**DL131/DL143 – Year 3 – Project**

**Feasibility Study and Requirements Document Guidelines**

The Feasibility Study and Requirements Document should have the following structure:

|  |  |
| --- | --- |
| 1. | Introduction  The introduction should describe the background to the project and the rationale for carrying out the project. The background of the project should include a description of context in which the application will be used. This may include a description of the business of the client (if there is one) and a description of any existing applications (or non-computerised systems) that the proposed application will replace, integrate or compete with). The rationale for the project should outline the benefits of implementing the proposed application with respect to this background. |
| 2. | Requirements Analysis  This section should define the requirements of the software application you are proposing to develop for your third year project. These requirements need to be described at a number of different levels of abstraction. Firstly, the user requirements should be described. This should be done for each potential user of the application, and should be illustrated using appropriate UML diagrams such as Use Case diagrams. According to Sommerville (2007, p. 118):  User requirements are statements, in a natural language plus diagrams, of what services the system is expected to provide and the constraints under which it must operate.  Secondly, the functional requirements should be described. These requirements should be of sufficient detail to represent a contract between a client and a software developer. Sommervile (2007, p. 118) describes the system requirements as follows:  System requirements set out the system's functions, services and operational constraints in detail. The system requirements document (sometimes called a functional specification) should be precise. It should define exactly what is to be implemented. It may be part of the contract between the system buyer and the software developers.  The functional requirements should describe, in outline, the principle elements of the user interface, and the functions to be performed in response to user interaction with the principal user interface components. This description should consist of outline wireframe diagrams of the user interface and an accompanying explanation of those diagrams and the functionality provided. The relationship between these diagrams should be described in a storyboard. In addition, the functional requirements should outline the major entities that will comprise the system data model.  Thirdly, any non-functional requirements should be described. Non-functional requirements include usability requirements, security and privacy requirements, and performance requirements. Given the scope of the third year project, non-functional requirements are often of lesser importance. However, this is not always the case.  In general, requirements analysis and specification should start by interviewing the client(s) (if there are any) to determine their requirements for the application. A survey of similar applications should also be carried out to determine the functionality provided by those applications, and possible features that may be included in the user requirements. Given the scope of the third year project, it may not be possible to implement all of the (possible) user requirements. In this case, a subset of those requirements should be identified, and a further analysis of those requirements should be carried out. This should result in a detailed description of the associated functional requirements. |
| 3. | System Model  A system model should be developed which identifies the principle system components for the application, and the interaction between these components should be described. It the application is being implemented as a networked application, the role of the different programs making up this application should be described, along with an overview of the communication requirements between those programs. The possible implementation platforms of the application should be identified and the advantages and disadvantages of these platforms should be discussed. One of these platforms should be selected for the project along with the rationale for that choice. |
| 4. | Feasibility Study  A feasibility study should be carried out to ascertain the possible risks associated with implementing the application. This should include a description of the possible technical and project management problems and risks. It should also include a description of how the risks will be managed and how the problems will be overcome. |
| 5. | Test Plan  A description of how the application will be tested should be included. This test plan should outline how the validity and quality of the application will be verified. The thought process involved in outlining a test plan will lead to a better understanding of application requirements. |
| 6. | Project Management  Finally, a project plan should be developed which outlines the principle objectives for the following stages of the development lifecycle:   * Research and analysis: background research in the problem domain and technical implementation platform; detailed analysis of functional requirements. * Outline design: including outline database, user interface and application logic design. * Detailed design: detailed specification of database, user interface and application logic. * Implementation: implementation of system components, integration of system components. * Testing: unit testing, integration testing, system testing, and test plan.   The project plan should also indicate how the different objectives of the project plan will be allocated to the different members of your project team. |

The following assessment rubric will be used to assess and grade your document:

|  |  |  |  |
| --- | --- | --- | --- |
|  | Poor | Fair | Good |
| User requirements | Poor research of requirements of client and/or functionality provided by existing applications and systems. | Reasonable description of requirements of client and/or functionality of existing applications and systems, including use of Use Case diagrams for principle user groups | Good description of requirements of client and/or functionality of existing applications and system, including use of Use Case diagrams for all user groups, and description of the benefits of implementing proposed application |
| Functional requirements | Inconsistent and incoherent description of the principle functionality of the proposed application with inadequate use of appropriate diagrams to illustrate the proposed functionality | Mostly consistent and coherent description of the principle functionality of the proposed application with some use of appropriate diagrams to illustrate the proposed functionality | Consistent and coherent description of all of the functionality of the proposed application and good use of appropriate diagrams to illustrate the proposed functionality |
| System Model | Inadequate and poor description of the system model, with some misunderstanding of how those requirements can be implemented; poor understanding of potential implementation platforms and poor selection of implementation platform | Reasonable description of the system model, with fair understanding of how those requirements can be implemented; adequate understanding of potential implementation platforms and good selection of implementation platform | Good description of the system model, with detailed understanding of how those requirements can be implemented; good understanding of potential implementation platforms and good explanation of rationale for choice of implementation platform |

|  |  |  |  |
| --- | --- | --- | --- |
|  | Poor | Fair | Good |
| Feasibility Study | Little or no understanding of the risks and problems involved in implementing the proposed application. | Reasonable understanding of the risks and problems involved in implementing the proposed application, and some proposals for handling risks and overcoming problems | Good understanding of the risks and problems involved in implementing the proposed application, and well thought out proposal for handling risks and overcoming problems |
| Test Plan | Little or no understanding of requirements of test plan; inadequate strategy for testing application. | Reasonable understanding of requirements of test plan; adequate strategy for testing application. | Good understanding of requirements of test plan; well thought out strategy for testing application. |
| Project Management | Little understanding of requirements of time and resource management, poor project plan, poor proposals for managing teamwork | Adequate understanding of requirements of time and resource management, reasonable project plan and proposals for managing teamwork | Good understanding of requirements of time and resource management, good project plan with contingency planning, good proposals for managing teamwork |

The quality of your requirements document is more important than the quantity. However, your document should be approximately 2000-2500 words in length (not including table of contents, references and any appendices). All external sources must be referenced properly using the ‘Communications of the ACM’ style (see below).

*You should discuss these guidelines, and the interpretation and application of these guidelines, in the context of your project, with your project supervisor.*

**References**

Sommerville, I. (2007). *Software Engineering* (Eighth ed.). Harlow, England: Pearson Education.