Project Plan

Subject: Project Plan   **Date:** 09 June 2005

         Issue 1.0   **From:** Daniel Calder

                                                    Mark Marwil

1. Project Goal

The project being developed is a library tool. Using various software tools that comply with open standards will allow a library system that will illustrate the robustness of the tool's capabilities to develop software. The project will show step by step how these tools can greatly aide the developers when designing, building and testing a system. Throughout the development of the project tutorials will build on how to use the tools so that students from various schools can easily and quickly learn how to use the tools when developing a system.

2. Table of Contents

3. Life Cycle Phases

In this section the various tools used at different phases of the software life cycle will be described.
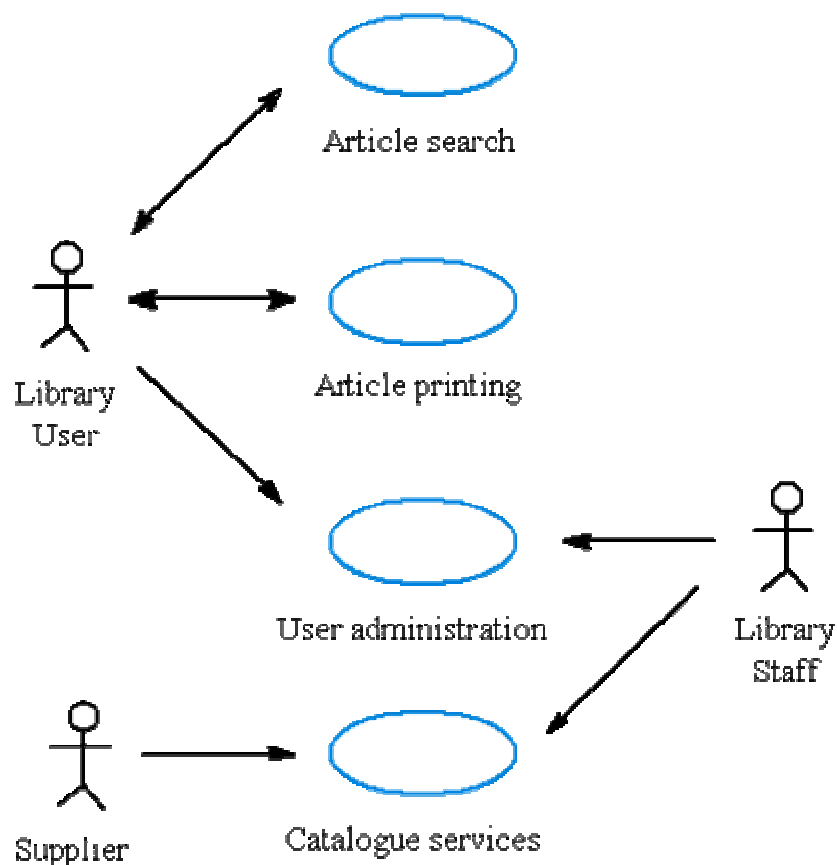
3.1 Requirements

There are several features that the customer will want the system to be able to perform. A Patron will need an intuitive Graphical User Interface (GUI)

that is easy to navigate. Each Patron will require the ability to be able to search the online library for books and see if they are available or when they will be due back. The system also needs to have the capability of checking books in/out, reserve books, create accounts and look up customer information.

To capture the requirements, use cases will be created in Requisite Pro from the Rational Software package. Next, the use cases that were created in Requisite Pro will be associated with Rational Rose which graphically illustrates the functionality of the system.

3.1.1   Use Case Diagram

After the Requirements have been completed and reviewed the Use Case Diagram will be created using Rational Rose Enterprise. The Use Case Diagram illustrates the behavior of the system. The following is an example of what the Use Case Diagram will look like when completed:

Article search

Article printing

Library
User

User administration

Library
Staff

Supplier

Catalogue services

3.1.2   Conceptual Model

After the Use Case Diagram has been completed and reviewed the Conceptual Model will be developed. This is an example of what the Conceptual Model will look like when completed:
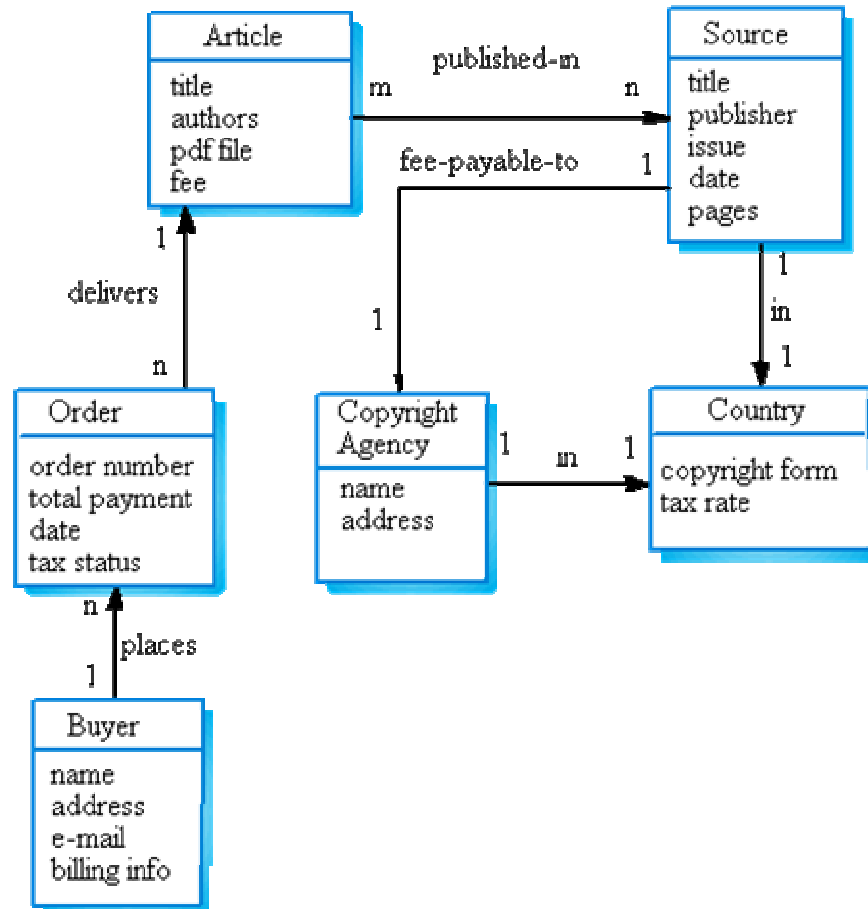

3.2   Design

In the design phase Rational Rose will be used to capture the class diagram, sequence diagram and collaboration diagram. When designing the class diagram the goal is to represent the classes using icons that describe their relationship with each other. The sequence diagram will aide the design of the library system by showing how the objects interact with each other in a time based sequence. Collaboration diagrams are used to display how the objects work together.
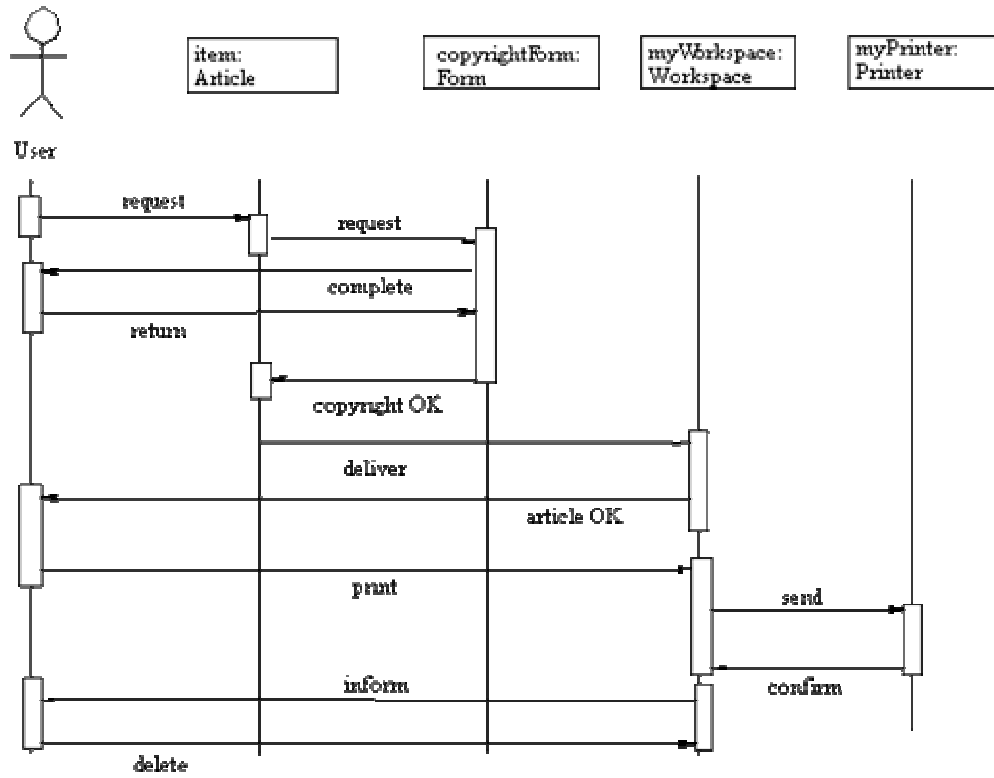
3.2.1   Class Diagram

After the Conceptual Model has been completed and reviewed the Class Diagrams will be created to show the relationships between the different

classes. An example of what the Class Diagram for the library system will look like is as follows:
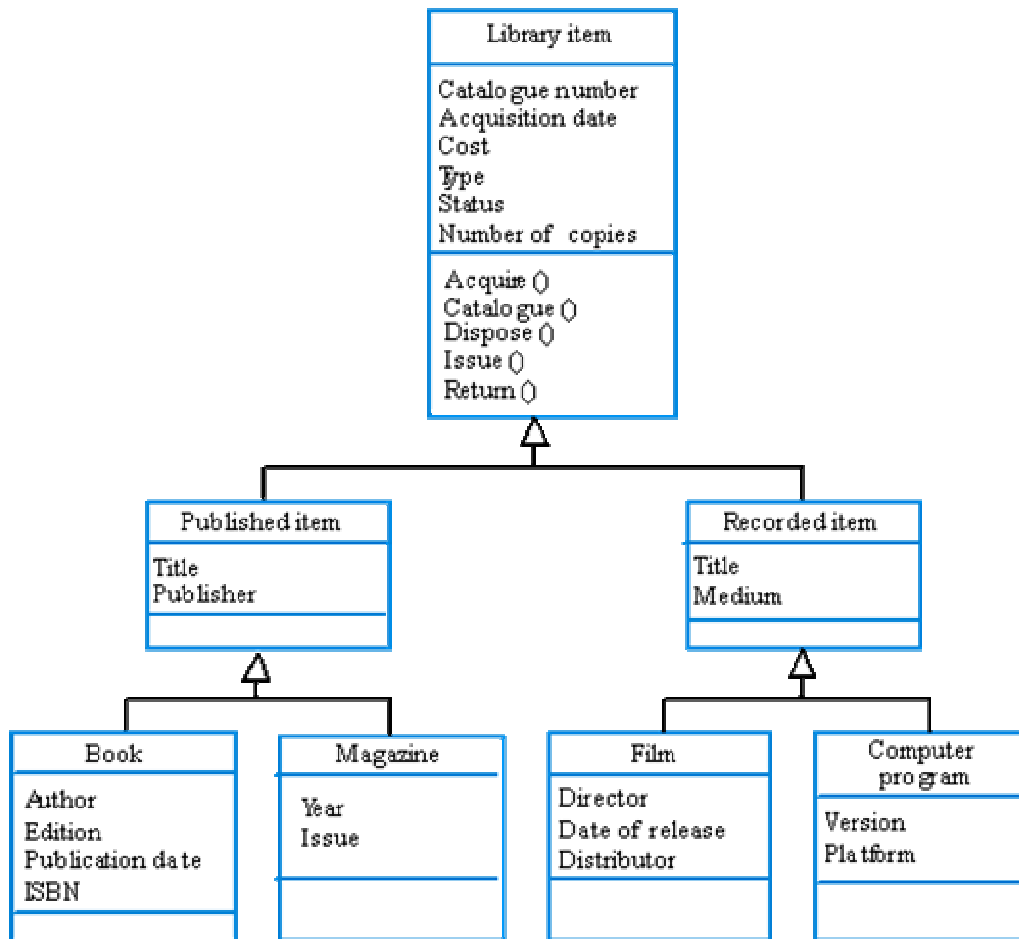

3.2.2 Sequence Diagrams

After the Class Diagrams have been completed and reviewed the Sequential Sequence Diagrams will be developed to show the interactions between objects in a time based sequence. An example of what the Sequential Sequence Diagrams for the library system will look like are as follows:

User

item:
Article

copyrightForm:
Form

myWorkspace:
Workspace

myPrinter:
Printer

request

request

complete

return

copyright OK

deliver

article OK

print

send

inform

confirm

delete

3.2.3 Collaboration Diagram

After the Sequential Sequence Diagrams have been completed and reviewed the Collaboration Diagrams will be created to illustrate how the objects interact. An example of what the Collaboration Diagram for the library system will look like is as follows:
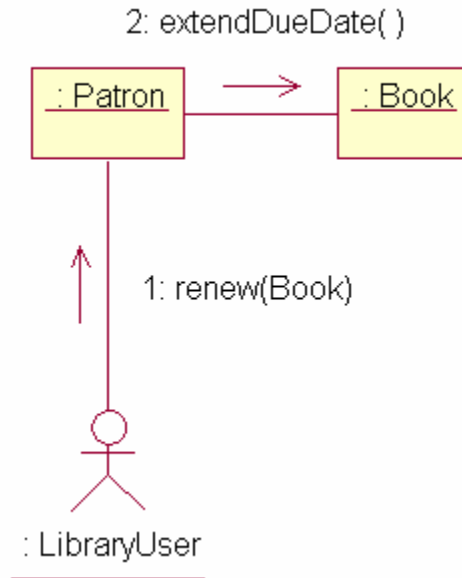
3.3 Coding

For coding the Eclipse development environment will be used. The Eclipse software is an open source development platform dedicated to enhancing productivity by creating a flexible and customizable development environment. Eclipse can be used with the Clear Case versioning control system. The client side will be coded using HTML and JavaScript.   The server side will be coded using PHP and Java. Half of the client side will be coded in PHP and the other half in Java.
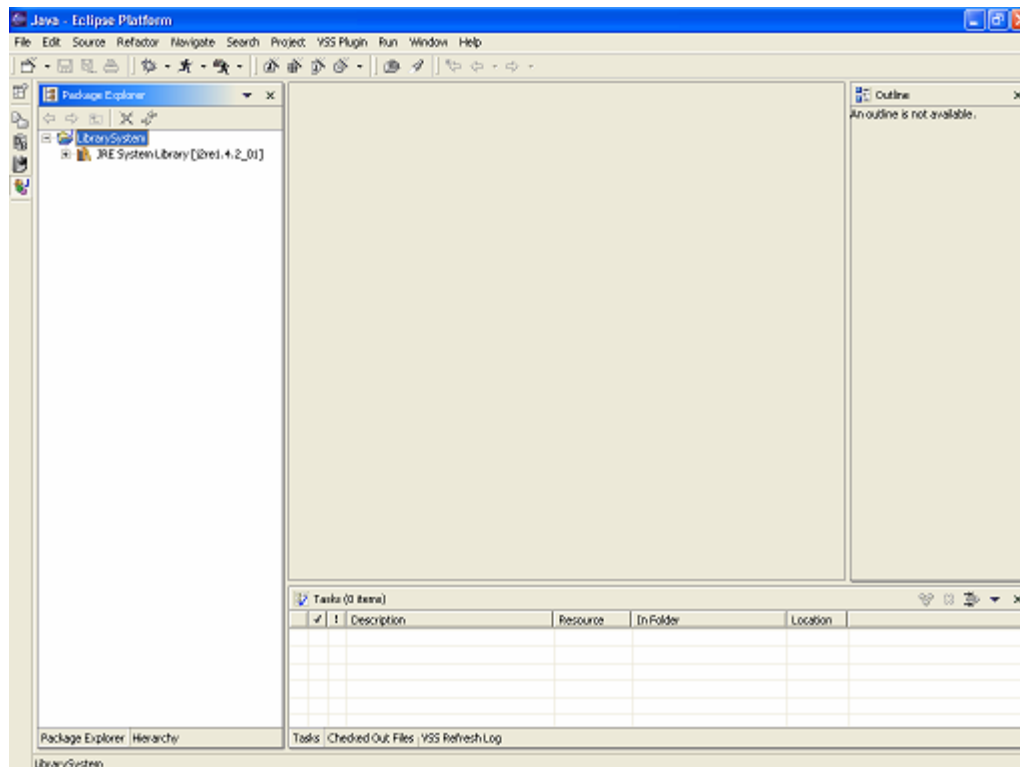
3.4 Testing

During the testing phase the Rational testing tools will be used to ensure that the library system is performing correctly. The programs that will be used are Rational Purify, Robot and Test Manager. When projects are being coded unit testing will be performed on each file.

Purify functions are an automatic error detection tool for finding runtime errors and memory leaks in every component of your program. This software is capable of pin-pointing hard-to-find bugs and memory errors that may not show up immediately but could potentially cause a program to crash someday. Rational Robot will run a complete set of automated tests on the

program to verify that the program functions properly for the selected test cases. Test Manager encompasses all aspects of test analysis from test management to execution to reporting and supports all kinds of testing from pure manual test approaches to various automated paradigms including unit testing, functional regression testing and performance testing.

3.5 Release

When the project is finished the project will be placed on the SSI Hub website. Tutorials will be created to explain step by step how the IBM Rational software was used when building the project.

4. Technical Information

The following table describes the software tools and technologies expected to be used in the project.

Tools/Techniques/Methods	Description
PHP	PHP5 will be used for the server side scripting. This version provides full object-oriented support and is easily accessible and open source. PHP is also available in multiple platforms. PHP is also easy to learn because PHP similar in some extent to high level languages such as C/C++ and large base of documentation. PHP comes with an extensive set of libraries and supports multiple types of Databases.
Apache	Apache 2 will be used for our Web Server. Apache provides an easy integration with PHP and is widely used in the internet community.
DB2	DB2 will be used for our database engine. This version is supported by PHP by default without adding additional PHP modules.
Requisite Pro	As a part of IBM's Rational Suite of software, Requisite Pro is an integrated tool for managing the requirements created during the software engineering process.
ClearCase	The target is to use ClearCase as our source control system. If the ClearCase is not available, CVS will used as our source control system.
ClearQuest	IBM's Rational ClearQuest is a customizable defect and change tracking system designed to deal with the dynamic nature of software development. The tool simplifies the process of managing every type of change activity associated with software development, including enhancement requests, defect reports and documentation modifications.
Project Console	IBM's Rational ProjectConsole is a part of the Rational Suite and is extremely useful in managing single or multiple projects by keeping track of project metrics.
Rational Rose	Rational Rose is a visual modeling software solution that can graphically depict an overview of the behavior of a system. This

	software shows objects interaction and links between them, life history of a given class, and workflow of a business process. Rose supports UML – a communication standard. Using UML, a common vocabulary is used and thus miscommunication is minimized.
Eclipse	The Eclipse Project is an open source development platform that is dedicated to enhancing productivity by creating a flexible and customizable development environment. Eclipse serves as a cross-platform, multi-language programming IDE and is specifically designed for tool building and integration.
Test Manager	Rational TestManager is a testing tool that encompasses all aspects of test analysis from test management to execution to reporting. Test Manager supports all kinds of testing from pure manual test approaches to various automated paradigms including unit testing, functional regression testing and performance testing.
Purify Plus	IBM's Rational PurifyPlus package consists of three user-friendly and powerful tools that deal with performance issues as you develop your code namely the Rational Purify, Rational PureCoverage and Rational Quantify. These tools work hand in hand to help you develop better applications more efficiently. Flexibility is also a key concept with the PurifyPlus package. For example, you can customize your preferences of what data to be collected while using Purify through the use of filters.
Robot	Rational Robot is a complete set of components for automating the testing of Microsoft Windows client/server and Internet applications running under Windows NT 4.0, Windows XP, Windows 2000, Windows 98, and Windows Me.
MS Office	MS Office will be used to create our process documentation and also for user documentation.