INTRODUCTION

According to UNESCO report, about 64% of population of the globe is English illiterate. The percentage is more in developing countries like India, China, and Pakistan etc. where nearly 76% people are English illiterate. It is obvious that these people are deprived from the advantages of Internet because majority of web pages are in English. Hence, there is a need to bridge the Digital divide exists since beginning of IT revolution i.e. the last decade of the previous century. Let's discuss about challenges ahead to access the Internet repository. In present scenario, users who are familiar with Indian languages and less conversant with English face difficulties in accessing the web services.

The traditional web services (like Indian Railways) generate dynamic web page with respect to query given by the user in English language as they maintain their database in the same language. Thus, the services are lacking of support to user's query in Indian languages and as a result, unable to produce dynamic web pages in any language excluding English. It may be also noted that many translation engines which convert web pages from English to Indian languages, have addressed the problem for static content of a web page.

For dynamic web page content, the success rate is very poor. This specific challenge has been addressed in this work. We propose a mechanism, called “Two-way Interaction” which enables a user to interact with the dynamic web pages in user’s mother language only and the results returned during interactions are displayed in same language

PROBLEM DEFINITION AND SCOPE OF THE PROJECT

1.1 PURPOSE:

The purpose of this document is to describe all the requirements, specifications and use of the Accessing Dynamic Webpage in user languages. It also describes the interfaces for the system. The SRS document describes the aim of the project and its wide scope of application. It intends to provide an overview of the implementation details including the platform, tools and e technologies to be used. It addresses the audience of/like examiner, project analyzers, users like companies, ambulance management councils, military and others.

1.2 SCOPE:

A dynamic web page is a page that changes based on the user. It responds to the user's needs, and provides relevant information to meet them, by accessing information in a connected database. A dynamic web page allows users to go beyond reading text and looking at graphics. It allows for an interactive experience, with the user being in control of the information he views. Thus the dynamic web pages maintain up to date information and provides a two - way interaction with server and can retrieve results. This problem had been addressed and then we can access static web page in any language required by user with the help of Google Translate. But , by static web pages user can't give input to the web page in his own language . To over come , this paper has been proposed to access dynamic web page in user language. This service provides the Internet content to millions of people who might not have good capability to read the web content in English. It will minimize the gap between the Internet and user due to language barrier. It helps the users to access Internet in their day to day life without worrying much about the language web page is originally written.

LITERATURE SURVEY

2.1 INTRODUCTION TO ORACLE 10G

Oracle Database 10g is the first database designed for grid computing, the most flexible and cost-effective way to manage enterprise information. It cuts costs of management while providing the highest possible quality of service.

In addition to providing numerous quality and performance enhancements, Oracle Database 10g significantly reduces the costs of managing the IT environment, with a simplified install, greatly reduced configuration and management requirements, and automatic performance diagnosis and SQL tuning.

These and other automated management capabilities help improve DBA and developer productivity and efficiency.10g is Oracle's grid computing product group including a database management system (DBMS) and an application server. In addition to supporting grid computing features such as resource sharing and automatic load balancing, 10g products automate many database management tasks. The Real Application Cluster (RAC) component makes it possible to install a database over multiple servers JDBC.

2.2 JDBC (JAVA DATABASE CONNECTIVITY)

JDBC technology is an API(included in both J2SE and J2EE releases) that provides cross DBMS connectivity to a wide range of SQL database and access to other tabular data sources, such as spreadsheets or flat files. With a JDBC technology-enabled driver, you can connect all corporate data even in a heterogeneous environment.

The JDBC API makes it possible to do three things:

- Establish a connection with a database or access any tabular data source
- Send SQL statements
- Process the results

2.2.1 TYPES OF JDBC TECHNOLOGY DRIVES

JDBC technology drivers fit into one of four categories:

- A JDBC-ODBC bridge provides JDBC API access via one or more ODBC drivers.
- A native-API partly Java technology-enabled driver converts JDBC calls into calls on the client API for Oracle, Sybase, Informix, DB2, or other DBMS.
- A net-protocol fully Java technology-enabled driver translates JDBC API calls into a DBMS-independent net protocol which is then translated to a DBMS protocol by a server.
- A native-protocol fully Java technology-enabled driver converts JDBC technology calls into the network protocol used by DBMSs directly.

2.2.2 JDBC ARCHITECTURE

The JDBC Architecture consists of two layers:

- The JDBC API, which provides the application-to-JDBC Manager connection.
- The JDBC Driver API, which supports the JDBC Manager-to-Driver Connection.

The JDBC API uses a driver manager and database-specific drivers to provide transparent connectivity to heterogeneous databases. The JDBC driver manager ensures that the correct driver is used to access each data source.

The location of the driver manager with respect to the JDBC drivers and the Java application is shown in the figure below:

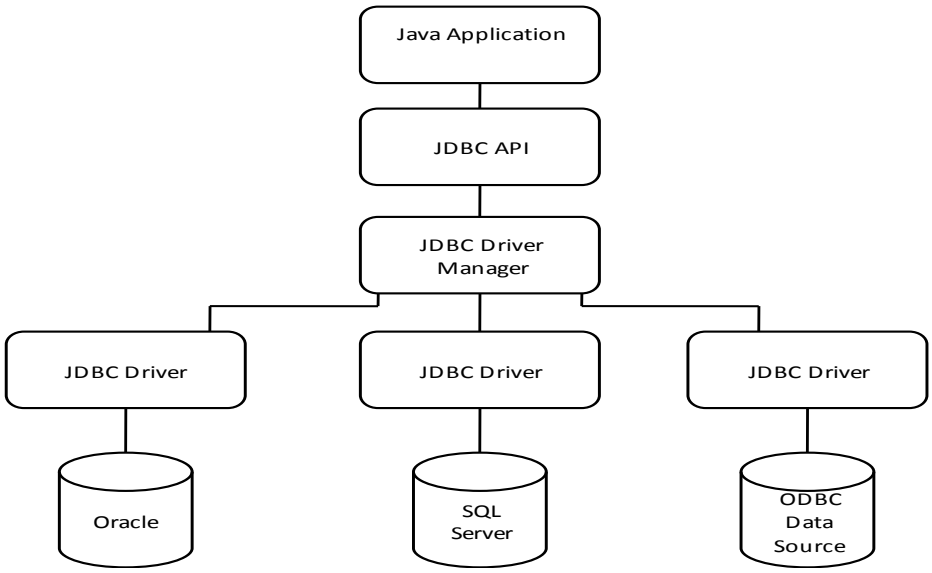

Fig 2.1: JDBC Architecture

2.3 TECHNOLOGY:

Terminology

Static web page, Dynamic web page, Client side scripting, server side scripting

Definitions

Static Web Page:

A **static web page** (sometimes called a **flat page**) is a web page that is delivered to the user exactly as stored, in contrast to dynamic web pages which are generated by a web application.

Dynamic Web page:

A **dynamic web page** is a kind of web page that has been prepared with fresh information (content and/or layout), for each individual viewing. It is not static because it changes with the

time (e.g. news content), the user (e.g. preferences in a login session), the user interaction (e.g. web page game), the context (e.g. parametric customization), or any combination thereof.

Client Side Scripting:

Client-side scripting to change interface behaviors *within* a specific web page, in response to mouse or keyboard actions or at specified timing events. In this case the dynamic behavior occurs within the presentation.

Server Side Scripting:

A program running on the web server (server-side scripting) is used to change the web content on various web pages, or to adjust the sequence of or reload of the web pages. Server responses may be determined by such Conditions as data in a posted HTML form, parameters in the URL, the type of browser being used, the passage of time, or a database or server state.

User Language: Any language chosen by user (mother language/required language).

2.3.1 TAG SIMULATION PROCESS

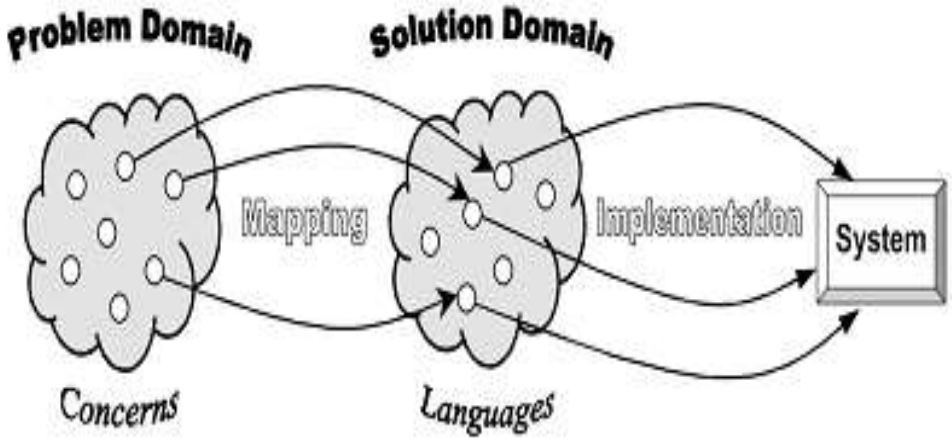

Fig:2.2 Basic Principles of multi-language design

- **Problem domain analysis -** analysis, identification and classification of concerns in a problem domain**.**
- **Solution domain analysis -** identification of programming languages and analysis of their capabilities in a solution domain**.**
- **Selection of languages -** mapping of each problem domain concern to the most suitable solution domain language.

Implementation - composition of a heterogeneous system from the multi-language components implemented using the selected languages

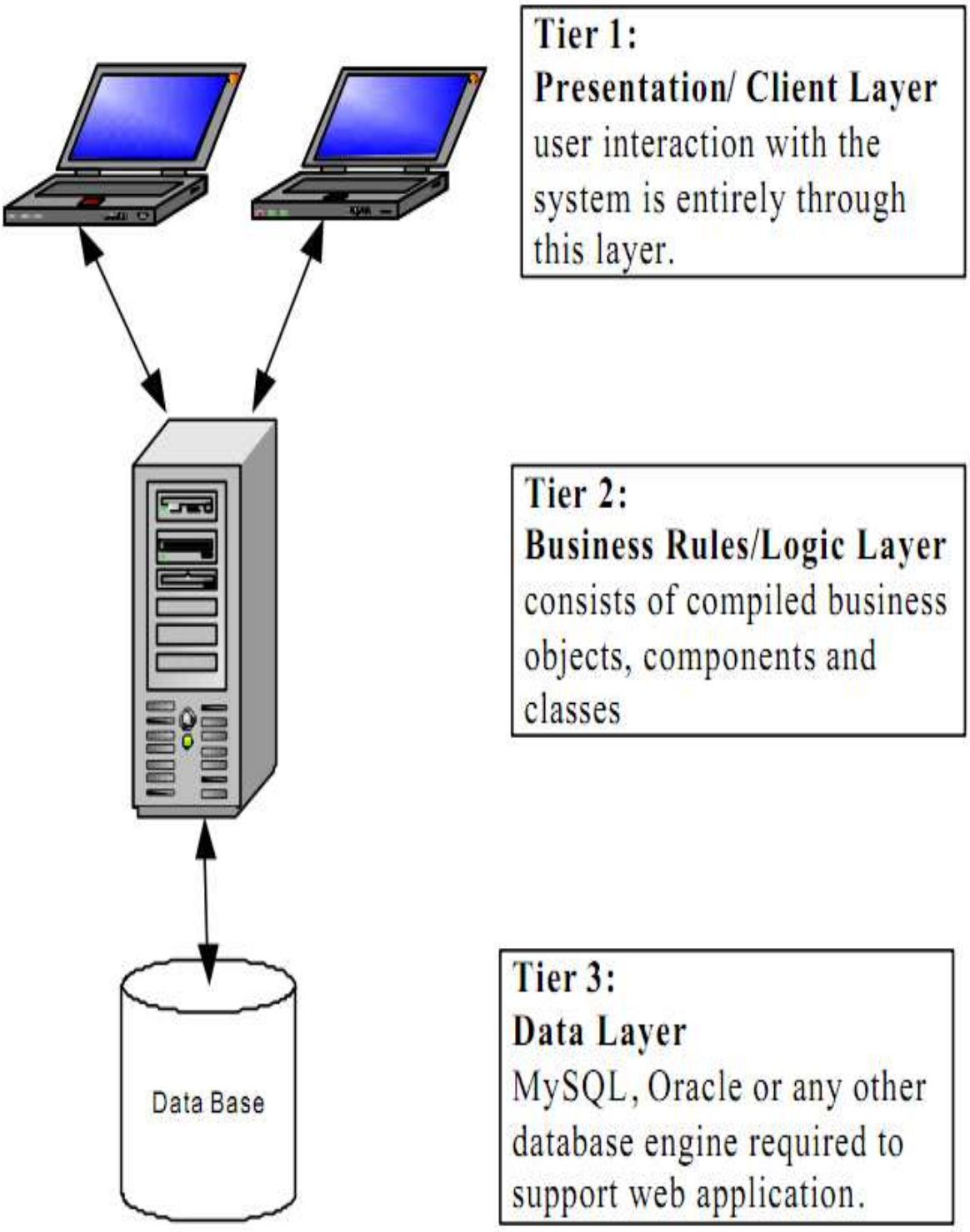

Fig: 2.3 Three tier Standard Architecture

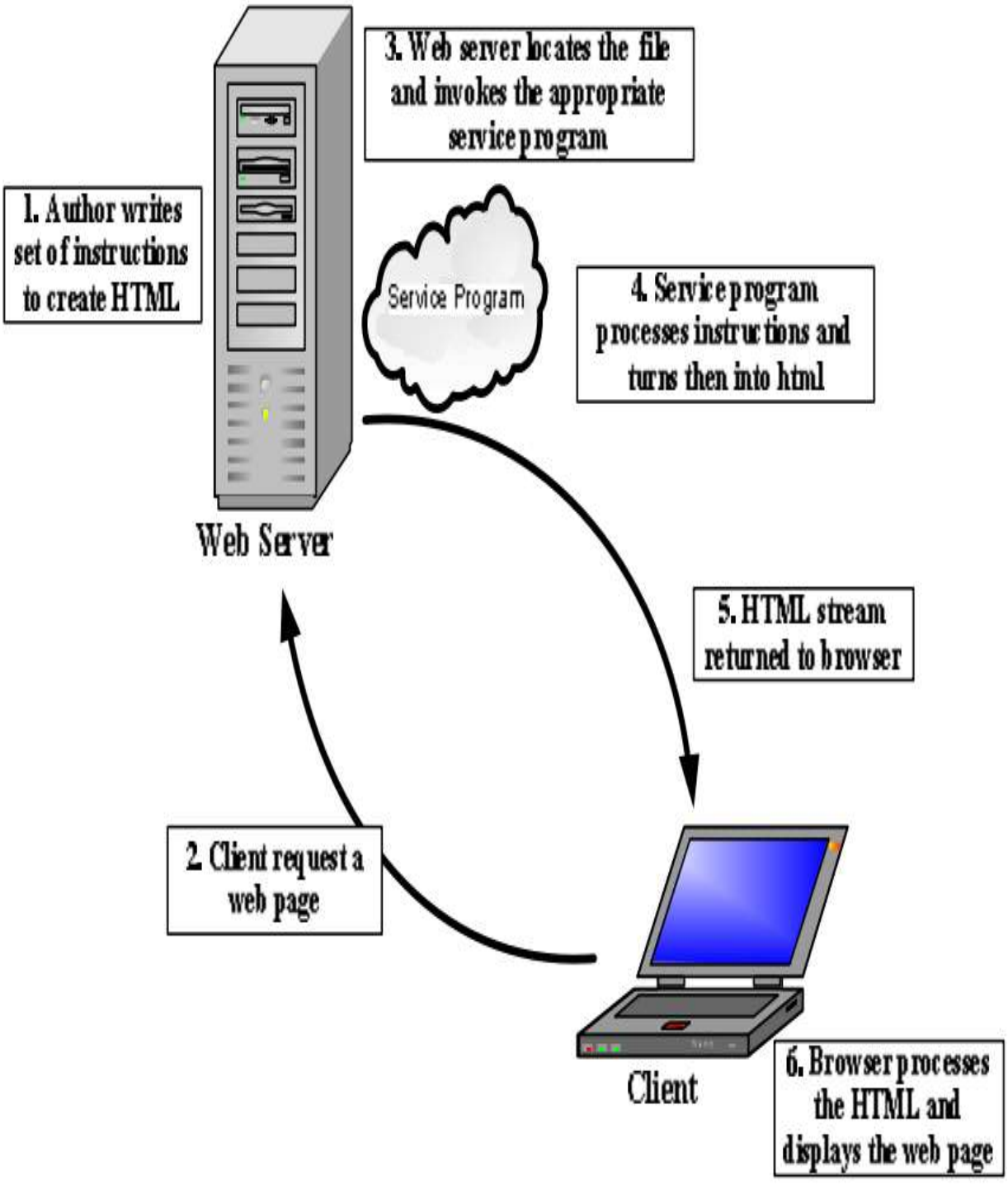

Fig: 2.4 Server side scripting for dynamic web page

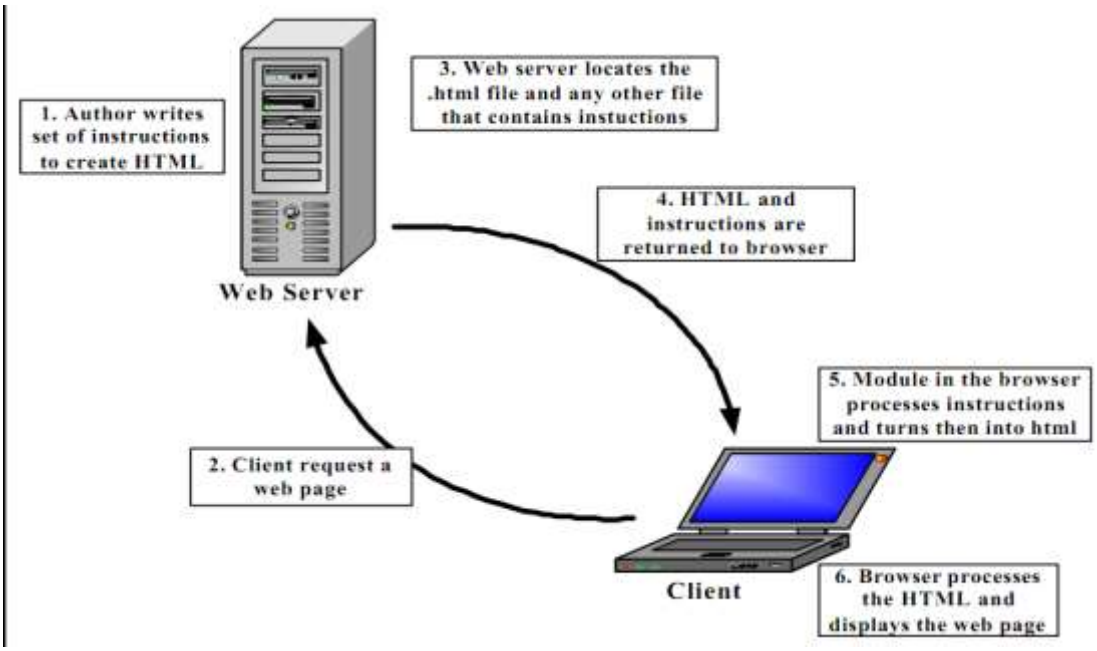

Fig: 2.5 Client side scripting for dynamic web pages

In a dynamic web page content (text, images, fields, etc.) on the web page can change, in response to different contexts or conditions. There are two ways to create this kind of web pages:

1. Using client-side scripting to change interface behaviors within a specific web page.

2. Using server-side scripting to change the sequence of the web pages or web content

A Static web page consists of some HTML code typed directly into a text editor and saved as a .html or .html file. It is a web page that is delivered to the user exactly as stored, in contrast to dynamic web pages which are generated by a web application.

Static pages are already established and are not drawn up from bits and pieces from a search query. However it takes longer for them to index all the data especially when there are thousands of products and if you have set up products with similar descriptions it can cause duplicate content issues.

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM:

The SRS document describes the aim of the project and its wide scope of application. It intends to provide an overview of the implementation details including the platform, tools and e technologies to be used. It addresses the audience of/like examiner, project analyzers, users like companies, ambulance management councils, military and others.

3.2 PROPOSED SYSTEM:

A dynamic web page is a page that changes based on the user. It responds to the user's needs, and provides relevant information to meet them, by accessing information in a connected database. A dynamic web page allows users to go beyond reading text and looking at graphics. It allows for an interactive experience, with the user being in control of the information he views.

Thus the dynamic web pages maintain up to date information and provides a two - way interaction with server and can retrieve results.

This problem had been addressed and then we can access static web page in any language required by user with the help of Google Translate. But , by static web pages user can't give input to the web page in his own language . To overcome , this paper has been proposed to access dynamic web page in user language.

This service provides the Internet content to millions of people who might not have good capability to read the web content in English . It will minimize the gap between the Internet and user due to language barrier. It helps the users to access Internet in their day to day life without worrying much about the language web page is originally written.

3.2.1 SYSTEM FEATURES

Retrieve

Description and Priority

The Retrieve sub module retrieves a web page as requested by the user and separates the content of web page (by using HTML parser) into HTML tags and English text.

Stimulus/Response Sequences

Whenever user requests any URL, the content of requested URL will be retrieved and displayed on the client window.

Functional Requirements

Req1: Query

- **Purpose/Description:** To retrieve desired content
- **Inputs:** Query
- **Processing:** Comparing query with the database
- **Outputs:** Displaying the desired web page

Translation

Description and Priority

The Translation sub module takes input as a language and translates the webpage into user language.

Stimulus/Response Sequences

Whenever user wants to translate his webpage into a language other than English then it translates into user language.

Functional Requirements

Req1: Selecting Language

- **Purpose/Description:** To translate into required language
- **Inputs:** Language
- **Processing:** Translates into selected language
- **Outputs:** Web page in user language.

Virtual Keyboard:

Description and Priority

Virtual Keyboard is used when user wants to give any input in his own language.

Stimulus/Response Sequences

User need to select virtual keyboard whenever he wants to give input.

Functional Requirements

Req1: Selecting Virtual Keyboard

- **Purpose/Description:** To allow user to give input in his own language
- **Inputs:** Selecting Virtual keyboard
- **Processing:** Takes input and displays

- **Outputs:** Displays the text given by user.

3.3 FEASIBILITY STUDY

The main goal of feasibility study is to assess the viability of the application developed. The outcome of the feasibility study will indicate whether or not to proceed with the proposed venture. If the result of the feasibility study is positive, then the cooperative can proceed to develop a business plan. is becoming increasingly prevalent as the price of the technology decreases.

3.4 Nonfunctional Requirements

Performance Requirements

The system which we are developing is used in internet. So there will be large number of users who are using this site. So performance should be more which will be achieved based on the database and application server we are using.

Safety Requirements

The database may get crashed at any certain time due to virus or operating system failure. Therefore, it is required to take the database backup.

Security Requirements

To access this system, the user is provided with a login entry. Unauthorized user cannot access the system so that system will be secure. Moreover different users will be given different access rights. It means if the user is an administrator then he can able to modify, delete the data. All other users can only access the data.

Software Quality Attributes

This system possess the quality characteristics that will be important to either the customers or the developers include adaptability, availability, accuracy, flexibility, interoperability, maintainability ,portability, reliability, reusability, robustness, testability, and usability.

3.4.1 TECHNICAL FEASIBILITY

The analyst must find out whether current technical resources can be upgraded or added to in a manner that fulfills the request under consideration. Before implementing the project, we have to consider what are the devices required for implementation. This comes under technical feasibility of the project. In our system the devices required are and readers, virtual keyboard, Web connections These devices are easily available. Thus our project is technically feasible.

TRANSLATION PROCESS:

The translation processes implies an entire process of how a translator produces equivalences between a text or portions of a text into another language. The translation process can be described as:

• Decoding the meaning of the source text, and

•Re-encoding or translating this meaning in the target language.

Behind this simple process lies various activities like checking grammar, syntax, idioms, semantics, and the like of the source language and also the culture of its speakers.

Translation Types:

There are different types of Translation

- ✓Technical translation

- ✓ scientific translation
- ✓ artistic (literary) Translation
- ✓ translation of documents
- ✓ general translation

Translation used in this approach is artistic translation. It can be defined as:

In this paper, we only deal with artistic translation, i.e. only the text in the html page need to be translated . so it is literary translation. Literary translations don't only require a thorough knowledge of the source and target languages, but also the ability to correctly translate the original feelings and to employ the most appropriate language means in the translation. A good translation agency will provide you with an experienced proofreader whose native language is the target language of the translation.

Steps Involved in Language Translation

To maintain the highest quality of translation, usually three step quality processes is followed:

- ✓ The document is translated by a professional translator. This translator is a native speaker of the required language. He is also an expert in his area of specialization.
- ✓ The document is then proof-read by another translator for accuracy. He can offer suggestions for any changes that make the document easier to understand.
- ✓ Finally, the document is reviewed to ensure accuracy, consistency, appropriate formatting, and overall quality of the final product so that everything meets the expectations and is ready for delivery to the clients.

Translation Tools:

- ✓ Google Translate
- ✓ Tricia

Google Translate

Google Translate is a free, web-based and statistically-based machine translation service provided by Google. It enables to translate section of text, document or webpage, from one language to another.

- **Google translate API**

The Google Translate API lets websites and programs integrate with Google Translate programmatically. During the project development phase Google provides two versions of API. The version 2 is the latest available version of the Google Translate API. We decided to use version 2 within the project. After that point whenever we refer to translate API in this report, we mention version 2. One needs a Google account to use the translation service. Because, the Translate API requires the use of an API key and it can only be received from Google APIs console8. There are two ways to invoke the API: Using REST 9 directly or using REST from JavaScript (This does not require server-side coding.). JSON10 is used by Google as a data format.

- **Limitations**

When one uses the Google API, it is required to accept the terms and conditions. Here are the most important limits for our project:

1. Every request text to be translated can be up to maximum of 5000 characters long.
2. Daily limit is 100.000 characters per API key.
3. Continuous translation requests successively results in ’Suspected Terms of Service Abuse’.
4. Batch requests are against Terms of Service.

Tricia

Tricia Platform is an Open Source platform for developing dynamic web applications. It is built in Java and realizes the model-view-controller pattern, which consists of the following components:

I. An abstraction for defining control flow

II. A templating language

III. An object/relational persistence mapping

It has to be emphasized that what has been described so far is called Tricia Platform, since it is a generic platform for building arbitrary dynamic web applications. Tricia Platform is implemented in the Eclipse project Toro. In addition to the pure infrastructure, the toro project already comes with the basic asset types Person and Group, which are required by almost any dynamic web application. Additionally, Toro comes with an existing basic layout. On the other hand, there are concrete plug-in built on top of the Tricia platform and the basic Tricia plug-in. combining these plug-in results in the Tricia Application. A typical configuration of Tricia consists of the plug-in Toro, File, Wiki, and Blog. The distinction between the platform and the application is similar to Ruby on Rails as a platform and applications like Base camp or Backpack, which are built using Ruby on Rails.

Client and Server Side:

I. Tricia is a web knowledge collaboration and communication tool. As common to every web tool, Tricia has client side features and server side functionalities.

II. The server side functionalities of Tricia include the user management and management of the plug-ins to the Tricia core like blog, wiki page.

The client side features of Tricia includes the handling of inputs (user generated contents) wiki page content, blog content, comment, feedbacks.

What needs to be translated?

The internationalization plug-in of Tricia needs to translate of the server side static texts such as labels of elements, warning messages along with the dynamic user generated content such as the blog content generated by the user.

The server side static text includes:

- The text available in Messages class. This text represents the static String messages that are used display information to the user
- The text available in messages.xml
- The static text messages available as objects of SimpleMessage.java class
- The domain values. For example, instances of Membership Visibility class
- The static text messages embedded within the template htm files .

The dynamic user generated text includes:

The RichStringProperty fields which includes the user-generated content of the blog page, wiki page, comment and description fields, etc.

Existing Internalization System

The present internationalization system in Tricia is not fully tested and does not perform its full intended functionality. It is a partially implemented system. The internalization features present in Tricia are:

- **Language Handler:**

The Language Handler handles the change of language parameter as triggered by the user action. The Language Handler identifies the user chosen language from the URL parameter and sets in to the session variable. The session variable is set when the user clicks his/her preferred language on the Tricia page. The languages are identified with the flags as icons. The user can chose any of the language available from the flags displayed on the top right corner of the home page.

- **Translator Configurator:**

The Language flags on the Tricia pages can be configured through the Translator configurator tool of Tricia Platform. This tool provides a way to add the desired languages

(according to the multiple languages required to be supported by the Tricia). A language can be added through the configurator tool.

The Internationalization plug in for Tricia has to

- Identify the texts appropriate for translation.
- Store and handle the transalted text.
- Update the Tricia Web page according to the language chosen by the user.

Regularly update the translations

3.4.2 ECONOMIC FEASIBILITY

Economic Feasibility is the second part of resource determination. The basic resources to consider are: our time and that of systems analysis team, the cost of doing a full systems study, cost of business employee time, estimated cost of hardware, and estimated cost of software and/or software development.

In our project the devices required are virtual keyboard and Reader which are available now a day at low cost. The entire database is maintained by the Administrator. So, extra manual power is not required. Thus our project is economically feasible.

3.4.3 BEHAVIORAL FEASIBILITY

Based on the targeted user's behavior, this can be judged. An estimate should be done of how strong the administrator is likely to have the development of the current system. This proposed system "**ACCESSING DYNAMIC WEB PAGE IN USER**" has much behavioral feasibility because users are provided with a better facility. So, project is also behaviorally feasible.

3.4.4 TIME FEASIBILITY

Based on the time limitations, time feasibility is done. Time Feasibility is done to evaluate the time in which the application can be done. If the project takes much more time than the dated time, then the Time Feasibility is not possible. Then further justifications and adjustments are to be made to complete the task before the date of completion. Hence time feasibility can be achieved in this system.

3.5 PROTOTYPE MODEL:

In this model the requirements are gathered firstly, and the prototype is deployed according to the requirements. This prototype is a quick design which goes through the coding, design and testing. The phases are not done in detail. By seeing this prototype the client feels like a real system, so that the client understands the entire requirements of the systems.

ADVANTAGES:

1. During the development process the developers are interestingly engaged.
2. The prototype developed that is used by the users for well understanding of the methodology
3. The user involvement is increased and improved.
4. The flaws and faults are identified early.
5. The user's opinion about the product is known early which leads to an improved system.

DISADVANTAGES:

1. This model focuses on design quite than functionality.
2. The model is implemented firstly and then errors are evaluated later which becomes a complex process
3. The model is also known as throw-away prototype.
4. More time spent on development of the prototype that result in delay of the final product.

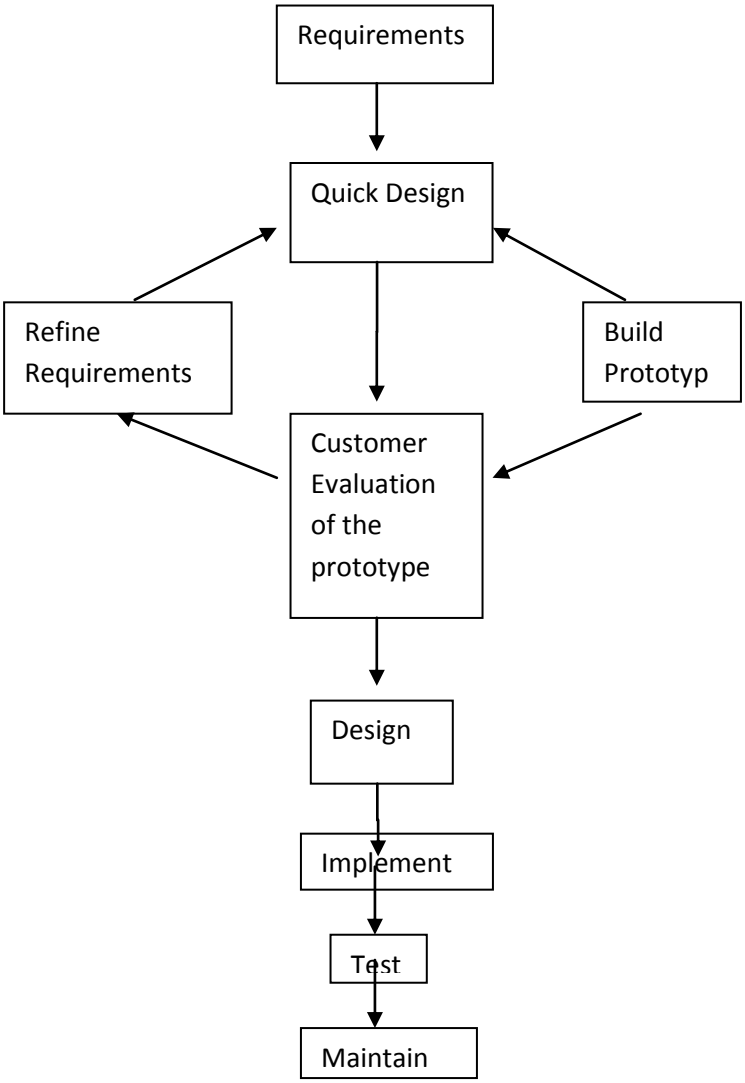

Fig: 3.1 Prototyping Methodology

.

SYSTEM REQUIREMENT SPECIFICATION

4.1 MODULES

1. USER REGISTRATION

2. TRANSLATION

3. DEVELOPMENT OF VIRTUAL KEYBOARD

4. DATA RETRIEVAL

Each module is provided with different functionalities. Each module can be described as follows:

1. USER REGISTRATION:

The first module which is used for registration purpose and then the user can login through this module.

PROVISIONS:

The registration module deals with the registration of the person who uses the application i.e. user. The registration module should be provided with the following fields which are to be filled by the user. The fields are as follows:

- Username
- Password
- Phone No
- Email id
- Address

FUNCTIONALITIES:

By using this module the user can be registered giving user details and can use the application in any language selected by user. The user can Log in to the application after getting registered using this "USER REGISTRATION" module.

Registration

Login

QUERIES:

In this registration module the various queries are arises. The Queries arise in this module are based on the fields used in this module .The queries like:

- What is username?
- How many minimum numbers of characters should be there in password?
- Is the password valid or not?
- What is the current address?
- What is alternate mail id?

ALERTS:

Alerts are alert messages that are shown in this module. In this Registration module there are different alert messages like:

- You are successfully registered.
- Invalid password.
- Invalid phone no
- Password should have at least 1 number and 1 special symbol.
- Invalid Username.

REPORTS:

Reports are useful to know the feedback of our application . Based on the registration module we can maintain the reports i.e, number of users using this application. We can have their contact

details like name , phone no , email id and address which are useful for knowing the working or feedback of the application.

2. TRANSLATION:

The translation module deals with the translation of the user language into the language understand by the server. i.e. the language in which database is stored. In this project the main module is translation module which deals with the translation of the given input into the server understandable language.

PROVISIONS:

In translation module to translate into the required language we should provide the details like :

- Number of languages
- Name of languages

FUNCTIONALITIES:

In translation module we need to perform different functionalities to translate the language into server understandable language or user understandable language.

The functionalities are as follows:

- Selection of language.
- Translating from user language to language stored in database.
- Translating from server understandable language to user language.

QUERIES:

While running this module, various queries are raised. The queries raised in the translation module are:

- What are different languages used for the application?

- What are the number of languages used?
- How to translate the user language to English?
- How to translate the language in database to user language?

ALERTS:

In translation module, while translating it checks whether the language is having respective virtual keyboard or not to give the input by the user.

The various alert messages possible in the translation module are as given below:

- Select the language available in list box.
- The selected language is not available.
- Translation is not possible.

Similarly, different alert messages are shown while translating from user language to the language in database or vice versa.

REPORTS:

In this translation module we can maintain the reports based on user selection. The reports need to be maintained to know the feedback. We can know the details like:

- Most Frequently selected language.
- Can know the location of user based on user selection.
- To which kind of users it is most useful?

3. DEVELOPMENT OF VIRTUAL KEYBOARD:

This module plays a key role in the project. The virtual keyboard is the keyboard with the letters of the user selected language. The development of virtual keyboard plays a keen role for developing the project.

PROVISIONS:

To develop a virtual keyboard in the user given language ,we need to know about the number of letters and the grammar of the respective language should be known. The following details should be provided:

- Number of letters to be used in virtual key board in respective language.
- Number of syllables to be used in virtual key board in respective language.
- Letters to be kept in virtual keyboard.
- Syllables to be kept in virtual key board.

FUNCTIONALITIES:

The main functionality of the virtual keyboard is to run the application by giving the input . Whenever the language is selected , the respective virtual keyboard should be displayed. The functionalities are as follows:

- Development of virtual keyboard in a particular script.
- Input to be given by user.

The input to the application can only be given by the designed virtual keyboard in respective language. After giving input through virtual keyboard the server can respond and can show the desired results.

QUERIES:

While developing the virtual keyboard we may have many queries i.e. how to develop and based on which conditions which are as follows:

- Which script is used for developing the virtual key board?
- How many different scripts are available to develop a virtual keyboard?
- What are different letters to be kept in virtual key board?
- What are different symbols to be kept in virtual key board?

ALERTS:

While developing the virtual key board and giving input to the application, the various alert messages displayed are:

- Selected language is not available in list box.
- Press the key properly.
- Give input using virtual keyboard.

REPORTS:

By reports we can maintain different results or feedback based on the usage of the application or number of users using which language.

- Script used for development of virtual keyboard.
- Frequently selected language by the users.
- Mostly visited websites.

4. DATA RETRIEVAL:

Data retrieval is the next step after giving the input using the virtual keyboard, then the result of the given input should also be displayed in user selected language. This can be done by this data retrieval module. It performs many functionalities which are discussed briefly as follows:

PROVISIONS:

For retrieving data from the database we need to maintain database in the server. After giving input, the given data should be translated. We need some provisions like:

- Data should be available in database.
- Should be able to retrieve data.

FUNCTIONALITIES:

The retrieval module is mainly used for displaying the results for the given input by retrieving data from database. The various functionalities of the data retrieval module are as follows:

- Input given in the user language should be translated to English.
- Data with respect to the user input should be retrieved from the database.
- Data retrieved should be translated to user language.
- Results after translating should be displayed to the user.

To translate the English language to user language or the user language to English, there should be some translator. It should be able to enough to translate into any language by maintaining database in the form of dictionary.

QUERIES:

While retrieving data from the database we need to check whether data retrieved is related to the given input or not. The following queries are raised:

- Data retrieved is related content or not?
- Is the given input correct or not?

ALERTS:

The alert messages displayed for the data retrieval module are as follows:

- Data not found.
- Invalid input.
- Data inadequate.

REPORTS:

The reports need to be maintained for any application. The results should be displayed. The reports show us the usage of the application and various types of users using the application. By reports we can maintain feedback which is useful for knowing the usage of the application.

There are two modules in the basic framework of two-way interaction. They are:

1. IHDD module

2. RTR module

1. IHDD module:

The IHDD (Input Handler and Data Dispatcher) module is responsible for converting users input from language L to English and forward them to original web server. The module has been subdivided into two sub modules namely Input Handler and Data Dispatcher.

The Input Handler first extracts the input data from web page in language L and then converts the data from that language to English. The module then invokes the Data Dispatcher which performs the authentication needed to access the web site and finally regenerates the query to be posted to original web server in English. The working of the module is described as follows. User enters the URL of a web page what he wants to get serviced

2. RTR module:

The RTR (Retrieve, Translation and Render) module searches the Internet for the web page with respect to request given by the user. After getting the requested page, it translates the retrieved web page into language L and renders the resulting web page in proper manner to the client machine. This module consists of three sub modules namely Retrieve, Translation and Render.

The Retrieve sub module retrieves a web page as requested by the user and separates the content of web page (by using HTML parser) into HTML tags and English text. Proper indexing of links in the page is done and maintained using a table called Index Table. The Index Table handles the

layout and links present in the original web page. The Translate sub module takes the extracted English text from the Retrieve sub module and converts the text to the language L. The Render sub module fully furnishes the web page in language L. Once text is converted, it merges the content in language L and set HTML tags using Index Table. In fact, the Render sub module recreates the web page in user's language having same look as the original web page.

Technical Design:

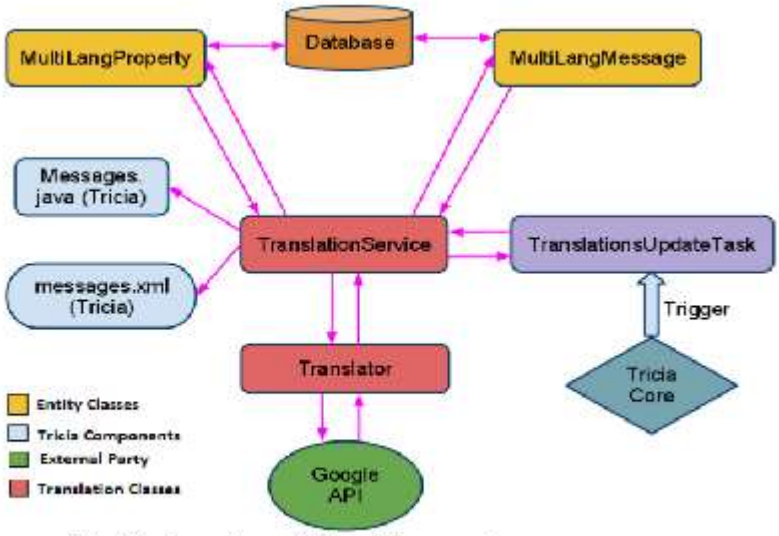

Fig:4.1 Overview of Traslation system

4.2 SOFTWARE REQUIREMENTS

Operating System: WINDOWS-XP/WINDOWS 7

Technology: JSP, JDBC, Servlets

Databases: Oracle 10g

Server: Apache Tomcat 6.0

Translator: Google Translator V2 API and Microsoft Translator

4.3 HARDWARE REQUIREMENTS

Processor	:	Pentium III 630MHz
RAM	:	128 MB
Hard Disk	:	20GB

SYSTEM DESIGN

5.1 DESIGN

Design is the first step in the development phase for any engineering product or system. It may be defined as a "the process of applying various techniques and principles for the purpose of defining a device, a process, or a system insufficient detail to permit its physical realization".

Software design is an iterative process through which requirements are translated into a "blue print" for the constructing software. The design is represented at high level of abstraction, a level that can be directly translated to a specific data, functional behavior requirements. Preliminary design is concerned the transformation of requirements into a data and software architecture.

The design is solution, a "how to" approach to the creation of a new system. This is composed of several steps. It provides the understanding and procedural details necessary for implementing the system recommended.

The database design transforms the information domain model created during analysis into the data structures that will be required to implemented software.

The architecture design describes how the software communicates within itself, to systems that interoperate with it, and with humans who use it. An interface implements flow of information.

The interface design describes how the software communicates within itself, to systems that interoperate with it, and with humans who use it. An interface implements flow of information.

5.1.1 INPUT DESIGN

Input design is the process of converting user-oriented input to a computer based format. Input design is a part of overall system design, which requires very careful attention. Often the collection of input data is the most expensive part of the system. The main objectives of the input design are

- Produce cost effective method of input
- Achieve highest possible level of accuracy
- Ensure that the input is acceptable to and understood by the staff.

5.1.1.1 INPUT DATA

The goal of designing input data is to make entry easy, logical and free from errors as possible. The entering data entry operators need to know the allocated space for each field; field sequence and which must match with that in the source document. The format in which the data fields are entered should be given in the input form .Here data entry is online; it makes use of processor that accepts commands and data from the operator through a key board. The input required is analyzed by the processor. It is then accepted or rejected. Input stages include the following processes:

- Data Recording
- Data Transcription
- Data Conversion
- Data Verification
- Data Control
- Data Transmission
- Data Correction

One of the aims of the system analyst must be to select data capture method and devices, which reduce the number of stages so as to reduce both the changes of errors and the cost .Input types, can be characterized as:

- External
- Internal
- Operational
- Computerized
- Interactive

Input files can exist in document form before being input to the computer. Input design is rather complex since it involves procedures for capturing data as well as inputting it to the computer.

5.1.2 OUTPUT DESIGN

Outputs from computer systems are required primarily to communicate the results of processing to users. They are also used to provide a permanent copy of these result for latter

consultation .Computer output is the most important and direct source of information to the users. Designing computer output should proceed in an organized well throughout the manner. The right output must be available for the people who find the system easy o use. The outputs have been defined during the logical design stage. If not, they should defined at the beginning of the output designing terms of types of output connect, format, response etc, Various types of outputs are

- External outputs
- Internal outputs
- Operational outputs

- Interactive outputs
- Turn around outputs

All screens are informative and interactive in such a way that the user can fulfill his requirements through asking queries.

5.2 UML DIAGRAMS

Design is the first step in the development phase for an engineered product or system. Design is the place where quality is fostered in software development. Design is the only way that we can accurately translate a user's requirements into a finished software product or system. Software design serves as the foundation for all software engineers and software maintenance steps that follow. Without design we risk building an unstable design -one that will fail when small changes are made, one that may be difficult to test, and one whose quantity cannot be accessed until late in the software engineering process.

Taking software requirements specification document of analysis phase as input to the design phase we have drawn Unified Modeling Language (UML) diagrams. UML depends on the visual modeling of the system. Visual modeling is the process of taking the information from the model and displaying it graphically using some sort of standards set of graphical elements.

UML Diagrams are drawn using the Pace Star UML Diagrammed Software. We seem too able to understand complexity better when it is displayed to us visually as opposed to written textually. By producing visual models of a system, we can show how system works on several levels. We can model and the interactions between the users and the system.

Each UML diagram is designed to let developers and customers view a software system from a different perspective and in varying degrees of abstraction. UML diagrams commonly created in visual modeling tools include

5.2.1 USE CASE DIAGRAM

Use case Diagrams describe the functionality of a system and users of the system. Use cases are used during requirements elicitation and analysis to represent the functionality of the system. These diagrams contain the following elements:

- Actors, which represent users of a system, including human users and other systems.
- Use cases, which represent functionality or services provided by a system to user. Use case Diagrams address the static use case view of a system. These diagrams are especially important in organizing and modeling the behavior of a system.

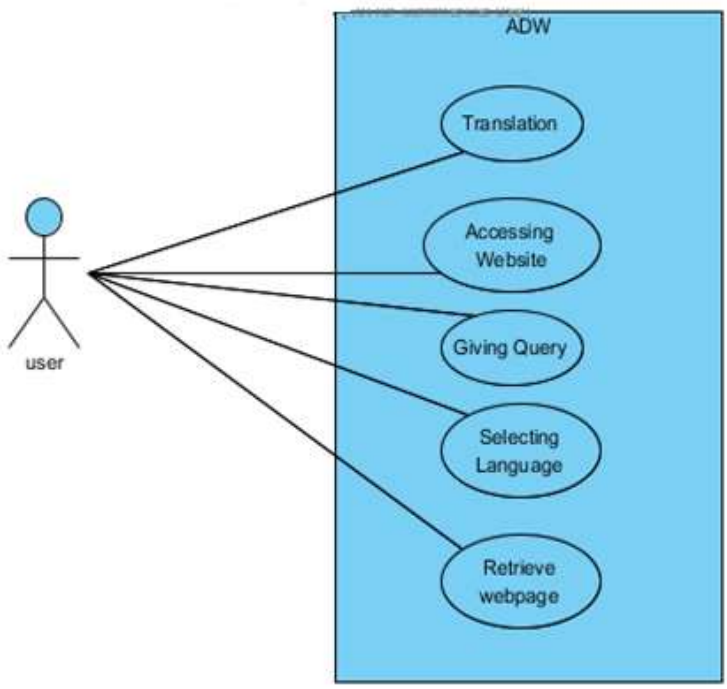

Fig 5.1: Use case Diagram

5.2.2 CLASS DIAGRAM

Class diagrams describe the static structure of a system, or how it is structured rather than how it behaves. These diagrams contain the following elements:

- Classes, which represent entities with common characteristics or futures. These features include attributes, operations and associations.
- Associations, which represent relationships that relate two or more other classes where the relationships have common characteristics or features, attributes and operations.

5.2.3 SEQUENCE DIAGRAM

An interaction diagram shows an interaction, consisting of a set of objects and their relationships, including the messages that may be dispatched among them.

Interaction diagrams address the dynamic view of the system. A sequence diagram is an interaction diagram that emphasizes the time ordering of messages

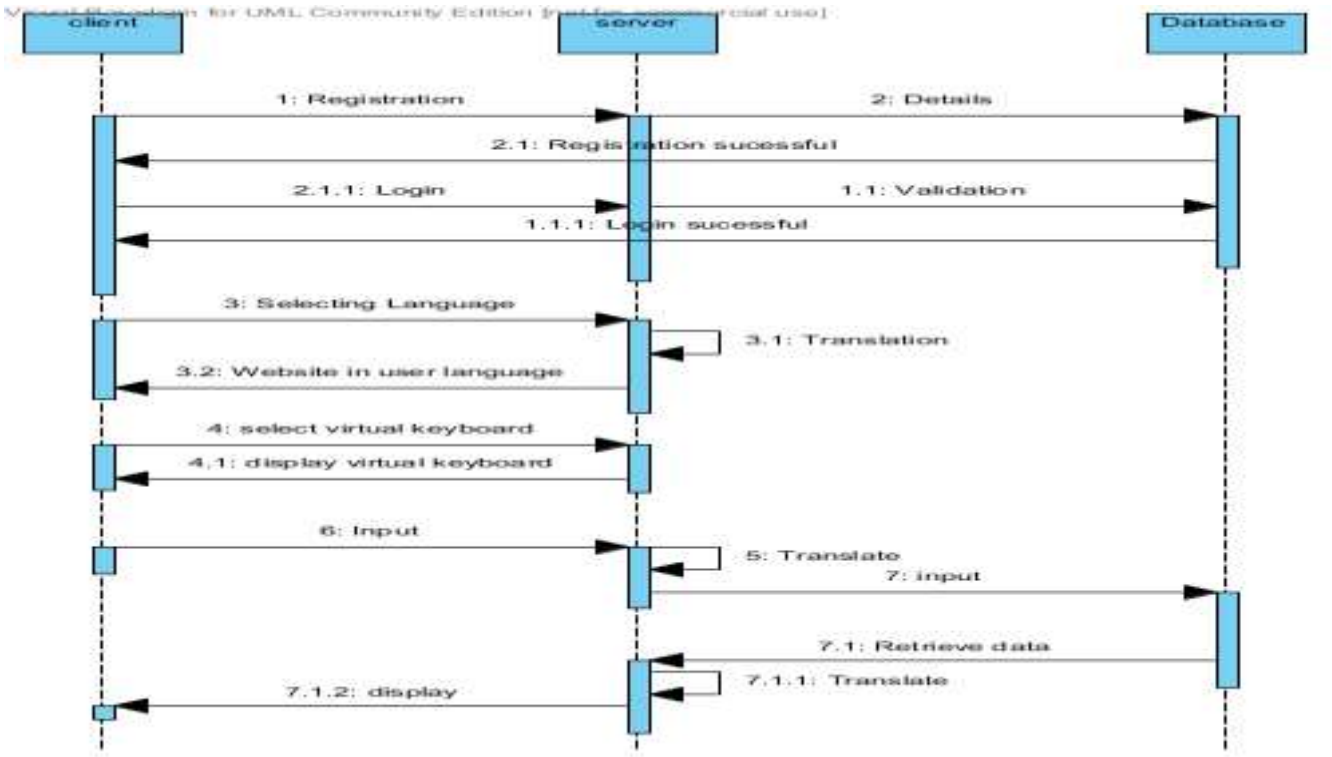

Fig 5.3.1: Sequence Diagram

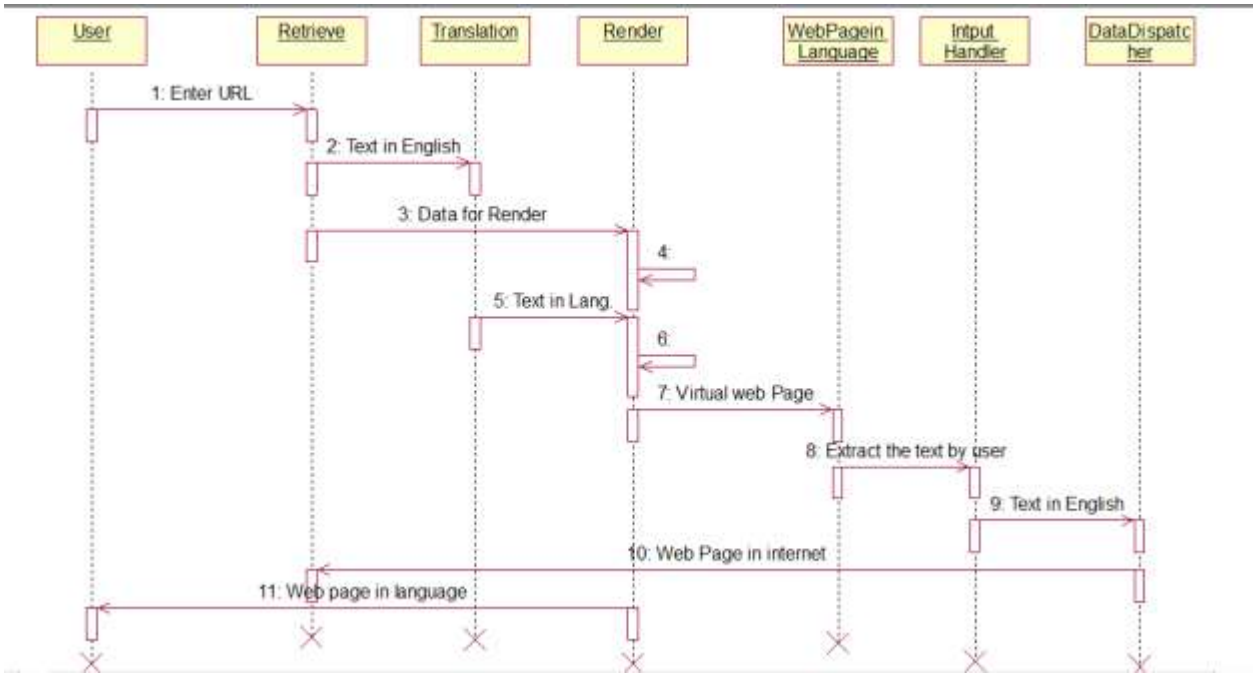

Fig 5.3.2: Sequence Diagram

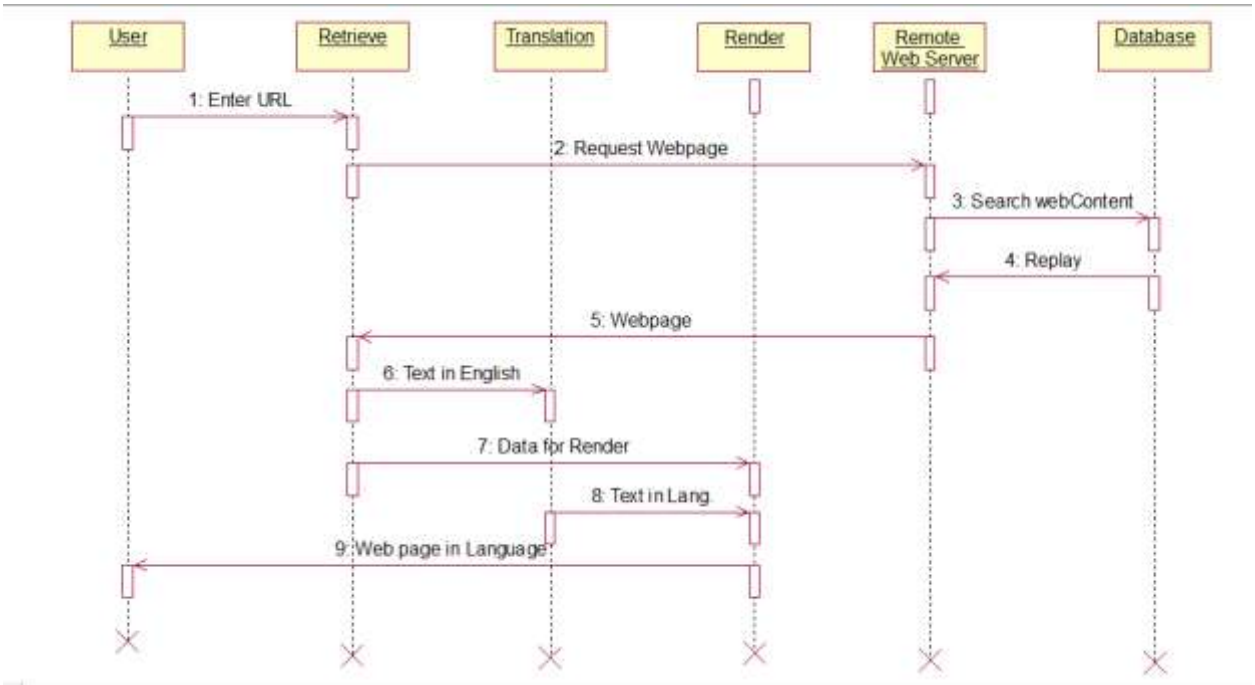

Fig 5.3.3: Sequence Diagram

5.2.4 ACTIVITY DIAGRAM

Activity Diagram displays a special state diagram where most of the states are action states and most of the transitions are triggered by completion of the actions in the source states. This diagram focuses on flows driven by internal processing.

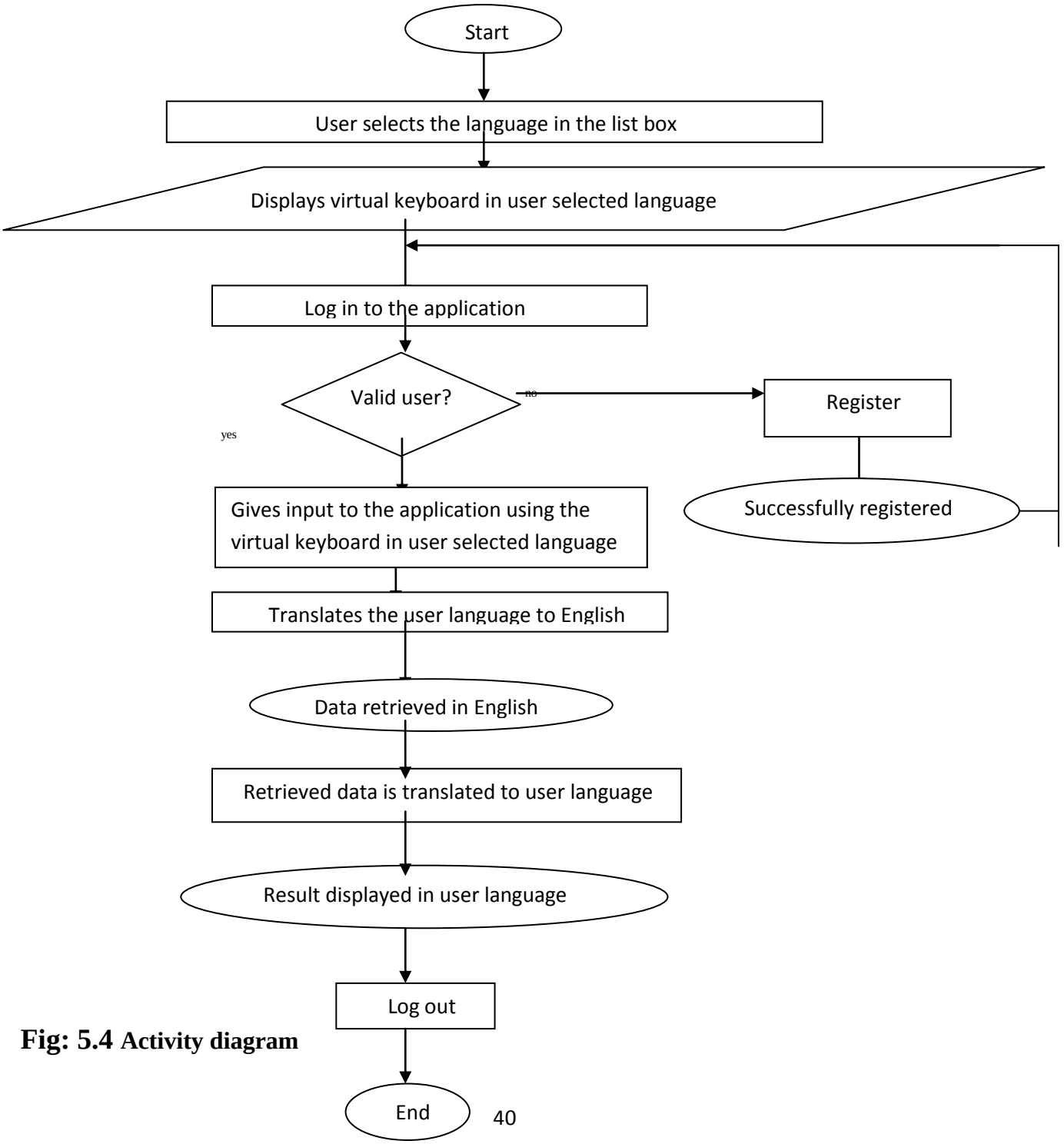

Fig: 5.4 Activity diagram

5.3 DATABASE DESIGN

The general theme behind a database is to handle information as an integrated whole. A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and effectively. After designing input and output, the analyst must concentrate on database design or how data should be organized around user requirements. During database design the following objectives are concerned:

- Controlled Redundancy
- Data independence
- Accurate and integrating
- More information at low cost
- Recovery from failure
- Privacy and security
- Performance
- Ease of learning and use

5.3.1 TABLE STRUCTURES

Table	Column	Data Type	Length
USER1	FIRSTNAME	Varchar2	20
	LASTNAME	Varchar2	20
	USERNAME	Varchar2	20
	PASSWORD	Varchar2	20
	EMAIL	Varchar2	20
	CONTACT	Number	-
	ADDRESS	Varchar2	100
	COUNTRYNAME	Varchar2	20
	STATE	Varchar2	20

Table 5.3.1: User Registration Form

Table	Column	Data Type	Length
BOOKING	PNR	Number	-
	TRAINNAME	Varchar2	20
	BCLASS	Varchar2	20
	BDATE	Varchar2	20
	TRAINNO	Number	-
	BFROM	Varchar2	20
	BTO	Varchar2	20
	BOARDING	Varchar2	20
	UPTO	Varchar2	20
	QUOTA	Varchar2	20
	SNO	Number	-
	NAME	Varchar2	20
	AGE	Number	-
	SEX	Varchar2	20
	BERTH	Varchar2	20
	TOTAL	Number	-

Table 5.3.2 : Ticket Booking Form

Table	Column	Data Type	Length
TRAVEL	TFROM	Varchar2	20
	TTO	Varchar2	20
	TRAIN	Number	-
	TRAINNAME	Varchar2	20
	ORGIN	Varchar2	20
	DEPTTIME	Varchar2	40
	DESTINATION	Varchar2	20
	ARRIVAL	Varchar2	40
	TRAVELTIME	Varchar2	40
	TOTAL	Number	-

Table 5.3.3: Travelling Details

TFROM	TTO	TRAIN	TRAINNAME	ORGIN	DEPTTIME	DESTINATION	ARRIVAL	TRAVELTIME	TOTAL
guntur	hyderabad	12334	narsapur	swarupa	10	hyderabad	guntur	10	500
guntur	bombai	12335	palnadu	guntur	5am	bombai	9pm	16	700
guntur	mumbai	12335	palnadu	guntur	5am	mumbai	9pm	16	700
hyderabad	delhi	6543	himsaexpress	kanyakumari	5pm	kashmir	9am	16	2000
gandhinagar	bhuwaneswari	98634	AzadHindExpress	gujarath	9pm	ArunachalPradesh	6am	9	1000

Table 5.3.4: Information about Trains

SYSTEM IMPLEMENTATION

.

A dynamic web page is a page that changes based on the user. It responds to the user's needs, and provides relevant information to meet them, by accessing information in a connected database. A dynamic web page allows users to go beyond reading text and looking at graphics. It allows for an interactive experience, with the user being in control of the information he views.

Thus the dynamic web pages maintain up to date information and provides a two - way interaction with server and can retrieve results.

This problem had been addressed and then we can access static web page in any language required by user with the help of Google Translate. But , by static web pages user can't give input to the web page in his own language . To overcome , this paper has been proposed to access dynamic web page in user language.

This service provides the Internet content to millions of people who might not have good capability to read the web content in English . It will minimize the gap between the Internet and user due to language barrier. It helps the users to access Internet in their day to day life without worrying much about the language web page is originally written.

6.1 PROJECT MODULES:

1. USER REGISTRATION

2. TRANSLATION

3. DEVELOPMENT OF VIRTUAL KEYBOARD

4. DATA RETRIEVAL

6.2 SAMPLE CODE:

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
   <head>
      <script src="valid.js" type="text/javascript"></script>
      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
      <title>register</title>
                  <script type="text/javascript" src="http://www.google.com/jsapi"></script>
   <script type="text/javascript" src="http://www.google.com/jsapi"></script>
   <script type="text/javascript">

    // Load the Google Onscreen Keyboard API
    google.load("elements", "1", {
       packages: "keyboard"
    });

    function onLoad() {
      var kbd1 = new google.elements.keyboard.Keyboard(
```

```
        [google.elements.keyboard.LayoutCode.HINDI],
        ['t1','t2','t3','t4','t5','t6','t7','t8','t9','t10']);

      /* var kbd2 = new google.elements.keyboard.Keyboard(
        [google.elements.keyboard.LayoutCode.ARABIC],
        ['ar1', 'ar2']);*/
      }

     google.setOnLoadCallback(onLoad);
   </script>
<script src="valid.js" type="text/javascript"></script>

   </head>
   <body bgcolor="gray">
      <HR/>
      <div align="center"> <b><FONT color="#990000" size="4">INDIAN RAILWAYS REGISTRATION FORM</font></b></div>
      <HR/>
      <br>
      <div align="right"/>
 <a href="index.jsp">Home</a>
```

```
</div>
<form name="userform" action="user.jsp" method="post"  onsubmit="return validate();">
<table width="350px" border=0 align="center" bgcolor="#EED45F">
    <tr>
        <td colspan=2 style="font-weight:bold;font-size:20pt;" align="center">User Registration</td>
    </tr>
    <tr>
        <td colspan=2> </td>
    </tr>
    <tr>
        <td>First Name </td>
        <td><input type="text" name="firstName" value="" id="t1" ></td>
    </tr>
    <tr>
        <td>Last Name </td>
            <td><input type="text" name="lastName" value="" id="t2"</td>
    </tr>
    <tr>

        <td>Login Name</td>
```

```
        <td><input type="text" name="userName" value="" id="t3"></td>
         </tr>
<tr>
        <td>Password</td>
        <td><input type="password" name="password" value="" id="t4"></td>
</tr>
<tr>
        <td>Confirm Password</td>
        <td><input type="password" name="confirmPassword" value="" id="t5"></td>
</tr>
<tr>
        <td>Email</td>
        <td><input type="text" name="email" value="" id="t6"></td>
</tr>
<tr>
        <td>Contact No</td>
        <td><input type="text" name="contactNo" value="" id="t7"></td>
</tr>
<tr>
        <td>Address</td>
```

```
            <td><textarea  name="address" rows=5 cols=25 id="t8"></textarea></td>
        </tr>
        <tr>
            <td>Country Name</td>
            <td><input type="text" name="countryName" value="" id="t9"></td>
        </tr>
        <tr>
            <td>State</td>
            <td><input type="text" name="state" value="" id="t10"></td>
        </tr>
        <tr>
            <td></td>
            <td><input type="submit" name="Submit" value="Save User"></td>
        </tr>
        <div align="right" id="google_translate_element"></div><script>
function googleTranslateElementInit() {
  new google.translate.TranslateElement({
    pageLanguage: 'en',
    layout: google.translate.TranslateElement.InlineLayout.SIMPLE
  }, 'google_translate_element');
}
```

```
</script><script
src="//translate.google.com/translate_a/element.js?cb=googleTranslateElementInit"></script>


</table>

</form>

    </body>

</html>


<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<%@ page language="java" import="java.sql.*;"%>

<html>

<head>

<script type="text/javascript" src="https://www.google.com/jsapi=INSERT-YOUR-KEY"></script>

    <script type="text/javascript">


      // Load the Google Onscreen Keyboard API

      google.load("elements", "1", {

          packages: "keyboard"

      });
```

```
    function onLoad() {
      var kbd = new google.elements.keyboard.Keyboard(
        [google.elements.keyboard.LayoutCode.HINDI],
        ['t1','t2']);
    }
    google.setOnLoadCallback(onLoad);
   </script>
</head>
<body>
<%
 String userName = request.getParameter("userName");
 String password = request.getParameter("password");
%>
 <script type="text/javascript">
var Translate={
 baseUrl:"http://api.microsofttranslator.com/V2/Ajax.svc/",
 appId:"F1B50AB0743B541AA8C07089042D7B57E9B28D25",
 translate:function(text,from,to,callback){
 var s = document.createElement("script");
 s.src =this.baseUrl+"/Translate";
 s.src +="?oncomplete="+callback;
```

```
  s.src +="&appId="+this.appId;
  s.src +="&from" + from ;
  s.src += "&to=" + to ;
  s.src += "&text=" + text;
  document.getElementsByTagName("head")[0].appendChild(s);
 }
}
var userNam=function(userName){document.getElementById('boldStuff2').value = userName;};
Translate.translate("<%=userName%>","hi","en","userNam");
var passwor=function(password){document.getElementById('boldStuff3').value = password;};
Translate.translate("<%=password%>","hi","en","passwor");
</script>
 <div align="right" id="google_translate_element"></div><script>
function googleTranslateElementInit() {
  new google.translate.TranslateElement({
    pageLanguage: 'en',
    layout: google.translate.TranslateElement.InlineLayout.SIMPLE
  }, 'google_translate_element');
}
</script><script src="//translate.google.com/translate_a/element.js?cb=googleTranslateElementInit">
```

```
</script>

<form method="post" action="logimid.jsp"+fn id='blodStuff'>

<input type='hidden' id= 'boldStuff2' name='hd2' value='' />

<input type='hidden' id= 'boldStuff3' name='hd3' value='' />

  <input type="submit" name="Submit" value="Submit">

</form>

</body>

</html>

        <%@page contentType="text/html" pageEncoding="UTF-8"%>
        <script type="text/javascript" src="http://www.google.com/jsapi"></script>
        <script type="text/javascript">
          // Load the Google Onscreen Keyboard API
          google.load("elements", "1", {
              packages: "keyboard"
          });
          function onLoad() {
            var kbd1 = new google.elements.keyboard.Keyboard(
              [google.elements.keyboard.LayoutCode.HINDI],
              ['th1', 'th2']);
           /* var kbd2 = new google.elements.keyboard.Keyboard(
              [google.elements.keyboard.LayoutCode.ARABIC],
              ['ar1', 'ar2']);*/
          }
          google.setOnLoadCallback(onLoad);
        </script>
```

```
<!DOCTYPE html>
<html>
   <head>
      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
      <title>JSP Page</title>
   </head>
   <body bgcolor="gray">

      <HR/>
    <table width="100%" border="0" cellspacing="0" cellpadding="0">
      <tr>
        <td width="280" align="center" valign="middle"><img src="rail_logo_new_red.gif"
vspace="2" alt="Rail Logo"/></td>
      <td width="734" align="center" valign="center">
           <b><FONT color="#990000" size="4">INDIAN RAILWAYS PASSENGER
RESERVATION ENQUIRY</font></b></td>
      <td align="right"><img src="india.gif"/></td>
    </tr>
</table><HR/>
<div align="right" id="google_translate_element"></div><script>
function googleTranslateElementInit() {
  new google.translate.TranslateElement({
    pageLanguage: 'en',
    layout: google.translate.TranslateElement.InlineLayout.SIMPLE
  }, 'google_translate_element');
}
</script><script
src="//translate.google.com/translate_a/element.js?cb=googleTranslateElementInit"></script>
```

```
  <form action="find2.jsp" nethod="post"><br/><br/><br/><br/>
    <table width="200" border="0" align="center" bgcolor="#DDFF44">
<tr>
  <td>From:-</td>
  <td><select NAME="selfrom">
        <option>vijayawada</option>
        <option>guntur</option>
                <option>Karimnagar</option>
                <option>rajamundry</option>
                <option>hyderabad</option>
                <option>gandhinagar</option>
                <option>pune</option>
     </select>
 </td>
</tr>
<tr>
  <td>To:-</td>
  <td><select name="selto">
        <option>secunderabad</option>
        <option>warangal</option>
                <option>Tirupati</option>
                <option>hyderabad</option>
                <option>mumbai</option>
                <option>delhi</option>
                <option>pune</option>
                <option>vizag</option>
                <option>bhuvaneswari</option>
     </select>
  </td>
</tr>
```

```
<tr>
  <td>Date:-</td>
  <td>
    <input type="text" name="txtdate" id="txtdate" />
 </td>
</tr>
<tr>
  <td>Ticket:-</td>
  <td>
    <select name="txttick" id="txttick">
      <option>Sleeper</option>
      <option>First Class</option>
      <option>2 AC</option>
      <option>3 AC</option>
    </select>
 </td>
</tr>
<tr>
  <td>Quota</td>
  <td>
      <input type="radio" name="radio" id="txtq" value="txtq" />
      Tatkal<br/>
     <input type="radio" name="radio" id="txtq" value="txtq" />
        General<br/>
        <input type="radio" name="radio" id="txtq3" value="txtq" />
      Ladies
</td>
</tr>
<tr>
```

```
    <td> <input type="submit" name="txtfind" id="txtfind" value="Find Train" />
    </td>
    <td><input type="reset" name="txtreset" id="txtreset" value="Reset" />
    </td>
  </tr>
</table>
</form>
    </body>
</html>
```

SYSTEM TESTING

Is the menu bar displayed in the appropriate contested some system related features included either in menus or tools? Do pull –Down menu operation and Tool-bars work properly? Are all menu function and pull down sub function properly listed?; Is it possible to invoke each menu function using a logical assumptions that if all parts of the system are correct, the goal will be successfully achieved .? In adequate testing or non-testing will leads to errors that may appear few months later.

This creates two problems:

- Time delay between the cause and appearance of the problem.
- The effect of the system errors on files and records within the system.

The purpose of the system testing is to consider all the likely variations to which it will be suggested and push the systems to limits.

The testing process focuses on the logical intervals of the software ensuring that all statements have been tested and on functional interval is conducting tests to uncover errors and ensure that defined input will produce actual results that agree with the required results. Program level testing, modules level testing integrated and carried out.

7.1 TESTING OBJECTIVE

The main aim of testing is to uncover a host of errors, systematically and with minimum effort and time. Starting formally, we can say, Testing is a process of executing a program with the intent of finding an error. A successful test is one that uncovers an as yet undiscovered error.

As he good test case is one that has a high probability of finding errors, if it exists. But there is one thing that testing cannot do testing cannot show the absence of defects it can only show that software defects are present.

As the test results are gathered and evaluated they begin to give a qualitative indication of the reliability of the software. If servers' errors are detected, the overall quality of the software is a natural suspect. If, on the other hand, all the errors, which are encountered, are easily modifiable, then one of the two conclusions can be made:

For the purpose of the current system we are assuming that in the event that errors that are easily modifiable points to the later possibility, since repeating the entire testing routine can be

very time consuming. What we propose to do instead is to get it tested by one or more persons who are not a part of the development team but is well versed with the subject and with the concept of software testing. If we cannot detect any serious errors, it will enable us to state with the more confidence that the software does actually conform to expected standards.

7.1.1 TEST PLAN

The importance of software testing and its implementations cannot be overemphasized. Software testing is a critical element of Software Quality Assurance and represents the ultimate review of the specifications, design and coding.

7.1.2 TESTING METHODOLOGIES

In order to uncover the errors present in the different phases we have the concept of levels of testing. The basic levels of testing are as shown below:

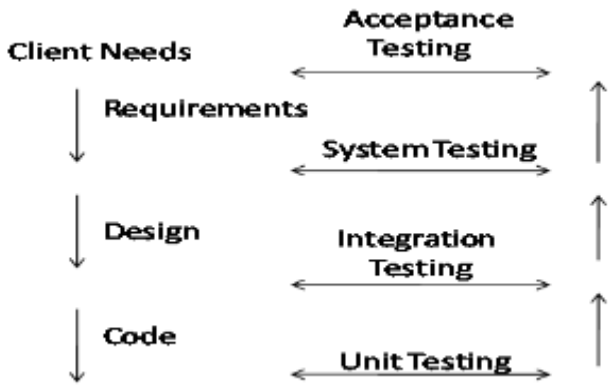

Fig 7.1:Various levels of Testing

7.2 TESTING METHODS

- Unit Testing.
- Integration Testing.
- Validation Testing.
- User Acceptance Testing.

7.2.1 UNIT TESTING

Unit testing is a software verification and validation method in which a programmer tests if individual units of source code are fit for use.

A unit is the smallest testable part of an application. In procedural programming a unit may be an individual function or procedure.

Ideally, each test case is independent from the others: substitutes like method stubs, objects, fakes and test harnesses can be used to assist testing a module in isolation.

- In Registration module, we provide registration for clerks, tags, racks and products. During the process of registration all the fields are mandatory. After registration, we check the permissions for entry into supermarket for all the clerks. Here we check whether the permissions are given correctly or not and if the permits enter the database correctly or not and we check if it works correctly or not.
- In Product Tracking, all the details of the product should be displayed when Product Id is entered. With these details the clerk can go for billing the product. If the Product Id is not present, it displays a message "Product not found".
- In Rack Tracking, using the Product Id, we check whether a particular product is appropriately placed in it's associated rack or not. If the product is misplaced, an alert gets generated.
- Billing module is done along with product tracking. Clerk upon tracking the product will generate bill. Here we allocate a Bill Id for each bill. Only the cost of all the products with same Bill Id must be added. This module ends with an indication that "Bill is generated Successfully".
- In Stock Details module, we keep track of all the products that are sold. Here, we have to retrieve the details of all the products that are sold on a particular date.
- In Expiry Products module, we maintain the details of all expired products in the supermarket. As the application starts loading, we must be able to get all the expiry products. Here upon seeing the details of expiry product, we can remove it.

7.2.2 INTEGRATION TESTING

This testing is sometimes called Integration and Testing. Integration testing is the phase in software testing in which individual software modules are combined and tested as a group. It occurs after unit testing and before system testing. Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates and delivers as its output the integrated system ready for system testing. Here we combine all the modules into a group and test the results are correct or not with the help of tracking screens.

- In Product Registration, we integrate the Product Ids with the Tagnos in the Tag Registration, so that only registered Tagnos can be used for Product Registration.
- We integrate Billing and Stock Details module so that whenever a bill gets generated, Stock Details must get updated.

7.2.3 VALIDATION TESTING

Validation Testing can be defined in many ways, but a simple definition is that validation succeeds when the software functions in a manner that can reasonably expected by a customer. After validation test has been conducted, the functions or performance characteristics confirm to specification and are accepted.

- Whenever a clerk or administrator logins, he/she must enter the valid username and password. Invalid username or password leads to login failure.
- In all the Registrations, all the fields must be filled.
- In Rack Tracking, when a product is misplaced in a rack, misplacement alert must be generated.

7.2.4 USER ACCEPTANCE TESTING

User acceptance of a system is a key factor of any system. The system under consideration is tested for the acceptance by constantly keeping in touch with the prospective system users at the same time of developing and marketing changes whenever required. This is done in regard to the following points:

- Input Screen Design.
- Output Screen Design

7.3 TEST CASES

7.3.1 TEST CASE FOR LOGIN

Test Case # : TC1	**Priority (H.L) : High**
Test Objective: To check whether Administration is validating the clerk or not.	
Test Description: In the Login screen, when the clerk enters invalid username or password, a message indicating "Login Unsuccessful" is displayed.	
Requirement Verified: Yes	
Test Environment virtual keyboard connected to system using comport API, Tested in n with the Net Beans with the help of database.	
Test Set Up/ Pre Condition: Connection should be established.	
Actions	**Expected Results**
In the Login screen, when the clerk enters invalid username or password, a message indicating "Login Unsuccessful" is displayed.	If the clerk enters a valid username and password, then he is given access to do the necessary services.
Pass: Yes	
Problems / Issues: None	
Notes: Successfully Tested and Executed.	

Table 7.1: Test case for Login

SAMPLE SCREENS

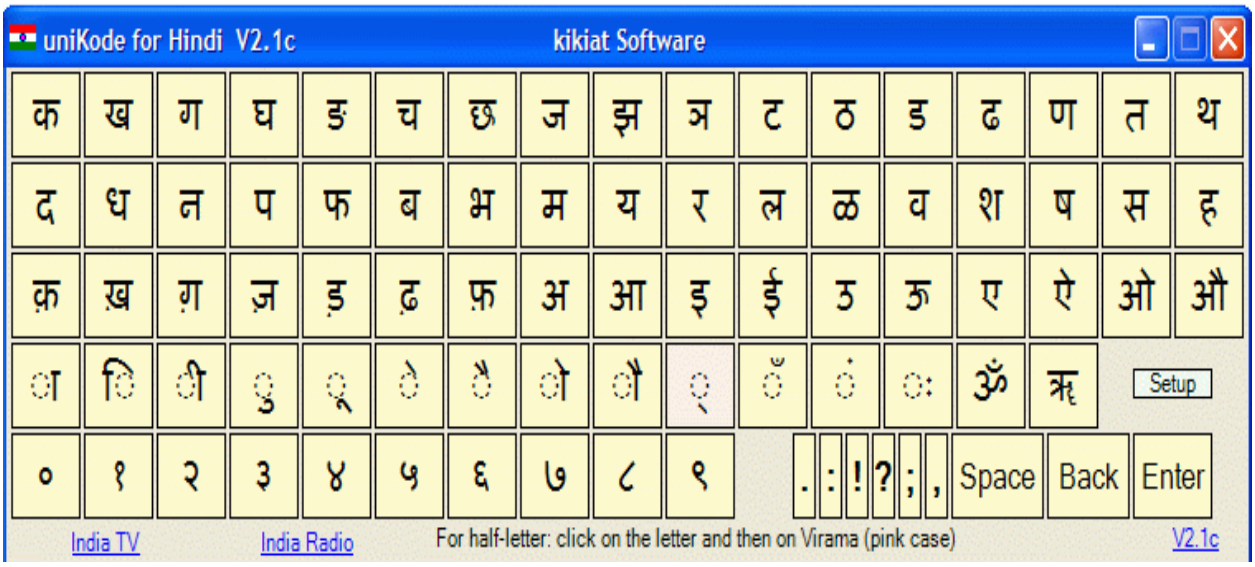

Fig 8.1: Sample Virtual Keyboard in Hindi

Fig8.1.1: Original content of IR-PRS

Fig8.2: Entering input in Hindi to virtual web page of IR-PRS

Fig8.3: original Results returned by server

Fig8.4: Displaying the desired result in Hindi

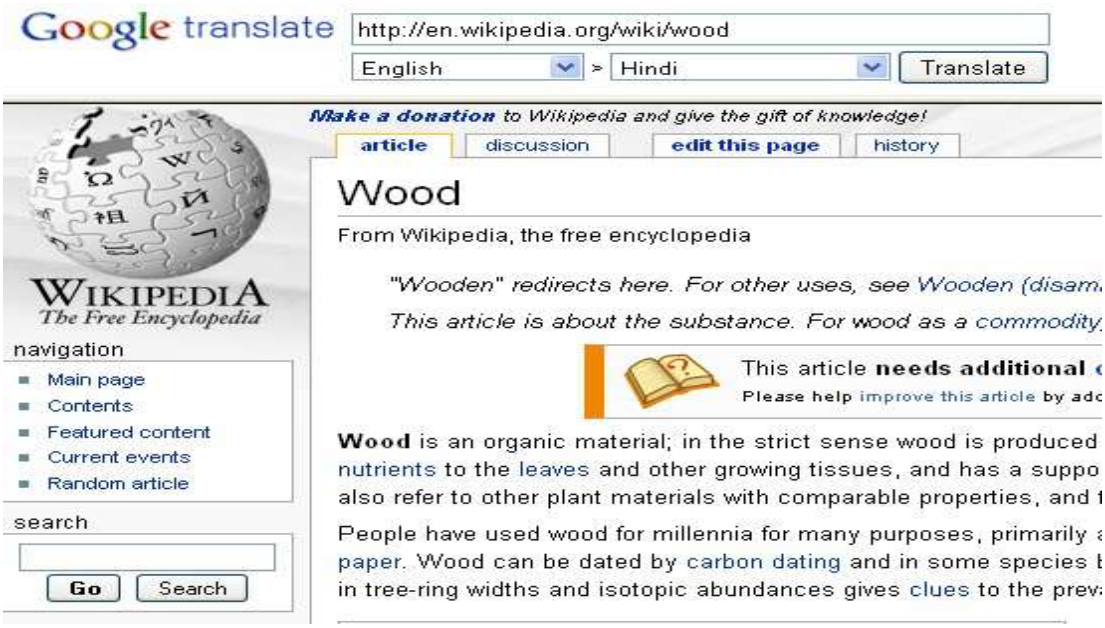

Fig 8.5 : Actual results returned by server

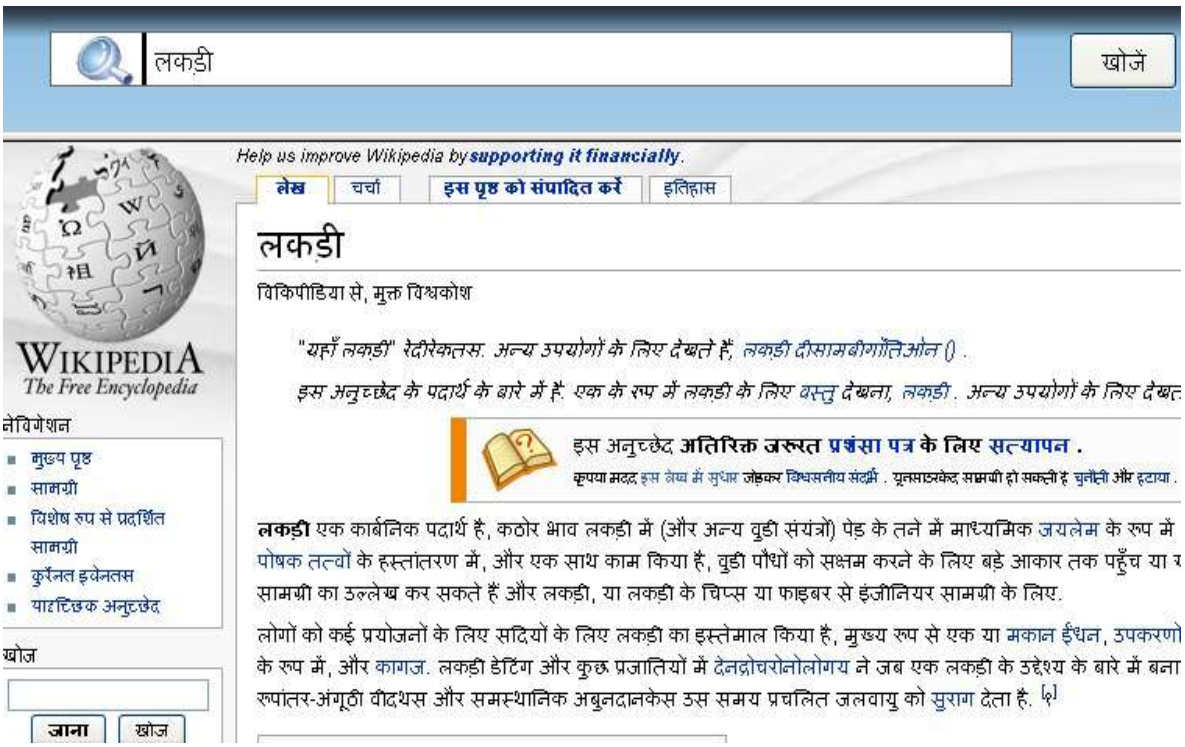

Fig 8.6: Desired Results in Language Hindi

Fig8.7: Accessing Dynamic web page of Andhra transport in English language

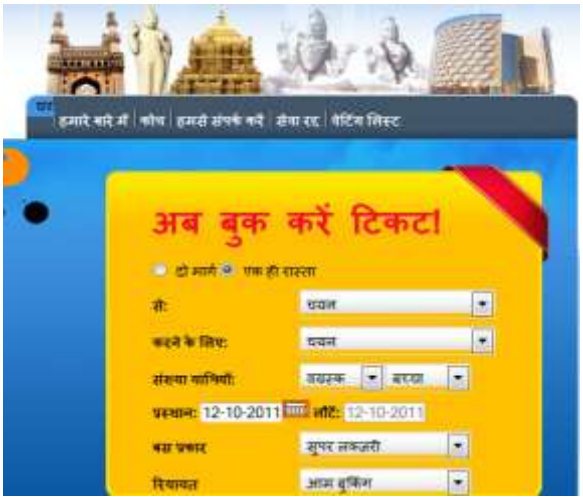

Fig8.8: Accessing Dynamic web page of Andhra transport in Hindi language

CONCLUSION

- Accessing Dynamic Web Page in Users Language plays a great role, as it will minimize the gap between the Internet and user due to language barrier.
- This service provides the Internet content to millions of people who might not have good capability to read the web content in English .
- It will minimize the gap between the Internet and user due to language barrier.
- It helps the users to access Internet in their day to day life without worrying much about the language web page is originally written.
- The user can give input in the form of voice based communication rather than using virtual keyboard. By voice based communication the stress on user can be reduced**.**
- The user can give input by using touch screen keyboard which should be designed in respective language.

REFERENCES

10.1 BOOKS AND JOURNALS

[1] "Unesco, international literacy statistics a review of concepts, methodology and current data," http://unesdoc.unesco.org/images/0016/001 628/162808e.pdf .

[2] "India world's second largest English speaking country," http://tesolindia. c.in/ English Teaching Industry/India-worlds-second-large stenglish- speaking-country

[3] "Languages and cultures on the internet study 2007," http://dtil.unilat.org/LI/2007/ro/resultados ro.html.

[4] "Wikipedia, dynamic web page," http://en.wikipedia.org/wiki/ Dynamic web page

[5]"Google translate ,"http://translate.google.com

[6] "Translate a block of text," http://in.babelfish.yahoo.com

[7] "World lingo, free website translator," http://www.worldlingo.com/en/websites/url translator.html **[8]**"Free text translation ,"http://www.freetranslation.com

[9] "Free automatic translators, machine translations comparison tests," http://www.humanitas-international.org/newstran/more-trans.htm

[10] "Language translation, translate phrase or word," http://www.translation.langenberg.com

[11] "Websites translator, translate your website from english," http://www.websitestranslator.com 2010).

[12] "Taming the beast, web page language translation,"http://www.tamingthebeast.net/articles6/page-languagetranslation.html

[13] "Ability, website translation and localization," http://www.localization-translation.com/translation-ocalizationservices/ web-sites-localization.html

[14] Google translate api documentation, URL http://code.google.com/apis/

language/translate/overview.html.

[15] Nereid documentation. URL http://www.matthes.in.tum.de/wikis/nereid/home.

[16] Tricia developer documentation,. URL http://infoasset.de/wikis/infoasset/tricia- developer-documentation .

[17] Tricia framework, . URL http://www.infoasset.de/wikis/infoasset/tricia .

10.2 WEBSITES

http://translate.google.com

http://www.freetranslation.com/

http://www.websitestranslator.com

http://www.ijser.org/researchpaper%5CAccessing-of-Dynamic-Web-Page-in-Users-Language.pdf