

# Lab 7, Group A - Repositories and Unit Tests

CS295N, Web Development 1: ASP.NET

## Objectives

The purpose of this lab is to help you learn to:

- Create real and fake repositories
- Use dependency injection to provide controller objects with the correct repository
- Write controller methods that use the repositories
- Write Unit tests to test the controller methods

## Part 1: Chapter 7 Exercise

Do the exercise in Freeman, *Pro ASP.NET Core 2*, chapter 7 through page 185 (stop before the section on Moq).

If you haven't done the chapter 6, exercise, you can download the completed Ch. 6 code from the author's GitHub repository, <https://github.com/apress/pro-asp.net-core-mvc-2> .

## Part 2: Community Information Site

You will be adding testing capability to the Contact page. At this point, messages will still just be stored in memory, not in a database. Here is what you need to add:

- 1) A Message repository; which means you will need:
  - a) An interface
  - b) A real repository that in the future will store messages entered by the user. You should already have this, but will need to modify it to implement the interface above
  - c) A fake repository that contains hard-coded messages for testing
- 2) View(s) and Controller method(s) for replying to messages (if you haven't already done this).
- 3) Modify the Contact controller to work with the new ContactRepository.
- 4) Unit tests for the methods in the Contact controller.
  - a) The tests should use the fake Message repository
  - b) Write tests for all the methods that do something more than just invoke a view.
  - c) You should end up with tests for at least 3 or 4 methods.

*Continued on the next page*

## **Submission to Moodle**

### Beta Version

Upload the following to the Code Review Forum:

1. A document containing screen-shots of the web app in exercise running in your browser. (please use .docx or .pdf format)
2. A zip file containing your web app's Visual Studio solution folder.  
Or, a link to a repository containing your web site solution source code.  
You can put the link on the same document with the report on your tutorial exercise.
3. A code review of your lab partner's work. (You do this after your lab partner submits items 1 and 2 and you review them.)

### Production Version

1. Items 1 and 2 above, but revised as needed.
2. The code review of your work (the one done by your lab partner) with the second column ("Production") completed by you.