol style="list-style-type: none">a. myThoughts inserts the Debate Topic in the database.
Alternative Paths	<ol style="list-style-type: none">1. If the Moderator/User presses "Cancel" during step 2, the Moderator/User is sent back to the home page.
Exception Paths	
Extension	
Triggers	The Moderator/User presses the "New Debate Topic" button.
Assumption	
Pre-conditions	
Post-conditions	A new Debate Topic is added in the system.

2.5.8 – Comment on Debate Topics

Name	Comment on Debate Topics
ID	CommentOnDebateTopics
Version	1.0
Author	Logan Jahnke
Date	03/17/2017
Summary	Use case to allow a User to Comment on a Debate Topic .
Basic Path	<ol style="list-style-type: none">1. The User presses the "Comment" button.<ol style="list-style-type: none">a. myThoughts presents a form that allows the User to enter a <i>subject line</i>, an <i>argument</i>, and <i>whether they agree or disagree</i>.2. The User enters the required information.<ol style="list-style-type: none">a. myThoughts inserts the Comment in the database.
Alternative Paths	<ol style="list-style-type: none">1. If the User presses "Cancel" during step 2, the User is sent back to the Debate Topic page.
Exception Paths	
Extension	
Triggers	The User presses the "Comment" button.
Assumption	
Pre-conditions	
Post-conditions	A new Comment is added in the system.

2.5.9 – Reply to Comment

Name	Reply to Comment
ID	ReplyToComment
Version	1.0
Author	Logan Jahnke
Date	03/18/2017
Summary	Use case to allow a User to Reply on a Comment .
Basic Path	<ol style="list-style-type: none">1. The User presses the "Reply" button under a Comment.<ol style="list-style-type: none">a. myThoughts presents a form that allows the User to enter a <i>subject line</i>, an <i>argument</i>, and <i>whether they agree or disagree</i>.2. The User enters the required information.<ol style="list-style-type: none">a. myThoughts inserts the Reply in the database.
Alternative Paths	<ol style="list-style-type: none">1. If the User presses "Cancel" during step 2, the User is sent back to the Debate Topic page.
Exception Paths	
Extension	
Triggers	The User presses the "Reply" button.
Assumption	
Pre-conditions	
Post-conditions	A new Reply is added in the system.

2.5.10 – Vote on Debate Topic

Name	Vote on Debate Topic
ID	VoteOnDebateTopic
Version	1.0
Author	Logan Jahnke
Date	03/18/2017
Summary	Use case to allow a User to Vote either an <i>upvote</i> or a <i>downvote</i> to indicate whether the Debate Topic is a good topic that facilitates debate.
Basic Path	1. The User presses the "Upvote" button near a Debate Topic . a. myThoughts increments the vote for the Debate Topic and records that the User upvoted the topic.
Alternative Paths	1. The User presses the "Downvote" button near a Debate Topic . a. myThoughts decrements the vote for the Debate Topic and records that the User downvoted the topic and the User's karma score is decremented.
Exception Paths	
Extension	
Triggers	The User presses the "Upvote" or "Downvote" button.
Assumption	
Pre-conditions	
Post-conditions	The Debate Topic's score is either incremented or decremented.

2.5.11 – Vote on Comment

Name	Vote on Comment
ID	VoteOnComment
Version	1.0
Author	Logan Jahnke
Date	03/18/2017
Summary	Use case to allow a User to Vote either an <i>upvote</i> or a <i>downvote</i> to indicate whether the Comment is a good comment that facilitates debate.
Basic Path	2. The User presses the "Upvote" button near a Comment . a. myThoughts increments the vote for the Comment and records that the User upvoted the comment.
Alternative Paths	2. The User presses the "Downvote" button near a Comment . a. myThoughts decrements the vote for the Comment and records that the User downvoted the comment and the User's karma score is decremented.
Exception Paths	
Extension	
Triggers	The User presses the "Upvote" or "Downvote" button.
Assumption	
Pre-conditions	
Post-conditions	The Comment's score is either incremented or decremented.

2.5.12 – Support Debate Topic Side

Name	Support Debate Topic Side
ID	SupportDebateTopicSide
Version	1.0
Author	Logan Jahnke
Date	03/18/2017
Summary	Use case to allow a User to Agree or Disagree with a Debate Topic .
Basic Path	1. The User presses the "Agree" button near a Debate Topic . a. myThoughts increments the agree score for the Debate Topic and records that the User agreed with a Debate Topic .
Alternative Paths	1. The User presses the "Disagree" button near a Debate Topic . a. myThoughts increments the disagree score for the Debate Topic and records that the User disagreed with the Debate Topic .
Exception Paths	
Extension	
Triggers	The User presses the "Agree" or "Disagree" button.
Assumption	
Pre-conditions	
Post-conditions	The Debate Topic's agree score or disagree score is incremented.

2.5.13 – Support Comment Side

Name	Support Comment Side
ID	SupportCommentSide
Version	1.0
Author	Logan Jahnke
Date	03/18/2017
Summary	Use case to allow a User to Agree or Disagree with a Comment .
Basic Path	1. The User presses the "Agree" button near a Comment . a. myThoughts increments the agree score for the Comment and records that the User agreed with a Comment .
Alternative Paths	1. The User presses the "Disagree" button near a Comment . a. myThoughts increments the disagree score for the Comment and records that the User disagreed with the Comment .
Exception Paths	
Extension	
Triggers	The User presses the “Agree” or “Disagree” button.
Assumption	
Pre-conditions	
Post-conditions	The Comment’s agree score or disagree score is incremented.

2.5.14 – Remove Debate Topic

Name	Remove Debate Topic
ID	RemoveDebateTopic
Version	1.0
Author	Logan Jahnke
Date	03/18/2017
Summary	Use case to allow a Moderator to remove a Debate Topic . Also allows a User to delete their own Debate Topic .
Basic Path	<ol style="list-style-type: none"> 1. The Moderator presses the "Delete Topic" button near a Debate Topic. <ol style="list-style-type: none"> a. myThoughts presents a warning to the Moderator. 2. The Moderator presses the "Confirm Deletion" button. <ol style="list-style-type: none"> a. myThoughts will create a snowball effect and delete all Comments attached to the Debate Topic followed by the Debate Topic. Furthermore, the User who created the topic will have their karma decremented by 5.
Alternative Paths	<ol style="list-style-type: none"> 1. The Moderator presses the "Cancel" button in step 2, and the Debate Topic is not deleted.
Exception Paths	
Extension	
Triggers	The Moderator presses the "Delete Topic" button.
Assumption	
Pre-conditions	
Post-conditions	All Comment's on the topic and the Debate Topic are removed from the database.

2.5.15 – Remove Comment

Name	Remove Comment
ID	RemoveComment
Version	1.0
Author	Logan Jahnke
Date	03/18/2017
Summary	Use case to allow a Moderator to remove a Comment . Also allows a User to delete their own Comment .
Basic Path	<ol style="list-style-type: none"> 1. The Moderator presses the "Delete Topic" button near a Comment. <ol style="list-style-type: none"> a. myThoughts presents a warning to the Moderator. 2. The Moderator presses the "Confirm Deletion" button. <ol style="list-style-type: none"> a. myThoughts will replace the text of the Comment with [Removed]. This is to keep the Replies intact.
Alternative Paths	<ol style="list-style-type: none"> 1. The Moderator presses the "Cancel" button in step 2, and the Comment is not deleted.
Exception Paths	
Extension	
Triggers	The Moderator presses the "Delete Comment" button.
Assumption	
Pre-conditions	
Post-conditions	The Comment's text is changed to [Removed].

2.5.16 – Flag a Debate Topic

Name	Flag a Debate Topic
ID	FlagDebateTopic
Version	1.0
Author	Logan Jahnke
Date	03/18/2017
Summary	Use case to allow a User to flag a Debate Topic .
Basic Path	1. The User presses the "Flag" button near a Debate Topic . a. myThoughts sends the flag to the Moderators .
Alternative Paths	
Exception Paths	
Extension	
Triggers	The User presses the “Flag” button.
Assumption	
Pre-conditions	
Post-conditions	The Flag noted in the database.

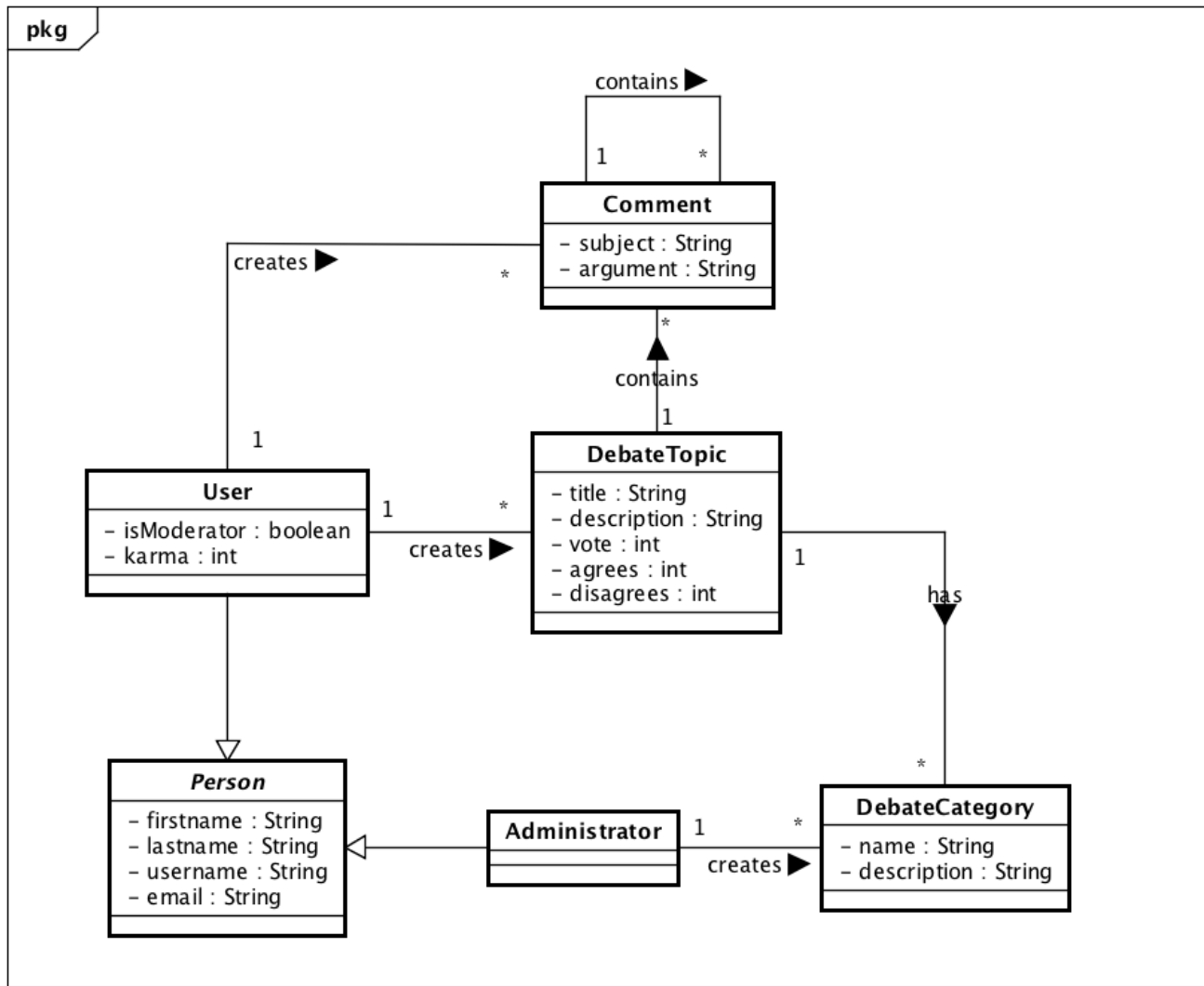
2.5.17 – Flag a Comment

Name	Flag a Comment
ID	FlagComment
Version	1.0
Author	Logan Jahnke
Date	03/18/2017
Summary	Use case to allow a User to flag a Comment .
Basic Path	1. The User presses the "Flag" button near a Comment . a. myThoughts sends the flag to the Moderators .
Alternative Paths	
Exception Paths	
Extension	
Triggers	The User presses the “Flag” button.
Assumption	
Pre-conditions	
Post-conditions	The Flag noted in the database.

2.5.18 – View Flags

Name	View Flags
ID	ViewFlags
Version	1.0
Author	Logan Jahnke
Date	03/18/2017
Summary	Use case to allow a Moderator to view all Flagged Comments and Flagged Debate Topics .
Basic Path	1. The Moderator presses the "View Flags" button. a. myThoughts responds with a page containing all posts that have been flagged..
Alternative Paths	
Exception Paths	
Extension	
Triggers	The Moderator presses the “View Flags” button.
Assumption	
Pre-conditions	
Post-conditions	

2.6 – Class Diagram



Section 3: Glossary

Word	Definition
Administrator	The person that manage myThoughts, Admins can create debate categories and assign moderators.
Multi user access	A system that allows multiple users to access the system at the same time
Persistent data storage	Storing data and saving it so that it is not lost if the system goes down
System	This refers to the program, and what the program was created to accomplish/compute
User interface	The graphical interface presented to the user