SUPERVISED TERRITORY VARIATION FOR CONDITION VIA EXTRACTION MULTIVOCAL SYNERGY TRANSMISSION

A PROJECT REPORT

Submitted by

V.SANGEETHA (212619104013)

S.VIJAYAMURUGAN (212619104019)

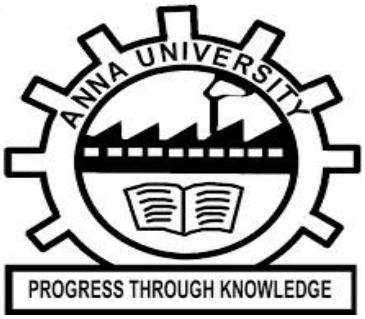
in partial fulfilment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING



SRI MUTHUKUMARAN INSTITUTE OF TECHNOLOGY

CHENNAI 600 069

ANNA UNIVERSITY: CHENNAI 600 025

MAY 2023

BONAFIDE CERTIFICATE

Certified that this Project Report "**SUPERVISED TERRITORY VARIATION FOR CONDITION VIA EXTRACTION MULTIVOCAL SYNERGY TRANSMISSION"** is the bonafide work of "Ms.V.SANGEETHA (212619104013) and Mr.S.VIJAYAMURUGAN (212619104019)" who carried out the project work under my supervision.

SIGNATURE

Dr.D.RAJINIGIRINATH, M.Tech,Ph.D.,

HEAD OF THE DEPARTMENT

Professor

Department of Computer Science and Engineering

Sri Muthukumaran Institute of Technology,

Chikkarayapuram, Chennai-600 069

SIGNATURE

Mrs.G.SANDRA KARUNYA M.E.Ph.D.,

SUPERVISOR

Assistant Professor

Department of Computer Science and Engineering

Sri Muthukumaran Institute of Technology,

Chikkarayapuram, Chennai-600 069

Submitted for the Anna University Project Viva Voice held on _____

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

We wish to express our sincere gratitude to our beloved Managing Trustee **Mrs Gomathi Radhakrishan** and our beloved Principal **Dr. K. Somasundaram, M.E, Ph.D.,** for providing us an adequate infrastructure and congenial academic environment.

We thank our respected Vice Principal **Dr.V.Anitha, M.E, Ph. D.,** for her encouragement and support in doing the project.

We express our deepest gratitude to our respectable Head of the Department and Project Co Ordinator **Dr.D.Rajinigirinath, M.Tech, Ph.D.,** Professor of Computer Science and Engineering whose encouragement has helped us in completing this project work.

We express our deepest gratitude to our respectable Project Supervisor **Mrs. G.Sandra Karunya, M.E, Ph.D., Assistant Professor** of Computer Science and Engineering for her special interest on our project and giving this consistent support and guidance during all the stages of this project work.

Finally, we thank all the teaching and non-teaching staff members of the Department of Computer Science and Engineering of our college who helped us to complete this project.

Above all we thank our parents, family members and my friends for their constant support and encouragement for completing this project.

ABSTRACT

This proposal, a secure data transmission technique from source to destination using the TCP/IP Protocol with IP Configuration. Satisfying the logic with the sophisticated blowfish and Logistic Regression algorithms in different occurrence in this proposed model of approach. The purpose of Blowfish technique is applied for the secure way of communication with data encryption technique in cryptographically approach which is using for the transformation through the TCP/IP protocol. In this proposed system, leverage symmetric encryption. The Encryption and decryption of data are accomplished using a single symmetric key generation for each appropriate data which is in unique format. Here, Logistic Regression Algorithm plays vital role in this part of Supervised Learning approach in this proposed model. It may use to satisfy the outcome of a user interface based on client objectives.

TCP/IP stands for transmission control Protocol/internet protocol. The connectivity between computers on the internet is governed by a set of rules. Using an IP address, data is routed through a network. TCP enables applications to demonstrate communication channels across a network, it also enables data to be fragmented into smaller packets before being broadcast over the Internet and then assembled appropriately at the destination Node**.** The IP address acknowledges Datagram of the domain and route to the appropriate address. This proposed system facilitates in the implementation of user interfaces based on client objectives and associated secure transmission via TCP/IP Protocol.

TABLE OF CONTENTS

CHAPTER	TITLE	PAGE NO
	ABSTRACT	**iv**
	LIST OF FIGURES	**viii**
	LIST OF ABBRIVATIONS	**ix**
1.	**INTRODUCTION**	
	1.1 DOMAIN SPECIFICATION	**2**
	1.2 OBJECTIVE	**3**
	1.3 SCOPE OF THE PROJECT	**4**
2.	**LITERATURE SURVEY**	
	2.1 EXISTING SYSTEM	**7**
	2.1.1 WEAKLY SUPERVISED DOMAIN ADAPTATION FOR ASPECT EXTRACTION VIA MULTILEVEL INTERACTION TRANSFER	**7**
	2.1.2 OPINION MINING AND SENTIMENT ANALYSIS	**8**
	2.1.3 MINING AND SUMMARIZING CUSTOMER REVIEWS	**9**
	2.1.4 OPINION WORD EXPANSION AND TARGET EXTRACTION THROUGH DOUBLE PROPAGATION	**9**
	2.1.5 RATED ASPECT SUMMARIZATION OF SHORT COMMENTS	**10**
	2.2 LIMITATIONS OF EXISTING SYTEN	**10**

2.3 PROPOSED SYSTEM 11

2.4 OBJECTIVES OF PROPOSED SYSTEM 11

3. SYSTEM REQUIREMENTS

3.1 SOFTWARE REQUIREMENTS 13

3.2 HARDWARE REQUIREMENTS 16

3.3 FUNCTIONAL REQUIREMENTS 17

3.4 NON-FUNCTIONAL REQUIREMENTS 17

3.5 PERFORMANCE REQUIREMENTS 18

4. SYSTEM DESIGN

4.1 ARCHITECTURE DIAGRAM 20

4.2 DATA FLOW DIAGRAM 21

4.3 USE CASE DIAGRAM 22

4.4 SEQUENCE DIAGRAM 23

4.5 CLASS DIAGRAM 24

4.6 ACTIVITY DIAGRAM 25

5. SYSTEM MODULES

5.1 MODULES IN PROPOSED SYSTEM 27

5.1.1 MODULE 1 : ADMIN 27

5.1.2 MODULE 2 : CLIENT 28

5.1.3 MODULE 3 : DESIGN **28**

5.1.4 MODULE 4 : DEVELOPMENT **29**

5.1.5 MODULE 5 : TESTING **30**

6. SYSTEM TESTING AND IMPLEMENTATION

6.1 APPROACH TO SOFTWARE TESTING **32**

6.2 UNIT TESTING **33**

7. SYSTEM SECURITY

7.1 SECURITY IN SOFTWARE **37**

7.1.1 CLIENT-SIDE VALIDATION **37**

7.1.2 SERVER-SIDE VALIDATION **38**

RESULT AND DISCUSSION **39**

CONCLUSION AND FUTURE ENHANCEMENT **44**

APPENDIX 1 **45**

REFERENCES **53**

PUBLICATION **54**

LIST OF FIGURES

CHAPTER	TITLE	PAGE NO
1.1	MACHINE LEARNING	4
5.1	ARCHITECTURE DIAGRAM	25
5.2	E-R DIAGRAM	26
5.3	USE CASE DIAGRAM	27
5.4	SEQUENCE DIAGRAM	28
5.5	CLASS DIAGRAM	29
7.1	LEVELS OF TESTING	37

LIST OF ABBRIVATIONS

ACRONYM	ABBRIVATION
JSP	JAKARTA SERVER PAGES (FORMALLY JAVA SERVER PAGES)
CGI	COMMON GATEWAY INTERFACE
JVM	JAVA VIRTUAL MACHINE
JDBC	JAVA DATABASE CONNECTIVITY
HTML	HYPER TEXT MARKUP LANGUAGE
SGML	STANDARD GENERALIZED MARKUP LANGUAGE

CHAPTER 1

INTRODUCTION

This approach, a secure data transfer method is used in TCP/IP Protocol with IP Configuration from source to destination. Conceptual scheme of methodology, the supporting the logic with complex blowfish and logistic regression methods in various instances. A set of guidelines control how computers connected to one another across the internet. Data is moved via a network using an IP address. Statistics may be broken up into tiny packages before being distributed across the Internet and then reconstituted correctly at the destination Host thanks to TCP, which also enables apps to show communication channels throughout a network. The component's goal is to transfer routing inside the same LAN. Even though to its standardized existence, TCP/IP is not supervised by a single firm. Therefore, the IP suite can be modified easily. As it works with every operating system, it can talk to any other system. Also, the IP suite works with all network architectures and computer systems TCP/IP can select the most effective way over the network since it is a routable protocol and is extremely scalable. It is widely used in current internet architecture.

The LAN is a local area network A network is comprised of two or more associated computers, whereas a LAN is a network that is contained into a small geographic area. A proposed model known as star networking is one in which each device is routed to a central hub. In this scenario, all devices are linked to a central place over the same LAN. The star topology is used in the TCP/IP protocol. The IP address acknowledges Network packets from the domain and directs them to the appropriate address. The architecture of user interfaces based on client goals and related secure transmission over TCP/IP Protocol is made easier by this proposed solution.

1.1 DOMAIN SPECIFICATION

CYBER SECURITY:

Cyber security is the practice of defending computers, servers, mobile devices, electronic systems, networks, and data from malicious attacks. It's also known as information technology security or electronic information security.

The Blowfish is a symmetric **encryption algorithm** created by Bruce Schneider in 1993. Symmetric encryption uses a single encryption key to both encrypt and decrypt data. The sensitive data and the symmetric encryption key are utilized within the encryption algorithm to turn the sensitive data into cipher text.

MACHINE LEARNING

Machine learning is an application of Artificial Intelligence (AI) that provides systems the ability to automatically learn and improve experience without being explicitly programmed. Machine learning focuses on the development of computer programs that can access data and use it learn for themselves. The primary aim is to allow the computer to learn automatically without human intervention or assistance and adjust actions accordingly.

The test for a machine learning model is a validation error on new data not a theoretical test that proves a null hypothesis. Because this often uses an iterative approach to learn from data, the learning can be easily automated.

MACHINE LEARNING METHODS

Some of the methods of Machine Learning algorithm are categorized as

1) SUPERVISED LEARNING

A Supervised learning algorithm learns from labelled training data, helps you to predict outcomes for unforeseen data. It is highly accurate and trustworthy method.

2) UNSUPERVISED LEARNING

Unsupervised learning algorithm is the type of self - organized with the help of previously unknown patterns in dataset without pre-existing labels.

3) SEMI-SUPERVISED LEARNING

Semi-supervised learning is the combination of both supervised and unsupervised which means labelled and unlabelled data.

4) REINFORCEMENT MACHINE LEARNING

Reinforcement machine learning is an area of machine learning concerned with how software agents ought to take actions in an environment so as to maximize some notion of cumulative reward.

APPLICATIONS OF MACHINE LEARNING

- Video Surveillance
- Social Media Services
- Email Spam and Malware Filtering
- Financial Services
- Health Care
- Retail
- Transportation

1.2 OBJECTIVE

The purpose of the proposed system would incorporate a user interface contingent on the client objective. Transferring the processed data via TCP/IP technique. We deploy IP addresses in the TCP/IP protocol to transmit processed data to the route IP address in an encrypted manner.

1.3 SCOPE OF THE PROJECT

The purpose of the proposed system would incorporate a user interface contingent on the client objective. Transferring the processed data via TCP/IP technique. We deploy IP addresses in the TCP/IP protocol to transmit processed data to the route IP address in an encrypted manner. The client will proffer the request to the server, and the server will provide the processed data to the client according on the client's request. The module objective is to exchange routing data across the same LAN. The LAN is a local area network A network is comprised of two or more associated computers, whereas a LAN is a network that is contained into a small geographic area. The star topology is utilized in the TCP/IP protocol, A network structure known as star networking is one in which each device is routed to a central hub, this scenario, all devices are connected to a central hub via the same LAN. This approach, we use the Blowfish algorithm to securely transmit data across a network. We prototype the user interface based on the client's objectives, we will ensure that the user interface is designed in accordance with the client's specifications by leveraging the Machine learning Logistic regression algorithm.

CHAPTER 2

2. LITERATURE SURVEY

2.1 EXISTING SYSTEM

The existing model while we are transferring the data there is no security provided to the data by the TCP/IP protocol while it is being transmitted over a network. When data is routed from client to server in the network, an unauthenticated intruder can led to rising significant data. The TCP/IP protocol does not protect associations from unauthorized access attacks. It does not endorse encryption and decryption while transferring data via the network. Before transmitting data to a router, TCP/IP will divide the data into datagram in a labeled manner. When one or more transmitted datagram refuse to reach their destination this is termed to as packet loss . TCP performs a negotiation between the listener and the sender when granting access. This diminishes configuration. TCP includes a sophisticated error-checking methodology. It primarily provides simple error-checking approaches through the use of checksums.

2.1.1 Weakly Supervised Domain Adaptation for Aspect Extraction via Multilevel Interaction Transfer, 2021 Tao Liang, Wenya Wang, and Fengmao Lv

It aims to identify the aspect terms (also known as opinion targets) of a product or service in each sentence. To learn a good aspect extraction model, an expensive annotation process is usually involved to acquire sufficient token-level labels for each domain, which is not realistic. To address this limitation, some previous works propose domain adaptation strategies to transfer knowledge from a sufficiently labeled source domain to unlabelled target domains. However, due to both the difficulty of fine-grained prediction problems and the large domain gap between different domains, the performance is still far from satisfactory.

Specifically, the aspect category information can be used to construct pivot knowledge for transfer with the assumption that the interactions between the sentence-level aspect category and the token-level aspect terms are invariant across domains. To this end, we propose a novel multilevel reconstruction mechanism that aligns both the fine- and coarse-grained information in multiple levels of abstractions. Comprehensive experiments over several benchmark data sets clearly demonstrate that our approach can fully utilize the sentence-level aspect category labels to improve cross-domain aspect term extraction with a large performance gain.

2.1.2 D. Ma, S. Li, X. Zhang, and H. Wang, "Interactive attention networks for aspect-level sentiment classification," in Proc. 26th Int. Joint Conf. Artif. Intell. (IJCAI), Aug. 2017, pp. 4068–4074.

Aspect-level sentiment classification aims at identifying the sentiment polarity of specific target in its context. Previous approaches have realized the importance of targets in sentiment classification and developed various methods with the goal of precisely modelling their contexts via generating target-specific representations. However, these studies always ignore the separate modelling of targets. In this paper, we argue that both targets and contexts deserve special treatment and need to be learned their own representations via interactive learning. Then, we propose the interactive attention networks (IAN) to interactively learn attentions in the contexts and targets and generate the representations for targets and contexts separately. With this design, the IAN model can well represent a target and its collocative context, which is helpful to sentiment classification.

2.1.3 L. Dong, F. Wei, C. Tan, D. Tang, M. Zhou, and K. Xu, "Adaptive recursive neural network for target-dependent Twitter sentiment classification," in Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics (ACL), 2019, pp. 49–54

Target-dependent sentiment analysis on Twitter has attracted increasing research attention. Most previous work relies on syntax, such as automatic parse trees, which are subject to noise for informal text such as tweets. In this paper, we show that competitive results can be achieved without the use of syntax, by extracting a rich set of automatic features. We split a tweet into a left context and a right context according to a given target, using distributed word representations and neural pooling functions to extract features. Both sentiment-driven and standard embeddings are used, and a rich set of neural pooling functions are explored.

2.1.4 G. Qiu, B. Liu, J. Bu, and C. Chen, "Opinion word expansion and target extraction through double propagation," Compute. Linguistics, vol. 37, no. 1, pp. 9–27, Mar. 2016.

Analysis of opinions, known as opinion mining or sentiment analysis, has attracted a great deal of attention recently due to many practical applications and challenging research problems. In this article, we study two important problems, namely, opinion lexicon expansion and opinion target extraction. Opinion targets (targets, for short) are entities and their attributes on which opinions have been expressed. To perform the tasks, we found that there are several syntactic relations that link opinion words and targets. We call it double propagation as it propagates information between opinion words and targets. A key advantage of the proposed method is that it only needs an initial opinion lexicon to start the bootstrapping process. Thus, the method is semi-supervised due to the use of opinion word seeds. In evaluation, we compare the proposed method with several state-of-the-art methods using a standard product review test collection.

2.1.5 Y. Lu, C. Zhai, and N. Sundaresan, “Rated aspect summarization of short comments,” in Proc. 18th Int. Conf. World Wide Web (WWW), 2018, pp. 131–140.

Web 2.0 technologies have enabled more and more people to freely comment on different kinds of entities (e.g., sellers, products, services). The large scale of information poses the need and challenge of automatic summarization. In many cases, each of the user-generated short comments comes with an overall rating. In this paper, we study the problem of generating a ``rated aspect summary'' of short comments, which is a decomposed view of the overall ratings for the major aspects so that a user could gain different perspectives towards the target entity. We formally define the problem and decompose the solution into three steps. We demonstrate the effectiveness of our methods by using eBay sellers' feedback comments. We also quantitatively evaluate each step of our methods and study how well human agree on such a summarization task. The proposed methods are quite general and can be used to generate rated aspect summary automatically given any collection of short comments each associated with an overall rating.

2.2 LIMITATIONS OF EXISTING SYSTEM

- It does not accommodate encryption and decryption while transferring data across a network.
- Data loss happens when one or more broadcast datagram’s refuse to reach their intended destination.
- Datagram that are acquired out of sequence while being transmitted over the network. There is no confidentiality given when transferring data.
- If the associated network crashes, the terminals affiliated are deactivated and cannot engaged in frame relay.

2.3 PROPOSED SYSTEM

The proposed system we propose the Encryption and Decryption technique while exchanging data over TCP/IP Protocol in the suggested system. In the process of transferring the data to client from server, the client sends request to the server and the server will authenticate the client after that the server transfer the data to the client by using the IP address and Port number. Before transferring data, it recognizes a network connection between the recipient and the sender. In this proposed system we cultivate the user interface based on client objectives, which will be synthesized using the software development life cycle aspect of the user interface will be compare client objective by using the logistic regression algorithm. If any data is lost after the transmission of data, the client will send an endorsement to the server The TCP/IP protocol possesses the star network and data transmission going to take place within the LAN network and also increasing the checksum in the destination part. The proposed system satisfies the Data transmission via the TCP/IP technique in the Secured manner.

2.4 OBJECTIVES OF PROPOSED SYSTEM

- The total process is monitored by the administration. It will help to prevent data control.
- The Blowfish algorithm will encrypt the data as it is transmitted to the route through the TCP/IP protocol.
- If the data get lost failing to reach their destination, they will be sent back to the sender from the receiver. The sender can retransmit the data.
- Before transferring data, it demonstrates a network connection between the client and server.
- The authenticated client is only permitted to receive data using an encrypted key.

CHAPTER 3

3. SYSTEM REQUIREMENTS

3.1 SOFTWARE REQUIREMENTS

The software requirements are description of features and functionalities of the target system. Requirements convey the expectations of users from the software product. The requirements can be obvious or hidden, known or unknown, expected or unexpected from client's point of view. List of software requirements for this project include:

- Backend : MySQL 5.1, JAVA / J2EE
- Frontend : core java , css, js, servlet
- Web application : J2ee Frameworks, Hibernate

- **Java**

Initially the language was called as "oak" but it was renamed as "Java" in 1995.The primary motivation of this language was the need for a platform independent (i.e. Architecture neutral) language that could be used to create software to be embedded in various consumer electronic devices.

- Java is a programmer's language.
- Java is cohesive and consistent.
- Except for those constraints imposed by the Internet environment. Java gives the programmer, full control.

Finally, Java is for Internet Programming where c was to System Programming.

- **Servlets/JSP**

A Servlet is a generic server extension. Java classes that can be loaded dynamically to expand the functionality of a server. Servlets are commonly used with web servers. Where they can take the place CGI scripts. A Servlet is similar to proprietary server extension, except that it runs inside a Java Virtual Machine (JVM) on the server, so it is safe and portable Servlets operate solely within the domain of the server. Unlike CGI and Fast CGI, which use multiple processes to handle separate program or separate requests, separate threads within the web server process handle all servlets. This means that servlets are all efficient and scalable.

Servlets are portable; both across operating systems and also across web servers. Java Servlets offer the best possible platform for web application development. Servlets are used as a replacement for CGI scripts on a web server; they can extend any sort of server, such as a mail server that allows servlets extend its functionality, perhaps by performing a virus scan on all attached documents or handling mail filtering tasks. Servlets provide a Java-based solution used to address the problems currently associated with doing server-side programming including inextensible scripting solutions platform-specific API's and incomplete interface.

- **JDBC**

Any relational database. One can write a single program using the JDBC API, and the JDBC is a Java API for executing SQL, Statements(As a point of interest JDBC are trademarks names and is not an acronym; nevertheless, JDBC is often thought of as standing for Java Database Connectivity. It consists of a set of classes and interfaces written in the Java Programming language.

Using JDBC, it is easy to send SQL statements to virtually program will be able to send SQL Statements to the appropriate database. The Combination of Java and JDBC lets a programmer writes it once and run it anywhere.

- **HTML**

Hypertext Markup Language (HTML), the languages of the World Wide Web (WWW), allows users to produce web pages that included text, graphics and pointer to other web pages (Hyperlinks).

HTML is not a programming language, but it is an application of ISO Standard 8879, SGML (Standard Generalized Markup Language), but Specialized to hypertext and adapted to the Web. The idea behind Hypertext one point to another point. We can navigate through the information based on out interest and preference. A markup language is simply a series of items enclosed within the elements should be displayed. Hyperlinks are underlined or emphasized works that load to other documents or some portions of the same document.

HTML can be used to display any type of document on the host computer, which can be geographically at a different location. It is a versatile language and can be used on any platform or desktop.

HTML provides tags (special codes) to make the document look attractive.
HTML provides are not case-sensitive. Using graphics, fonts, different sizes, colour, etc., can enhance the presentation of the document. Anything that is not a tag is part of the document itself.

- **Java Script**

JavaScript is a compact, object-based scripting language for developing client and server internet applications. Netscape Navigator 2.0 interprets JavaScript statements embedded directly in an HTML page. And Livewire enables you to create server-based applications similar to common gateway interface (CGI) programs.

For example, you can write a JavaScript function to verify that users enter valid information into a form requesting a telephone number or zip code. Without any network transmission, an Html page with embedded Java Script can interpret the entered text and alert the user with a message dialog if the input is invalid or you can use JavaScript to perform an action (such as play an audio file, execute an applet, or communicate with a plug-in) in response to the user opening or exiting a page.

3.2 HARDWARE REQUIREMENTS

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware, a hardware requirements list is often accompanied by a hardware compatibility list (HCL), especially in case of operating systems. An HCL lists tested, compatible, and sometimes incompatible hardware devices for a particular operating system or application. The following sub-sections discuss the various aspects of hardware requirements.

- Processor : Intel (R) Pentium (R)
- Speed : 1.6 GHz and Above
- RAM : 4 GB and Above
- Hard Disk : 120 GB
- Monitor : 15” LED SVGA
- Input Devices : Keyboard, Mouse

3.3 FUNCTIONAL REQUIREMENTS:

Following is a list of functionalities of the browsing enabled system.

- An Activity with a UI that allows you to browser settings. Provide a second Activity that allows users to access the share with permission from the administrator. Handle activity lifecycle appropriately. A precondition for any points in this part of the grade is code that compiles and runs.
- Your application should allow a user to browse the shares, buy and sell the shares with specific metadata. The assignment requires you to create a UI for browsing and a UI for integrating the two.
- The Net beans provide a number of useful layout components, views, and tools that you may want to use to create your location browser. As with the final project, you should design your application to only use the buttons on the Keyboard and mouse as input. Your application should use the Keyboard, Mouse and keywords.

3.4 NON-FUNCTIONAL REQUIREMENTS:

- The system should be supported Net beans. The member should use the System browser. Each member should have a separate system.
- The system should ask the username and password to open the application.
  It doesn't permit to unregistered user to access the System.
- The system should have Modular customization components so that they can be reused across the implementation.
- These are the mainly following:
- Secure access of confidential data. 24 X 7 availability
- Better component design to get better performance at peak time

3.5 PERFORMANCE REQUIREMENTS

Performance is measured in terms of the output provided by the application. Requirement specification plays an important part in the analysis of a system. Only when the requirement specifications are properly given, it is possible to design a system, which will fit into required environment. It rests largely in the part of the users of the existing system to give the required specifications because they are the people who finally use the system. This is because the requirements have to be known during the initial stages so that the system can be designed according to those requirements. It is very difficult to change the system once it has been designed and on the other hand designing a system, which does not cater to the requirements of the user, is of no use.

The requirement specification for any system can be broadly stated as given below:

- The system should be able to interface with the existing system
- The system should be accurate
- The system should be better than the existing system

The existing system is completely dependent on the user to perform all the duties.

CHAPTER 4

SYSTEM DESIGN

4.1 ARCHITECTURE DIAGRAM

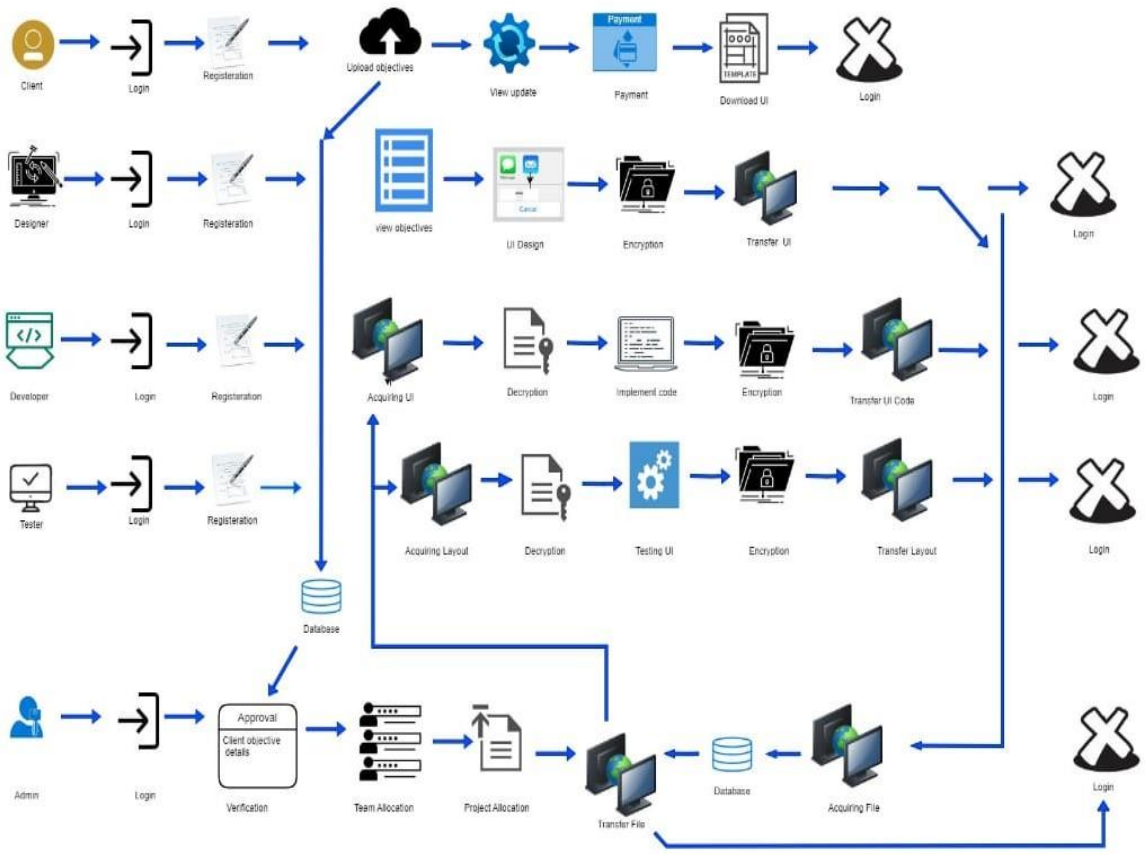
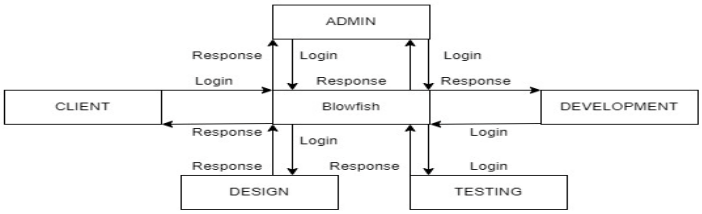


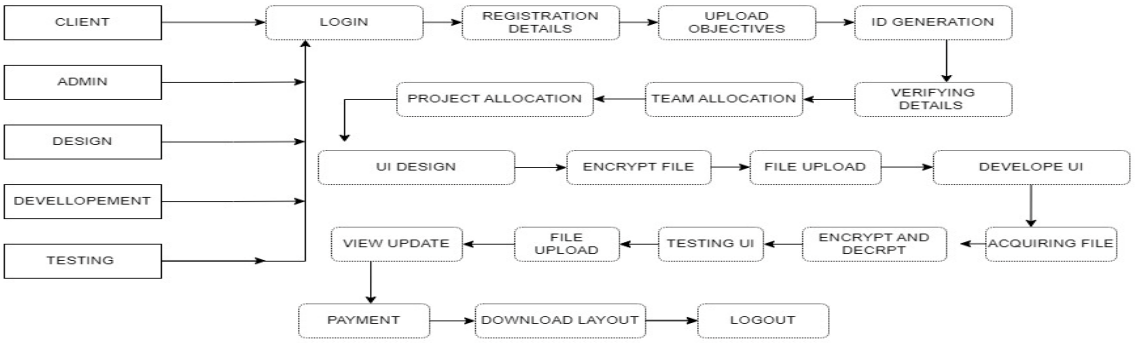
Figure 4.1 Architecture Diagram

4.2 DATA FLOW DIAGRAM

LEVEL0



LEVEL 1



LEVEL 2

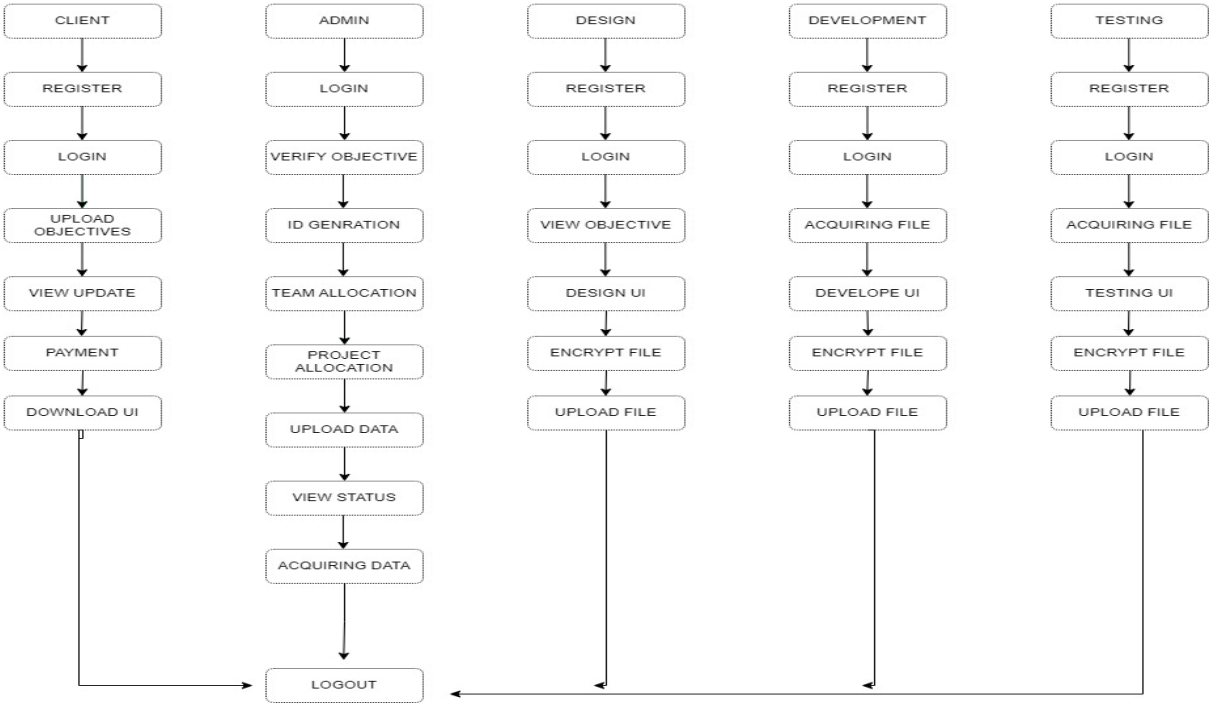


Figure 4.2 Data Flow Diagram

4.3 USE CASE DIAGRAM

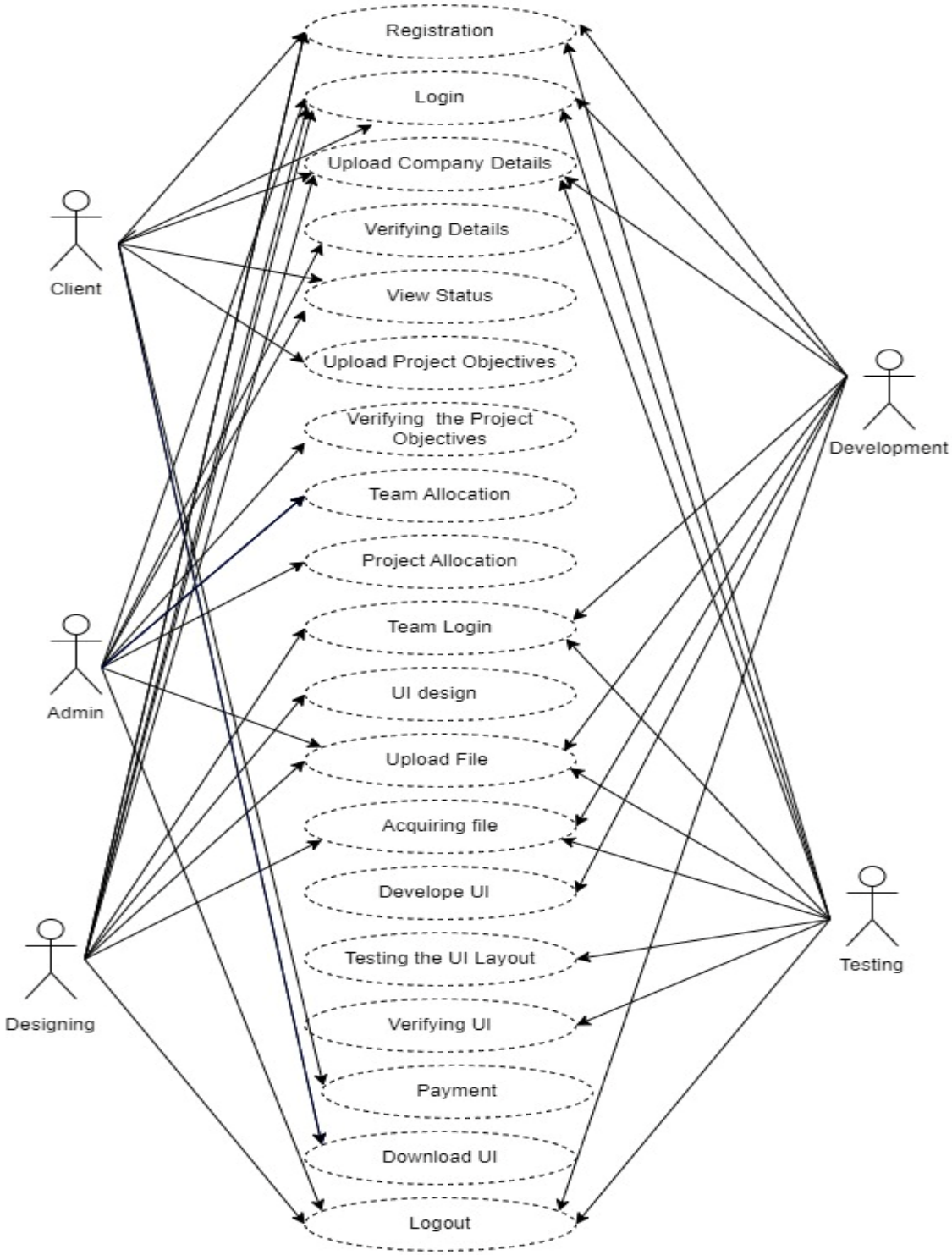


Figure 4.3 Use Case Diagram

4.4 SEQUENCE DIAGRAM

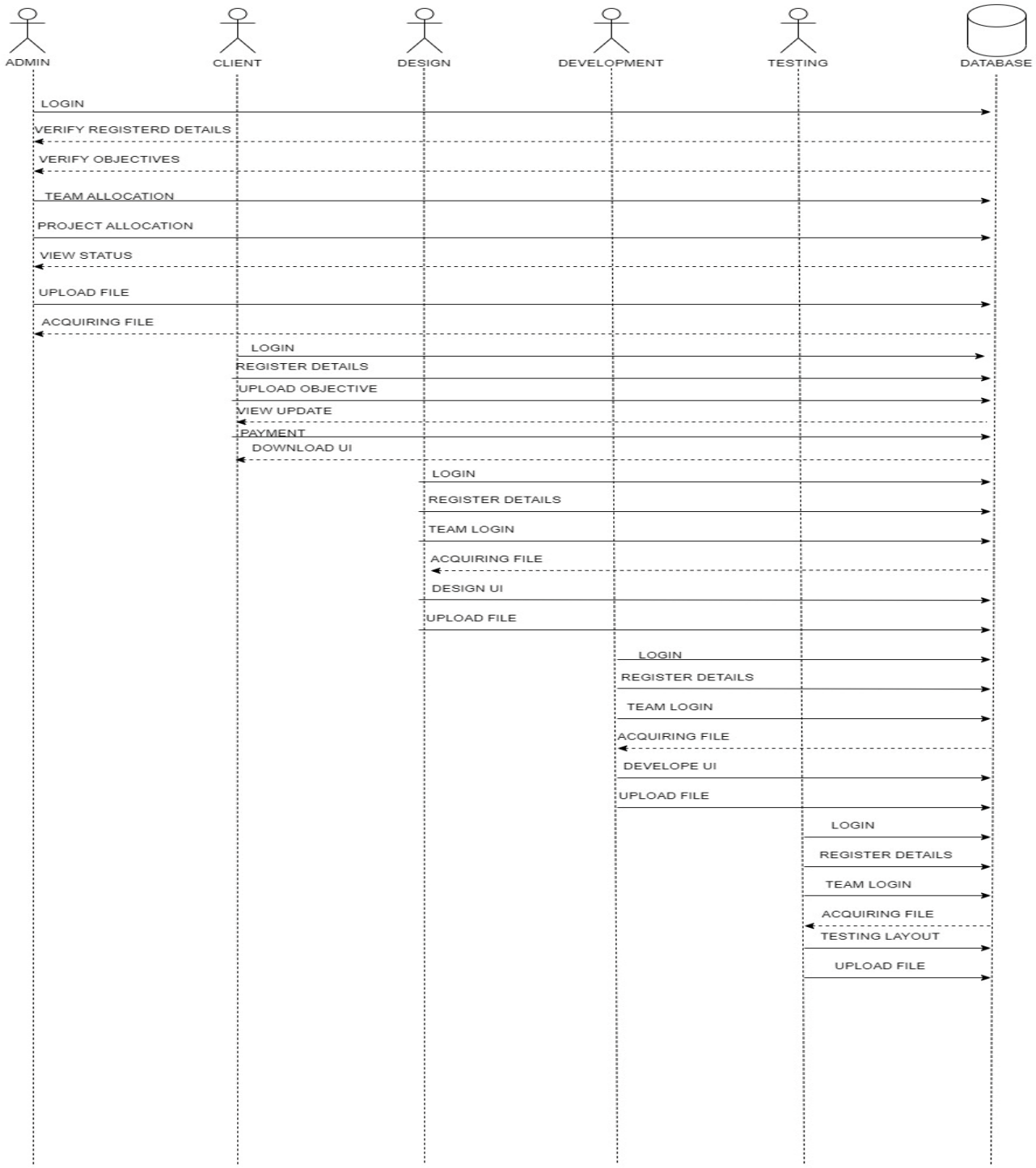


Figure 4.4 Sequence Diagram

4.5 CLASS DIAGRAM

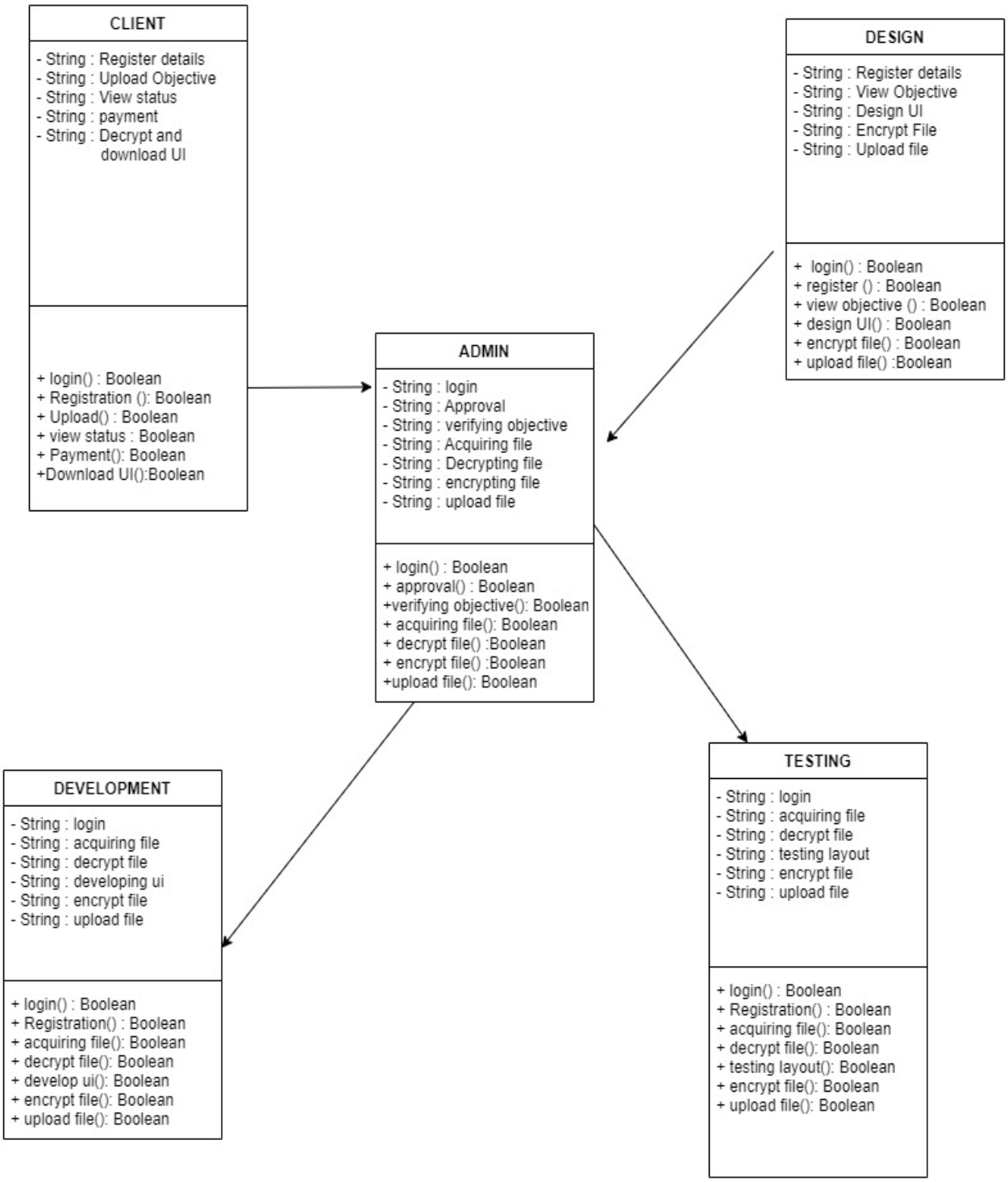


Figure 4.5 Class Diagram

4.6 ACTIVITY DIAGRAM

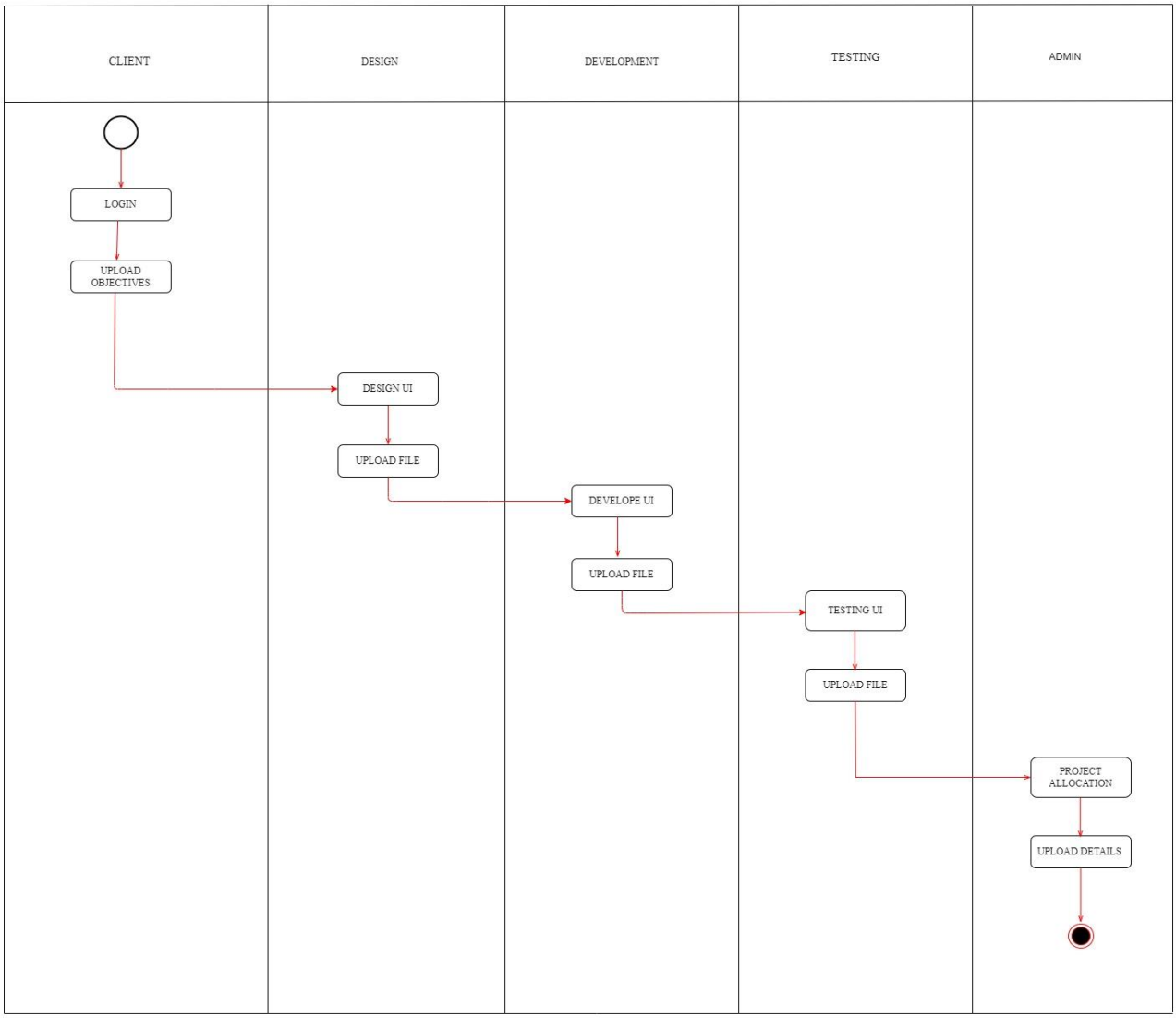


Figure 4.6 Activity Diagram

CHAPTER 5

SYSTEM MODULES

5.1 MODULES IN PROPOSED SYSTEM

5.1.1 MODULE 1: ADMIN:

The admin module, when the admin logins, the page will redirect to the admin home page. The admin page menu includes options for approving registrations, company registrations, client objectives, view layout, team allocation, progress updates, file uploads, files acquisition and review. The admin will verify the firm and registration details. Once the registration details are verified then only the admin will approve to further process otherwise not permitted to proceed. The team allocation approach will be proposed based on the metadata repository after the client objective has been verified. The admin will assign the team. Provide them the project and sample layout and charge them with generating the team's ID and password. The admin is the centre hub in this module, after the id generation the admin also entrusted with obtaining files from other modules and uploading files via the TCP/IP Protocol, after acquiring the file the decryption process will be done in the decrypt file menu. The admin can also view the status of the layout, when the admin uploading the file before that the encryption process will be held in encrypt file menu. In the progress updates the admin will verifies whether user interface is designed based on the client objectives or not and also update the process to the client, Once the payment process is done the admin will upload the user interface to the client. In the review option the admin check for the review of the layout from the client.

5.1.2 MODULE 2: CLIENT:

This module the client will register and login, once the login is done it will redirect to the client homepage. In the client homepage the menu includes options for view details, view layout, register company info, upload objective, view status, payment, download layout, feedback and logout. As soon as the client completes the registration of firm information, the admin must approve it after that the client receives an email. The client will upload the objectives if the upload objective option is selected; otherwise, the client cannot upload the objectives. The client can choose an appropriate layout via the view layout terminal, and the layout they choose will be designed. The client can access the view status terminal to observe the state of the process once the uploading of the objective has been updated. The client will receive the information in the view status terminal and updated through email once the user interface is designed. The client can download the layout from the download layout terminal and also through the email after accomplishing the payment process. Once the layout process has been downloaded, the client must accomplish the feedback approach at the feedback terminal.

5.1.3 MODULE 3: DESIGN:

This module the designer will register and login, Once the login is done it will redirect to the design homepage In the design homepage the menu includes options for view profile , register company info ,view update, team login is the sub module in that it contain view objective, design layout, upload file ,view transforming and logout. Once the login is done the designer will upload the company info in the register company info. The designer can view the firm details, registration details, and amend the details in the View profile terminal. The designer will garner the update in the view update terminal as well as via email when the admin assigns the team.

Once the team has logged in, the assigned team designer will sketch the ui based on the client's objectives and a sample layout in the design layout terminal. The designer will encrypt the file in the encrypt file terminal after finishing the user interface, and then upload the encrypted file to the admin using the TCP/IP protocol through the upload file terminal, If the user interface has been updated, it will persist in the view update terminal.

5.1.4 MODULE 4: DEVELOPMENT:

The development module developer will register and login, once the login is done it will be redirect to the development homepage, In the development homepage the menu includes options view profile, register the company info, view update, team login, the team login is the sub module in that it include acquiring file, decrypt file, develop ui, encrypt file, upload file and logout. Once the developer register the company information the info will verified by the admin after that the developer will receive the updation process through the email and also the view update terminal. Once the team login is done the developer wants to acquiring file using the TCP/IP Protocol in the acquiring file terminal. After receiving the file the developer will decrypt the file in the decrypt file terminal. Based on the design of the layout and sample layout the developer will develop the user interface after the development process is completed, the developer will encrypt the file after the process is completed in the encrypt file terminal. The developer will upload the encrypted file using TCP/IP Protocol in the upload terminal.

5.1.5 MODULE 5: TESTING:

The testing module, tester will register and login, when the login is done it will be redirect to the testing homepage. In the testing homepage the menu includes view profile, register company info, view updation, team login, the team login is the sub module in the menu includes acquiring file, decrypt file, testing layout, encrypt file, upload file and logout . Once the registration of the company info is done, the tester will receive the updation in the view updation menu, when the team login is done the tester wants to acquiring the file using the TCP/IP Protocol, In the decrypt file menu the tester will decrypt the acquiring file and test the layout whether the layout designed based on the client objectives and sample layout or not ,Once the testing process is completed the file encryption process is done in the encrypt file menu ,Once the encryption process is done the tester will upload the encrypted file using the TCP/IP Protocol to the admin and the tester can also view the layout updation in the view updation menu in this module.

CHAPTER 6

SYSTEM TESTING AND IMPLEMENTATION

6.1 APPROACH TO SOFTWARE TESTING

The software engineering process can be viewed as a spiral. Initially system engineering defines the role of software and leads to software requirement analysis where the information domain, functions, behaviour, performance, constraints, and validation criteria for software are established. Moving inward along the spiral, we come to design and finally to coding. To develop computer software, we spiral in along streamlines that decrease the level of abstraction on each turn.

A strategy for software testing may also be viewed in the context of the spiral. Unit testing begins at the vertex of the spiral and concentrates on each unit of the software as implemented in source code. Testing progress is done by moving outward along the spiral to integration testing, where the focus is on the design and the construction of the software architecture. Talking another turn on outward on the spiral we encounter validation testing where requirements established as part of software requirements analysis are validated against the software that has been constructed. Finally, we arrive at system testing, where the software and other system elements are tested.

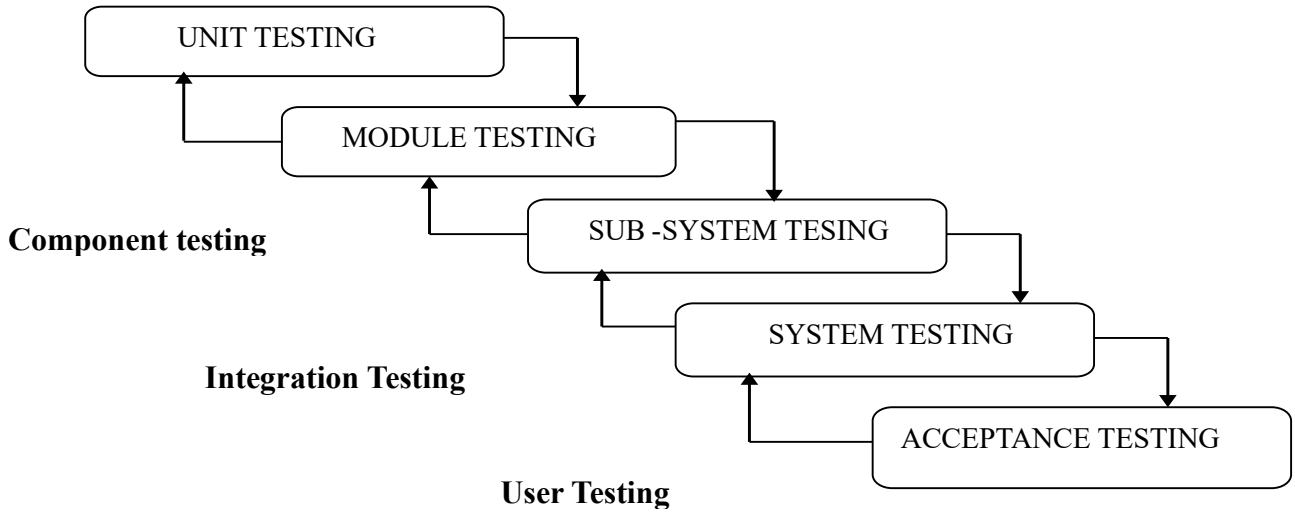


Figure 6.1 Levels of Testing

6.2 UNIT TESTING

Unit testing focuses verification effort on the smallest unit of software design, the module. The unit testing we have is white box oriented and some modules the steps are conducted in parallel.

1. WHITE BOX TESTING

This type of testing ensures that

- All independent paths have been exercised at least once
- All logical decisions have been exercised on their true and false sides
- All loops are executed at their boundaries and within their operational bounds
- All internal data structures have been exercised to assure their validity.

To follow the concept of white box testing we have tested each form .We have created independently to verify that Data flow is correct, All conditions are exercised to check their validity, All loops are executed on their boundaries.

2. BASIC PATH TESTING

The established technique of flow graph with Cyclamate complexity was used to derive test cases for all the functions. The main steps in deriving test cases were:

Use the design of the code and draw correspondent flow graphs.

Determine the Cyclamate complexity of the resultant flow graph, using formula:

V (G) =E-N+2 or

V (G) =P+1 or

V (G) =Number of Regions

Where V (G) is Cyclomatic complexity,

E is the number of edges,

N is the number of flow graph nodes,

P is the number of predicate nodes.

Determine the basis of set of linearly independent paths.

3. CONDITIONAL TESTING

This part of the testing each of the conditions were tested to both true and false aspects. And all the resulting paths were tested. So that each path that may be generated on particular condition is traced to uncover any possible errors.

4. DATA FLOW TESTING

This type of testing selects the path of the program, according to the location of the definition and use of variables. This kind of testing was used only when some local variable were declared. The definition-use chain method was used in this type of testing. These were particularly useful in nested statements.

5. LOOP TESTING

No	Test Scenario	Expected Result	Test Result
1	Username is correct. Password is incorrect.	Username and Password is incorrect.	Username and Password is incorrect.
2	Username is incorrect. Password is correct.	Username and Password is incorrect.	Username and Password is incorrect.
3	Username is empty. Password is correct.	Username is required.	Username is required.
4	Username is correct. Password is empty.	Password is required.	Password is required
5	Both Username and Password is incorrect.	Username and Password is incorrect.	Username and Password is incorrect.
6	Both Username and Password is empty.	Username and Password is required.	Username and Password is required.
7	Both Username and Password is correct.	Login Successful.	Login Successful.

- All the loops were tested at their limits, just above them and just below them.
- All the loops were skipped at least once.

CHAPTER 7

SYSTEM SECURITY

7.1 SECURITY IN SOFTWARE

System security refers to various validations on data in the form of checks and controls to avoid the system from failing. It is always important to ensure that only valid data is entered, and only valid operations are performed on the system.

The system employs two types of checks and controls:

7.1.1 CLIENT-SIDE VALIDATION

Various client-side validations are used to ensure on the client side that only valid data is entered. Client-side validation saves server time and load to handle invalid data. Some checks are imposed:

- JavaScript in used to ensure those required fields are filled with suitable data only. Maximum lengths of the fields of the forms are appropriately defined.
- Forms cannot be submitted without filling up the mandatory data so that manual mistakes of submitting empty fields that are mandatory can be sorted out at the client side to save the server time and load.
- Tab-indexes are set according to the need and taking into account the ease of use while working with the system.

7.1.2 SERVER-SIDE VALIDATION

Some checks cannot be applied on the client side. Server-side checks are necessary to save the system from failing and intimating the user that some invalid operation has been performed or the performed operation is restricted. Some of the server-side checks imposed is:

- A server-side constraint has been imposed to check for the validity of primary key and foreign key. A primary key value cannot be duplicated. Any attempt to duplicate the primary value results in a message intimating the user about those values through the forms using foreign key can be updated only of the existing foreign key values.
- The user is intimated through appropriate messages about the successful operations or exceptions occurring at server side.
- Various Access Control Mechanisms have been built so that one user may not agitate upon another. Access permissions to various types of users are controlled according to the organizational structure. Only permitted users can log on to the system and can have access according to their category. User- name, passwords and permissions are controlled the server side.
- Using server-side validation, constraints on several restricted operations are imposed.

RESULT AND DISCUSSION

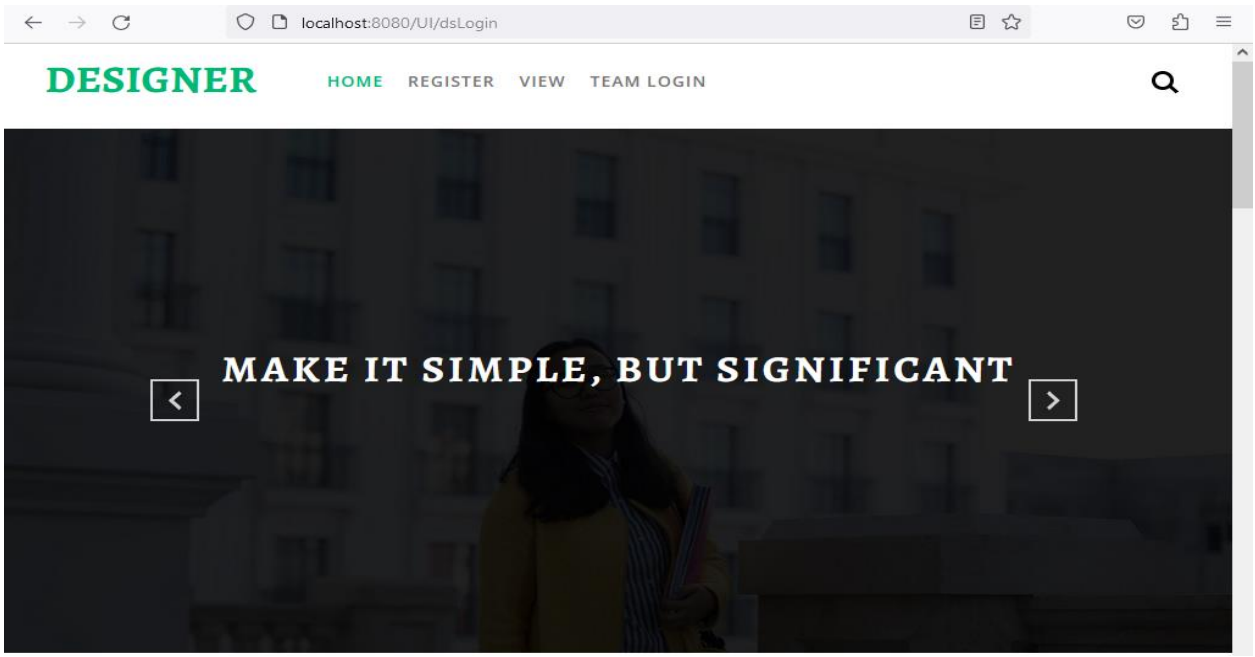


Figure A1: Designer Team Homepage

This is the designer team homepage where the designer team members can login, register and view the file that needs a new design to be added.

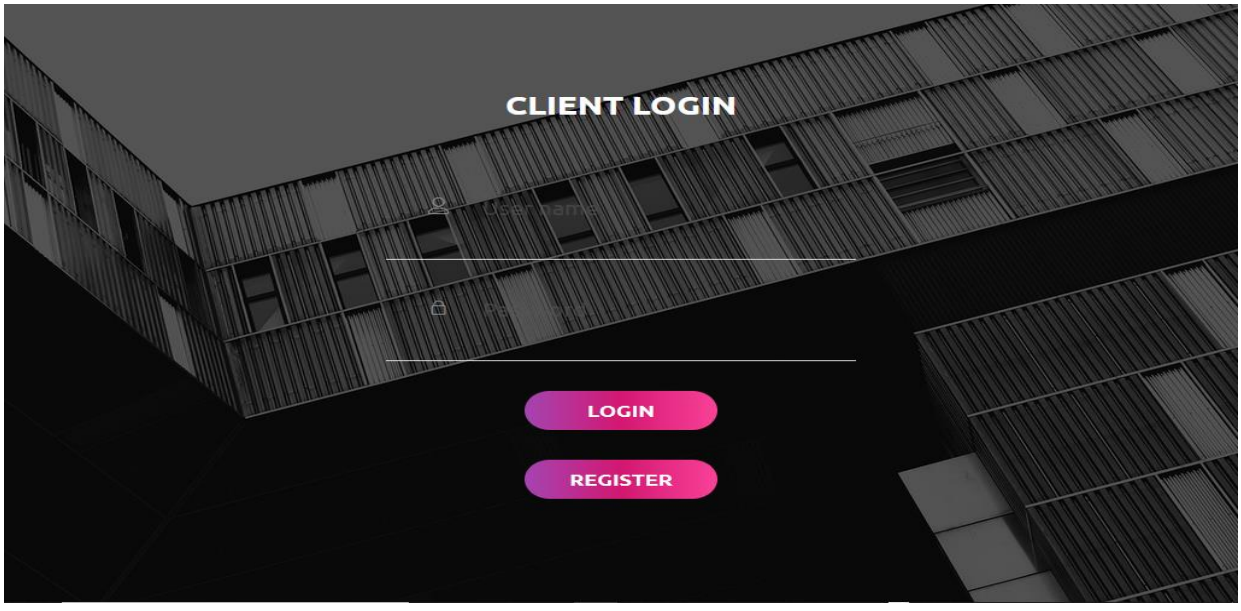


Figure A2: Client Login

This is the client login page where you will be redirected from the homepage and from here after registering or logging in the client will be able to request their UI.

Figure A3: File Transfer page

This is the file transfer page from which the file created will be transferred from one module to another only after receiving the request from another module.

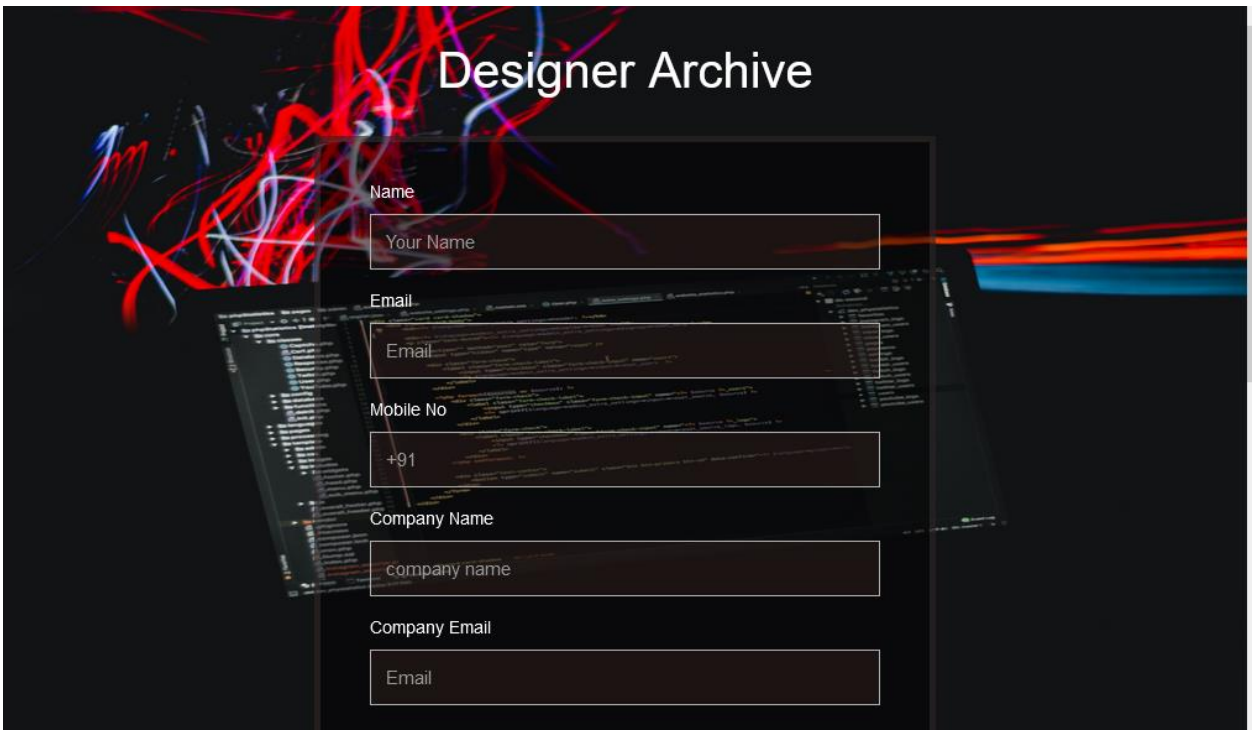


Figure A4: Designer Register page

This is the designer register page where a new designer can register themselves.

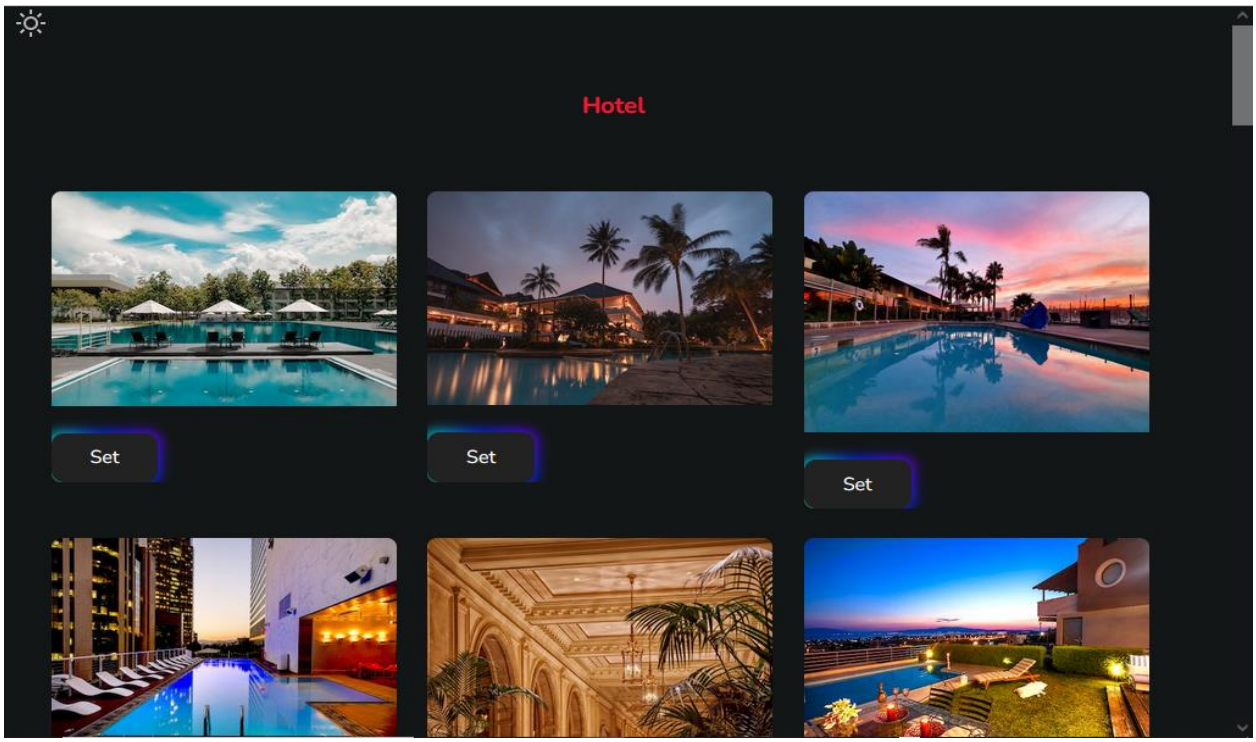


Figure A5: Interface Selecting Page

This is the page where the clients can select their interface based on their requirements and needs .

Figure A6: File Receiving Page

This is the page where the files sent from another module can be received after sending the request.

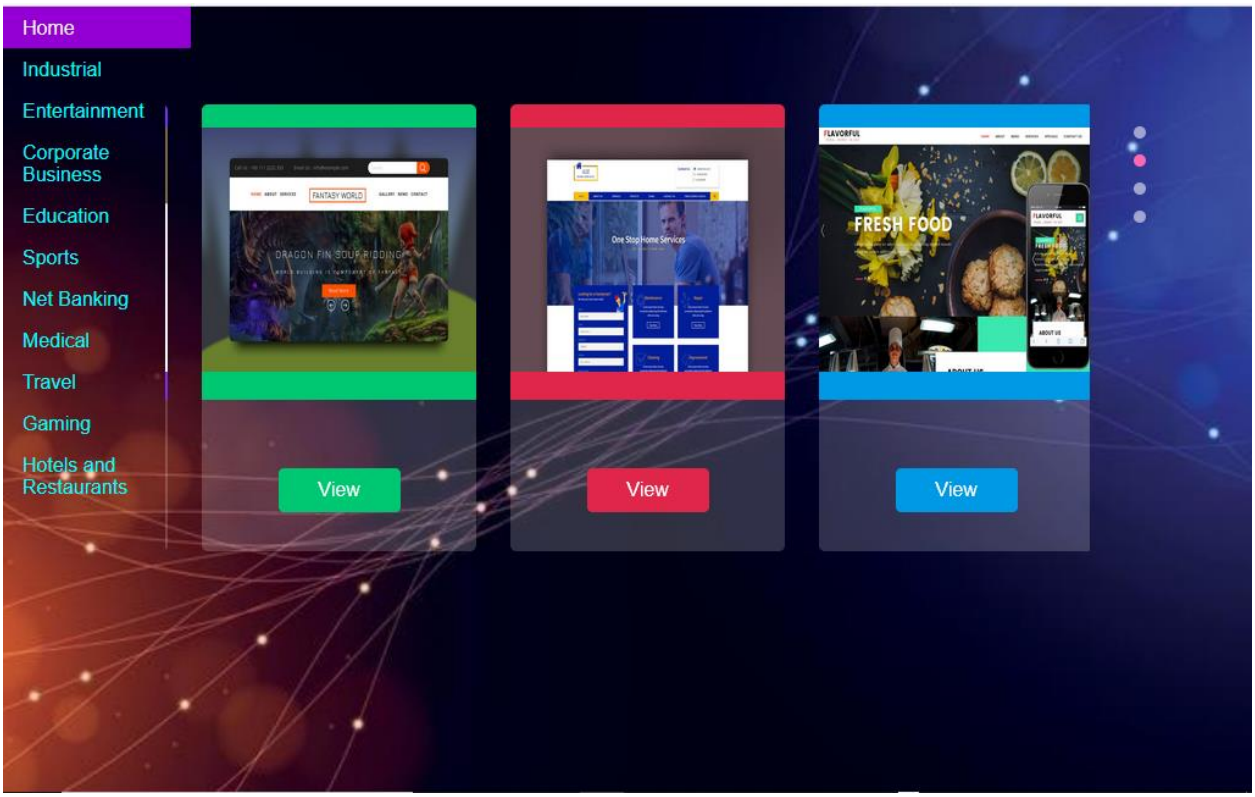


Figure A7: UI Selecting Interface

This is the page where the client chooses their user interface based on their preferences.

Figure A8: File Upload page

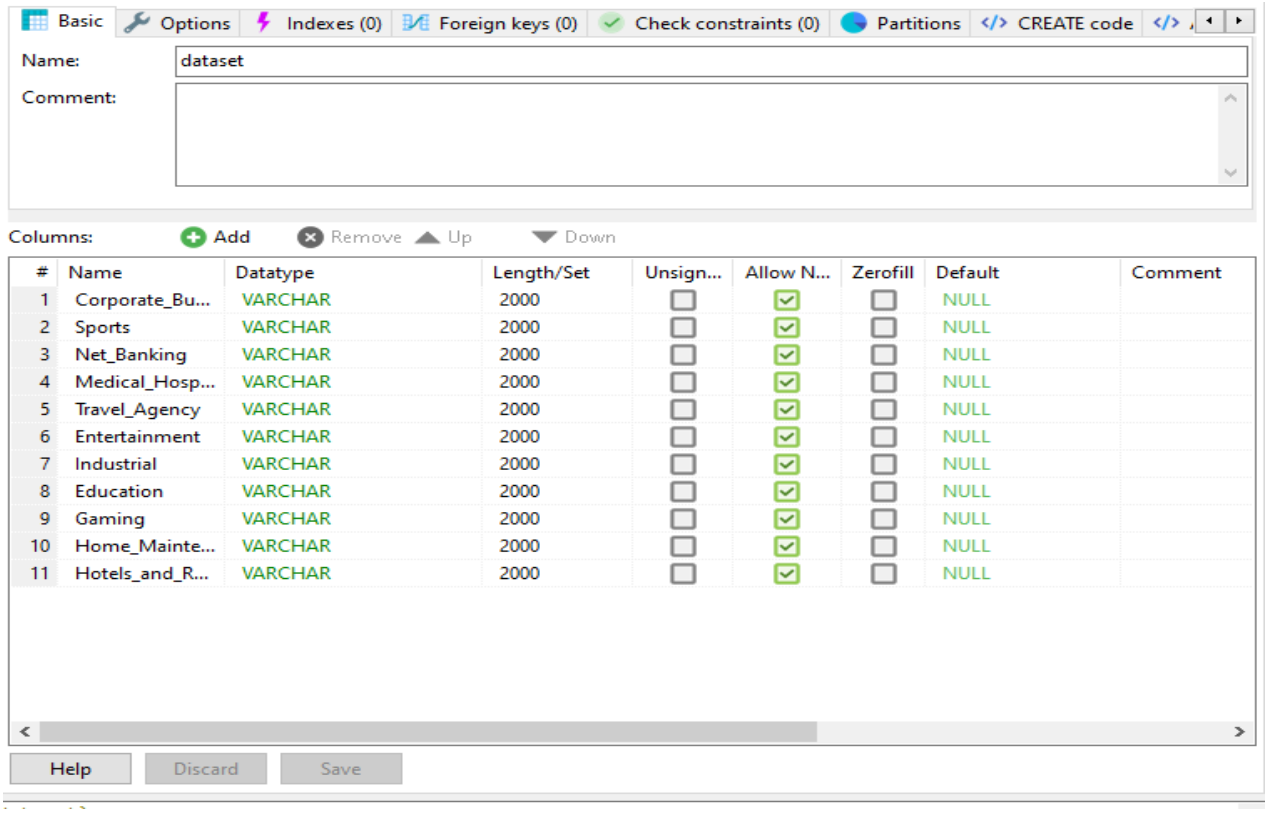


Figure A9: Interfaces Database

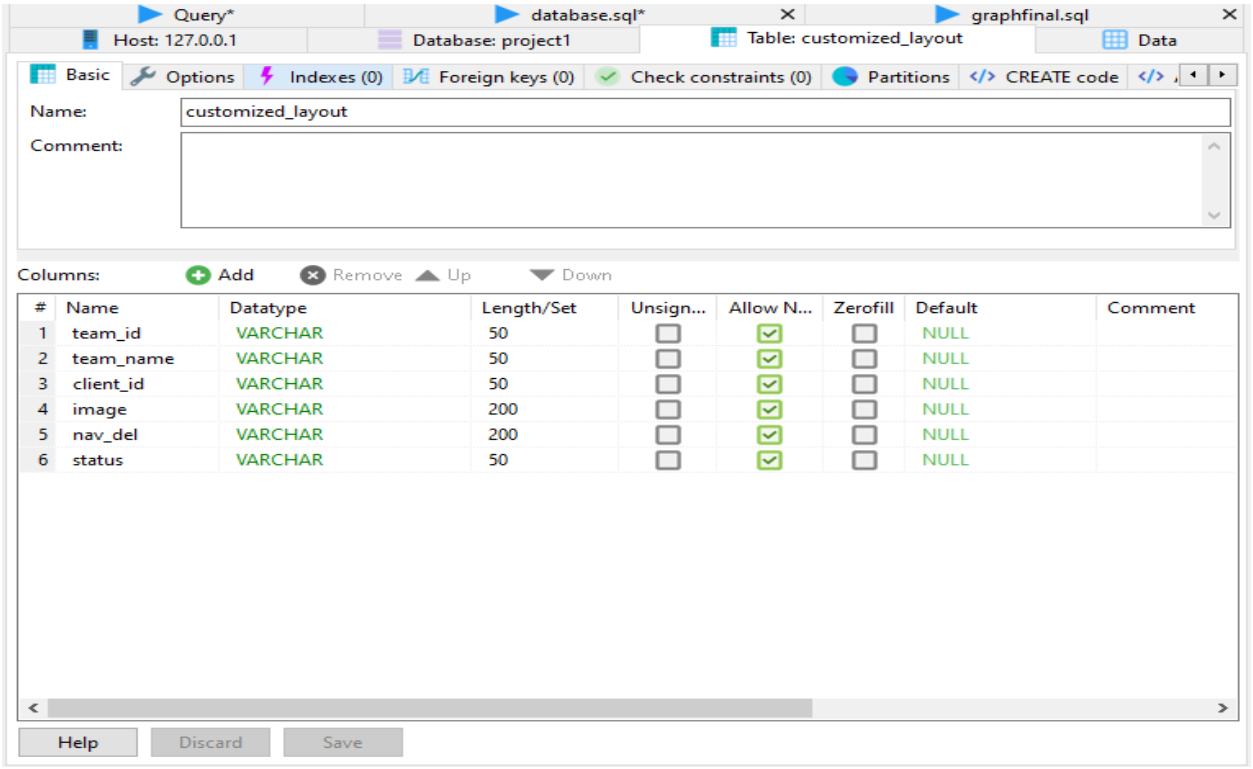


Figure A10: Team Details Database

CONCLUSION

This proposed model has accomplished the Blowfish algorithm and logistic regression algorithm. Based on the client requirements the proposed model will generate the user interface. This proposed model we must implement the data transfer using TCP/IP process within same LAN and also encrypting the data while data transfer.

FUTURE ENHANCEMENT

In future we enhance to transmit the processed data from server to client within one MAN (**Metropolitan Area Network)** to another LAN (**Local Area Network)** via UDP Protocol. UDP protocol is used for data transport. Fabricating the wireless connectivity that the UDP offers, we may expedite the data transfer. Thus, this proposed model makes a great impact and satisfies the required need in the IT industry. In the future, it has been enhanced and applied with experimentation for an effective needed situation.

APPENDIX 1

SOURCE CODE

clientLogin.java

```
package mypack;
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
@WebServlet("/clientLogin")
public class clientLogin extends HttpServlet {
private static final long serialVersionUID =1L;
/**
 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
response)
 */
protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
response.setContentType("text/html");
PrintWriter pw= response.getWriter();
```

```
String email=request.getParameter("email");
String password= request.getParameter("ps");
System.out.println(email);
System.out.println(password);
try
{
Class.forName("com.mysql.jdbc.Driver");// used to register the driver class
Connectioncon=DriverManager.getConnection("jdbc:mysql://localhost:3306/proje
ct1","root","root");
PreparedStatement ps  =con.prepareStatement ("select * from clientreg where
email='"+email+"' and password='"+password+"' ");
        ResultSet rs = ps.executeQuery();
        if(rs.next()) {
pw.print("<html><body><script> alert('Login
successfully..:)')</script></body></html>");
RequestDispatcher rd =request.getRequestDispatcher("C_client.html");
        rd.include(request, response);}
        else
        {
pw.print("<html><body><script> alert('Login Failed.. :(
')</script></body></html>");
RequestDispatcher rd = request.getRequestDispatcher("index.html");
                rd.include(request, response);}
}
catch(Exception ex)
{
        ex.printStackTrace();}}}
```

Devtermlogin.java

```
package mypack;
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
@WebServlet("/DEVtermlogin")
public class DEVtermlogin extends HttpServlet {
private static final long serialVersionUID =1L;
/**
 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
response)
 */
protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
String frontid=request.getParameter("Un");
String frontpassword=request.getParameter("Ps");
System.out.println("frontid"+frontid);
  System.out.println("frontpassword"+frontpasword);
PrintWriter pw= response.getWriter();
String password=null;
```

```
String userid=null;
HttpSession session=request.getSession();
session.setAttribute("userid",frontid);
session.setAttribute("frontpassword22", frontpassword);
String dep="developer";
try
{
Class.forName("com.mysql.jdbc.Driver");
        Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/project1","root","root")
;
PreparedStatement ps = con.prepareStatement("select user_id,user_password from
team where department ='"+dep+"'");
        ResultSet rs = ps.executeQuery();
        while(rs.next())
        {
                userid=rs.getString("user_id");
     password=rs.getString("user_password");
        }
        String[] userid1=userid.split(",");
     System.out.println("password"+password);
        System.out.println("for starts----------");
        for(String id:userid1)
        {
                System.out.println("userid"+id);
        }
        int j =0;
```

```
        for(int i=0;i<userid1.length;i++)
{
  if(frontid.equalsIgnoreCase(userid1[i])&&frontpassword.equals(password)) {
                j++;
                break;}
        else if(frontid!=userid1[i]&&frontpassword!=pasword)
        { j=0; }
        }
        if(j>0)
        {
        pw.print("<html><body><script> alert('Login successfully.. :)')
</script></body></html>");
RequestDispatcher rd = request.getRequestDispatcher("DEV_interface.html");
        rd.include(request, response);}
        else if(j<=0)
        {
        pw.print("<html><body><script> alert ('Login Failed.. :( ')
</script></body></html>");
RequestDispatcher rd = request.getRequestDispatcher("DEV_dev.html");
        rd.include(request, response);}
}
catch(Exception ex)
{
        ex.printStackTrace();
}
}
}
```

dstermlogin.java

```
package mypack;
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import javax.servlet.RequestDispatcher;
@WebServlet("/dsLogin")
public class dsLogin extends HttpServlet {
private static final long serialVersionUID = 1L;
/**
 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
response)
 */
protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
// TODO Auto-generated method stub
String frontid=request.getParameter("Username");
String frontpassword=request.getParameter("Password");
PrintWriter pw= response.getWriter();
String password=null;
String userid=null;
HttpSession session=request.getSession();
session.setAttribute("userid",frontid);
```

```
session.setAttribute("frontpassword22", frontpassword);
String dep="Design";
try
{
	Class.forName("com.mysql.jdbc.Driver");// used to register the driver class
	Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/project1","root","root")
;
PreparedStatement ps = con.prepareStatement("select user_id,user_password from
team where department = '"+dep+"'");
	ResultSet rs = ps.executeQuery();
	while(rs.next())
	{
		userid=rs.getString("user_id");
		password=rs.getString("user_password");
	}
	String[] userid1=userid.split(",");
	System.out.println("password--------"+password);
	for(String id:userid1)
	{
		System.out.println("userid"+id);}
	int j =0;
	for(int i=0;i<userid1.length;i++)
	{
	if(frontid.equalsIgnoreCase(userid1[i])&&frontpassword.equals(password)) {
	j++;
	break;}
```

```
else if(frontid!=userid1[i]&&frontpassword!=password)
{
  j=0;}
}
if(j>0)
{
pw.print("<html><body><script> alert('Login successfully..
:)')</script></body></html>");
RequestDispatcher rd =request.getRequestDispatcher("DS_interface.html");
rd.include(request, response);
}
else if(j<=0)
{
pw.print("<html><body><script> alert('Login Failed.. :(
')</script></body></html>");
RequestDispatcher rd = request.getRequestDispatcher("DS_design.jsp");
rd.include(request, response);
}
}
catch(Exception ex)
{
ex.printStackTrace();
}
}
}
```

REFERENCES

[1] B. Pang and L. Lee, "Opinion mining and sentiment analysis," Found. Trends Inf. Retr., vol. 2, nos. 1–2, pp. 1–135, 2008.

[2] M. Hu and B. Liu, "Mining and summarizing customer reviews," in Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (SIGKDD), 2004, pp. 168–177.

[3] G. Qiu, B. Liu, J. Bu, and C. Chen, "Opinion word expansion and target extraction through double propagation," Comput. Linguistics, vol. 37, no. 1, pp. 9–27, Mar. 2016.

[4] Y. Lu, C. Zhai, and N. Sundaresan, "Rated aspect summarization of short comments," in Proc. 18th Int. Conf. World Wide Web (WWW), 2009, pp. 131–140.

[5] H. Lakkaraju, R. Socher, and C. D. Manning, "Aspect specific sentiment analysis using hierarchical deep learning," in Proc. NIPS Workshop Deep Learn. Represent. Learn., 2014, pp. 1–9.

[6] L. Jiang, M. Yu, M. Zhou, X. Liu, and T. Zhao, "Target-dependent Twitter sentiment classification," in Proc. Annu. Meeting Assoc. Comput. Linguistics (ACL HLT), 2011, pp. 151–160

[7] L. Dong, F. Wei, C. Tan, D. Tang, M. Zhou, and K. Xu, "Adaptive recursive neural network for target-dependent Twitter sentiment classification," in Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics (ACL), 2014, pp. 49–54

[8] D. Ma, S. Li, X. Zhang, and H. Wang, "Interactive attention networks for aspect-level sentiment classification," in Proc. 26th Int. Joint Conf. Artif. Intell. (IJCAI), Aug. 2017, pp. 4068–4074.

[9] I. Titov and R. McDonald, "Modeling online reviews with multi-grain topic models," in Proc. 17th Int. Conf. World Wide Web (WWW), 2008, pp. 111–120.

[10] N. Jakob and I. Gurevych, "Extracting opinion targets in a single and cross-domain setting with conditional random fields," in Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP), 2010, pp. 1035–1045.