University of Westminster – School of Computer Science and Engineering

Assessment for 5COSC021W/5COSC003W Software Development Group Project (subject to External Examiner's approval)

Project Brief

In this coursework, your team will be working on a software application development project. The software application will be for students in the School of Humanities who are learning a foreign language (e.g. French or German). The students will typically use the application in class for practising and developing their spoken language. For this, the application will provide a context for conversation and allow a student-teacher pair to practice certain phrases within that context. The role and the requirements of this application are detailed in the next section.

The client for the project, as implied above, is the School of Humanities at the University of Westminster. You will have the opportunity to meet the representatives of the client in designated classes to clarify any questions or gaps. Your team's job is to create this application. The software produced will be available as open source and most of the media work will be available through Creative Commons.

Support Structure

Project Manager aka Module Leader:
Strategic planning, support of tutors.

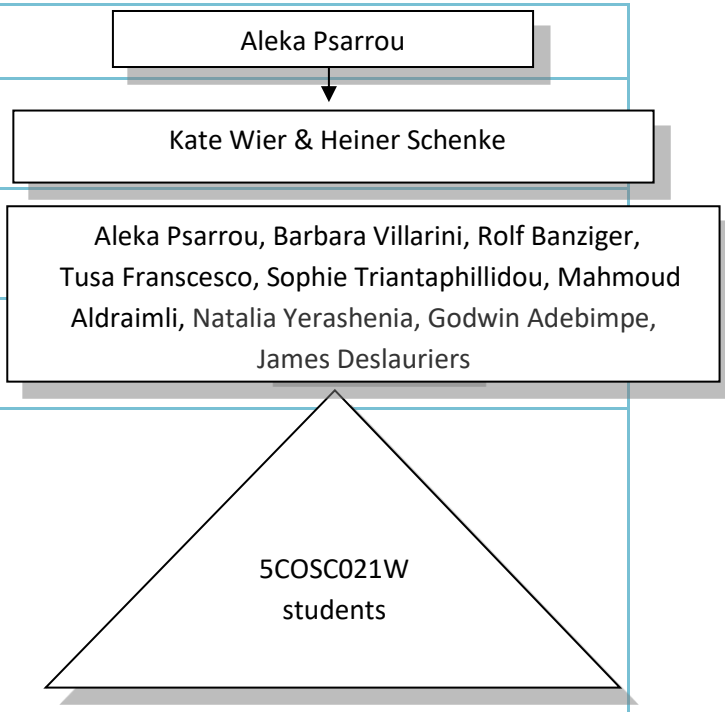
Client manager / representative:
To provide requirements and clarify any questions.
They will join the module at designated times. You will be informed of the timings.

Senior Managers / Mentors aka Tutors:
Weekly support and guidance for project teams (that's you), solve (rarely) team issues.
Evaluation / appraisals of project teams and all work you produce and your professional conduct.

Project teams aka you in groups:
Team members will provide peer feedback to each other, test each other's work, point out flaws that need fixing.

Project team member aka each of you:
Each of you will work as part of a team but you will have both group and individual tasks to do that contributes to the overall project. The allocation of tasks in each team member is decided by the team. Collectively as a team you will:

- model the data (including users) using conceptual and logical diagrams
- design the use cases, sequence diagrams, class diagrams and test plans for your use cases
- read the data from a file into an SQL structure that suits your specific problem and functionality required
- design and develop a desktop front-end for a mobile compatible web app on Java swing
- develop appropriate database structures and queries to handle user data (eg login, administration of user accounts, retrieve and graphically display user interactions with data and other users)



Coursework description

Your work will be assessed in coursework 1 (design) and coursework 2 (implementation).

The application

The application facilitates the learning of a foreign language by providing a conversation application that generates context- and level-specific stimuli for both parties in a 2-person conversation with each person only having access to their part of the conversation. Language learners will be able to search for appropriate role play stimuli by context, language function and level. All stimuli will be produced in English. The application will provide vocabulary support in the target language (eg Spanish, French or German). The application will also need to support panels for the login of the users, administration of the users, and monitoring panels to display the history of the interactive conversation between a student-teacher pair, and to display the student's progress and/or level reached for each context-language function pair in the target language, each with some appropriate analytics function. As this is the first prototype of the application, you are not asked to implement the functionality of sharing the space with a teacher, but instead a separate panel will need to be implemented that shows the teacher's front-end on your device.

The language application will comprise an interface which enables language learners to search by context, language function or level. For example, language learners may choose to practise a range of dialogues related to one context eg transport, or they may prefer to practise making an apology across a range of contexts ("I'm sorry, I can't come to your party", "I do apologise, I don't have enough money") or may prefer to practise a range of contexts and language functions across a particular level. The application will record several information. For example, when and how often a learner has engaged with a task, any vocabulary support that has been accessed, or how many times a task has been visited by students.

Each team will need to develop a working, stand-alone Java application, making use of the knowledge you have gained in level 4 and the semester 1 core modules this academic year: eg Web development, Object oriented programming and Databases. The functionality of the application has a number of parts: a) login into the application and administration of the users' login activity b) be able to read data from a database and display data as to facilitate a dialogue between a student and a teacher c) develop panels to record when and how often a learner has engaged with a task with appropriate analytics, and d) a panel presenting a summary of the performance of the student, with appropriate analytics . Broadly your application should include panels that facilitate the following functionalities; however, it is the responsibility of each team to understand the full requirements of the project from the project owners:

1. Allow users of your application to register with username and password
2. Allow users to enter the application using their username and password
3. Log the user's activity (name of user, time stamp of login and logout to the application)
4. Allow administration of the users' data and display of user activity
5. Allow users to select a language and teacher/other student from other registered users
6. Read data from the data files into appropriate database structures which you will design – the database design can be created using SQLite (or mySQL), displaying both the student and the teacher/other student interface. This functionality should facilitate the language practice activities as described by the project owners.

7. Implement student progress monitoring panels, eg one that monitors how often a student-teacher pair has engaged with a task, one that monitors the performance of the student, both with appropriate analytics functionality.

The coursework includes both group and individual tasks.

Group tasks: As a team you are responsible for the design of the application and the implementation of the login panel and the data structures to store the language data, the users' login data. The parts developed individually (see individual tasks below) should be able to link to create a complete application. As a team you are responsible to connect all the individual panels and display the results to the user as one application including the teacher panel, the login/admin front-end and the front ends for monitoring. There are also group report writing tasks. All members of the group are responsible for all group elements, however, you can allocate and split the GROUP work as you see fit, between yourselves. The number of panels displayed after the user is logged in will depend on the number of students in each team.

Individual tasks: In addition to the group tasks above, each member of the team needs to individually implement additional panel(s) that implement the functionality of the application. Suggestions include but are not limited to: one member to implement the panel(s) that selects the language and the teacher/other student, one member of the group will implement the language panel(s) to select topic/level/conversation for student and teacher, one member will implement the panel(s) to administer users' data with appropriate analytics (not on mobile platform format), one member to implement the panel(s) that monitors the performance of the student with appropriate analytics, one member will be responsible to implement the monitoring panel that presents how often a student-teacher pair has engaged with a task with appropriate analytics, one member to monitor how often different topics/ conversations at various levels are invoked with appropriate analytics. There are also individual report writing tasks. If a group has fewer members then the last panel can be omitted.

Working in a group

Everyone in the same team will be using the same source data. Talking to each other is encouraged. Checking each other's work and providing feedback to each other is part of the coursework itself: it is something you will be expected to do. However, doing each other's individual work is not allowed; it will be against university regulations and appropriate penalties will follow. However, as you need to do collaborative software development you would need to agree on interfaces between structures,

During the duration of the coursework you will need to comment on each other's work. These comments will have to be recorded on-line on one or more of the following platforms: the module's blackboard site, we have created a group on BB for each team, or on the Trello board you share with your team where you can comment on activities, or in any other communication tool that your team may be using (eg WhatsApp, Discord etc). You will also need to take on board the comments of your team-mates on your own work and specify how you used them. You need to keep all the communication between the team members. You will be required to submit some of this as evidence of collaboration in your coursework submission. In addition, your tutor/module leader may request additional evidence in case of a dispute between members of a team.

Submission of coursework

See detailed Coursework 1 and Coursework 2 briefs below.

Coursework 1 Project Design

Coursework 1 administration

Module Leader	Aleka Psarrou
Unit	Coursework 1
Weighting	40%
Qualifying Mark	30%
Description	Group coursework (70% Team, 30% Individual)
Learning Outcomes Covered in this Assignment	LO1 gain practical experience of software project management throughout the software lifecycle with analysis of a problem domain towards a solution that can be demonstrated to be ready for beta-testing; LO3 be able to design/apply appropriate testing methods and tools to evaluate software performance; LO4 gain practical experience of working as a member of a software development project team demonstrating collegiality and professional standards of conduct in communication, time management, project documentation and version control;
Available on Blackboard	Monday 24 January 2022/ Updated after minor moderation comments on Monday 31 January 2022
Due date	**Thursday 3rd March 2022, 1pm – Each student should submit BOTH the group and his/her individual templates (TWO documents).**
Start thinking about it	Now
Expected Deliverables	No later than the due date and preferably a few days earlier each student needs to upload on Blackboard both the group and individual templates that include: • The conceptual model representing the data in the problem domain • The use cases, sequence, and class diagrams • Your proposed test plan for each of the use cases • The wireframe and high-fidelity interface design of your application • The comments you received from your team-mates about your application and how you dealt with them • The comments you have given your team-mates about their applications • Your record of engagement with the module demonstrating time management and version control Added regularly on your project management and communication sites: • Minutes of meetings and agreements • On-line discussions • Research links **On week 7 (during your tutorial or as arranged by tutors):**

	• **Assessed presentation [All team members must be present. Any team member that is not present will receive a maximum mark of 30%. Submission of MCs is required if you cannot present on that day]**
Method of Submission	Online via Blackboard. Instructions will be on Blackboard well before the deadline.
Feedback before submission	During tutorials from tutors (verbal) and from team-mates (peer feedback, verbal and online)
Feedback after submission	• Feedback from tutors (verbal) • Written feedback and marks 15 working days (3 weeks) after the submission deadline. All marks will remain provisional until formally agreed by an Assessment Board.
BCS Criteria covered in this Assignment:	2.1.1 Knowledge and understanding of facts, concepts, principles & theories 2.1.2 Use of such knowledge in modelling and design 2.1.3 Problem solving strategies 2.1.6 Recognise legal, social, ethical & professional issues 2.1.7 Knowledge and understanding of commercial and economic issues 2.1.8 Knowledge of management techniques to achieve objectives 2.1.9 Knowledge of information security issues 2.2.1 Specify, design or construct computer-based systems 2.3.1 Work as a member of a development team 2.3.2 Development of general transferable skills 3.1.2 Methods, techniques and tools for information modelling, management and security 3.2.2 Defining problems, managing design process and evaluating outcomes 3.2.3 System Design
Assessment Regulations	See last section of this document on regulations

Coursework 1 marking scheme

The Coursework will be marked based on the following marking criteria:

Criteria	Marks per component[1]	Marks provided	Comments
Submitted in CWK1 GROUP template (70/100)			
Logical ERD			
Accuracy and completeness of model - should include both data and users (it should not have mistakes; it should not have data missed)	10		
User interaction design			
Story boards (wireframe and mock up)	10		

[1]Do your best in all sections; those who go for the minimum pass mark in each may never survive in industry and may end up unemployed.

UML diagrams			
Use case diagrams	10		
Sequence diagrams	15		
Class diagrams	15		
Use case test plans			
Use case test plans	10		
Submitted in CWK1 INDIVIDUAL template (30/100)			
Professional conduct – Communication			
How you engaged with your team-mates to get their comments on your work and how you dealt with their comments professionally	10		
The comments you have given your team-mates (including your justification for each)			
Time management and documentation version control			
Your reflection on the time management and project management process for coursework 1, discussing both good examples and areas for improvement	10		
Your reflection on the group documents version control, discussing both good examples and areas for improvement	10		
Total	**100**		

Coursework 2 Project Implementation

Coursework 2 administration

Module Leader	Aleka Psarrou
Unit	Coursework 2
Weighting	60%
Qualifying Mark	30%
Description	Group work – (70% Individual, 30% Team)
Learning Outcomes Covered in this Assignment	LO2 following distinct software development phases and considering and applying HCI principles, produce code that is clear and easy to integrate. LO3 be able to apply appropriate testing methods and tools to evaluate software performance; LO5 evaluate security risks and demonstrate how computer security decisions affect software development. LO6 be aware of ethical and legal considerations of a software project, including issues of intellectual property and data protection
Available on Blackboard	Monday 24 January 2022/ Updated after minor moderation comments on Monday 31 January 2022
Due date	**Thursday 5 May 2022, 1pm – each student should submit code and both group and his/her individual templates - groups should also upload video of work on google drive**
Start thinking about it	Now

Expected Deliverables	No later than the due date and preferably a few days earlier upload on Blackboard:
	• The **code (all project files)** with the implementation of your system, incorporating comments by your tutor from previous **tutorials (the tutors should be able to run your program by just clicking on your submitted work)**
	• the coursework 2 group and individual templates that include:
	  ○ discussion of the functionality and quality of your code
	  ○ The output of implementing your test plan
	  ○ Your list of HCI issues you have addressed (and how you addressed them) and any HCI issues still remaining
	  ○ Your list of security risks you have addressed (and how you addressed them) and any security risks still remaining
	  ○ Your plan to manage legal and ethical constraints appropriate for the problem domain, supported by appropriate research
	  ○ The comments you received from your team-mates on the quality of your code and documentation and how you dealt with them
	  ○ The comments you have given your team-mates on their code, documentation and evidence of version control
	• A 5-10 minute video of your group application on google drive
	On date set by your tutor:
	• **Assessed group demonstrations [All team members must be present. Any team member that is not present will receive a maximum mark of 30%. Submission of MCs is required if you can't demonstrate your work on that day]**
	Added regularly on-line:
	• Minutes of meetings and agreements

	• On-line discussions • Research links
Method of Submission	Online via Blackboard / video on google drive
Feedback before submission	During tutorials from tutors (verbal) and from class (peer feedback, verbal and online)
Type of Feedback and due date after submission	Written feedback and marks 15 working days (3 weeks) after the submission deadline. All marks will remain provisional until formally agreed by an Assessment Board.
BCS Criteria covered in this Assignment:	2.1.1 Knowledge and understanding of facts, concepts, principles & theories 2.1.2 Use of such knowledge in modelling and design 2.1.3 Problem solving strategies 2.1.5 Deploy theory in design, implementation, and evaluation of systems 2.1.9 Knowledge of information security issues 2.2.1 Specify, design or construct computer-based systems 2.2.2 Evaluate systems in terms of quality and trade-offs 2.3.1 Work as a member of a development team 2.3.2 Development of general transferable skills 3.1.2 Methods, techniques and tools for information modelling, management and security 3.2.2 Defining problems, managing design process and evaluating outcomes 3.2.3 System Design
BCS Criteria covered in this Assignment:	
Assessment Regulations	See last section of this document on regulations

Coursework 2 marking scheme

The Coursework will be marked based on the following marking criteria:

Criteria	Mark per component[2]	Mark provided	Comments
Submitted in CWK2 INDIVIDUAL template (70/100)			
Code			
Code functionality – individual element	15		
Code functionality – integrated element (group work (database and login panel) and functionality of final prototype)	15		
Code quality - maintainability	10		
Code quality – version control	10		
Code quality – output of test plans	10		
Professional conduct – communication			
The comments you received from your team-mates on the quality of your code and documentation how you dealt with them	10		
The comments you have given your team-mates on their code and documentation			
Submitted in CWK 2 GROUP template (30/100)			
Human Computer Interaction			
Your list of HCI issues you have addressed (and how you addressed them) and any HCI issues still remaining;	10		
Security risks			
Your list of security risks you have addressed (and how you addressed them) and any security risks still remaining;	10		
Professional conduct – Legal and Ethical			
Legal constraints and how you will manage them, as appropriate for the problem domain, supported by appropriate research	10		
Ethical constraints and how you will manage them, as appropriate for the problem domain, supported by appropriate research			
Total	**100**		

[2]Do your best in all sections; those who go for the minimum pass mark in each may never survive in industry and may end up unemployed.

Coursework tips

a) In the second year of your degree you should be able to resolve a lot of problems by yourselves. If you do get stuck, look at the later section about 'managing issues'. Don't go to your tutor for every question you could find the answer yourself; you are preparing to be independent professionals.
b) Independent professionals who disappear and no one knows what they are doing, are not professionals and sooner or later get fired. Make sure your tutor knows what you are doing every week (i.e. attend tutorial slots).
c) You should manage and complete the work as a team. You will have key points where you will show your work to your team-mates to get their feedback. There are more marks for a collegiate, helpful, behaviour to others (without doing their work for them!) There will be marks lost for ignoring your team-mates, working in isolation, being late with your work, and so on.
d) Identifying team issues early and managing them is an important part of the module. Minutes of meetings and agreements must also be updated online regularly for your tutor (tutor) to see but also as a record for the team of what you have discussed and agreed.

Managing issues

All work, however well planned, can encounter something unexpected. It is important that you see such issues as learning opportunities and that you manage them, from beginning to end, **professionally**.

To resolve an issue yourself:

a) Do you have all the information about the problem? If not, try to find out the full story.
b) Could you do something differently yourself and solve the issue? The success of the project will affect you so it is to your interest to adapt if you can.
c) If the team needs to do something differently, can you convince the group what to do? How can you convince them that what you want will benefit them?

Raising a case with your tutor:

a) Have you exhausted the steps for resolving it yourself?
b) Plan ahead what you want to ask and what you want from them (guidance? action? decision?).
c) Be respectful of your tutor's time. Your access to them is during the tutorial slot and – only very exceptionally and if it is urgent – by email.
d) Be respectful to your colleagues. All resources are shared and limited, and that includes your manager/tutor. Ensure you are not asking for so much time that others will have to get little or nothing. (It's actually unprofessional.)
e) Separate private issues (eg "I have a doctor's appointment next week" is something you should ask to discuss privately), from professional issues ("I'm not clear on how to do/handle this").
f) Most of the times your tutor will not solve your problem but will try instead to help you solve it yourself, so you know how to do this in the future.

Raising a case with your module leader:

a) Any communication to the module leader must show that it has exhausted all avenues of resolving the issue yourself or with your tutor.

What do you need to do to get a good mark?

Read this document carefully. Manage your work so that you do, typically, another 10 hours of work per week, outside lectures and tutorials. Go to lectures and go to tutorials as you can't get anywhere else the help and information you need to do well. Seek feedback from your tutor and your teammates. Use that feedback to improve your work. Don't miss what is on blackboard. Log on blackboard blog what you are doing. Polish your work: document well, write clearly, pay attention to the little details. Always leave enough time to look at your work again and fix any missed errors.

What do you need to do to get a bad mark?

Read this quickly and expect someone else will tell you what you missed. Only do a couple of hours of work a week on this module. Don't come to lectures expecting that you will pick things up from others or from blackboard. Ignore blackboard announcements. Write badly and submit work you did the night before the deadline. Ignore your team mates and their feedback – and give them low quality feedback in return. Assume that you can copy from someone else. Assume that the answer can be found on the internet (it cannot). Miss tutorials and the opportunity to get useful feedback from your tutor. Leave work for the last minute. Come to the presentation unprepared. Think that this is an easy coursework (it is not).

Managing issues

All work, however well planned, can encounter something unexpected. It is important that you see such issues as learning opportunities and that you manage them, from beginning to end, **professionally**.

Regulations

Assessment regulations

Refer to part 3 of the Academic Regulations for a clarification of how you are assessed, penalties and late submissions, what constitutes plagiarism, etc.

Penalty for Late Submission

If students submit coursework late but within 24 hours of the published deadline, the work will be marked and will have ten percentage points of the overall available marks deducted, to a minimum of the pass mark (40% at undergraduate level). If students submit coursework more than 24 hours after the specified deadline, a mark of zero will be awarded for the work in question – unless a claim of Mitigating Circumstances has been submitted and accepted as valid.

For more detailed information regarding University Assessment Regulations, please refer to the following website:
http://www.westminster.ac.uk/study/current-students/resources/academic-regulations

Attendance at the time and date of your presentations/demonstrations

All members of a team should be present and take part at the scheduled group presentations and demonstrations. If any student does not attend, the student's mark will be capped to 30%. If a student can't attend due to extenuating circumstances he/she needs to submit MCs claim.

Finally…

There is nothing saying that you cannot do **very** well in this module. You all start with an equal chance to do very well, regardless of how you did last year. But how well you do is up to you and how professionally you are engaging with all aspects of the module. There is a lot of learning to be done (about 200 hours worth) with very interesting stuff directly related to how you survive in your future employment. Take it very seriously, but also try to enjoy it.