****

**Faculty of Engineering, Environment and Computing**

**6002CEM Mobile App Development**

**Assignment Brief and Grading Rubric 2020/21**

|  |  |  |  |
| --- | --- | --- | --- |
| Module Title  **Mobile App Development** | Ind/Group  **Individual** | Cohort  **January** | Module Code  **6002CEM** |
| Coursework Title  **Designing and Building a mobile App (CW2)** | | | Hand Out Date  **04/01/21** |
| Lecturer  **Xin Lu** | | | Due Date  **09/04/21**  **Online: 18:00:00** |
| Estimated Time (hrs):  Word Limit\*: | Coursework Type  **Individual Practical Element** | | % of Module Mark  **75%** |
| Submission Arrangement Online via Aula: **A text file with 2 links (Video Link and Github Repository) upload through Assignment link**  File Type:  **A text/word/pdf file with two links**   * **A 8-min demonstration video upload to GoogleDrive/OneDrive/Youtube** * **A Coventry University GitHub Repository link where hosted your App project**   **Note: In summary two links (video and github links) should put in your submission file**  Mark and Feedback Date: 19/04/21  Mark and Feedback Method: **Rubric marks and comments**  **Failing to submit either the Coventry University Github Repository link or the video link will result in a zero Mark for this assessment** | | | |

|  |
| --- |
| **Module Learning Outcomes Assessed**  1. Design and implement a mobile application, showing systematic knowledge of relevant tools, methods and processes.  2. Understand and critically evaluate the different tools and techniques for mobile application development.  3. Scope, design, implement and critically evaluate a basic security policy to keep confidential data safe on a mobile device.  4. Demonstrate an ability to maintain ease of data access/usability across several platforms.  5. Understand, articulate, and provide sustained argument on how Cloud Computing can be used to enhance the use of mobile computing in business. |
| **Resit**  If you fail to submit or are awarded a grade of less than 40% (CW1+CW2) you will be required to complete a resit assignment. This will be released after the semester 2 exam period. |

|  |
| --- |
| **Task and Mark distribution:**  **Assignment Brief:**  You are required to design, build and test an **Android or iOS** app that demonstrates your proﬁciency in the skills that have been taught during the module. You can free to choose use Android studio or Xcode to design your App.  You can think of and develop any kind of app that you feel will be useful for the potential user and also aligns well with your own interests. But the app you decide to build should give you sufficient scope to demonstrate the required skills such as using different views, layouts, controls, Web APIS, Sensors and hardware along with data persistence to address the learning outcomes of the grading rubric below. Example app ideas include:   * A University tour guide to help people find important places and useful information. * A treasure hunt app that provides a clue to a location within the University and City. Once the user has arrived at that location it issues a new clue. This should be aimed at University freshers. * A paper-trail app. One person is nominated to escape, and the rest of the users are trying to find him or her.   As mentioned above, the main app’s idea is totally open for you to decide and implement based on your own understandings and interests of specific domains and needs that you wish to address as long as you apply the features, frameworks and models taught during the labs.  You may use sample code from the lab exercises, but this must be clearly labelled in your source code and you will only be assessed on the code you have written yourself.  **Submission file requirements:**  **Video:**  You are required to record a screencast in which you are expected to both demonstrate the features of your app and also explain key code written for testing and developing these features. You will only be awarded marks for features demonstrated in the video and also meet the requirements of the grading rubric. There are a number of restrictions:   1. The video must last no longer than 8 minutes 2. The video quality must be at least 720HD 3. Show your app running on the screen and how to use it 4. Explain features of your app and how those meet your design requirements 5. Show parts of your source codes that implement the features mentioned above. 6. The audio needs to be in high-quality 7. The video must be produced in high quality so that technical details such as source codes can be clearly viewed. The video quality must be at least 720HD. 8. The video link must be uploaded on **GoogleDrive/OneDrive**/**Youtube** and provide the link in submission file.   **University GitHub Repository:**  You must also provide a link to your Coventry University GitHub repository that contains all source codes for your app in your report. Your code repository should be hosted at:  <https://github.coventry.ac.uk/6002CEM-2021JANMAY>  The name of repository must be **6002CEM-xxxxxxx** where xxxxxxx is your student ID number and your name. Note this must be your Coventry University GitHub repository (github.coventry.ac.uk) and **NOT** the public GitHub repository (github.com) |
| **Note**  For Android app, you must build the app with Native Android SDK. DO NOT use commercial software and / or 3rd party packages / libraries. For example, using Butter Knife libraries or Adobe PhoneGap to create your application is not allowed.  For iOS app, it should be developed using the latest version of XCode and use the Swift5 programming language. You must not use any other languages or tools.  If you make use of any code from other sources (tutorials, help sites, etc) this must be clearly indicated in your source code. You should not share any of your assignment code with your fellow students. We use a sophisticated tool that can identify similarities between submissions. If these are detected both parties will be referred to the Academic Conduct Office accused of collusion. |
| 1. You are expected to use the [Coventry University APA](https://libguides.coventry.ac.uk/apa) style for referencing. For support and advice on this students can contact [Centre for Academic Writing (CAW)](http://www.coventry.ac.uk/study-at-coventry/student-support/academic-support/centre-for-academic-writing/?theme=main). 2. Please notify your registry course support team and module leader for disability support. 3. Any student requiring an extension or deferral should follow the university process as outlined [here](https://share.coventry.ac.uk/students/Registry/Pages/Deferrals-and-Extension.aspx). 4. The University cannot take responsibility for any coursework lost or corrupted on disks, laptops or personal computer. Students should therefore regularly back-up any work and are advised to save it on the University system. 5. If there are technical or performance issues that prevent students submitting coursework through the online coursework submission system on the day of a coursework deadline, an appropriate extension to the coursework submission deadline will be agreed. This extension will normally be 24 hours or the next working day if the deadline falls on a Friday or over the weekend period. This will be communicated via your Module Leader. 6. Assignments that are more than 10% over the word limit will result in a deduction of 10% of the mark i.e. a mark of 60% will lead to a reduction of 6% to 54%. The word limit includes quotations, but excludes the bibliography, reference list and tables. 7. You are encouraged to check the originality of your work by using the draft Turnitin links on Aula. 8. Collusion between students (where sections of your work are similar to the work submitted by other students in this or previous module cohorts) is taken extremely seriously and will be reported to the academic conduct panel. This applies to both courseworks and exam answers. 9. A marked difference between your writing style, knowledge and skill level demonstrated in class discussion, any test conditions and that demonstrated in a coursework assignment may result in you having to undertake a Viva Voce in order to prove the coursework assignment is entirely your own work. 10. If you make use of the services of a proof reader in your work you must keep your original version and make it available as a demonstration of your written efforts. 11. You must not submit work for assessment that you have already submitted (partially or in full), either for your current course or for another qualification of this university, with the exception of resits, where for the coursework, you maybe asked to rework and improve a previous attempt. This requirement will be specifically detailed in your assignment brief or specific course or module information. Where earlier work by you is citable, i.e. it has already been published/submitted, you must reference it clearly.  Identical pieces of work submitted concurrently may also be considered to be self-plagiarism. |

**Mark allocation guidelines to students (to be edited by staff per assessment)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0-39 | 40-49 | 50-59 | 60-69 | 70+ | 80+ |
| Work mainly incomplete and /or weaknesses in most areas | Most elements completed; weaknesses outweigh strengths | Most elements are strong, minor weaknesses | Strengths in all elements | Most work exceeds the standard expected | All work substantially exceeds the standard expected |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Grading Rubric** | 0 | 4 | 8 | 12 | 16 | 20 |
| **Version Control and Unit Testing**  20% | No access given to the remote repository on GitHub. No Unit tests have been written or run. No commits. | Some code access provided. Limited attempt at flawed Unit tests and marginal commits performed. | Access provided to GitHub. An unsuccessful attempt has been made to write simple Unit tests. More than 20 commits performed. | Access has been given to the code on GitHub with multiple commits. Evidence of a limited number of Unit tests written and run. More than 30 commits performed. | Evidence of regular commits over an extended period of time (more than 50) together with a range of Unit tests showing how they contribute to code quality. | Commits over an extended period of time (more than 60), demonstrating the use of branching and merging. A full suite of automated Unit tests ensuring full code coverage. |
| **Layout and Design**  20% | No evidence of the use of standard views. | The application makes use of some basic views with a ﬁxed screen size/orientation. | The app makes use of a limited range of view but only works in a single screen size/ orientation. | The app makes use of a wide range of appropriate views and layouts. It works eﬀectively in diﬀerent screen size/ orientation. | The app adapts itself to work effectively on both phones and tablets or any screen sizes and orientation. | The app makes use of custom views and animations to produce a professional standard app that works on any device. |
| **Data Persistence**  20% | The app does not work. | Data is displayed with errors. | The correct data is displayed in the app and data is saved in a file or shared preference (no database has been used) | Data is persisted locally when the app is closed fully (Data is persisted in Local data base) | Data includes complex relationships (save image to database) and is persisted locally. | Data is persisted in both local and Web based APIs databases with a complex relationship (Save image data) |
| **Programme Language**  20% | No code has been made available through online repository. | The app demonstrates only limited understanding of the programming language. | The application is fully functional and the explanation of the code is clear. | Application uses complex class relations, code is full documented | The application makes full use of the range of structures and APIs (used 2 Web APIs) covered in the module. | The app makes appropriate use of APIS (2 Web APIs) and advanced, cutting- edge APIs (Machine learning API such as TensorFlow Lite) and techniques not mentioned during the course. |
| **Hardware and Sensors**  20% | The app makes no use of any phone sensors or wearables. | A (failed) attempt has been made to incorporate a phone sensor or wearable into the app | One sensor or wearable has been incorporated into the app. | The app makes appropriate use of two phone sensors (at least one is Location sensor or Camera) | The app makes appropriate use of multiple sensors (at least 3 sensors) and must include Location and Camera sensors) | The app integrates a wide range of wearables and makes appropriate use of multiple phone sensors. |