

A

Mini Project Report On

“PRIVACY POLICY INFERENCE OF USER-UPLOADED IMAGES”

Submitted to JNTUH in partial fulfilment of the
Requirements for the award of the Degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE & ENGINEERING

By

VADLURI ROHAN	21N61A0535
BOLISHETTI SRINIVAS	21N61A0548
ADEPU VARUN KUMAR	21N61A0552
VOOTKURI LAVAN KUMAR	21N61A0527

Under the Guidance of

Mrs. Thangallapally Sahithi

Assistant professor CSE Dept, VITS



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
VIVEKANANDA INSTITUTE OF TECHNOLOGY & SCIENCE

(Approved by AICTE New Delhi & Affiliated to JNTU, Hyderabad) An ISO 9001:2015 Certified Institution

KARIMNAGAR-505001

2021-2025

VIVEKANANDA INSTITUTE OF TECHNOLOGY & SCIENCE(N6)

(Approved by AICTE New Delhi & Affiliated to JNTU, Hyderabad)

An ISO 9001:2015 Certified Institution

KARIMNAGAR-505001

CERTIFICATE



This is to certify that the mini-project report titled “**PRIVACY POLICY INFERENCE OF USER-UPLOADED IMAGES**” is being submitted by 21N61A0535 VADLURI ROHAN, 21N61A0548 BOLISHETTI SRINIVAS, 21N61A0552 ADEPU VARUN KUMAR, 21N61A0527 VOOTKURI LAVAN KUMAR in B.Tech IV-I semester, Computer Science & Engineering is a record bonafide work carried out by them. The results embodied in this report have not been submitted to any other University for the award of any degree.

Internal Guide

Mrs. Thangalapally Sahithi

Assistant Professor

Head of the Department & Principal

Dr. Mula Veera Hanumantha Reddy

Dept of CSE

External Examiner

VIVEKANANDA INSTITUTE OF TECHNOLOGY & SCIENCE

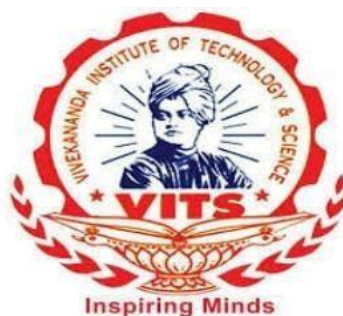
(Approved by AICTE New Delhi & Affiliated to JNTU, Hyderabad)

An ISO 9001:2015 Certified Institution

Karimnagar-505001

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION



I **VADLURI ROHAN** bearing Hall ticket no **21N61A0535** here by declare that the Project report entitled “**PRIVACY POLICY INFERENCE OF USER-UPLOADED IMAGES**” submitted in partial fulfillment of the requirements for the award of degree in B. Tech IV-I semester, Computer Science & Engineering This is a record bonafide work carried out by me. The results embodied in this report have not been submitted to any other University for the award of any degree or diploma.

VADLURI ROHAN

HTNO: -21N61A0535

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VIVEKANANDA INSTITUTE OF TECHNOLOGY & SCIENCE, Karimnagar.

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to our guide, **Mrs. Thangallapally Sahithi** Assistant Professor whose knowledge and guidance has motivated us to achieve goals we never thought possible. She has consistently been a source of motivation, encouragement, and inspiration. The time we have spent working under her supervision has truly been a pleasure.

We thank our H.O.D **Dr.M.V.Hanumantha Reddy** for his effort and indefatigable guidance rendered throughout the progress of project work. Thanks to programmers and nonteaching staff of CSE Department of VITS(N6).

We thank our Principal **Dr.M.V.Hanumantha Reddy** and Management for providing excellent facilities to carry out project work.

Finally, Special thanks to our parents for their support and encouragement throughout this course.

And thanks to our friends and well wishers for their constant support.

Name of Team Members	Roll No:
VADLURI ROHAN	21N61A0535
BOLISHETTI SRINIVAS	21N61A0548
ADEPU VARUN KUMAR	21N61A0552
VOOTKURI LAVAN KUMAR	21N61A0527

ABSTRACT

With the increasing volume of images users share through social sites, maintaining privacy has become a major problem, as demonstrated by a recent wave of publicized incidents where users inadvertently shared personal information. In light of these incidents, the need of tools to help users control access to their shared content is apparent. Toward addressing this need, we propose an Adaptive Privacy Policy Prediction (A3P) system to help users compose privacy settings for their images.

We examine the role of social context, image content, and metadata as possible indicators of users privacy preferences. We propose a two-level framework which according to the user's available history on the site, determines the best available privacy policy for the user's images being uploaded.

Our solution relies on an image classification framework for image categories which may be associated with similar policies, and on a policy prediction algorithm to automatically generate a policy for each newly uploaded image, also according to users' social features.

Over time, the generated policies will follow the evolution of users' privacy attitude. We provide the results of our extensive evaluation over 5,000 policies, which demonstrate the effectiveness of our system, with prediction accuracies over 90 percent.

INDEX

TITLE	PAGENO
1. INTRODUCTION	
1.1 privacy policy inference of user-uploaded images	1
1.2 Existing System	4
1.3 Disadvantages of Existing System	4
1.4 Proposed System	5
1.5 Advantages of Proposed System	5
2. PRELIMINARY INVESTIGATION	
2.1 Request Clarification	6
2.2 Request Approval	8
2.3 System Design	9
3. SYSTEM REQUIREMENTS SPECIFICATION	
3.1 Software Requirements	11
3.2 Hardware Requirements	
3.3 Feasibility Study	
4. DESIGN METHODOLOGY	
4.1 Architecture Diagram	13
4.2 Modules	14

4.3 Data Flow Diagram	16
4.4 UML Diagrams	17
5. SOFTWARE ENVIRONMENT	
5.1 Java Technology	23
5.2 ODBC	29
5.3 JDBC	31
5.4 Network Concept	33
5.5 Tomcat server	36
OUTPUTSCREENS	38
6.TESTINGSTRATEGY	53
1) System Testing	
2) Unit Testing	
3) Integration Testing	
4) User Acceptance Testing	
5) Output Testing	
6) Validation Testing	
7) User Training	
8) Maintenance	
CONCLUSION	61
REFERENCES	62

1.INTRODUCTION

IMAGES are now one of the key enablers of users' connectivity. Sharing takes place both among previously established groups of known people or social circles (e. g., Google+, Flickr or Picasa), and also increasingly with people outside the users social circles, for purposes of social discovery-to help them identify new peers and learn about peers interests and social surroundings. However, semantically rich images may reveal content sensitive information .

Consider a photo of a students 2012 graduation ceremony, for example. It could be shared within a Google+ circle or Flickr group, but may unnecessarily expose the students Apos family members and other friends. Sharing images within online content sharing sites, therefore, may quickly lead to unwanted disclosure and privacy violations [3], [24]. Further, the persistent nature of online media makes it possible for other users to collect rich aggregated information about the owner of the published content and the subjects in the published content [3], [20], [24].

The aggregated information can result in unexpected exposure of one's social environment and lead to abuse of one's personal information. Most content sharing websites allow users to enter their privacy preferences. Unfortunately, recent studies have shown that users struggle to set up and maintain such privacy settings [1], [11], [22], [33].

One of the main reasons provided is that given the amount of shared information this process can be tedious and error-prone. Therefore, many have acknowledged the need of policy recommendation systems which can assist users to easily and properly configure privacy settings [7], [22], [28], [30]. However, existing proposals for automating privacy settings appear to be inadequate to address the unique privacy needs of images [3], [5], [41], due to the amount of information implicitly carried within images, and their relationship with the online environment wherein they are exposed.

In this paper, we propose an Adaptive Privacy Policy Prediction (A3P) system which aims to provide users a hassle free privacy settings experience by automatically generating personalized policies. The A3P system handles user uploaded images, and factors in the following criteria that influence one's privacy settings of images: _ The impact of social environment and personal characteristics.

Social context of users, such as their profile information and relationships with others may provide useful information regarding users' privacy preferences. For example, users interested in

photography may like to share their photos with other amateur photographers. Users who have several family members among their social contacts may share with them pictures related to family events. However, using common policies across all users or across users with similar traits may be too simplistic and not satisfy individual preferences. Users may have drastically different opinions even on the same type of images.

For example, a privacy adverse person may be willing to share all his personal images while a more conservative person may just want to share personal images with his family members.

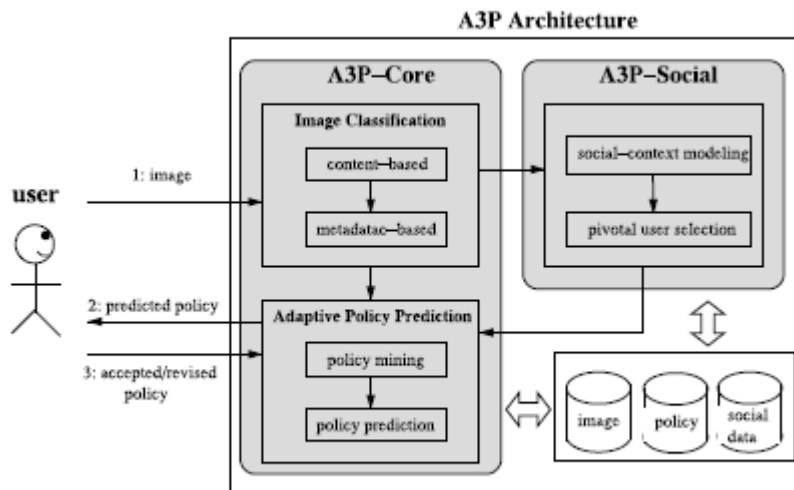


Fig. 1. System overview.

In light of these considerations, it is important to find the balancing point between the impact of social environment and users' individual characteristics in order to predict the policies that match each individual's needs. Moreover, individuals may change their overall attitude toward privacy as time passes. In order to develop a personalized policy recommendation system, such changes on privacy opinions should be carefully considered.

The role of image's content and metadata. In general, similar images often incur similar privacy preferences, especially when people appear in the images. For example, one may upload several photos of his kids and specify that only his family members are allowed to see these photos. He may upload some other photos of landscapes which he took as a hobby and for these photos, he may set privacy preference allowing anyone to view and comment the photos. Analyzing the visual content may not be sufficient to capture users' privacy preferences.

Tags and other metadata are indicative of the social context of the image, including where it was taken and why [4], and also provide a synthetic description of images, complementing the

information obtained from visual content analysis. Corresponding to the aforementioned two criteria, the proposed A3P system is comprised of two main building blocks (as shown in Fig. 1): A3P-Social and A3P-Core. The

A3P-core focuses on analyzing each individual user's own images and metadata, while the A3P-Social offers a community perspective of privacy setting recommendations for a user's potential privacy improvement. We design the interaction flows between the two building blocks to balance the benefits from meeting personal characteristics and obtaining community advice.

1.2 EXISTING SYSTEM

- ❖ Most content sharing websites allow users to enter their privacy preferences. Unfortunately, recent studies have shown that users struggle to set up and maintain such privacy settings.
- ❖ One of the main reasons provided is that given the amount of shared information this process can be tedious and error-prone. Therefore, many have acknowledged the need of policy recommendation systems which can assist users to easily and properly configure privacy settings.

1.3 DISADVANTAGES OF EXISTING SYSTEM

- ❖ Sharing images within online content sharing sites, therefore, may quickly lead to unwanted disclosure and privacy violations.
- ❖ Further, the persistent nature of online media makes it possible for other users to collect rich aggregated information about the owner of the published content and the subjects in the published content.
- ❖ The aggregated information can result in unexpected exposure of one's social environment and lead to abuse of one's personal information.

1.4 PROPOSED SYSTEM

- In this paper, we propose an Adaptive Privacy Policy Prediction (A3P) system which aims to provide users a hassle free privacy settings experience by automatically generating personalized policies. The A3P system handles user uploaded images, and factors in the following criteria that influence one's privacy settings of images:
- The impact of social environment and personal characteristics. Social context of users, such as their profile information and relationships with others may provide useful information regarding users' privacy preferences. For example, users interested in photography may like to share their photos with other amateur photographers.
- The role of image's content and metadata. In general, similar images often incur similar privacy preferences, especially when people appear in the images. For example, one may upload several photos of his kids and specify that only his family members are allowed to see these photos.

1.4 ADVANTAGES OF PROPOSED SYSTEM

The A3P-core focuses on analyzing each individual user's own images and metadata, while the A3P-Social offers a community perspective of privacy setting recommendations for a user's potential privacy improvement. We design the interaction flows between the two building blocks to balance the benefits from meeting personal characteristics and obtaining community advice

2.PRELIMINARY INVESTIGATION

The first and foremost strategy for development of a project starts from the thought of designing a mail enabled platform for a small firm in which it is easy and convenient of sending and receiving messages, there is a search engine ,address book and also including some entertaining games. When it is approved by the organization and our project guide the first activity, ie. preliminary investigation begins. The activity has three parts:

- **Request Clarification**
- **Feasibility Study**
- **Request Approval**

2.1 REQUEST CLARIFICATION

After the approval of the request to the organization and project guide, with an investigation being considered, the project request must be examined to determine precisely what the system requires.

Here our project is basically meant for users within the company whose systems can be interconnected by the Local Area Network(LAN). In today's busy schedule man need everything should be provided in a readymade manner. So taking into consideration of the vastly use of the net in day to day life, the corresponding development of the portal came into existence.

FEASIBILITY ANALYSIS

An important outcome of preliminary investigation is the determination that the system request is feasible. This is possible only if it is feasible within limited resource and time. The different feasibilities that have to be analyzed are

- **Operational Feasibility**
- **Economic Feasibility**
- **Technical Feasibility**
- **Operational Feasibility**

Operational Feasibility deals with the study of prospects of the system to be developed. This system operationally eliminates all the tensions of the Admin and helps him in effectively tracking the project progress. This kind of automation will surely reduce the time and energy, which previously consumed in manual work. Based on the study, the system is proved to be operationally feasible.

Economic Feasibility

Economic Feasibility or Cost-benefit is an assessment of the economic justification for a computer based project. As hardware was installed from the beginning & for lots of purposes thus the cost on project of hardware is low. Since the system is a network based, any number of employees connected to the LAN within that organization can use this tool from at anytime. The Virtual Private Network is to be developed using the existing resources of the organization. So the project is economically feasible.

Technical Feasibility

According to Roger S. Pressman, Technical Feasibility is the assessment of the technical resources of the organization. The organization needs IBM compatible machines with a graphical web browser connected to the Internet and Intranet. The system is developed for platform Independent environment. Java Server Pages, JavaScript, HTML, SQL server and WebLogic Server are used to develop the system. The technical feasibility has been carried out. The system is technically feasible for development and can be developed with the existing facility.

2.2 REQUEST APPROVAL

Not all request projects are desirable or feasible. Some organization receives so many project requests from client users that only few of them are pursued. However, those projects that are both feasible and desirable should be put into schedule. After a project request is approved, its cost, priority, completion time and personnel requirement is estimated and used to determine where to add it to any project list. Truly speaking, the approval of those above factors, development works can be launched.

2.3 SYSTEM DESIGN AND DEVELOPMENT

INPUT DESIGN

Input Design plays a vital role in the life cycle of software development, it requires very careful attention of developers. The input design is to feed data to the application as accurate as possible. So inputs are supposed to be designed effectively so that the errors occurring while feeding are minimized. According to Software Engineering Concepts, the input forms or screens are designed to provide to have a validation control over the input limit, range and other related validations.

This system has input screens in almost all the modules. Error messages are developed to alert the user whenever he commits some mistakes and guides him in the right way so that invalid entries are not made. Let us see deeply about this under module design

. Input design is the process of converting the user created input into a computer-based format. The goal of the input design is to make the data entry logical and free from errors. The error is in the input are controlled by the input design. The application has been developed in user-friendly manner. The forms have been designed in such a way during the processing the cursor is placed in the position where must be entered. The user is also provided with in an option to select an appropriate input from various alternatives related to the field in certain cases.

Validations are required for each data entered. Whenever a user enters an erroneous data, error message is displayed and the user can move on to the subsequent pages after completing all the entries in the current page.

OUTPUT DESIGN

The Output from the computer is required to mainly create an efficient method of communication within the company primarily among the project leader and his team members, in other words, the administrator and the clients. The output of VPN is the system which allows the project leader to manage his clients in terms of creating new clients and assigning new projects to them, maintaining a record of the project validity and providing folder level access to each client on the user side depending on the projects allotted to him. After completion of a project, a new project may be assigned to the client. User authentication procedures are maintained at the initial stages itself. A new user may be created by the administrator himself or a user can himself register as a new user but the task of assigning projects and validating a new user rests with the administrator only.

The application starts running when it is executed for the first time. The server has to be started and then the internet explorer is used as the browser. The project will run on the local area network so the server machine will serve as the administrator while the other connected systems can act as the clients. The developed system is highly user

3. SYSTEM REQUIREMENTS SPECIFICATION

3.1 SOFTWARE REQUIREMENTS

- Operating system : Windows Family.
- Coding Language : J2EE (JSP, Servlet, Java Bean)
- Data Base : MS Access.
- Web Server : Tomcat 6.0

3.2 HARDWARE REQUIREMENTS

- System : Pentium IV 3.4 GHz.
- Hard Disk : 40 GB.
- Floppy Drive : 1.44 Mb.
- Monitor : 14' Colour Monitor.
- Mouse : Optical Mouse.
- Ram : 1 GB.

3.3 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

- ◆ ECONOMICAL FEASIBILITY
- ◆ TECHNICAL FEASIBILITY
- ◆ SOCIAL FEASIBILITY

ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

TECHNICAL FEASIBILITY

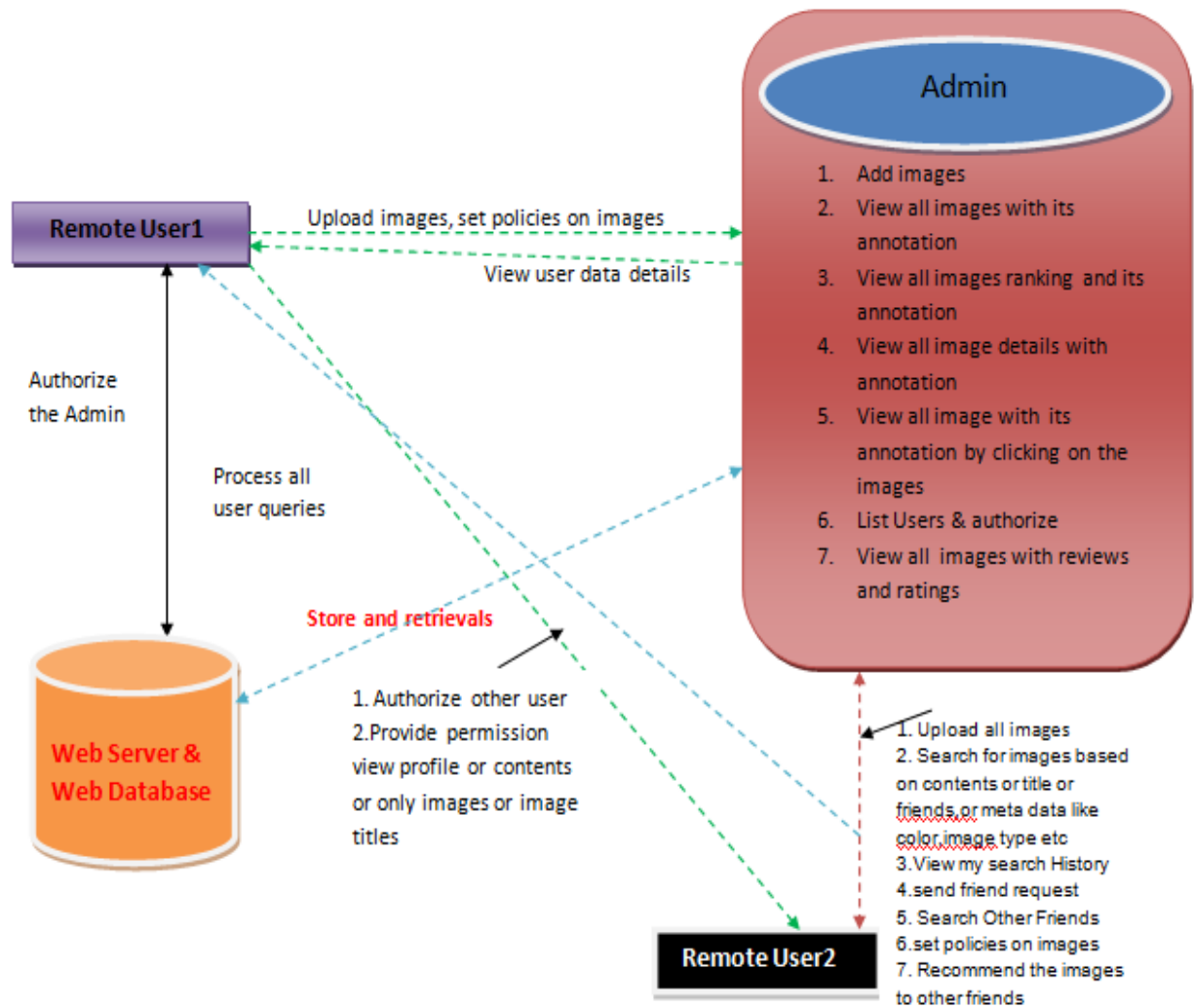
This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

4.DESIGN METHODOLOGY

4.1 ARCHITECTURE DIAGRAM



4.2 MODULES

- ❖ System Construction Module
- ❖ Content-Based Classification
- ❖ Metadata-Based Classification
- ❖ Adaptive Policy Prediction

MODULES DESCRIPTION:

System Construction Module

The A3P system consists of two main components: A3P-core and A3P-social. The overall data flow is the following. When a user uploads an image, the image will be first sent to the A3P-core. The A3P-core classifies the image and determines whether there is a need to invoke the A3P-social. In most cases, the A3P-core predicts policies for the users directly based on their historical behavior. If one of the following two cases is verified true, A3P-core will invoke A3Psocial: (i) The user does not have enough data for the type of the uploaded image to conduct policy prediction; (ii) The A3P-core detects the recent major changes among the user's community about their privacy practices along with user's increase of social networking activities (addition of new friends, new posts on one's profile etc).

Content-Based Classification

To obtain groups of images that may be associated with similar privacy preferences, we propose a hierarchical image classification which classifies images first based on their contents and then refine each category into subcategories based on their metadata. Images that do not have metadata will be grouped only by content. Such a hierarchical classification gives a higher priority to image content and minimizes the influence of missing tags. Note that it is possible that some images are included in multiple categories as long as they contain the typical content features or metadata of those categories. Our approach to content-based classification is based on an efficient and yet accurate image similarity approach. Specifically, our classification algorithm compares image signatures defined based on quantified and sanitized version of Haar wavelet transformation. For each image, the wavelet transform encodes frequency and spatial information related to image color, size, invariant transform, shape, texture, symmetry, etc. Then, a small number of coefficients are selected to form the signature

of the image. The content similarity among images is then determined by the distance among their image signatures.

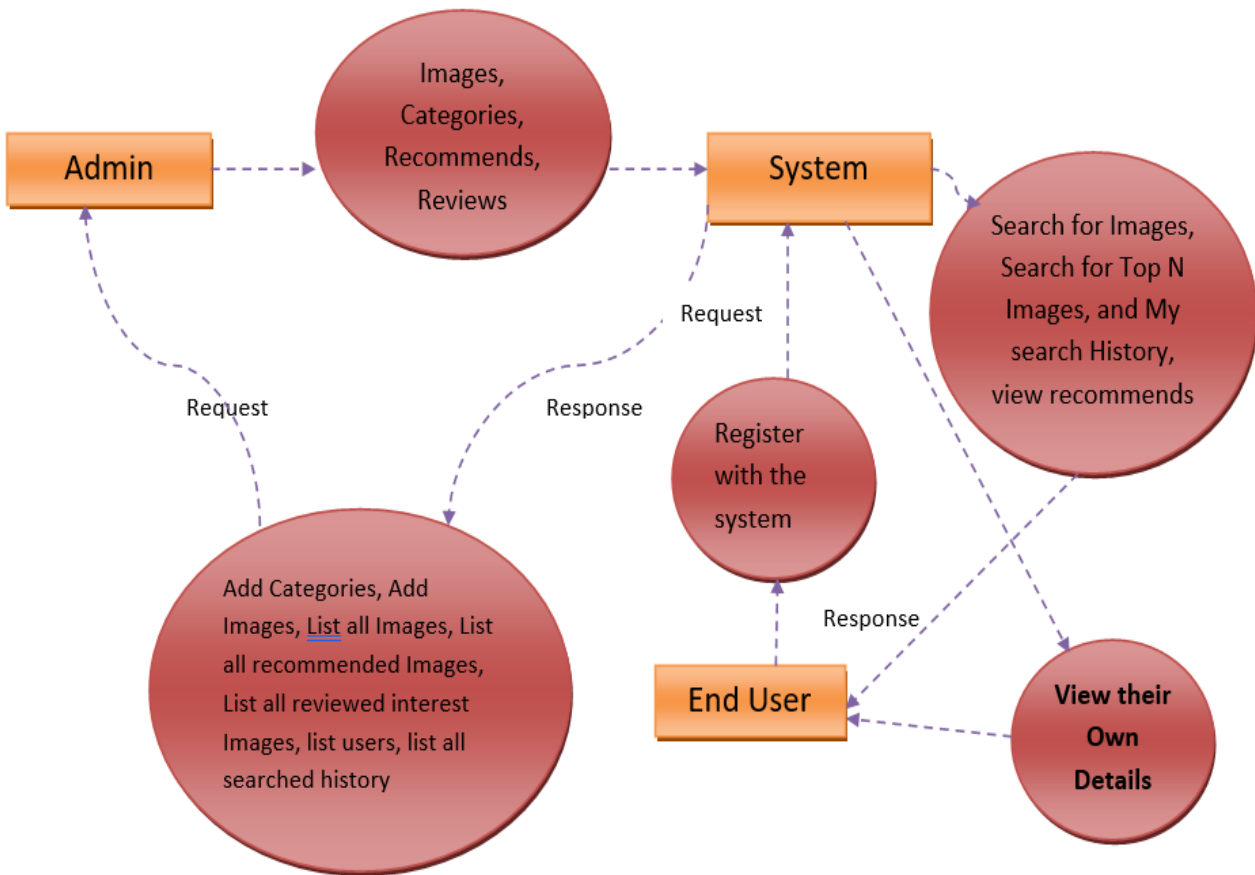
Metadata-Based Classification

The metadata-based classification groups images into subcategories under aforementioned baseline categories. The process consists of three main steps. The first step is to extract keywords from the metadata associated with an image. The metadata considered in our work are tags, captions, and comments. The second step is to derive a representative hypernym (denoted as h) from each metadata vector. The third step is to find a subcategory that an image belongs to. This is an incremental procedure. At the beginning, the first image forms a subcategory as itself and the representative hypernyms of the image becomes the subcategory's representative hypernyms.

Adaptive Policy Prediction

The policy prediction algorithm provides a predicted policy of a newly uploaded image to the user for his/her reference. More importantly, the predicted policy will reflect the possible changes of a user's privacy concerns. The prediction process consists of three main phases: (i) policy normalization; (ii) policy mining; and (iii) policy prediction

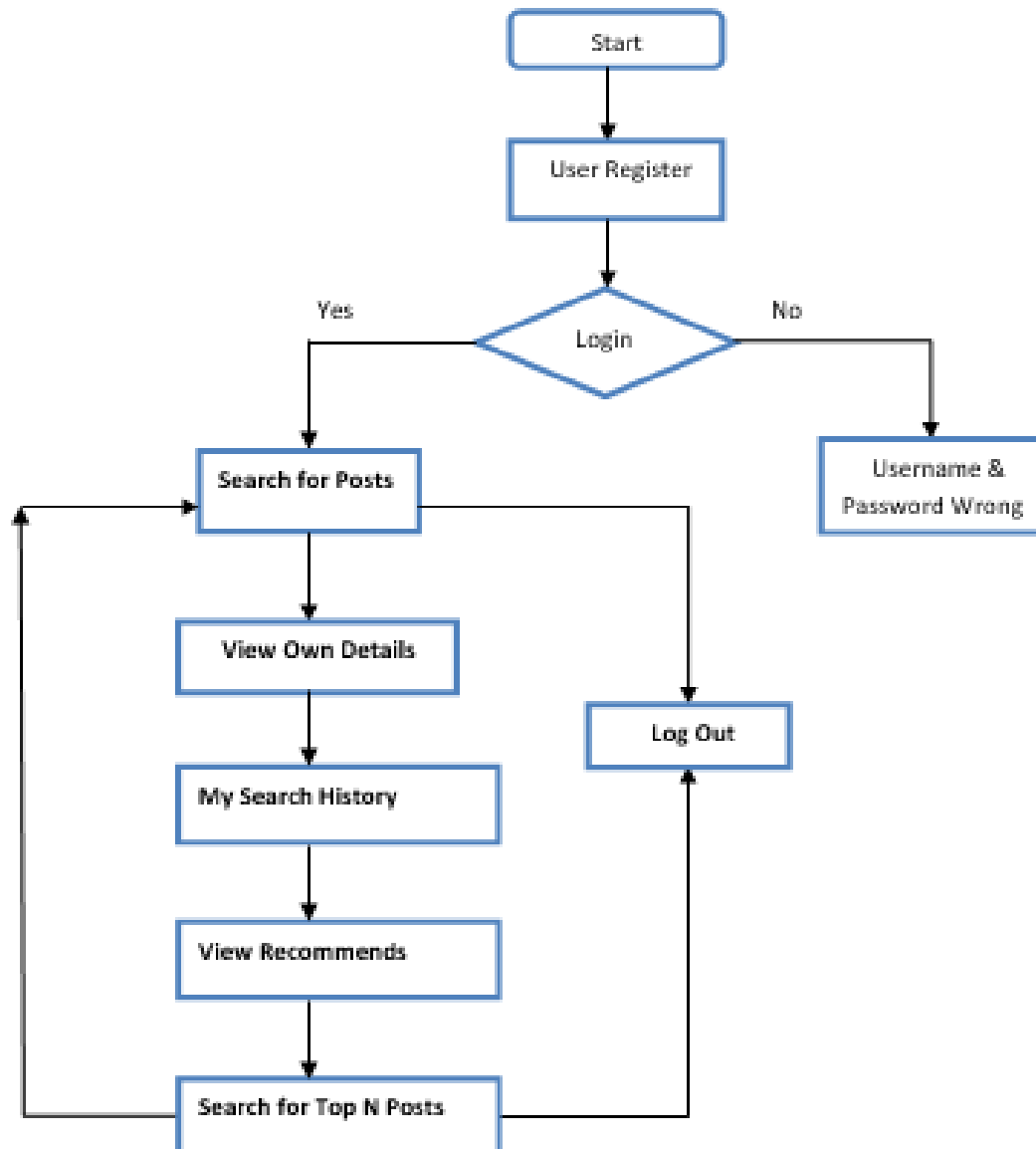
4.3 DATA FLOW DIAGRAM



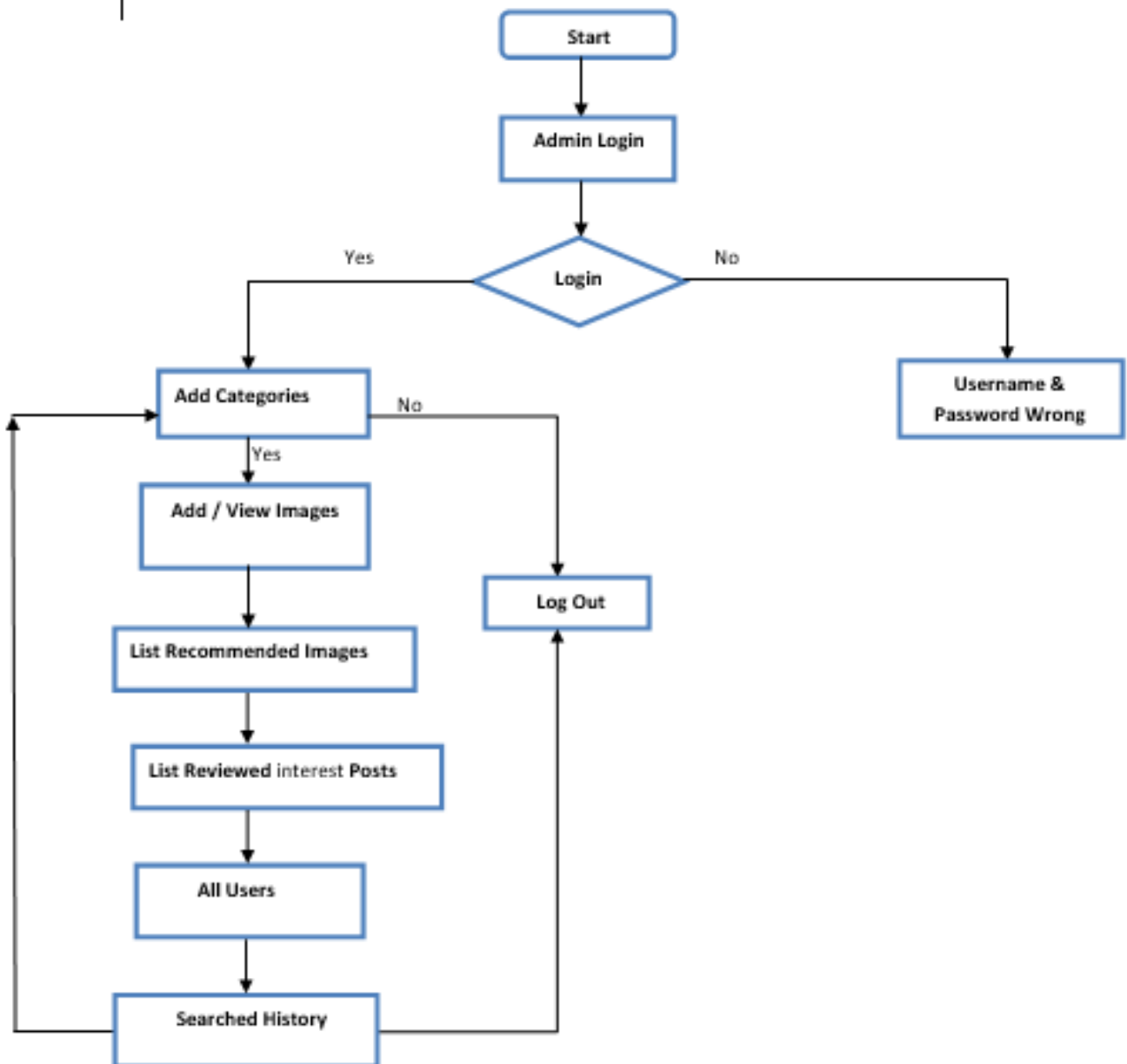
4.4 UML DIAGRAMS

Flow Charts

Flow Chart1: User

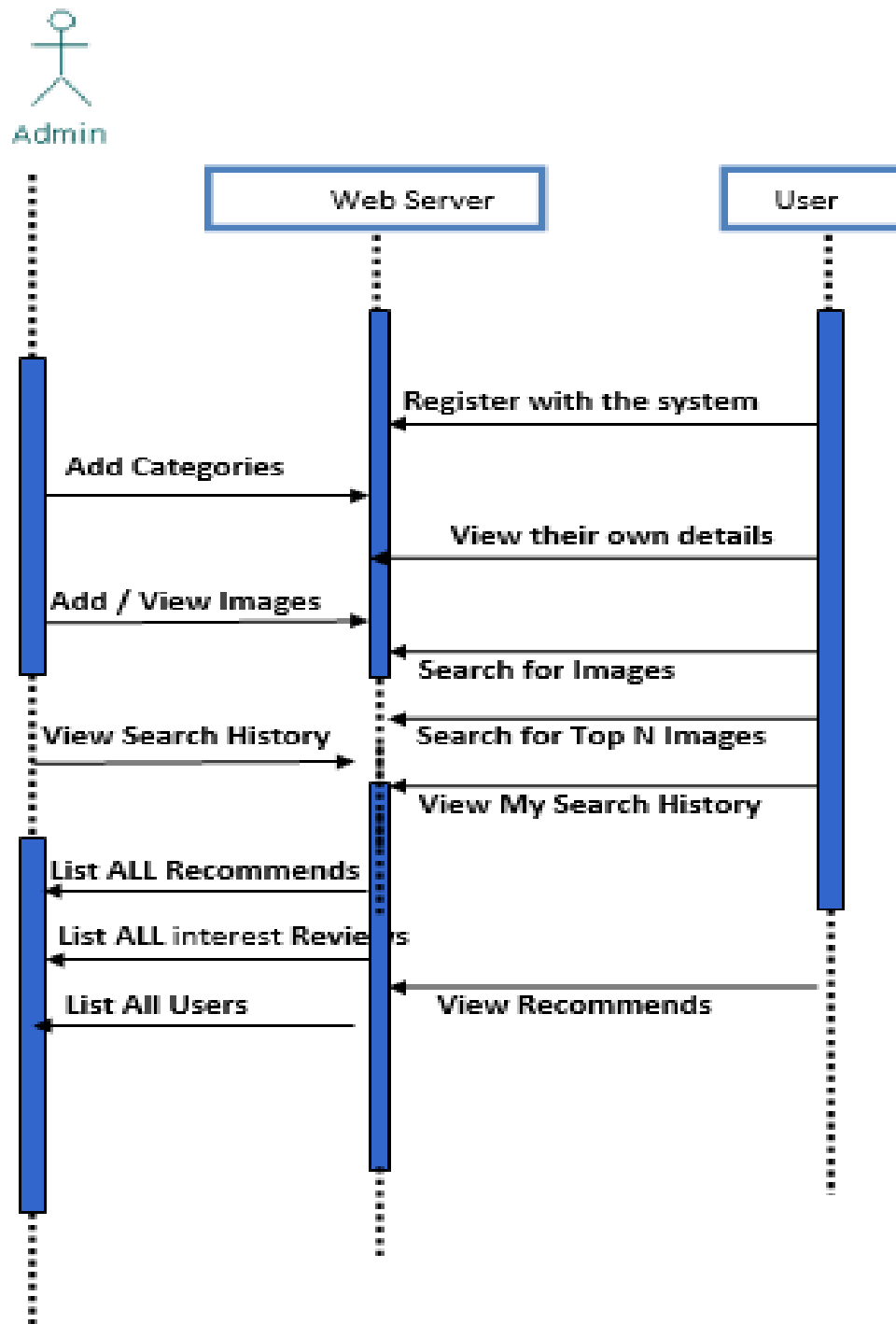


Flow Chart 2: Admin

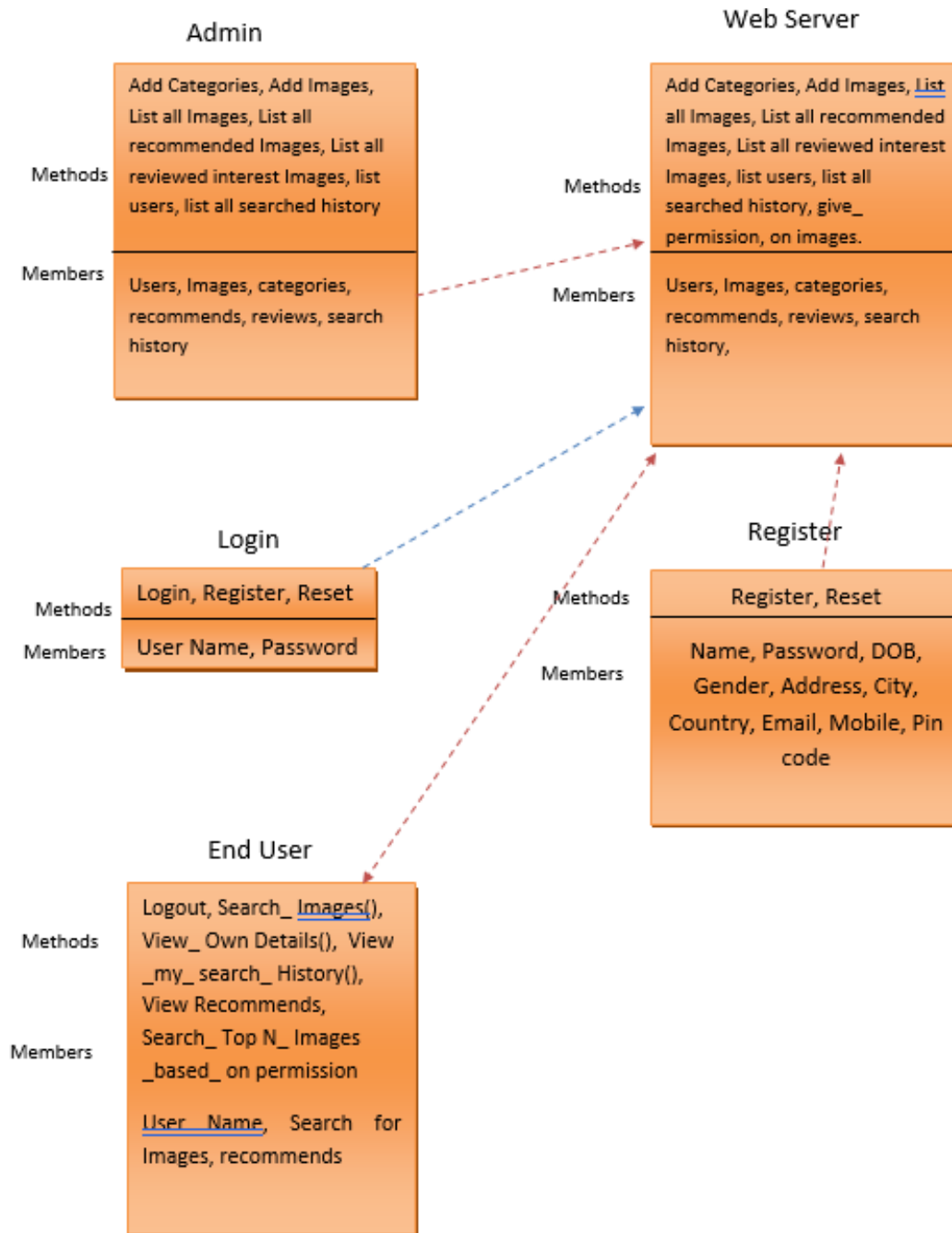


SEQUENCE DIAGRAM

Sequence Diagram:



CLASS DIAGRAM



The class diagram is the main building block of [object oriented](#) modeling. It is used both for general [conceptual modeling](#) of the systematic of the application, and for detailed modeling translating the models into [programming code](#). Class diagrams can also be used for modeling. The classes in a

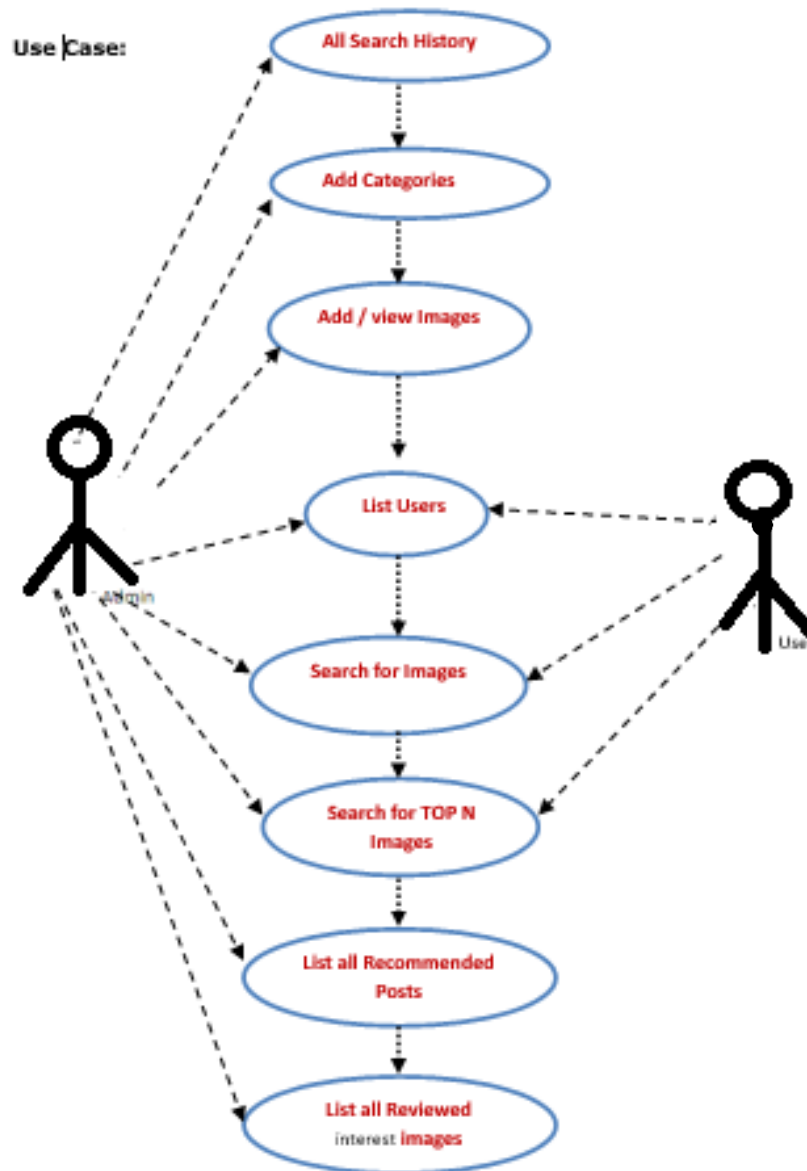
class diagram represent both the main objects, interactions in the application and the classes to be programmed.

In the diagram, classes are represented with boxes which contain three parts

- The upper part holds the name of the class
- The middle part contains the attributes of the class
- The bottom part gives the methods or operations the class can take or undertake

In the design of a system, a number of classes are identified and grouped together in a class diagram which helps to determine the static relations between those objects. With detailed modeling, the classes of the conceptual design are often split into a number of subclasses.

USE CASE DIAGRAM



5. SOFTWARE ENVIRONMENT

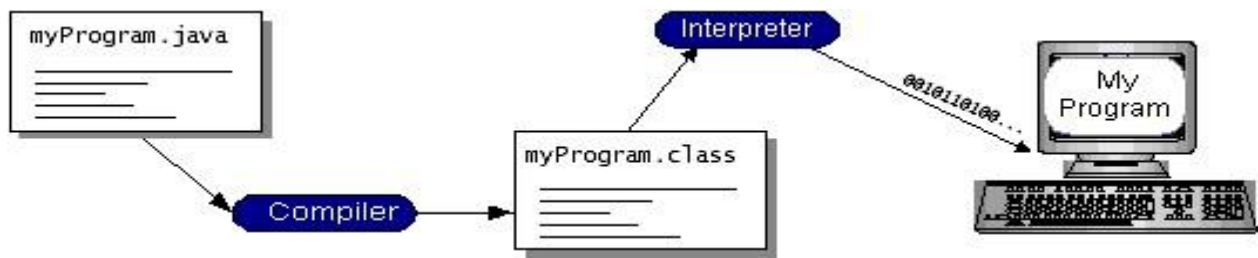
5.1 JAVA TECHNOLOGY

Java technology is both a programming language and a platform.

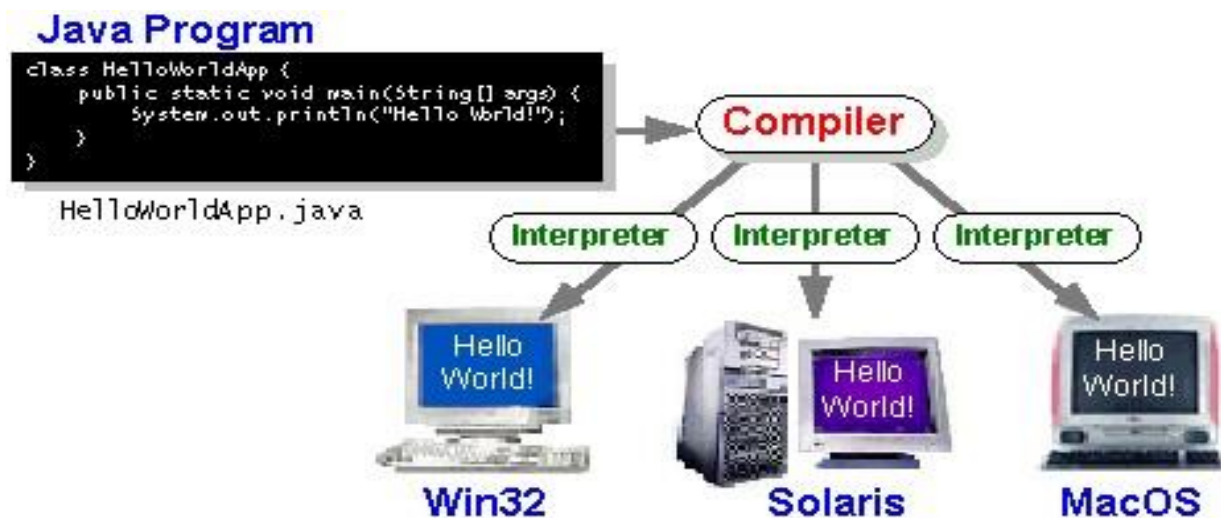
The Java programming language is a high-level language that can be characterized by all of the following buzzwords:

- Simple
- Architecture neutral
- Object oriented
- Portable
- Distributed
- High performance
- Interpreted
- Multithreaded
- Robust
- Dynamic
- Secure

With most programming languages, you either compile or interpret a program so that you can run it on your computer. The Java programming language is unusual in that a program is both compiled and interpreted. With the compiler, first you translate a program into an intermediate language called Java byte codes —the platform-independent codes interpreted by the interpreter on the Java platform. The interpreter parses and runs each Java byte code instruction on the computer. Compilation happens just once; interpretation occurs each time the program is executed. The following figure illustrates how this works.



You can think of Java byte codes as the machine code instructions for the Java Virtual Machine (Java VM). Every Java interpreter, whether it's a development tool or a Web browser that can run applets, is an implementation of the Java VM. Java byte codes help make “write once, run anywhere” possible. You can compile your program into byte codes on any platform that has a Java compiler. The byte codes can then be run on any implementation of the Java VM. That means that as long as a computer has a Java VM, the same program written in the Java programming language can run on Windows 2000, a Solaris workstation, or on an iMac.



The Java Platform

A platform is the hardware or software environment in which a program runs. We've already mentioned some of the most popular platforms like Windows 2000, Linux, Solaris, and MacOS. Most platforms can be described as a combination of the operating system and hardware. The Java platform differs from most other platforms in that it's a software-only platform that runs on top of other hardware-based platforms.

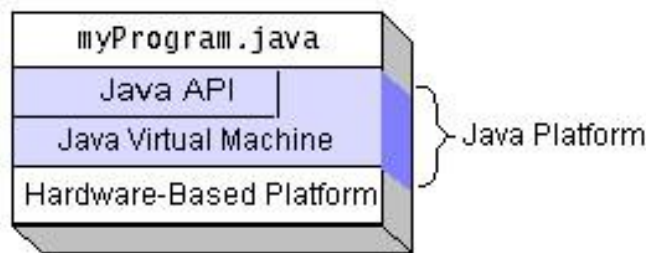
The Java platform has two components:

- **The Java Virtual Machine (Java VM)**
- **The Java Application Programming Interface (Java API)**

You've already been introduced to the Java VM. It's the base for the Java platform and is ported onto various hardware-based platforms.

The Java API is a large collection of ready-made software components that provide many useful capabilities, such as graphical user interface (GUI) widgets. The Java API is grouped into libraries of related classes and interfaces; these libraries are known as packages.

The following figure depicts a program that's running on the Java platform. As the figure shows, the Java API and the virtual machine insulate the program from the hardware.



Native code is code that after you compile it, the compiled code runs on a specific hardware platform. As a platform-independent environment, the Java platform can be a bit slower than native code. However, smart compilers, well-tuned interpreters, and just-in-time byte code compilers can bring performance close to that of native code without threatening portability.

What Can Java Technology Do?

The most common types of programs written in the Java programming language are applets and applications. If you've surfed the Web, you're probably already familiar with applets. An applet is a program that adheres to certain conventions that allow it to run within a Java-enabled browser.

However, the Java programming language is not just for writing cute, entertaining applets for the Web. The general-purpose, high-level Java programming language is also a powerful software platform. Using the generous API, you can write many types of programs.

An application is a standalone program that runs directly on the Java platform. A special kind of application known as a server serves and supports clients on a network. Examples of servers are Web servers, proxy servers, mail servers, and print servers. Another specialized program is a servlet. A servlet can almost be thought of as an applet that runs on the server side. Java Servlets are a popular choice for building interactive web applications, replacing the use of CGI scripts. Servlets are similar to applets in that they are runtime extensions of applications. Instead of working in browsers, though, servlets run within Java Web servers, configuring or tailoring the server.

Every full implementation of the Java platform gives you the following features:

•**The essentials:**

Objects, strings, threads, numbers, input and output, data structures, system properties, date and time, and so on.

•**Applets:**

The set of conventions used by applets.

•**Networking:**

URLs, TCP (Transmission Control Protocol), UDP (User Datagram Protocol) sockets, and IP (Internet Protocol) addresses.

•**Internationalization:**

Help for writing programs that can be localized for users worldwide. Programs can automatically adapt to specific locales and be displayed in the appropriate language.

•**Security:**

Both low level and high level, including electronic signatures, public and private key management, access control, and certificates.

•**Software components:**

Known as JavaBeans, can plug into existing component architectures.

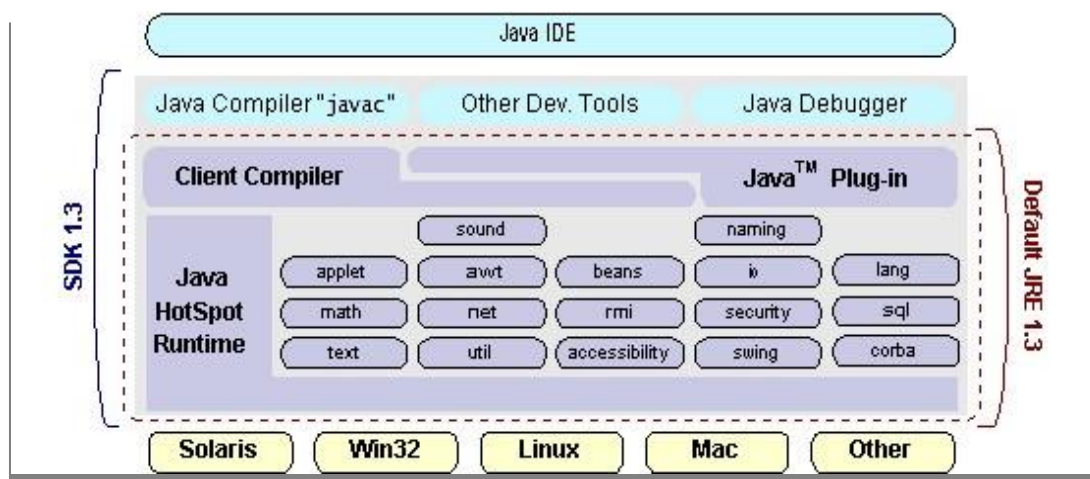
- Object serialization:**

Allows lightweight persistence and communication via Remote Method Invocation (RMI).

- Java Database Connectivity (JDBCTM):**

Provides uniform access to a wide range of relational databases.

The Java platform also has APIs for 2D and 3D graphics, accessibility, servers, collaboration, telephony, speech, animation, and more. The following figure depicts what is included in the Java 2 SDK.



How Will Java Technology Change My Life?

We can't promise you fame, fortune, or even a job if you learn the Java programming language. Still, it is likely to make your programs better and requires less effort than other languages. We believe that Java technology will help you do the following:

- Get started quickly:**

Although the Java programming language is a powerful object-oriented language, it's easy to learn, especially for programmers already familiar with C or C++.

- Write less code:**

Comparisons of program metrics (class counts, method counts, and so on) suggest that a program written in the Java programming language can be four times smaller than the same program in C++.

- Write better code:**

The Java programming language encourages good coding practices, and its garbage collection helps you avoid memory leaks. Its object orientation, its JavaBeans component architecture, and its wide-ranging, easily extendible API let you reuse other people's tested code and introduce fewer bugs.

- Develop programs more quickly:**

Your development time may be as much as twice as fast versus writing the same program in C++.

Why? You write fewer lines of code and it is a simpler programming language than C++.

- Avoid platform dependencies with 100% Pure Java:**

You can keep your program portable by avoiding the use of libraries written in other languages. The 100% Pure Java TM Product Certification Program has a repository of historical process manuals, white papers, brochures, and similar materials online.

- Write once, run anywhere:**

Because 100% Pure Java programs are compiled into machine-independent byte codes, they run consistently on any Java platform.

5.2 ODBC

Microsoft Open Database Connectivity (ODBC) is a standard programming interface for application developers and database systems providers. Before ODBC became a *de facto* standard for Windows programs to interface with database systems, programmers had to use proprietary languages for each database they wanted to connect to. Now, ODBC has made the choice of the database system almost irrelevant from a coding perspective, which is as it should be. Application developers have much more important things to worry about than the syntax that is needed to port their program from one database to another when business needs suddenly change.

Through the ODBC Administrator in Control Panel, you can specify the particular database that is associated with a data source that an ODBC application program is written to use. Think of an ODBC data source as a door with a name on it. Each door will lead you to a particular database. For example, the data source named Sales Figures might be a SQL Server database, whereas the Accounts Payable data source could refer to an Access database. The physical database referred to by a data source can reside anywhere on the LAN. The ODBC system files are not installed on your system by Windows 95. Rather, they are installed when you setup a separate database application, such as SQL Server Client or Visual Basic 4.0.

When the ODBC icon is installed in Control Panel, it uses a file called ODBCINST.DLL. It is also possible to administer your ODBC data sources through a stand-alone program called ODBCADM.EXE.

From a programming perspective, the beauty of ODBC is that the application can be written to use the same set of function calls to interface with any data source, regardless of the database vendor.

The source code of the application doesn't change whether it talks to Oracle or SQL Server. We only mention these two as an example. There are ODBC drivers available for several dozen popular database systems. Even Excel spreadsheets and plain text files can be turned into data sources. The operating system uses the Registry information written by ODBC Administrator to determine which low-level ODBC drivers are needed to talk to the data source (such as the interface to Oracle or SQL Server). The loading of the ODBC drivers is transparent to the ODBC application program. In a client/server environment, the ODBC API even handles many of the network issues for the application programmer.

The advantages of this scheme are so numerous that you are probably thinking there must be some catch. The only disadvantage of ODBC is that it isn't as efficient as talking directly to the native database interface. ODBC has had many detractors make the charge that it is too slow.

Microsoft has always claimed that the critical factor in performance is the quality of the driver software that is used. In our humble opinion, this is true. The availability of good ODBC drivers has improved a great deal recently. And anyway, the criticism about performance is somewhat analogous to those who said that compilers would never match the speed of pure assembly language. Maybe not, but the compiler (or ODBC) gives you the opportunity to write cleaner programs, which means you finish sooner. Meanwhile, computers get faster every year.

5.3 JDBC

In an effort to set an independent database standard API for Java; Sun Microsystems developed Java Database Connectivity, or JDBC. JDBC offers a generic SQL database access mechanism that provides a consistent interface to a variety of RDBMSs. This consistent interface is achieved through the use of “plug-in” database connectivity modules, or drivers.

If a database vendor wishes to have JDBC support, he or she must provide the driver for each platform that the database and Java run on.

To gain a wider acceptance of JDBC, Sun based JDBC’s framework on ODBC. As you discovered earlier in this chapter, ODBC has widespread support on a variety of platforms. Basing JDBC on ODBC will allow vendors to bring JDBC drivers to market much faster than developing a completely new connectivity solution.

JDBC was announced in March of 1996. It was released for a 90 day public review that ended June 8, 1996. Because of user input, the final JDBC v1.0 specification was released soon after. The remainder of this section will cover enough information about JDBC for you to know what it is about and how to use it effectively. This is by no means a complete overview of JDBC. That would fill an entire book.

JDBC Goals

Few software packages are designed without goals in mind. JDBC is one that, because of its many goals, drove the development of the API. These goals, in conjunction with early reviewer feedback, have finalized the JDBC class library into a solid framework for building database applications in Java.

The seven design goals for JDBC are as follows:

1. SQL Level API

The designers felt that their main goal was to define a SQL interface for Java. Although not the lowest database interface level possible, it is at a low enough level for higher-level tools and APIs to be created. Conversely, it is at a high enough level for application programmers to use it confidently. Attaining this goal allows for future tool vendors to “generate” JDBC code and to hide many of JDBC’s complexities from the end user.

2. SQL Conformance

SQL syntax varies as you move from database vendor to database vendor. In an effort to support a wide variety of vendors, JDBC will allow any query statement to be passed through it to the underlying database driver. This allows the connectivity module to handle nonstandard functionality in a manner that is suitable for its users.

3. JDBC must be implemental on top of common database interfaces

The JDBC SQL API must “sit” on top of other common SQL level APIs. This goal allows JDBC to use existing ODBC level drivers by the use of a software interface. This interface would translate JDBC calls to ODBC and vice versa.

4. Provide a Java interface that is consistent with the rest of the Java system

Because of Java’s acceptance in the user community thus far, the designers feel that they should not stray from the current design of the core Java system.

5. Keep it simple

This goal probably appears in all software design goal listings. JDBC is no exception. Sun felt that the design of JDBC should be very simple, allowing for only one method of completing a task per mechanism. Allowing duplicate functionality only serves to confuse the users of the API.

6. Use strong, static typing wherever possible

Strong typing allows for more error checking to be done at compile time; also, less error appear at runtime.

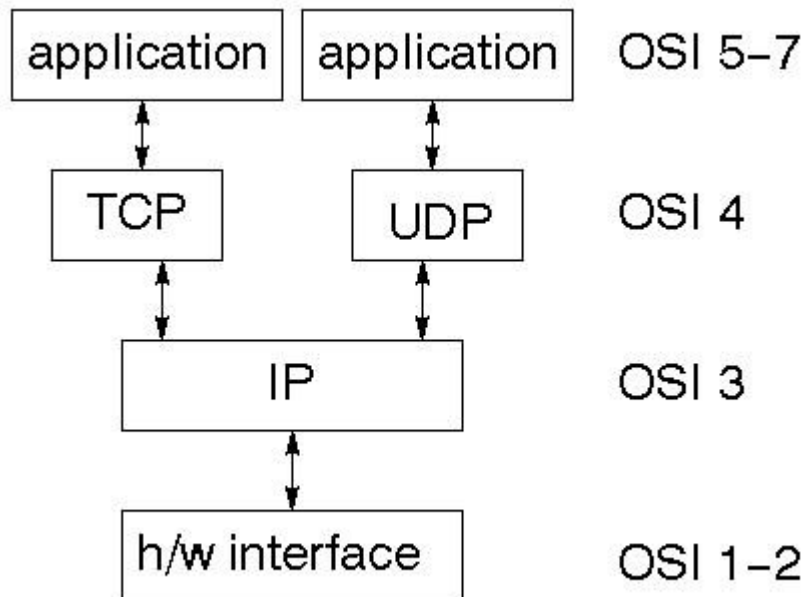
7. Keep the common cases simple

Because more often than not, the usual SQL calls used by the programmer are simple SELECT’s, INSERT’s, DELETE’s and UPDATE’s, these queries should be simple to perform with JDBC. However, more complex SQL statements should also be possible. Finally, we decided to proceed the implementation using JavaNetworking. And for dynamically updating the cache table we go for MSAccess database.

5.4 NETWORK CONCEPTS

TCP/IP stack

The TCP/IP stack is shorter than the OSI one:



TCP is a connection-oriented protocol; UDP (User Datagram Protocol) is a connectionless protocol.

IP datagram's

The IP layer provides a connectionless and unreliable delivery system. It considers each datagram independently of the others. Any association between datagram must be supplied by the higher layers. The IP layer supplies a checksum that includes its own header. The header includes the source and destination addresses. The IP layer handles routing through an Internet. It is also responsible for breaking up large datagram into smaller ones for transmission and reassembling them at the other end.

UDP

UDP is also connectionless and unreliable. What it adds to IP is a checksum for the contents of the datagram and port numbers. These are used to give a client/server model - see later.

TCP

TCP supplies logic to give a reliable connection-oriented protocol above IP. It provides a virtual circuit that two processes can use to communicate.

Internet addresses

In order to use a service, you must be able to find it. The Internet uses an address scheme for machines so that they can be located. The address is a 32 bit integer which gives the IP address. This encodes a network ID and more addressing. The network ID falls into various classes according to the size of the network address.

Network address

Class A uses 8 bits for the network address with 24 bits left over for other addressing. Class B uses 16 bit network addressing. Class C uses 24 bit network addressing and class D uses all 32.

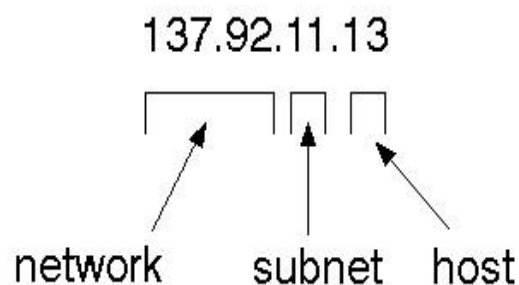
Subnet address

Internally, the UNIX network is divided into sub networks. Building 11 is currently on one sub network and uses 10-bit addressing, allowing 1024 different hosts

.Host address

8 bits are finally used for host addresses within our subnet. This places a limit of 256 machines that can be on the subnet.

Total address



Port addresses

A service exists on a host, and is identified by its port. This is a 16 bit number. To send a message to a server, you send it to the port for that service of the host that it is running on. This is not location transparency! Certain of these ports are "well known".

Sockets

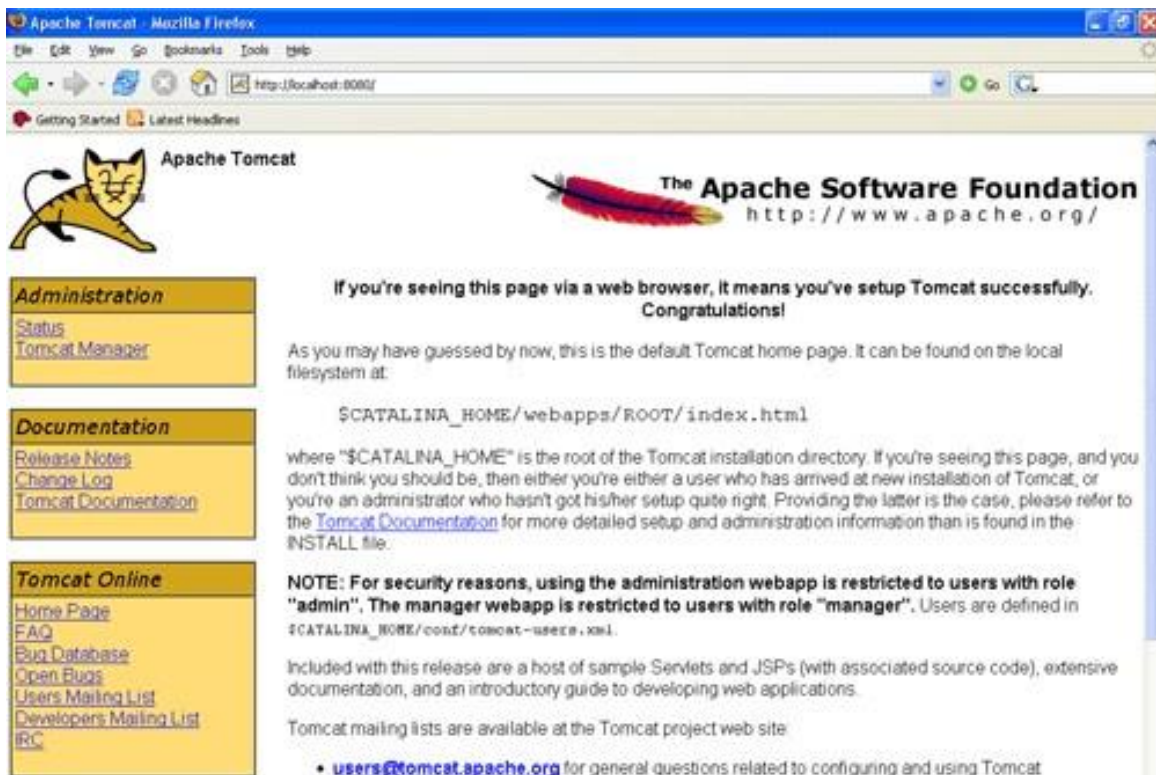
A socket is a data structure maintained by the system to handle network connections. A socket is created using the call `socket`. It returns an integer that is like a file descriptor. In fact, under Windows, this handle can be used with Read File and Write File functions.

```
#include    <sys/types.h>
#include <sys/socket.h>
int socket (int family, int type, int protocol);
```

Here "family" will be `AF_INET` for IP communications, protocol will be zero, and type will depend on whether TCP or UDP is used. Two processes wishing to communicate over a network create a socket each. These are similar to two ends of a pipe - but the actual pipe does not yet exist.

5.5 TOMCAT SERVER

Tomcat is an open-source web server developed by Apache Group. Apache Tomcat is the servlet container that is used in the official Reference Implementation for the Java Servlet and Java Server Pages technologies. The Java Servlet and Java Server Pages specifications are developed by Sun under the Java Community Process. Web Servers like Apache Tomcat support only web components while an application server supports web components as well as business components (BEAs WebLogic, is one of the popular application servers).



Tomcat's architecture is built upon a modular design, allowing it to function as a standalone server or be integrated seamlessly with other web servers like Apache HTTP Server. Its core components include the Catalina servlet container, which handles the execution of Java Servlets and JSP pages, and the Coyote connector, responsible for managing communication between Tomcat and external clients using various protocols such as HTTP.

Key Features:

Servlet and JSP Support:

Tomcat provides full support for Java Servlets and Java Server Pages, enabling developers to build dynamic and interactive web applications.

Open-Source Nature:

Being open-source, Tomcat is freely available, fostering a vibrant community of developers who contribute to its improvement and ensure its continual evolution.

Platform Independence:

Tomcat is platform-independent, making it compatible with various operating systems, including Windows, Linux, and macOS.

Scalability:

Tomcat offers scalability, allowing it to efficiently handle a growing number of requests by adding more resources or by clustering multiple Tomcat instances.

Security Features:

Security is a top priority for Tomcat, with features such as SSL/TLS support, authentication, and authorization mechanisms, safeguarding web applications from potential threats.

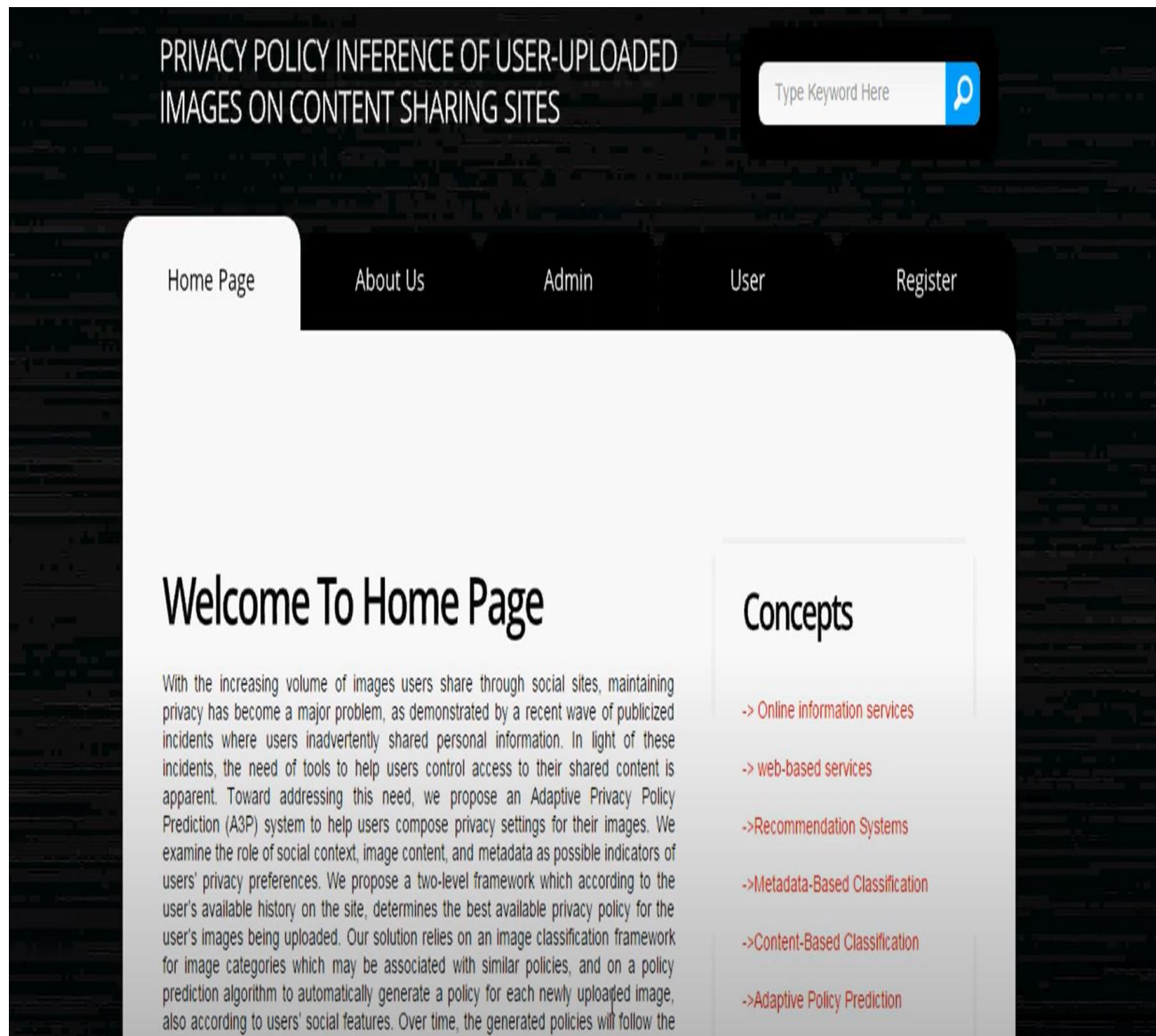
Management and Monitoring:

Tomcat includes a web-based management interface, the Tomcat Manager, facilitating easy deployment, undeployment, and monitoring of applications. Additionally, tools like JConsole and JVisualVM can be employed for in-depth performance analysis.

Deploying applications on Tomcat is a straightforward process. Developers can package their applications as WAR (Web Application Archive) files, which can then be deployed to Tomcat using the Tomcat Manager or by placing the WAR files directly into the designated deployment directory.

Apache Tomcat continues to be a stalwart choice for developers seeking a reliable, scalable, and open-source solution for hosting Java web applications.

7. OUTPUT SCREENS



PRIVACY POLICY INFERENCE OF USER-UPLOADED IMAGES ON CONTENT SHARING SITES

Type Keyword Here



Home Page

About Us

Admin

User

Register

ADMIN LOGIN DETAILS !!

Username

Password

Submit

Clear

01 Policy Mining

02 Policy Setting

03 Image Classificatio

PRIVACY POLICY INFERENCE OF USER-UPLOADED IMAGES ON CONTENT SHARING SITES

Type Keyword Here



Home Page

About Us

Admin

User

Register

USER LOGIN DETAILS!!

Username

Password

Submit

Clear

01

Policy
Prediction

02

Social
Data

03

Social
Context



Welcome to Admin

- Add Image
- View All Images With Policy
- View All Images Ranking
- View All Image Details
- View Search History
- View All Users
- Logout

Welcome To Admin Menu



PRIVACY POLICY INFERENCE OF USER-UPLOADED IMAGES ON CONTENT SHARING SITES

Type Keyword Here



Welcome to Admin

- Add Image
- View All Images With Policy
- View All Images Ranking
- View All Image Details
- View Search History
- View All Users
- Logout

Image Uploading !!!

Image Tag	<input type="text"/>
Image Color	<input type="text"/>
Image Description	<input type="text"/>
Image Uses	<input type="text"/>
Policy	<input type="checkbox"/> All <input type="checkbox"/> Content <input type="checkbox"/> Meta Data <input type="checkbox"/> Image
Select Image	<input type="button" value="Choose file"/> No file chosen
<input type="button" value="Submit"/> <input type="button" value="Reset"/>	

PRIVACY POLICY INFERENCE OF USER-UPLOADED IMAGES ON CONTENT SHARING SITES

Type Keyword Here



Welcome to Admin

[Add Image](#)

[View All Images With Policy](#)

[View All Images Ranking](#)

[View All Image Details](#)

[View Search History](#)

[View All Users](#)

[Logout](#)

List Of Images



Image Added By :Rakesh

Image Policy : |Content|Metadata



Image Added By :Rakesh

Image Policy :All



Image Added By :raju

Image Policy :All



Image Added By :admin

Image Policy : |Content|Image

PRIVACY POLICY INFERENCE OF USER-UPLOADED IMAGES ON CONTENT SHARING SITES

Type Keyword Here



Welcome to Admin

- [Add Image](#)
- [View All Images With Policy](#)
- [View All Images Ranking](#)
- [View All Image Details](#)
- [View Search History](#)
- [View All Users](#)
- [Logout](#)

List Of Images



Image Added By :Rakesh

Image Policy : |Content|Metadata



Image Added By :Rakesh

Image Policy :All



Image Added By :raju

Image Policy :All



Image Added By :admin

Image Policy : |Content|Image

PRIVACY POLICY INFERENCE OF USER-UPLOADED IMAGES ON CONTENT SHARING SITES

Type Keyword Here



Welcome to Admin

- Add Image
- View All Images With Policy
- View All Images Ranking
- View All Image Details
- View Search History
- View All Users
- Logout

List Of Images With Content



Rank:[8] Rating : ★★ ★

Image Added By :Rakesh

Image Policy : Content/Metadata

Tag :Tiger

Color :Brown

Content: Tiger is wild animal and we get this type of animal in forest and zoo but in old days some people put tigers in house for safety purpose and fashion.



Rank:[6] Rating : ★★ ★

Image Added By :Rakesh

Tag :Tiger

Color :Black

Content: The tiger is the largest cat species, reaching a total body length of up to 3.38 m over curves and now a days tigers we are seeing in zoo.

PRIVACY POLICY INFERENCE OF USER-UPLOADED IMAGES ON CONTENT SHARING SITES

Type Keyword Here



Welcome to Admin

- Add Image
- View All Images With Policy
- View All Images Ranking
- View All Image Details
- View Search History
- View All Users
- Logout

List Of Users !!!

Username	Mobile	Status	View
Rakesh	8956412344	Authorized	more info...
Raju	9955662233	Authorized	more info...
tmksmanju	9955662211	Authorized	more info...

PRIVACY POLICY INFERENCE OF USER-UPLOADED IMAGES ON CONTENT SHARING SITES

Type Keyword Here



Welcome to Admin

Add Image
View All Images With Policy
View All Images Ranking
View All Image Details
View Search History
View All Users
Logout

List Of Users !!!

Username	Mobile	Status	View
Rakesh	8956412344	Authorized	more info...
Raju	9955662233	Authorized	more info...
tmksmanju	9955662211	Authorized	more info...

PRIVACY POLICY INFERENCE OF USER-UPLOADED IMAGES ON CONTENT SHARING SITES

Type Keyword Here



Welcome to Admin

Add Image
View All Images With Policy
View All Images Ranking
View All Image Details
View Search History
View All Users
Logout

User Details !!!

	Username	Rakesh
	E-Mail	rakesh@gmail.com
	Mobile	8956412344
	Date Of Birth	12/12/1990
	Address	Belgaum
	Status	Authorized

PRIVACY POLICY INFERENCE OF USER-UPLOADED IMAGES ON CONTENT SHARING SITES

Type Keyword Here



Welcome to Admin

Add Image
View All Images With Policy
View All Images Ranking
View All Image Details
View Search History
View All Users
Logout

List Of Users !!!

Username	Mobile	Status	View
Rakesh	8956412344	Authorized	more info...
Raju	9955662233	Authorized	more info...
tmksmanju	9955662211	Authorized	more info...

PRIVACY POLICY INFERENCE OF USER-UPLOADED IMAGES ON CONTENT SHARING SITES

Type Keyword Here



Welcome to Admin

Add Image
View All Images With Policy
View All Images Ranking
View All Image Details
View Search History
View All Users
Logout

User Details !!!

	Username	Raju
	E-Mail	raju@gmail.com
	Mobile	9955662233
	Date Of Birth	14/05/1988
	Address	Bangalore
	Status	Authorized

PRIVACY POLICY INFERENCE OF USER-UPLOADED IMAGES ON CONTENT SHARING SITES



Welcome to Admin

[Add Image](#)
[View All Images With Policy](#)
[View All Images Ranking](#)
[View All Image Details](#)
[View Search History](#)
[View All Users](#)
[Logout](#)

List Of Users !!!

Username	Mobile	Status	View
Rakesh	8956412344	Authorized	more info..
Raju	9955662233	Authorized	more info..
tmksmanju	9955662211	Authorized	more info..

Welcome to Admin

[Add Image](#)
[View All Images With Policy](#)
[View All Images Ranking](#)
[View All Image Details](#)
[View Search History](#)
[View All Users](#)
[Logout](#)

User Details !!!



Username	tmksmanju
E-Mail	tmksmanju@gmail.com
Mobile	9955662211
Date Of Birth	13/2/1980
Address	Bangalore
Status	Authorized

PPI : View All Users x PPI : Registration page x Technical Details

localhost:9090/Privacy%20Policy%20Inference/register.jsp

PRIVACY POLICY INFERENCE OF USER-UPLOADED IMAGES ON CONTENT SHARING SITES

Type Keyword Here

Home Page About Us Admin User Register

User Registration Form

Name:

Password:

E-Mail:

Mobile:

Dob:

Profile No file chosen

Address:

Login

Username

Password

PPI : View All Users x PPI : Registration page x Technical Details

localhost:9090/Privacy%20Policy%20Inference/register.jsp

PRIVACY POLICY INFERENCE OF USER-UPLOADED IMAGES ON CONTENT SHARING SITES

Type Keyword Here

Home Page About Us Admin User Register

User Registration Form

Name:

Password:

E-Mail:

Mobile:

Dob:

Profile No file chosen

Address:

Login

Username

Password

localhost:9090/Privacy%20Policy%20Inference/register.jsp

PRIVACY POLICY INFERENCE OF USER-UPLOADED IMAGES ON CONTENT SHARING SITES

Type Keyword Here

Home Page About Us Admin User Register

User Registration Form

Name: Manjunath

Password: *****

E-Mail: tmksmanju14@gmail.com

Mobile: 9535866270

Dob: 05/06/1983

Profile: Choose file manju.jpg

Address: Rajaji Nagar, Marivappa Palya, Bangalore

Submit Reset

Login

Username

Password

Submit

PRIVACY POLICY INFERENCE OF USER-UPLOADED IMAGES ON CONTENT SHARING SITES

Type Keyword Here



Welcome to Admin

[Add Image](#)

[View All Images With Policy](#)

[View All Images Ranking](#)

[View All Image Details](#)

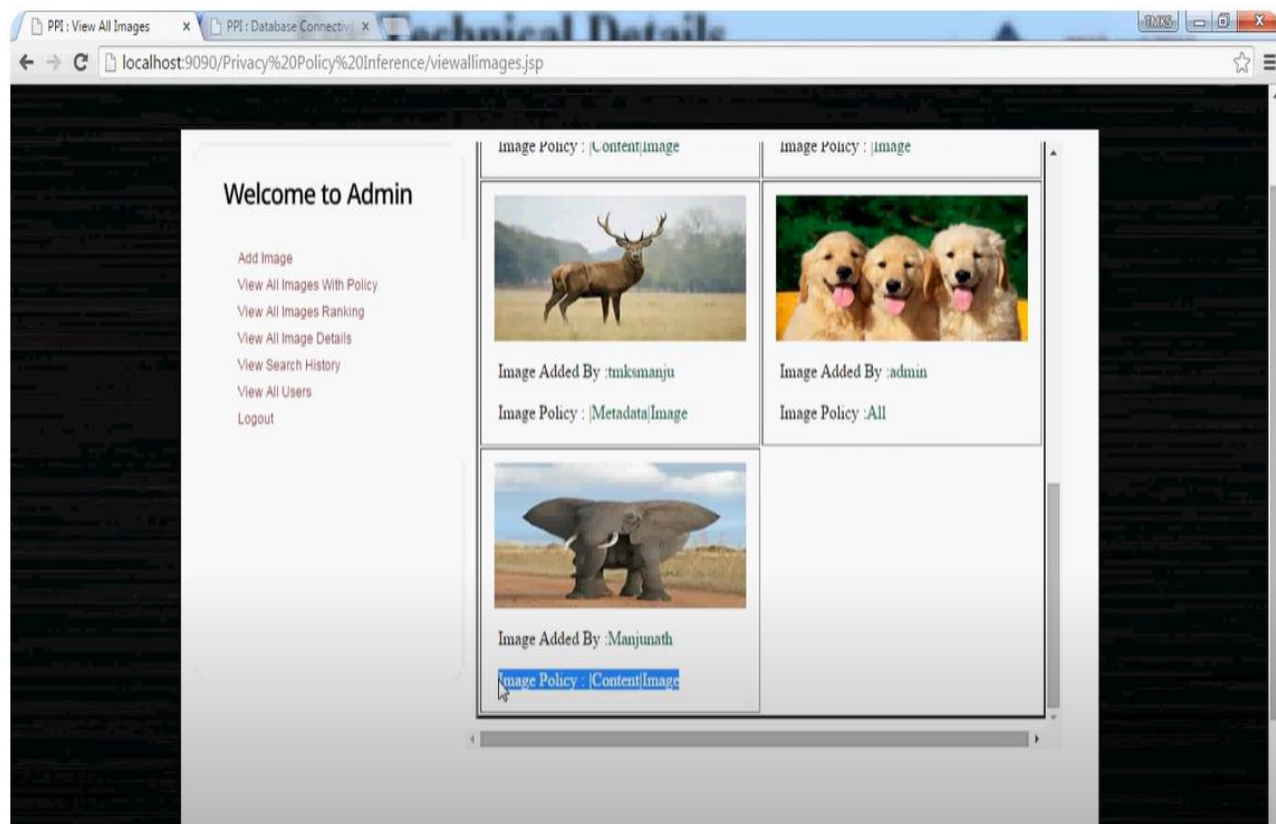
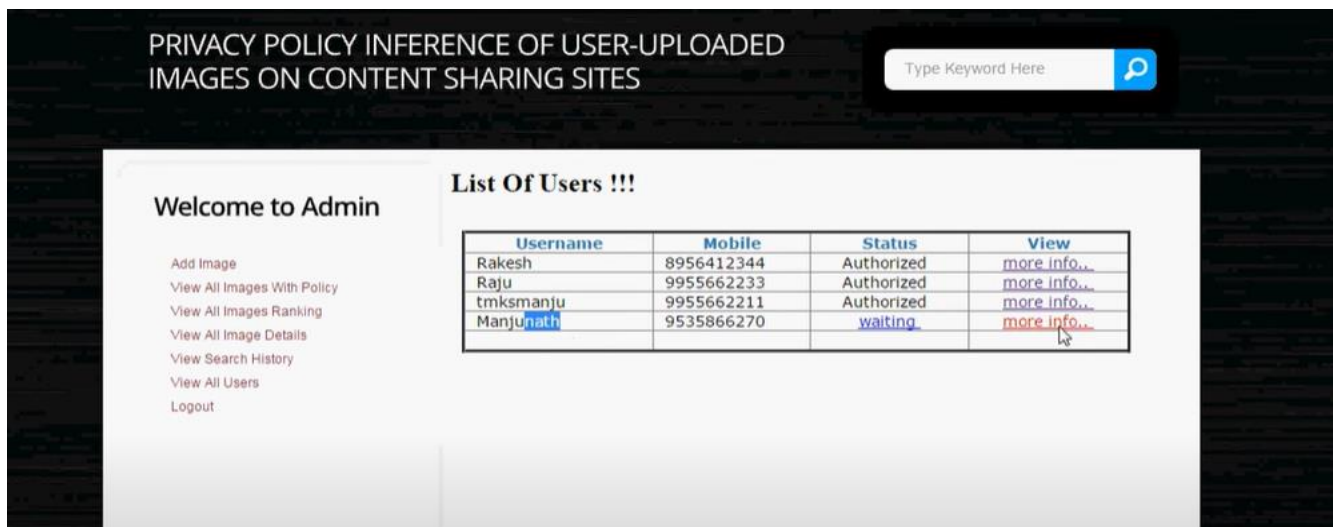
[View Search History](#)

[View All Users](#)

[Logout](#)

List Of Users !!!

Username	Mobile	Status	View
Rakesh	8956412344	Authorized	more info..
Raju	9955662233	Authorized	more info..
tmksmanju	9955662211	Authorized	more info..



6.TESTINGSTRATEGY

SYSTEM TESTING*

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

TYPES OF TESTS

Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

1)Top-Down Integration

This method is an incremental approach to the construction of program structure. Modules are integrated by moving downward through the control hierarchy, beginning with the main program module. The module subordinates to the main program module are incorporated into the structure in either a depth first or breadth first manner.

In this method, the software is tested from main module and individual stubs are replaced when the test proceeds downwards.

2. Bottom-up Integration

This method begins the construction and testing with the modules at the lowest level in the program structure. Since the modules are integrated from the bottom up, processing required for modules subordinate to a given level is always available and the need for stubs is eliminated. The bottom up integration strategy may be implemented with the following steps:

- The low-level modules are combined into clusters into clusters that perform a specific Software sub-function.
- A driver (i.e.) the control program for testing is written to coordinate test case input and output.
- The cluster is tested.
- Drivers are removed and clusters are combined moving upward in the program structure

The bottom up approaches tests each module individually and then each module is module is integrated with a main module and tested for functionality.

Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

User Acceptance Testing

User Acceptance of a system is the key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes wherever required. The system developed provides a friendly user interface that can easily be understood even by a person who is new to the system.

Output Testing

After performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in the specified format. Asking the users about the format required by them tests the outputs generated or displayed by the system under consideration. Hence the output format is considered in 2 ways – one is on screen and another in printed format.

Validation Checking

Validation checks are performed on the following fields.

Text Field:

The text field can contain only the number of characters lesser than or equal to its size. The text fields are alphanumeric in some tables and alphabetic in other tables. Incorrect entry always flashes and error message.

Numeric Field:

The numeric field can contain only numbers from 0 to 9. An entry of any character flashes an error messages. The individual modules are checked for accuracy and what it has to perform. Each module is subjected to test run along with sample data. The individually tested modules are integrated into a single system. Testing involves executing the real data information is used in the program the existence of any program defect is inferred from the output. The testing should be planned so that all the requirements are individually tested.

A successful test is one that gives out the defects for the inappropriate data and produces and output revealing the errors in the system.

Preparation of Test Data

Taking various kinds of test data does the above testing. Preparation of test data plays a vital role in the system testing. After preparing the test data the system under study is tested using that test data. While testing the system by using test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

USER TRAINING

Whenever a new system is developed, user training is required to educate them about the working of the system so that it can be put to efficient use by those for whom the system has been primarily designed. For this purpose the normal working of the project was demonstrated to the prospective users. Its working is easily understandable and since the expected users are people who have good knowledge of computers, the use of this system is very easy

MAINTAINENCE

This covers a wide range of activities including correcting code and design errors. To reduce the need for maintenance in the long run, we have more accurately defined the user's requirements during the process of system development. Depending on the requirements, this system has been developed to satisfy the needs to the largest possible extent. With development in technology, it may be possible to add many more features based on the requirements in future. The coding and designing is simple and easy to understand which will make maintenance easier.

CONCLUSION

We have proposed an Adaptive Privacy Policy Prediction (A3P) system that helps users automate the privacy policy settings for their uploaded images. The A3P system provides a comprehensive framework to infer privacy preferences based on the information available for a given user. We also effectively tackled the issue of cold-start, leveraging social context information. Our experimental study proves that our A3P is a practical tool that offers significant improvements over current approaches to privacy.

REFERENCES

- [1] A. Acquisti and R. Gross, “Imagined communities: Awareness, information sharing, and privacy on the facebook,” in Proc. 6th Int. Conf. Privacy Enhancing Technol. Workshop, 2006, pp. 36–58.
- [2] R. Agrawal and R. Srikant, “Fast algorithms for mining association rules in large databases,” in Proc. 20th Int. Conf. Very Large Data Bases, 1994, pp. 487–499.
- [3] S. Ahern, D. Eckles, N.S. Good, S. King, M. Naaman, and R. Nair, “Over-exposed?: Privacy patterns and considerations in online and mobile photo sharing,” in Proc. Conf. Human Factors Comput. Syst., 2007, pp. 357–366.
- [4] M. Ames and M. Naaman, “Why we tag: Motivations for annotation in mobile and online media,” in Proc. Conf. Human Factors Comput. Syst., 2007, pp. 971–980.
- [5] A. Besmer and H. Lipford, “Tagged photos: Concerns, perceptions, and protections,” in Proc. 27th Int. Conf. Extended Abstracts Human Factors Comput. Syst., 2009, pp. 4585–4590.
- [6] D. G. Altman and J. M. Bland, “Multiple significance tests: The bonferroni method,” Brit. Med. J., vol. 310, no. 6973, 1995.
- [7] J. Bonneau, J. Anderson, and L. Church, “Privacy suites: Shared privacy for social networks,” in Proc. Symp. Usable Privacy Security, 2009.