Java Technologies
ICP 2152

Programming Group Project

Dave Perkins

Submission due: Friday 21st April 2017 at 11:59pm

Introduction

These notes introduce the programming group project for *ICP 2152 Java Technologies* (10 Credits). This assignment is worth **40%** of your final module mark and is therefore important. To maximise your marks be sure to plan your work so that you have enough time to complete the program and thoroughly test it. Good planning will mean that you can invest the appropriate amount of effort in completing the work to the best of your ability.

The assignment will ask you to make use of many of the technologies you have come across over the course of this semester and some new to you. The assignment should be a good practical test of your competence as well as your overall programming ability. Hopefully, the assignment will also consolidate and deepen your knowledge whilst giving you some practice in using Java web technologies as they might be applied to a real world problem.

System Requirements for Academi Gymraeg

Mastering a foreign language involves, amongst other things, learning a basic vocabulary of nouns, verbs, and adjectives. In this project we are going to develop a Web based teaching tool to assist the process of acquiring a basic vocabulary in Welsh.

To ensure a manageable workload the tool that you are asked to develop has a single purpose, namely to help learners acquire knowledge of Welsh nouns. You are not required to deal with the other parts of speech – verbs, adverbs, adjectives etc. To undertake this exercise you need to know a little Welsh grammar all of which is covered in the Appendix.

Your program will maintain a file or perhaps a data base recording the gender and meaning of a specified list of Welsh nouns (see Appendix for where the list can be obtained).

Your task will be to produce a web front end which allows a user to register as a student of our imaginary *Academi Gymraeg*. Registered users then have the opportunity to test their knowledge of Welsh vocabulary by taking vocabulary tests. Each test consists of twenty randomly generated questions of a specific form. Questions take one of three forms:

- What is the gender of the Welsh noun tatws?
- What is the meaning of the Welsh noun *tatws*?
- What is the Welsh noun for the English word for potato?

For the last type of question the student must provide the noun with the correct gender marker. When a student has completed the test a mark out of twenty should be displayed.

Users of the system are classified as students, instructors or system administrators. Only system administrators of the *Academi* can register users and only instructors can add, delete or modify entries in the vocabulary list. You should therefore provide a *minimal level* of security to prevent users accidentally or otherwise destroying or corrupting data. This will involve associating usernames with passwords so as to ensure that only authorised users have access to the database.

Because the application is meant to be used in the context of an educational organisation the system will use session information to permanently record the marks of each user. Users can have access to their own grade history whilst system administrators and instructors have access to the grade history of all users.

Your site should have the following:

- A welcome page which provides links to the other pages of your site.
- A page for adding new words to or removing words from the database.
- A page for editing entries so as to correct typographical mistakes.
- A page which enables users to take a test.
- A page which displays a user's test history.

You may use one or more pages in order to achieve any of the above objectives. You may also add any additional functionality you wish but do **remember to fulfil the core requirements!** If you want to get marks for something extra please discuss your ideas with the lecturer before proceeding.

How you develop your solution is to some extent up to you. For instance, there is no necessity to use a database to store information about nouns, the vocabulary list could instead be maintained in an XML document. However, whatever implementation route you select, be prepared to explain your choice during the presentation.

Your web application must be implemented using *at least two* of the technologies i.e. Servlets/Web-services (see extra lab notes), JDBC or XML or JSF (see additional lab notes), e.g. you might use a MySQL database, together with the JSF framework. Before making a final decision on technologies you *must* consult with the course coordinator. You will need to use Maven, but this does NOT count as one of the technologies.

Marks in this assessment will be awarded on the following basis:

- Program structure and design (10%)
- Easy to use interface (10%)
- Correctness (20%)
- System security (5%)
- Internal program documentation (5%)
- Appropriate extra functionality (10%)
- Installation process (10%)
- Documentation (20%)
    - 10% for user guide
    - 10% for installation guide
- Presentation (10%)

NB. Students who are unable to explain how their program works will be *heavily penalized*. Therefore, it is suggested, that you study your program carefully before attending a demonstration session.

Submission

The code for your finished site must be packaged up in such a way that it is ready to be deployed to a Tomcat server immediately. I*f your site doesn't run without error, you will lose marks*. You are required to submit your NetBeans project (this will include your web application directory containing your compiled Servlets etc).

You are also required to produce documentation containing the following information:

- A short description of the login system for students, instructors and administrative users (e.g. rules for valid passwords, sample password files etc)
- A set of UML diagrams describing the static and dynamic components of the system
- A description of the database design (or XML file structure) together with the table contents (or the XML document containing the vocabulary list)
- A short user guide describing the functionality provided by your site e.g. a description of each page present and what it does. Make sure to use screen shots when describing the operation of your system.
- A short installation guide. This will be used to install your application on the departmental server, this should be operating system agnostic. You will demonstrate your work from this installation.
- Include your word list for testing purposes.

Create a zip archive containing your code and documentation and submit this via Blackboard.

Group Project Organisation

Please take note of the following points:

- This assessed work can *only* be submitted on a group basis
- Groups must contain 3 members
- Groups should be notified to the course coordinator as soon as possible
- Each member should have a clearly defined roles
- Possible roles might include
    - Web site designer
    - Project coordinator
    - Chief programmer
    - Programmer
    - Designer
- Each member must be involved in coding and be responsible for one or more classes.
- At the end of the project the group should meet and assess the contribution of each group member (more information about this aspect of the project will be provided)

Appendix 1: Welsh Nouns

Welsh like many other languages assigns a gender – masculine or feminine - to each of its nouns. Consider the following examples:

- **Tatws**   potato   feminine
- **Cyfrifiadur**   computer   masculine
- **brechdan**   sandwich   feminine
- **myfyriwr**   **student(m)**   masculine
- **myfyrwraig**   **student(f)**   feminine

To obtain a dictionary of welsh words try the site below:

http://geiriadur.bangor.ac.uk/

Alternatively use the list contained in the Word document accompanying these notes.

Please note that your noun list *is your own – you will need to create this resource, you could start with a list such as http://mylanguages.org/welsh_vocabulary.php but may need to look up the gender.*

You will need a list that is sufficient to allow your progam to be fully tested, this may mean around 30-40 words at a minimum.

Not:e there are some different characters in Welsh you will need to consider this in your implementation.

****