1. Course Code

2293

2. Course Title

Software Development Exercises

3. Teacher

WANNOUS, Muhammad

4. Term

Fall 1

5. Course Requirements (Courses / Knowledge prerequisite for this course)

[1] Programming (command-line and web applications).
[2] Database design and management.
[3] System administration (package installation, file management...)

6. Course Overview and Objectives

This course includes a number of exercises and one project that require the students to use the technical skills and knowledge they acquired to complete. For each exercise, document describing the task is provided and every student is required to design, prototype, and test an application.
This course DOES NOT include lecture slides and the lecture time will be spent on writing the application code. Every participant will be required to complete a number of DESKTOP and WEB applications in the programming language he/she is comfortable with.

7. Course Outline

1. Course orientation and introduction to the exercises.
2. Exercise-1: develop a sample desktop application (in Java) to draw a graph for the length of words in a text file.
3. Exercise-2: develop a desktop application to handle a QR-code (generate and decode)
4. Exercise-3: convert the QR-code application to a web component (Controller).
5. Exercise-4: develop a desktop application to take a photo using the laptop's front camera.
6. Exercise-5: convert the camera application to a web component (Controller).
7. Exercise-6: develop a desktop application to generate random hash codes.
8. Exercise-7: convert the hash code generator to a web component (Controller).
9. Exercise-8: Design and develop a database for a e-payment application and write a desktop application to handle the data in it (read/write)
10. Exercise-9: Convert the desktop application to a web component (Controller).
11. Project: develop an e-payment web application. Design and code the user page (JSP)
12. Project: develop an e-payment web application. Design and code the shop page (JSP)
13. Project: develop an e-payment web application. Develop the main application controller.
14. Project: develop an e-payment web application. Integrate the QR-code and the camera controllers.

15 Project: develop an e-payment web application. Integrate the hash-code and the database controllers.
16 Not implemented.

8. Textbooks (Required Books for this course)

For this course, a set of lecture slides, handouts, and other resources will be distributed in timely manner through Moodle.

9. Reference Books (optional books for further study)

None

10. Course Goals (Attainment Targets)

(1) Examine a description provided to develop an application.
(2) Formulate a number of requirements based on the description
(3) Propose a design for the solution system
(4) Decide the best technologies for implementation
(5) Implement the design of the solution system
(6) Experiment with the of the solution system
(7)
(8)

11. Correspondence relationship between Educational goals and Course goals

Educational goals of the school			Course Goals
High level ICT skills	Basic academic skills		(1) (2) (3)
	Specialized knowledge and literacy		(4) (5) (6)
Human skill (Tankyu skill)	Ability to continually improve own strengths		
	Ability to discover and resolve the problem in society	Problem setting	
		Hypothesis planning	
		Hypothesis testing	
		Practice	
	Fundamental Competencies for Working Persons	Ability to step forward	
		Ability to think through	
		Ability to work in a team	
Professional ethics			

12. Evaluation

Goals	Evaluation method & point allocation					
	examination	Quiz	Reports	Presentation	Deliverables	Other
(1)					○	
(2)					○	
(3)					○	
(4)					○	
(5)					○	
(6)					○	
Allocation					100	

13. Evaluation Criteria

Examination	

Quiz	
Reports	
Presentation	
Deliverables	For each exercise and project in this course, every student is required to deliver a working application (executable file / web archive). The lecturer will run the application on his own computer and test its functions (as indicated in the exercise/project description) and the grade will be based on the following points: [1] the number of functions completed in each delivery [2] whether every function performs its task correctly [3] whether the function handles errors properly
Other	

14. Active Learning

Hourly percentage of active learning within the whole class time	95%

1	Active learning such as problem solving assignment using the knowledge and skills acquired in class.	All the time
2	Active learning such as group works and discussions.	Sometimes
3	Outcome presentations and feedbacks.	Not at all
4	Students actively make decisions on how the class should be conducted.	Not at all

15. Notes

This course mainly contains practical parts. Be prepared for using Integrated Development Environment and for coding.
Exercises have deadlines and they won't be postponed unless a serious issue occurs.

16. Course plan

(Notice) This plan is tentative and might be changed at the time of delivery

Lesson 1: (Course orientation, exercises) (Discussion and Lecture 90 minutes)

In this first lesson, the lecturer will brief the students on the following topics:
[1] Course syllabus
[2] Grading
[3] The exercises covered in the course

Lesson 2: (Exercise-1: word-length graph)
(Discussion & coding session 90 minutes)

In this session, the lecturer will distribute the description of one exercise on developing a desktop application that draws a graph of the length of words included in a text file. The application will run in the command line and generate the graph using standard characters (no graphical user interface is required). After discussing the description and answering any questions that might arise, students start coding the application on their laptops.

Lesson 3: (Exercise-2: desktop QR-code handler)
(Discussion & coding session 90 minutes)

In this session, the lecturer will distribute the description of the desktop application for QR-code handling exercise. After discussing the description and answering any questions that might arise, students start coding the application on their laptops.

Lesson 4: (Exercise-3: web-based QR-code handler)
(Discussion & coding session 90 minutes)

In this session, the lecturer will distribute the description of the web application for QR-code handling exercise. After discussing the description and answering any questions that might arise, students start coding the application on their laptops.

Lesson 5: (Exercise-4: desktop application to take a photo with the laptop's camera)
(Discussion & coding session 90 minutes)

In this session, the lecturer will distribute the description of the desktop application for acquiring an image through the laptop's camera. After discussing the description and answering any questions that might arise, students start coding the application on their laptops.

Lesson 6: (Exercise-5: web application to take a photo with the laptop's camera)
(Discussion & coding session 90 minutes)

In this session, the lecturer will distribute the description of the web application for acquiring an image through the laptop's camera. After discussing the description and answering any questions that might arise, students start coding the application on their laptops.

Lesson 7: (Exercise-6: desktop application to generate random hash codes)

(Discussion & coding session 90 minutes)

In this session, the lecturer will distribute the description of the desktop application for generating random hash codes. After discussing the description and answering any questions that might arise, students start coding the application on their laptops.

Lesson 8: (Exercise-7: web application to generate random hash codes)

(Discussion & coding session 90 minutes)

In this session, the lecturer will distribute the description of the web application for generating random hash codes. After discussing the description and answering any questions that might arise, students start coding the application on their laptops.

Lesson 9: (Exercise-8: design a database for an e-payment system)

(Discussion & coding session 90 minutes)

In this session, the lecturer will distribute the description of the e-payment database system and a desktop application to test it. After discussing the description and answering any questions that might arise, students start coding the application on their laptops.

Lesson 10: (Exercise-9: web application to test an e-payment system database)

(Discussion & coding session 90 minutes)

In this session, the lecturer will distribute the description of the web application to test the e-payment database. After discussing the description and answering any questions that might arise, students start coding the application on their laptops.

Lesson 11: (Design and code the user page)

(Discussion & coding session 90 minutes)

In this session, the lecturer will distribute the description of the web page to handle a user. After discussing the description and answering any questions that might arise, students start coding the application on their laptops.

Lesson 12: (Design and code the shop page)

(Discussion & coding session 90 minutes)

In this session, the lecturer will distribute the description of the web page to handle a shop. After discussing the description and answering any questions that might arise, students start coding the application on their laptops.

Lesson 13: (Design and code the application main controller)

(Discussion & coding session 90 minutes)

In this session, the lecturer will distribute the description of the web application main controller. After discussing the description and answering any questions that might arise, students start coding the application on their laptops.

Lesson 14: (Integrate the QR-code and the camera controllers)

(Discussion & coding session 90 minutes)

In this session, students integrate the QR-code and the camera controllers, which were developed earlier, into the e-payment application.

Lesson 15: (Integrate the hash code handling and the database controllers.)

(Discussion & coding session 90 minutes)

In this session, students integrate the has-code and database controllers, which were developed earlier, into the e-payment application.