

IskoExchange

Analysis Model

Submitted to:

Asst. Prof. Ma. Rowena C. Solamo
Faculty Member
Department of Computer Science
College of Engineering
University of the Philippines, Diliman

Submitted by:
Balingit, Ivan Carlo M.
Cajaljal, Patricia Mae G.
Tan, Luis Carlo B.

In partial fulfillment of Academic Requirements
for the course
CS 191 Software Engineering I
of the
1st Semester, AY 2017-2018

Revision Control

History Revision:

Revision Date	Person Responsible	Version Number	Modification
10/20/17	Ivan Balingit	1.0	Initial Document
10/21/17	Luis Tan	2.0	Added Class Diagram
10/23/17	Ivan Balingit	2.1	Added Boundary and Entity Classes
10/23/17	Patricia Cajaljal	2.2	Added Control Classes

Purpose:

The document contains the analysis model for IskoExchange. The document includes the Class Diagram, Boundary, Entity, and Control Classes of the System.

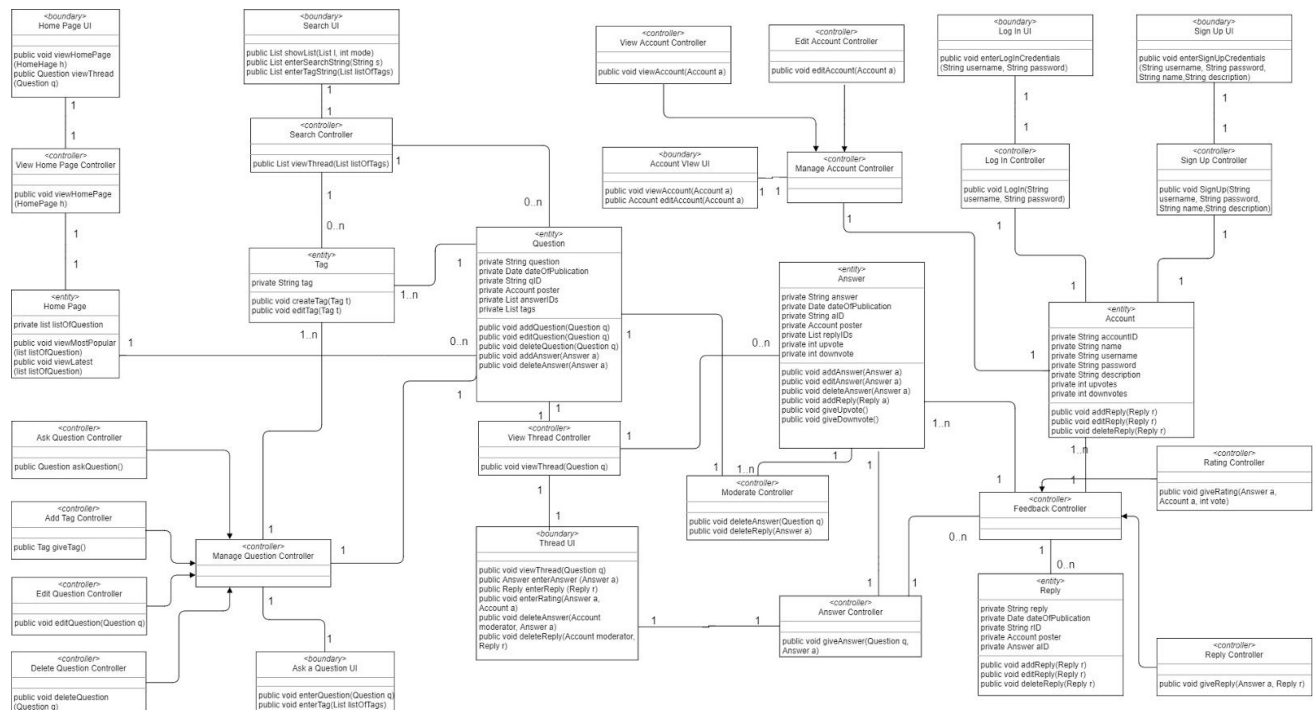
Audience:

This document is made as partial fulfillment of academic requirements for the course CS 191 – Software Engineering I, handled by Prof. Ma. Rowena C. Solamo.

System Name: IskoExchange

Description: IskoExchange is a platform for UP students to ask questions and share insights related to UP. These topics can vary from academics to certain University processes, and many more. IskoExchange aims to connect UP students together and answer each other's concerns quickly and clearly.

Class Diagram:



Boundary Classes:

Class Name	Description
HomePageUI	This interface allows the user to have an overview of the entire site. From the home page, a user can choose to login or sign-up, ask a question, and see the recent and popular questions.
ThreadUI	<p>This interface allows the user to see more details about a question/thread. In this interface, the user can see the entirety of the question, as well as the answers and their respective comments and replies.</p> <p>The user can also post an answer if the question does not belong to them. If a user sees an inappropriate question, answer, or reply, they can report it so that the moderators can do the necessary actions.</p>
SearchUI	In this interface, the user can input a question or a part of it, and the site will return the questions that are most relevant to the search query. The user can also search by using tags, an attribute that a question may have.
AccountViewUI	This interface allows the user to view the details of their own account, or those of others. From the interface, details such as full name, questions posted, or answers can be seen.
AskQuestionUI	<p>When a user asks a question, this interface allows them to input the necessary details, such as the title and description of the question.</p> <p>From this interface, the user can also include the necessary tags so that other users can find the question easily.</p>
SignUpUI	This interface allows the user to create their own account. The interface includes a form which the user can fill up with the necessary details.
LogInUI	This interface allows the user to login to their own account. This interface includes a form which asks for the username and password of the user. Logging in enables the user to ask their own questions.

Control Classes:

Class Name	Description
ViewHomeController	This controller is for viewing the home page.
LoginController	This controller is for logging in to the user's account.
SignupController	This controller lets the user signup for an account.
SearchController	This controller allows the user to search for questions.
ManageAccountController	This controller allows the user to manage his/her account.
EditAccountController	This controller is extended from the ManageAccountController. It allows the user to edit his/her account to update information about the user's profile.
ViewAccountController	This controller is extended from the ManageAccountController. It allows the user to view his/her account as well as the account of other users.
ManageQuestionController	This controller is for managing questions asked/submitted by the users.
AskQuestionController	This controller is extended from the ManageQuestionController. This lets the user ask/ submit a question for other users to browse and answer.
AddTagController	This controller is extended from the ManageQuestionController. This lets the question be under specific tags for convenient searching of questions.
EditQuestionController	This controller is extended from the ManageQuestionController. This lets the questioner edit his/her question.
DeleteQuestionController	This controller is extended from the ManageQuestionController. This lets the questioner delete his/her question.
ModerateThreadController	This controller moderates the thread. This is for maintaining quality and informative conversation/threads for the users.
AnswerController	This controller allows the user to give an answer to a question of choice.
FeedbackController	This controller allows the user to give feedback to answers to questions on threads.
ReplyController	This controller is extended from the FeedbackController. This allows the user to give a reply to an answer and also updates the thread.
GiveRatingController	This controller is extended from the FeedbackController. This allows the user to give rating to an answer based on his/her satisfaction to the answer.
ViewThreadController	This controller allows the user to view the thread including the question, its answers, the answers' replies and the upvotes/downvotes of the answers.

Entity Classes:

Class Name	Description
HomePage	The HomePage represents the content of the website when you visit. This can include announcements or content for the header of the home page.
Question	The Question is an entity that has the question title and description. A question can be created, viewed, updated, or deleted by a logged-in user. Other logged-in users can post an answer to the question. Visitors that are not logged-in can only view a question.
Answer	The Answer is part of a question. A question can have many answers. As stated above, other users that are logged-in can post an answer to a question that is not their own.
Reply	Any logged-in user can post a reply to an answer of a question.
UserAccount	The User Account contains the necessary details of the user such as name, username, email, and other necessary information. The user account is created when someone signs-up. An account is required to ask a question or post an answer.
Tag	A Tag is part of a question. Questions can have many tags. These tags refer to the words or topics related to the question. Tags help users to find a question easily.