<table>
<thead>
<tr>
<th><strong>Version</strong></th>
<th>1.0</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Print Date</strong></td>
<td>dd/mm/yy hh:mm</td>
</tr>
<tr>
<td><strong>Release Date</strong></td>
<td>dd/mm/yy</td>
</tr>
<tr>
<td><strong>Release State</strong></td>
<td>Initial/Core/Final</td>
</tr>
<tr>
<td><strong>Approval State</strong></td>
<td>Draft/Approved</td>
</tr>
<tr>
<td><strong>Approved by</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Prepared by</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Reviewed by</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Confidentiality Category</strong></td>
<td>Public/Confidential</td>
</tr>
</tbody>
</table>
## Document Change Control

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Authors</th>
<th>Summary of Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Contents

1 Introduction and Purpose [10 marks]  1
   1.1 Scope ................................................................. 1

2 Architectural Design [30 marks]  1

3 Detailed Design [30 marks]  1

4 Design Decisions [10 marks]  2

5 User Interface [20 marks]  2

A Definitions, Acronyms and Abbreviations  3

B References  3

C Scenarios  3

D Task Schedule  3
List of Figures
Having analysed the system’s requirements and having quantified it in the previous deliverable, it is now necessary to develop a design of the system in readiness for implementation. The PAGE LIMIT for this document, not including the appendices, is 25 PAGES. This limit will be strictly enforced.

1 Introduction and Purpose [10 marks]

Tell the reader what is the purpose of this document. You should also identify the intended audience for this document.

1.1 Scope

Give a short overview of the target system and then define the boundaries of the prototype that will be developed. You should state what it does (and what it does not do). This will help the reader distinguish your system from others systems.

2 Architectural Design [30 marks]

This section will develop your use cases for the whole product into a Class Diagram. It should contain a diagrammatic description of the use cases from your specification document, and use them to derive classes of the target product. In particular, include the following:

1. For the scenarios of all use cases already presented in Deliverable 2, construct interaction diagrams (sequence and/or collaboration diagrams). You must do so for both normal and exception scenarios (at least one each for each use case). For cross referencing purposes, reproduce the scenarios from Deliverable 2 in an appendix.

2. Extract the classes and their attributes and represent them in a UML class diagram. You may use your ERDs from your specification document as a starting point for identifying these entity classes. You are expected, in this part, to use your knowledge of object-oriented analysis and design in constructing classes that are modular and encapsulated, making use of inheritance, association and other OO principles. Make sure that the extracted entity classes are traceable to the use cases. CRC cards may be used to extract entity classes, and in that case, you must describe them.

3. Design the product in terms of classes and associations. Use the interaction diagrams to draw a class diagram showing the associations and generalisations between objects. This completes the overall architectural design.

Note that the above steps must be carried out iteratively and incrementally, with a review of the results of the steps at frequent intervals. Any time you make a change in one model, make sure that you revise the other models.

While you will present only the final versions of the models, do include a short discussion of the revisions made and explain your decisions on why you made changes.

3 Detailed Design [30 marks]

In this section, you will present the detailed design of just your PROTOTYPE, and not the entire target system. The following must be included:

1. The complete class diagram for the prototype, by adding the operations (methods) and attributes to portions of the class diagram of the previous section that pertain to the prototype. Include a discussion of which responsibilities are assigned to which class and why. You may need to refer to your interaction diagrams in the previous section for this. For each class,
make sure that the following are covered: general list of responsibilities, alternative list of possible responsibilities assigned to other classes and reasons for these design decision.

2. Perform the detailed design for all classes in the prototype. Present the detailed design of important classes in tabular form, as discussed in lectures.

4 Design Decisions [10 marks]

In addition to object-oriented design, many design decisions must be made on selection of programming language, reuse of existing software products, allocation of software components to hardware components etc. In addition, partitioning of the software product input subsystems also occurs as tradeoffs made in terms of usability, reliability, robustness, maintainability, security etc must be recorded. Budget and time constraints must be respected when making these decisions. Discuss these issues for design of the target system AND the prototype.

5 User Interface [20 marks]

Identify the Screen Design Standards you will be using for your system. Things to consider include colour schemes, screen layout, fonts, general navigation information, menu formats, use of icons, input and output methods, and user feedback. You may use any drawing tool for the diagrams in this section.

For each User Interface class, present a basic design of a sample screen or two. Some of the screen design issues you may wish to address in your design include colours, individual screen layout, how much information to include on each screen, navigation information, icons, use of menus and windows, type of screen based controls (e.g., buttons, text entry, drop-down menus, function keys), user feedback and error handling. Provide a written description of any critical screen design issues. Discuss the types of issues that need to be addressed to make the interfaces more usable.

In addition, include a sample layout for critical reports generated by the system. Some of the things to consider include text alignment, font style and size, use of tables, categories and groupings, as well as how much information to include on each page.

Also, if appropriate, include a sample design of any other user interfaces, including those that are not just screen based, such as a bar code scanner. Discuss how your User Interface classes relate to the user interface issues already discussed in your requirements document.
A Definitions, Acronyms and Abbreviations

This subsection contains definitions of all the terms, acronyms and abbreviations used in the document. Special attention should be paid to the clarification of terms and concepts from the domain of application.

B References

References to all documents that are referred to in the remainder of the requirements specification.

C Scenarios

Record of scenarios from Deliverable 2, for purposes of cross-checking.

D Task Schedule

An updated task schedule must be included here.