Activity 2

Erick Grant

CSET, Grand Canyon University

CST-339: Java III

Prof. Sparks

October 1, 2023

Part 1

Hello World!

Figure 1- Part 1- Step 6

Hello Spring MVC Framework

Figure 2- Part 1- Step 11

Hello World from ModelAndView!

Another Hello World from ModelAndView!

Figure 3- Part 1- Step 14

Hello World from a Thymelead template!

Go to /test4

Figure 4- Part 1- Step 16

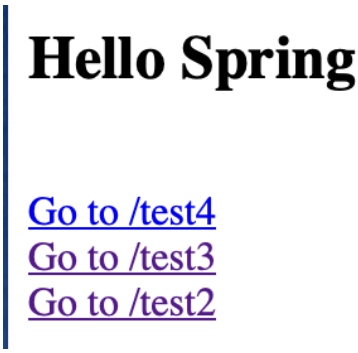

Figure 5- Part 1- Step 18A

Figure 6- Part 1- Step 18B

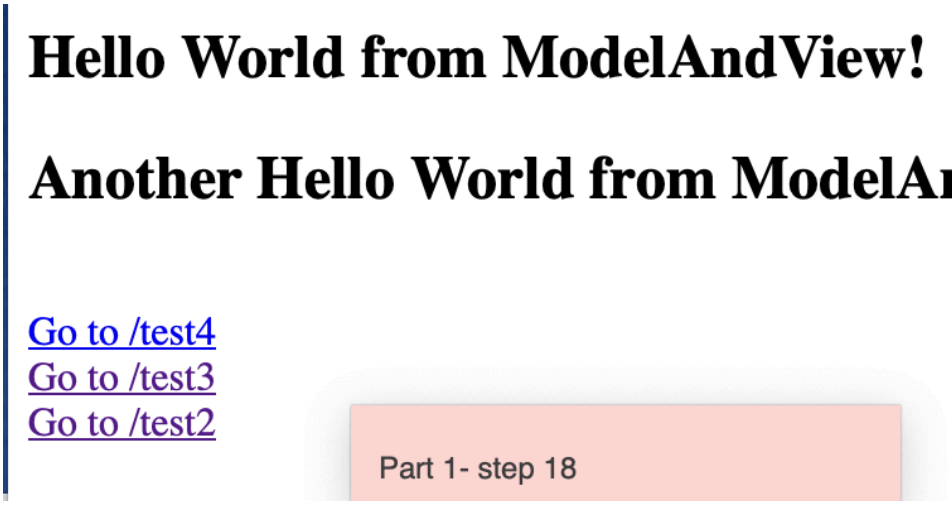

Figure 7- Part 1- Step18C

root directory

Go to /test4
Go to /test3
Go to /test2

Figure 8- Part 1- Step 19B (Controller /)

Welcome to CST-339 Topic 2 Activity for Erick Grant

Hello Spring MVC Framework

Go To /test2
Go To /test3
Go to /test4

Figure 9- Part 1- Step 19C (/Test2)

Welcome to CST-339 Topic 2 Activity for Erick Grant

Hello World from ModelAndView!

Another Hello World from ModelAndView!

Go To /test2
Go To /test3
Go to /test4

Figure 10- Part 1- /Test3

Welcome to CST-339 Topic 2 Activity for Erick Grant

Hello World from a Thymeleaf template!

Go To /test2
Go To /test3
Go to /test4

Figure 11- Part 1- /Test4

Part 2

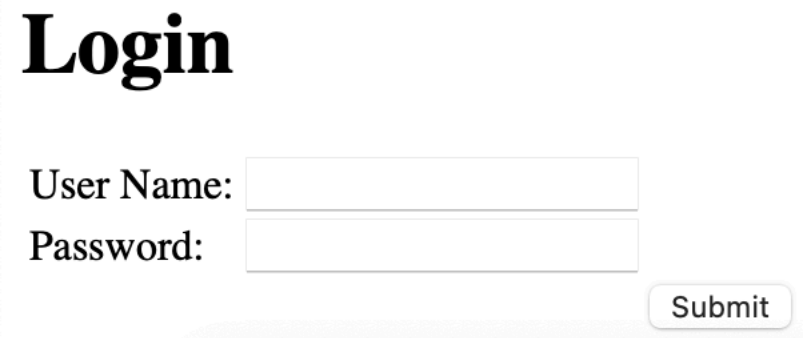

Figure 12- Login Form Part 2- Step 13

Order Number	Product Name	Price	Quantity
00000000001	Product 1	1.0	1
00000000002	Product 2	2.0	2
00000000003	Product 3	3.0	3
00000000004	Product 4	4.0	4
00000000005	Product 5	5.0	5

Figure 13- Part 2- Step 21- Orders View

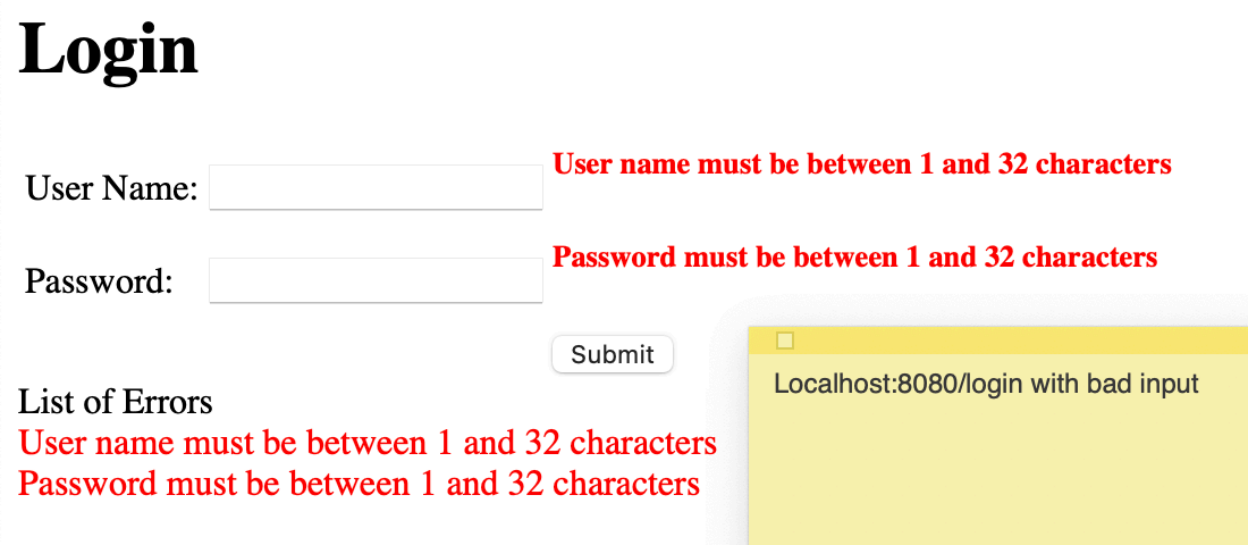

Figure 14- Part 2- Part 27- Bad inputs in login form

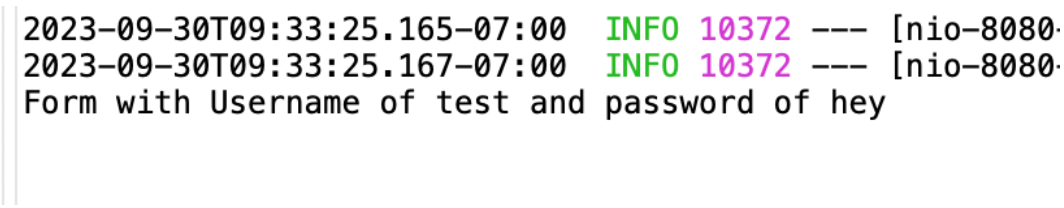

Figure 15- Part 2- Login form inputs are put into console

Part 3

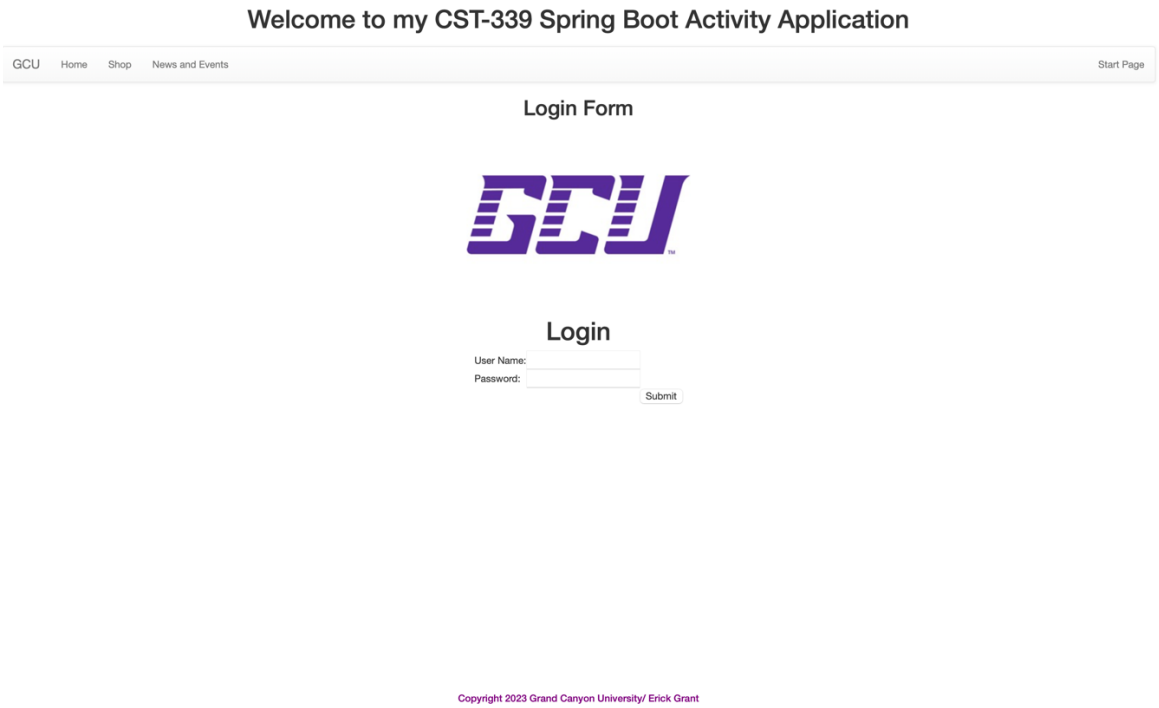

Figure 16- Part 3- Finished Login Form with Layouts

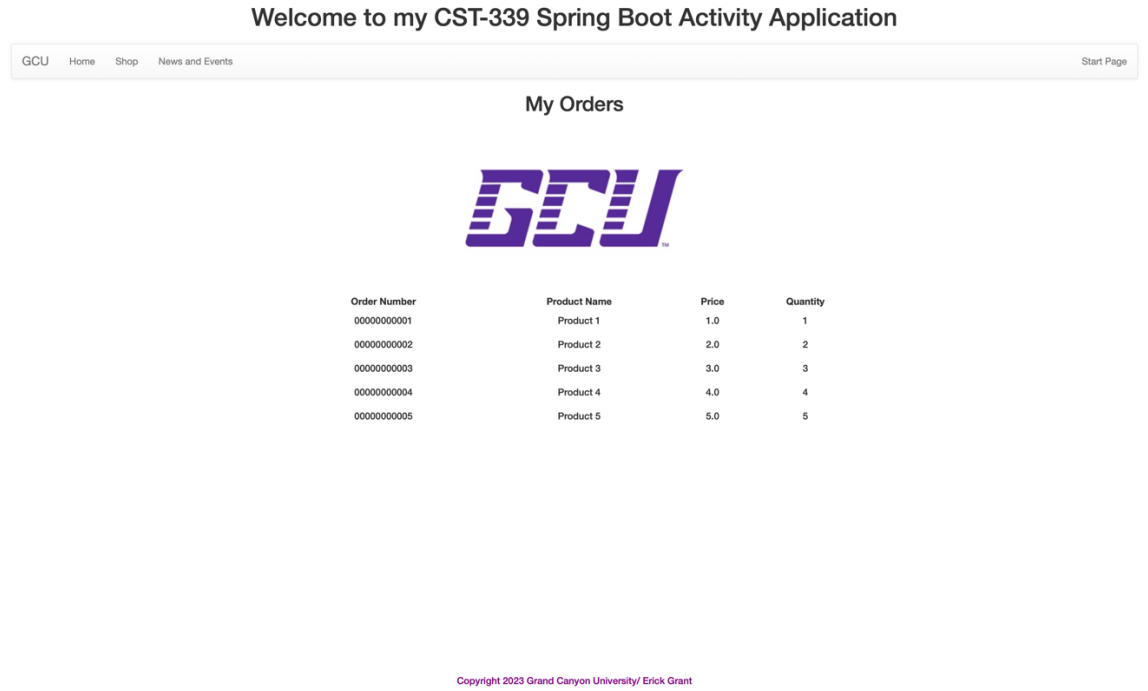

Figure 17- Part 3- Finished Orders page with Layouts

Research Questions

1. How does Spring MVC support the MVC design pattern? Draw a diagram that supports the answer to this question.

The Spring-Model View-Controller supports the MVC design pattern in numerous ways, such as using the singleton pattern ensuring only one instance of an object exists inside of an application. It also uses autowired singletons, which means that the object ID inside the controller for a group is the same as the ID used in the controller for the single item. This is used in aggregation systems such as databases. The Spring MVC allows for external configuration, which means versatility such as templates can be used as well as behavior of a application. This is helpful to simplify the core application which increases speeds and decreases complexity.

Diagram

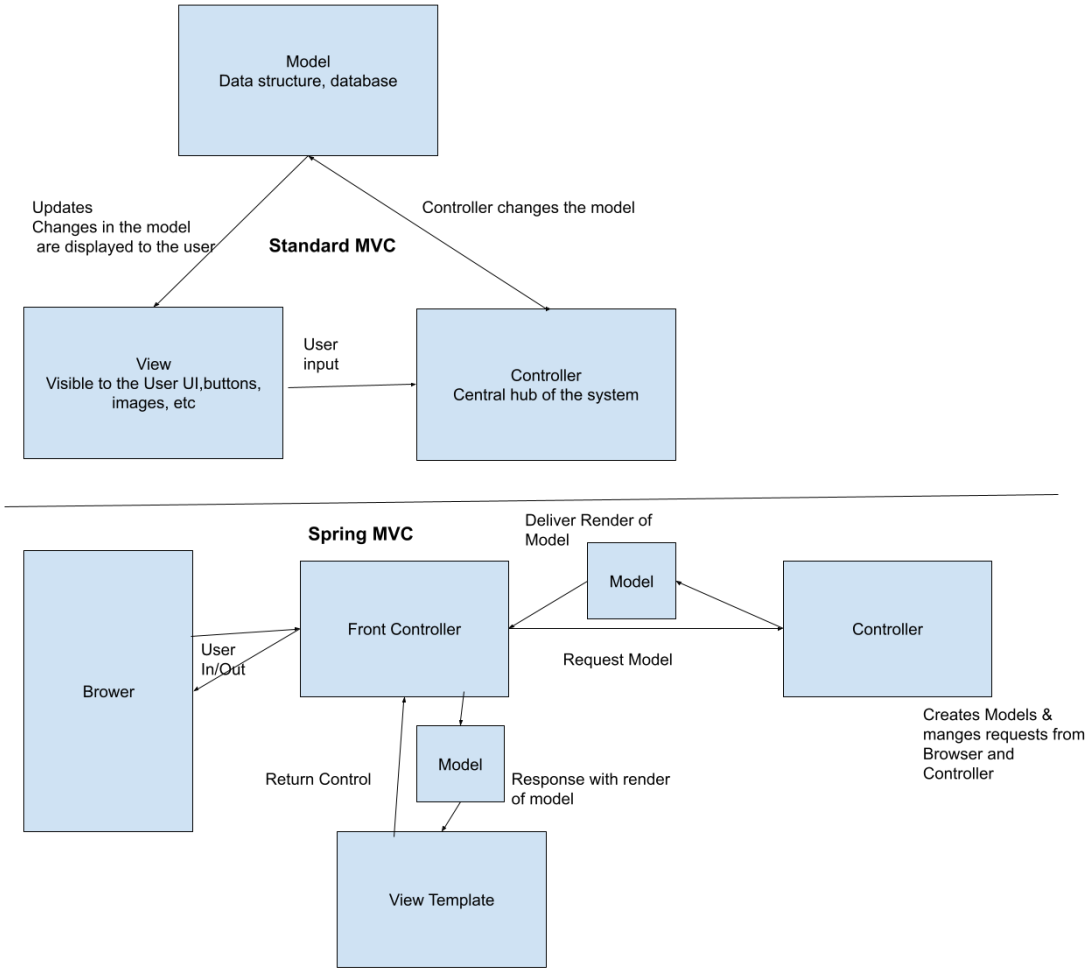

Figure 18- Standard MVC (above) vs the browser-based Spring MVC (below)

2. Research and identify 2 MVC Frameworks other than Spring MVC. What are the frameworks and how do they differ from Spring MVC?

Ruby on Rails and Django are both very popular MVCs, they all have very large development communities and documentation. Ruby has a more readable language than Spring's Java and Object-Oriented based language. Spring tends to be more scalable and overall better performance. Ruby also tends to be more set-in stone, which can be helpful for finding suitable APIs but can be a bit of a hardship when applying the system to the development needs.

Django like Ruby imposes numerous constraints and gives less freedom to the developer, this can help decrease the risk of code works together incorrectly but increases the learning curve for the language. Spring is designed to mold to the needs of the developer as best as possible, this is helpful to allow Spring to be used in a very wide range of applications and this can increase confusion among developers as code may not work together as well as they would of Ruby or Django. Django requires more decision making in the development stage which can increase time and if the wrong decision is made more time can be spent fixing it.

References:
https://stackoverflow.com/questions/8547226/just-curious-to-know-the-difference-between-spring-mvc-vs-django