

“Proficio”  
Software Engineering C  
Week 12 Group Report

**Project Leader:**

Adam Bramley

**Team Members:**

Stephen Allan

Leith Bade

Peter Williamson

## Table of Contents

1. Introduction .....	4
1.1 Team Contribution .....	4
1.2 End User Interactions .....	4
1.3 Project Plan .....	4
1.3.1 Features left out from final product.....	5
2. Requirements Revision .....	5
2.1 Functional Requirements: .....	5
2.2 Non-Functional Requirements: .....	6
2.3 Summary of changes .....	6
3. System Architecture.....	7
4. Deployment Strategy .....	8
4.1 Deploying a JSP Application .....	8
4.2 The Database.....	8
5. Issue Tracking Policy .....	8
5.1 Issue Tracking Workflow .....	9
6. Testing and Metrics.....	10
6.1 Unit Test Results.....	10
Summary .....	10
Packages .....	10
6.2 SVN Metrics .....	11
6.3 Compatibility testing .....	12
7. User Manual.....	13
7.1 Basic Search .....	13
7.2 Results Returned .....	14

7.3 Advanced Search .....	15
7.3 Glossary of Terms .....	16
8. Installation and Maintenance guide .....	17
8.1 Installation .....	17
8.1.1 Build the WAR .....	17
8.1.2 Setup the server .....	17
8.1.3 Setup MySQL .....	17
8.1.4 Setup Tomcat .....	17
8.2 Maintenance .....	18
8.2.1 Updating the WAR code .....	18
8.2.2 Installing new plugins .....	18
9. Appendices .....	19
9.1 Appendix A – System Architecture .....	19
9.2 Appendix B - Database Schema .....	20
9.3 Appendix C – Use Case Diagram .....	21
9.4 Appendix D – Gantt chart .....	22
9.5 Appendix E – JDepend Report .....	23
9.6 Appendix F – Emma Report .....	27
9.7 Appendix G – PMD Report .....	28
9.8 Appendix H – Issue tracking .....	34

## 1. Introduction

The goal of this project is to provide the administration staff at Massey University with a program to assist with the submission of academic publications to the Research Administration Management System (RIMS). The program is to capture relevant information on the academic paper from a generic input (document object identifier), and give the user an automatically generated form of information to submit to RIMS.

Our project team consists of 4 members each with different skill sets and experiences to contribute to an overall solid workforce. The following paragraphs outline brief autobiographies from the team members with relevant information about their specific skills and talents.

### 1.1 Team Contribution

At the beginning of the project our team discussed the various skill sets that each person had, from this we decided on the different threads that each member would mainly work on. As the team leader, Adam is in charge of delegating work, arranging meetings with the project manager, client, and team. Along with this, Adam has been focussing for the most part on designing the interface for the system and coding the jsp files up. This included writing any Javascript and CSS as well as the obvious HTML. Peter has been in charge of the database and all things related to the database. He has designed the DB from the start including the structure and all scripts required for creating, updating, and retrieving information in the DB. Steve has been working with all the APIs for the various meta-data gathering websites (CrossRef and Scopus so far) and extracting the information out of those, along with all searching functionality of the program. Finally, Leith has been in charge of structuring the project in regards to packages, classes, builds, interfaces, and plugins.

### 1.2 End User Interactions

During weeks 8-12 met the client on a fortnightly basis. The first meeting was a rough demo of the product with Craig and Jens attending also. We stepped them through our system showing a rough demo of each module and the new advanced searching functionality. During the second meeting we allowed Karen to use our system for what it is intended to do, and she submitted an article to RIMS with the assistance of our product. In doing this we got a huge amount of constructive criticism about changes we could make to the interface to give a more streamlined operation. As our functionality had already been finalised with her previously there weren't any major changes we found from this meeting.

### 1.3 Project Plan

*Please refer to Appendix D for a snapshot of the project plan and progress.*

Overall, we have achieved what we set out to do from the beginning of the project. Along the way there have been many changes and things we have had to leave out due to time constraints.

### 1.3.1 Features left out from final product

1. At the beginning of the project we thought we would need to include an email client to email the authors for confirmation of the submission. Before we began implementing this feature we found that it was redundant as the user must manually email the authors regardless.
2. We were planning to have more data sources than just CrossRef. Unfortunately, it took us quite a while to get Scopus working properly as we not able to obtain the proper developer's key for the full API. Due to this we had to use the Javascript API with limited functionality. This resulted in it taking much longer to finish the Scopus plug-in, giving us no time to continue with any other possible plugins
3. Screen scrapping was going to be a last resort for us when our plug-ins had failed to return any data. Near the end of the project we realised that it was far too messy and unreliable to use this feature, and decided to stick only to the reliable sources of CrossRef and Scopus.
4. Since neither CrossRef nor Scopus return an abstract, we were working on screen scrapping for the abstract off of the publisher's website. We ran out of time to effectively implement this feature, without having html tags and other unwanted characters left in the description.

## 2. Requirements Revision

After speaking with Karen and Craig on the 21<sup>st</sup> July and multiple group meetings we came out with a refined list of functional and non-functional requirements. These requirements were further refined after receiving feedback from our project manager.

### 2.1 Functional Requirements:

1. The system needs to be able to take a DOI as an input, and output metadata for the article. This is to be done through a series of steps, going through the major databases including CrossRef and Scopus. The meta-data output includes: authors, article title, journal title, publication year, URL, page numbers, volume, and issue for journal articles. For conference articles we will also include conference location and start/end dates. For books we will also include the place published and editors of the book.
2. If the system cannot extract any meta-data, the raw article is to be produced.
3. The system has to display the meta-data on screen and allow the user to edit the data.
4. Access to a local database is needed. The author details and DOIs are stored in the database so author details can be automatically generated for known authors, and the system can check for duplicate DOIs.
5. The system needs to be able to print the cover sheet containing all the metadata.
6. If the user doesn't have a DOI, the system needs an advanced search function with the ability to search the local database and online sources for matches when given the author name, article title, and publication date.
7. Provide a seamless operation of copy and pasting for the user

8. Provide auto-completion on the article's publisher.

## 2.2 Non-Functional Requirements:

1. The system is to be run on windows
2. The system is to return local database results in under 3 seconds and web results in under 15 seconds per plugin.
3. The system needs to handle plug-in architecture
4. The system needs to have an automated deployment and maintenance procedure.
5. The system must have support for access to the internet and local database through a proxy.

These requirements have also been signed off by Karen on 5<sup>th</sup> August.

## 2.3 Summary of changes

These requirements have not changed much since the last report. The main changes have been that we are no longer attempting to “screen scrape” any information from publisher's websites if our plugins fail. This is because it will be very rare that both Scopus and CrossRef return no data, and it isn't very elegant relying on page layout and html tags to extract reliable information.

We have also decided not to change our plugin architecture to allow for article types that may need to be added in the future. We have decided this because Karen had told us at the very beginning of the project that the only things she deals with are Journal articles, Books, and Conferences. Since we structured our whole system around this there will need to be significant changes to switch to a plug-in type architecture for the types, and we simply don't have the time to implement this.

The final two points on the functional requirements are new. As neither of our plugins returns the publisher of the article, we asked Karen how she would like to get around this. We agreed on an auto-completion function where the most common publishers are stored in a text file, and when the user starts typing in the publisher field a table with suggestions will pop up. When the user commits an article to the database, if the publisher submitted was not in the text file, it is added to it.

Finally, we decided to implement buttons next to each of the text fields on our results page to allow for easy-to-use copy and pasting as that's a major part of the user's workload when submitting from our article to the RIMS system. However, this function will only work in the latest version of Internet Explorer, as other browsers do not allow access to the host's clipboard.

### 3. System Architecture

*Please refer to Appendix A.*

From the figure, you can see that there are several main features that encompass our system architecture. Beginning at the left, the user (or the internet they are using) is directly interacting with the various servlets. There is one servlets per Java Service Page, which is for each different page that the user sees, that acts as the interface between the user and the service. The servlets' input is http; this is the input that the system will receive from the forms that the user is filling out. The servlets' output is html to display results of the system working on the input. The JSPs format the output of the servlets into html.

The first JSP the user interacts with in the application is the index page. This is where the user enters the DOI in. When the user submits the DOI it is sent to an object called the DOIRequest (or AdvancedRequest if the user is using the advanced search). The DOIRequest class then sends the DOI string to the MetadataRetreiverFactory. The class acts as a Factory for the different plugins, these plugins are the different websites we are using to gather the meta-data. Each plugin is tested to return a publication and when one is found the factory will return it to the DOIRequest. The plugins work by using the sites' APIs and filling the Publication object with information gathered through parsing the XML from the data source.

Once the DOIRequest receives the publication object it then either outputs an error message if something has gone wrong, or there is no such article; or it forwards the information to a results page. There are 3 separate results pages linked to each of the 3 publication types (book, conference, and journal). Once the user has finished editing the author and publication details, the publication is then sent for committing to the database and, if the checkbox is ticked, a separate window is opened for printing.

The database is used to store DOIs as an extra safeguard against duplication. Along with this, author name, id, and affiliation will be stored in the database so that once the user has entered an affiliation for an author once, it will be automatically generated for them next time. There is a small possibility of the database returning the wrong affiliation, but our client has specified that as long as it is editable it will still be very helpful.

#### 3.1 Summary of changes

We have removed the Service class from the application, as it ended up being a redundant class. This allowed us to directly interact with the servlet (DOIRequest, AdvancedRequest, CommitRequest, etc). The servlets also now send the request objects to the plugins. These are used to append error messages to, so if there is an error in gathering the meta-data, the error will be displayed on the index page with a well formed message.

## 4. Deployment Strategy

The deployment strategy focuses on how to deploy a JSP application but some thought is also given to other deployment strategies should they become relevant (e.g. The same web application could possibly be built to incorporate a web service and client that connects to it which would have a different deployment set up).

### 4.1 Deploying a JSP Application

Any JSP application is going to need a web server to run on. In the Massey environment that could possibly even include a server set up on the client's individual machine, but will most likely require being set up on an existing Massey server. Internal Massey policy will most likely have some impact on the final location and the final installation procedure.

A permanent location on a Massey server allows for easier deployment for other users if the application becomes more popular. An individual machine set up would need to be replicated on each user's computer, taking considerable time and making maintenance very difficult.

Deploying the actual files and directories that make up the application would be a simple matter of deploying as a WAR file (Web Archive) to the server or copying the file set up directly via FTP or similar.

### 4.2 The Database

A locally accessed database is required for the author details and DOIs of articles that have already been entered. Storing the database on a server along with the application has several advantages; namely, that the database can be used by multiple users very easily and is easier to update, backup and maintain. A database on the user's computer lacks these advantages and also has the additional issue of replication of data if other users have their own databases set up. The local database needs to be able to avoid repetition of work which it would be unable to do if several copies of the same database existed.

Relocation of the database after installation can be managed by altering the web.xml file which contains the connection information.

## 5. Issue Tracking Policy

*Please refer to Appendix H for issue tracking snapshot.*

From weeks 8-12 our team took issue tracking much more seriously, mainly because there were potential marks involved in doing so. To our surprise, using them actually helped our project quite considerably, resulting in more bugs being fixed at a faster rate than before. We are now posting issues wherever we can't find a solution immediately, or when the issue is with another member's code.



## 5.1 Issue Tracking Workflow

- Open a new issue through the Google code project page
- Fill out the summary and description of the issue.
- Cc to all team members.
- Make sure the issue type and priority are set accordingly.
- Submit the issue.

*After the issue has been submitted:*

- Adam will assign the issue to the team member that was responsible for creating the issue. For example, since Peter is the main contributor to designing the database, any issues relating to the database will belong to him. This assures that issues will be fixed as fast as possible.

*After the issue has been resolved:*

- Adam will check the issue is solved, and if it is, set the issue to “Done” on the project page.

If an issue is open for more than 7 days, the team will have an extra meeting discussing the issue until it is resolved. In most cases, we will be able to resolve the issue with all team members discussing it. If the issue is still unable to be solved then further investigation will take place by a specific team member. If the particular issue is due to an extra feature, then that feature may be removed.

## 6. Testing and Metrics

Please refer to Appendices E, F, and G for the reports.

For testing our system and generating metrics we will utilise as many automated systems as possible. For testing we have decided to use Junit 3. This allows us to frequently run tests on our code to test for any new bugs that may have been introduced from freshly written code or refactoring. As part of our testing policy, each time a new issue is submitted a new test case will be added to account for the newly introduced bug. This will ensure the system is safe from the bug and allow us to change our code to suit it.

To automate metrics, we will be using EMMA, PMD, and JDepend to gather information on test coverage, code complexity, and dependencies. This will give us a good range of metrics to ensure our code is up to standard. To generate the reports we are using an ANT build script. This puts all the results in html files inside the resulting build file.

**Please note, we have excluded the database and servlet packages from the EMMA build as EMMA does not pick up coverage reports on test cases run on Tomcat. For running the tests through Massey, the proxy needs to be added to the VM arguments of each test case.**

### 6.1 Unit Test Results

The following are the summaries for the Junit tests on our packages.

#### Summary

Tests	Failures	Errors	Success rate	Time
19	0	0	100%	11.874

Note: *failures* are anticipated and checked for with assertions while *errors* are unanticipated.

#### Packages

Note: package statistics are not computed recursively, they only sum up all of its testsuites numbers.

Name	Tests	Errors	Failures	Time(s)	Time Stamp	Host
<a href="#">nz.ac.massey.rimgroup3.database.test</a>	5	0	0	6.797	2010-10-13T22:39:50	IT042193
<a href="#">nz.ac.massey.rimgroup3.metadata.bean.test</a>	7	0	0	1.953	2010-10-13T22:40:00	IT042193
<a href="#">nz.ac.massey.rimgroup3.metadata.plugin.test</a>	2	0	0	2.421	2010-10-13T22:40:14	IT042193
<a href="#">nz.ac.massey.rimgroup3.metadata.test</a>	1	0	0	0.219	2010-10-13T22:40:20	IT042193
<a href="#">nz.ac.massey.rimgroup3.servlet.test</a>	4	0	0	0.484	2010-10-13T22:40:22	IT042193

## 6.2 SVN Metrics

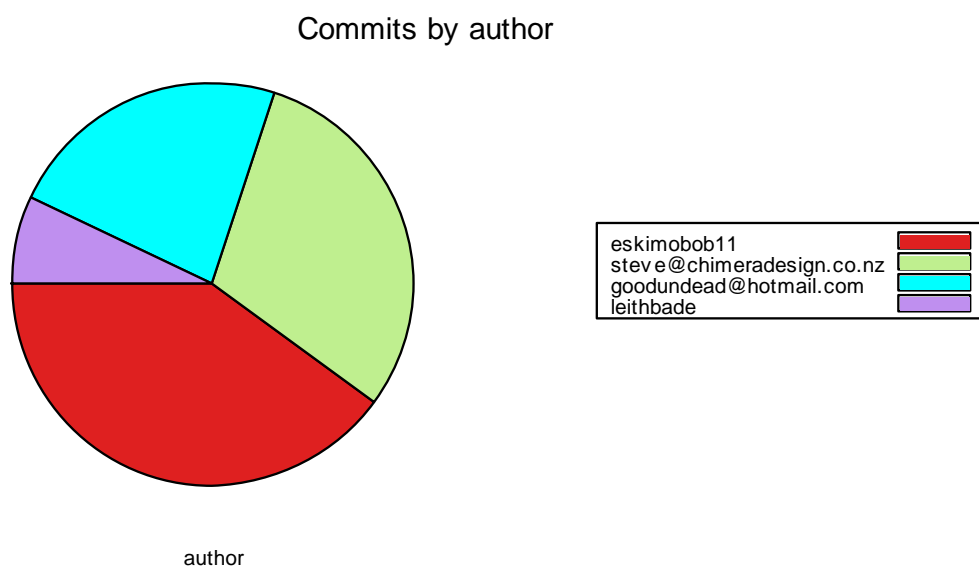
For the last 5 weeks (7/09/10 to 12/10/10) the following SVN statistics have been produced using TortoiseSVN's analysis:

Total Commits analysed: 100

Total file changes: 361

Commit Details Per Week			
	Average	Min	Max
Commits per week	20	8	26
Most active author (Adam Bramley)	8	2	10
Least Active Author (Leith Bade)	1	0	4
File Changes per week	72	22	93

The following graph shows the commits per author in a pie chart:



Eskimobob11: Adam Bramley,

steve@chimeradesign.co.nz: Steve Allan,

goodundead@hotmail.com: Peter Williamson,

leithbade: Leith Bade

### 6.3 Compatibility testing

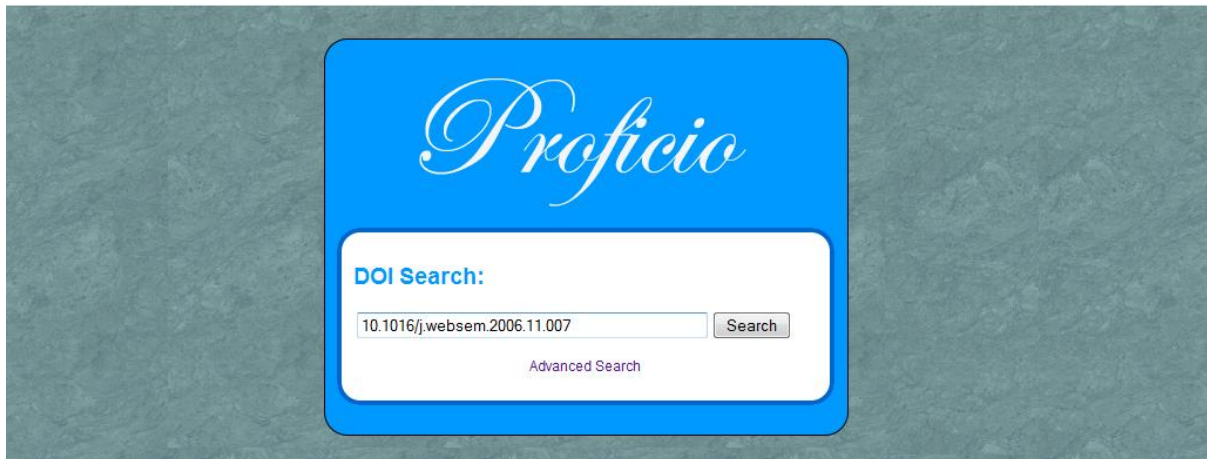
All testing was done using Windows XP, this is because the only platform we expect the admin at Massey to be using is Windows and since it's a web application the compatibility relies more on the browser.

Browser	Comments
Mozilla 3.6.8	Completely functional, all styles are showing along with AJAX animation. Meta-data gathering working as expected.
Internet Explorer 6	Something is very wrong here. Even the button to send the DOI away isn't showing. Curved borders also aren't showing but that was expected. We are not worried about this as the users are very unlikely to be using IE6.
Google Chrome 6.0.4	Completely functional, all styles are showing along with AJAX animation. Meta-data gathering working as expected.
Opera 10.62	Completely functional, all styles are showing along with AJAX animation. Meta-data gathering working as expected.

## 7. User Manual

This user manual is available via the application on the index page through the help icon. Each section explains the different pages and how to use them. There is also a section at the end explaining the basic terms on the results page.

### 7.1 Basic Search



- Enter the DOI into the text box and click the Search button to search for the information regarding the DOI.
- Please wait while Proficio finds the article.

## 7.2 Results Returned

The screenshot shows the Proficio web application interface. The header features the 'Proficio' logo in a cursive font. Below the header, there are several sections for data entry:


- Journal Contribution:** A section with radio buttons for selecting the type of contribution: Review Article, Full Article in Personal Publication, Journal Editor, Full Article in Journal, Editorial, Brief Communication, Letter or Note, and Other.
- Research or Professional Community:** Radio buttons for Research (PBRF) and Professional Community.
- Quality Assured?:** Radio buttons for Yes and No.
- Confidential?:** Radio buttons for Yes and No.
- Authors:** A section with two rows of input fields for First Name, M. Name, Last Name, Affiliation, ID Number, and IU Number. A 'Submitter' label is present on the right. 'And' and 'Delete' buttons are at the bottom.
- Publication Details:** A section with input fields for Publication ID, Article/Output Title, Journal/Publication Title, Publication Year, Publisher, ISSN, Volume, Number, Start Page, End Page, UHL Address, DOI, and Keywords. A 'Follow Link' button is next to the UHL Address field. An 'Abstract' label is on the right side of this section.

At the bottom of the form, there are 'Back', 'Confirm', and 'Print' buttons.

- Use the information gathered by the Proficio to fill in the details in the RIMS system. This can be done by highlighting the relevant field's text, then copying (Ctrl+C) and pasting (Ctrl+V) the text into corresponding text field in the RIMS system.
- Any information not retrieved from the Proficio will have to be filled out manually. Click the new tab that has appeared to access the remaining information. Alternatively click the 'Follow Link' link
- If you wish to print set the check box to print.  
Fill out any blank text fields and click the appropriate radio buttons in Proficio, that are relevant to information that is required for printing (any information left blank will not be printed).
- Publication ID is the RIMS system ID number after committing the article to the RIMS system.  
This information can only be gathered from the RIMS system.
- Click the Confirm button.

- Now you are done! Any information in the DOI text box and the Authors text boxes has been saved to the database, The DOI has been saved to prevent anyone from duplicating the work, and if the same author has any more publications all the details will be returned next time.

### 7.3 Advanced Search



The screenshot shows the Proficio web application interface. At the top, the word "Proficio" is written in a large, white, cursive font on a blue background. Below this, there are two main search sections. The first section, titled "DOI Search:", contains a text input field with the value "10.1016/j.websem.2006.11.007" and a "Search" button. Below the input field is a link labeled "Advanced Search". The second section, titled "Advanced Search:", contains a note "All fields are required." followed by four text input fields labeled "Last Name:", "Journal/Publication Title:", "Article Title:", and "Publication Year:". Below these fields is a "Search" button. At the bottom of this section, there is a text input field with the value "http://dx.doi.org/" and a "Locate Article" button. Above this field is the text "Or use the DOI DX tool to go to the article:".

- If for whatever reason you are not supplied with a DOI, click the Advanced Search link
- Fill out the all the appropriate fields, click the bottom Search button
- If the application somehow fails please use the DX DOI tool to locate the article. Enter in the DOI and click the Locate Article button. We apologise for the inconvenience that was caused.

## 7.3 Glossary of Terms

The screenshot shows a web browser window displaying the Proficio application. The interface includes a header with the Proficio logo, a navigation menu, and a main content area with several sections. The following table lists the numbered annotations and their corresponding UI elements:

Annotation Number	UI Element
1	Browser (Address Bar)
2	Tabs
3	Application - Proficio (Logo)
4	Radio Button (Journal Contribution)
5	Text Box (Authors)
6	Button (Add)
7	Link (Follow Link)
8	Check Box (Publication Details)

- |                           |                 |
|---------------------------|-----------------|
| 1. Browser                | 2. Tabs         |
| 3. Application - Proficio | 4. Radio Button |
| 5. Text Box               | 6. Button       |
| 7. Link                   | 8. Check Box    |



## 8. Installation and Maintenance guide

This is a guide to show how to install and maintain our system.

### 8.1 Installation

The following sections explain how to install the various components required to run our system.

#### 8.1.1 Build the WAR

1. Check out the latest code from SVN.
2. Run the Ant build script “build.xml”
3. Locate the WAR package at “lib/rmsgroup3-1.0.war”

#### 8.1.2 Setup the server

1. Install an OS such as Ubuntu
2. Install Apache Tomcat 6.0
3. Install MySQL 5.1
  - Refer to your OS’s documentation for installation steps

#### 8.1.3 Setup MySQL

1. Start MySQL
2. Create a blank MySQL database
3. Run the “create-tables.sql” SQL script from the SVN folder “scripts” to create the required database tables

#### 8.1.4 Setup Tomcat

1. Copy the WAR file to Tomcat’s “webapps” folder
2. Copy all the files from the “config” file in the SVN to Tomcat’s “config” folder
3. Edit the “Connector” element in “server.xml” to use port 80
4. Edit the “Resource” element in “context.xml”
  - a. Edit the “username” property to match the MySQL database’s user name
  - b. Edit the “password” property to match the MySQL database’s password
  - c. Edit the “url” property to match the MySQL server’s name, port, and the database name
    - The format is “jdbc:mysql://server\_name:port/database\_name?...”
5. Edit the “webapps/rmsgroup3/WEB-INF/classes/configuration.properties file”
  - a. Edit “crossrefEmail” to your Crossref API developer key
  - b. Edit “scopusDevId” to your Scopus API developer key
6. Start Tomcat
7. Open a browser to verify the server is running

## 8.2 Maintenance

The following sections explain how to update code in our system, and how to add new plugins to it.

### 8.2.1 Updating the WAR code

1. To install bug fixes or code changes, build a new WAR
2. Overwrite the WAR file in Tomcat's "webapps" folder with the new one
3. Restart Tomcat

### 8.2.2 Installing new plugins

1. Generate a JAR file containing the MetadataRetriever implemented class
2. Add a "META-INF\services\nz.ac.massey.rimgroup3.metadata.MetadataRetriever" file to the JAR that contains the name of the implementing class
3. Place the JAR file in Tomcat's "webapps\rimgroup3\WEB-INF\lib" folder
4. Restart Tomcat

## 9. Appendices

### 9.1 Appendix A – System Architecture

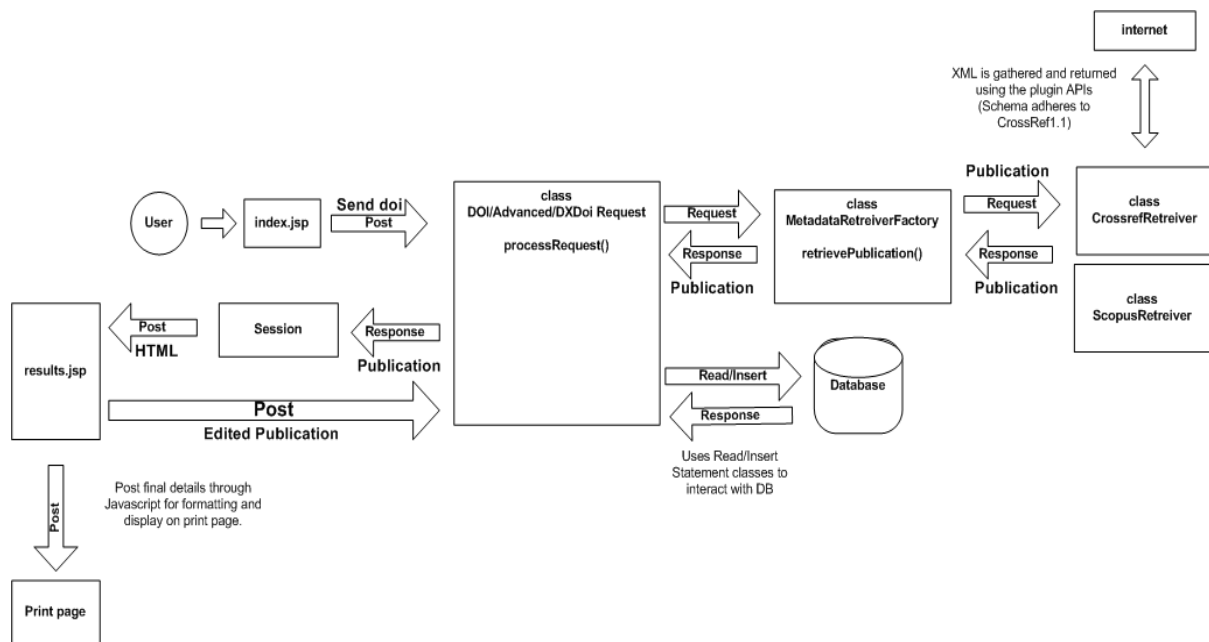
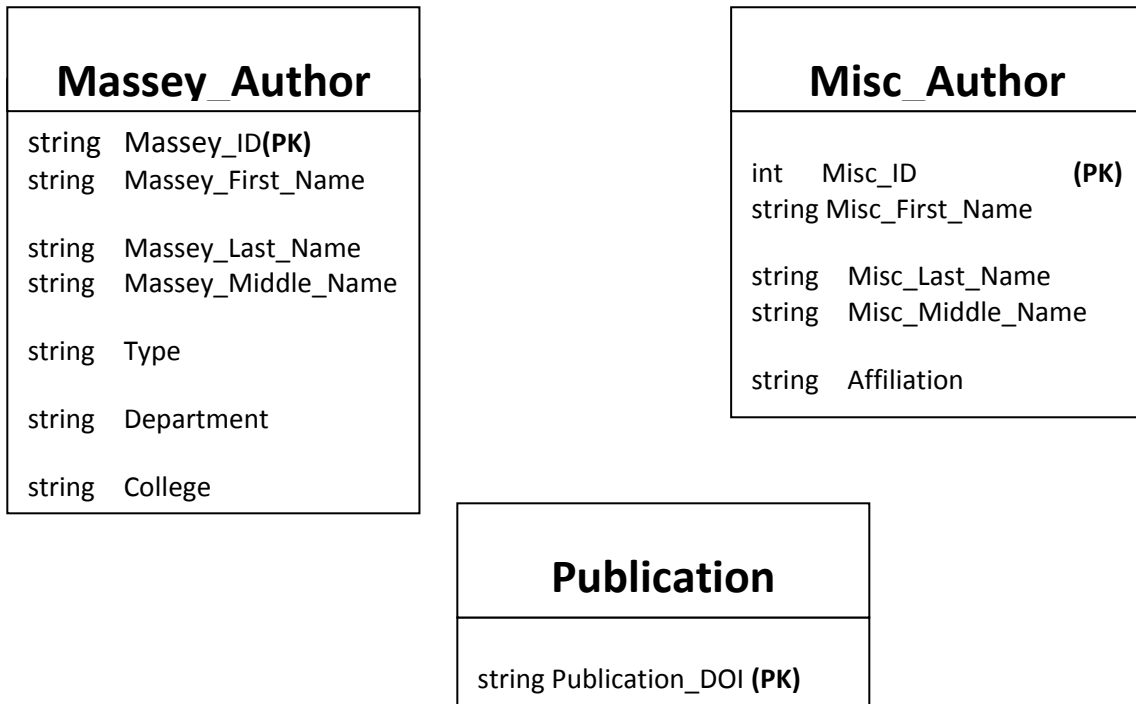


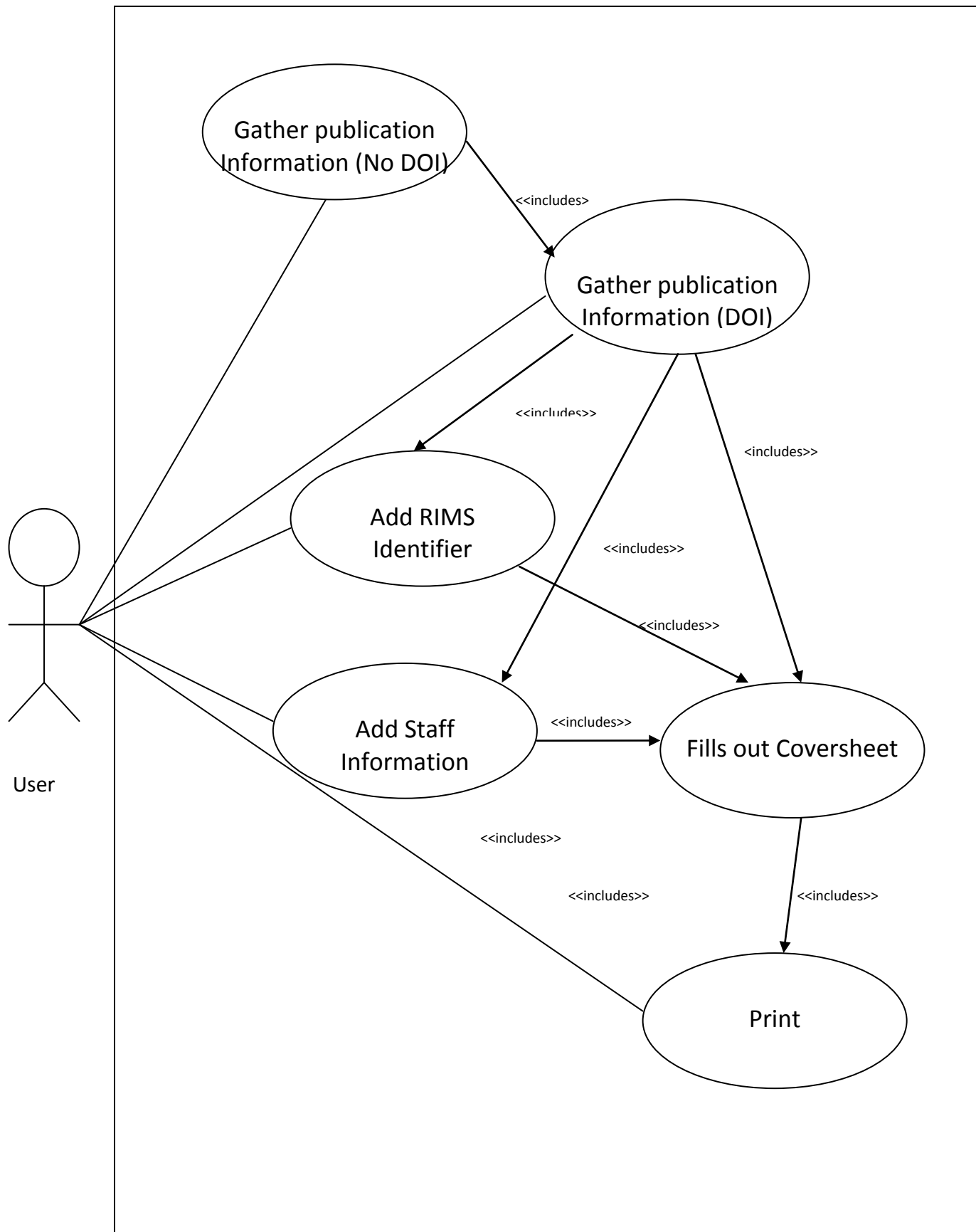
Figure 1. Improved System Architecture

## 9.2 Appendix B - Database Schema

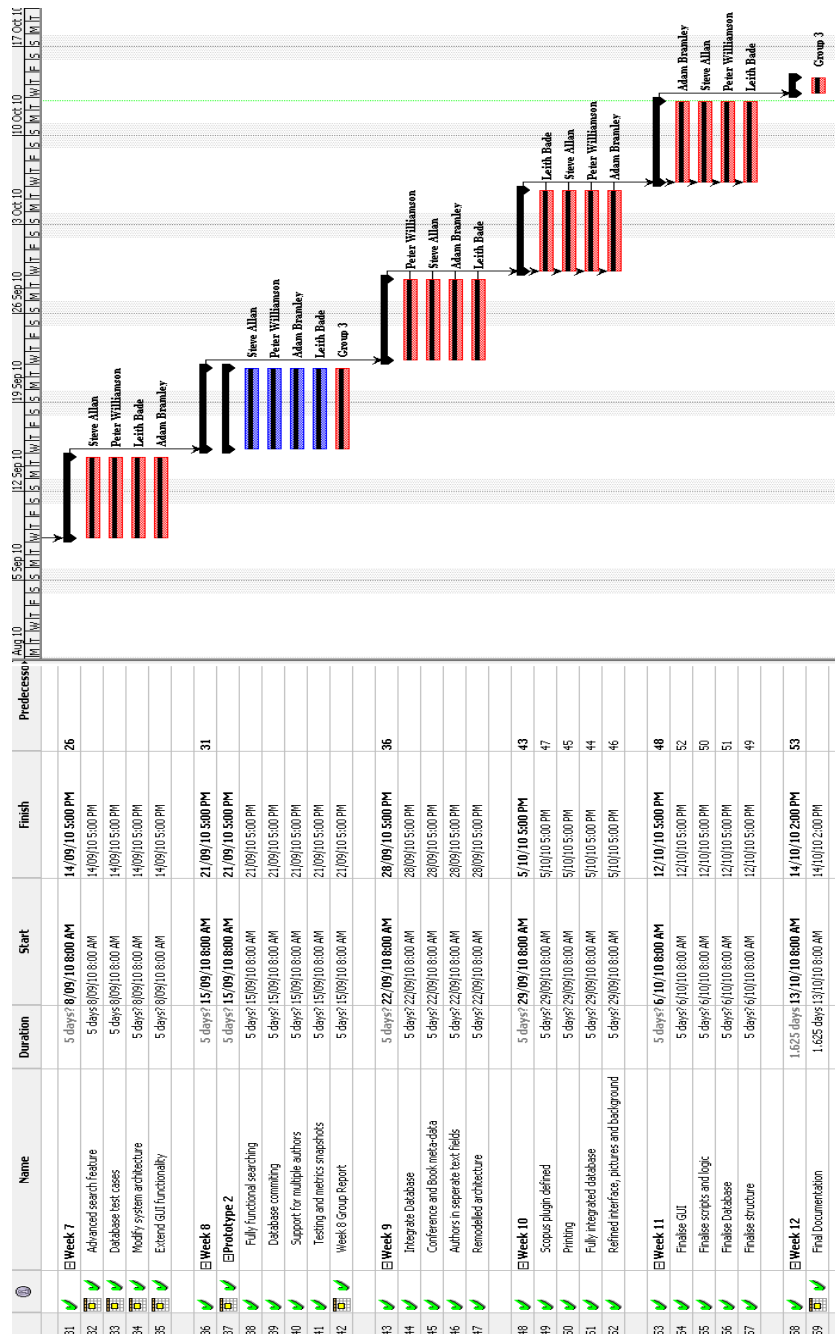


Please note this is not incomplete. Our database only contains 3 tables of which none are related to each other.

### 9.3 Appendix C – Use Case Diagram



## 9.4 Appendix D – Gantt chart



## 9.5 Appendix E – JDepend Report

Report generated by JDepend.

### Summary

Package	Total Classes	<a href="#">Abstract Classes</a>	<a href="#">Concrete Classes</a>	<a href="#">Afferent Couplings</a>	<a href="#">Efferent Couplings</a>	<a href="#">Abstractness</a>	<a href="#">Instability</a>	<a href="#">Distance</a>
<a href="#">nz.ac.massey.rimgroup3.database</a>	6	1	5	2	1	0.17	0.33	0.5
<a href="#">nz.ac.massey.rimgroup3.database.test</a>	2	0	2	0	3	0	1	0
<a href="#">nz.ac.massey.rimgroup3.metadata</a>	2	1	1	3	1	0.5	0.25	0.25
<a href="#">nz.ac.massey.rimgroup3.metadata.bean</a>	7	0	7	8	0	0	0	1
<a href="#">nz.ac.massey.rimgroup3.metadata.bean.test</a>	7	0	7	0	1	0	1	0
<a href="#">nz.ac.massey.rimgroup3.metadata.plugin</a>	5	0	5	1	3	0	0.75	0.25
<a href="#">nz.ac.massey.rimgroup3.metadata.plugin.test</a>	2	0	2	0	2	0	1	0
<a href="#">nz.ac.massey.rimgroup3.metadata.test</a>	1	0	1	0	2	0	1	0
<a href="#">nz.ac.massey.rimgroup3.runner</a>	1	0	1	1	0	0	0	1
<a href="#">nz.ac.massey.rimgroup3.servlet</a>	5	0	5	0	3	0	1	0
<a href="#">nz.ac.massey.rimgroup3.servlet.test</a>	1	0	1	0	0	0	0	1

### Packages

#### [nz.ac.massey.rimgroup3.database](#)

[Afferent Couplings](#): 2    [Efferent Couplings](#): 1    [Abstractness](#): 0.17    [Instability](#): 0.33    [Distance](#): 0.5

Abstract Classes	Concrete Classes	Used by Packages	Uses Packages
nz.ac.massey.rimgroup3.database	nz.ac.massey.rimgroup3.database.DatabaseConnection	<a href="#">nz.ac.massey.rimgroup3.database.test</a>	<a href="#">nz.ac.massey.rimgroup3.metadata.bean</a>
.DatabaseConnectI	nz.ac.massey.rimgroup3.database.InsertDetails	<a href="#">nz.ac.massey.rimgroup3.servlet</a>	
	nz.ac.massey.rimgroup3.database.InsertStatements		
	nz.ac.massey.rimgroup3.database.ReadStatements		
	nz.ac.massey.rimgroup3.database.SearchAuthors		

#### [nz.ac.massey.rimgroup3.database.test](#)

[Afferent Couplings](#): 0    [Efferent Couplings](#): 3    [Abstractness](#): 0    [Instability](#): 1    [Distance](#): 0

Abstract Classes	Concrete Classes	Used by Packages	Uses Packages
None	nz.ac.massey.rimgroup3.database.test.BlankTest	None	<a href="#">nz.ac.massey.rimgroup3.database</a>
	nz.ac.massey.rimgroup3.database.test.FilledTest		<a href="#">nz.ac.massey.rimgroup3.metadata.bean</a>
			<a href="#">nz.ac.massey.rimgroup3.runner</a>

**[nz.ac.massey.rimgroup3.metadata](#)**[Afferent Couplings](#): 3[Efferent Couplings](#): 1[Abstractness](#): 0.5[Instability](#): 0.25[Distance](#): 0.25

Abstract Classes	Concrete Classes	Used by Packages	Uses Packages
nz.ac.massey.rimgroup3.metadata.MetadataRetriever	nz.ac.massey.rimgroup3.metadata.MetadataRetrieverFactory	<a href="#">nz.ac.massey.rimgroup3.metadata.plugin</a> <a href="#">nz.ac.massey.rimgroup3.metadata.test</a> <a href="#">nz.ac.massey.rimgroup3.servlet</a>	<a href="#">nz.ac.massey.rimgroup3.metadata.bean</a>

**[nz.ac.massey.rimgroup3.metadata.bean](#)**[Afferent Couplings](#): 8[Efferent Couplings](#): 0[Abstractness](#): 0[Instability](#): 0[Distance](#): 1

Abstract Classes	Concrete Classes	Used by Packages	Uses Packages
None	nz.ac.massey.rimgroup3.metadata.bean.Author nz.ac.massey.rimgroup3.metadata.bean.Book nz.ac.massey.rimgroup3.metadata.bean.Conference nz.ac.massey.rimgroup3.metadata.bean.Editor nz.ac.massey.rimgroup3.metadata.bean.Journal nz.ac.massey.rimgroup3.metadata.bean.Person nz.ac.massey.rimgroup3.metadata.bean.Publication	<a href="#">nz.ac.massey.rimgroup3.database</a> <a href="#">nz.ac.massey.rimgroup3.database.test</a> <a href="#">nz.ac.massey.rimgroup3.metadata</a> <a href="#">nz.ac.massey.rimgroup3.metadata.bean.test</a> <a href="#">nz.ac.massey.rimgroup3.metadata.plugin</a> <a href="#">nz.ac.massey.rimgroup3.metadata.plugin.test</a> <a href="#">nz.ac.massey.rimgroup3.metadata.test</a> <a href="#">nz.ac.massey.rimgroup3.servlet</a>	None

**[nz.ac.massey.rimgroup3.metadata.bean.test](#)**[Afferent Couplings](#): 0[Efferent Couplings](#): 1[Abstractness](#): 0[Instability](#): 1[Distance](#): 0

Abstract Classes	Concrete Classes	Used by Packages	Uses Packages
None	nz.ac.massey.rimgroup3.metadata.bean.test.AuthorTest nz.ac.massey.rimgroup3.metadata.bean.test.BookTest nz.ac.massey.rimgroup3.metadata.bean.test.ConferenceTest nz.ac.massey.rimgroup3.metadata.bean.test.EditorTest nz.ac.massey.rimgroup3.metadata.bean.test.JournalTest nz.ac.massey.rimgroup3.metadata.bean.test.PersonTest nz.ac.massey.rimgroup3.metadata.bean.test.PublicationTest	None	<a href="#">nz.ac.massey.rimgroup3.metadata.bean</a>



**[nz.ac.massey.rimgroup3.metadata.plugin](#)**[Afferent Couplings](#): 1[Efferent Couplings](#): 3[Abstractness](#): 0[Instability](#): 0.75[Distance](#): 0.25

Abstract Classes	Concrete Classes	Used by Packages	Uses Packages
None	nz.ac.massey.rimgroup3.metadata.plugin .CrossrefRetriever nz.ac.massey.rimgroup3.metadata.plugin .ScopusRetriever nz.ac.massey.rimgroup3.metadata.plugin .ScopusRetriever\$JsonObject nz.ac.massey.rimgroup3.metadata.plugin .ScopusRetriever\$JsonPartOK nz.ac.massey.rimgroup3.metadata.plugin .ScopusRetriever\$JsonResult	<a href="#">nz.ac.massey.rimgroup3.metadata.plugin.test</a>	<a href="#">com.google.gson</a> <a href="#">nz.ac.massey.rimgroup3.metadata</a> <a href="#">nz.ac.massey.rimgroup3.metadata.bean</a>

**[nz.ac.massey.rimgroup3.metadata.plugin.test](#)**[Afferent Couplings](#): 0[Efferent Couplings](#): 2[Abstractness](#): 0[Instability](#): 1[Distance](#): 0

Abstract Classes	Concrete Classes	Used by Packages	Uses Packages
None	nz.ac.massey.rimgroup3.metadata.plugin.test.CrossrefRetrieverTest nz.ac.massey.rimgroup3.metadata.plugin.test.ScopusRetrieverTest	None	<a href="#">nz.ac.massey.rimgroup3.metadata.bean</a> <a href="#">nz.ac.massey.rimgroup3.metadata.plugin</a>

**[nz.ac.massey.rimgroup3.metadata.test](#)**[Afferent Couplings](#): 0[Efferent Couplings](#): 2[Abstractness](#): 0[Instability](#): 1[Distance](#): 0

Abstract Classes	Concrete Classes	Used by Packages	Uses Packages
None	nz.ac.massey.rimgroup3.metadata.test.MetadataRetrieverFactoryTest	None	<a href="#">nz.ac.massey.rimgroup3.metadata</a> <a href="#">nz.ac.massey.rimgroup3.metadata.bean</a>

**[nz.ac.massey.rimgroup3.runner](#)**[Afferent Couplings](#): 1[Efferent Couplings](#): 0[Abstractness](#): 0[Instability](#): 0[Distance](#): 1

Abstract Classes	Concrete Classes	Used by Packages	Uses Packages
None	nz.ac.massey.rimgroup3.runner.ScriptRunner	<a href="#">nz.ac.massey.rimgroup3.database.test</a>	None

**[nz.ac.massey.rimgroup3.servlet](#)**[Afferent Couplings](#): 0[Efferent Couplings](#): 3[Abstractness](#): 0[Instability](#): 1[Distance](#): 0

Abstract Classes	Concrete Classes	Used by Packages	Uses Packages
None	nz.ac.massey.rimgroup3.servlet.AdvancedRequest nz.ac.massey.rimgroup3.servlet.AutoComplete nz.ac.massey.rimgroup3.servlet.CommitRequest nz.ac.massey.rimgroup3.servlet.DXDoiRequest nz.ac.massey.rimgroup3.servlet.DoiRequest	None	<a href="#">nz.ac.massey.rimgroup3.database</a> <a href="#">nz.ac.massey.rimgroup3.metadata</a> <a href="#">nz.ac.massey.rimgroup3.metadata.bean</a>

**[nz.ac.massey.rimgroup3.servlet.test](#)**[Afferent Couplings](#): 0[Efferent Couplings](#): 0[Abstractness](#): 0[Instability](#): 0[Distance](#): 1

Abstract Classes	Concrete Classes	Used by Packages	Uses Packages
None	nz.ac.massey.rimgroup3.servlet.test.AutoCompleteTest	None	None

### Cycles

There are no cyclic dependancies.

## 9.6 Appendix F – Emma Report

### OVERALL COVERAGE SUMMARY

name	class, %	method, %	block, %	line, %
all classes	100% (13/13)	91% (101/111)	57% (1149/2028)	56% (328.8/592)

### OVERALL STATS SUMMARY

total packages: 3  
total executable files: 10  
total classes: 13  
total methods: 111  
total executable lines: 592

### COVERAGE BREAKDOWN BY PACKAGE

name	class, %	method, %	block, %	line, %
<a href="#">nz.ac.massey.rimgroup3.metadata.pl ugin</a>	100% (5/5)	73% (16/22)	56% (589/1061)	53% (134.4/254)
<a href="#">nz.ac.massey.rimgroup3.metadata.be an</a>	100% (7/7)	95% (82/86)	58% (536/930)	57% (187/326)
<a href="#">nz.ac.massey.rimgroup3.metadata</a>	100% (1/1)	100% (3/3)	65% (24/37)	62% (7.4/12)

## 9.7 Appendix G – PMD Report

#	File	Line	Problem
1	nz\ac\massey\rimsgroup3\database\DatabaseConnection.java	9	<u>Found non-transient, non-static member. Please mark as transient or provide accessors.</u>
2	nz\ac\massey\rimsgroup3\database\DatabaseConnection.java	11	<u>Document empty constructor</u>
3	nz\ac\massey\rimsgroup3\database\DatabaseConnection.java	23	<u>Position literals first in String comparisons</u>
4	nz\ac\massey\rimsgroup3\database\InsertDetails.java	10	<u>All methods are static. Consider using Singleton instead. Alternatively, you could add a private constructor or make the class abstract to silence this warning.</u>
5	nz\ac\massey\rimsgroup3\database\InsertDetails.java	21	<u>Ensure that resources like this Connection object are closed after use</u>
6	nz\ac\massey\rimsgroup3\database\InsertStatements.java	13	<u>All methods are static. Consider using Singleton instead. Alternatively, you could add a private constructor or make the class abstract to silence this warning.</u>
7	nz\ac\massey\rimsgroup3\database\ReadStatements.java	11	<u>All methods are static. Consider using Singleton instead. Alternatively, you could add a private constructor or make the class abstract to silence this warning.</u>
8	nz\ac\massey\rimsgroup3\database\ReadStatements.java	11	<u>The class 'ReadStatements' has a Cyclomatic Complexity of 12 (Highest = 16).</u>
9	nz\ac\massey\rimsgroup3\database\ReadStatements.java	20	<u>The method 'miscReadStatement' has a Cyclomatic Complexity of 16.</u>
10	nz\ac\massey\rimsgroup3\database\ReadStatements.java	20	<u>The method miscReadStatement() has an NPath complexity of 321</u>
11	nz\ac\massey\rimsgroup3\database\ReadStatements.java	66	<u>Avoid if (x != y) ..; else ..;</u>
12	nz\ac\massey\rimsgroup3\database\ReadStatements.java	74	<u>Ensure that resources like this ResultSet object are closed after use</u>
13	nz\ac\massey\rimsgroup3\database\ReadStatements.java	106	<u>The method 'masseyReadStatement' has a Cyclomatic Complexity of 16.</u>
14	nz\ac\massey\rimsgroup3\database\ReadStatements.java	106	<u>The method masseyReadStatement() has an</u>

<u>NPath complexity of 321</u>		
15	nz\ac\massey\rimsgroup3\database\ReadStatements.java	152 <u>Avoid if (x != y) ..; else ..;</u>
16	nz\ac\massey\rimsgroup3\database\ReadStatements.java	160 <u>Ensure that resources like this ResultSet object are closed after use</u>
17	nz\ac\massey\rimsgroup3\database\ReadStatements.java	199 <u>Ensure that resources like this ResultSet object are closed after use</u>
18	nz\ac\massey\rimsgroup3\database\SearchAuthors.java	9 <u>All methods are static. Consider using Singleton instead. Alternatively, you could add a private constructor or make the class abstract to silence this warning.</u>
19	nz\ac\massey\rimsgroup3\database\SearchAuthors.java	32 <u>Avoid unnecessary comparisons in boolean expressions</u>
20	nz\ac\massey\rimsgroup3\metadata\MetadataRetrieverFactory.java	17 <u>A class which only has private constructors should be final</u>
21	nz\ac\massey\rimsgroup3\metadata\MetadataRetrieverFactory.java	20 <u>Found non-transient, non-static member. Please mark as transient or provide accessors.</u>
22	nz\ac\massey\rimsgroup3\metadata\MetadataRetrieverFactory.java	20 <u>Private field 'metadataRetrievers' could be made final; it is only initialized in the declaration or constructor.</u>
23	nz\ac\massey\rimsgroup3\metadata\MetadataRetrieverFactory.java	29 <u>Use block level rather than method level synchronization</u>
24	nz\ac\massey\rimsgroup3\metadata\MetadataRetrieverFactory.java	36 <u>Use block level rather than method level synchronization</u>
25	nz\ac\massey\rimsgroup3\metadata\bean\Author.java	10 <u>The class 'Author' has a Cyclomatic Complexity of 3 (Highest = 17).</u>
26	nz\ac\massey\rimsgroup3\metadata\bean\Author.java	91 <u>Ensure you override both equals() and hashCode()</u>
27	nz\ac\massey\rimsgroup3\metadata\bean\Author.java	91 <u>The method 'equals' has a Cyclomatic Complexity of 17.</u>
28	nz\ac\massey\rimsgroup3\metadata\bean\Author.java	91 <u>The method equals() has an NPath complexity of 4096</u>
29	nz\ac\massey\rimsgroup3\metadata\bean\Book.java	13 <u>The class 'Book' has a Cyclomatic Complexity of 2 (Highest = 16).</u>
30	nz\ac\massey\rimsgroup3\metadata\bean\Book.java	18 <u>Found non-transient, non-static member. Please mark as transient or provide accessors.</u>
31	nz\ac\massey\rimsgroup3\metadata\bean\Book.java	71 <u>Avoid using implementation types like 'ArrayList'; use the interface instead</u>
32	nz\ac\massey\rimsgroup3\metadata\bean\Book.java	79 <u>Ensure you override both equals()</u>

			<u>and hashCode()</u>
33	nz\ac\massey\rimsgroup3\metadata\bean\Book.java	79	<u>The method 'equals' has a Cyclomatic Complexity of 16.</u>
34	nz\ac\massey\rimsgroup3\metadata\bean\Book.java	79	<u>The method equals() has an NPath complexity of 2048</u>
35	nz\ac\massey\rimsgroup3\metadata\bean\Conference.java	10	<u>The class 'Conference' has a Cyclomatic Complexity of 3 (Highest = 19).</u>
36	nz\ac\massey\rimsgroup3\metadata\bean\Conference.java	91	<u>Ensure you override both equals() and hashCode()</u>
37	nz\ac\massey\rimsgroup3\metadata\bean\Conference.java	91	<u>The method 'equals' has a Cyclomatic Complexity of 19.</u>
38	nz\ac\massey\rimsgroup3\metadata\bean\Conference.java	91	<u>The method equals() has an NPath complexity of 8192</u>
39	nz\ac\massey\rimsgroup3\metadata\bean\Journal.java	10	<u>The class 'Journal' has a Cyclomatic Complexity of 3 (Highest = 16).</u>
40	nz\ac\massey\rimsgroup3\metadata\bean\Journal.java	76	<u>Ensure you override both equals() and hashCode()</u>
41	nz\ac\massey\rimsgroup3\metadata\bean\Journal.java	76	<u>The method 'equals' has a Cyclomatic Complexity of 16.</u>
42	nz\ac\massey\rimsgroup3\metadata\bean\Journal.java	76	<u>The method equals() has an NPath complexity of 2048</u>
43	nz\ac\massey\rimsgroup3\metadata\bean\Person.java	10	<u>The class 'Person' has a Cyclomatic Complexity of 3 (Highest = 16).</u>
44	nz\ac\massey\rimsgroup3\metadata\bean\Person.java	21	<u>Ensure you override both equals() and hashCode()</u>
45	nz\ac\massey\rimsgroup3\metadata\bean\Person.java	21	<u>The method 'equals' has a Cyclomatic Complexity of 16.</u>
46	nz\ac\massey\rimsgroup3\metadata\bean\Person.java	21	<u>The method equals() has an NPath complexity of 2048</u>
47	nz\ac\massey\rimsgroup3\metadata\bean\Publication.java	12	<u>The class 'Publication' has a Cyclomatic Complexity of 2 (Highest = 37).</u>
48	nz\ac\massey\rimsgroup3\metadata\bean\Publication.java	15	<u>Found non-transient, non-static member. Please mark as transient or provide accessors.</u>
49	nz\ac\massey\rimsgroup3\metadata\bean\Publication.java	41	<u>Ensure you override both equals() and hashCode()</u>
50	nz\ac\massey\rimsgroup3\metadata\bean\Publication.java	41	<u>The method 'equals' has a Cyclomatic Complexity of 37.</u>
51	nz\ac\massey\rimsgroup3\metadata\bean\Publication.java	41	<u>The method equals() has an NPath complexity of 3354432</u>
52	nz\ac\massey\rimsgroup3\metadata\plugin\CrossrefRetriever.java	30	<u>The class 'CrossrefRetriever' has a Cyclomatic Complexity of 4 (Highest = 14).</u>


53	nz\ac\massey\rimsgroup3\metadata\plugin\CrossrefRetriever.java	55	<u>The method 'processResult' has a Cyclomatic Complexity of 14.</u>
54	nz\ac\massey\rimsgroup3\metadata\plugin\CrossrefRetriever.java	55	<u>The method processResult() has an NPath complexity of 432</u>
55	nz\ac\massey\rimsgroup3\metadata\plugin\CrossrefRetriever.java	84	<u>Avoid empty catch blocks</u>
56	nz\ac\massey\rimsgroup3\metadata\plugin\CrossrefRetriever.java	93	<u>Avoid empty catch blocks</u>
57	nz\ac\massey\rimsgroup3\metadata\plugin\CrossrefRetriever.java	126	<u>Avoid empty catch blocks</u>
58	nz\ac\massey\rimsgroup3\metadata\plugin\CrossrefRetriever.java	141	<u>Avoid using implementation types like 'ArrayList'; use the interface instead</u>
59	nz\ac\massey\rimsgroup3\metadata\plugin\CrossrefRetriever.java	161	<u>Avoid if (x != y) ..; else ..;</u>
60	nz\ac\massey\rimsgroup3\metadata\plugin\CrossrefRetriever.java	221	<u>Avoid reassigning parameters such as 'result'</u>
61	nz\ac\massey\rimsgroup3\metadata\plugin\CrossrefRetriever.java	307	<u>Avoid unused local variables such as 'config'.</u>
62	nz\ac\massey\rimsgroup3\metadata\plugin\CrossrefRetriever.java	358	<u>Avoid if (x != y) ..; else ..;</u>
63	nz\ac\massey\rimsgroup3\metadata\plugin\ScopusRetriever.java	30	<u>The class 'ScopusRetriever' has a Cyclomatic Complexity of 5 (Highest = 14).</u>
64	nz\ac\massey\rimsgroup3\metadata\plugin\ScopusRetriever.java	54	<u>Avoid reassigning parameters such as 'json'</u>
65	nz\ac\massey\rimsgroup3\metadata\plugin\ScopusRetriever.java	54	<u>The method 'decodeJson' has a Cyclomatic Complexity of 14.</u>
66	nz\ac\massey\rimsgroup3\metadata\plugin\ScopusRetriever.java	54	<u>The method decodeJson() has an NPath complexity of 768</u>
67	nz\ac\massey\rimsgroup3\metadata\plugin\ScopusRetriever.java	75	<u>Position literals first in String comparisons</u>
68	nz\ac\massey\rimsgroup3\metadata\plugin\ScopusRetriever.java	75	<u>Position literals first in String comparisons</u>
69	nz\ac\massey\rimsgroup3\metadata\plugin\ScopusRetriever.java	75	<u>Position literals first in String comparisons</u>
70	nz\ac\massey\rimsgroup3\metadata\plugin\ScopusRetriever.java	99	<u>Avoid if (x != y) ..; else ..;</u>
71	nz\ac\massey\rimsgroup3\metadata\plugin\ScopusRetriever.java	145	<u>Too many fields</u>
72	nz\ac\massey\rimsgroup3\metadata\plugin\ScopusRetriever.java	178	<u>Avoid unused local variables such as 'config'.</u>
73	nz\ac\massey\rimsgroup3\metadata\plugin\ScopusRetriever.java	218	<u>Avoid if (x != y) ..; else ..;</u>
74	nz\ac\massey\rimsgroup3\servlet\AdvancedRequest.java	3	<u>Avoid duplicate imports such as 'java.io.IOException'</u>

75	nz\ac\massey\rimsgroup3\servlet\AdvancedRequest.java	3	<u>Avoid duplicate imports such as 'java.io.IOException'</u>
76	nz\ac\massey\rimsgroup3\servlet\AdvancedRequest.java	4	<u>Avoid duplicate imports such as 'java.io.InputStream'</u>
77	nz\ac\massey\rimsgroup3\servlet\AdvancedRequest.java	4	<u>Avoid duplicate imports such as 'java.io.InputStream'</u>
78	nz\ac\massey\rimsgroup3\servlet\AdvancedRequest.java	28	<u>Document empty constructor</u>
79	nz\ac\massey\rimsgroup3\servlet\AdvancedRequest.java	94	<u>Avoid if (x != y) ..; else ..;</u>
80	nz\ac\massey\rimsgroup3\servlet\AdvancedRequest.java	99	<u>Avoid empty catch blocks</u>
81	nz\ac\massey\rimsgroup3\servlet\AdvancedRequest.java	117	<u>When doing a String.toLowerCase()/toUpperCase() call, use a Locale</u>
82	nz\ac\massey\rimsgroup3\servlet\AdvancedRequest.java	135	<u>Avoid if (x != y) ..; else ..;</u>
83	nz\ac\massey\rimsgroup3\servlet\AutoComplete.java	18	<u>Found non-transient, non-static member. Please mark as transient or provide accessors.</u>
84	nz\ac\massey\rimsgroup3\servlet\AutoComplete.java	47	<u>When doing a String.toLowerCase()/toUpperCase() call, use a Locale</u>
85	nz\ac\massey\rimsgroup3\servlet\AutoComplete.java	65	<u>When doing a String.toLowerCase()/toUpperCase() call, use a Locale</u>
86	nz\ac\massey\rimsgroup3\servlet\AutoComplete.java	77	<u>Avoid unnecessary comparisons in boolean expressions</u>
87	nz\ac\massey\rimsgroup3\servlet\AutoComplete.java	100	<u>Document empty method</u>
88	nz\ac\massey\rimsgroup3\servlet\CommitRequest.java	25	<u>The class 'CommitRequest' has a Cyclomatic Complexity of 3 (Highest = 12).</u>
89	nz\ac\massey\rimsgroup3\servlet\CommitRequest.java	27	<u>Found non-transient, non-static member. Please mark as transient or provide accessors.</u>
90	nz\ac\massey\rimsgroup3\servlet\CommitRequest.java	28	<u>Found non-transient, non-static member. Please mark as transient or provide accessors.</u>
91	nz\ac\massey\rimsgroup3\servlet\CommitRequest.java	49	<u>Avoid really long methods.</u>
92	nz\ac\massey\rimsgroup3\servlet\CommitRequest.java	49	<u>The method 'processRequest' has a Cyclomatic Complexity of 12.</u>
93	nz\ac\massey\rimsgroup3\servlet\CommitRequest.java	72	<u>When doing a String.toLowerCase()/toUpperCase() call, use a Locale</u>
94	nz\ac\massey\rimsgroup3\servlet\CommitRequest.java	80	<u>Avoid unnecessary comparisons in boolean expressions</u>
95	nz\ac\massey\rimsgroup3\servlet\CommitRequest.java	111	<u>Avoid if (x != y) ..; else ..;</u>
96	nz\ac\massey\rimsgroup3\servlet\CommitRequest.java	133	<u>Ensure that resources like this Connection object are closed after use</u>



























97	nz\ac\massey\rimsgroup3\servlet\DXDoiRequest.java	31	<u>Avoid empty catch blocks</u>
98	nz\ac\massey\rimsgroup3\servlet\DXDoiRequest.java	42	<u>Avoid empty catch blocks</u>
99	nz\ac\massey\rimsgroup3\servlet\DoiRequest.java	29	<u>Found non-transient, non-static member. Please mark as transient or provide accessors.</u>
100	nz\ac\massey\rimsgroup3\servlet\DoiRequest.java	30	<u>Found non-transient, non-static member. Please mark as transient or provide accessors.</u>
101	nz\ac\massey\rimsgroup3\servlet\DoiRequest.java	52	<u>Ensure that resources like this Connection object are closed after use</u>
102	nz\ac\massey\rimsgroup3\servlet\DoiRequest.java	69	<u>Avoid unnecessary comparisons in boolean expressions</u>
103	nz\ac\massey\rimsgroup3\servlet\DoiRequest.java	83	<u>Avoid if (x != y) ..; else ..;</u>

## 9.8 Appendix H – Issue tracking

 **rimsgroup3**  
RIMS Project Group 3

Select: [All](#) [None](#) Actions...

1 - 24 of 24 [List](#) | [Grid](#)

ID	Type	Status	Priority	Milestone	Owner	Summary + Labels	Opened	Closed	...
<input type="checkbox"/>  1	Defect	Done	Low	----	eskimobob11	test issue	Jul 26	Aug 16	
<input type="checkbox"/>  2	Task	Done	Medium	Release1.0	eskimobob11	Tomcat and Eclipse at Massey University <a href="#">Usability</a>	Aug 09	Aug 16	
<input type="checkbox"/>  3	Defect	Done	High	----	steve@chimeradesign.co.nz	Can't run results.jsp on server	Aug 20	Aug 25	
<input type="checkbox"/>  4	Defect	Done	Medium	----	eskimobob11	Javascript run on Tomcat not responding	Sep 05	Sep 05	
<input type="checkbox"/>  5	Defect	Done	Medium	----	steve@chimeradesign.co.nz	Advanced search returning 400 error - bad request	Sep 10	Sep 10	
<input type="checkbox"/>  6	Enhancement	Done	Low	----	goodundead@Hotmail.com	Code review request	Sep 13	Sep 13	
<input type="checkbox"/>  7	Defect	Done	High	----	goodundead@Hotmail.com	no relative path	Sep 16	Sep 22	
<input type="checkbox"/>  8	Defect	Done	Medium	----	goodundead@Hotmail.com	Failure to commit to DB	Sep 16	Sep 28	
<input type="checkbox"/>  9	Defect	Fixed	Medium	----	steve@chimeradesign.co.nz	doi error	Sep 23	Sep 29	
<input type="checkbox"/>  10	Defect	Done	Medium	----	goodundead@Hotmail.com	Failure to connect with database, no error shown	Sep 24	Sep 28	
<input type="checkbox"/>  11	Defect	Done	Medium	----	leithbade	Emma not working properly	Sep 26	2 days ago	
<input type="checkbox"/>  12	Defect	Done	Medium	----	goodundead@Hotmail.com	Committing authors that are already in database.	Sep 28	Sep 29	
<input type="checkbox"/>  13	Defect	Done	Medium	----	goodundead@Hotmail.com	Adding new authors to JSP	Sep 28	Oct 03	
<input type="checkbox"/>  14	Defect	Done	Low	----	steve@chimeradesign.co.nz	DX-Doi tool not responding	Sep 29	3 days ago	
<input type="checkbox"/>  15	Defect	WontFix	Low	----	goodundead@Hotmail.com	Internet Explorer Inappropriate caching!?	Sep 29	Oct 03	
<input type="checkbox"/>  16	Defect	Done	Medium	----	eskimobob11	Enter Button	Sep 29	5 days ago	
<input type="checkbox"/>  17	Defect	Done	Medium	----	steve@chimeradesign.co.nz	Autocomplete on publisher text field failing	Oct 01	15 hours ago	
<input type="checkbox"/>  18	Defect	Done	Medium	----	goodundead@Hotmail.com	First and middle names with full stops	Oct 03	Oct 04	
<input type="checkbox"/>  19	Defect	Done	Low	----	steve@chimeradesign.co.nz	DX-DOI class not found	6 days ago	2 days ago	
<input type="checkbox"/>  20	Enhancement	Done	Medium	----	eskimobob11	Clean up TODOs	3 days ago	2 hours ago	
<input type="checkbox"/>  21	Defect	Done	High	----	steve@chimeradesign.co.nz	DOI Type	25 hours ago	2 hours ago	
<input type="checkbox"/>  22	Defect	Done	Medium	----	goodundead@Hotmail.com	IE foils me again ( autocomplete in IE failing)	24 hours ago	14 hours ago	
<input type="checkbox"/>  23	Defect	Done	Medium	----	leithbade	CrossRefRetreiverTest not passing	2 hours ago	moments ago	
<input type="checkbox"/>  24	Defect	Done	Medium	----	leithbade	ScopusRetreiverTest not passing	2 hours ago	moments ago	

1 - 24 of 24 [CSV](#)

No issues are left open at the end of the project.

Number of closed issues: 24

Average time to close issues: 3-4 days.