**CSC3047 IT Enterprise Project**

**Final System & Report (55%)**

**Overview**

The final report complements the submitted system itself, describing the entire process of development, including project management, specification, design, implementation/testing and any resources, research or analysis included within the process. The running system itself and the underlying source code will be assessed together with the report as the main reference in assessing the process of development of the system, including justification of decisions made throughout. Therefore the report should cover all aspects of the system functionality and the development process.

The final report will be assessed on the detail, correctness and thoroughness of its content and also on how it is organized and presented. The actual audience of the report will be the module lecturer, who will be assessing the report and system based on the criteria presented. However you should imagine that a further intended audience will be a team of developers who would be extending your system in a future “phase 2” of development. This means that there should be no ambiguities within the design and implementation of the system which would require further clarification. All sections of the system should be clearly documented and all decisions and assumptions should be fully justified with appropriate references if required.

**Notes**

The document should be submitted as a single (one only per group) printed hard-copy, bound, with an appropriate cover page.

As a rough guide, the main body of the final report is expected to be no less than 50 pages and no more than 200 pages. The most common range of report size is between 100 and 150 pages. There will be no penalty for going outside these ranges. However, a report with fewer pages risks being “light” on the required detail or missing important sections (as outlined in the report format). A report with an excessive amount of pages risks “bloat”, with possible unnecessary or repetitive content, affecting the flow and readability of the report.

The page length guide does not include additional Appendices. Certain areas of content are expected to be included as Appendices, but you may also add additional material as you see fit, as long as this is relevant and adds to the overall report and reference information to the reader. There is no guidance in terms of page length for the Appendices, with the only proviso being that if this starts to reach 30 pages or more, the team may wish to provide access in soft format or on an alternative medium (such as disk, pen drive or shared folder space).

Please note – there are NO samples available to view from previous years to avoid the risk/temptation of plagiarism. This document is provided as a very clear and detailed guide to help you create and complete your report, and has been moderated by external staff and confirmed as sufficient for this purpose.

**Format of the Report**

The document should include the following sections:

1. Introduction

This is the opening section of the report, and is a brief introduction to the purpose of the document, the aims of the work being undertaken and a discussion of what is being presented within the remaining sections of the report. It should also include the following:

- Explanation of a Student Information System

- Main users of the system

- Problems with existing or incumbent Systems

- Benefits to client of a new approach and system (efficiencies, cost, user experience) – Some quantifiable statistics / facts (as opposed to generalities) should be included

- Brief justification of how and why the team as a whole are qualified to deliver this solution

***Guide: Approx 5 - 10 pages***

2. Project Management

This section should describe and justify how the project has been organized and managed. Any artefacts of the project management process that are produced during the project should be provided and explained (e.g. any spreadsheets, Gantt charts etc. should be provided in appendices for inspection – although small examples can be included here for quick reference). This should provide a form of audit trail showing who was active and responsible for each part of the work during each phase.

This section should cover:

- The Team and its members, with brief descriptions and what they can bring to the project

- The main user roles per member (include multiple roles per member if appropriate) and brief descriptions of each

- The general approach to team-working used

- Development and Management tools/processes used, including “Agile” based methodolgies

- Description of the team communication processes, meetings and protocols/rules

- The general approach to planning sprints and development, from requirements to development plan

- Outline the plans created and adopted during development

- Where appropriate, describe any change management processes which may have been required (adjustments based on circumstances or changes in direction)

- Summary of progress during the implementation of the plan

- Overall project management lessons learnt (both positive and negative)

This section should also include, a brief paragraph for each team member outlining:

* Their contribution(s) to the final system itself
* Their contribution(s) to researching, creating and writing this report

Any large planning documentation/diagrams can be referenced here and outlined in full within the Appendices.

***Guide: Approx 15 - 20 pages***

3. Requirements Specification and Analysis

This section will provide a detailed analysis of the requirements provided by the customer, including:

* A formal description of each element of the user requirements in an appropriate form along with any acceptance tests which will be used to verify them.
* An analysis of the requirements which will show their interdependencies, relative cost to implement etc.
* A declaration of any assumptions which are being made.

The precise format of the presentation of these details is left up to the group, but must be clear and have the relevant information provided. Only those requirements (of the entire list proposed by the client) which are to be included in the final system should be listed. Relevant non-functional requirements should also be included.

Where the length of this section may become excessively long due to many requirements, the authors can select the most important of them and place the remaining specifications within the Appendix. Repetitious requirements can be outlined fully once, then summarised and referred to for the rest.

***Guide: Approx 20 - 25 pages***

4. Design

This section should illustrate all levels of the design process, including reasoning and justification within the decision-making process. Any final designs should be clearly presented, although a description of how this end point was reached, including intermediate or subsequently changed designs, can be included. This can include mock-ups or “static” versions of User Interfaces and components used.

This section will comprise several aspects of the design process, as outlined below.

*User Interface Design:*

This should provide a detailed description and justification of the design of the User Interface in the final system. The UI should be carefully designed and modelled around the specific requirements of each type of user of the system.

This should include:

* Initial GUI designs for main screens
* Sketched / mockups of screen layouts
* Format notes for suggested “look and feel” aspects
* Suggested components for important elements (e.g. timetable)
* All relevant research undertaken into all aspects of the design including style/layout, HCI and components to be used – references must be included where appropriate

A sound justification of the choices made in all of these areas is important and should be based on good principles of HCI.

*Database Design:*

This should provide a detailed and complete description and justification of a suitable Database design. Effective use of Entity Relationship diagram(s) should be used, clearly showing relationships between data entities and use of Key or Foreign field references.

The design can be presented as separate sections if required. Full reference must be made as to the purpose and use of each table/section within the system itself. Populated tables of sample data can be presented if effective, showing how this is then represented within the User Interface.

Consideration must be given to performance and scalability, both in terms of volume of data and (future) functional extensibility. Justifications of design should also include aspects of these considerations, where appropriate.

*High-Level Design:*

This should describe the overall design and make-up of the entire system, in terms of how pages and components are linked and the workflows, use-cases and navigation of the system.

This should include:

* A site map for the suggested system outlining the connections between the pages/views of the system.
* Outline page designs/story boards for common “work-flows” and use-cases.
* Indication of commonly used or re-used sections of the site where appropriate, showing efficiency for future development and maintainance.

As with User Interface Design, a sound justification of the choices made in all of these areas is important and should be based on good principles of HCI.

***Guide: Approx 30 - 40 pages***

5. Implementation and Development

This section should describe the decisions, process and main elements of the development of the system, taking the system from design to a fully developed, working system. It should focus on two to three main areas of the system and describe the development methodologies and tools used to create a functional product. Algorithms can be presented and explained for complex tasks (e.g. timetable clash-checking). Problems and challenges can be outlined, including where code required major overhaul due to unknown issues, mistakes, re-design, performance or UI refactoring, etc.

The general development methodology can be explained here, including technical descriptions of the code-base, including separation of the “front-end” client-based code with the “back-end” server-based code. All components used and their interfaces should be described here. The group can also highlight their team-working process in this section, including the benefits and challenges of this and use of the Code repository, including versioning if appropriate.

This section should also include the following main areas:

*Unit / page Implementation*

A selection (three or four) of “units of implementation” (e.g. web page or class) should be described in detail in this section. This will explain the purpose of the unit within the web site and should describe the important details of how it is implemented. If any alternative implementations were considered then these should also be discussed. The people responsible for the development of each unit should be named in this section. If possible, try to choose based on all team members contributing.

*Quality Assurance and Testing*

This section should describe and justify the quality assurance processes employed during the development. This might include details of the extent of the unit testing carried out, acceptance testing, end user testing etc. and should explain at what stage each form of testing/evaluation was carried out.

If any code corrections or site updates were carried out as a result of the testing processes or if any development procedures were adopted or changed in order to avoid future repetitions of the same errors occurring then these should all be documented.

Details of some (a small sample) specific tests carried out can be illustrated here for example, with explanation and outcomes. All other tests should be documented in the appendices for inspection (if this runs into 10 pages or more, please include on a disk or shared web folder).

All **known** bugs/issues in the system should be identified and detailed. Known bugs are much more acceptable than **unknown** bugs which are later encountered during assessment.

*Coding Standards and Conventions*

This section should detail and justify any conventions followed by the team during the development. Coding standards should (to a reasonable degree) be based on industry-standard best practice.

*Security Implementation*

All aspects of security within the system should be outlined here, with justification and must include references to source material where applicable. Examples can be given for certain sections of the system, showing usage and effect of implemented security measures.

***Guide: Approx 25 - 35 pages***

6. Conclusions

This final section should provide a thorough evaluation of the project which has been carried out. It should highlight what the team feels have been the most and least successful aspects of the project as a whole. It should include all phases of the project since the start of the module, not just the implementation phase (both semesters). It should focus purely upon the activities of the team and the documentation/system they have produced (as opposed to being an evaluation of the module) and the team should identify any lessons learned during the project.

A good evaluation will be one which shows that there are few, if any, blind spots in the teams’ assessment of the project and has identified and explained the strongest and weakest elements of the work carried out. This should cover both the system itself and the process/team involved in its creation.

The final part of this section should highlight a number (roughly six) of new features or improvements which could be added to the system for a future release. These should have a reasonable amount of detail and explain what is proposed, as opposed to an aspiration without any method. For example, the team may want to improve system performance. In this example, the report should identify which particular area(s) of the system requires this and a reasonable attempt provided to explain what is required to achieve this.

The development diary is a journal, maintained and kept by each individual student, over both Semesters of the module. It is intended as a record of the work, ideas and input each student has had to the overall team effort, and also as a way of expressing thoughts on how they have contributed, as well as the contribution or issues encountered with other team members. It is a confidential and very individual account of the journey for each student.

***Guide: Approx 5 - 10 pages***

Appendices

The appendices should include the following material:

*Technical Guide*

* Technical Infrastructure/architecture
* Basic deployment, setup/installation instructions
* Full “Data Dictionary” with accompanying Entity-Relationship diagram

***Guide: Approx 5 - 10 pages***

*User Guide*

* General brief overview/guide of each specific page of functionality
* Use of screenshots to help illustrate
* For each role, step-through of selected functionality
* Explanations of specific functionality where appropriate

***Guide: Approx 10 - 15 pages***

*Other possible Appendix sections (if relevant)*

* User Research/questionnaires
* Reference Sources
* Cite and describe 3rd-party components or tools used

***General Style and Format***

Presentation and layout is important in this document. Consideration should be given to good use of diagrams, images, charts (where appropriate), but not at the expense of written content (don’t “bulk it out” with lots of images). They should complement and enhance the written content as opposed to replace it.

The “main body” of the report should flow well and stay relatively concise, but the use of references to further material in the Appendix can maintain this balance while providing further information.

Ensure your cover page is appropriate and table of contents is complete, accurate and clear.

**The System**

The system will be assessed along with the report for this component, using the system to verify report content and the report as a guide to step through functionality and workflows. To facilitate this, the following should be made available for assessment:

* Demo of the system hosted (QUB or externally), with URL(s) provided
* Database populated with demo data
* All user roles created, with login/authentication details provided
* Should be available and accessible from Friday 20th April, for approx. 5 weeks to allow time for assessment

All details should be emailed by one member of the group to the lecturer no later than Friday 20th April.

**Marking Scheme**

The assessment will be based on (effective) group-size and any known and approved extenuating circumstances for team members)

*Based on an overall mark of 100%:*

**Final System**

- Core Functionality:

*For each core requirement and chosen non-core, how well are these fulfilled in terms of:*

* *Satisfaction level of requirement – to what extent does the product satisfy the requirement*
* *Usability and HCI (poor, basic, good, excellent)*
* *Style, Look and Feel*
* *Consistency within main system*

*[Marks 20%]*

*- Non-Core Functionality:*

*For each chosen Non-core requirement (based against amount of and Satisfaction levels of core functionality):*

* *Have enough non-core been tackled?*
* *Satisfaction level of non-core requirements*

*[Marks 15%]*

*- Other aspects:*

* *Overall consistency, navigation and professionalism*
* *Performance*
* *Security*
* *Additional extras - Relevant and useful add-on*
* *Wider applicability (branding, configuration/settings, general market appeal)*

*[Marks 10%]*

**Final Report**

- General aspects:

*Layout and Style - Cover page, contents, general layout, effective/sufficient use and presentation of diagrams, tables and code-snippets.*

*Flow – Reads well and doesn’t feel dis-jointed, with effective balance of examples and Appendices throughout*

*Content - Sufficient amount of detail (doesn’t feel too sparse or too much unnecessary content).*

*References – All references to external sources used correctly.*

*Appendices – Effective use of Appendices for references, research sources, additional information, overflow from main body*

*Technical / User Guides – Useful and clear User Guide and sufficiently detailed and clearly laid out Technical Guide*

*[Marks 4%]*

- Introduction

*Good introduction to the purpose of the document, including aims and document structure?*

*Provides a reasonable explanation of a Student Information System, including users and known issues*

*Details out (in realistic quantifiable terms) the cost/benefit to the client. Are figures reasonably believable?*

*Does the team describe effectively how/why they are qualified to deliver this solution?*

*[Marks 4%]*

- Project Management

*Is there a clear description (with justification) of how the project has been organized and managed, and inclusion of relevant and well-presented artefacts of the project management process?*

*Overall, does the ready have a good feel for who was responsible and contributed to each part of the work during the process?*

*Good overview of the Team, its members and the main user roles per member?*

*Does the team-working approach come across, including use of tools/processes and does the reader get a good feel for general communication and team organisation?*

*Has the planning approach and the actual plans developed been well outlined? Full plan included?*

*Good summary of progress during the implementation of the plan, including any changes which may have occurred and lessons learnt (positive and negative)?*

*Inclusion of contribution of each team member for both system and report?*

*[Marks 8%]*

- Requirements Specification and Analysis

*For each user requirement, is a sufficient formal description included, along with verification acceptance tests?*

*Is there sufficient analysis of requirement interdependencies, costs, assumptions, etc?*

*Good amount of relevant non-functional requirements included?*

*[Marks 10%]*

- Design

*Good general level of justification and clear presentation of User interface designs?*

*Sufficient inclusion of GUI designs and mock-ups for layouts, etc?*

*Evidence to show thinking, research and experimentation behind layouts, styles, navigation and general look and feel?*

*Good consideration evident and presented on HCI and User Experience?*

*Good description / justification of database design, including detailed ER diagram?*

*Is it clear that consideration has been given to performance and scalability, with clear and correct reasoning behind the choices?*

*Inclusion of a clear and detailed site map, providing a good overview of the entire system interconnectivity?*

*Sufficient examples of work-flows included?*

*General efficient and clever consolidation of site pages and components?*

*[Marks 14%]*

- Implementation and Development

*Good choice of main areas of the system to highlight development methodology and effectively comes across?*

*If algorithms are included, are they well explained/presented and clearly described?*

*If problems/challenges are described, were they accompanied with a clear example of good decision making?*

*Good description of general development based on front-end and back-end code?*

*Where components are used, does the reader get an insight into how they were implemented/embedded and interact with the main code body?*

*Good overview of the team-working process?*

*Good section of units of implementation, with this purpose and implementation, including any alternative requirements and those involved in the work?*

*Full set of quality assurance processes, with all types of consideration included?*

*Were results of testing outcomes described well, including steps taken based on this?*

*Good and varied list of specific tests included, with accompanying explanation/outcomes?*

*Are all known bugs included, and do any bugs exist (found during testing the system) which have not been listed?*

*Relevant and sufficient amount of (industry-based) coding standards/conventions listed?*

*Good consideration for all aspects of security provided, with examples?*

*[Marks 12%]*

- Conclusions

Good level of reflection on the project provided, throughout the entire process and including most/least successful aspects of the project with lessons learnt.

Enough and relevant new features which could be added to a future system, with sufficient detail included?

*[Marks 3%]*