

System and Software Architecture Description (SSAD)

Data Mining of Digital Library Usage Data

Client: Jewel Ward

Team #7

Bo Lee

Genesan Kim

Maxim Krivokon

Vu Nguyen

Version History

Date	Author	Version	Changes made
9/28/04	Hsiao-Han Huang Pei-Han Li	1.0	<ul style="list-style-type: none">• Early Section
10/08/04	Hsiao-Han Huang Pei-Han Li	1.1	<ul style="list-style-type: none">• Correction and add diagrams and tables
10/14/04	Hsiao-Han Huang Pei-Han Li	1.2	<ul style="list-style-type: none">• Finish the section 3
10/24/04	Hsiao-Han Huang Pei-Han Li	2.0	<ul style="list-style-type: none">• Correction and add 3.5
11/17/04	Hsiao-Han Huang Pei-Han Li	3.1	<ul style="list-style-type: none">• Correct section 2 section 3 and add section4
12/6/04	Hsiao-Han Huang Pei-Han Li	4.0	<ul style="list-style-type: none">• Complete SSAD at LCA stage
02/09/05	Maxim Krivokon	5.0	<ul style="list-style-type: none">•

Table of Contents

SYSTEM AND SOFTWARE ARCHITECTURE DESCRIPTION (SSAD)	1
VERSION HISTORY	2
TABLE OF CONTENTS	3
TABLE OF TABLES	4
TABLE OF FIGURES	6
1. Introduction.....	9
1.1 Purpose of the SSAD Document.....	9
1.2 Standards and Conventions.....	9
1.3 References.....	9
1.4 Change Summary.....	11
2. System Analysis.....	12
2.1 Structure.....	12
2.2 Artifacts & Information	13
2.3 Behavior.....	15
2.4 L.O.S. Goals.....	25
2.5 Rules	25
3. Architecture Design & Analysis	26
3.1 Structure.....	26
3.2 Analysis Classes.....	50
3.3 Behavior.....	51
3.4 L.O.S., Projected.....	58
3.5 Architectural Styles, Patterns & Frameworks.....	59
Table 18 Architectural Styles, Patterns & Frameworks.....	59
4. Implementation Design.....	60
4.1 Structure.....	60
4.2 Behavior.....	82
4.3 L.O.S.....	88
4.4 Patterns & Frameworks.....	89
4.5 Project Artifacts	89
5. Glossary for System Analysis and Design.....	91
6. Appendices.....	91

Table of Tables

<i>Table 1 Stakeholders Descriptions.....</i>	<i>Error! Bookmark not defined.</i>
<i>Table 2 Actor1.....</i>	<i>Error! Bookmark not defined.</i>
<i>Table 3 Actor2</i>	<i>Error! Bookmark not defined.</i>
<i>Table 4 Actor 3.....</i>	<i>Error! Bookmark not defined.</i>
<i>Table 5 Use-Case Description for Manage log data process.....</i>	<i>Error! Bookmark not defined.</i>
<i>Table 6 Use-Case Description for log analysis process.....</i>	<i>Error! Bookmark not defined.</i>
<i>Table 7 Use-Case Description for visualization process.....</i>	<i>Error! Bookmark not defined.</i>
<i>Table 8 L.O.S Goals.....</i>	<i>25</i>
<i>Table 9Use case description for Delete data</i>	<i>35</i>
<i>Table 10Use case description for Import data.....</i>	<i>36</i>
<i>Table 11 L.O.S goals for Interface to import log data.....</i>	<i>37</i>
<i>Table 12Use case description for Generate report.....</i>	<i>38</i>
<i>Table 13Use case description for delete report</i>	<i>40</i>
<i>Table 14 L.O.S goals for log analysis</i>	<i>42</i>
<i>Table 15Use case description for Browse graph.....</i>	<i>43</i>
<i>Table 16 L.O.S goals for Visualization of analysis result</i>	<i>45</i>
<i>Table 17 L.O.S goals for Database</i>	<i>46</i>
<i>Table 18 Use case description for Manage log data process.....</i>	<i>52</i>

<i>Table 19 Use case description for Log analysis process.....</i>	<i>55</i>
<i>Table 20 Use case description for Visualization process.....</i>	<i>57</i>
<i>Table 21 L.O.S projected.....</i>	<i>59</i>
<i>Table 22 Architectural Styles, Patterns & Frameworks.....</i>	<i>59</i>
<i>Table 23 Use case description for Delete data.....</i>	<i>65</i>
<i>Table 24 Use case description for Import data.....</i>	<i>66</i>
<i>Table 25L.O.S goal for Interface to import log data.....</i>	<i>67</i>
<i>Table 26L.O.S. Goals for SQLite database.....</i>	<i>69</i>
<i>Table 27Use case description for Generate report.....</i>	<i>70</i>
<i>Table 28Use case description for Delete report.....</i>	<i>72</i>
<i>Table 29L.O.S goal for Log analysis.....</i>	<i>74</i>
<i>Table 30Use case description for Browse graph.....</i>	<i>76</i>
<i>Table 31L.O.S goals for Visualization of analysis result.....</i>	<i>77</i>
<i>Table 32 Use case description for Manage log data process.....</i>	<i>84</i>
<i>Table 33 Use case description for Log analysis.....</i>	<i>86</i>
<i>Table 34 Use case description for Visualization process.....</i>	<i>88</i>
<i>Table 35L.O.S projected.....</i>	<i>89</i>
<i>Table 36 Architectural Styles, Patterns & Frameworks.....</i>	<i>89</i>

Table of Figures

<i>Figure 1 System Analysis Package.....</i>	<i>Error! Bookmark not defined.</i>
<i>Figure 2 System Static Structure Diagram</i>	<i>12</i>
<i>Figure 3 System Collaboration Diagram.....</i>	<i>13</i>
<i>Figure 4 System Artifacts</i>	<i>14</i>
<i>Figure 5 Behavior Model Use-Case Diagram.....</i>	<i>16</i>
<i>Figure 6 Activity diagram for Manage log data process - delete.....</i>	<i>Error! Bookmark not defined.</i>
<i>Figure 7 Activity diagram for Manage log data process - import</i>	<i>Error! Bookmark not defined.</i>
<i>Figure 8 Activity diagram for Log Analysis process - generate report</i>	<i>Error! Bookmark not defined.</i>
<i>Figure 9 Activity diagram for Log Analysis process - delete report.....</i>	<i>Error! Bookmark not defined.</i>
<i>Figure 10 Activity diagram for Visualization process.....</i>	<i>Error! Bookmark not defined.</i>
<i>Figure 11 Topology</i>	<i>Error! Bookmark not defined.</i>
<i>Figure 12 Hardware Classifier Model.....</i>	<i>28</i>
<i>Figure 13 Software Classifier Model.....</i>	<i>Error! Bookmark not defined.</i>
<i>Figure 14 Deployment Model.....</i>	<i>31</i>
<i>Figure 15 Use-Case diagram for interface to import log data</i>	<i>34</i>
<i>Figure 16 Activity diagram for delete data.....</i>	<i>35</i>
<i>Figure 17 Activity diagram for Import data</i>	<i>36</i>
<i>Figure 18 Use-Case diagram for Log analysis.....</i>	<i>37</i>
<i>Figure 19 Activity diagram for generate report</i>	<i>39</i>
<i>Figure 20 Activity diagram for delete report.....</i>	<i>41</i>

<i>Figure 21 Use-Case diagram for visualization of log analysis result.....</i>	<i>42</i>
<i>Figure 22: Activity diagram for Browse graph.....</i>	<i>44</i>
<i>Figure 23 Use-Case diagram for Database.....</i>	<i>46</i>
<i>Figure 24 Software component model</i>	<i>48</i>
<i>Figure 25 Class Model: Analysis Classes.....</i>	<i>50</i>
<i>Figure 26 Behavior Model Use-Case Diagram.....</i>	<i>51</i>
<i>Figure 27 Sequence diagram for Manage log data process-import data</i>	<i>53</i>
<i>Figure 28 Sequence diagram for Manage log data process-delete.....</i>	<i>54</i>
<i>Figure 29 Sequence diagram for Log analysis process -generate report</i>	<i>56</i>
<i>Figure 30 Sequence diagram for log analysis process-delete report.....</i>	<i>57</i>
<i>Figure 31 Sequence diagram for visualization process.....</i>	<i>58</i>
<i>Figure 32 Topology</i>	<i>60</i>
<i>Figure 33 Hardware classifier model</i>	<i>61</i>
<i>Figure 34 Software classifier model</i>	<i>62</i>
<i>Figure 35 Deployment model</i>	<i>63</i>
<i>Figure 36 Use-Case diagram for Interface to import log data</i>	<i>64</i>
<i>Figure 37 Activity diagram for Delete data.....</i>	<i>65</i>
<i>Figure 38 Activity diagram for Import data</i>	<i>66</i>
<i>Figure 39 Internal Architecture for Interface to import log data</i>	<i>67</i>
<i>Figure 40 Use-Case diagram for SQLite Database.....</i>	<i>68</i>
<i>Figure 41 Use-Case diagram for log analysis.....</i>	<i>69</i>
<i>Figure 42 Activity diagram for Generate report.....</i>	<i>71</i>

<i>Figure 43 Activity diagram for Delete report.....</i>	<i>73</i>
<i>Figure 44 : Internal Architecture: Log analysis.....</i>	<i>74</i>
<i>Figure 45 Use-Case diagram for Visualization of log analysis result.....</i>	<i>75</i>
<i>Figure 46: Activity diagram for Browse graph.....</i>	<i>76</i>
<i>Figure 47 Behavior Model Use-Case Diagram.....</i>	<i>83</i>
<i>Figure 48 Sequence diagram for Manage log data process-import data.....</i>	<i>84</i>
<i>Figure 49 Sequence diagram for Manage log data process-delete.....</i>	<i>85</i>
<i>Figure 50 Sequence diagram for Log analysis -generate report.....</i>	<i>87</i>
<i>Figure 51 Sequence diagram for log analysis-delete report.....</i>	<i>87</i>
<i>Figure 52 Sequence diagram for visualization process.....</i>	<i>88</i>

1. Introduction

1.1 Purpose of the SSAD Document

This document is a refinement of architecture description compiled during the inception phase. The purpose of this document is to bring all the necessary details into architecture model and make it implementation specific. This document provides clear and sufficient description of system's structure and design and thus makes transition to implementation phase more efficient.

1.2 Standards and Conventions

- Standard
 - MBASE Guideline for 577b version 2.4.2
- Notation
 - UML: version 1.4
- Naming Conventions
 - Components and object are Nouns.
 - Behaviors and Operations are Verbs.

1.3 References

MBASE Guidelines version 2.4.2

http://sunset.usc.edu/classes/cs577a_2004/guidelines/MBASE_Guidelines_v2.4.1.pdf

MBASE Electronic Process Guide

<http://cse.usc.edu/research/MBASE/EPG>

USC Information Services Division

<http://www.usc.edu/isd/about/about.html>

Fall 2004 CS 577a Project #7 Description

http://sunset.usc.edu/classes/cs577a_2004/projects/description/project7.htm

UML Guidelines

http://sunset.usc.edu/classes/cs577a_2003/coursenotes/ep/Introduction_to_UML.pdf

CS577a – Software Engineering I website

http://sunset.usc.edu/classes/cs577a_2004/

Fall 2003 CS 577a Project #8 LCO portion of OCD

http://ebase.usc.edu/eservices/cs577a_2003/team08a/LCO/OCD_LCO_F03a_T08.pdf

Fall 2003 CS 577a Project #8 LCO portion of SSAD

http://www-scf.usc.edu/%7Ecsci577/www/team8a/LCO/SSAD_LCO_F03a_T08.doc

Fall 2003 CS 577a Project #22 LCO portion of SSAD

http://www-scf.usc.edu/%7Ecsci577/www/team22a/LCO/SSAD_LCO_F03a_T22.doc

[c](#)

Fall 2003 CS 577a Project #15 LCO portion of SSAD

http://www-scf.usc.edu/%7Ecsci577/www/team15a/LCO/SSAD/SSAD_LCO_F03a_T15.doc

Fall 2004 CS 577a Project #7 OCD Document

1.4 Change Summary

- Section 2.1 – changed to one generic user – remove description for business workers – system does not care about that
- Introduced new artifacts – made it to correspond with the processes.
- Updated Behavior Model Use-Case Diagram
Replaced old processes by: new 6 use cases
Updated the Rose model correspondingly.
- Changed architecture to three layered system – Model View Controller is an established design pattern.
- All users are defined in OCD – references to them from all other diagrams
- OCD – no maintainer
- System boundary diagram - changed

2. System Analysis

This section refines the proposed system in terms of its interactions with its users and artifacts it operates on as it was defined in OCD 4 “Proposed system”.

2.1 Structure

We describe the actors who interact with the system when it is operational. The proposed system does not have different modes of behavior based on which business or outside actor is interacting with it. Therefore the system will have one generic user and will provide all of its capabilities in one operational mode.

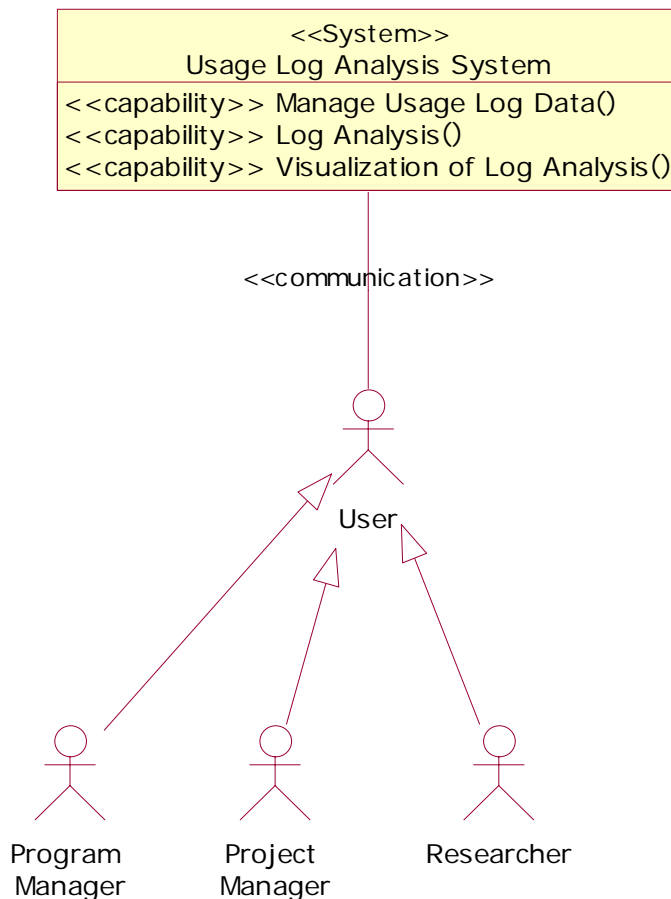


Figure 1 System Static Structure Diagram

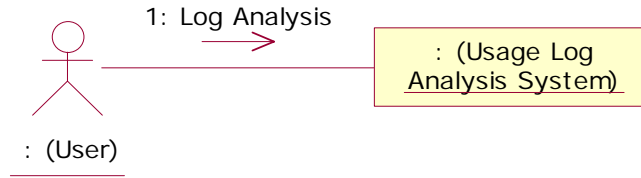


Figure 2 System Collaboration Diagram

Figure 3 shows one possible configuration of system usage – the user invokes one of the system’s capabilities – “Log Analysis”.

2.1.1 System

The proposed system is a tool that allows visualization and analysis of digital library usage data. Provided with input log files the system generates and visualizes digital library item collection structure that reflects similarity relationships between items that are viewed together most often. In summary the system provides the following major capabilities:

- Manage usage log data
- Log Analysis
- Visualization of log analysis result\

2.1.2 User

The user of the proposed system is a generic actor that is expected to have experience using other software and being familiar with common computer interface concepts such like input forms, action buttons etc. The user does not have any attributes used by the system and is not required to provide any services.

2.2 Artifacts & Information

The proposed system takes “Usage log files” as input data and analyses it in two dimensions: item similarity relationships, which produces “Item graph” artifact; and usage statistics, which results in “Item usage” artifact. The produced “Item graph” is analyzed further through graph clustering and centrality analysis to produce “Item tree” artifact which represents Digital Archive collection structure based on its usage. The “Item usage” and “Item tree” artifacts combined constitute “Usage Analysis Report”.

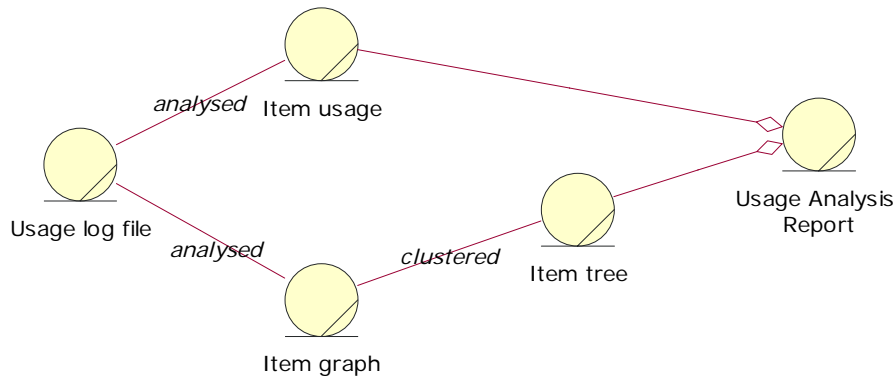


Figure 3 Artifact model

2.2.1 Usage Log File

The “Usage log file” is a text file with each line representing item retrieval from DA collection. This artifact is necessary as input data for the system’s “Log Analysis” capability. “Manage log data” capability allows the user to specify location of the input log file and import the usage data into the system’s database. The system tries to import the following data fields for each retrieval: time of retrieval, user id, item id.

2.2.2 Item graph

The “Log Analysis” capability mines the imported usage data for item relationships based on their co-retrievals and produces “Item graph” which represents those. The “Item graph” is stored in a database table where each record stands for one relationship.

2.2.3 Item usage

“Log Analysis” capability collects usage information for each item at the time of “Item graph” generation. The usage data is collected separately for each unique user id and stored in a database table.

2.2.4 Item tree

“Log Analysis” capability need to transform “Item graph” into “Item tree” so that the resulting information is compatible with input format of “Visualization” capability.

“Item tree” is a tree representation of “Item graph” generated by applying multi-level graph clustering and centrality analysis.

2.2.5 Usage analysis report

“Usage analysis report” is a composition of “Item usage”, “Item tree” and report meta-data that was entered at the moment of generation such like: report name, report author, notes, etc.

2.3 Behavior

This section shows the processes that realize the each of the system’s capability defined in OCD section 4.3.

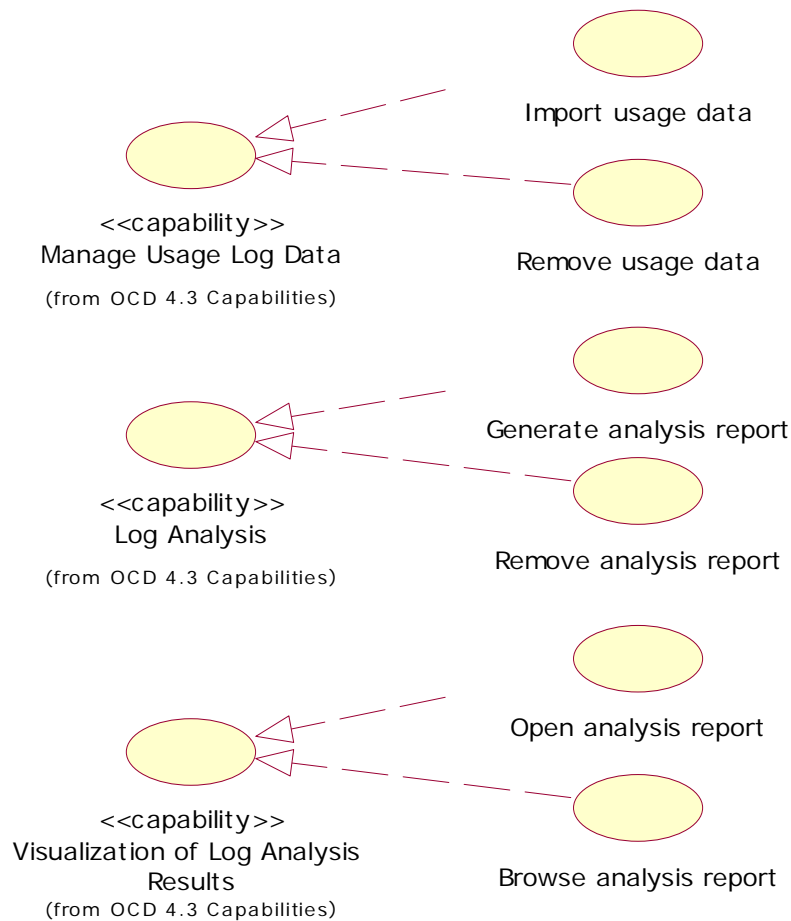


Figure 4 System Processes

As seen from the above diagram there are six processes that implement three major capabilities of the system. These processes will be described in more detail in the following sections.

2.3.1 Processes

2.3.1.1 Import usage data (UC-01)

Identifier	UC-01
Use-Case Name	Import Usage Data
Abstract	No
Purpose	Demonstrates how usage data is imported
Actors	User
Importance	Primary

Capability	C-01 (OCD 4.3.1)
Requirements	SR-1 (SSRD 3.2.1), IR-1 (SSRD 4.1.1)
Risks	None
High-Risk?	No
Architecturally Significant?	No
Development Status	RLCA
Overview	After user specifies source log file and initiates importing, system parses retrieval records from the file and stores them to the database.
User Interface	User is provided with standard file selection-browsing menu to pick the source log file. User is provided with a “Import” button to initiate importing of data.
Pre-conditions	Source log file is available from the local disk.
Post-conditions	System’s database contains representation of all well-formatted retrieval records from the source file
Specializes	None
Includes	None
Extends	None
Extension Points	None

Table 1 Use-Case Description for Import usage data

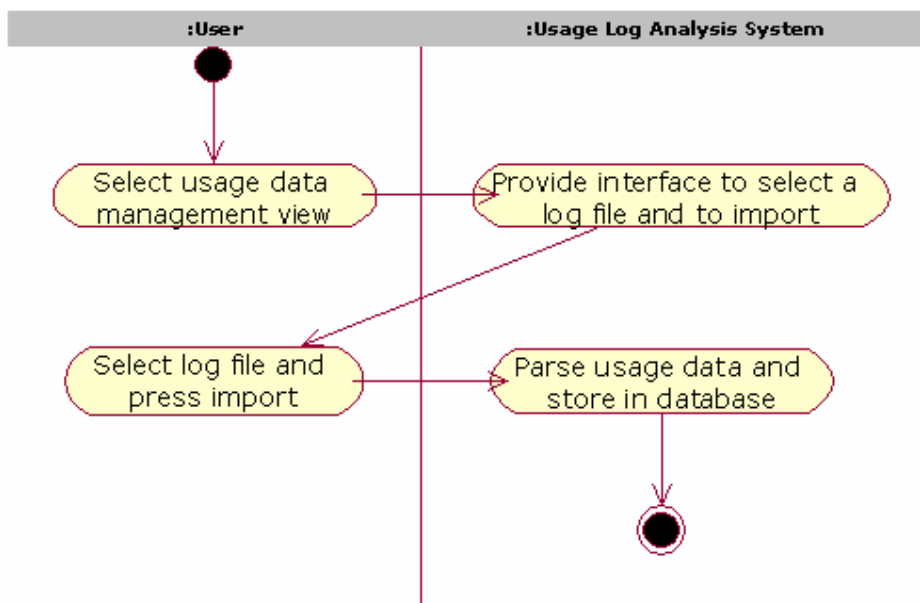


Figure 5 Activity diagram for Import usage data

2.3.1.2 Remove usage data (UC-02)

Identifier	UC-02
Use-Case Name	Remove usage data
Abstract	No
Purpose	Demonstrates how user can remove old or otherwise outdated usage information.
Actors	User
Importance	Primary
Capability	C-01 (OCD 4.3.1)
Requirements	SR-2 (SSRD 3.2.1), IR-1 (SSRD 4.1.1)
Risks	None
High-Risk?	No
Architecturally Significant?	No
Development Status	RLCA
Overview	User is presented with a list of filenames from which usage data was imported previously. After user selected a specific file and pressed delete button, the system removes all retrieval records that were imported from that file.
User Interface	Already imported files are presented in a scrollable selectable list.
Pre-conditions	Usage data was previously imported from one or more files
Post-conditions	System's database does not contain any retrieval record that was associated with the selected log file
Specializes	None
Includes	None
Extends	None
Extension Points	None

Table 2 Use-Case Description for Remove usage data

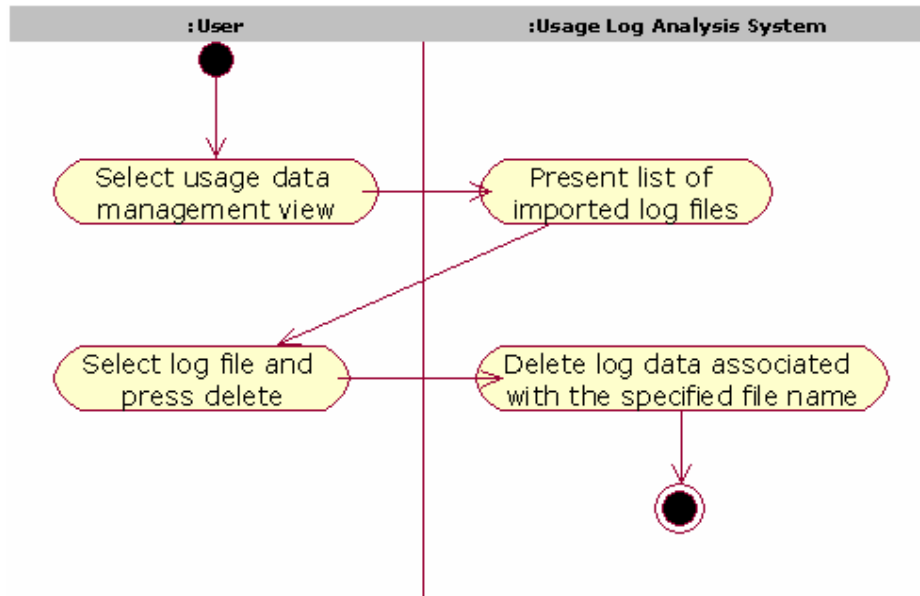


Figure 6 Activity diagram for Remove usage data

2.3.1.3 Generate analysis report (UC-03)

Identifier	UC-03
Use-Case Name	Generate analysis report
Abstract	No
Purpose	Demonstrates how user can generate analysis report
Actors	User
Importance	Primary
Capability	C-02 (OCD 4.3.2)
Requirements	SR-3,4,5 (SSRD 3.2.1), IR-2 (SSRD 4.1.1)
Risks	None
High-Risk?	No
Architecturally Significant?	No
Development Status	RLCA
Overview	In order to generate a new analysis report user has to select date range of usage data to be used and parameters for analysis algorithm. After the user initiates report generation, system generated item relationship graph from the specified usage data and collects usage statistic for each item. Based on the generated item relationship graph system produces

	collection structure tree by applying multi-level graph clustering and centrality analysis.
User Interface	User is provided with a form for inputting algorithm parameters.
Pre-conditions	Usage data was previously imported from one or more log files.
Post-conditions	System database contains representation of collection structure tree and usage statistics for each item in that tree.
Specializes	None
Includes	None
Extends	None
Extension Points	None

Table 3 Use-Case description for Generate analysis report

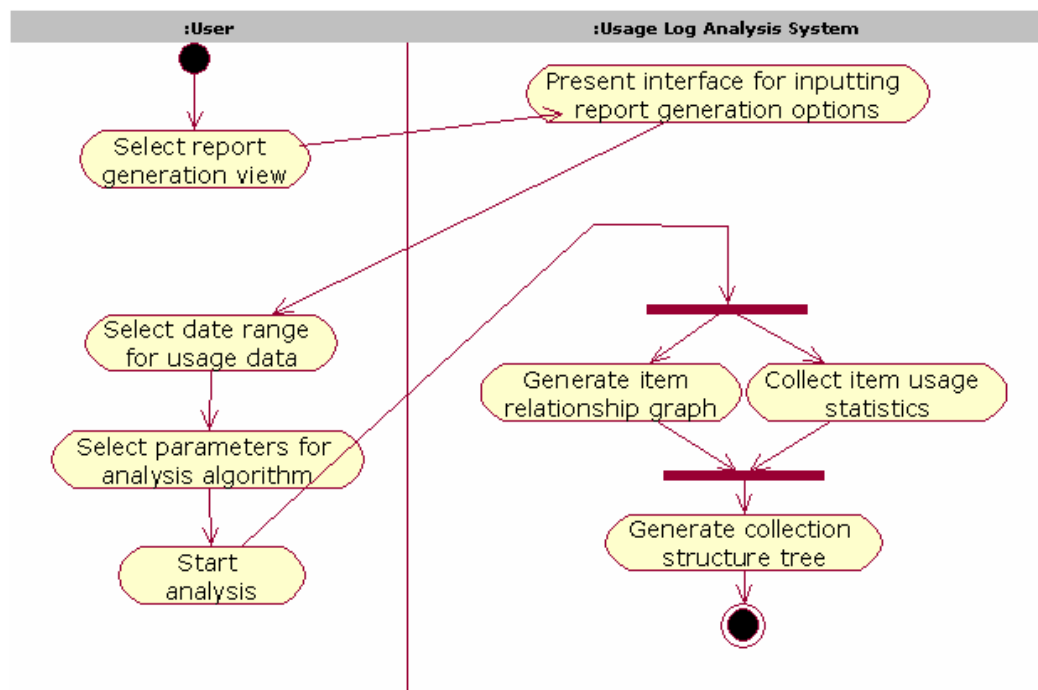


Figure 7 Activity diagram for Generate analysis report

2.3.1.4 Remove analysis report (UC-04)

Identifier	UC-04
Use-Case Name	Remove analysis report
Abstract	No
Purpose	Demonstrates how user can remove analysis report

Actors	User
Importance	Primary
Capability	C-02 (OCD 4.3.2)
Requirements	IR-4 (SSRD 4.1.1)
Risks	None
High-Risk?	No
Architecturally Significant?	No
Development Status	RLCA
Overview	User selects id of the analysis report to be deleted and pressed remove button
User Interface	User is presented with a scrollable selectable list of previously generated analysis reports and a button to initiate removal
Pre-conditions	One or more analysis reports have been generated
Post-conditions	System database does not contain any analysis information that was associated with the removed report
Specializes	None
Includes	None
Extends	None
Extension Points	None

Table 4 Use-Case Description for Remove analysis report

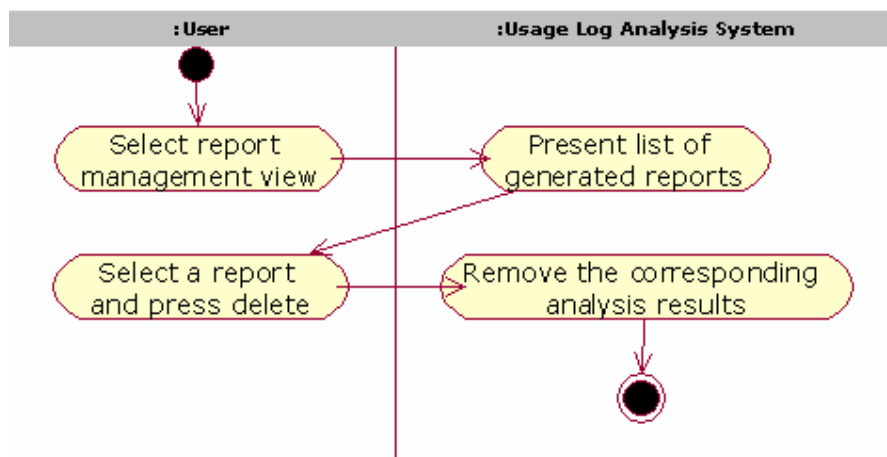


Figure 8 Activity diagram for Remove analysis report

2.3.1.5 Open analysis report (UC-05)

Identifier	UC-05
Use-Case Name	Open analysis report
Abstract	No
Purpose	Demonstrates how user selects an analysis report in order to visualize it
Actors	User
Importance	Primary
Capability	C-03 (OCD 4.3.2)
Requirements	SR-6 (SSRD 3.2.1) IR-4 (SSRD 4.1.1)
Risks	None
High-Risk?	No
Architecturally Significant?	No
Development Status	RLCA
Overview	In order to visualize an analysis report user selects it from a list of generated reports
User Interface	User is presented with a scrollable selectable list of previously generated analysis reports and a button to visualize them.
Pre-conditions	One or more analysis reports have been generated
Post-conditions	A new window with 3d visualization of the selected analysis report is opened
Specializes	None
Includes	None
Extends	None
Extension Points	None

Table 5 Use-Case Description for Open analysis report

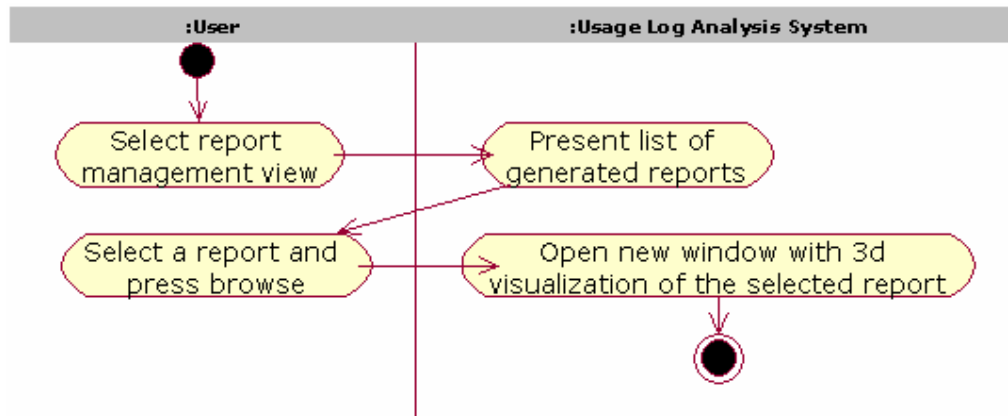


Figure 9 Activity diagram for Visualize analysis report

2.3.1.6 Browse analysis report (UC-05)

Identifier	UC-06
Use-Case Name	Browse analysis report
Abstract	No
Purpose	Demonstrates how user can interact with the visualization of the analysis report
Actors	User
Importance	Primary
Capability	C-03 (OCD 4.3.2)
Requirements	SR-6 (SSRD 3.2.1) IR-5 (SSRD 4.1.1)
Risks	None
High-Risk?	No
Architecturally Significant?	No
Development Status	RLCA
Overview	User can browse the 3d visualization of analysis report by panning and rotating the view using left and right mouse buttons correspondingly. User can examine usage statistic for a particular node by selecting it in the view.
User Interface	Collection structure tree is presented in a 3d hyperbolic interactive view.
Pre-conditions	Analysis report has been generated and selected for visualization.

Post-conditions	System changes the view and presented information based on the dynamic user input.
Specializes	None
Includes	None
Extends	None
Extension Points	None

Table 6 Use-Case description for Browse analysis report

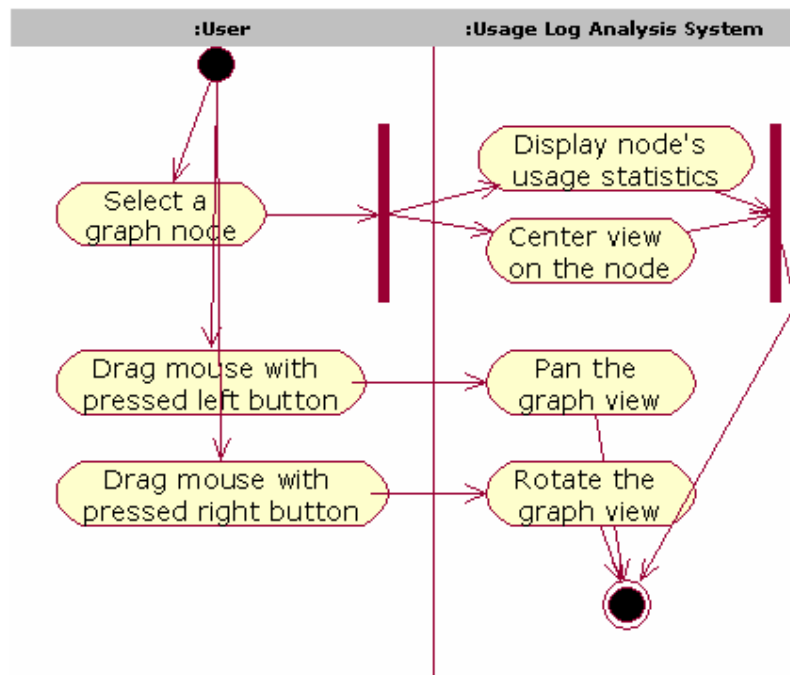


Figure 10 Activity diagram for Browse analysis report

2.3.2 Modes of Operation

2.3.2.1 Operational mode

The proposed system has only one mode of operation in which it will provide all the capabilities defined in OCD 4.3.

Capability	Processes	Mode Impact
Manage usage log data	<ul style="list-style-type: none"> Import usage data Remove usage data 	N / A
Log Analysis	<ul style="list-style-type: none"> Generate analysis report Remove analysis report 	N / A
Visualization of Log	<ul style="list-style-type: none"> Open analysis report 	N / A

Analysis	<ul style="list-style-type: none"> • Browse analysis report 	
----------	--	--

Table 7 Available Capabilities & Processes in Operational Mode

2.4 L.O.S. Goals

L.O.S requirement	Applies to	How
LR-1: Dependability[SSRD5]	Prc-1, Prc-2, Prc-3	Equally
LR-2: Usability[SSRD5]	Prc-3	Equally
LR-3: Operability in multitasking environment[SSRD5]	Prc-2	Equally
LR-4: Performance on data of current scale [SSRD5]	Prc-3	Equally

Table 8 L.O.S Goals

2.5 Rules

The proposed system will not change any current organization rules identified in OCD 4.5.4.

3. Architecture Design & Analysis

The architecture design and analysis provide a high-level architecture for the system and also explain the relationship between the components. This section will describe what are the work units (both hardware and software *components*) of the system and what the components are expected to do.

3.1 Structure

3.1.1 Topology

The proposed system is a hybrid of simple and composite system. The system is simple since it's a standalone application that has a primitive “Hardware classifiers model” and “Deployment model”. However software classifiers of this system are split into three layers: Interface, Controller and Model. Detailed description of each layer is provided in the following sections.

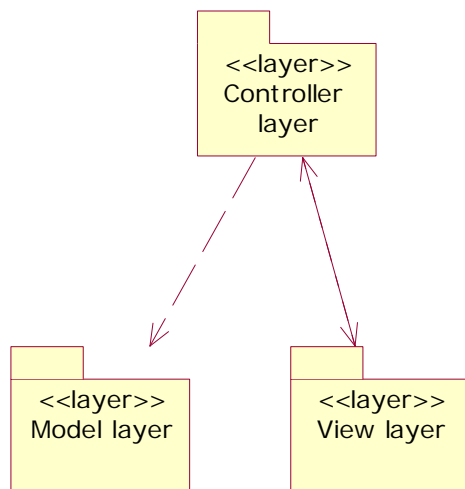


Figure 11 Topology of the system

3.1.1.1 View layer

The interface layer contains components that allow system to communicate with the user. These components provide functionality for taking user input and displaying system's output. In particular interface layer contains component providing three

dimensional hyperbolic visualization of a tree representing the Digital Archive collection.

Below is the list of interface components:

- Interface component
 - Importing usage data
 - Removing usage data
 - Generating new report
 - Removing report
 - Opening report
- Visualization component
 - Three dimensional view of collection tree
 - Display of usage statistics for selected item

3.1.1.2 Controller layer

Controller layer receives requests from Interface layer, performs requested operations, sends updates to the Model layer and notifies Interface about completion of operation.

This layer includes the following components:

- Dispatcher
- Data import
- Relationship & statistics generation
 - Generate item relationship graph
 - Collect item usage statistics
- Tree generation
 - Cluster item relationship graph
 - Perform centrality analysis
 - Generate item tree

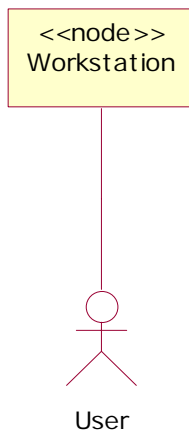
3.1.1.3 Model layer

Model layer provides persistent representation of the system states. This layer contains components that provide storage for input data and intermediate operational results of components. This layer contains one Database component that provides four interfaces to other components as follows:

- Usage data table
- Relationships table
- Statistics table
- Tree table

3.1.2 Hardware Classifier Model

This section describes hardware components that are either part of the system or on which this system will run and actors which will interact with the hardware components. The proposed system is a standalone application that will be installed and operated on a local workstation. The system does not have remotely deployed components and does not communication with other systems.



(from SSAD 2.1 System Context)

Figure 12 Hardware Classifier Model

3.1.3 Software Classifier Model

This section describes in detail software component models for each layer of the system (Figure 12), interactions between them and with the user.

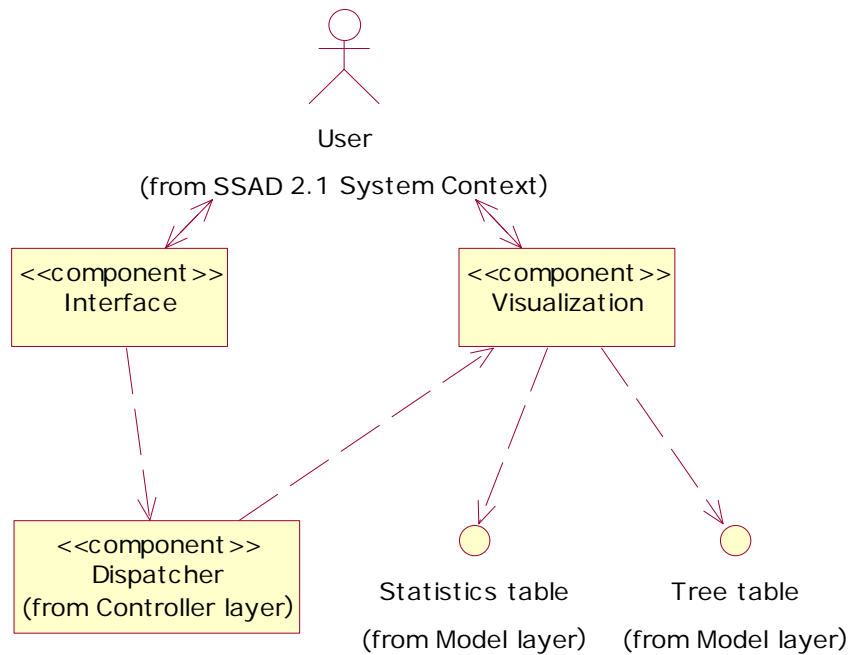


Figure 13 View layer component diagram

The Interface component of the View layer provides user with ability to pass input data and parameters to the system and invoke the system's functionality. Visualization component provides interactive three-dimensional display of generated item relations and usage statistics.

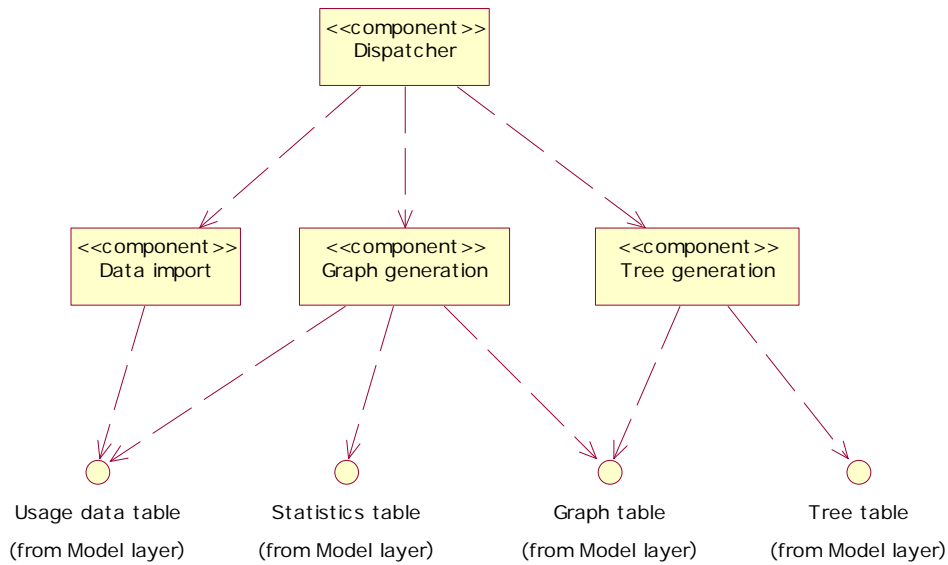


Figure 14 Controller layer component diagram

Controller layer contains components that provide data management and data mining capabilities. “Data import” component is responsible for parsing usage data from input log file, normalizing it and storing into the “Usage data table”. “Relationship & statistics generation component” analyses the imported usage data to produce item relationships based on their co-retrievals. As part of analysis process this component collects usage information for each item and stores it into the corresponding table. The “Tree generation component” produces tree representation of the item relationship graph using graph clustering and centrality analysis.

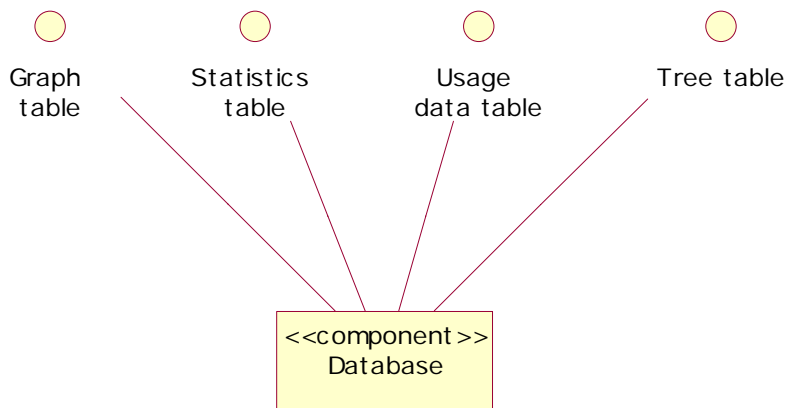


Figure 15 Model software component diagram

The “Model” layer contains “Database” component that provides several interfaces for data storage, searching and retrieval in form of SQL tables. Table design specializes SQL interface provided by the database by specifying number of columns, their names and data types.

3.1.4 Deployment Model

Deployment model describes component and connector configuration(s) that make a working version of the system. In this configuration, it describes the instances of hardware and software component that participate in the configuration, the allocation of software components to the hardware components.

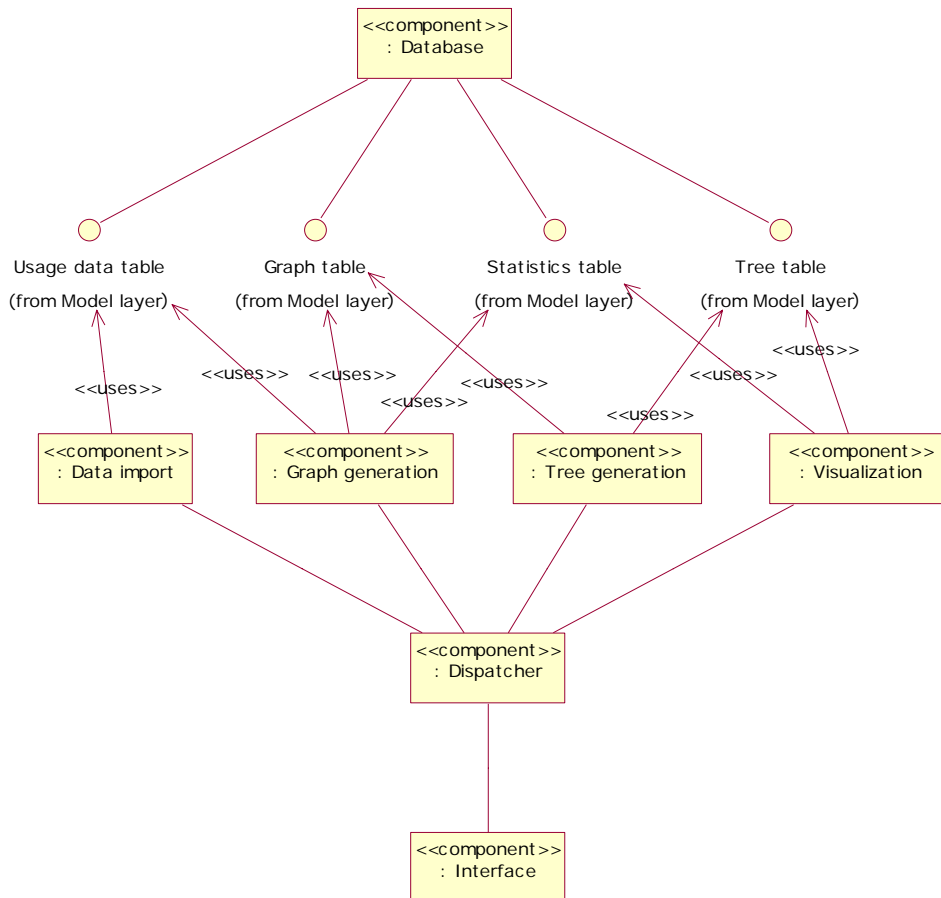


Figure 16 Deployment Model

3.1.5 Hardware Component Classifiers

3.1.5.1 Classifier: Workstation (HCC-01)

This section describes the only hardware component that will be used by the system – which is Workstation.

3.1.5.1.1 Purpose

The purpose of this hardware component is to serve as a host for the proposed application. This is the environment where all the software components of the system will be installed and operated.

3.1.5.1.2 L.O.S. Goals

None of the LOS requirements apply here.

3.1.6 Hardware Connector Classifiers

No Hardware Connector Classifiers have been defined because the proposed system is a standalone application.

3.1.7 Software Component Classifiers

3.1.7.1 Menus & Forms (SCC-01)

3.1.7.1.1 Purpose

The purpose of this component is to provide Graphical User Interface that allows user to input necessary information and invoke different system's functionalities.

3.1.7.1.2 Interfaces

The “Menus & Forms” component will have two major subcomponents “Manage usage data and “Manage analysis reports”. These components are responsible for interaction with the user and passing user input to other components therefore they do not perform any operations and do not have defined interfaces.

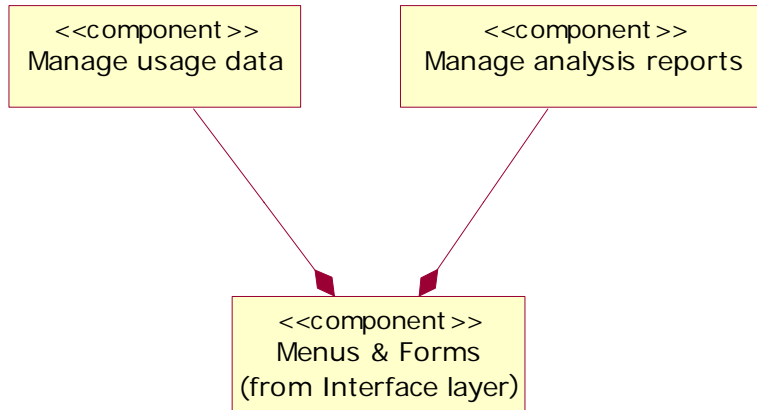


Figure 17 Interfaces & Parameters for SCC-01

3.1.7.1.2.1 Manage usage data

The component should present a list of imported usage data and provide interface to import new usage data or remove usage data that was imported before.

3.1.7.1.2.2 Manage analysis reports

This component should present a list of generated analysis reports and provide interface to create new analysis report or remove previously created one.

3.1.7.1.3 Parameters

The “Menus & Forms” component is a static component which has a predefined behavior and therefore does not have any parameters.

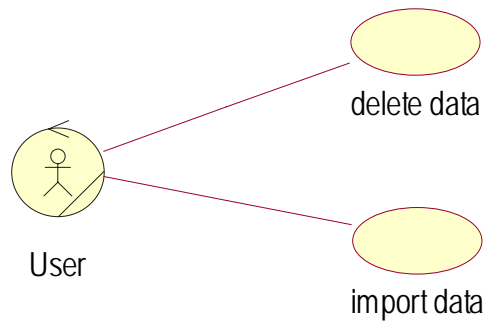
3.1.7.1.4 Behavior**3.1.7.1.4.1 Processes**

Figure 18 Use-Case diagram for interface to import log data

3.1.7.1.4.1.1**Process: delete data**

Identifier	Prc-1.1
Use-Case Name	Delete data
Abstract	No
Purpose	To delete log data
Actors	User
Importance	secondary
Capability	C-01 (OCD 4.3)
Requirements	SSRD. 3.2.1 (SR-2)
Risks	None
High-Risk?	No
Architecturally Significant?	No
Development Status	LCA
Overview	To delete log data that user has previously imported.
USER INTERFACE	See OCD 5 - prototype
Pre-conditions	Log data exists.
Post-conditions	Log data deleted.
Specializes	None
Includes	None
Extends	None

Extension Points	None
-------------------------	------

Table 9 Use case description for Delete data

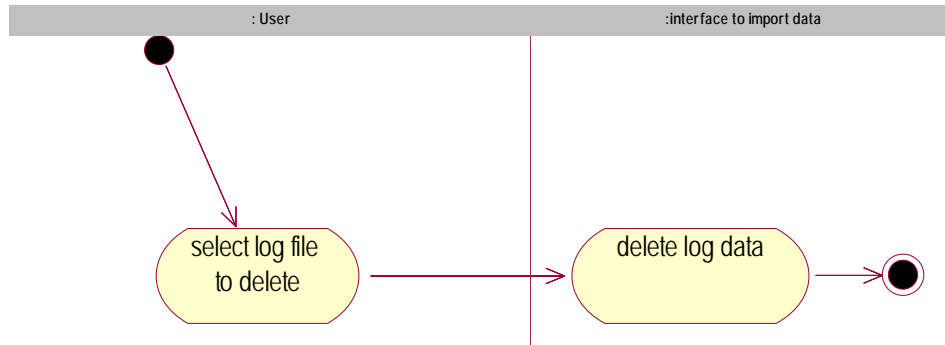


Figure 19 Activity diagram for delete data

3.1.7.1.4.1.2

Process: import data

Identifier	Prc-1.2
Use-Case Name	Import log data.
Abstract	No
Purpose	To import log data
Actors	User
Importance	secondary
Capability	C-01 (OCD 4.3)
Requirements	SSRD 3.2.1 (SR-1)
Risks	None
High-Risk?	No
Architecturally Significant?	No
Development Status	LCA
Overview	To import log data stored in user’s local machine to be analyzed.
USER INTERFACE	See OCD 5 - prototype
Pre-conditions	Log file exists.
Post-conditions	Log file imported.
Specializes	None
Includes	None

Extends	None
Extension Points	None

Table 10 Use case description for Import data

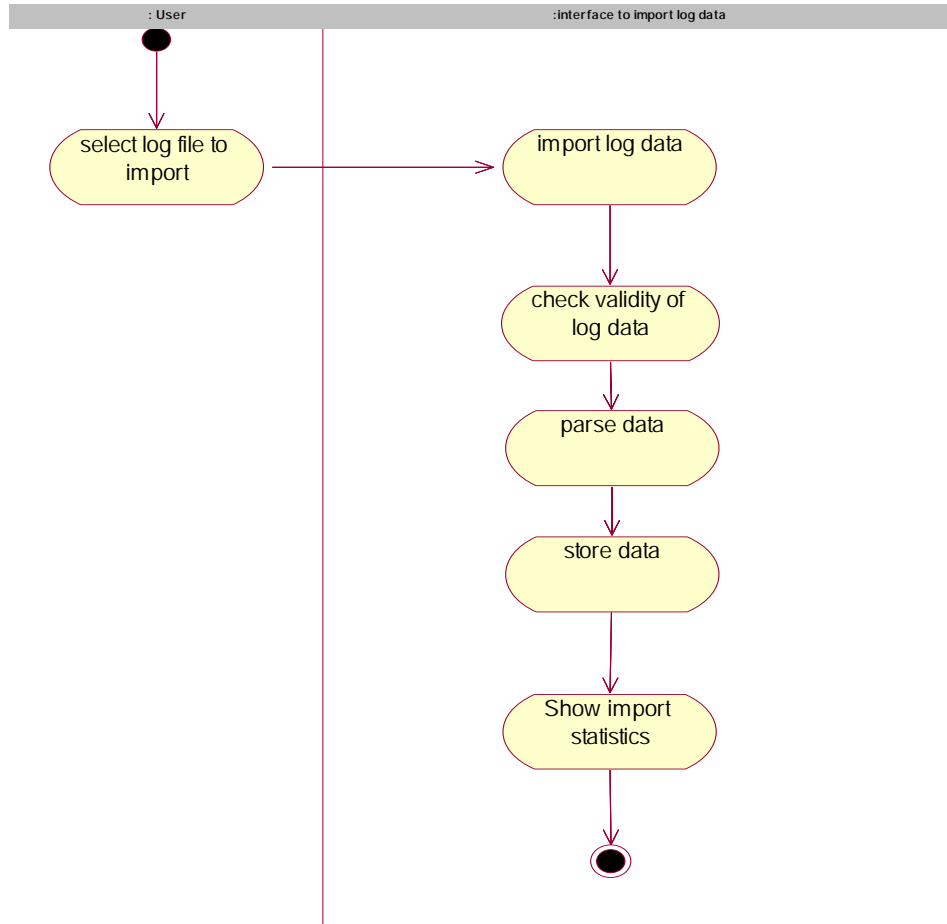


Figure 20 Activity diagram for Import data

3.1.7.1.5 L.O.S. Goals

L.O.S. Requirement:	Dependability [SSRD 5, LR-1]
Description:	This component should operate without crashes

Measurable:	Stability of the component should be tested through a set of use cases. Number of the test cases will be determined by the Test plan.
Relevant:	Realizes: LG 3, Win condition 4.1
Specific:	Stability is most critical within SSRD 3.2.1 (SR-2, SR-3, SR-5)

Table 11 L.O.S goals for Interface to import log data

3.1.7.2 Classifier: log analysis (SCC-03)

3.1.7.2.1 Purpose

To analyze the log data and generate/delete report.

3.1.7.2.2 Behavior

3.1.7.2.2.1 Processes

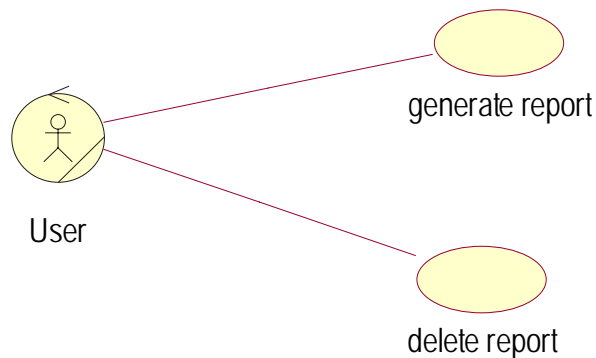


Figure 21 Use-Case diagram for Log analysis

3.1.7.2.2.1.1

Process: generate report

Identifier	Prc-2.1
Use-Case Name	Generate report.
Abstract	No
Purpose	Generate item relationship based on usage data
Actors	User

Importance	Primary
Capability	C-02 (OCD4.3)
Requirements	SSRD 3.2.1 (SR-3,4,5)
Risks	The correctness of the analyzing algorithm.
High-Risk?	Yes
Architecturally Significant?	Yes
Development Status	LCA
Overview	Parse the imported file, produce item relationship, graph clustering, centralized analysis, store result in a tree structure in to files.
USER INTERFACE	See OCD 5 - prototype
Pre-conditions	Log files imported
Post-conditions	Item relationship is generated.
Specializes	None
Includes	None
Extends	None
Extension Points	None

Table 12 Use case description for Generate report

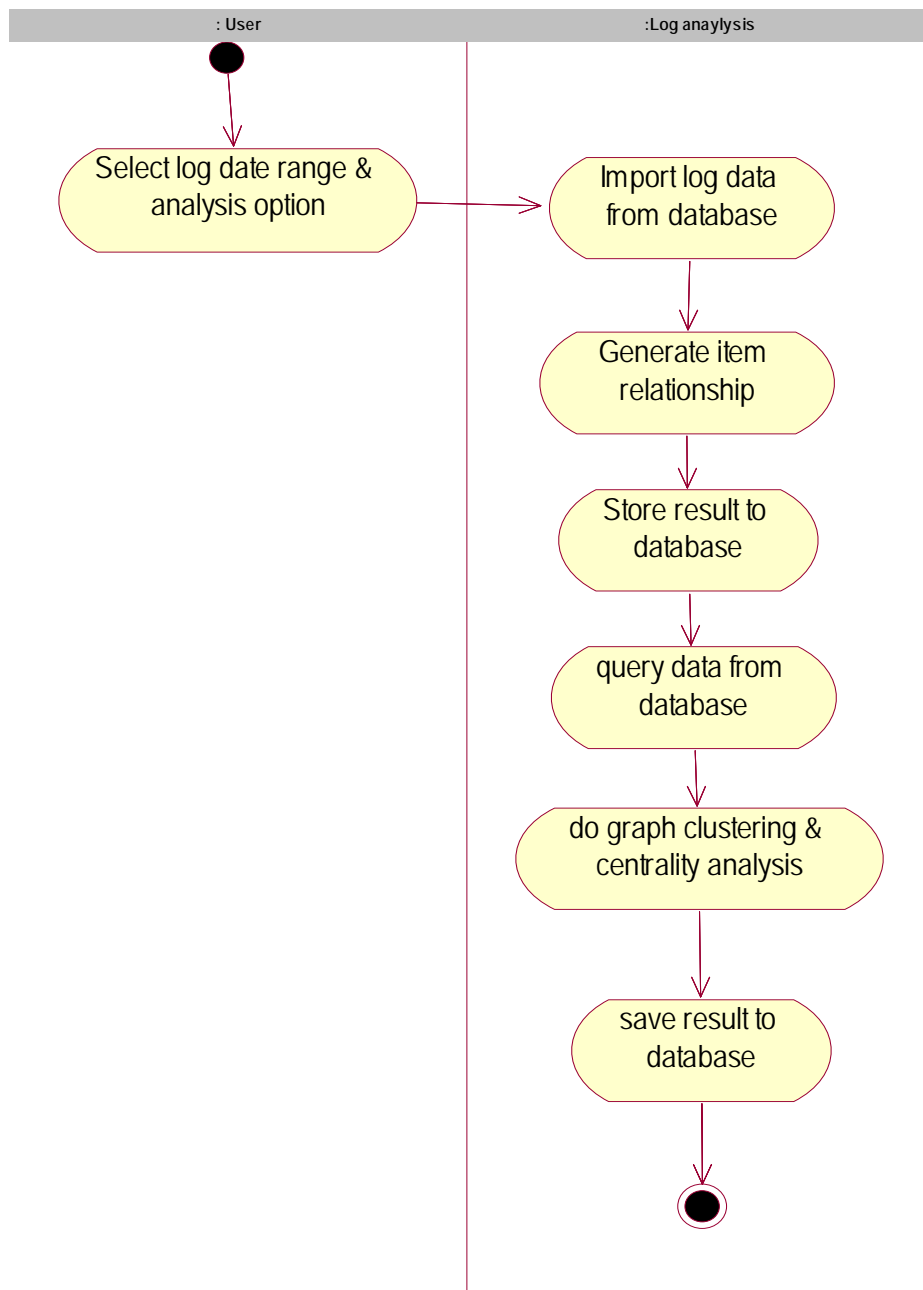


Figure 22 Activity diagram for generate report

3.1.7.2.2.1.2

Process: delete report

Identifier	Prc-2.2
Use-Case Name	Delete report
Abstract	No
Purpose	Delete report.
Actors	User
Importance	Secondary
Capability	C-02 (OCD4.3)
Requirements	SSRD 4.1.1(IR-4)
Risks	N/A
High-Risk?	No
Architecturally Significant?	No
Development Status	LCA
Overview	Delete the report that user has generated.
USER INTERFACE	See OCD 5 - prototype
Pre-conditions	Report existed.
Post-conditions	Report deleted.
Specializes	None
Includes	None
Extends	None
Extension Points	None

Table 13 Use case description for delete report

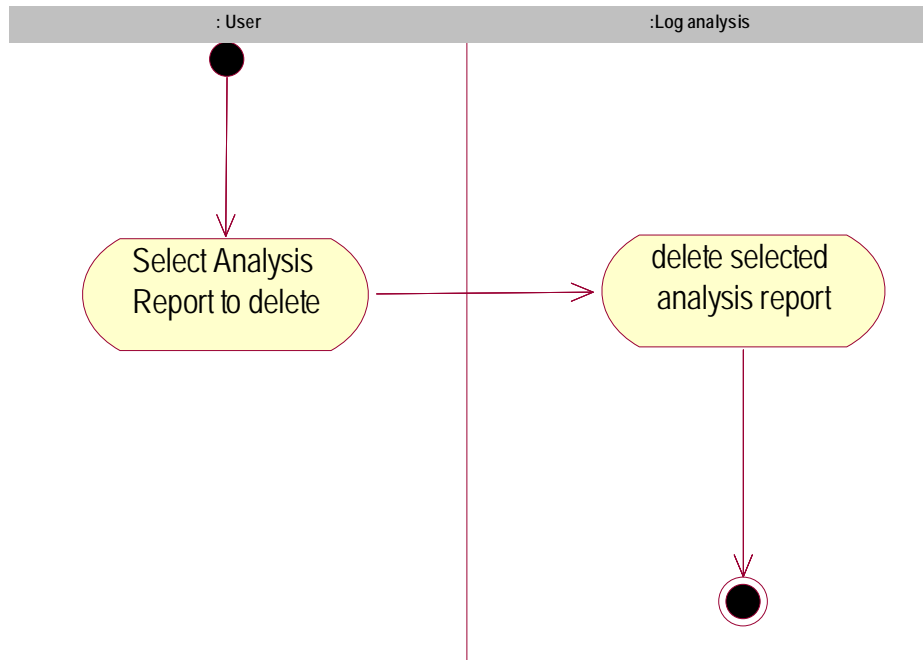


Figure 23 Activity diagram for delete report

3.1.7.2.3 L.O.S. Goals

L.O.S. Requirement:	Dependability [SSRD 5, LR-1]
Description:	This component should operate without crashes
Measurable:	Stability of the component should be tested through a set of use cases. Number of the test cases will be determined by the Test plan.
Relevant:	Realizes: LG 3, Win condition 4.1
Specific:	Stability is most critical within SSRD 3.2.1(SR-2, SR-3, SR-5)

L.O.S. Requirement:	Operability in multitasking environments [SSRD 5, LR-3]
Description:	This component should operate in multitasking environments

Measurable:	Usability of a host while performing a heavy system task can be tested experimentally, and measured by number of routine tasks user can perform per unit time.
Relevant:	Ensures usability of the system in standard user mode
Specific:	Expresses which programs should be able to run in parallel

Table 14 L.O.S goals for log analysis

3.1.7.3 Classifier: Visualization of log analysis result (SCC-04)

3.1.7.3.1 Purpose

To display the graphical representation of the analysis result and allow the user to browse it by selecting the nodes.

3.1.7.3.2 Behavior

3.1.7.3.2.1 Processes



Figure 24 Use-Case diagram for visualization of log analysis result

3.1.7.3.2.1.1

Process: browse graph

Identifier	Prc-3.1
Use-Case Name	Browse graph
Abstract	No
Purpose	Let user browse the graph by selecting nodes.
Actors	User

Importance	Primary
Capability	C-03 (OCD4.3)
Requirements	SSRD 3.2.1 (SR-6), 4.1.1(IR-5)
Risks	This use case is implemented by open source product.
High-Risk?	Yes
Architecturally Significant?	No.
Development Status	LCA
Overview	Given the input file, produce graph nodes.
USER INTERFACE	See OCD 5 - prototype
Pre-conditions	Analysis result produced by 'log analysis' exits.
Post-conditions	Display graph.
Specializes	None
Includes	None
Extends	None
Extension Points	None

Table 15 Use case description for **Browse graph**

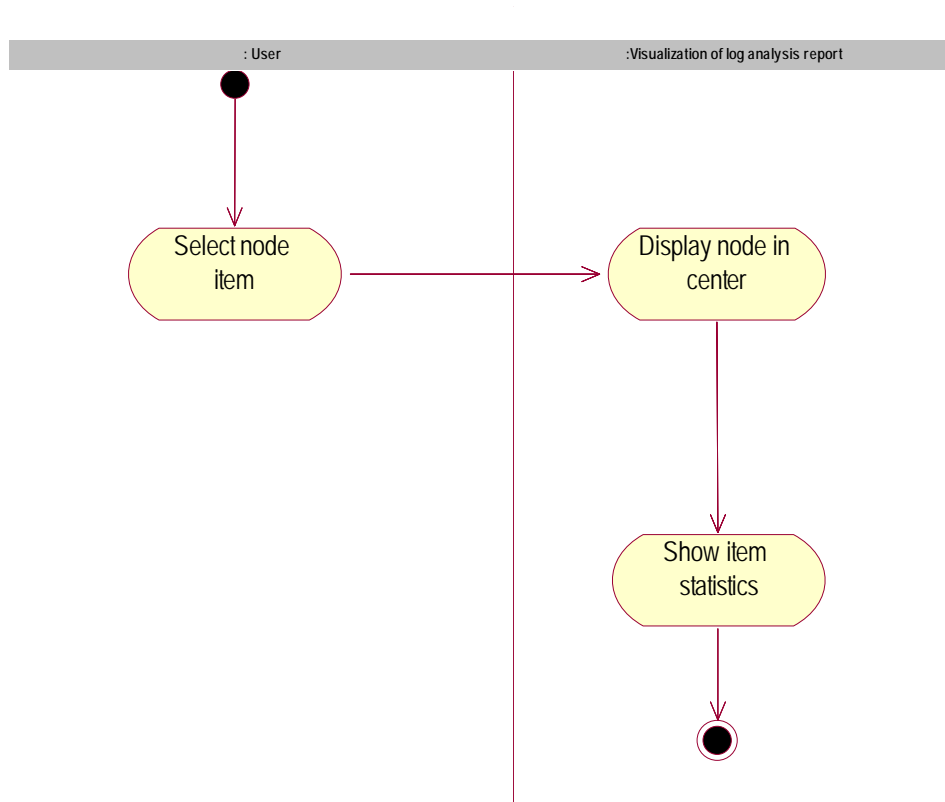


Figure 25: Activity diagram for Browse graph

3.1.7.3.3 L.O.S. Goals

L.O.S. Requirement:	Dependability [SSRD 5, LR-1]
Description:	This component should operate without crashes
Measurable:	Stability of the component should be tested through a set of use cases. Number of the test cases will be determined by the Test plan.
Relevant:	Realizes: LG 3, Win condition 4.1
Specific:	Stability is most critical within SSRD 3.2.1 (SR-2, SR-3, SR-5)

Level of Service:	Usability [SSRD 5, LR-2]
Description:	Should be user-friendly, readable and easy to navigate.
Measurable:	Usability of the system can be tested experimentally, and measured by number of tasks first-time user can accomplish per unit time.
Relevant:	Realizes: LG-4, LG-5, Determines usability of the system
Specific:	Realizes: Win condition 3.3.

Level of Service:	Performance on data of current scale [SSRD 5, LR-4]
Description:	The component should be able to scale as the collection grows.
Measurable:	The component should be able to generate, cluster and visualize for graphs with desired number of nodes.
Relevant:	Defines performance requirement of the proposed system in terms of scale of input data
Specific:	Provides specific scale requirements.

Table 16 L.O.S goals for Visualization of analysis result

3.1.7.4 Classifier: Database (SCC-02)

3.1.7.4.1 Purpose

‘Database’ is to store the log data imported/pared from user’s local directory. This component also provide the data for ‘Log Analysis’ component and store the results of the analysis.

3.1.7.4.2 Behavior

3.1.7.4.2.1 Processes

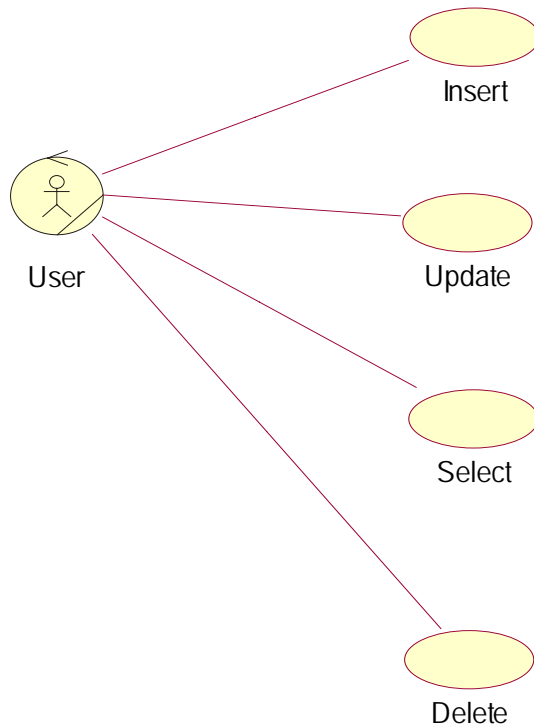


Figure 26 Use-Case diagram for Database

There is no use case description and activity diagram for this component classifier, because it is an open source component. It’s out of the project scope to describe it.

3.1.7.4.3 L.O.S. Goals

L.O.S. Requirement:	Dependability [SSRD 5, LR-1]
Description:	This component should operate without crashes
Measurable:	Stability of the component should be tested through a set of use cases. Number of the test cases will be determined by the Test plan.
Relevant:	Realizes: LG 3, Win condition 4.1
Specific:	Stability is most critical within SSRD 3.2.1(SR-2, SR-3, SR-5)

Table 17 L.O.S goals for Database

3.1.8 Software Connector Classifiers

N/A. because the size (small) of our project, this is not applicable here.

3.1.9 Hardware Components

The hardware components identified here is computer. Computer will be used to run the analysis program. The system will first get the raw log data from the local directory which are downloaded by user to their local directory, after the analysis, the result will be stored again.

3.1.9.1 Computer

3.1.9.1.1 Purpose

The purpose of computer is to contain four software components: Interface to import log data, Database, Log analysis, and Visualization of analysis result.

3.1.9.1.2 Classifier

This hardware component is an instance of the classifier computer HCC-01

3.1.9.1.3 L.O.S.

Hardware component classifier HCC-01 applies here.

3.1.10 Hardware Connectors

N/A, because this is a stand-alone system.

.

3.1.11 Software Components

This section describes the kinds of software components that are part of the system, the actors for the system with which the system interacts, and the kinds of connectors that will be used to connect them.

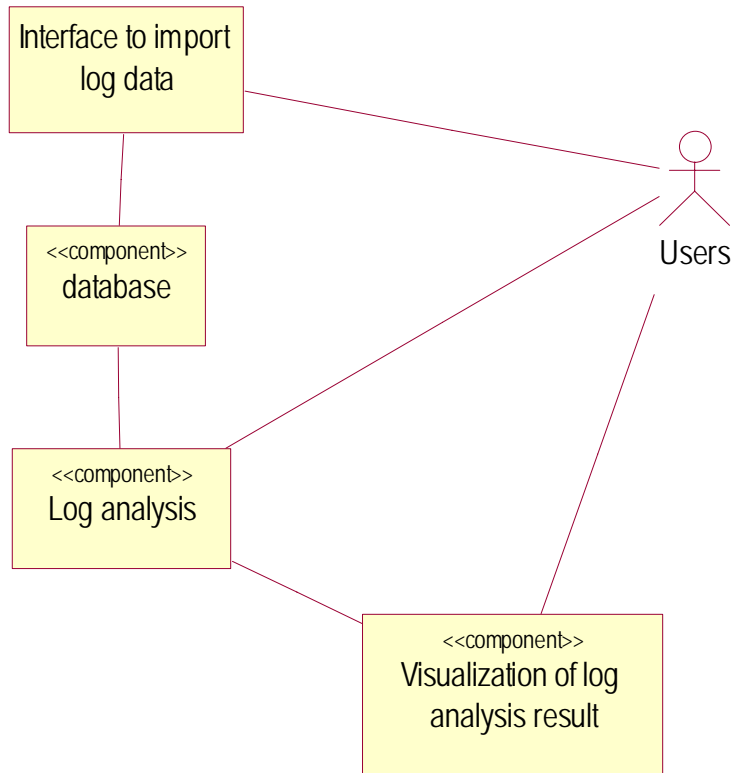


Figure 27 Software component model

3.1.11.1 Interface to import log data (SCO-01)

3.1.11.1.1 Purpose

To import log data from local directory and delete the imported log data.

3.1.11.1.2 Classifier

This software component is an instance of the classifier SCC-01.

3.1.11.1.3 L.O.S. Goals

Software component classifier SCC-01 applies here.

3.1.11.2 Database (SCO-02)

3.1.11.2.1 Purpose

'Database' is to store the log data imported from the local file directory. This component also provide the data for 'Log Analysis' component and store the results of the analysis.

3.1.11.2.2 Classifier

This software component is an instance of the classifier SCC-02.

3.1.11.2.3 L.O.S. Goals

Software component classifier SCC-02 applies here.

3.1.11.3 Log analysis (SCO-03)

3.1.11.3.1 Purpose

To generate/delete report.

3.1.11.3.2 Classifier

This software component is an instance of the classifier SCC-03.

3.1.11.3.3 L.O.S. Goals

Software component classifier SCC-03 applies here.

3.1.11.4 Visualization of log analysis result (SCO-04)

3.1.11.4.1 Purpose

To show the user the graphical representation of the analysis result.

3.1.11.4.2 Classifier

This software component is an instance of the classifier SCC-04.

3.1.11.4.3 L.O.S. Goals

Software component classifier SCC-04 applies here.

3.1.12 Software Connectors

N/A. because the size(small) of our project, this is not applicable here.

3.2 Analysis Classes

This section will describe the information classes that support architectural structure and implement the system behavior.



Figure 28 Class Model: Analysis Classes

3.2.1 Log data

The USC-ISD has maintained a log file of usage statistic of the USC digital archive. When users visit the Digital archive web site and retrieve certain images, their activities will be logged, the information such as user's IP, item id and time of retrieval are recorded in the log file. These log files will then be imported by user to their local directory to be analyzed by the system to produce the usage analysis report. System Capability C-01 [see OCD 4.3] uses this artifact.

3.2.2 Automated Usage Analysis Report

By importing the log file from user’s local machine, usage analysis report is produced automatically based on the user’s selection of date ranges, the user may want to see the log analysis within a certain period of time. The analysis report will include information such as: report name, report start and end date, reinforcement function chosen, and the graph. System Capability C-02 [see OCD 4.3] produce this artifact.

3.3 Behavior

This section will give more detailed description about how the components in section 3.1.4. work together to implement the processes in section 2.3.

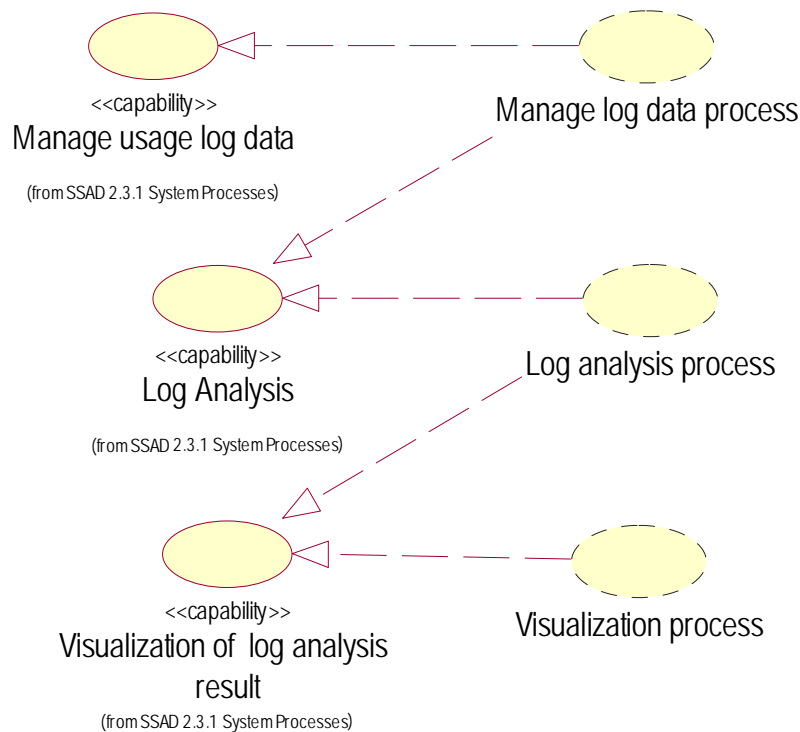


Figure 29 Behavior Model Use-Case Diagram

3.3.1.1 Process realization: Manage log data process

Identifier	Prc-1
-------------------	-------

Use-Case Name	Manage log data process
Abstract	No
Purpose	To import, view & delete log data
Actors	User
Importance	Primary
Capability	C-01 (OCD 4.3)
Requirements	SSRD. 3.2.1 (SR-1, SR-2) 4.1.1(.IR-1)
Risks	None
High-Risk?	No
Architecturally Significant?	No
Development Status	LCA
Overview	To import log data stored in user's local machine to be analyzed. To delete the log data based on user's selection.
USER INTERFACE	See OCD 5 - prototype
Pre-conditions	Log file exists.
Post-conditions	Log file imported/deleted.
Specializes	None
Includes	None
Extends	None
Extension Points	None

Table 18 Use case description for Manage log data process

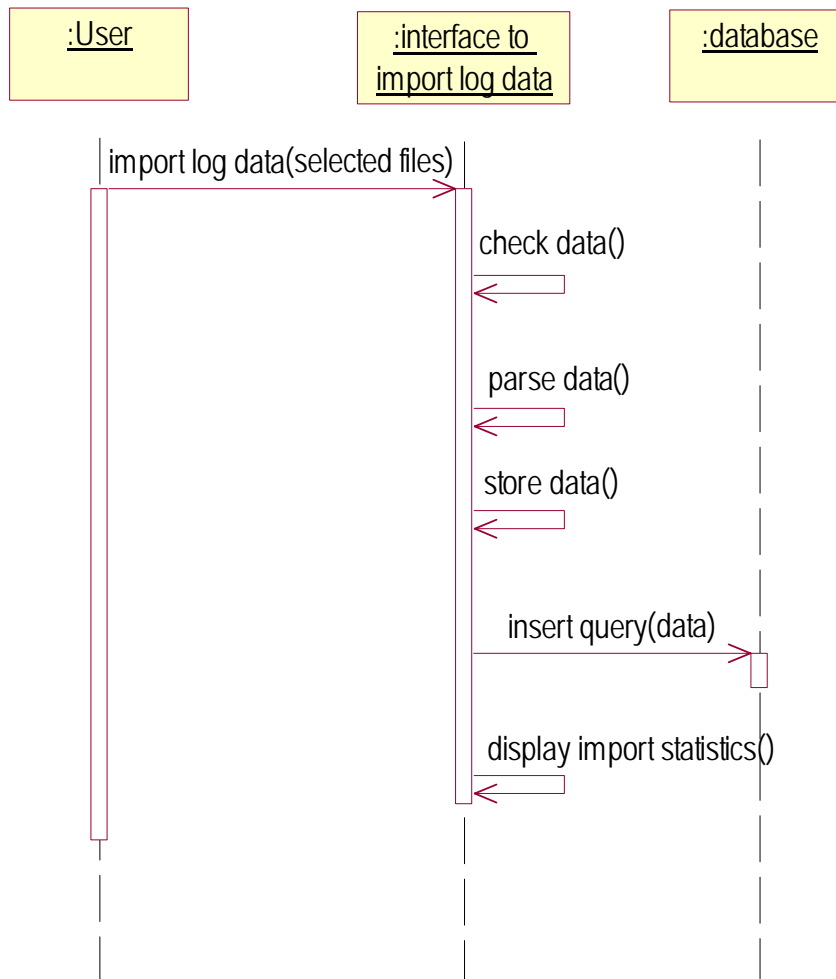


Figure 30 Sequence diagram for Manage log data process-import data

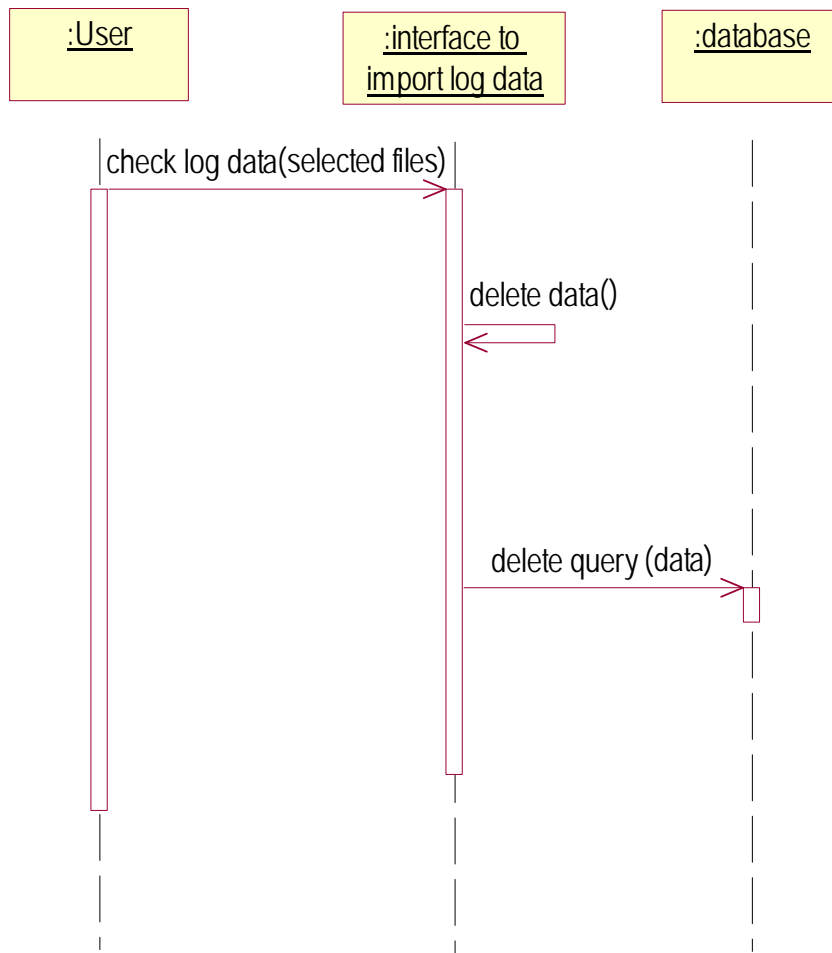


Figure 31 Sequence diagram for Manage log data process-delete

3.3.1.2 Process realization: Log analysis process

Identifier	Prc-2
Use-Case Name	Log analysis process
Abstract	No
Purpose	Generate and delete analysis report.
Actors	User
Importance	Primary
Capability	C-02 (OCD4.3)
Requirements	SSRD 3.2.1(SR3,4,5)
Risks	The correctness of the analyzing algorithm.
High-Risk?	Yes

Architecturally Significant?	No
Development Status	LCA
Overview	Parse the imported file, produce item relationship, graph clustering, centralized analysis, store result in a tree structure in to files.
USER INTERFACE	See OCD 5 - prototype
Pre-conditions	Log files imported
Post-conditions	Analysis report generated/deleted.
Specializes	None
Includes	None
Extends	None
Extension Points	None

Table 19 Use case description for Log analysis process

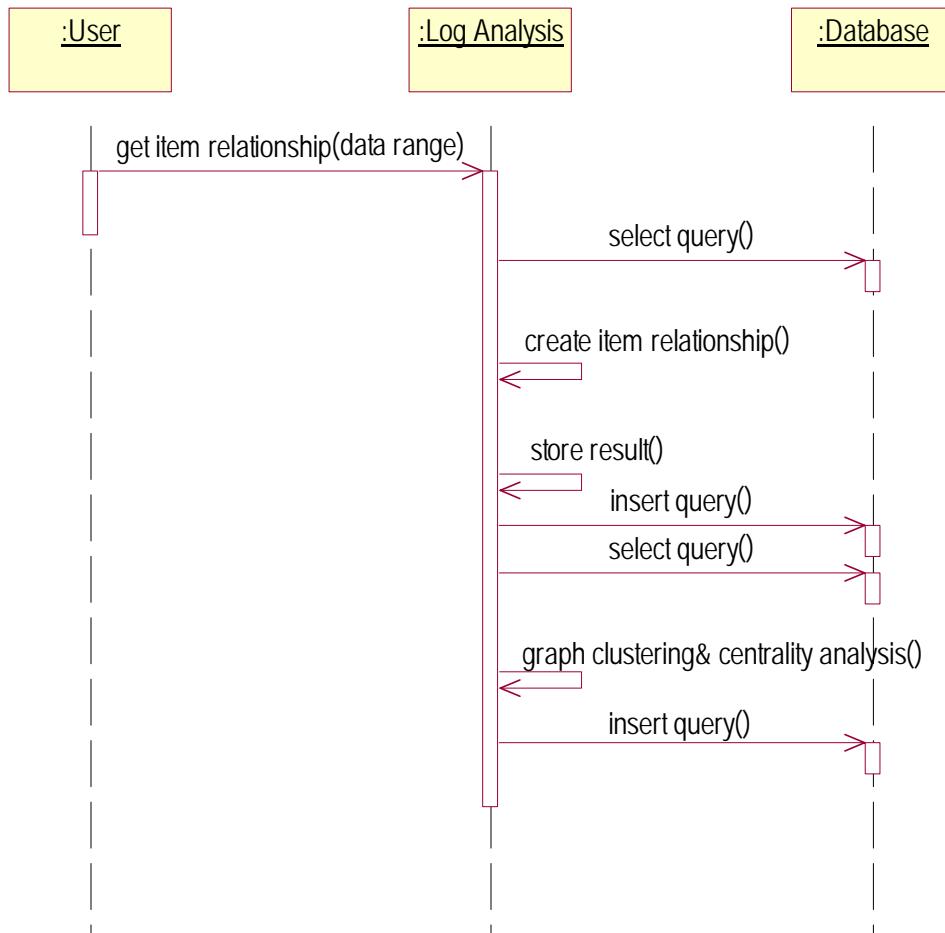


Figure 32 Sequence diagram for Log analysis process -generate report

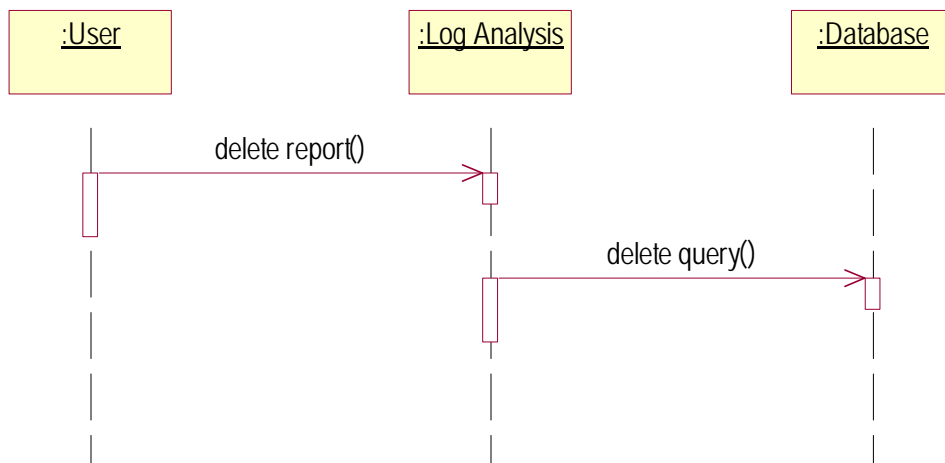


Figure 33 Sequence diagram for log analysis process-delete report

3.3.1.3 Process realization: Visualization process

Identifier	Prc-3
Use-Case Name	Visualization process
Abstract	No
Purpose	Let user brows the graph by selecting nodes.
Actors	User
Importance	Primary
Capability	C-03 (OCD4.3)
Requirements	SSRD 4.1.1 (IR-5), 3.2.1 (SR-6)
Risks	This use case is implemented by open source product.
High-Risk?	Yes
Architecturally Significant?	No
Development Status	LCA
Overview	Given the input file, produce graph nodes. And allow user to browse it by selecting nodes.
USER INTERFACE	See OCD 5 - prototype
Pre-conditions	Analysis data produced by 'log analysis process' exists.
Post-conditions	Display graph.
Specializes	None
Includes	None
Extends	None
Extension Points	None

Table 20 Use case description for Visualization process

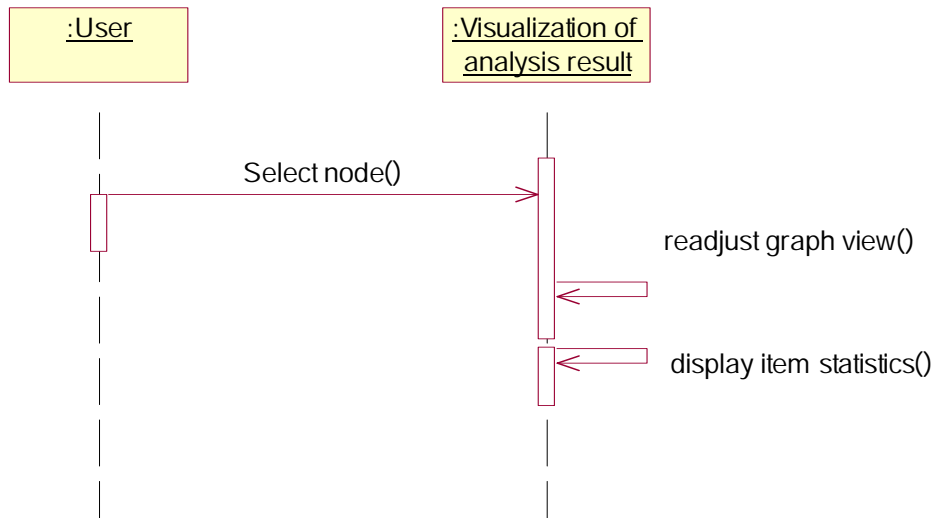


Figure 34 Sequence diagram for visualization process

3.4 L.O.S Projected

This section describes the elements of the architecture to which each system L.O.S. applies and the projected value of each system L.O.S.

L.O.S Goal	Applies To	How	Projected Value	Evaluation Technique
LR-1 [SSRD 5]: Dependability	SCO-01, SCO -02, SCO -03, SCO -04	Equally	Pass 95% of test cases	Estimation
LR-2 [SSRD 5]: Usability	SCO -04	Equally	Satisfied with the client	Estimation

LR-3 [SSRD 5]: Operability in multitasking environment	SCO -03	Equally	Pass 95% of test cases	Estimation
LR-4[SSRD 5]: Performance on data of current scale	SCO -04	Equally	Able to perform properly with 10% increase.	Estimation

Table 21 L.O.S projected

3.5 Architectural Styles, Patterns & Frameworks

Our system's architecture is identified as multi-layer pattern architecture. We will give more information in the following table.

Name	Description	Benefits, Cots & limitations
Multi-layer Pattern	Multi-Layer system is a kind of software system and uses multi-layer architecture pattern. See http://www.site.uottawa.ca:4321/oose/index.html#multi-layer_table	It replaces a layer by an improved version, or by one of different capabilities. In addition, it also increases reusability. It divides and conquers since the separate layers can be independently designed.

Table 22 Architectural Styles, Patterns & Frameworks

4. Implementation Design

In this section we will design a technology-specific implementation for the system by refining the general architecture defined during Architecture Design & Analysis (SSAD 3).

4.1 Structure

4.1.1 Topology

Here we describe how the components and component classes of the system are organized. The typology used here in implementation is the same as the typology in architecture design (SSAD 3.1.1). We have three layers: Interface, Process and Database.

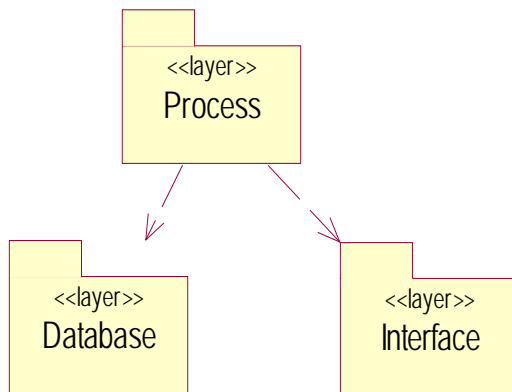


Figure 35 Topology

4.1.1.1 Interface Layer

The following shows the components contain in the interface layer:

- Interface to import log data (hand-coded)
- Visualization of analysis result (Open source- H3/viewer, Open Motif,)

4.1.1.2 Process Layer

The following shows the components contained in the process layer:

- Log Analysis (hand-coded & open source graph algorithm-Bollen & Markov)

4.1.1.3 Database Layer

The following lists the components contained in the database layer:

- SQLite database (Open source-SQLite).

4.1.2 Hardware Classifier Model

This section describes the kinds of hardware components that are part of the system, the actors for the system with which the components interact, and the kinds of connectors that will be used to connect them.

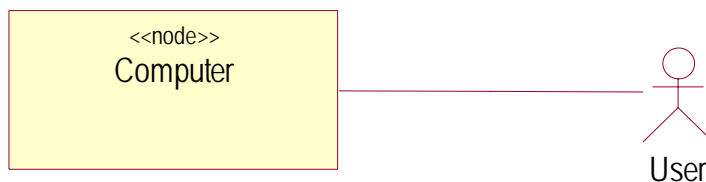


Figure 36 Hardware classifier model

4.1.3 Software Classifier Model

This section describes the kinds of software components that are part of the system, the actors for the system with which the system interacts. The model shown here is the refinement of software classifier model shown in architecture design (SSAD 3.1.3) by describing how each component is implemented, i.e. hand-coded, COTS or open source.

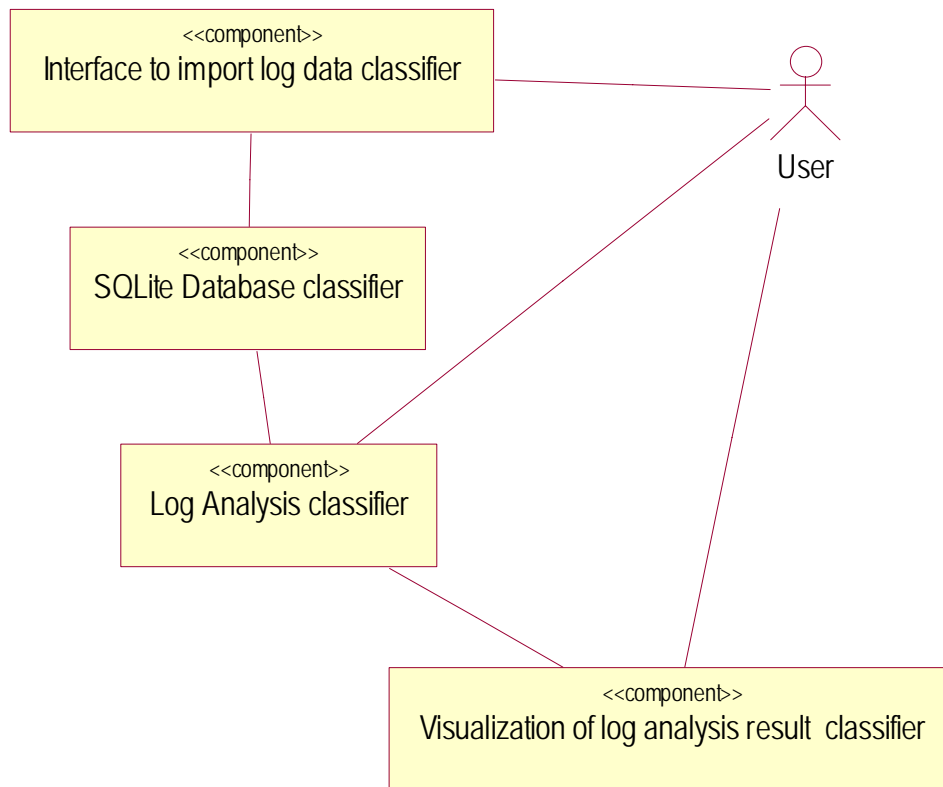


Figure 37 Software classifier model

4.1.4 Deployment Model

Deployment model describes how software & hardware components work together to make a working version of the system. The model shown here is the refinement of deployment model shown in architecture design (SSAD 3.1.4) by describing how each component is implemented, i.e. hand-coded, COTS or open source.

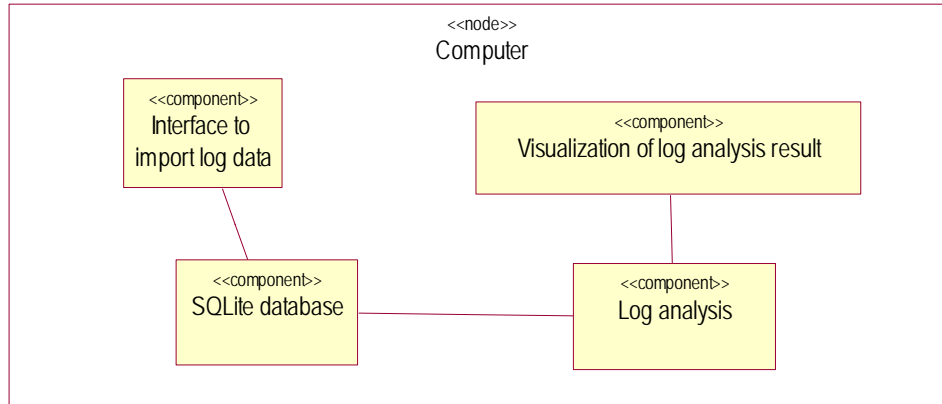


Figure 38 Deployment model

4.1.5 Hardware Component Classifiers

4.1.5.1 Classifier: Computer (HCC-01)

4.1.5.1.1 Purpose

The purpose of computer is to contain two software components. The component of this classifier will be implemented by Unix, PC or MAC computer.

4.1.5.1.2 L.O.S. Goals

Same as 3.1.5.1.2.

4.1.6 Hardware Connector Classifiers

Same as 3.1.6.

4.1.7 Software Component Classifiers

4.1.7.1 Classifier: Interface to import log data (SCC-01)

4.1.7.1.1 Purpose

To import log data from local file directory and to delete the log data. The component of this classifier will be hand-coded by using C++. The component will also be designed in a way to cater for future evolution requirement in SSRD 6.2 ER-2.(Interface with other usage log formats)

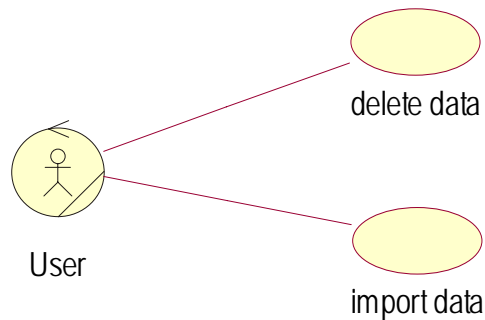
4.1.7.1.2 Behavior**4.1.7.1.2.1 Processes**

Figure 39 Use-Case diagram for Interface to import log data

4.1.7.1.2.1.1**Process: delete data**

Identifier	Prc-1.1
Use-Case Name	Delete data
Abstract	No
Purpose	To delete log data
Actors	User
Importance	secondary
Capability	C-01 (OCD 4.3)
Requirements	SSRD. 3.2.1 (SR-2)
Risks	None
High-Risk?	No
Architecturally Significant?	No
Development Status	LCA
Overview	To delete log data that user has previously imported.
USER INTERFACE	See OCD 5 - prototype
Pre-conditions	Log data exists.
Post-conditions	Log data deleted.
Specializes	None
Includes	None
Extends	None

Extension Points	None
-------------------------	------

Table 23 Use case description for Delete data

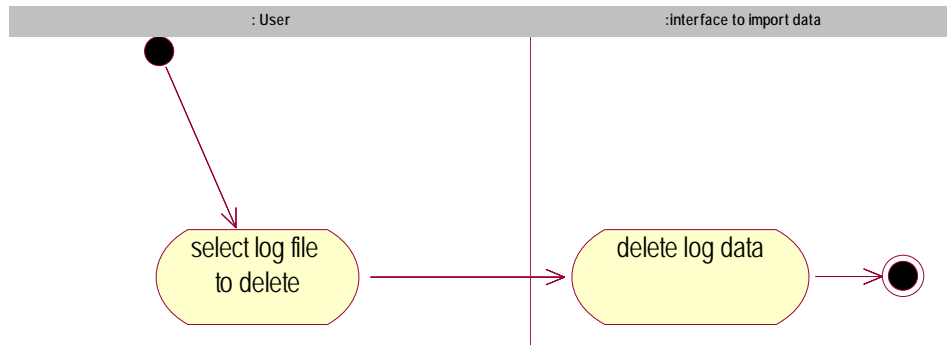


Figure 40 Activity diagram for Delete data

4.1.7.1.2.1.2

Process: import data

Identifier	Prc-1.2
Use-Case Name	Import data.
Abstract	No
Purpose	To import log data
Actors	User
Importance	secondary
Capability	C-01 (OCD 4.3)
Requirements	SSRD 3.2.1 (SR-1), 3.2.2(SR-7,8), 4.1.1(IR-1)
Risks	None
High-Risk?	No
Architecturally Significant?	No
Development Status	LCA
Overview	To import log data stored in user's local machine to be analyzed.
USER INTERFACE	See OCD 5 - prototype
Pre-conditions	Log file exists.
Post-conditions	Log file imported.

Specializes	None
Includes	None
Extends	None
Extension Points	None

Table 24 Use case description for Import data

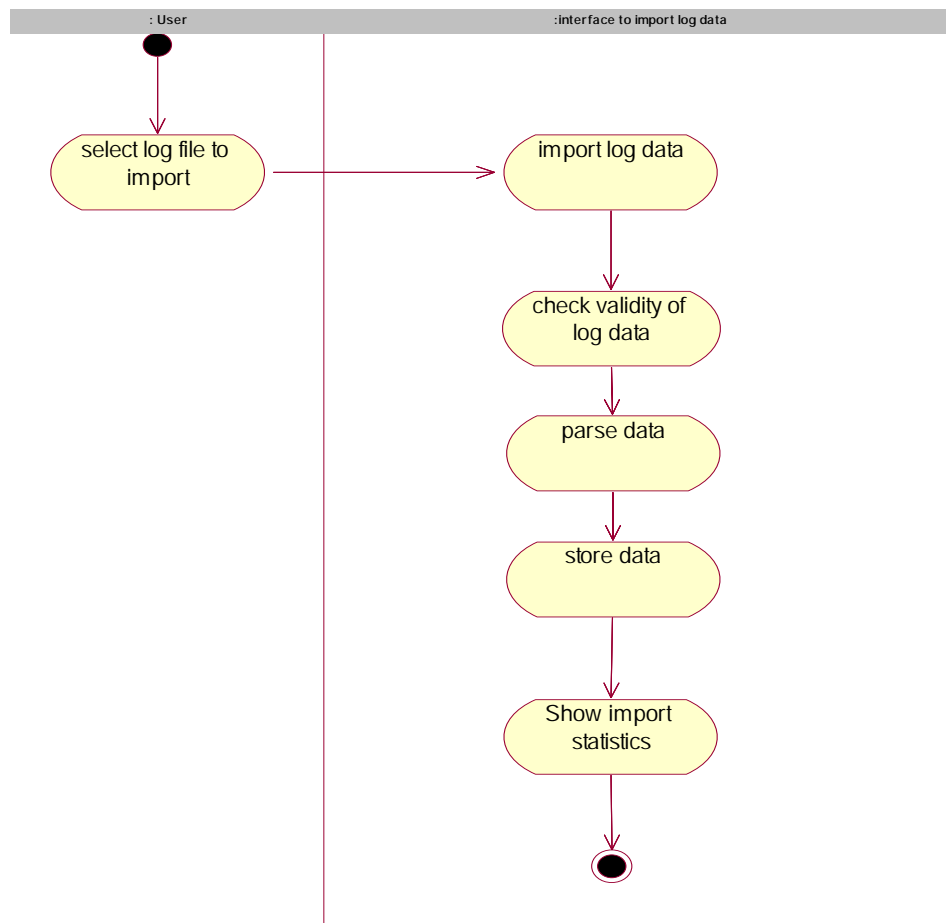


Figure 41 Activity diagram for Import data

4.1.7.1.3 L.O.S Goals

L.O.S. Requirement:	Dependability [SSRD 5, LR-1]
Description:	This component should operate without crashes
Measurable:	Stability of the component should be tested through a set of use cases. Number of the test cases will be determined by the Test plan.
Relevant:	Realizes: LG 3, Win condition 4.1
Specific:	Stability is most critical within SSRD 3.2.1 (SR-2, SR-3, SR-5)

Table 25 L.O.S goal for Interface to import log data

4.1.7.1.4 Internal Architecture



Figure 42 Internal Architecture for Interface to import log data

4.1.7.2 Classifier: SQLite Database (SCC-02)

4.1.7.2.1 Purpose

‘SQLite Database’ is to store the log data imported from the local file directory. This component also provide the data for ‘Log Analysis’ component and store the results of the analysis The component of this classifier will be implemented by using open source software, SQLite.

4.1.7.2.1.1 Processes

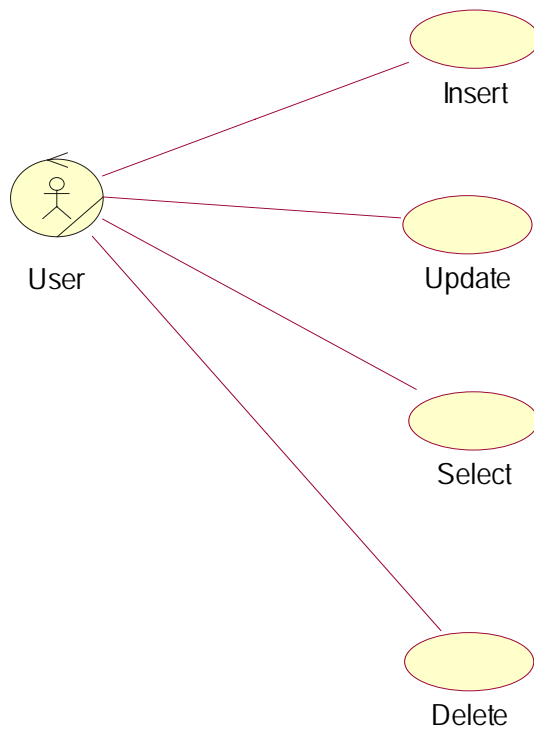


Figure 43 Use-Case diagram for SQLite Database

There is no use case description and activity diagram for this component classifier, because it is an open source component. It's is out of project scope to describe it.

4.1.7.2.2 L.O.S. Goals

L.O.S. Requirement:	Dependability [SSRD 5, LR-1]
Description:	This component should operate without crashes
Measurable:	Stability of the component should be tested through a set of use cases. Number of the test cases will be determined by the Test plan.
Relevant:	Realizes: LG 3, Win condition 4.1
Specific:	Stability is most critical within SSRD 3.2.1 (SR-2, SR-3, SR-5)

Table 26 L.O.S. Goals for SQLite database

4.1.7.2.3 Internal Architecture

It is out of project scope and complex to define open source architecture.

4.1.7.3 Classifier: log analysis (SCC-03)

4.1.7.3.1 Purpose

To analyze the log data and produce item relationship. The component of this classifier will be hand-coded by using C++ and open source algorithm.(Bollen, Markov) The component will also be designed in a way to allow the future evolution requirement in SSRD 6.1 ER-1.(Advanced analysis)

4.1.7.3.2 Behavior

4.1.7.3.2.1 Processes

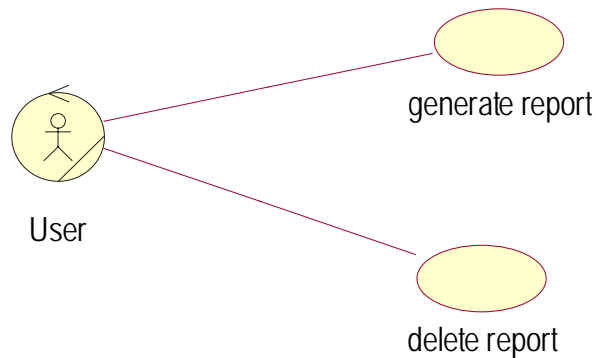


Figure 44 Use-Case diagram for log analysis

4.1.7.3.2.1.1

Process: generate report

Identifier	Prc-2.1
Use-Case Name	Generate report.
Abstract	No
Purpose	Generate item relationship based on usage data
Actors	User

Importance	Primary
Capability	C-02 (OCD4.3)
Requirements	SSRD 3.2.1 (SR-3,4,5) 3.2.2 (SR-9), 4.1.1(IR-2,3,4)
Risks	The correctness of the analyzing algorithm.
High-Risk?	Yes
Architecturally Significant?	Yes
Development Status	LCA
Overview	Parse the imported file, produce item relationship, graph clustering, centralized analysis, store result in a tree structure in to files. Analysis report should not be created and database should not be changed due to user request or some system / algorithm error. System will have the functionality to allow user to stop the report generation.
USER INTERFACE	See OCD 5 - prototype
Pre-conditions	Log files imported
Post-conditions	Item relationship is generated.
Specializes	None
Includes	None
Extends	None
Extension Points	None

Table 27 Use case description for Generate report

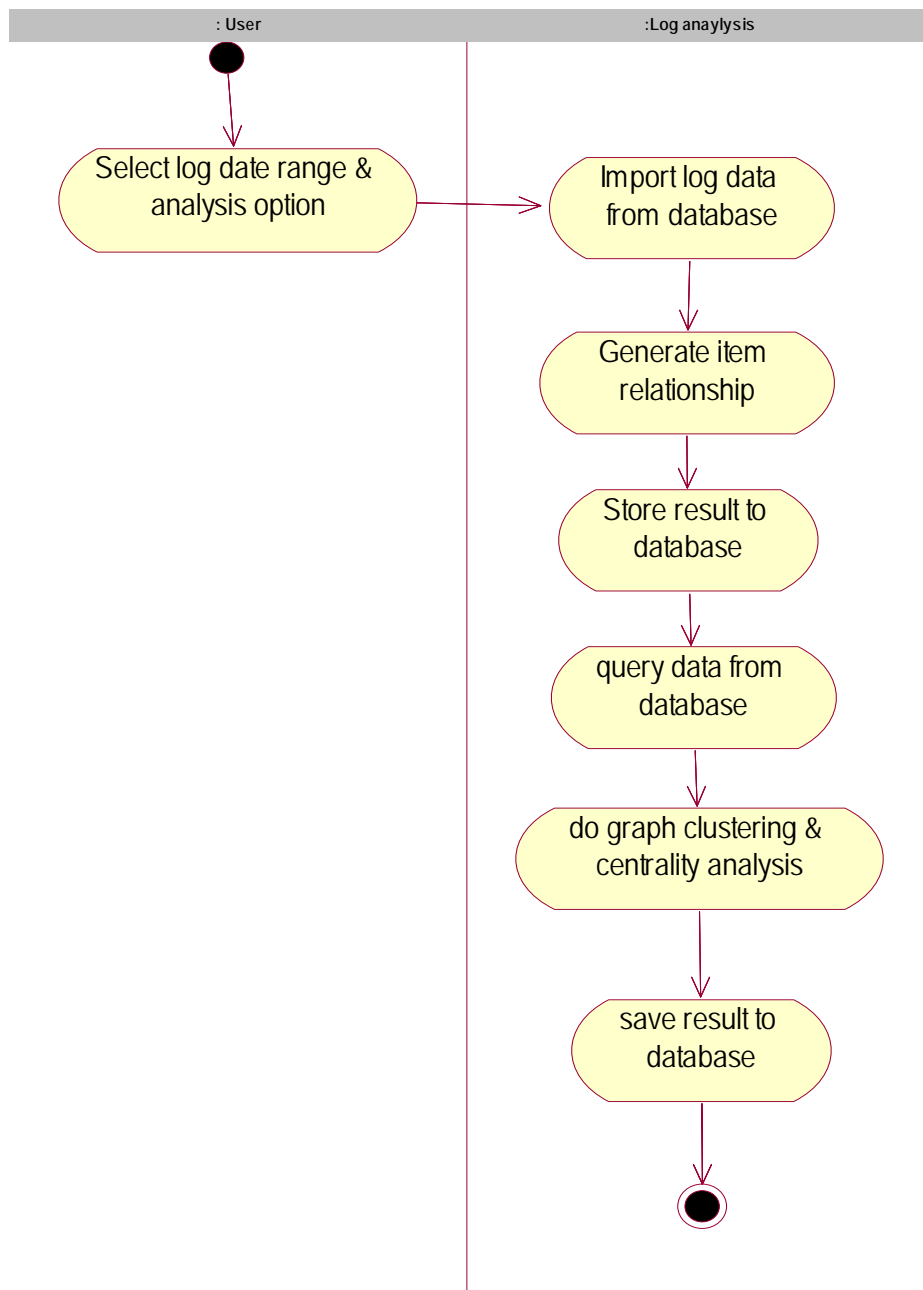


Figure 45 Activity diagram for Generate report

4.1.7.3.2.1.2

Process: delete report

Identifier	Prc-2.2
Use-Case Name	Delete report
Abstract	No
Purpose	Delete report.
Actors	User
Importance	Secondary
Capability	C-02 (OCD4.3)
Requirements	SSRD 4.1.1(IR-4)
Risks	N/A
High-Risk?	No
Architecturally Significant?	No
Development Status	LCA
Overview	Delete the report that user has generated.
USER INTERFACE	See OCD 5 - prototype
Pre-conditions	Report existed.
Post-conditions	Report deleted.
Specializes	None
Includes	None
Extends	None
Extension Points	None

Table 28 Use case description for Delete report

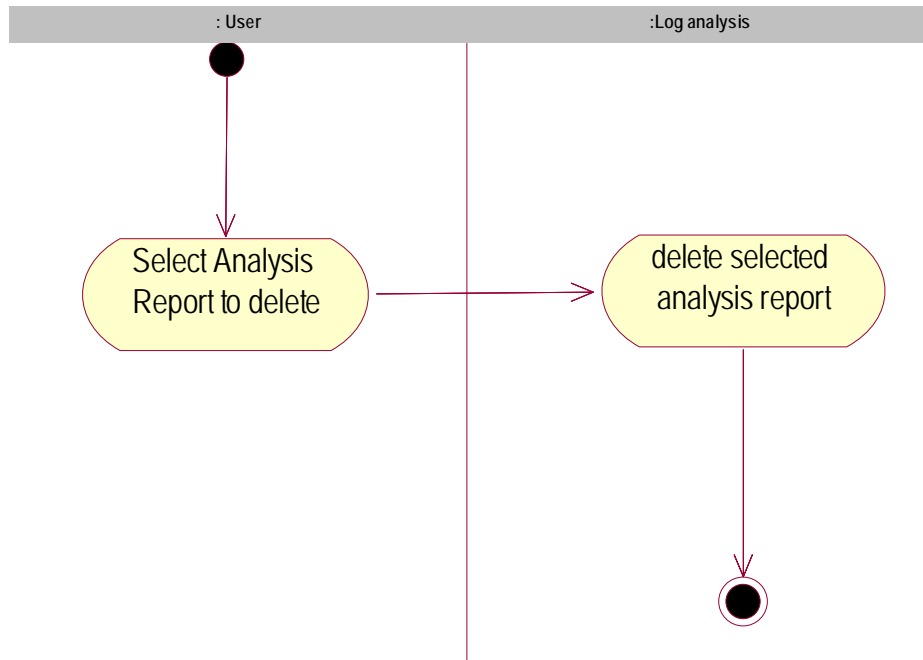


Figure 46 Activity diagram for Delete report

4.1.7.3.3 L.O.S. Goals

L.O.S. Requirement:	Dependability [SSRD 5, LR-1]
Description:	This component should operate without crashes
Measurable:	Stability of the component should be tested through a set of use cases. Number of the test cases will be determined by the Test plan.
Relevant:	Realizes: LG 3, Win condition 4.1
Specific:	Stability is most critical within SSRD 3.2.1 (SR-2, SR-3, SR-5)

L.O.S. Requirement:	Operability in multitasking environments [SSRD 5, LR-3]
----------------------------	---

Description:	This component should operate in multitasking environments
Measurable:	Usability of a host while performing a heavy system task can be tested experimentally, and measured by number of routine tasks user can perform per unit time.
Relevant:	Ensures usability of the system in standard user mode
Specific:	Expresses which programs should be able to run in parallel

Table 29L.O.S goal for Log analysis

4.1.7.3.4 Internal Architecture

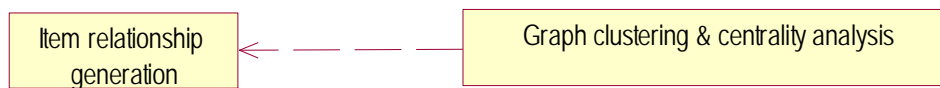


Figure 47 : Internal Architecture: Log analysis

4.1.7.4 Classifier: Visualization of analysis result (SCC-04)

4.1.7.4.1 Purpose

To show the user the graphical representation of the analysis result. The component of this classifier will be implemented by using open-source software (H3 viewer, Open Motif). The component will also be designed in a way to allow the future evolution requirement in SSRD 6.4 ER-3. (Accommodating collection growth) and 6.5 ER-4(Workload growth should affect system performance in linear proportion).

4.1.7.4.2 Behavior**4.1.7.4.2.1 Processes**

Figure 48 Use-Case diagram for Visualization of log analysis result

4.1.7.4.2.1.1**Process: browse graph**

Identifier	Prc-3.1
Use-Case Name	Browse graph
Abstract	No
Purpose	Let user browse the graph by selecting nodes.
Actors	User
Importance	Primary
Capability	C-03 (OCD4.3)
Requirements	SSRD 3.2.1 (SR-6), 3.2.2(SR-10) 4.1.1(IR-5)
Risks	This use case is implemented by open source product.
High-Risk?	Yes
Architecturally Significant?	No.
Development Status	LCA
Overview	Given the input file, produce graph nodes. If in the analysis results of the chosen report are of corrupted format then visualization module should stop visualization action and should provide user with corresponding error message.
USER INTERFACE	See OCD 5 - prototype

Pre-conditions	Analysis result produced by 'log analysis' exits.
Post-conditions	Display graph.
Specializes	None
Includes	None
Extends	None
Extension Points	None

Table 30 Use case description for Browse graph

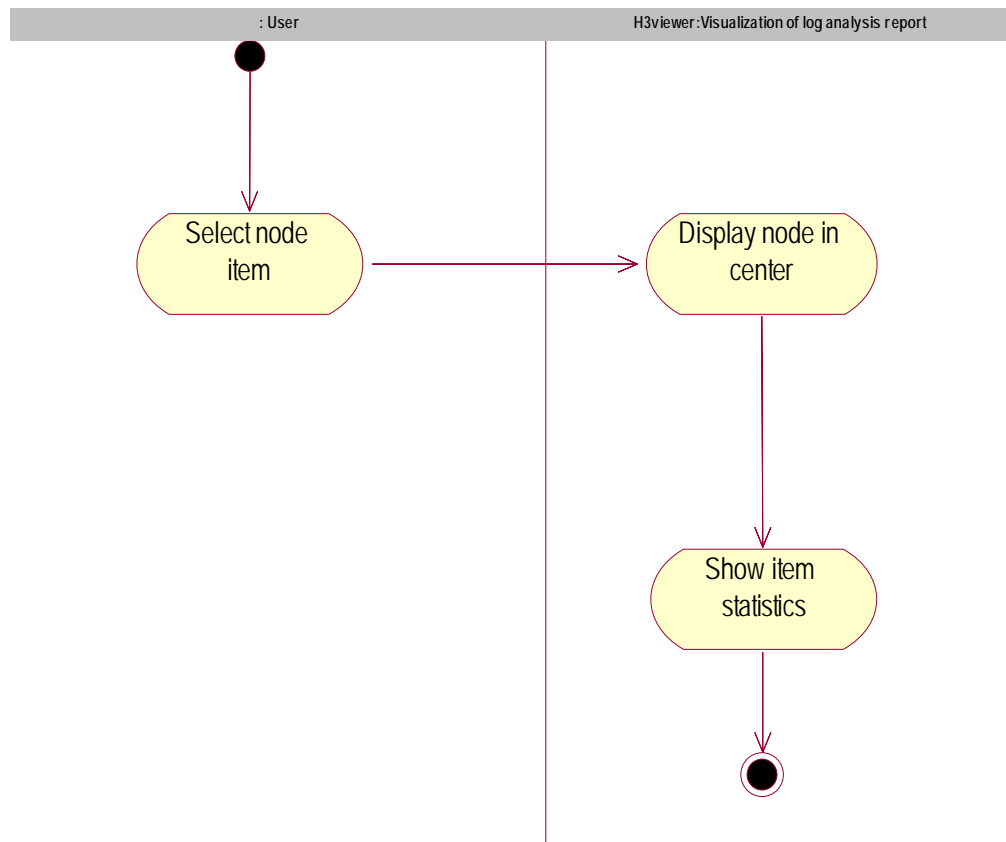


Figure 49: Activity diagram for Browse graph

4.1.7.4.3 L.O.S. Goals

L.O.S. Requirement:	Dependability [SSRD 5, LR-1]
Description:	This component should operate without crashes
Measurable:	Stability of the component should be tested through a set of use cases. Number of the test cases will be determined by the Test plan.
Relevant:	Realizes: LG 3, Win condition 4.1
Specific:	Stability is most critical within SSRD 3.2.1 (SR-2, SR-3, SR-5)

Level of Service:	Usability [SSRD 5, LR-2]
Description:	Should be user-friendly, readable and easy to navigate.
Measurable:	Usability of the system can be tested experimentally, and measured by number of tasks first-time user can accomplish per unit time.
Relevant:	Realizes: LG-4, LG-5, Determines usability of the system
Specific:	Realizes: Win condition 3.3.

Level of Service:	Performance on data of current scale [SSRD 5, LR-4]
Description:	The component should be able to scale as the collection grows.
Measurable:	The component should be able to generate, cluster and visualize for graphs with desired number of nodes.
Relevant:	Defines performance requirement of the proposed system in terms of scale of input data
Specific:	Provides specific scale requirements.

Table 31L.O.S goals for Visualization of analysis result

4.1.7.4.4 Internal Architecture

It is out of project scope and complex to define open source architecture.

4.1.8 Software Connector Classifiers

N/A. because the size (small) of our project, this is not applicable here.

4.1.9 Hardware Components

4.1.9.1 Computer

4.1.9.1.1 Purpose

The purpose of computer is to contain four software components: Interface to import log data, SQLite database, Log analysis and Visualization of log analysis result.

4.1.9.1.2 Classifier

This hardware component is an instance of the classifier computer HCC-01

4.1.9.1.3 L.O.S.

Hardware component classifier HCC-01 applies here.

4.1.10 Hardware Connectors

Same as 4.1.10.

4.1.11 Software Components

4.1.11.1 Interface to import log data (SCO-01)

4.1.11.1.1 Purpose

To import log data from local directory and delete the imported log data. This component will be hand-coded by using C++. This component will also be designed in a way to cater for future evolution requirement in SSRD 6.2 ER-2.(Interface with other usage log formats)

4.1.11.1.2 Classifier

This software component is an instance of the classifier SCC-01.

4.1.11.1.3 L.O.S. Goals

Software component classifier SCC-01 applies here.

4.1.11.2 SQLite database (SCO-02)

4.1.11.2.1 Purpose

'SQLite database' is to store the log data imported from the local file directory. This component also provide the data for 'Log Analysis' component and store the results of the analysis.

4.1.11.2.2 Classifier

This software component is an instance of the classifier SCC-02.

4.1.11.2.3 L.O.S. Goals

Software component classifier SCC-02 applies here.

4.1.11.3 Log analysis (SCO-03)

4.1.11.3.1 Purpose

To analyze the log data and produce item relationship. This component will be hand-coded by using C++ and open source algorithm.(Bollen, Open Markov) This

component will also be designed in a way to allow the future evolution requirement in SSRD 6.1 ER-1.(Advanced analysis)

4.1.11.3.2 Classifier

This software component is an instance of the classifier SCC-03.

4.1.11.3.3 L.O.S. Goals

Software component classifier SCC-03 applies here.

4.1.11.4 Visualization of log analysis result (SCO-04)

4.1.11.4.1 Purpose

To show the user the graphical representation of the analysis result. This component will be implemented by using open-source software (H3 viewer, Open Motif). This component will also be designed in a way to allow the future evolution requirement in SSRD 6.4 ER-3.(Accommodating collection growth)

4.1.11.4.2 Classifier

This software component is an instance of the classifier SCC-04.

4.1.11.4.3 L.O.S. Goals

Software component classifier SCC-04 applies here.

4.1.12 Software Connectors

N/A. because the size (small) of our project, this is not applicable here.

4.1.13 Implementation Classes

4.1.13.1 Implementation Class: File Input/Output

4.1.13.1.1 Purpose

Input data from local file directory.

4.1.13.1.2 Defined In

Interface to import log data

4.1.13.1.3 Operations

Import log data() – import the log data from user’s local directory.

4.1.13.2 Implementation Class: Format conversion/Parsing**4.1.13.2.1 Purpose**

To convert the original log data format to match the table schema defined in the database. To check validity of log data, parse, store the result to database. Also show the import statistics.

4.1.13.2.2 Defined In

Interface to import log data.

4.1.13.2.3 Operations

Delete data() – delete log data based on user’s selection of log files

Check data() – check for the validity of log data, such as mal-formatted log records.

Parse data() – parse the log record to retrieve the information system needs.

Store data() – store the parsed data into database.

Display import statistics() – after importing and parsing, show the statistics of number of log records imported, and number of bad records identified.

4.1.13.3 Implementation Class: Item relationship generation**4.1.13.3.1 Purpose**

To generate item relationships based on log data.

4.1.13.3.2 Defined In

Log analysis

4.1.13.3.3 Operations

Get item relationship() – get the item relationship data from the database.

Delete report() – delete the analysis report based on user’s selection.

Create item relationship() – create item relationship based on the log data.

Store result() – store the analysis result back to database.

4.1.13.4 Implementation Class: graph clustering & centrality analysis

4.1.13.4.1 Purpose

To perform graph clustering & centrality analysis.

4.1.13.4.2 Defined in

Log analysis.

4.1.13.4.3 Operations

Graph clustering & centrality analysis() – Based on item relationship data, the detailed relationship graph collection structure tree should be produced using centrality analysis and graph clustering. For each tree level the item with highest centrality rank in the current graph is selected to be the local root. All of remaining items of the current graph are clustered based on strengths of their relations and resulting sub-graphs are made children of the local root. Then the procedure is repeated for each child sub-graph containing more than N nodes.

4.1.14 Objects

N/A, Because no objects identified in SSAD 4.1.3.

4.2 Behavior

This section will give a more detailed description about the relationship of the components working with each other and how to implement the required behavior of the system defined in [SSAD 3.3].

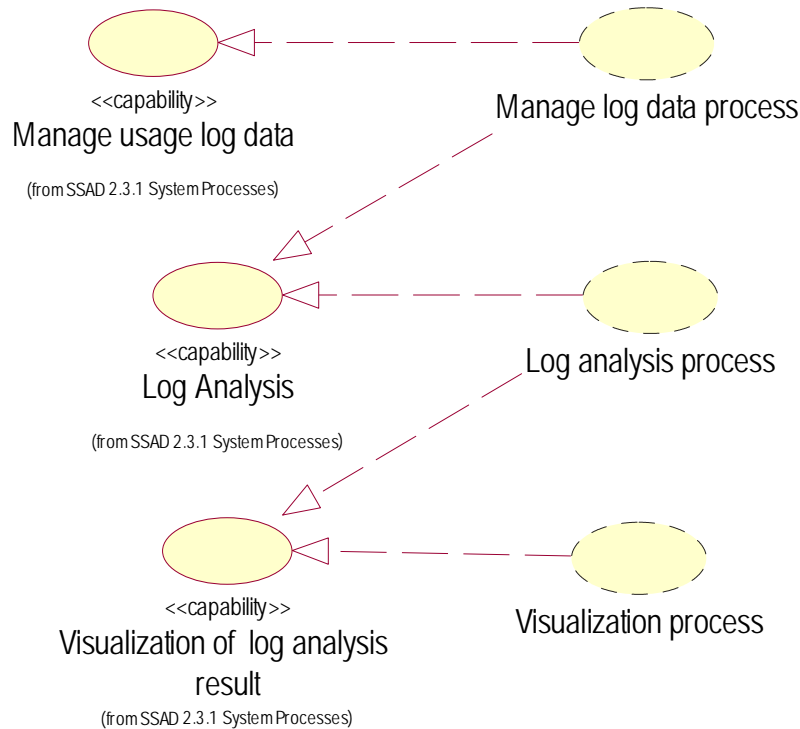


Figure 50 Behavior Model Use-Case Diagram

4.2.1.1 Process realization: Manage log data process

Identifier	Prc-1
Use-Case Name	Manage log data process
Abstract	No
Purpose	To import, view & delete log data
Actors	User
Importance	Primary
Capability	C-01 (OCD 4.3)
Requirements	SSRD. 3.2.1 (SR-1, SR-2) 3.2.2(SR-7,8) 4.1.1(.IR-1)
Risks	None
High-Risk?	No
Architecturally Significant?	No
Development Status	LCA
Overview	To import log data stored in user’s local machine to be analyzed. To delete the log

	data based on user's selection.
USER INTERFACE	See OCD 5 - prototype
Pre-conditions	Log file exists.
Post-conditions	Log file imported/deleted.
Specializes	None
Includes	None
Extends	None
Extension Points	None

Table 32 Use case description for Manage log data process

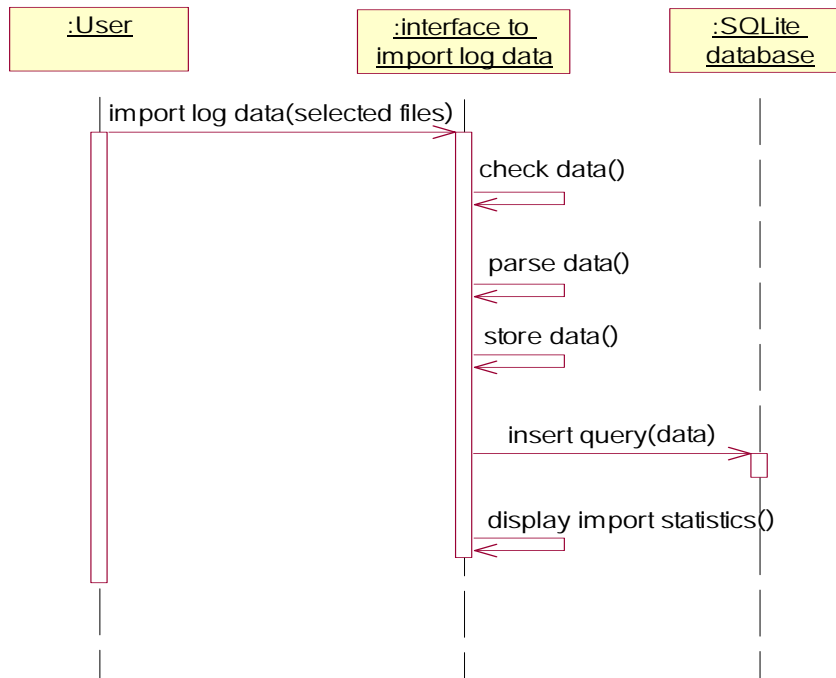


Figure 51 Sequence diagram for Manage log data process-import data

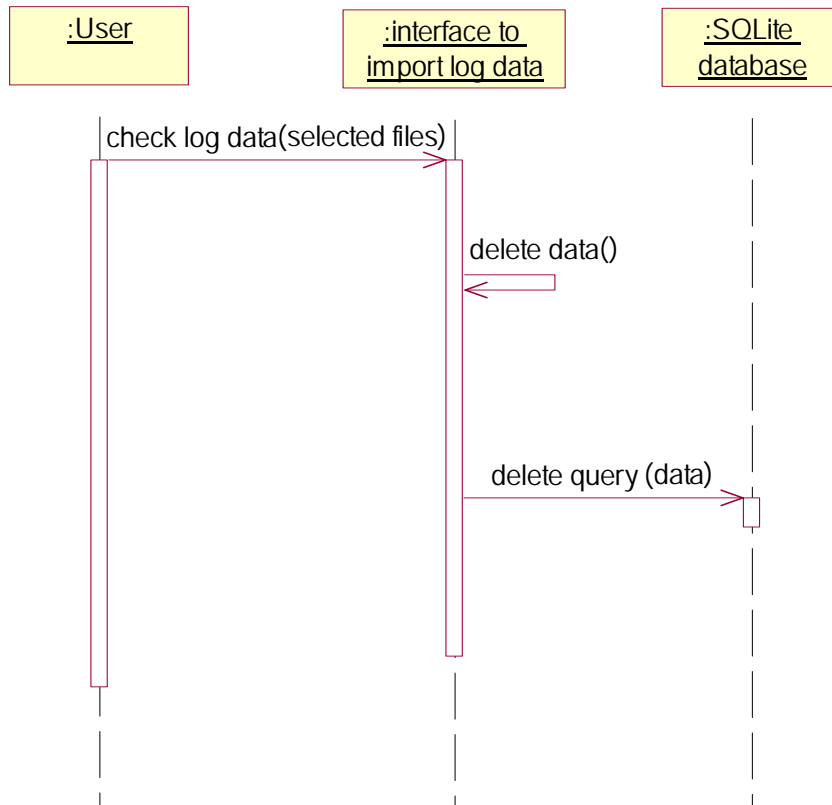


Figure 52 Sequence diagram for Manage log data process-delete

4.2.1.2 Process realization: Log analysis

Identifier	Prc-2
Use-Case Name	Log analysis process
Abstract	No
Purpose	Generate and delete analysis report.
Actors	User
Importance	Primary
Capability	C-02 (OCD4.3)

Requirements	SSRD3.2.1(SR-3,4,5), 3.2.2(SR-9),4.1.1(IR-2,3,4)
Risks	The correctness of the analyzing algorithm.
High-Risk?	Yes
Architecturally Significant?	No
Development Status	LCA
Overview	Parse the imported file, produce item relationship, graph clustering, centralized analysis, store result in a tree structure in to files.
USER INTERFACE	See OCD 5 - prototype
Pre-conditions	Log files imported
Post-conditions	Analysis report generated/deleted.
Specializes	None
Includes	None
Extends	None
Extension Points	None

Table 33 Use case description for Log analysis

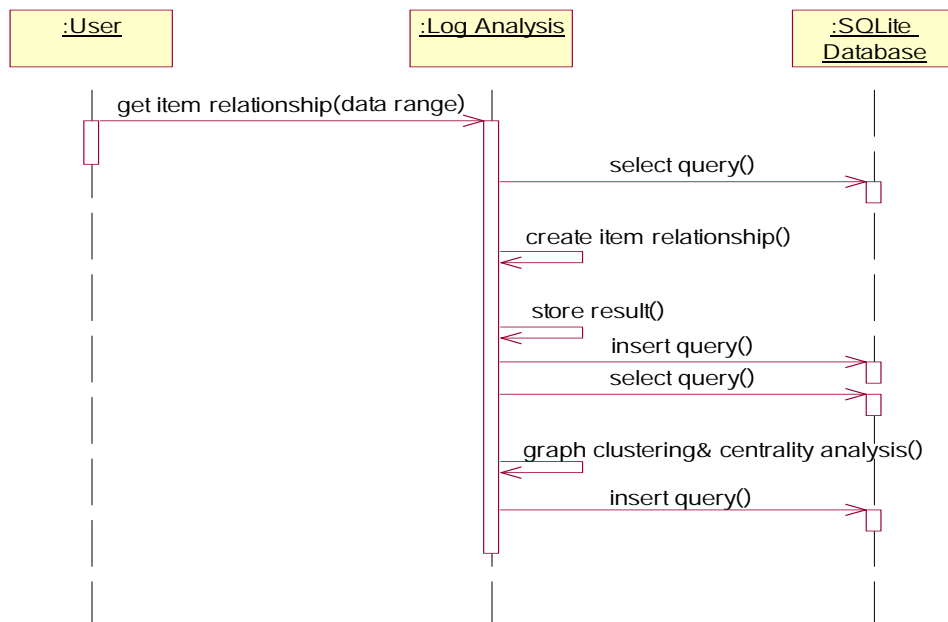


Figure 53 Sequence diagram for Log analysis -generate report

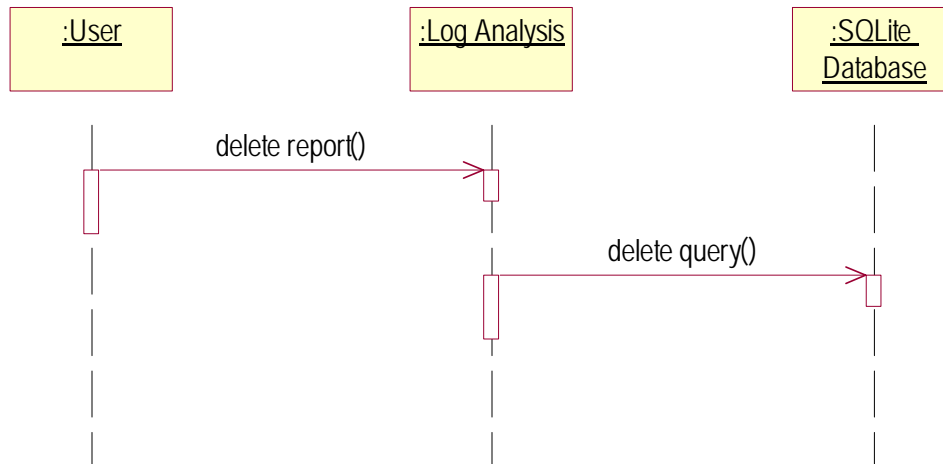


Figure 54 Sequence diagram for log analysis-delete report

4.2.1.3 Process realization: Visualization Process

Identifier	Prc-3
Use-Case Name	Visualization process
Abstract	No
Purpose	Let user brows the graph by selecting nodes.
Actors	User
Importance	Primary
Capability	C-03 (OCD4.3)
Requirements	SSRD 4.1.1 (IR-5), 3.2.1 (SR-6), 3.2.2(SR-10)
Risks	This use case is implemented by open source product.
High-Risk?	Yes
Architecturally Significant?	No
Development Status	LCA
Overview	Given the input file, produce graph nodes. And allow user to browse it by selecting nodes.
USER INTERFACE	See OCD 5 - prototype
Pre-conditions	Analysis data produced by 'log analysis

	process' exits.
Post-conditions	Display graph.
Specializes	None
Includes	None
Extends	None
Extension Points	None

Table 34 Use case description for Visualization process

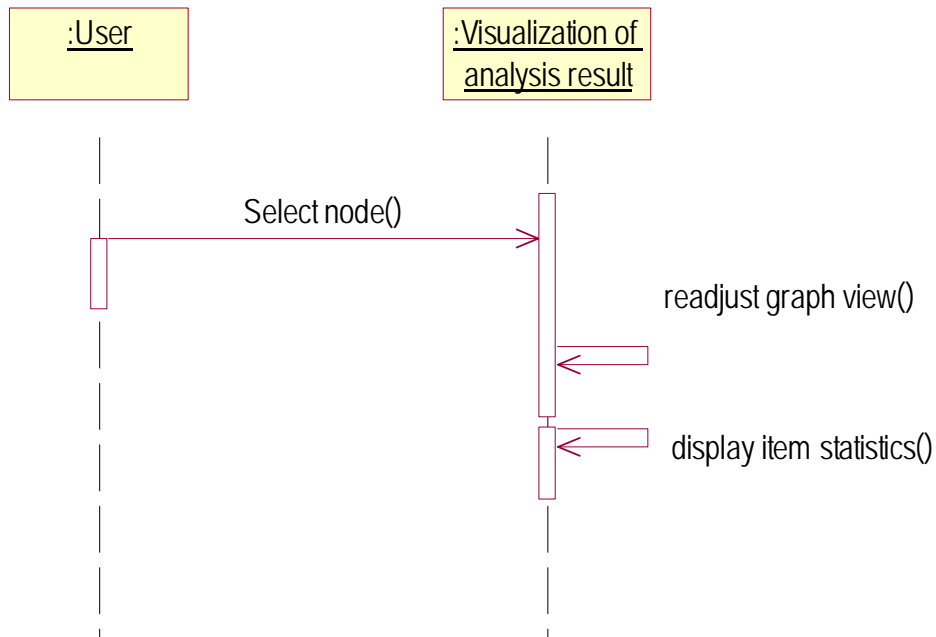


Figure 55 Sequence diagram for visualization process

4.3 L.O.S. projected

L.O.S Goal	Applies To	How	Projected Value	Evaluation Technique
LR-1 [SSRD 5]: Dependability	SCO-01, SCO -02, SCO -03, SCO -04	Equally	Pass 95% of test cases	Estimation
LR-2 [SSRD 5]:	SCO -04	Equally	Satisfied with	Estimation

Usability			the client	
LR-3 [SSRD 5]: Operability in multitasking environment	SCO -03	Equally	Pass 95% of test cases	Estimation
LR-4[SSRD 5]: Performance on data of current scale	SCO -04	Equally	Able to perform properly with 10% increase.	Estimation

Table 35 L.O.S projected.

4.4 Patterns & Frameworks

We have chosen multi-layer pattern architecture.

Name	Description	Benefits, Cots & limitations
Multi-layer Pattern	Multi-Layer system is a kind of software system and uses multi-layer architecture pattern. See http://www.site.uottawa.ca:4321/oose/index.html#multi-layer_table	It replaces a layer by an improved version, or by one of different capabilities. In addition, it also increases reusability. It divides and conquers since the separate layers can be independently designed.

Table 36 Architectural Styles, Patterns & Frameworks

4.5 Project Artifacts

This section is to represent about how the software components and classes are assigned into project artifacts. We will show the hierarchical structure in this section.

<<Directory>> Data Mining in Digital Library

File-01: Interface to import log data (SCO-01)

File I/O

Format Conversion

File-02: SQLite database (SCO-02)

File-03: log analysis (SCO-03)

Item Relationship generation

Graph clustering & centrality analysis

File-04: Visualization of analysis result (SCO-04)

5. Glossary for System Analysis and Design

DA- Digital Archive : Collection of electronic and multi-media documents, such as images, photos, audios, and videos

GUI: graphical user interface.

Log Data: they are the data gathered when users visit the library's digital achieves, they contain: user's IP address, item's id, and the time of visit.

LANL : Los Alamos National Laboratory.

L.O.S – Level of Service

MBASE: Model-based System Architecting and Software Engineering (MBASE).

UML: Unified Modeling Language (UML).

6. Appendices

N/A