Requirements Specification
for
<Project Name>

March 2010
Version 0.1

Prepared By
J2Lite Software

Change History

Date	Version	Description	Updated By
24 Mar 2010	0.1	Initial Draft	J2Lite Software

Document Approvals

Name	Role	Signature

Table of Contents

1 Introduction

1.1 Purpose

<Specify the purpose of the document including its target audience.>

1.2 Document Conventions

<Indicate any special document conventions used including notation or approaches used for requirements capture. For example, you can specify that Crow's notation is used for ER diagrams.>

1.3 Project Scope

<Specify the exact project scope indicating project boundaries. This can also include the purpose of the software project, its benefits and overall goals. In the case of a software product, this should contain product vision and should indicate the exact user base for the product. If you are aware of features that should go into a future version, list them here or add a new section on "Features for Future Releases".>

1.4 References

<This should contain links to all external documents and web resources referred in the requirements. Sometimes prototypes or product vision are specified in a separate document. Link them here. Also ensure that you link to the correct version.>

2 System Description

<This section can contain a detailed overview of the software system being built. It can contain a subsection detailing the existing system followed by the proposed system details. If relevent, this section can also contain business details such as organizational hierarchy. Basically provide everything that is necessary to understand the Fuctional Requirements section which follows this.>

3 Functional Requirements

<This section contains system requirements followed by various requirement models which can be used for detailed design. In addition to the following sections, you can also add process flow diagrams, data flow diagrams, flowcharts and decision tables if required.>

3.1 System Features

<This section specifies the high level system features required in the software product. Each requirement given below has a unique tag that can be referred in the traceability matrix.>

3.1.1 System Feature 1

3.1.2 System Feature 2

3.2 Use Cases

3.2.1 Use Case Diagrams

<One or more diagrams depicting how various actors interact with the software system.>

3.2.2 Use Case 1

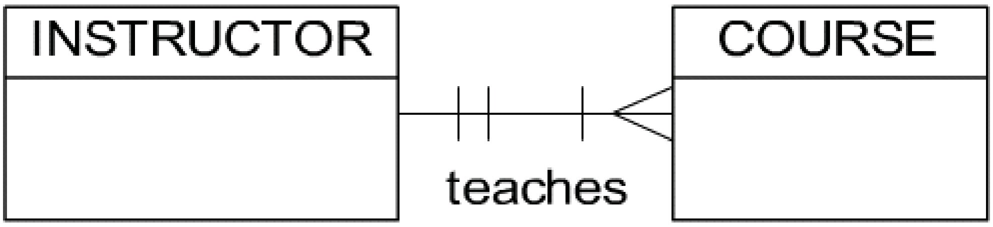
<This provides a detailed description of the use case. Usually it is captured in the following table format. Add more rows or removing rows depending on your specific requirement.>

ID	*<unique use case id>*
Description	*<Detailed description of the use case>*
Actors	*<Specify various actors who will invoke this use case>*
Preconditions	*<An example for a purchase item use case would be - user must be logged in to invoke purchase item.>*
Basic Steps	*<Usual flow of the use case indicating the various steps in it>*
Alternate Steps	
Exceptions	*<Exceptions indicate what should happen if a rare unexpected condition occurs. For example, if the use case is purchase item, what happens when the the stock runs out during purchase?>*
Business validations/Rules	*<If business rules are already specified in the system features section, provide a link. Otherwise specify the rules.>*
Postconditions	

3.2.3 Use Case 2

3.3 Entity Relationship Diagrams

<One or diagrams to depict all the entities in the system and their relationships. You can use different notations for ER diagrams. Following is a very simple ER diagram in Crow's notation.>



3.4 Data Dictionary

3.4.1 Entity 1

<A brief description of the entity followed by a table containing all its attributes as shown below. >

Attribute	**Type**	**Optional?**	**Notes**
<Attribute Name>	*<Data type of the attribute>*	*<Y or N>*	*<Explain any specific restrictions or rules applcable on this attribute>*

3.4.2 Entity 2

4 External Interface Requirements

<Provide sub sections for each external interface. Identify all the input and ouput from the external interfaces.>

5 Technical Requirements (Non functional)

<Please note that all the following subsections may not be applicable for a system. Sometimes you will have to add additional sections (for example, Legal requirements)>

5.1 Performance

<For example, What is the response time required?>

5.2 Scalability

<For example, how many users the system should support after two years of operation?>

5.3 Security

<For example, is data encryption required?>

5.4 Maintainability

5.5 Usability

5.6 Multi lingual Support

<What are the languages that software system should support?>

5.7 Auditing and Logging

5.8 Availability

<For example, Is any kind of downtime acceptable or required?>

6 Open Issues

<There could be open issues even at the end of the requirements elicitation. List of all of them here so it can be tracked and closed later. Some of these issues may later become project risks as well.>