**BTEC Assignment Brief**

|  |  |
| --- | --- |
| **Qualification** | Pearson BTEC Higher Nationals in Digital Technologies |
| **Unit number and title** | Unit 4: Programming |
| **Learning aim(s)** | **LO1** Define basic algorithms to carry out an operation and outline the process of programming an application  **LO2** Explain the characteristics of procedural, object-orientated and event-driven programming  **LO3** Implement basic algorithms in code using an IDE  **LO4** Determine the debugging process and explain the importance of a coding  standard. |
| **Assignment title** | Designing and Developing a Mobile Application with Flutter |
| **Assessor** | Muhammad Tog’oyev |
| **Issue date** | March 14, 2025 |
| **Hand in deadline** |  |

|  |  |
| --- | --- |
| Scenario or Context | **Assignment Title: Designing and Developing a Mobile Application with Flutter**  **Introduction and requirements:**  **Purpose of this Assignment**  **This assignment aims to develop your programming skills by applying algorithmic thinking and implementing real-world solutions using Flutter. You will design and develop a unique mobile application that demonstrates clean architecture, proper use of state management (BLoC or Riverpod), integration with a backend (REST API or Firebase), and testing (unit, widget, and integration). Your work must also reflect your understanding of programming paradigms, debugging practices, and coding standards.** **Scenario** **You have recently joined a mobile app development firm as a junior Flutter developer. Your manager assigns you to develop a standalone mobile application. However, each team member must independently design and implement a different type of application (e.g., to-do app, weather app, e-book reader, finance tracker, etc.).**  **You must demonstrate strong software engineering principles including:**   * **Algorithmic problem-solving** * **Clean and scalable architecture** * **Testing strategy (unit, widget, integration)** * **Application of state management (BLoC or Riverpod)** * **Backend integration (REST API or Firebase)**  **Tasks** **Task 1: Planning and Design**   * **Define at least one non-trivial algorithm used in your application (e.g., sorting/filtering logic, recommendation engine).** * **Describe how the algorithm is translated into code.** * **Outline the full development lifecycle from algorithm design to code execution.** * **Provide architectural design (clean architecture) with layered diagrams and file structures.**   **Covers P1, P2, M1, D1Task 2: Programming Paradigms**   * **Discuss the characteristics of procedural, object-oriented, and event-driven programming.** * **Compare their usage in your Flutter app and analyze the code where applicable.** * **Critically evaluate your source code in terms of structural decisions and programming paradigms.**   **Covers P3, M2, D2Task 3: Implementation**   * **Build your app using Flutter in a suitable IDE (e.g., Android Studio, VS Code).** * **Apply clean architecture principles and preferred state management (Riverpod/BLoC).** * **Connect to a backend service (Firebase or REST API).** * **Implement basic and advanced features to showcase algorithmic thinking.**   **Covers P4, M3, D3Task 4: Testing and Debugging**   * **Use testing tools to implement unit tests, widget tests, and integration tests.** * **Document the debugging process using IDE tools (breakpoints, watch expressions, logs).** * **Explain your chosen coding standard and how it ensures readability and maintainability.** * **Reflect on how debugging improved app security and reliability.**   **Covers P5, P6, M4, D4** **Submission Format**  1. **Flutter project source code (in a zipped folder or GitHub repo)** 2. **Design document (PDF or Word)**    * **Algorithm description**    * **Architecture diagram**    * **Programming paradigm analysis**    * **Explanation of coding standards used** |
|  |  |

Learning Outcomes and Assessment Criteria

|  |  |  |
| --- | --- | --- |
| **Pass** | **Merit** | **Distinction** |
| **LO1** Define basic algorithms to carry out an operation  and outline the process of programming an application | | **D1** Evaluate the implementation of an algorithm in a suitable language and the relationship between the written algorithm and the code variant. |
| **P1** Define an algorithm  and outline the process  in building an  application.  **P2** Determine the steps  taken from writing code  to execution. | **M1** Analyse the process of  writing code, including the  potential challenges faced. |
| **LO2** Explain the characteristics of procedural, object orientated and event-driven programming | | **D2** Critically evaluate the  source code of an application which implements the procedural, object-orientated and event driven paradigms, in terms of the code structure and characteristics |
| **P3** Discuss what  procedural, object-  orientated and event-  driven paradigms are;  their characteristics and  the relationship between them. | **M2** Compare the procedural,  object-orientated and event-  driven paradigms used in  given source code of an  application. |
| **LO3** Implement basic algorithms in code using an IDE | | **D3** Evaluate the use of an IDE for development of applications contrasted with not using an IDE. |
| **P4** Write a program that  implements an algorithm  using an IDE. | **M3** Enhance the algorithm  written, using the features of  the IDE to manage the  development process. |
| **LO4** Determine the debugging process and explain the importance of a coding standard | | **D4** Critically evaluate why a coding standard is necessary in a team as well as for the individual. |
| **P5** Explain the debugging  process and the  debugging facilities  available in the IDE.  **P6** Explain the coding  standard you have used  in your code. | **M4** Examine how the  debugging process can be  used to help develop more secure, robust applications. |

|  |  |
| --- | --- |
| **Sources of information to support you with this Assignment** | **Submission Requirements:**  Each student has to submit their assignment as guided in the assignment brief. The students are guided what sort of information is to produce to meet the criteria targeted. You are required to make use of headings, paragraphs and subsections as appropriate, and all work must be supported with research and referenced using the Harvard referencing system.  **Important:**   * Accepted Sources: Research Papers (Journal Articles, Conference Proceedings, Thesis), Text Books, Governmental Data, Websites (only a registered organization, an educational institution, government agency) * Information taken from unreliable sources will not be accepted * Must follow Harvard Reference Style.   **Books:**  AHO, A. V. et al. (1987) Data Structures and Algorithms. 1st Ed. Addison-Wesley.  HUNT, A. et al. (2000) The Pragmatic Programmer: From Journeyman to Master.  1st Ed. Addison-Wesley.  MCCONNELL, S. (2004) Code Complete: A Practical Handbook of Software Construction.  2nd Ed. Microsoft Press. |