POLITEKNIK BRUNEI	SCHOOL OF POLITEKNIK BRUNEI WEEKLY MODULE PLAN	**REFERENCE NUMBER:** PB/AS/LP/002 **VERSION:** 1.2 **REVISION DATE:** 06/10/2018 **EFFECTIVE DATE:** 1/1/2019

MODULE CODE/NAME	**NS4307, Network Programming**	**TOPIC**	**Java Web Application**
SEMESTER	**Semester 2, 2021 / 2022**	**GROUP CODE/INTAKE**	**DITN12 (Intake 11), DITN10R (Intake 9)**
SCHOOL/DEPT	**School of ICT**	**WEEK**	**Week 6 - 27/02 – 05/03**

PEDAGOGICAL APPROACH

☐ Knowledge Building ☑ Presentation ☐ Co-operative/Collaborative Learn ☐ Experimental Learning ☑ Case Studies
☐ Discussion ☑ Blended Learning ☐ Debate ☐ Others, please specify _____

RESOURCES

Learning management sytem

LESSON LEARNING OBJECTIVES	METHOD OF INSTRUCTION/LEARNING ACTIVITIES	REFLECTIVE EVALUATION
At the end of this lesson, the student should be able to: **1) Identify the tools needed for building Web Application using Spring Framework.** **2) Able to implement basic response for a Uniform Resource Identifier.**	**Session 1:** **Lecture: 2 hours** **- Explain the required tools for this topic.** **- Explain the tern Web Application.** **- Explain the purpose of Web Framework and list different Web Framework for different programming language.** **- Explain what is Spring Framework.** **- Explain what is Maven.** **- Explain the purpose of Maven's dependency management.** **- Emphasis that the Maven Repository being shared contains all the required third partly library needed for developing Spring Application.** **- After everyone copied Spring Tool Suite and Maven Repository, ask the student to follow the following instructions.** **- Show example how to create Maven Project.** **- Explain the folder structure in the Maven Project.** **- Explain how to open pom.xml** **- Explain Spring Boot** **- Explain how to configure Spring Boot to Maven Project.** **- Explain other dependencies required and how to add to Maven Project: Spring Boot Starter Web, Spring Boot Starter Thymeleaf, Spring Boot Starter Test and NekoHTML.** **- Emphasis there are more dependencies that will be added in furture topics.** **- Explain the concept of Model View Controller (MVC)** **- Explain how to set up the Build.** **- Explain how to run Spring Web Application.** **- Explain another way to create Maven Project for Spring Framework (Spring Initializr).** **- Show how to use Spring Initializr** **- Show how to import existing Maven Project.** **- Explain how to run main class.** **Practical: 1 hour** **- Share Spring Tool Suite and Maven Repository to students.** **- Facilitate Spring Tool Suite installation to students.** **- Facilitate Maven Repository installation to students.** **- Facilitate Maven Project creation with students.**	**Encountered major issue where half of students implementation does not work. I was not able to resolve it within the class slot. But luckily there are few students who were having the issue staying back to wait for their next class in the afternoon.** **Researched and tried a lot of things, using @RestController annotation, clearing maven repository and redownloading them, adding @ComponentScan, etc. Spent almost 1hr and 30 mins, found a solution where the Spring boot version needs to be downgraded from version 3.0.3 to 2.7.9. Unfortunately after extensive research, I am still not sure why version 3.0.3 having issue where the request mapping is not properly mapped to the spring application which causes error. But using older version should still be fine for covering the topics in this module.**
At the end of this lesson, the student should be able to: **1) Able to implement the controller component of Spring Framework.** **2) Able to implement the view component of Spring Framework.** **3) Able to implement basic Thymeleaf syntax.**	**Session 2:** **Lecture:** **- Explain why the current application only show error page.** **- Explain Spring Controller** **- Explain Uniform Resource Locator (URL)** **- Explain Uniform Resource Identifier (URI)** **- Show example how to implement Spring Controller** **- Explain that a method that return String data type need to be created to response to each URI request** **- Explain Request Mapping** **- Explain RequestMapping annotation** **- Explain ResponseBody annotation** **- Show example of handling URI with RequestMapping and ResponseBody annotation.** **- Explain that html tags can be used in the Controller Class.** **- Explain that typically a web application uses HTML file to provide the content.** **- Explain Spring View** **- Explain Thymeleaf** **- Explain how Thymeleaf works** **- Show example on how to create HTML file.** **- Emphasis that this is what was learned during Basic Web Programming module.** **- Explain how to response with HTML file.** **- Explain what is static files.** **- Explain how to integrate static files into the Spring Web Application.** **- Show example of implementation of static files.** **- Explain Thymeleaf URL Expression** **- Explain how to implement Thymeleaf URL Expression** **- Explain Thymeleaf XML Namespace** **- Explain Absolute URLs, Context-relative URLs, Server-relative URLs and Protocol-relative URLs.** **Practical:** **- Facilitate students to configure Maven Project.** **- Facilitate students to implement a student management system.**	**Spent 15 minutes to make sure everyone managed to get their spring application to work.** **Found out that only a few remembers some of the tags in HTML from Basic Web Programming. Need to remember to recap some of the important tags that is needed in this module to make sure they understand how it connects to spring application..** **Another issue were found when connecting spring application with thymeleaf. Somehow spring tool suite exclude src/main/resources folder in its compilation which causes the html files not to be found by the application. Same case, half of them are having this issue. But luckily, it only needs a quick diagnostic and it is resolved quick. By right-clicking src/main/resource folder > Build Path > Configure Inclusions / Exclusions Filters. Then removing the "**" from the exclusion filter.** **Spent 10 minutes to show students what is the difference between context-relative URLs and server-relative URLs by changing the application.properties attribute, server. servlet.context-path, to /myapp.** **Otherwise, students are able to make use of Thymeleaf and it's URL syntax.**
At the end of this lesson, the student should be able to:	**Session 3:**	

NAME & SIGNATURE		
MODULE LECTURER Jailani Abdul Rahman Jailani Abdul Rahman DATE: 07/03/2023	PROGRAMME LEADER / ~~ASSISTANT HOS (aHOS)/ HEAD OF SCHOOL (HOS)~~* Jamiatul Zuhriah DATE: 09/03/2023	
COMMENTS BY PROGRAMME LEADER/ ASSISTANT HOS (aHOS)/ HEAD OF SCHOOL (HOS) (If any)		
Checked. Keep up a good work.		