TABLE OF CONTENTS

Objectives

This document describes the plan for testing NKUNet of the Ruby on Rails framework.

- Describe what tests are performed
- Show how each test arises from a requirements document
- Describe integration tests that verify major application flows

Project Tests

The tests that are implemented in our application are housed within our controllers. Here is where we test…

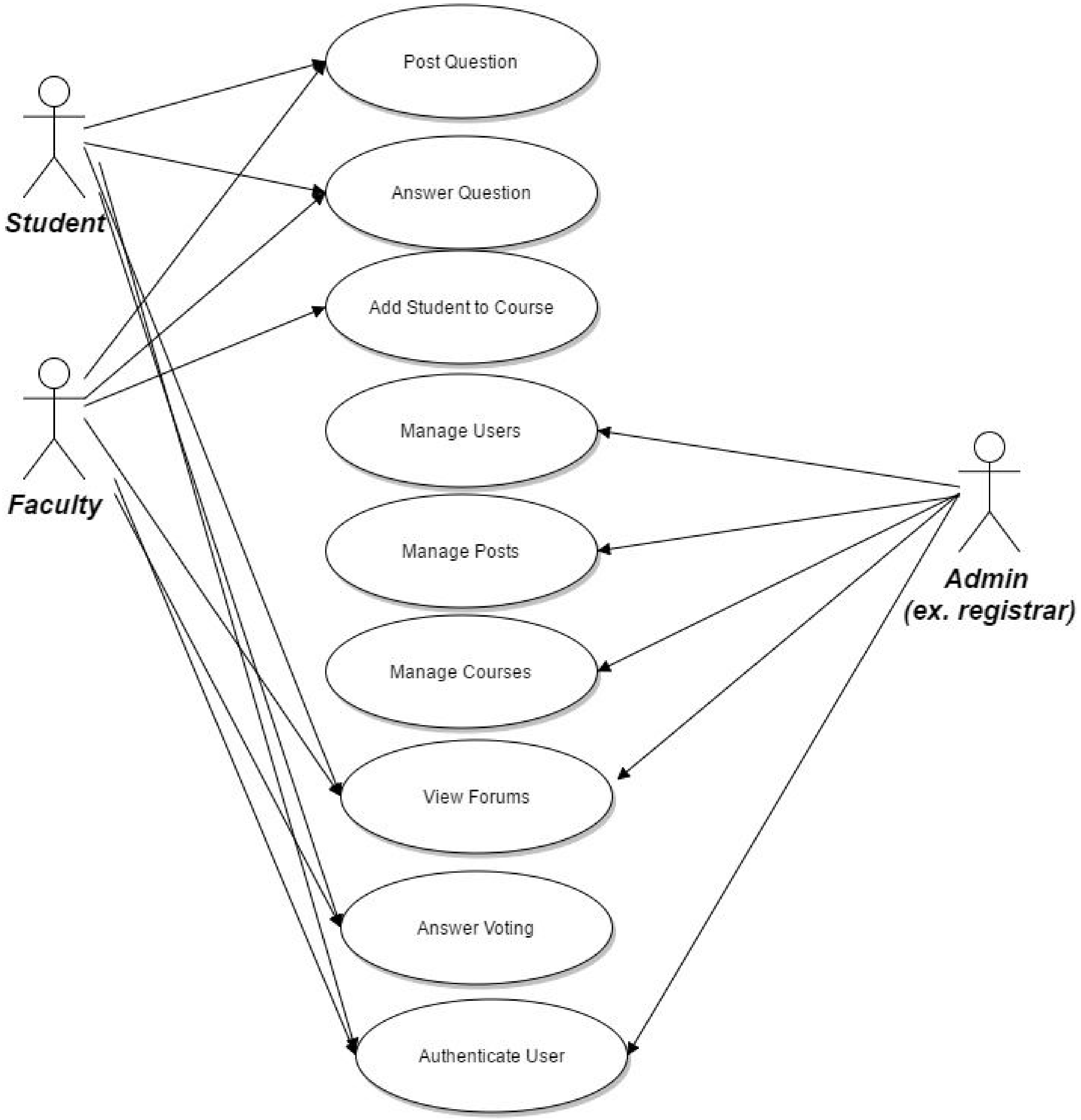
- Index methods
- New methods
- Create methods
- Show methods
- Edit methods
- Update methods
- Destroy methods

These tests are covered for all of our controllers that we use other than the users controller. All of the user tests were done with the implementation of the gem devise. The controllers that are covered are…

- Answers
- Courses
- Enrolled_Courses
- Forums
- Questions

Derived Tests

Almost all of our tests were derived from our use case model.



Integration Tests

The major application flows are all handled using assertions. I'll use the questions controller test suite as an example that is similar to the rest.

- The setup method is a default method to the questions controller
- The "should get index" test method generates a get request and returns a list of all the questions
- The "should get new" test method generates a get request and renders new
- The "should create question" test method creates an instance of a question storing it in the variable @question
- The "should show question" test method generates a get request, gets the question id and renders show
- The "should get edit" test method generates a get request, gets the question id and renders edit
- The "should update question" test method generates a get request, gets the question id, body, forum_id, and user_id and updates question
- The "should destroy question" test method decrements the total question count, gets the question id and calls the destroy method