University of Westminster – School of Computer Science and Engineering

Assessment for 5COSC021W 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 regarding the development of a web-based inventory application, to track the number and type assets available and loan of assets. The functional and non-functional requirements are given in the application brief section of this document. The client for the project is School’s Technical Unit. You will have the opportunity to meet the representatives of the client in designated classes to clarify any questions or gaps. We have also organised meetings with industry mentors to discuss with you issues related to group work and also provide feedback on your work. Your team’s job is to create this application.

## Project Support Structure

|  |  |
| --- | --- |
| **Project Manager aka Module Leader:** Strategic planning, support and guidance for the client and support for the tutors. | Aleka Psarrou |
| **Client manager / representative**: To provide requirements and clarify any questions.  CSE Technical Unit  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, the work the teams and their members produce and their professional conduct. | Aleka Psarrou, Barbara Villarini, Kumudini Sarathchandra, Birzo Ismael, Tusa Francesco, Philip Trwoga, Mohammad Shah, Hari Konda Ramamoorthy, Koushik Modak, Aran Hakki , Natalia Yerashenia, David Huang |
| **Project teams aka you in groups**: Team members will create project plan together, provide peer -review on each other’s work, 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 logical diagrams * design the use cases, activity diagrams and test plans for your use cases * design relational database structures that suit the specific problem and functionality required * design and develop a web-based front-end to meet the requirements of the project, design a site-map * develop appropriate back-end functionalities, using appropriate programming or query languages, to handle the functional requirements of the application | 5COSC021W students |

**Application brief – CSE Equipment Inventory**

You are designing the School’s Inventory Management web Application with the following functional and non-functional requirements:

**Functional Requirements**

## Users

* User must be able to sign up for an account, once approved update that account information (only design user requirement not implementation). (Simple sign up functionality provided by skeleton)
* User must be able to securely log in to access the system (Simple login functionality provided by skeleton)
* User should have the ability to view list of equipment
  + User should be able to filter this list by availability, date return date, type of equipment
  + User must be able to sort this list by predefined fields
* User should have the ability to search for equipment based on various fields like ID, Asset Tab, name or type
* User must be able to reserve i.e. book/check out equipment for CW/FYP/other study related tasks. Bookings be approved/rejected by an admin users. (only design user requirement not implementation
* Users should be able to receive alerts when equipment is overdue to be returned (only design user requirement not implementation)
* Users must be able to view their current reservations and bookings as well as historical bookings.
  + Users must be able to cancel a reservation
  + Users should be able easily reserve equipment from a historical booking
* Not all inventory items can be borrowed or lent – items to be used only onsite should be marked as such as an attribute.

## Admins

* Admin must have their own sign area with secure log-in or be differentiated in a way that gives them access to administrative level functionalities (Simple admin functionality provided by skeleton)
* User should have the ability to view list of equipment
* Admin must have the ability to add new pieces of equipment with all the details
* Admin should be able to update existing equipment details
* Admin should be able to approve new user sign ups, add, remove or update user accounts and roles (from regular user to admin user)
* **Admin must be able to generate various types of reports, such as current inventory status, equipment usage history, warranty, overdue equipment etc.**
* **Admin must be able to get inventory count of equipment, for overdue equipment as well as various categories/type.**

## Non-functional requirements

* System should separate business logic with application logic;
* System must be built using Django
* System should store data in DBMS (SQLite)
* The system should be able to handle multiple users with fast response times (not an implementational requirement)
* The user interface should be user-friendly, intuitive and can be learned quickly by all users
* System must be able to encrypt data communications (not an implementational requirement)
* The system should be able to backup and recover its data in case of system failure; whether manual or automated (not an implementational requirement)

**Coursework tasks:**

Your work will be assessed in coursework 1 (design) and coursework 2 (implementation). The courseworks include both group and individual tasks.

**Group tasks**: As a team you are responsible for the design of the application and the implementation of the data structures/database to store the application data identified (for example, user data, assets data, booking information). 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 front-end and back-end systems to display the results to the user as one. There are also group report writing tasks. All members of the group are responsible for all group elements, however, you can allocate and split leading the GROUP work as you see fit, between yourselves. However, all group members are expected to either actively review or provide feedback to the group work. Students that don’t somehow participate to a part of the group work will not get marks for that part.

**Individual tasks**: In addition to the group tasks above, each member of the team needs to individually implement one or more functionalities of the application, typically by creating its own web-pages (apps) and link it to the database. This allocation will be discussed and proposed by the team but agreed by the tutor during cwk1. Suggestions include but are not limited to: one member of the group to implement the viewing of the equipment and filters, or one to implement the reservation functionality, one to implement the booking functionality, implement booking history, implement updating equipment list, implement updating user accounts, for a user to view their current bookings/reservations, or for the admin to generate reports on inventory status, equipment usage history, overdue equipment etc

Please note: The framework to be used is Django, with SQLite.

### Working in a group

Everyone in the same team will be using the same data accessed through a commonly designed database. 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. 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: on the Trello board you share with your team where you can comment on activities (essential), and in any other communication tool that your team may be using (eg WhatsApp, Discord, Slack 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 Software/System 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, 22 January 2024 |
| **Due date** | **Thursday, 29th February 2024, 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 logical model representing the data in the problem domain * The use cases of the functional requirements given by the client * Prioritise the use cases you will implement and design mock-ups/ site-maps and activity diagrams for your application * Your proposed test plan for each of the use cases you have prioritised * 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 Trello project management 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]**   **Students in a group need to work together and submit the same group template. Any students that don’t work with the group and submit group template that is different from that submitted by the rest of the group will have their mark capped to 30%. Any disputes that may result of any group members submitting different group templates need to be brought to the attention of the tutor early in order to be resolved. Students need to provide clear reasons of the dispute supported by evidence together with attempts by the students to resolve the dispute.** |
| **Method of Submission** | Online via Blackboard. Instructions will be on Blackboard well before the deadline. Failure to not submit group or individual template will be seen as no submission of that component. |
| **Feedback before submission** | During tutorials from tutors (verbal), mentors (week 5) 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  2.1.8 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]](#footnote-0) | 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 missed data) (LO1) | 15 |  |  |
| **User Experience** | | | |
| Site-map, mock-up (LO1) of prioritised use cases | 15 |  |  |
| UML diagrams (LO1) | | | |
| Use case diagram(s) for all functional requirements identified by client | 15 |  |  |
| Activity diagram(s) of prioritised use cases | 15 |  |  |
| Use case test plans (LO3) |  |  |  |
| Test plans for prioritised use cases | 10 |  |  |
| Submitted in CWK1 INDIVIDUAL template (30/100) | | | |
| Professional conduct – Communication (LO4) | | | |
| 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 project management (LO4) | | | |
| Your reflection on the time management process for coursework 1, discussing both good examples and areas for improvement | 10 |  |  |
| Your reflection on the group’s project management 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 – (60% Individual, 40% 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, 22 January 2024 |
| **Due date** | **Thursday 2 May 2024, 1pm – each student should submit code and both group and his/her individual templates - groups should also provide a link on the submission to a video where their work is described** |
| **Start thinking about it** | Now |
| **Expected Deliverables** | No later than the due date and preferably a few days earlier upload on Blackboard:   * The **code – upload all files required (incl databases) so that your code can run when uploaded as Django project** * the coursework 2 group and individual templates that include:   + Discussion on the functionality and quality of your code   + Discussion on how individual parts were integrated to produce a single application   + Discussion on how you used version control and ensured compatibility between work of group members   + The output of the test plan for your individual work and the overall application   + Discussion on the main UI/UX principles you applied in your implementation.   + 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   + SQLlite database(s) for user accounts and equipment * A link to a 5-10 minute video of your group application * Added regularly on-line on Trello:   + Minutes of meetings and agreements   + On-line discussions   + Research links   **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]**   **Students in a group need to work together and submit the same group template. Any students that don’t work with the group and submit group template that is different from that submitted by the rest of the group will have their mark capped to 30%. Any disputes that may result of any group members submitting different group templates need to be brought to the attention of the tutor early in order to be resolved. Students need to provide clear reasons of the dispute supported by evidence together with attempts by the students to resolve the dispute.** |
| **Method of Submission** | Online via Blackboard including a link to the video. All students should submit both group and individual templates and link to the group application (or their own part if students did not manage to link their part to the group application). Failure to not submit group or individual template will be seen as no submission of that component. |
| **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 |
| **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]](#footnote-1) | Mark provided | Comments |
| Submitted in CWK2 INDIVIDUAL template (60/100) | | | |
| Code (LO2) | | | |
| Code functionality – individual element and integration to the application | 20 |  |  |
| Code quality – maintainability | 10 |  |  |
| Code quality – version control | 10 |  |  |
| Code quality – output of test plans (LO3) | 10 |  |  |
| Professional conduct – communication (LO4) | | | |
| 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 (40/100) | | | |
| Code (LO2) | | | |
| Code functionality – database implementation | 10 |  |  |
| Human Computer Interaction (LO2) | | | |
| Discuss principles of UI/UX in the implementation and any issues of consistency across group members’ implementation | 10 |  |  |
| Security risks (LO5) | | | |
| 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 (LO6) | | | |
| 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** |  |  |

### Coursework tips

1. 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.
2. 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).
3. 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.
4. 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:

1. Do you have all the information about the problem? If not, try to find out the full story.
2. 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.
3. 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:

1. Have you exhausted the steps for resolving it yourself?
2. Plan ahead what you want to ask and what you want from them (guidance? action? decision?).
3. 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.
4. 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.)
5. Separate private issues (e.g. “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”).
6. 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:

1. 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.

1. [↑](#footnote-ref-0)
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. [↑](#footnote-ref-1)