Gold Standard Website

Muhammad Azeem
&
Khurram Ahmad
Acknowledgement

This thesis is submitted for the degree of Master in Software Technology at the Växjö University. We would like to express our sincere thanks to our supervisors Tobias Andersson Gidlund and Professor Welf Löwe and for their guidance and help. We are greatly indebted to our parents for their love, support and inspiration. We would like to thank everyone who contributed to this project. Many thanks are also directed to our teachers and all the assistances for great collaboration and guidance throughout the duration of our Master degree in Software Technology. Finally thanks to all of our colleagues and friends at Vaxjo University whose support through all the ups and downs has been so very much appreciated.
Abstract

The aim of this thesis is to design a web base system which provides functionality of comparison between two java files on the basis of point-to information (P2I). User will upload Java files and analysis of Java files called point-to analysis (P2A). System will store the files in the file system for reference and download in later time. System will extract the information called P2I from P2A and it will store that information in the database.

Database should be flexible to accommodate the changes in P2A file and system should be able to extract the P2I and store it in database with minimum support of system administrator.

Keywords: Point-to analysis (P2A), point-to information (P2I), comparison
# Table of Contents

## 1 Introduction

1.1 Motivation ........................................................................................................ 1
1.2 Problem definition ....................................................................................... 1
1.3 Report Structure ......................................................................................... 1

## 2 Working with P2A

2.1 XML understanding .................................................................................. 2
    2.1.1 Analysis introduction ....................................................................... 2
    2.1.2 P2I ................................................................................................. 3
2.2 XML document validation ........................................................................ 3
2.3 Document Type Definition (DTD) ............................................................. 3
    2.3.1 XML Schema .................................................................................. 3
    2.3.2 Technology used in the thesis and why? .......................................... 3
2.4 XML PARSING ......................................................................................... 4
    2.4.1 SAX: Simple API for Xml ............................................................... 4
    2.4.2 XML DOM: Document Object Model ........................................... 5
2.5 Parsing P2A file ........................................................................................ 6

## 3 Requirement Analysis and Design

3.1 Methodology .............................................................................................. 7
    3.1.1 Waterfall software development methodology ................................ 7
    3.1.2 Waterfall software development methodology implementation ...... 8
3.2 Requirements .............................................................................................. 8
    3.2.1 Functional Requirements ................................................................ 8
    3.2.2 Non-Functional Requirements ....................................................... 9
3.3 Use Case Diagram ....................................................................................... 10
3.4 System Sequence Diagram ....................................................................... 12

## 4 Implementation

4.1 Dreamweaver MX 2004 ............................................................................. 13
4.2 Apache 2.2.11 ......................................................................................... 13
4.3 PHP 5.2.9-2 ............................................................................................ 13
4.4 MYSQL 5.1.33 ........................................................................................ 13
4.5 phpMyAdmin 3.1.3 ................................................................................ 13
4.6 Website Development ................................................................................. 14
4.6.1 Three tier architecture ................................................................................... 14
4.6.2 Client side development ................................................................................. 15
4.7 Server side development.................................................................................... 19
  4.7.1 Createuser.php ............................................................................................... 19
  4.7.2 fileDownload.php .......................................................................................... 19
  4.7.3 Fileupload.php ............................................................................................... 19
  4.7.4 loginScript.php .............................................................................................. 19
  4.7.5 Newschema.php ............................................................................................. 19
  4.7.6 Verification.php .............................................................................................. 19
  4.7.7 Connect.php .................................................................................................. 19

5 Database design and implementation ........................................................................ 20
  5.1.1 Users ............................................................................................................... 21
  5.1.2 Files ................................................................................................................ 21
  5.1.3 New Schema ................................................................................................... 21
  5.1.4 P2A ................................................................................................................. 21
  5.1.5 Main_Table .................................................................................................... 21
  5.1.6 P2I_analysis Setup (Example Table) ............................................................ 21
  5.1.7 P2I_Analyzed Project (Example Table) ....................................................... 21
  5.1.8 P2I_Analyzed Project Version (Example Table) ......................................... 22
  5.1.9 P2I_heapObj (Example Table) ....................................................................... 22
  5.1.10 P2I_Call (Example Table) .......................................................................... 22
  5.1.11 How to Extend database ............................................................................... 22

6 Conclusion and Feature enhancement ....................................................................... 23
  6.1 Conclusion .......................................................................................................... 23
  6.2 Future work ........................................................................................................ 23

7 References .............................................................................................................. 25
  7.1 XML basic and Schema ..................................................................................... 25
  7.2 Related Studies .................................................................................................. 25
## Table of figures

**Figure 2:1 DOM Tree [18]** ................................................................. 5  
**Figure 3:1 Waterfall Model [2]** .......................................................... 7  
**Figure 3:2 Functional Requirements** ................................................. 9  
**Figure 3:3 Non Functional Requirements** ........................................... 10  
**Figure 3:4 Use Case Diagram** ............................................................ 11  
**Figure 3:5 Sequences Diagram** .......................................................... 12  
**Figure 4:1 [13]** ............................................................................. 14  
**Figure 4:2 Home Page** ................................................................... 15  
**Figure 4:3 Administrator Home Page** .............................................. 16  
**Figure 4:4 Downloading and Comparison** ........................................ 17  
**Figure 4:5 Results of File Comparison** ............................................. 18  
**Figure 5:1 Database Design** ............................................................... 20
1 Introduction

System can store Java files and analysis of the Java files. System can extract the P2I from analysis file P2A and store it in the database. System provides the facility to view or download the Java files along with the analysis files. User can compare two files on the basis of P2I stored in the database. Database can store different type of P2I.

1.1 Motivation

Today software world is growing very fast especially lot of work in object oriented programming has been done in past few years [1]. Some time user has many solutions for a single problem, if a user wants to perform sort operation there are many algorithms available [2]. But the problem is to decide which solution is better and why.

This question is our motivation; this question demands a system which should be able to store the information based upon the analysis of files. Then system should provide a user friendly and easy to access interface for file comparison.

1.2 Problem definition

In this project our working domain is to provide a web base system for storing Java files and point-to information of Java files. Point-to information will be retrieved from the point-to analysis file provided by the user in XML format. We have to design a database which will store Java files, point-to analysis files and point-to information. Database will be flexible to store the variation of P2I. System will provide the interface for comparison of two Java files.

1.3 Report Structure

Chapter 2 is about point-to analysis (P2A); this chapter is about different parsing and validation technologies of XML. Chapter 3 is all about requirements analysis and design of the system. It also contains the description of the system in the form of use case diagram and system sequence diagram. Implementation details of the system and different tools and technologies used in the system development are described in Chapter 4. Chapter 5 is about the design and implementation of the database. It talks about how to extend the database in the future? Conclusion and future work are discussed in chapter 6.
2 Working with P2A

In this chapter we will discuss about basic XML, validating of XML and XML parsing. We will also discuss how to handle the analysis file in this project.

2.1 XML understanding

XML (eXtensible Markup Language) is a general purpose data format. Two version of xml are in place, First 1.0 version was introduced in 1998 without any new versions name it has five editions. Second version of XML was 1.1 and released in February 2004. In February 1998 XML become a W3C Standard. XML has basic syntax to carry information between different kinds of computers and applications. It is being used as a format for transforming the information. It is a famous data formats then the previous because it support both tabular data and structured data. It is a platform independent; XML does not based on any programming language, software vendors and operating system. [3][4][5][6]

Some Design Goals

- It should support wide variety of application
- It should be easy to write and process.
- It should be reasonably clear and easy to use over internet.

Xml document should be well formed and valid.

In this thesis users will upload an analysis file in the XML format and System need to extract P2I. This xml file will contain information about the object oriented file. It is a stand-alone object oriented program file.

P2A file is divided in two basic parts. Both parts are discussed in detail below.

- Analysis introduction
- P2I

2.1.1 Analysis introduction

From this part of the P2A file system will extract some basic information about Java/object oriented files. It will provide some basic information about the P2A file.

Project name

In the first part one element will contain the name of the project which we are going to analyzed. In P2A file this information can be seen under the tag of analyzed project.

\[
<analyzedProject>{name of the project}</analyzedProject>.
\]

\[
<Start tag>   data/ PCDTA   <end tag>
\]

Project version

One element in the P2A file will contain analysis version of the Java file. In this project user is allowed to upload different analysis version for one Java file. So it is important for the system to make a distinction between two P2A of one Java file. One XML statement will contain the project version information.

\[
<analyzedProjectVersion>{version of analysis}</analyzedProjectVersion>
\]
Project analysis setup

Third information in the introduction part is about the method, of which P2A is being generated. In the P2A file one statement will show, if analysis are generated automatically or manually. Within the syntax of P2A file all this information can be shown under the tag of “analysis Setup”.

<analysisSetup>{generation method}</analysisSetup>

2.1.2 P2I

Second part of P2A file will contain P2I. This is an important part, that need to be stored in the database and for the users P2I is the base for comparison of different files. In this part there can be any kind of information related to object oriented files, for example information about heap object and method invocation. For the heap object it can be, name of different object type attributes in the file and information about the data they are going to store. P2I can contain the information of different objects created in the file. It can be the information of methods called by different objects. P2I part of analysis file will contain any kind of information about object oriented files.

2.2 XML document validation

Validation means to check the document against the construction rules. Whenever users upload P2A file system needs to validate the analysis document. If this analysis file is valid then it will extract the P2I and store them to the database. If file is not validated by the used technology then user is not allowed to upload the file to the system. There are two widely used technologies to validate any xml file are.

- DTD (Document Type Definition)
- Xml Schema

2.3 Document Type Definition (DTD)

“A Document Type Definition (DTD) defines the legal building blocks of an XML document. It defines the document structure with a list of legal elements and attributes.”[7]

Purpose of DTD is to define a structure for XML document. It will define all the possible elements, attributes and attribute type that can be in the xml file. If any XML document is written on the base of DTD then it should be compare to the DTD defined format for validation. When an xml file will be created on the base of DTD then file will contain the information about DTD. [8]

2.3.1 XML Schema

When we talk about Xml schema it means W3C Xml Schema technology. It is another technology to validate the XML document. XML schema is also used to draw the boundaries of xml document. XML schema was approved as a W3C standard on 2-May-2001. It used to define the legal building blocks of an xml documents. It defines the structure of elements, attributes, child attributes and detail about the data they will contain. Schema will lay down the rules for the element data; it can be some text/number data, empty, fixed or flexible. [9][10]

2.3.2 Technology used in the thesis and why?

Both technologies have their own advantages but for this thesis XML schema will be used because
• Schema use basic XML syntax while DTD use separate syntax.
• Schema supports the Namespace recommendation.
• Schema facilitate user to create complex and reusable content very easily.
• Schema support the object oriented concept like inheritance.
• Schema allows to validate text content based on both built in and user defined data types.
• Extensible nature of schema. It can be reused in other schemas.

Schema offer all the features a DTD can do, in addition schema provides extra functionality. Schema allows more data types and to create custom data types. Schema provides more control on elements. [11]

2.4 XML PARSING

For XML document parsing there are lot of technologies are available in the market. There are two widely used xml parsing technologies [12]. Both have their advantages and disadvantages one can use them according to the requirements.

• Simple API for Xml (SAX).
• Document Object Model (DOM).

We will discuss these two in detail so we can decide which one can fulfill our needs.

2.4.1 SAX: Simple API for Xml

SAX is a standard API for xml. First SAX 1.0 was released in May-11-1998. It was developed to perform efficient parsing of large xml document. SAX does not have any default object model. It only read the events occurs in the documents. SAX moves from start tag of the document towards the end tag of the document in a linear fashion. It is fast so it is best choice for large documents. SAX parser is very useful when parsing a large file and we do not want to load complete file in the main memory. SAX parser can work at low connection speed because it processes the bytes as soon as they arrive instead of waiting for the document to complete first. SAX requires less memory and proceeds sequentially. SAX is an event driven parsing technology, it work on the XML tags whenever a tag will found in the file sax function will be called. Working of the SAX is depend on the tag, [13][14][15]

• Start tag (it includes tag name, attributes and other information in the tag).
• Tag data (this portion carries the XML element data).
• End tag (this is the last part which will ensure all the tag has been parsed).

There are three main steps to create SAX parser.

• Create an object model.
• Create a SAX parser.
• Create a document Handler.

Advantages

• It reduced memory and the CPU usage since it works one section at one time.
• It is customizable because when an event is fired then the associated applications will retrieve data.
• It is fast and support pipelining. It produced output as soon as data is parsed.
• We can stop processing at any time so it good to fetch particular data.
Disadvantages

- It works in one direction. It can only parse forward in the document.
- Since there is only one part of XML file is in the memory at one time so it is difficult to perform structure operation.
- It is not possible to access data in random order by using SAX.

2.4.2 XML DOM: Document Object Model

“The W3C Document Object Model (DOM) is a platform and language-neutral interface that allows programs and scripts to dynamically access and update the content, structure, and style of a document.”[16]

Document object model is W3C (World Wide Web consortium) standard to parse the XML document. DOM places a layer between the XML file and the application. Parser read the XML data and feed it to the DOM. Then DOM is used by application and performs different operation on this data. DOM present XML file in a tree structure. It is language and platform independent. DOM has three levels, XML DOM define some standard set of object for xml file, now it is a standard way to access any xml document.

DOM level 1: It used to provide some basic level of functionality for navigating and operating HTML and XML documents. First time level 1 declared as W3C standard.

DOM level 2: This is an up graded version of level 1. It gives supports for XML namespaces and style sheet.

DOM level 3: This is a very comprehensive level since it gives a complete mapping between DOM and XML. It also supports entity declaration. [17][18]

DOM Functionality is very interesting; it accesses the XML document and creates a tree in the memory. The tree is made up of a hierarchy of nodes. Mostly compose the nodes in different category like element node, attributes node, entity node and text node. It allows random access to each node. Unlike SAX it allows very powerful document navigation. Working with large size documents it needs more memory. [19]
In figure 2.1 tree structure of DOM is shown. Root node (address book) contains two records (persons) show in tree structure. Unlike SAX, DOM provides complete access by allowing backward and forward movements. Although it is a not a memory efficient technology but it is a useful for complex search.

Advantages

- It provides random access to the data.
- It is good when we need to implement a complex searches.
- It is a best choice when we need to perform XSLT transformation.
- We can use when we need to modify the XML file.

Disadvantages

- DOM is a resource intensive parser, since xml structure stay in the memory, so the large file needs more memory.
- It is a slow when we compare it to SAX.

2.5 Parsing P2A file

After complete comparative study of both DOM and SAX we have decide to use SAX for parsing xml analysis file. Since we do not need to perform any changes in analysis file and search is also not complex so SAX will work fine with this scenario.” The thing I like most about SAX is that it allows you to ignore all the portions of your XML document that you don’t care about, making it not only trivial to only pick the information you are interested in, but also easier to migrate your schema over time, should you decide to do so [21]”. SAX is efficient and it save memory, SAX parser will parse only those file which are well formed.
3 Requirement Analysis and Design

In this chapter we will analyze the system requirements and interaction between user and the system, using UML diagrams.

3.1 Methodology

In this section we will discuss software development methodology which we have used for our work. After considering the goal of our project we have to attain we choose water fall model to develop our software.

3.1.1 Waterfall software development methodology

Waterfall model has different phases as shown in the figure 3.1. Waterfall model has different phases. First phase of the Waterfall model is requirements, in this phase all the software requirements are gathered. Better job in this phase leads to better end. In this phase software goals and constraints are analyzed.

System specification phase is based upon the requirement phase. This phase define the system functionality. System specification and requirement phase can be seen as single.

In system design phase system has to be designed properly before any implementation. This phase defines the main blocks and components of the system. Software components should be compatible with the user requirements and possible scalability of the system.

![Waterfall model diagram](image)[23]
In implementation phase software engineering work translated into code. In this phase software development started with small components called units. These components can be integrate later to achieve complete functionality of the system.

In system integration phase different software components integrated together. In above phase every software component developed independently and tested its functionality through unit testing. In this phase units are brought together and performed testing to insure that they working in combined.

Last phase of the waterfall model is maintenance, in this phase software handed over to the customer with expectation that it meets the user requirements. But if customer found some problems in the software then these problems are solved in this phase. [22][23]

3.1.2 Waterfall software development methodology implementation

We adopted this methodology because requirements and specifications of the system are crystal clear. Second reason for selecting Waterfall model is development of the system in units like user authentication, to reduce the complexity and testing the effectiveness and efficiency of every unit. After completing unit testing these units will be integrated to test the functionality of whole system.

3.2 Requirements

Requirements analysis is the first phase of the system development and waterfall model. Design and implementation of the system is based on the requirements specifications. Requirement specification helps us to understand and define the scope of system. Requirements specifications also draw the boundaries of the system. There are two types of requirements.

- Functional Requirements
- Non-Functional Requirements

3.2.1 Functional Requirements

In this section we will describe and discuss the functional requirements of the proposed system. Function requirements defines the important component and functions of the system, it also includes the required function of the proposed system, level of importance with a brief description.

<table>
<thead>
<tr>
<th>FR-01</th>
<th>Security</th>
</tr>
</thead>
<tbody>
<tr>
<td>Importance</td>
<td>Essential</td>
</tr>
<tr>
<td>Description</td>
<td>To upload the Java files and p2a user must login in the system.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FR-02</th>
<th>Web interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>Importance</td>
<td>Essential</td>
</tr>
<tr>
<td>Description</td>
<td>For communication between user and database.</td>
</tr>
</tbody>
</table>

<p>| FR-03 | Upload Java file |</p>
<table>
<thead>
<tr>
<th>Importance</th>
<th>Essential</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>User must upload Java file with P2A.</td>
</tr>
<tr>
<td>FR-04</td>
<td>Upload P2A</td>
</tr>
<tr>
<td>Importance</td>
<td>Essential</td>
</tr>
<tr>
<td>Description</td>
<td>User must upload P2A for extraction of P2I.</td>
</tr>
<tr>
<td>FR-05</td>
<td>File System</td>
</tr>
<tr>
<td>Importance</td>
<td>Essential</td>
</tr>
<tr>
<td>Description</td>
<td>File system is required to store the Java files.</td>
</tr>
<tr>
<td>FR-06</td>
<td>Database</td>
</tr>
<tr>
<td>Importance</td>
<td>Essential</td>
</tr>
<tr>
<td>Description</td>
<td>To store the P2I.</td>
</tr>
<tr>
<td>FR-07</td>
<td>Search</td>
</tr>
<tr>
<td>Importance</td>
<td>Desirable</td>
</tr>
<tr>
<td>Description</td>
<td>To find the files and schema in the database.</td>
</tr>
</tbody>
</table>

Figure 3.2 Functional Requirements

### 3.2.2 Non-Functional Requirements

In this section, we will discuss about the non-functional requirements of our system. Non-functional requirements are required to improve the functionality of the functional requirements.

<table>
<thead>
<tr>
<th>NFR-01</th>
<th>Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Importance</td>
<td>Essential</td>
</tr>
<tr>
<td>Description</td>
<td>The system should work fast.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NFR-02</th>
<th>Simple and user friendly interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>Importance</td>
<td>Essential</td>
</tr>
<tr>
<td>Description</td>
<td>To make the system easy to understand and use.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NFR-03</th>
<th>Paging</th>
</tr>
</thead>
<tbody>
<tr>
<td>Importance</td>
<td>Essential</td>
</tr>
<tr>
<td>------------</td>
<td>-----------</td>
</tr>
<tr>
<td>Description</td>
<td>To break the large amount of data in the pages so that easy to navigate.</td>
</tr>
</tbody>
</table>

**NFR-04**  
**Application Maintenance**

<table>
<thead>
<tr>
<th>Importance</th>
<th>Essential</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Application should be maintained and well documented.</td>
</tr>
</tbody>
</table>

**NFR-05**  
**Application Scalability**

<table>
<thead>
<tr>
<th>Importance</th>
<th>Desirable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>The system should be capable to enhance in future.</td>
</tr>
</tbody>
</table>

**NFR-06**  
**Platform Independent**

<table>
<thead>
<tr>
<th>Importance</th>
<th>Desirable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>The system should work on all computer machines.</td>
</tr>
</tbody>
</table>

---

**Figure 3.3** non Functional Requirements

### 3.3 Use Case Diagram

Figure 3.4 shows how the actors will interact with the system. There are three actors in the use case diagram. Solid lines show direct access of actors to different activities, while dotted lines show that this use case can be accessible only after fulfilling the pre requisite. Like in order to upload a file the actor should be log in to the system. Administrator login is provided for administrator to access admin panel. In admin panel administrator can perform the different tasks like user management, schema management and database management.
Figure 3.4 Use case Diagram
3.4 System Sequence Diagram

Sequence diagram use to show the order user will interact with the system.

- User will type the URL in the browser and browser will pass this HTTP request to web server if server is available it will open the application otherwise a message will be displayed. If server is available web page will be displayed.
- In the second step user will type user name and password for authentication if server will verify this username then a connection will be established otherwise a message will be shown for incorrect parameters.
- Once connection is established now user can perform any task. User can send different query to server until session expired. For disconnecting the current connection user need to send a request to close the user session, now server will reply with disconnecting the session.

Figure 3:5 sequences Diagram
4 Implementation

In this chapter we will describe the tools and technologies used in our implementation. In this chapter we also discuss our program files and purpose of those files.

4.1 Dreamweaver MX 2004

Dreamweaver MX 2004 is a web development tool with GUI and it support PHP. Dreamweaver has edge on other tools like zend studio due to its WYSIWYG functionality. It has strong support for CSS. It provides drag and drop functionality for HTML contents. Dreamweaver is suitable for beginner and expert web developers. It support different type of views like design view, code view, code and design view.

One more reason to use it is its direct connectivity with MYSQL and FTP support. It also checks the current browser compatibility and broken links. Dreamweaver has support for XML and JavaScript. [24][25][26]

4.2 Apache 2.2.11

Apache web server is very popular web server. According to our knowledge and experience there were two choices Apache and IIS. The major drawback with IIS is environment. IIS only work under windows environment while apache works with different environments. So apache is suitable for our system because our requirement is platform independence. Apache also speed up the execution of dynamic components for language like PHP. [27]

4.3 PHP 5.2.9-2

“PHP (recursive acronym for PHP: Hypertext Preprocessor) is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML.”

The reason for using PHP in this project is portability and platform independence. PHP can work in different environments like windows, Linux. It is easy to learn and simple to configure. PHP has support for web servers like apache, IIS, Netscape and many others.

We are working with XML and PHP has support for XML. PHP support SAX which is used in this project for information extraction from P2A.

PHP has support for wide range of databases including MYSQL which is used in this project. PHP has support for structure programming and object oriented programming. [28][29][30][31]

4.4 MYSQL 5.1.33

Although among many other choices for database server there were three most prominent Oracle, SQL Server and MYSQL. We choose MYSQL due to our requirements and constraints like flexibility, XML support, price and ease.

MYSQL fulfill all of our requirements. MYSQL has great features like it is easy to install and configure. It has high performance, scalability and flexibility. It has robust transactional support. It provides great backend support to web site due to its high performance query engine. MYSQL provides great security and reliability. [32][33]

4.5 phpMyAdmin 3.1.3

phpMyAdmin provides support for MYSQL administration by web interface. phpMyAdmin easy to understand and user. It provides the facility of managing MSQL users and privileges. It provides support for import data from CVS and SQL and export data to various formats like CVS, SQL, XML, PDF and some others. [34]
4.6 Website Development

We used three tiers architecture in our web development to make the system secure and reliable.

4.6.1 Three tier architecture

Three tiers architecture is any system which follows this division as mentioned in figure below. [35]

- Client Tier
- Middle Tier
- Data Storage Tier

Client Tier

Client tier is client side application like web browser. User send request by using client tier like user enter the URL in the browser or submit a form data.

Middle Tier

In web base system like we have middle tier is a web server. Web server plays the role of mediator between client tier and data storage tier. It receive the request from client tier and forward it to the data storage server similarly it receive the data from data storage tier and forward it to the user or client tier.

Data Storage Tier

In three tier architecture data storage is always a third tier like in this project. Data storage tier store the data and it communicate with the client tier through middle tier.

Figure 4:1 [36]
4.6.2 Client side development

In this section we will discuss about client side development of the system, interfaces and functionality.

In the home page all the controls are available for different users to access the system and various parts of the system.

Figure 4.2 Home Page

In figure 4.2 the home page of the Gold Standard website can be seen. Home page is divided into two panels. First is access controls, second is menu Control. By keeping the access control and menu control separate we can improve the readability of the home page.

Access controls

- Home
- New user
- Sign in / Sign out
- Admin Login

Menu

- Home
- Search
- Upload P2A
- View Schema
- File comparison
- User Management
- Schema Management

Access Controls

In the access control panel there are four tags in order to provide the easy accessibility to the users and the administrator.
User authentication

There are three types of users who will visit the system. First, an ordinary user who can only access the data in read only format as a guest. Guest user is not allow upload any file but can download different files, perform the comparison. Second type of user is authorized to upload the file as well.

Before getting access to different operation user need to register with the system. There is a registration form that needs to be filled in order to access the complete functionality of the system. User need to fill this form, without this information user can not be registered to the system. For security purpose there is one filed named “Security code” it contain the random generated code user need to read this code and write in the field given blow, Security Code is not case sensitive. After successful account creation user need to contact administrator by mail for account activation.

Third type of user is administrator. Administrator tasks are schema management, user management and database management.

Administrator Panel

A dedicated portion for the administrator, where administrator can perform required task, in this panel administrator can perform following tasks.

- Schema management
- User management

Figure 4:3 Administrator Home Page

User Management

Administrator can manage users account by changing the status of different users to active or de active. Administrator can change the status of any user by simply click on the “Change Status”. The status of the entire users will be shown under “Status” tag. When status will be changed by the administrator it will be updated. If status of any user is changed to “De Active”, now user is not allowed to login to the system. Whenever user will try to login, a message will be shown that contact to the administrator.
Schema Management
Administrator can upload new schema for different P2I comparison. Administrator can perform the changes in the schema.

Menu control
In this section users can access to all the operation available. Menu control provides easy access to different interfaces.

Uploading P2A and files
For uploading the P2A file and object oriented file some additional information need to store along these files. For this purpose user need to fill “Upload P2A” form. In this form user need to give file name, compiler version, how to compile and both object oriented and analysis file.

File Comparison and Downloading
In this section user can search different files in the data base for comparison and if want to download them.

<table>
<thead>
<tr>
<th>Select</th>
<th>File Name</th>
<th>P2A File</th>
<th>File Size</th>
<th>Upload Time</th>
<th>Uploaded By</th>
</tr>
</thead>
<tbody>
<tr>
<td>☑</td>
<td>file1</td>
<td>P2A File</td>
<td>683 bytes</td>
<td>2006-06-15 01:27:59</td>
<td>mailik</td>
</tr>
<tr>
<td>☑</td>
<td>file2</td>
<td>P2A File</td>
<td>724 bytes</td>
<td>2006-06-15 01:29:21</td>
<td>mailik</td>
</tr>
<tr>
<td>☑</td>
<td>file3</td>
<td>P2A File</td>
<td>204 bytes</td>
<td>2006-06-15 01:29:55</td>
<td>mailik</td>
</tr>
<tr>
<td>☑</td>
<td>file4</td>
<td>P2A File</td>
<td>790 bytes</td>
<td>2009-06-15 01:32:29</td>
<td>mailik</td>
</tr>
<tr>
<td></td>
<td>gfgj</td>
<td>P2A File</td>
<td>424 bytes</td>
<td>2006-06-15 15:50:56</td>
<td>khurram</td>
</tr>
</tbody>
</table>

Figure 4:4 Downloading and comparison

Compare
This is a core functionality of this thesis, system will facilitate user to compare the P2I of two file in order to check the quality of the Java files. For this purpose system will provide the check boxes along with files. User can mark different files and press the compare button in order to get a comparison view of the files. Users also need to specify the P2I on the bases of selected P2I comparison will take place. In the figure 4.4 there are two type of P2I are available for comparison one is “call” and second is “heap object”. When administrator will add new P2I to the system, then all the newly added P2I will be shown after the present P2I in a sequence. Comparison of these files will be shown in a new form. User can only compare two files at a time. Within the scope of this thesis user can only compare on the bases of the number of object available in the Java files. After the future extension user will be able to compare multiple files at the same time and on the bases of different P2I. Future enhancement for comparison of multiple files can be obtained by just removing the condition and provide space where results need to be shown.
In the figure 4.5 an example result of file comparison is shown. In this scenario two Java files are compared on the bases of “call” P2I, as it is shown in the figure there file name and then total number of P2I in the file. After that there is number of P2I in the file and detailed information about this P2I.

![File Comparison](image)

**Figure 4:5 Results of file comparison**

**View Schema / add new Schema**

For uploading any XML analysis file it is very important that XML file must satisfied the criteria defined by the schema. Users can download different schemas available in the system so user can validate XML file before uploading to the system. To ensure the validity of XML file we will validate the analysis file before uploading it. XML file should be well formed and according to the schema.

In this thesis example schema is being used but administrator can change this schema and upload a new extended schema. Users are not allowed to upload schema, it is only administrator who can modify the schema. User can access this new schema through “view schema” form.
4.7 Server side development

In this section we will discuss and describe our server side development, major files and their functionality.

4.7.1 Createuser.php

Createuser.php has server side file for creating new user. This file receives and verifies the data from the client and uploads it to the database.

4.7.2 fileDownload.php

FileDownload.php file handles the client request for Java and P2A file downloading from the file system.

4.7.3 Fileupload.php

Fileupload.php is used to handle the file upload form data. It receives the files from user, verify the files and upload the files in the file system.

4.7.4 loginScript.php

LoginScript.php receive the login data like username and password from user and it verifies that data with the database. If verification successful then it establishes a session otherwise it returns an error message to the user.

4.7.5 Newschema.php

Newschema.php is used to upload new schema in the database. It receives the schema from client and performs schema validation operation. If schema validation successful then it upload schema in the database otherwise it return an error message to the client.

4.7.6 Verification.php

Verification.php is used to handle the request for security code verification. Security code is used to prevent the database from spam. In case of success it returns true otherwise it return false.

4.7.7 Connect.php

Connect.php is used to establish a connection between web server and database server.
5 Database design and implementation

We have 10 tables in our database but our system support to add more tables as schema and P2I changed. Some tables are mandatory for the system like Users, Files, and P2A while remaining tables are just to save the testing data and these tables are based on example schema.

Figure 5:1 Database Design
5.1.1 Users

Users table contains all the information of users. Users table has following columns: uname, uid, password, email, city, country, gender, status and type. Status field is used to recognize that user is active or deactivate. Type field is used to distinguish between users group like user or administrator.

5.1.2 Files

Files table will store the object oriented files. Object oriented files are important to store to make them available for later use. File table has relationship with the following tables, like it is using the “user id” of user table as foreign key and P2A, P2Iheap, P2Icall, P2Iintro tables are using “File id” as foreign key. Files table will store all the relevant information about files, like

- Files size.
- Compiler version
- Type of the file
- Uploading data
- How to use/start

5.1.3 New Schema

New schema table will hold all the information about the available schema. In order to make sure that all the uploaded schema will remain available for downloading. Whenever administrator will upload a new schema it will store in the schema table and descending order to the user so the last uploaded schema comes first. New schema will use “user id” of User table as foreign key.

5.1.4 P2A

P2A table will use to store XML analysis files. System will extract P2I from P2A and store it in separate tables. But as an extra feature system will store P2A files in order to facilitate the user to access this file any time. We will only files with the “xml” extensions. P2A table is using “file id” of file table as foreign key.

5.1.5 Main_Table

Main Table is an epicenter of database. It will store information about all the P2I tables. Currently holds the information about P2I_heap and P2I_call. Whenever administrator need to change the schema it is important to add new tables in the database. All the newly created tables also need to add with main table.

5.1.6 P2I_analysis Setup (Example Table)

Since an example schema is being used in this testing phase, so there is only single information about analysis setup to store. But in the future administrator can add extra information in this part of the schema and data base is capable to adjust all the changes without any changes. Analysis setup will use “file id” as foreign key.

5.1.7 P2I_Analyzed Project (Example Table)

In this table information about the analyzed project will be stored. It will contain the project name but administrator can change this information when new schema will be introduced.
Extended information about analyzed project will be handled without any change in the data base or in the interface. Analyzed project will use “file id” as foreign key.

5.1.8 P2I_Analyzed Project Version (Example Table)

Analyzed project version is also a part of an introductory phase of analysis file. It is about the information of different analysis version on same Java file. Administrator can add some extra information to this part and it will be managed within the current database. Analyzed project version will use “file id” as foreign key.

5.1.9 P2I_heapObj (Example Table)

According to the current schema first point-to information is about heap object in the files. In this table all the information regarding the heap objects in the file will be stored. P2I heap is using “file id” of file table as foreign key and the heap id is being used in the P2I values tables.

5.1.10 P2I_Call (Example Table)

In the given P2A analysis file (which can be change in later) there are two types of P2I. One is heap and second is call. This table will store the call information. Call table will use the “fid” of file table as foreign key. Call table will contain three type of information. It will contain

- Target method
- Target object
- Return Value

5.1.11 How to Extend database

The data base of the system is extendable. Currently system is design to store two type of P2I. If administrator wants to add more P2I, it needs extends the data base first. In order to change schema administrator can add new P2I after the root element in the XML file. There is only one pre requisite that administrator need to create new table by using XML tag name. Administrator can extend by using following steps,

- Add new P2I in schema after the root element at any stage.
- Add new table in the database. Name of the P2I tag in XML and table in database must be same.
- New Table in the database must have following fields in the same order

<table>
<thead>
<tr>
<th>New_table_id</th>
<th>PK (int)</th>
</tr>
</thead>
<tbody>
<tr>
<td>File_id</td>
<td>FK (int)</td>
</tr>
<tr>
<td>New_P2I_data</td>
<td>(Varchar)</td>
</tr>
</tbody>
</table>

- Add this new table in the main table.
6 Conclusion and Feature enhancement

This section contains brief description of what we have done so far and what can be added in the future.

6.1 Conclusion

We have achieved our goal by doing following tasks:

Database

On backend MYSQL database is used to store the Java files and Point-to information from the analysis of the file using SAX parser. Database is flexible and it has capability to store different type of P2I without changing code.

Downloading and Comparison of the files

We provided an interface where user can download files, analysis of the files. Here user can compare two files at a time on the basis of the P2I of analysis files.

File Uploading

Through this feature system will get the Java files and analysis files from the user and store it to the file system.

Administrative System

We have designed an administrative system to make a check on the system. Administrator can upload the schema, manage the users and database

Interface

We have designed a user friendly web interface for interaction with the database.

Authentication system

For some operations like file uploading we build user authentication system. In this user will register itself with the system before getting access to full functionality.

6.2 Future work

We have met the entire requirement about the current domain of this project but still there is a possibility of enhancement and future. We suggest different aspects where project can be extend in future in the table.

1. We have work at the file level in this project it can be extend to the project level in the feature.
2. We have simply view the comparison of two files but it can be extended to the 3-D chart to visualize in very detailed mode.
3. We have use two file to compare at a time but this feature can be extended up to multiple file at the same time.
4. In this system the administrator is allowed to perform some standard functions but this functionality can be changed to more advance level.
5. We are using the Java file but in the future it can deal with all object oriented file.
6. In the future an interface can be designed for the administrator to add tables in the database.
7 References

7.1 XML basic and Schema


7.2 Related Studies

David hunter, 2000 Beginning XML
Erik Ray, Learning XML second edition