A

Major Project Report On

"BLOCKCHAIN BASED ACCOUNTS PAYABLE PLATFORM FOR GOODS TRADE"

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

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE & ENGINEERING

By

BOLISHETTI SRINIVAS 21N61A0548

Under the Guidance of

Dr. M.V.Hanumantha Reddy
Principal, CSE Dept, VITS

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
VIVEKANANDA INSTITUTE OF TECHNOLOGY & SCIENCE
(Approved by AICTE New Delhi & Affiliated to JNTU, Hyderabad) **An ISO 9001:2015 Certified Institution**
KARIMNAGAR-505001

2024-2025

VIVEKANANDA INSTITUTE OF TECHNOLOGY & SCIENCE(N6)

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

An ISO 9001:2015 Certified Institution

KARIMNAGAR-505001

CERTIFICATE

This is to certify that the major-project report "**BLOCKCHAIN BASED ACCOUNTS PAYABLE PLATFORM FOR GOODS TRADE**" is being submitted by 21N61A0548 BOLISHETTI SRINIVAS, in B.Tech IV-II semester, Computer Science & Engineering is a record bonafide work carried out by them. The results embodied in this report have not been submitted to any other University for the award of any degree.

Internal Guide

Dr. Mula Veera Hanumantha Reddy

Principal, Dept of CSE, Vits

Head of the Department & Principal

Dr. Mula Veera Hanumantha Reddy

Dept of CSE

External Examiner

Principal

Dr. Mula Veera Hanumantha Reddy

VIVEKANANDA INSTITUTE OF TECHNOLOGY & SCIENCE

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

Karimnagar-505001

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

I **BOLISHETTI SRINIVAS** 21N61A0548, here by declare that the Project **r**eport entitled **"BLOCKCHAIN BASED ACCOUNTS PAYABLE PLATFORM FOR GOODS TRADE"** submitted in partial fulfillment of the requirements for the award of degree in B. Tech IV-II semester, Computer Science & Engineering This is a record bonafide work carried out by me. The results embodied in this report have not been submitted to any other University for the award of any degree or diploma.

BOLISHETTI SRINIVAS

HTNO: -21N61A0548

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VIVEKANANDA INSTITUTE OF TECHNOLOGY & SCIENCE, Karimnagar.

ACKNOWLEDGEMENT

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

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

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

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

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

BOLISHETTI SRINIVAS

21N61A0548

ABSTRACT

Goods trade is a supply chain transaction that involves shippers buying goods from suppliers and carriers providing goods transportation. Various business documents like purchase order, despatch advice, invoices, and receive advice get exchanged among the trade participants during any trade transaction. Similarly, various business processes like freight transport, invoice generation, goods receiving, invoice processing, and payment processing get executed by the participants in a trade transaction.

Discrepancy during the execution of any of these processes leads to disputes between the participants involved, and the time consumed in resolving the disputes causes a delay in the process execution resulting in cost overhead for all the participants involved. Shippers are issued invoices from suppliers for the goods provided and from carriers for goods transportation. The shipper carries out goods receiving and invoice processing before proceeding to payment processing of bills for suppliers and carriers, where invoice processing includes tasks like processing claims and adjusting the payments.

Goods receiving involves verification of received goods by Shipper's receiving team. Processing claims and adjusting the payments are carried out by Shipper's accounts payable team, which in turn is verified by the accounts receivable teams of suppliers and carriers. This paper presents a blockchain-based accounts payable system for shippers, which generates claims for deficiency in the goods received and accordingly adjusts the payment in the bills for suppliers and carriers.

Primary motivations for these supply chain organizations to adopt blockchain-based accounts payable systems are to eliminate the process redundancies (accounts payable vs. accounts receivable), to reduce the number of disputes among the transacting participants, to reduce the dispute resolution time, and to accelerate the accounts payable processes via optimizations in the claims generation and blockchain-based dispute reconciliation.

INDEX

1.INTRODUCTION

Any trade transaction, be it domestic or global, involves exercising certain processes to complete. Domestic trade is the exchange of goods within country boundaries in contrast to between different countries in global/international trade. We describe the different processes involved in goods trade using a global trade transaction in figure Fig. 2. Shippers initiate a trade transaction by sending a purchase order (PO which consists of details of the requested goods to the suppliers. Suppliers typically package the goods into intermodal containers either by themselves or with the help of Origin Cargo Management (OCM) team. Suppliers issue dispatch advice (DA) that describes the goods packed details, and commercial invoice (CI) that describes the terms together with the details of the amount that shipper must pay for the goods supplied. Since global trade involves freight transportation across country borders, a typical freight journey involves multiple modes (e.g., road, rail, or sea) of carriers contributing to the container movement from origin to the destination. Moreover, freight transportation may also involve drayage providers to move containers a short distance via ground fright (e.g., move containers from truck to a ship).

Once reight reaches the delivery center at destination, the goods receiving team of the shipper verifies if the receive goods can be accepted or not. If there are any damages to the received goods or discrepancies in terms of received quantity/price against PO, then the receiving team records the same via receiving advice (RA). The different carriers involved in freight movement also issue their respective invoices for their services.

Payment processing involves executing a payment method as per the terms captured in the service contracts between trade participants. The most common payment method is open account, where the goods is shipped and delivered before the release of funds from the shipper within an agreed time frame. There is another payment method which involves financing facilitated by banks and financial institutions (e.g., letter of credit [1] for supplier invoices in global trade, factoring [2] for carrier invoices in global trade, and reverse factoring [3] for supplier invoices in domestic trade). Goods trade can be summarized as the shipper acquiring the goods by paying the bills to the supplier and carriers as captured in figure Fig. 2

The complexity of goods trade increases with the multiple en route handoffs between different parties involved in the goods over ment process. End-to-end shipping visibility becomes significantly challenging to put together. Hence, shippers and other parties involved, try to gather as many details as possible during the execution of various goods trade processes (figure Fig. 1). Eventually, each organization ends up accruing data with their priorities and results in mistrust of information in the multi-party planning and invoice processing. To address the trust and transparency issues among the competitive an mutually distrusting participants of the supply chain network, block chain-based the goods trade industry is adopting innovative solutions of late. Trade Lens [5] is a block chain based solution to provide visibility into the current status of freight transport with the help of real-time shipment tracking events originated from different supply chain participants. Generating the invoices for freight carriers (e.g., OCM, Land Carriers, Ocean Carriers, and Drayage) involved in the goods movement from origin to the destination using the real-time container tracking events and the shipment details captured by Trade Lens as depicted in figure Fig. 2 got carried out in [6].

Here are the key features of our block chain-based accounts payable platform. CA is generated by a block chain smart contract using the EDI documents PO, DA, RA, and supplier invoice. PAs for supplier and carriers are generated by a block chain smart contract using supplier invoice, carrier invoices, and CA. And our system allows the shipper, supplier, and carriers to raise disputes on the generated claim advice and payment advices and reconcile before payment processing with audit trails. Moreover, our system lets the reconciled and approved CA and PAs be sent to the customer's (shipper, supplier, or carrier) existing ERP systems via the API interfaces. Thus, our system is seamlessly integrated with the customer's downstream applications. Our system allows email/SMS alert notifications to be generated as per user notification settings based on different triggers (e.g., CA is issued or dispute raised).

In the following sections, first, we discuss related work in this domain and its shortcomings (Section II). Next, we present the architectural design of our system, describe various system components (Section III) and the implementation notes (Section IV). Finally, we describe the experimental setup used for evaluation of the proposed system along with the performance of different types of transactions (Section V) and summarize our contributions (Section VI).

1.2 EXISTING SYSTEM

- Organizations are innovating products to help digitize various processes (Figure 1) involved in the goods trade industry. TradeLens [5] is a solution to provide visibility into the current status of freight transport underpinned by blockchain technology. Producing blockchain-based e-invoices for the freight carriers as part of the invoice generation process got addressed in [6], [7].
- Several blockchain-based solutions exist related to financing during payment processing [8]–[11]. There are instances where the use of blockchain technology for accounts payable (receivable) got discussed [12]–[14]. Similarly, the possibility of carrying out matching (e.g., 3-way matching) of EDI documents on blockchain during invoice processing is also discussed [15].
- The blockchain-based invoice processing system in our paper breaks down the claims generation into claims under different categories where claims under certain categories can be generated and issued before the goods delivery to the shipper. The advantages of this approach are two-fold: invoice processing gets accelerated since the dispute process can take place before the goods delivery to the shipper, and dispute reconciliation becomes easier since disputes are handled at the granularity of claims under a category resulting in faster dispute resolution. Hence, it is crucial to explore systems like the one proposed in this paper that enhance the effectiveness of blockchain-based accounts payable systems in invoice processing. Such a solution also helps organizations to build end to- end blockchain-based platforms providing visibility across the various processes of goods trade.

.

1.3 DISADVANTAGES OF EXISTING SYSTEM

- Payment processing not involves executing a payment method as per the terms captured in the service contracts between trade participants.
- In the existing system,the complexity of goods trade increases with the multiple en route handoffs between different parties involved in thegoods movement process.

1.3 PROPOSED SYSTEM

- In this paper, the system proposes a blockchain-based accounts payable system extending the TradeLens platform with capabilities to fulfill the needs of the supply chain network participants related to invoice processing (generation of CA and PAs) and dispute handling.

- Here are the key features of our blockchain-based accounts payable platform. CA is generated by a blockchain smart contract using the EDI documents PO, DA, RA, and supplier invoice. PAs for supplier and carriers are generated by a blockchain smart contract using supplier invoice, carrier invoices, and CA. And our system allows the shipper, supplier, and carriers to raise disputes on the generated claim advice and payment advices and reconcile before payment processing with audit trails. Moreover, our system lets the reconciled and approved CA and PAs be sent to the customer's (shipper, supplier, or carrier) existing ERP systems via the API interfaces. Thus, our system is seamlessly integrated with the customer's downstream applications. Our system allows email/SMS alert notifications to be generated as per user notification settings based on different triggers (e.g., CA is issued or dispute raised).

1.4 ADVANTAGES OF PROPOSED SYSTEM

- The system is fast and an effective due to presence of blockchain between supplier and shipper and shipper to end user.
- In the proposed system, matching EDI documents ensures that the shipper doesn't pay for goods that are not received or overpay for goods received. The shipper is entitled to pay to the supplier for the quantity of the accepted goods at the agreed price as per the purchase order

.

2.PRELIMINARY INVESTIGATION

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

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

2.1 REQUEST CLARIFICATION

After the approval of the request to the organization and project guide, with an investigation being considered, the project request must be examined to determine precisely what the system requires.Here our project is basically meant for users within the company whose systems can be interconnected by the Local Area Network(LAN). In today's busy schedule man need everything should be provided in a readymade manner. So taking into consideration of the vastly use of the net in day to day life, the corresponding development of the portal came into existence.

FEASIBILITY ANALYSIS

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

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

Operational Feasibility

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

Economic Feasibility

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

Technical Feasibility

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

2.2 REQUEST APPROVAL

Not all request projects are desirable or feasible. Some organization receives so many project requests from client users that only few of them are pursued. However, those projects that are both feasible and desirable should be put into schedule. After a project request is approved, it cost, priority,

completion time and personnel requirement is estimated and used to determine where to add it to any project list. Truly speaking, the approval of those above factors, development works can be launched.

2.3 SYSTEM DESIGN AND DEVELOPMENT

INPUT DESIGN

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

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

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

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

OUTPUT DESIGN :

The Output from the computer is required to mainly create an efficient method of communication within the company primarily among the project leader and his team members, in other words, the administrator and the clients. The output of VPN is the system which allows the project leader to manage his clients in terms of creating new clients and assigning new projects to them, maintaining a record of the project validity and providing folder level access to each client on the user side depending on the projects allotted to him. The server has to be started and then the internet explorer in used as the browser. The project will run on the local area network so the server machine will serve as the administrator while the other connected systems can act as the clients. The developed system is highly user friendly and can be easily understood by anyone using it even for the first time.

3. SYSTEM REQUIREMENTS SPECIFICATION

3.1 H/W SYSTEM CONFIGURATION:-

- Processor - Pentium –IV
- RAM - 4 GB (min)
- Hard Disk - 20 GB
- Key Board - Standard Windows Keyboard
- Mouse - Two or Three Button Mouse
- Monitor - SVGA

3.2 SOFTWARE REQUIREMENTS:

- Operating System - Windows XP
- Coding Language - Java/J2EE(JSP,Servlet)
- Front End - J2EE
- Back End - MySQL

3.3 FEASIBILITY ANALYSIS

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

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

Operational Feasibility

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

Economic Feasibility

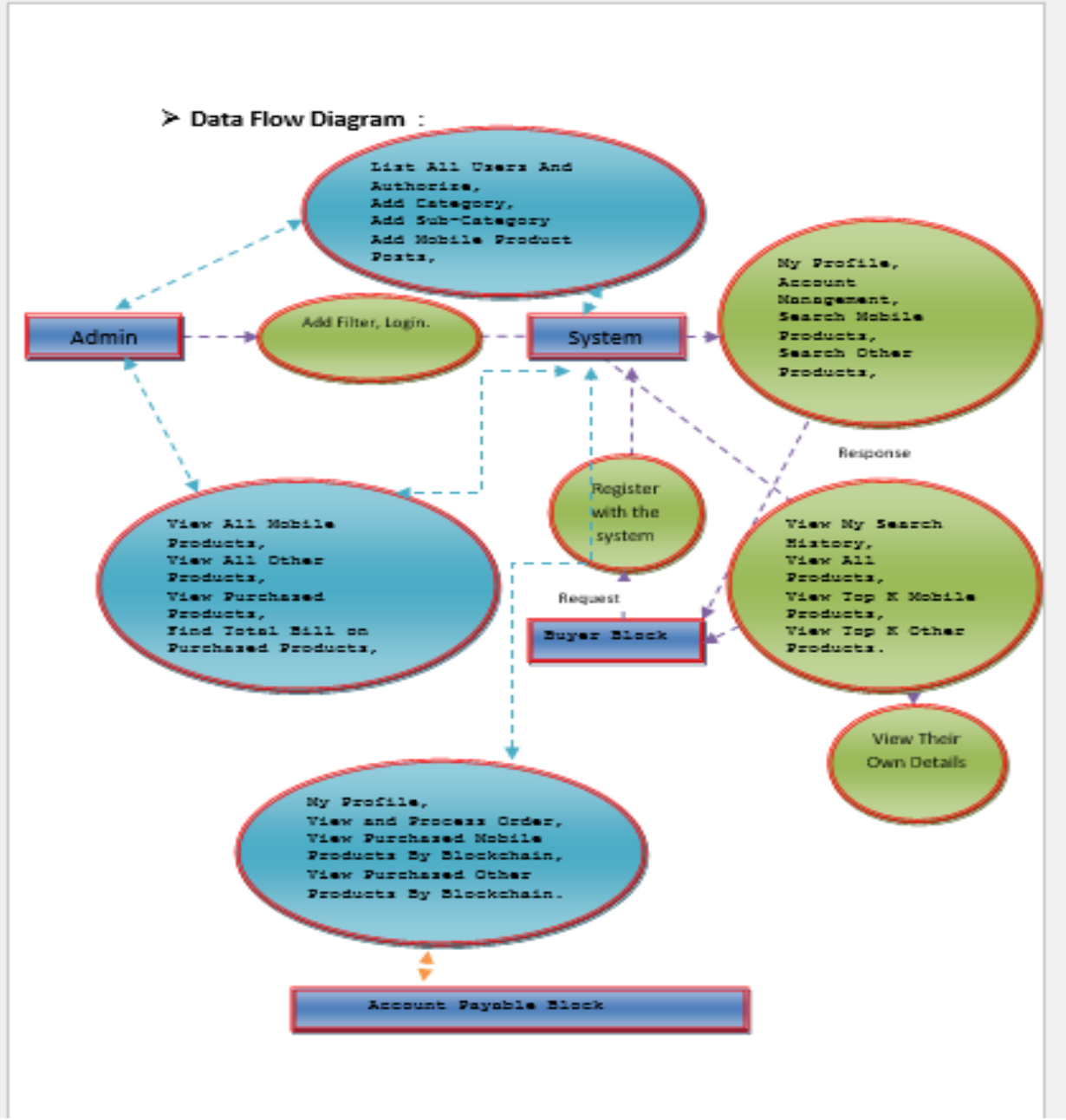
Economic Feasibility or Cost-benefit is an assessment of the economic justification for a computer based project. As hardware was installed from the beginning & for lots of purposes thus the cost on project of hardware is low. Since the system is a network based, any number of employees connected to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

Technical Feasibility

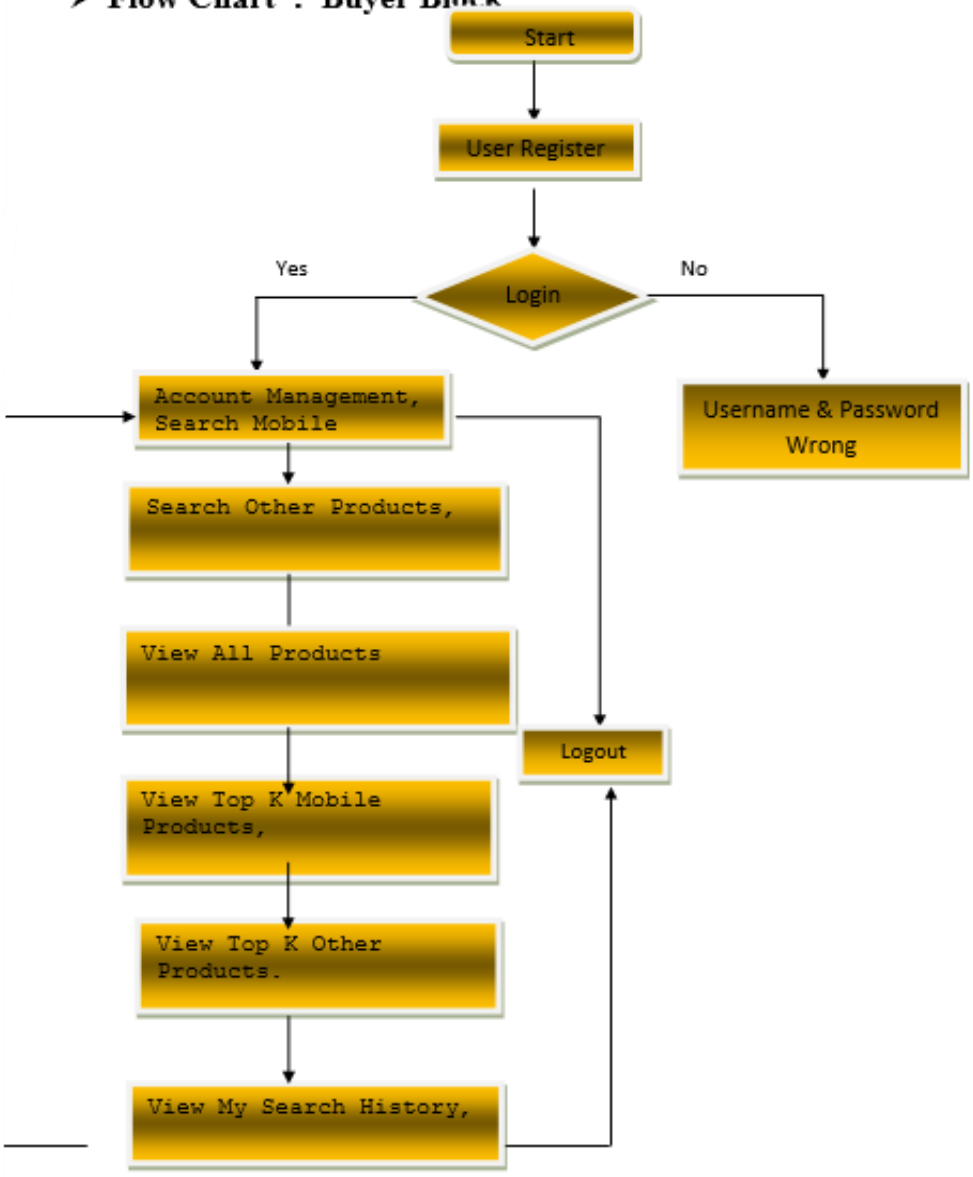
According to Roger S. Pressman, Technical Feasibility is the assessment of the technical resources of the organization. The organization needs IBM compatible machines with a graphical web browser connected to the Internet and Intranet. The system is developed for platform Independent environment. Java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

4.DESIGN METHODOLOGY

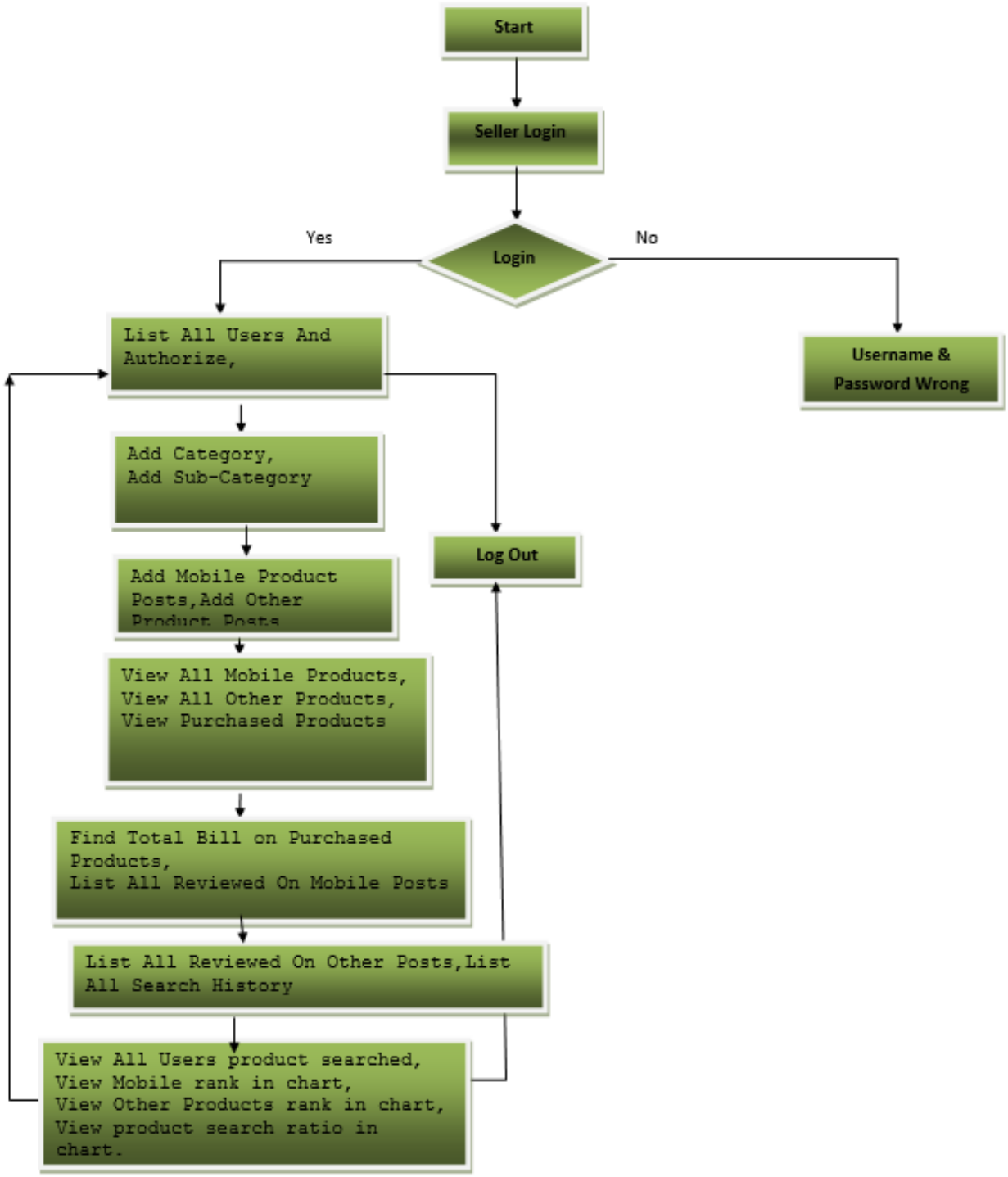
4.1 DATA FLOW DIAGRAM



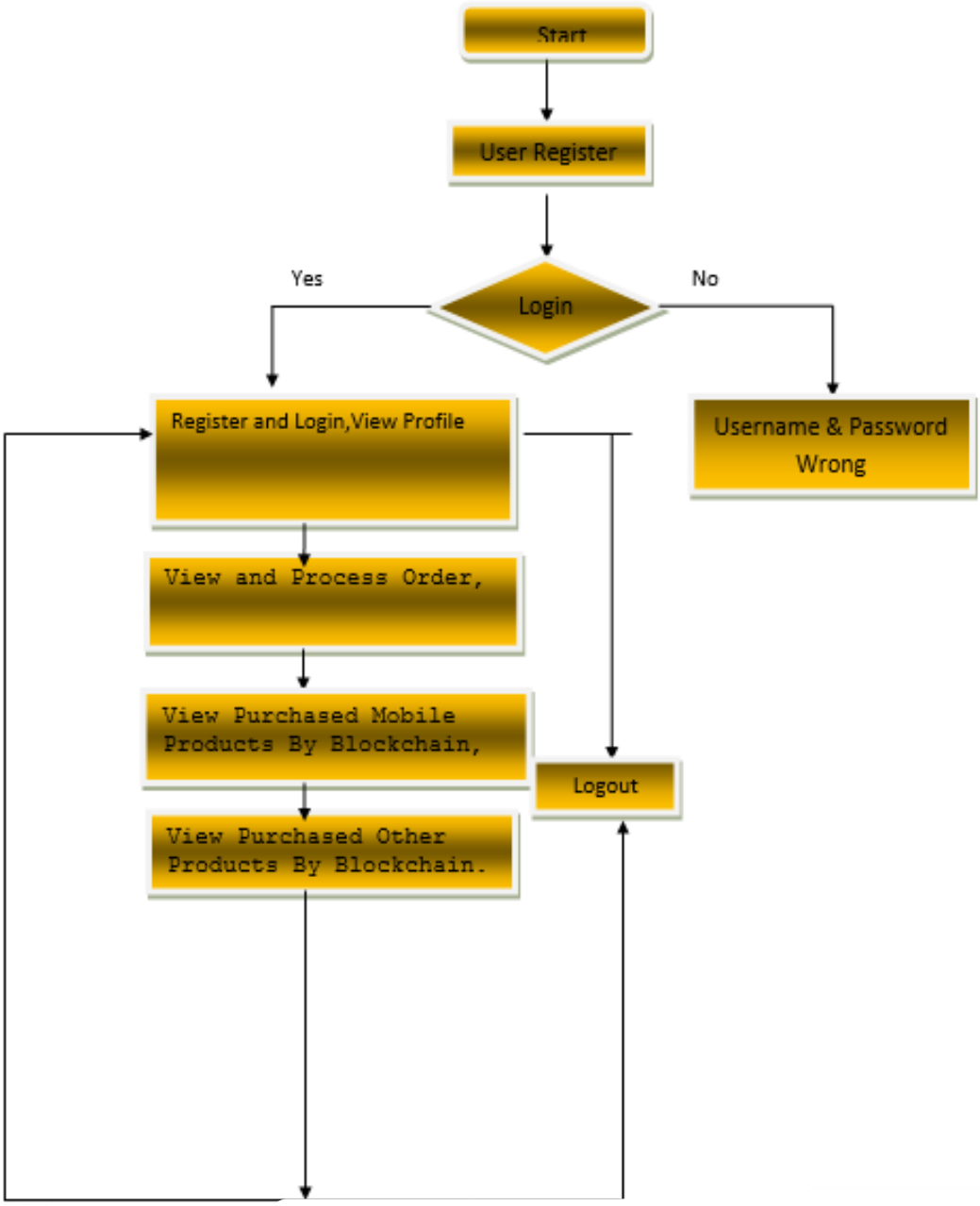
➢ **Flow Chart : Buyer Block**



Flow Chart : Seller Block



➢ **Flow Chart : Account Payable Block**



4.2 MODULES

IMPLEMENTATION

- **Seller Block**

In this module, the Seller Block has to login by using valid user name and password. After login successful he can do some operations such as List All Users And Authorize, Add Category, Add Sub-Category Add Mobile Product Posts, Add Other Product Posts, View All Mobile Products, View All Other Products, View Purchased Products, Find Total Bill on Purchased Products, List All Reviewed On Mobile Posts, List All Reviewed On Other Posts, List All Search History, View All Users product searched, View Mobile rank in chart, View Other Products rank in chart, View product search ratio in chart.

- **Account Payable Block**

Add App

In this module, there are n numbers of Account Payable Block are present. User should register before doing some operations. After registration successful he has to login by using authorized username and password. Login successful he will do some operations like the My Profile, View and Process Order, View Purchased Mobile Products By Blockchain,View Purchased Other Products By Block chain.

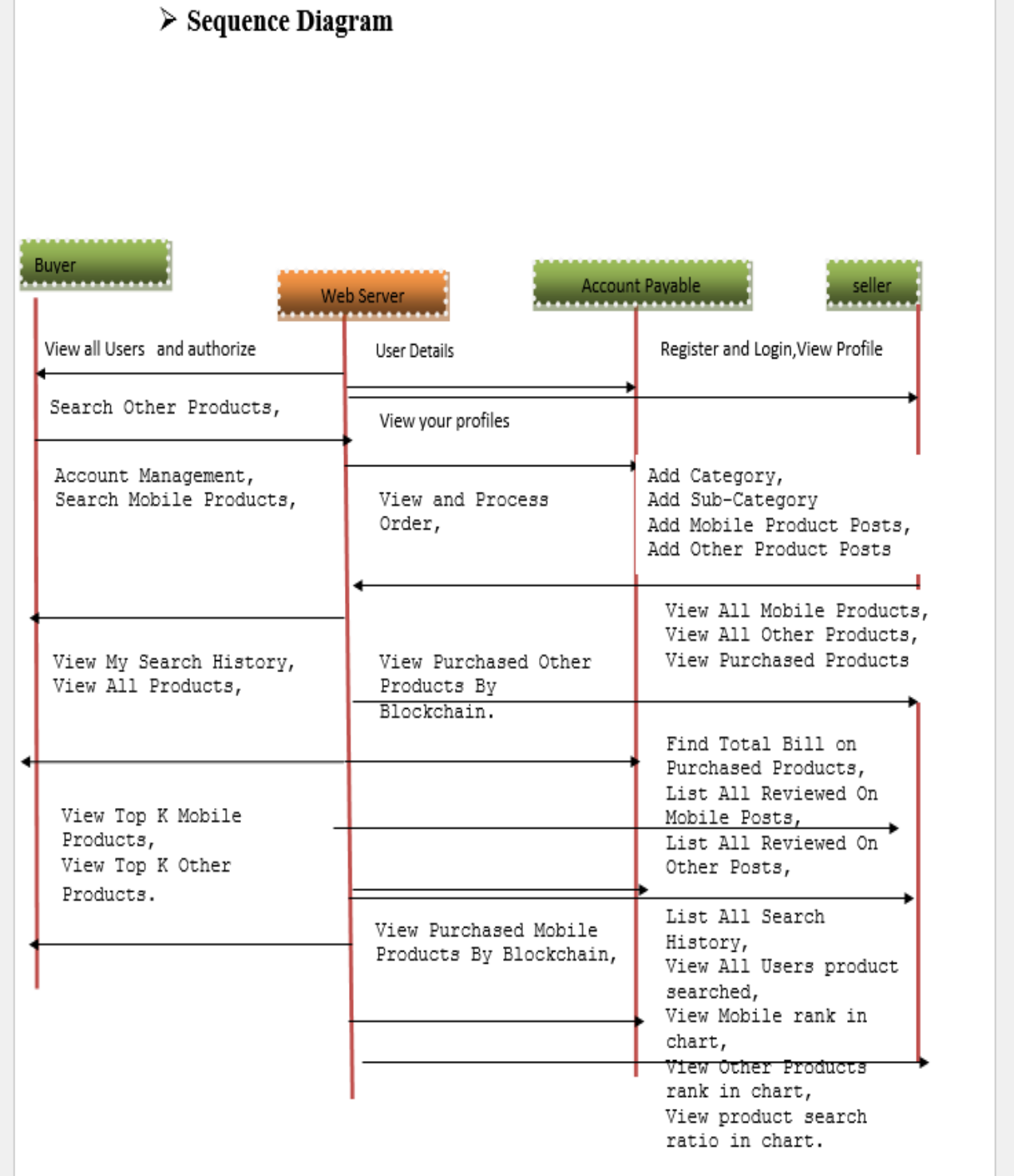
Buyer Block

In this module, there are n numbers of users are present. User should register before doing some operations. After registration successful he has to login by using authorized user name and password. Login successful he will do some operations like Account Management,Search Mobile Products,Search Other Products,View My Search History,View All Products,View Top K Mobile Products,View Top K Other Products.

Search for top K applications: In this module, user enter the application name and select the top N details then leading app details will be displayed such as applicationname,app description, mobile type, users, file name, application images and ratings will be displayed.

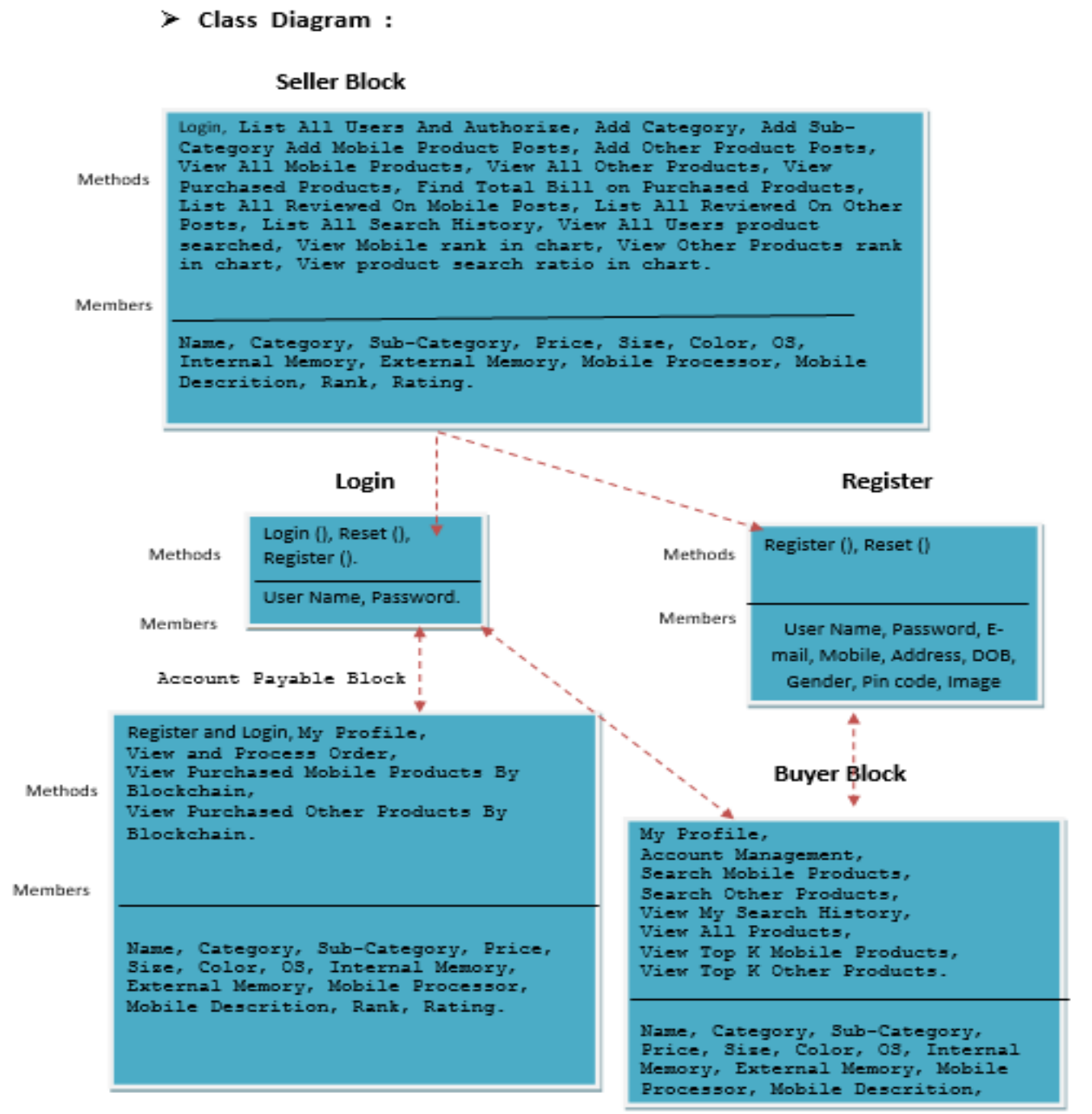
4.3 UML DIAGRAMS

SEQUENCE DIAGRAM:

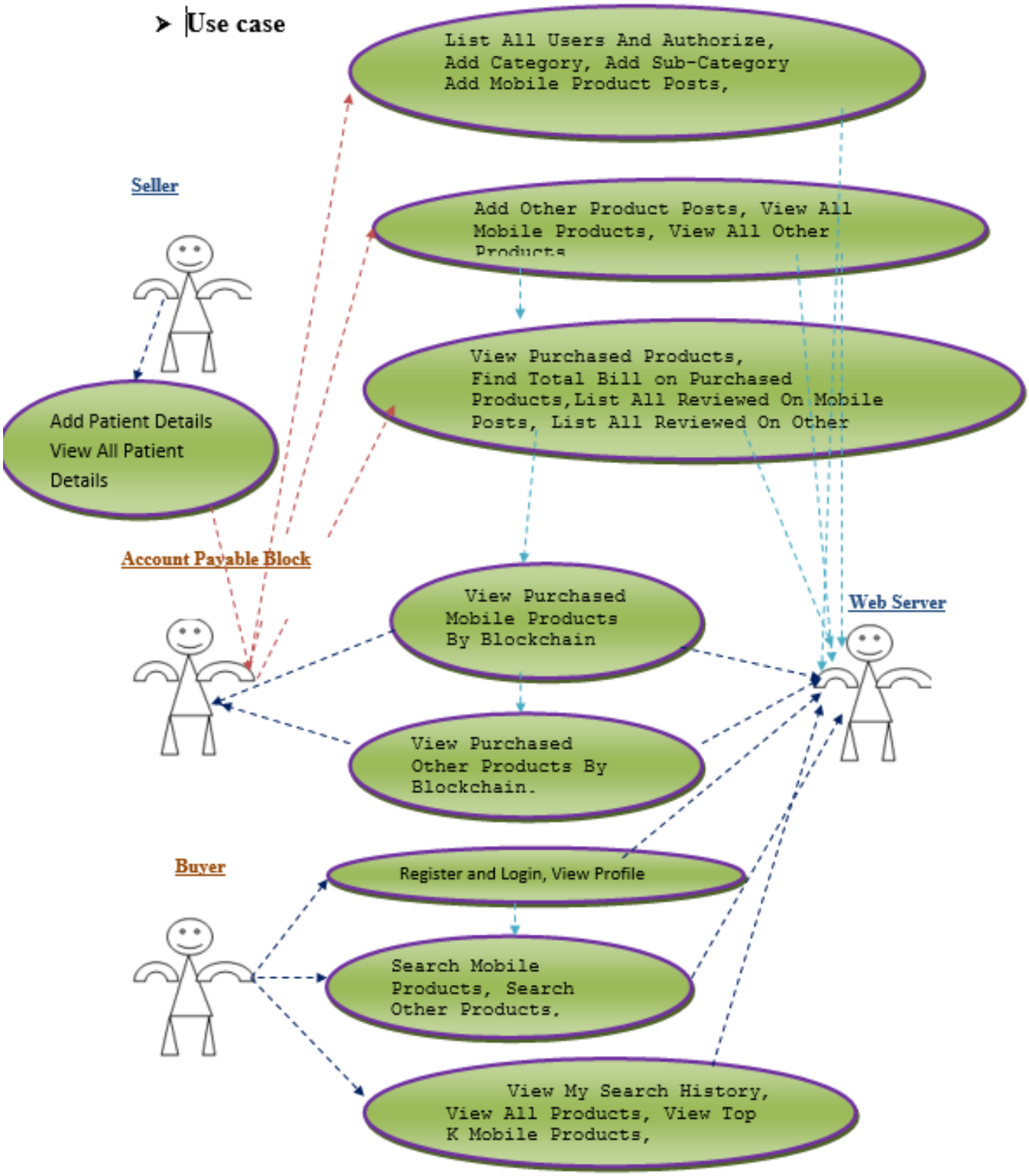


SEQUENCE DIAGRAM

CLASS DIAGRAM :



USE CASE DIAGRAM



5. SOFTWARE ENVIRONMENT

Java Technology

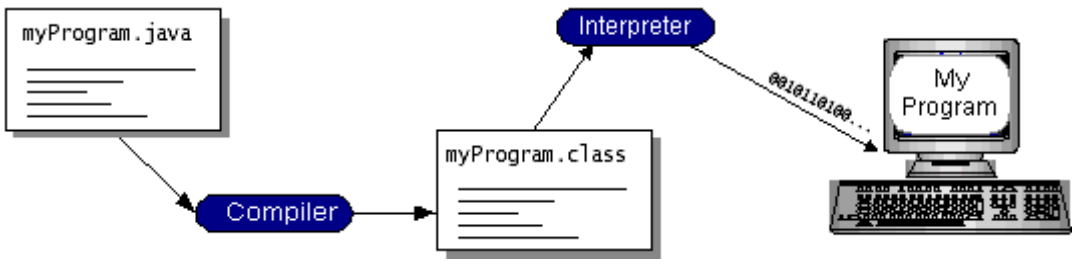
Java technology is both a programming language and a platform.

The Java Programming Language

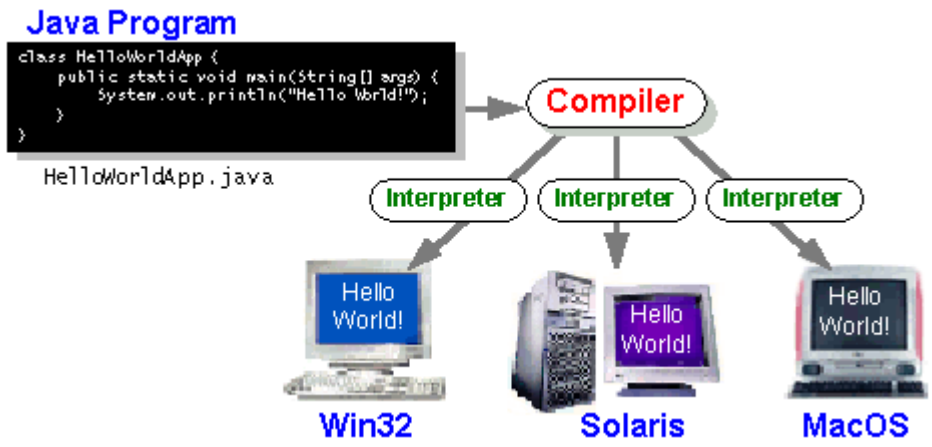
The Java programming language is a high-level language that can be characterized by all of the following buzzwords:

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

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



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



The Java Platform

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

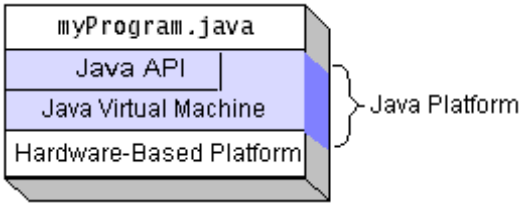
The Java platform has two components:

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

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

The Java API is a large collection of ready-made software components that provide many useful capabilities, such as graphical user interface (GUI) widgets. The Java API is grouped into libraries of related classes and interfaces; these libraries are known as *packages*. The next section, What Can Java Technology Do? Highlights what functionality some of the packages in the Java API provide.

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



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

What Can Java Technology Do?

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

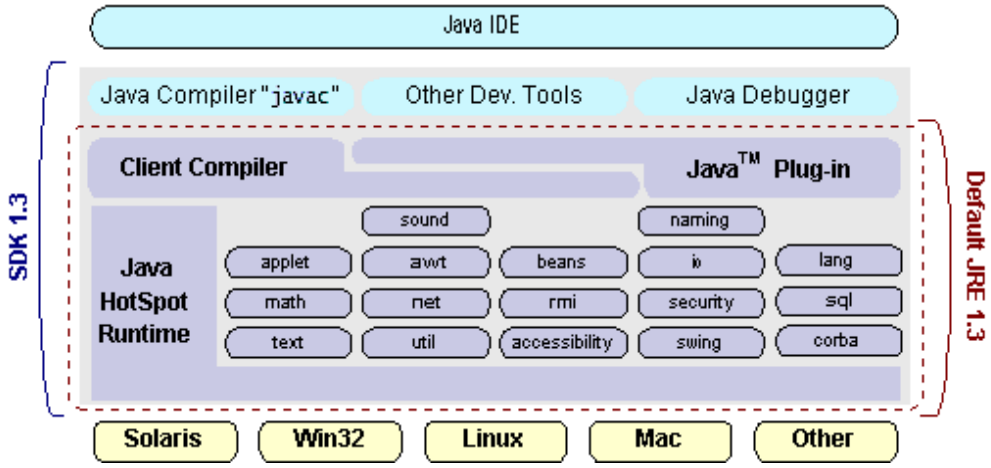
However, the Java programming language is not just for writing cute, entertaining applets for the Web. The general-purpose, high-level Java programming language is also a powerful software platform. Using the generous API, you can write many types of programs.

How does the API support all these kinds of programs? It does so with packages of software components that provides a wide range of functionality. Every full implementation of the Java platform gives you the following features:

- **The essentials**: Objects, strings, threads, numbers, input and output, data structures, system properties, date and time, and so on.
- **Applets**: The set of conventions used by applets.
- **Networking**: URLs, TCP (Transmission Control Protocol), UDP (User Data gram Protocol) sockets, and IP (Internet Protocol) addresses.
- **Internationalization**: Help for writing programs that can be localized for users worldwide. Programs can automatically adapt to specific locales and be displayed in the appropriate language.
- **Security**: Both low level and high level, including electronic signatures, public and private key management, access control, and certificates.

- **Software components**: Known as JavaBeans™, can plug into existing component architectures.
- **Object serialization**: Allows lightweight persistence and communication via Remote Method Invocation (RMI).
- **Java Database Connectivity (JDBC™)**: Provides uniform access to a wide range of relational databases.

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



ODBC

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

Through the ODBC Administrator in Control Panel, you can specify the particular database that is associated with a data source that an ODBC application program is written to use. Think of an ODBC data source as a door with a name on it. Each door will lead you to a particular database. For example, the data source named Sales Figures might be a SQL Server database, whereas the Accounts Payable data source

could refer to an Access database. The physical database referred to by a data source can reside anywhere on the LAN.

The ODBC system files are not installed on your system by Windows 95. Rather, they are installed when you setup a separate database application, such as SQL Server Client or Visual Basic 4.0. When the ODBC icon is installed in Control Panel, it uses a file called ODBCINST.DLL. It is also possible to administer your ODBC data sources through a stand-alone program called ODBCADM.EXE. There is a 16-bit and a 32-bit version of this program and each maintains a separate list of ODBC data sources.

The advantages of this scheme are so numerous that you are probably thinking there must be some catch. The only disadvantage of ODBC is that it isn't as efficient as talking directly to the native database interface. ODBC has had many detractors make the charge that it is too slow. Microsoft has always claimed that the critical factor in performance is the quality of the driver software that is used. In our humble opinion, this is true. The availability of good ODBC drivers has improved a great deal recently. And anyway, the criticism about performance is somewhat analogous to those who said that compilers would never match the speed of pure assembly language. Maybe not, but the compiler (or ODBC) gives you the opportunity to write cleaner programs, which means you finish sooner. Meanwhile, computers get faster every year.

JDBC

In an effort to set an independent database standard API for Java; Sun Microsystems developed Java Database Connectivity, or JDBC. JDBC offers a generic SQL database access mechanism that provides a consistent interface to a variety of RDBMSs. This consistent interface is achieved through the use of "plug-in" database connectivity modules, or *drivers*. If a database vendor wishes to have JDBC support, he or she must provide the driver for each platform that the database and Java run on.

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

JDBC was announced in March of 1996. It was released for a 90 day public review that ended June 8, 1996. Because of user input, the final JDBC v1.0 specification was released soon after.

The remainder of this section will cover enough information about JDBC for you to know what it is about and how to use it effectively. This is by no means a complete overview of JDBC. That would fill an entire book.

JDBC Goals

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

The goals that were set for JDBC are important. They will give you some insight as to why certain classes and functionalities behave the way they do. The eight design goals for JDBC are as follows:

1. SQL Level API

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

1. *SQL Conformance*

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

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

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

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

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

4. Keep it simple

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

5. Use strong, static typing wherever possible

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

6. Keep the common cases simple

Because more often than not, the usual SQL calls used by the programmer are simple `SELECT`'s, `INSERT`'s, `DELETE`'s and `UPDATE`'s, these queries should be simple to perform with JDBC. However, more complex SQL statements should also be possible.

Finally, we decided to proceed the implementation using Java Networking.

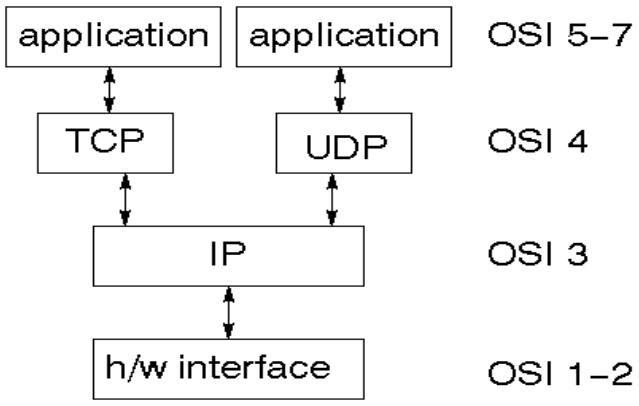
And for dynamically updating the cache table we go for MS Access database.

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

NETWORKING

TCP/IP stack

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



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

IP datagram's

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

UDP

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

TCP

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

Internet addresses

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

Network address

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

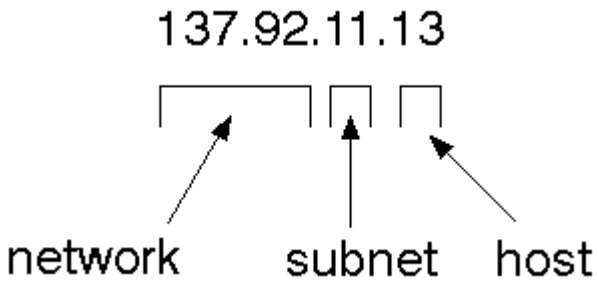
Subnet address

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

Host address

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

Total address



The 32 bit address is usually written as 4 integers separated by dots.

Port addresses

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

Sockets

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

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

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

JFree Chart

JFreeChart is a free 100% Java chart library that makes it easy for developers to display professional quality charts in their applications. JFreeChart's extensive feature set includes:

A consistent and well-documented API, supporting a wide range of chart types;

A flexible design that is easy to extend, and targets both server-side and client-side applications;

Support for many output types, including Swing components, image files (including PNG and JPEG), and vector graphics file formats (including PDF, EPS and SVG);

J2EE SOFTWARE ENVIRONMENT:

Client Server

Overview:

With the varied topic in existence in the fields of computers, Client Server is one, which has generated more heat than light, and also more hype than reality. This technology has acquired a certain critical mass attention with its dedication conferences and magazines. Major computer vendors such as IBM and DEC, have declared that Client Servers is their main future market. A survey of DBMS magazine reveled that 76% of its readers were actively looking at the client server solution. The growth in the client server development tools from $200 million in 1992 to more than $1.2 billion in 1996.

Client server implementations are complex but the underlying concept is simple and powerful. A client is an application running with local resources but able to request the database and relate the services from separate remote server. The software mediating this client server interaction is often referred to as MIDDLEWARE.

The typical client either a PC or a Work Station connected through a network to a more powerful PC, Workstation, Midrange or Main Frames server usually capable of handling request from more than one client. However, with some configuration server may also act as client. A server may need to access other server in order to process the original client request.

The key client server idea is that client as user is essentially insulated from the physical location and formats of the data needs for their application. With the proper middleware, a client input from or report can transparently access and manipulate both local database on the client machine and remote databases on one or more servers. An added bonus is the client server opens the door to multi-vendor database access indulging heterogeneous table joins.

What is a Client Server

Two prominent systems in existence are client server and file server systems. It is essential to distinguish between client servers and file server systems. Both provide shared network access to data but the comparison dens there! The file server simply provides a remote disk drive that can be accessed by LAN applications on a file by file basis. The client server offers full relational database services such as SQL-Access, Record modifying, Insert, Delete with full relational integrity backup/ restore performance

for high volume of transactions, etc. the client server middleware provides a flexible interface between client and server, who does what, when and to whom.

Why Client Server

Client server has evolved to solve a problem that has been around since the earliest days of computing: how best to distribute your computing, data generation and data storage resources in order to obtain efficient, cost effective departmental an enterprise wide data processing. During mainframe era choices were quite limited. A central machine housed both the CPU and DATA (cards, tapes, drums and later disks). Access to these resources was initially confined to batched runs that produced departmental reports at the appropriate intervals. A strong central information service department ruled the corporation. The role of the rest of the corporation limited to requesting new or more frequent reports and to provide hand written forms from which the central data banks were created and updated. The earliest client server solutions therefore could best be characterized as “SLAVE-MASTER”.

Time-sharing changed the picture. Remote terminal could view and even change the central data, subject to access permissions. And, as the central data banks evolved in to sophisticated relational database with non-programmer query languages, online users could formulate adhoc queries and produce local reports with out adding to the MIS applications software backlog. However remote access was through dumb terminals, and the client server remained subordinate to the Slave\Master**.**

Frontend or User Interface Design

The entire user interface is planned to be developed in browser specific environment with a touch of Intranet-Based Architecture for achieving the Distributed Concept.

The browser specific components are designed by using the HTML standards, and the dynamism of the designed by concentrating on the constructs of the Java Server Pages.

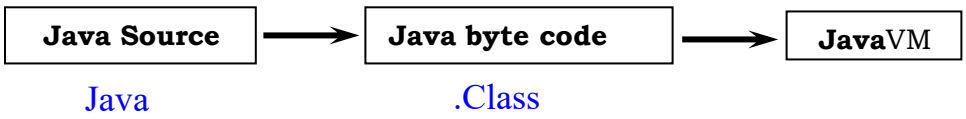
- **Communication or Database Connectivity Tier**

The Communication architecture is designed by concentrating on the Standards of Servlets and Enterprise Java Beans. The database connectivity is established by using the Java Data Base Connectivity.

The standards of three-tire architecture are given major concentration to keep the standards of higher cohesion and limited coupling for effectiveness of the operations.

- **Importance of Java to the Internet**

Java has had a profound effect on the Internet. This is because; Java expands the Universe of objects that can move about freely in Cyberspace. In a network, two categories of objects are transmitted between the Server and the Personal computer. They are: Passive information and Dynamic active programs. The Dynamic, Self-executing programs cause serious problems in the areas of Security and probability. But, Java addresses those concerns and by doing so, has opened the door to an exciting new form of program called the Applet.



Picture showing the development process of JAVA Program

Java programming uses to produce byte codes and executes them. The first box indicates that the Java source code is located in a. Java file that is processed with a Java compiler called javac. The Java compiler produces a file called a. class file, which contains the byte code. The. Class file is then loaded across the network or loaded locally on your machine into the execution environment is the Java virtual machine, which interprets and executes the byte code.

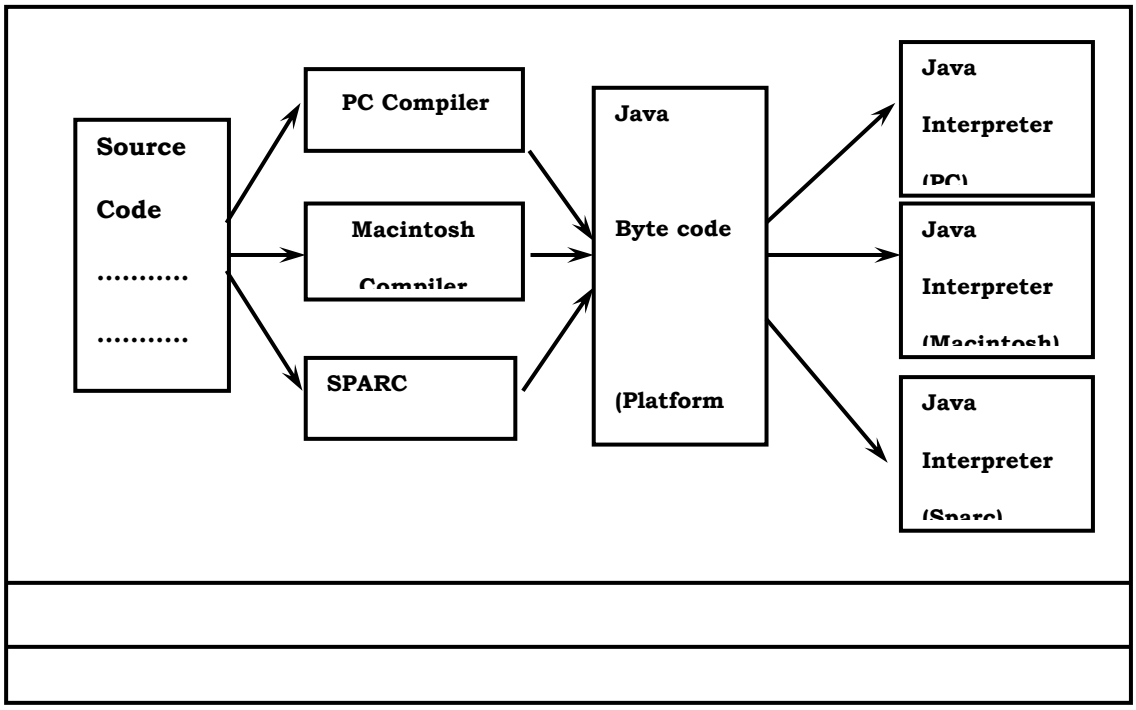
Java Architecture

Java architecture provides a portable, robust, high performing environment for development. Java provides portability by compiling the byte codes for the Java Virtual Machine, which is then interpreted on each platform by the run-time environment. Java is a dynamic system, able to load code when needed from a machine in the same room or across the planet.

Compilation of code

When you compile the code, the Java compiler creates machine code (called byte code) for a hypothetical machine called Java Virtual Machine (JVM). The JVM is supposed to execute the byte code. The JVM is created for overcoming the issue of portability. The code is written and compiled for one machine and interpreted on all machines. This machine is called Java Virtual Machine.

Compiling and interpreting Java Source Code



During run-time the Java interpreter tricks the byte code file into thinking that it is running on a Java Virtual Machine. In reality this could be a Intel Pentium Windows 95 or Sun SARC station running Solaris or Apple Macintosh running system and all could receive code from any computer through Internet and run the Applets.

Simple

Java was designed to be easy for the Professional programmer to learn and to use effectively. If you are an experienced C++ programmer, learning Java will be even easier. Because Java inherits the C/C++ syntax and many of the object oriented features of C++. Most of the confusing concepts from C++ are either left out of Java or implemented in a cleaner, more approachable manner. In Java there are a small number of clearly defined ways to accomplish a given task.

Object-Oriented

Java was not designed to be source-code compatible with any other language. This allowed the Java team the freedom to design with a blank slate. One outcome of this was a clean usable, pragmatic approach to objects. The object model in Java is simple and easy to extend, while simple types, such as integers, are kept as high-performance non-objects.

Robust

The multi-platform environment of the Web places extraordinary demands on a program, because the program must execute reliably in a variety of systems. The ability to create robust programs was given a high priority in the design of Java. Java is strictly typed language; it checks your code at compile time and run time.

Java virtually eliminates the problems of memory management and deallocation, which is completely automatic. In a well-written Java program, all run time errors can –and should –be managed by your program.

JAVASCRIPT

JavaScript is a script-based programming language that was developed by Netscape Communication Corporation. JavaScript was originally called Live Script and renamed as JavaScript to indicate its relationship with Java. JavaScript supports the development of both client and server components of Web-based applications. On the client side, it can be used to write programs that are executed by a Web browser within the context of a Web page. On the server side, it can be used to write Web server programs that can process information submitted by a Web browser and then updates the browser's display accordingly

Even though JavaScript supports both client and server Web programming, we prefer JavaScript at Client side programming since most of the browsers supports it. JavaScript is almost as easy to learn as HTML, and JavaScript statements can be included in HTML documents by enclosing the statements between a pair of scripting tags <SCRIPTS>..</SCRIPT>.

```
<SCRIPT LANGUAGE = "JavaScript">

JavaScript statements

</SCRIPT>
```

Here are a few things we can do with JavaScript :

- Validate the contents of a form and make calculations.
- Add scrolling or changing messages to the Browser's status line.
- Animate images or rotate images that change when we move the mouse over them.
- Detect the browser in use and display different content for different browsers.
- Detect installed plug-ins and notify the user if a plug-in is required.

We can do much more with JavaScript, including creating entire application.

Hyper Text Markup Language

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

A markup language is simply a series of elements, each delimited with special characters that define how text or other 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 tags are not case-sensitive. Using graphics, fonts, different sizes, color, etc., can enhance the presentation of the document. Anything that is not a tag is part of the document itself.

Basic HTML Tags :

<!-- -->	Specifies comments
<A>…….</A>	Creates hypertext links
<B>…….</B>	Formats text as bold
<BIG>…….</BIG>	Formats text in large font.
<BODY>…</BODY>	Contains all tags and text in the HTML document
<CENTER>...</CENTER>	Creates text
<DD>…</DD>	Definition of a term
<DL>...</DL>	Creates definition list
<FONT>…</FONT>	Formats text with a particular font
<FORM>...</FORM>	Encloses a fill-out form
<FRAME>...</FRAME>	Defines a particular frame in a set of frames
<H#>…</H#>	Creates headings of different levels
<HEAD>...</HEAD>	Contains tags that specify information about a document
<HR>...</HR>	Creates a horizontal rule

<HTML>…</HTML>	Contains all other HTML tags
<META>...</META>	Provides meta-information about a document
<SCRIPT>…</SCRIPT>	Contains client-side or server-side script
<TABLE>…</TABLE>	Creates a table
<TD>…</TD>	Indicates table data in a table
<TR>…</TR>	Designates a table row
<TH>…</TH>	Creates a heading in a table

ADVANTAGES

- A HTML document is small and hence easy to send over the net. It is small because it does not include formatted information.
- HTML is platform independent.
- HTML tags are not case-sensitive.

Java Database Connectivity

What Is JDBC?

JDBC is a Java API for executing SQL statements. (As a point of interest, JDBC is a trademarked name 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. JDBC provides a standard API for tool/database developers and makes it possible to write database applications using a pure Java API.

Using JDBC, it is easy to send SQL statements to virtually any relational database. One can write a single program using the JDBC API, and the program will be able to send SQL statements to the appropriate database. The combinations of Java and JDBC lets a programmer write it once and run it anywhere.

What Does JDBC Do? Simply put, JDBC makes it possible to do three things:

- Establish a connection with a database
- Send SQL statements
- Process the results.

JDBC versus ODBC and other APIs

At this point, Microsoft's ODBC (Open Database Connectivity) API is that probably the most widely used programming interface for accessing relational databases. It offers the ability to connect to almost all databases on almost all platforms.

So why not just use ODBC from Java? The answer is that you can use ODBC from Java, but this is best done with the help of JDBC in the form of the JDBC-ODBC Bridge, which we will cover shortly. The question now becomes "Why do you need JDBC?" There are several answers to this question:

1. ODBC is not appropriate for direct use from Java because it uses a C interface. Calls from Java to native C code have a number of drawbacks in the security, implementation, robustness, and automatic portability of applications.
2. A literal translation of the ODBC C API into a Java API would not be desirable. For example, Java has no pointers, and ODBC makes copious use of them, including the notoriously error-prone generic pointer "void *". You can think of JDBC as ODBC translated into an object-oriented interface that is natural for Java programmers.
3. ODBC is hard to learn. It mixes simple and advanced features together, and it has complex options even for simple queries. JDBC, on the other hand, was designed to keep simple things simple while allowing more advanced capabilities where required.
4. A Java API like JDBC is needed in order to enable a "pure Java" solution. When ODBC is used, the ODBC driver manager and drivers must be manually installed on every client machine. When the JDBC driver is written completely in Java, however, JDBC code is automatically installable, portable, and secure on all Java platforms from network computers to mainframes.

Java Server Pages (JSP)

Java server Pages is a simple, yet powerful technology for creating and maintaining dynamic-content web pages. Based on the Java programming language, Java Server Pages offers proven portability, open standards, and a mature re-usable component model .The Java Server Pages architecture enables the separation of content generation from content presentation. This separation not eases maintenance headaches, it also allows web team members to focus on their areas of expertise. Now, web page designer can concentrate on layout, and web application designers on programming, with minimal concern about impacting each other's work.

Features of JSP

Portability:

Java Server Pages files can be run on any web server or web-enabled application server that provides support for them. Dubbed the JSP engine, this support involves recognition, translation, and management of the Java Server Page lifecycle and its interaction components.

Components

It was mentioned earlier that the Java Server Pages architecture can include reusable Java components. The architecture also allows for the embedding of a scripting language directly into the Java Server Pages file. The components current supported include Java Beans, and Servlets.

Processing

A Java Server Pages file is essentially an HTML document with JSP scripting or tags. The Java Server Pages file has a JSP extension to the server as a Java Server Pages file. Before the page is served, the Java Server Pages syntax is parsed and processed into a Servlet on the server side. The Servlet that is generated outputs real content in straight HTML for responding to the client.

Access Models:

A Java Server Pages file may be accessed in at least two different ways. A client's request comes directly into a Java Server Page. In this scenario, suppose the page accesses reusable Java Bean components that perform particular well-defined computations like accessing a database. The result of the Beans computations, called result sets is stored within the Bean as properties. The page uses such Beans to generate dynamic content and present it back to the client.

In both of the above cases, the page could also contain any valid Java code. Java Server Pages architecture encourages separation of content from presentation.

Steps in the execution of a JSP Application:

1. The client sends a request to the web server for a JSP file by giving the name of the JSP file within the form tag of a HTML page.

2. This request is transferred to the JavaWebServer. At the server side JavaWebServer receives the request and if it is a request for a jsp file server gives this request to the JSP engine.
3. JSP engine is program which can understands the tags of the jsp and then it converts those tags into a Servlet program, and it is stored at the server side. This Servlet is loaded in the memory and then it is executed, and the result is given back to the JavaWebServer and then it is transferred back to the result is given back to the JavaWebServer and then it is transferred back to the client.

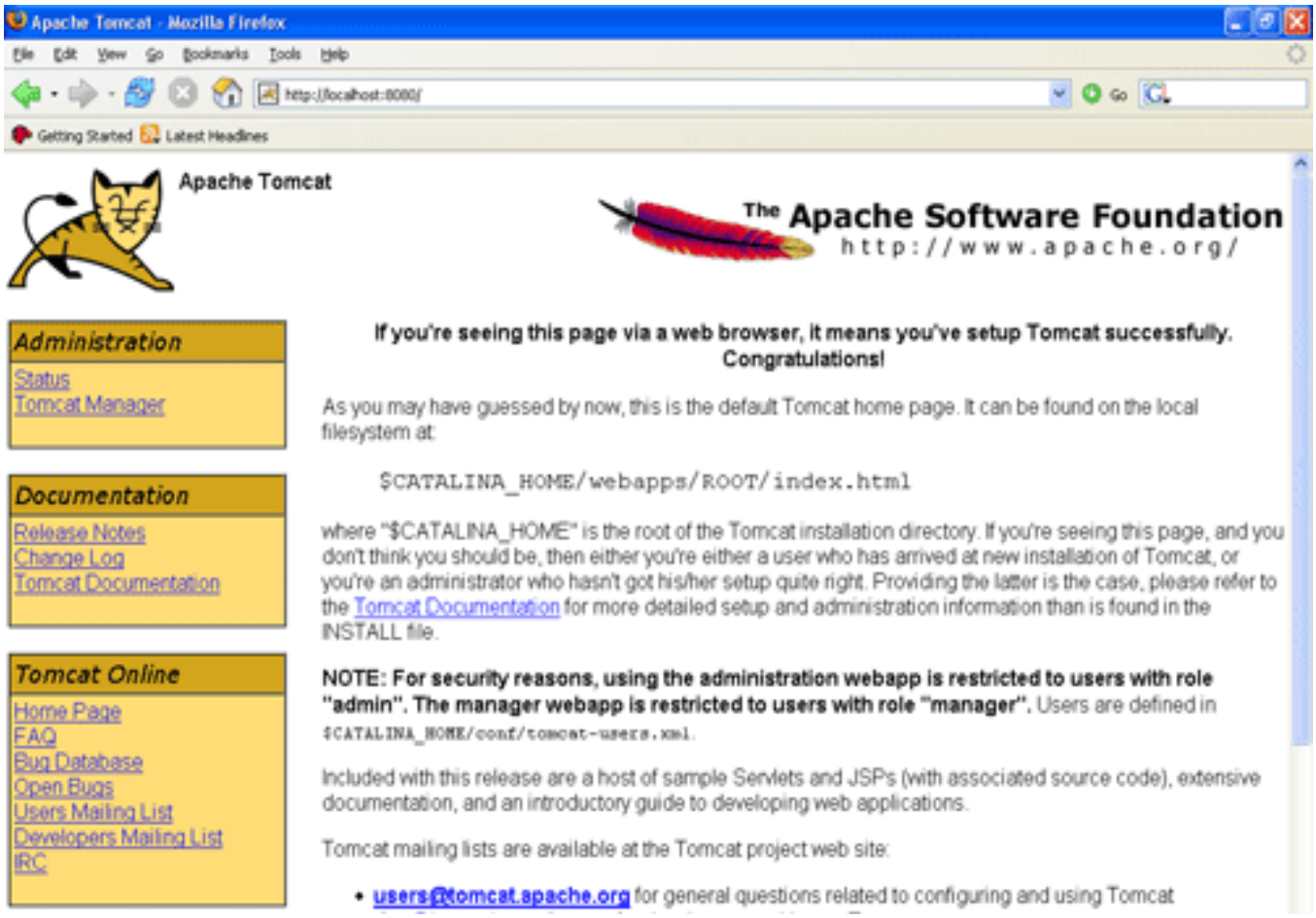
JDBC CONNECTIVITY

The JDBC provides database-independent connectivity between the J2EE platform and a wide range of tabular data sources. JDBC technology allows an Application Component Provider to:

- Perform connection and authentication to a database server
- Manager transactions
- Move SQL statements to a database engine for preprocessing and execution
- Execute stored procedures
- Inspect and modify the results from Select statements.

Tomcat 6.0 web server

Tomcat is an open-source web server developed by Apache Group. Apache Tomcat is the servlet container that is used in the official Reference Implementation for the Java Servlet and Java Server Pages technologies. The Java Servlet and Java Server Pages specifications are developed by Sun under the Java Community Process. Web Servers like Apache Tomcat support only web components while an application server supports web components as well as business components (BEAs Weblogic, is one of the popular application servers).To develop a web application with jsp/servlet install any web server like JRun, Tomcat etc to run your application.



Bibliography:

References for the Project Development were taken from the following Books and Web Sites.

Oracle

PL/SQL Programming by Scott Urman

SQL complete reference by Livion

JAVA Technologies

JAVA Complete Reference

Java Script Programming by Yehuda Shiran

Mastering JAVA Security

JAVA2 Networking by Pistoria

JAVA Security by Scotl oaks

Head First EJB Sierra Bates

J2EE Professional by Shadab siddiqui

JAVA server pages by Larne Pekowsley

JAVA Server pages by Nick Todd

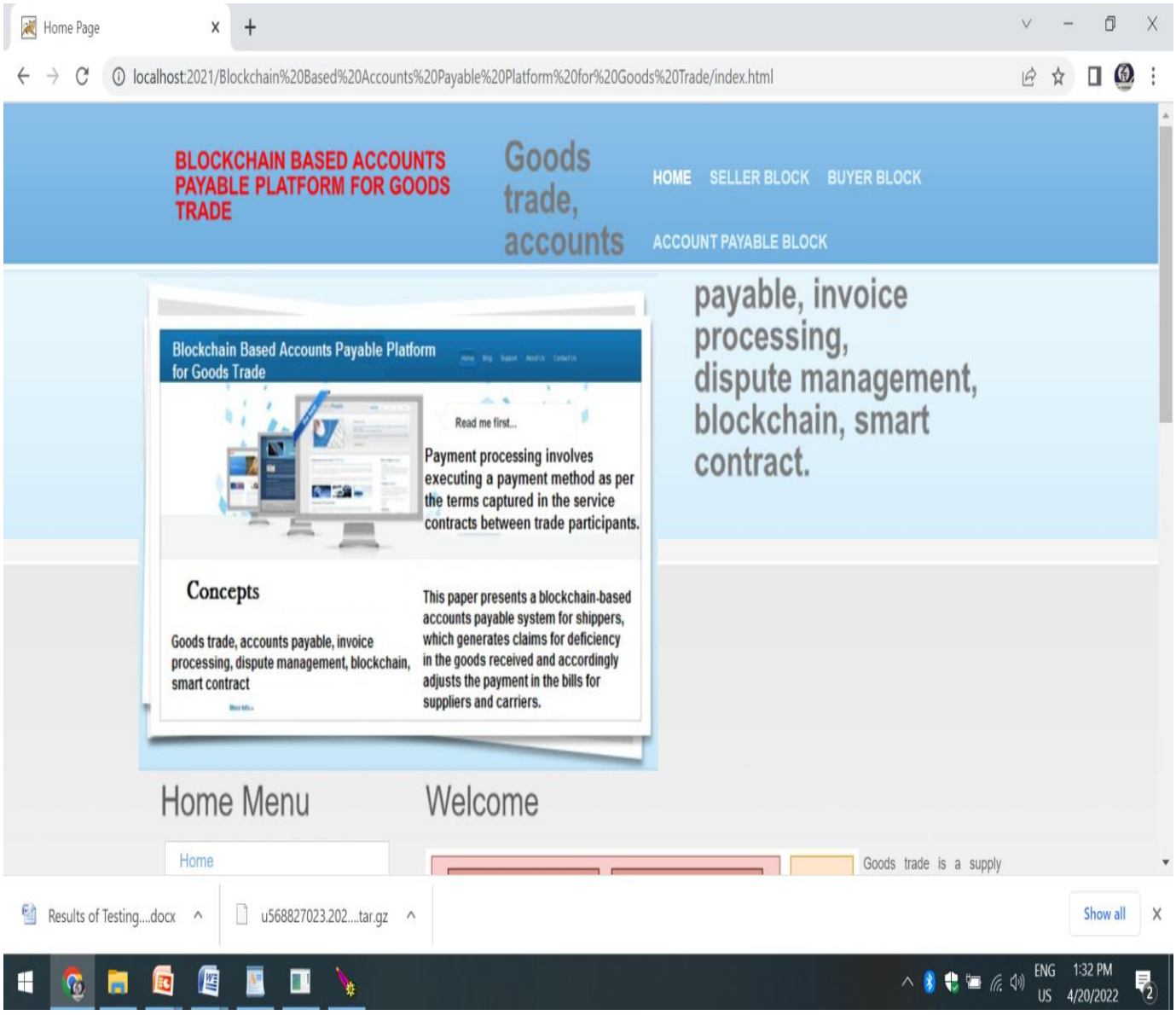
HTML

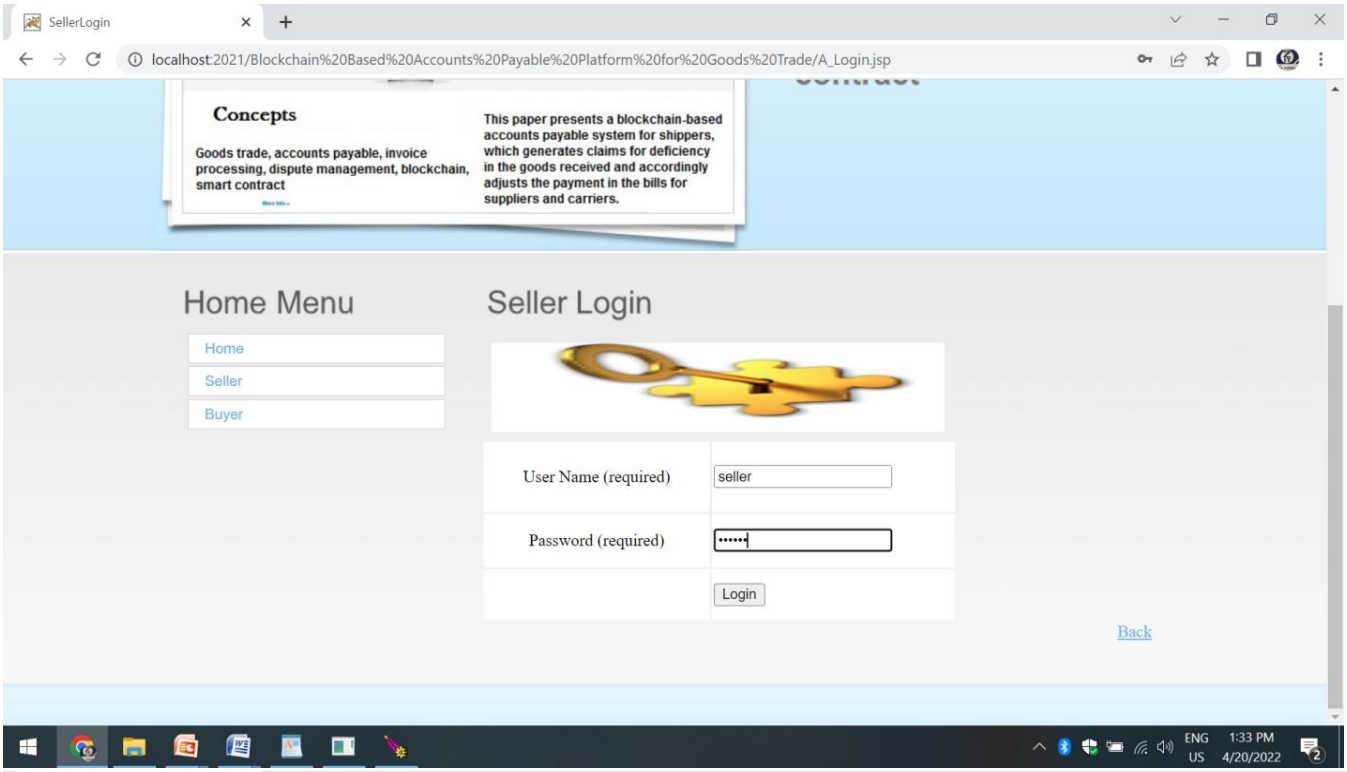
HTML Black Book by Holzner

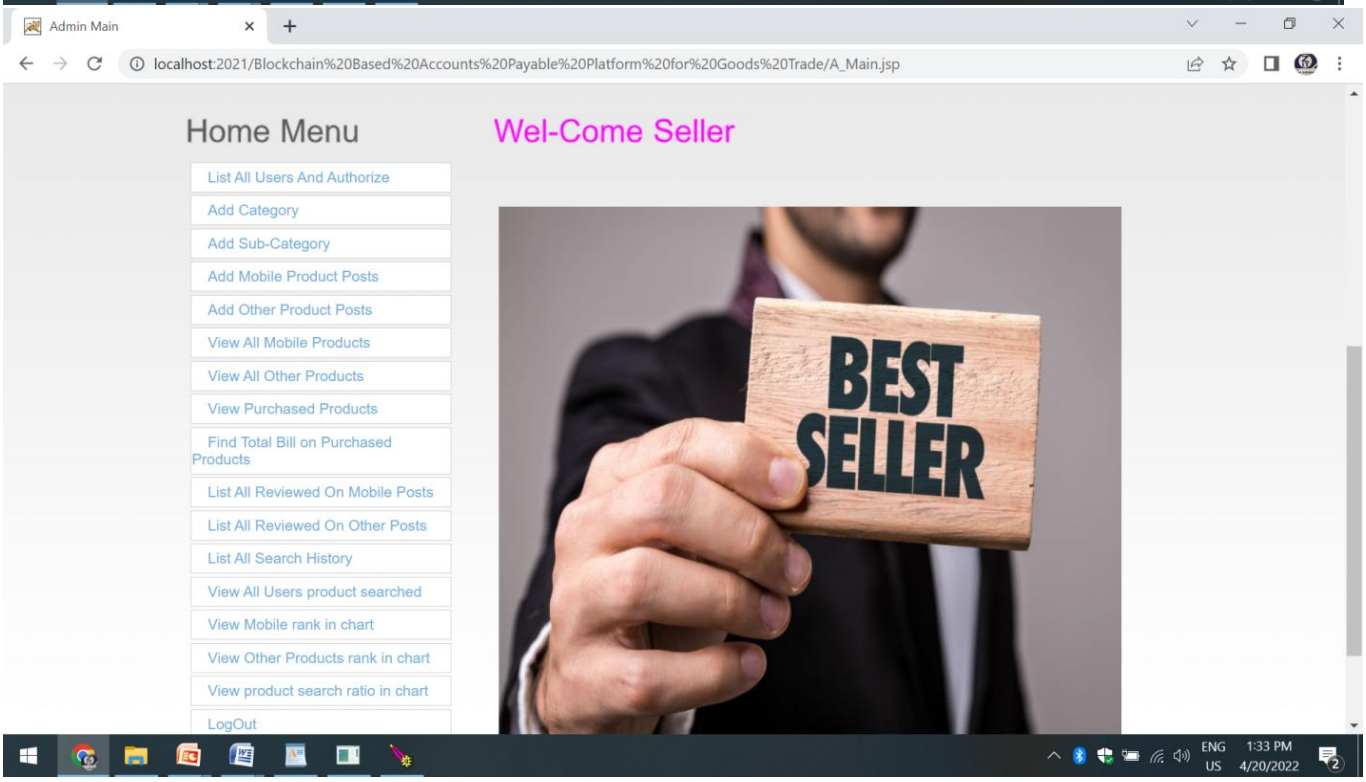
JDBC

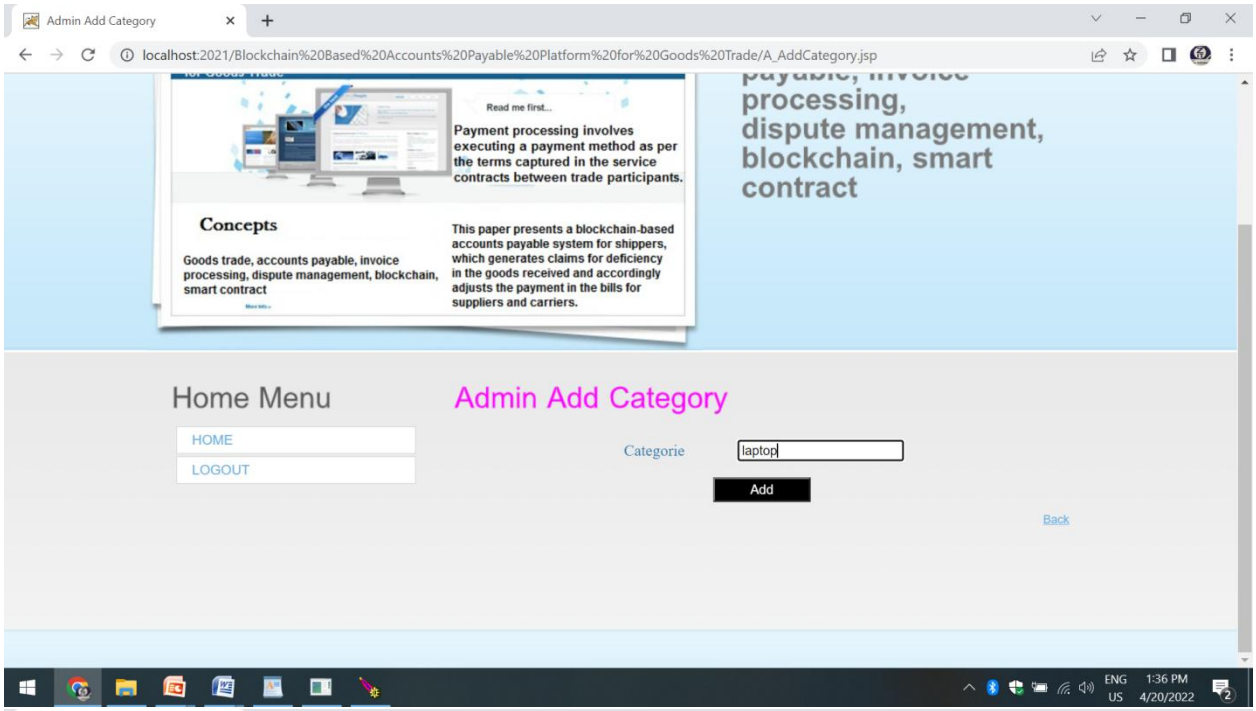
Java Database Programming with JDBC by Patel moss.

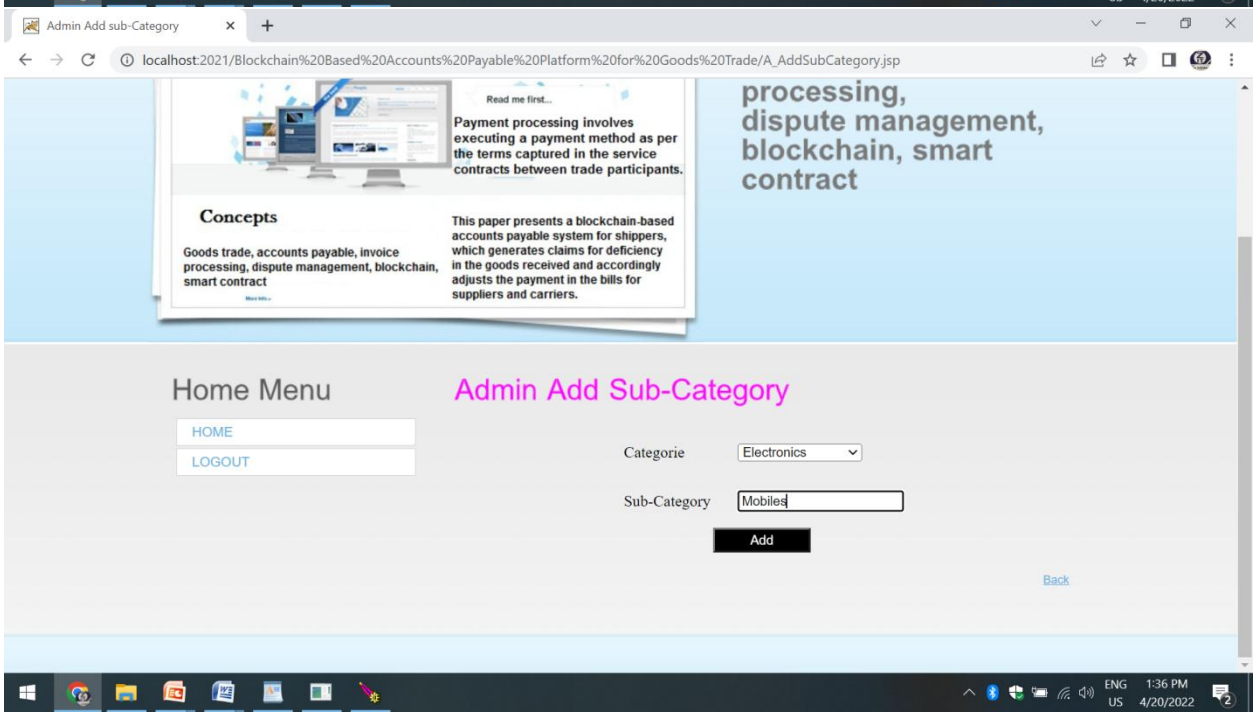
7.OUTPUT SCREENS

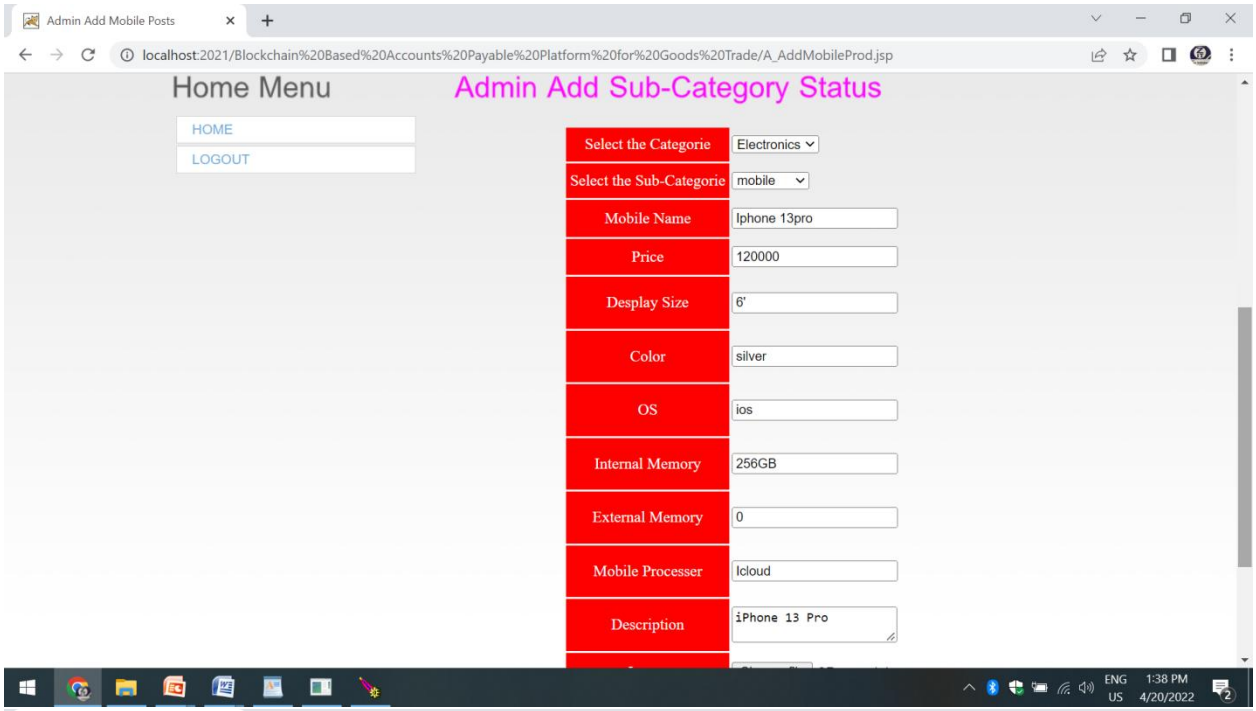


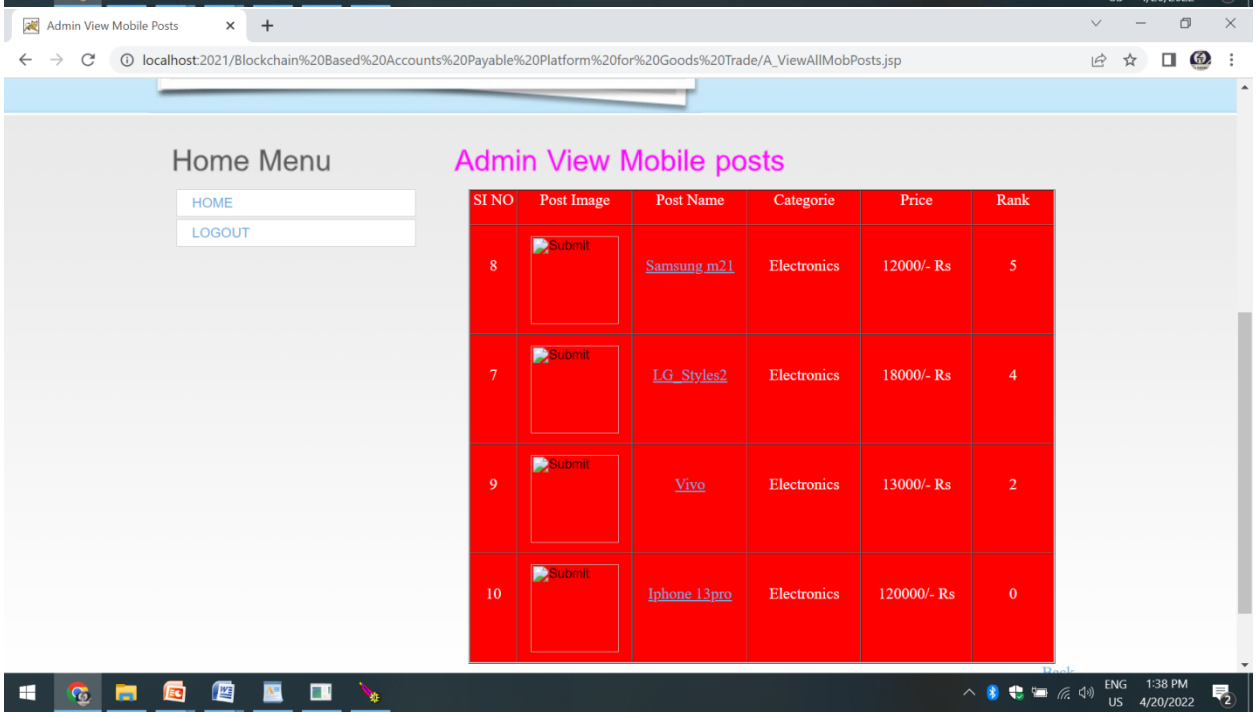

SellerLogin
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/A_Login.jsp
Concepts
Goods trade, accounts payable, invoice processing, dispute management, blockchain, smart contract
This paper presents a blockchain-based accounts payable system for shippers, which generates claims for deficiency in the goods received and accordingly adjusts the payment in the bills for suppliers and carriers.
Home Menu
Home
Seller
Buyer
Seller Login
User Name (required)
seller
Password (required)
Login
Back
ENG US
1:33 PM
4/20/2022

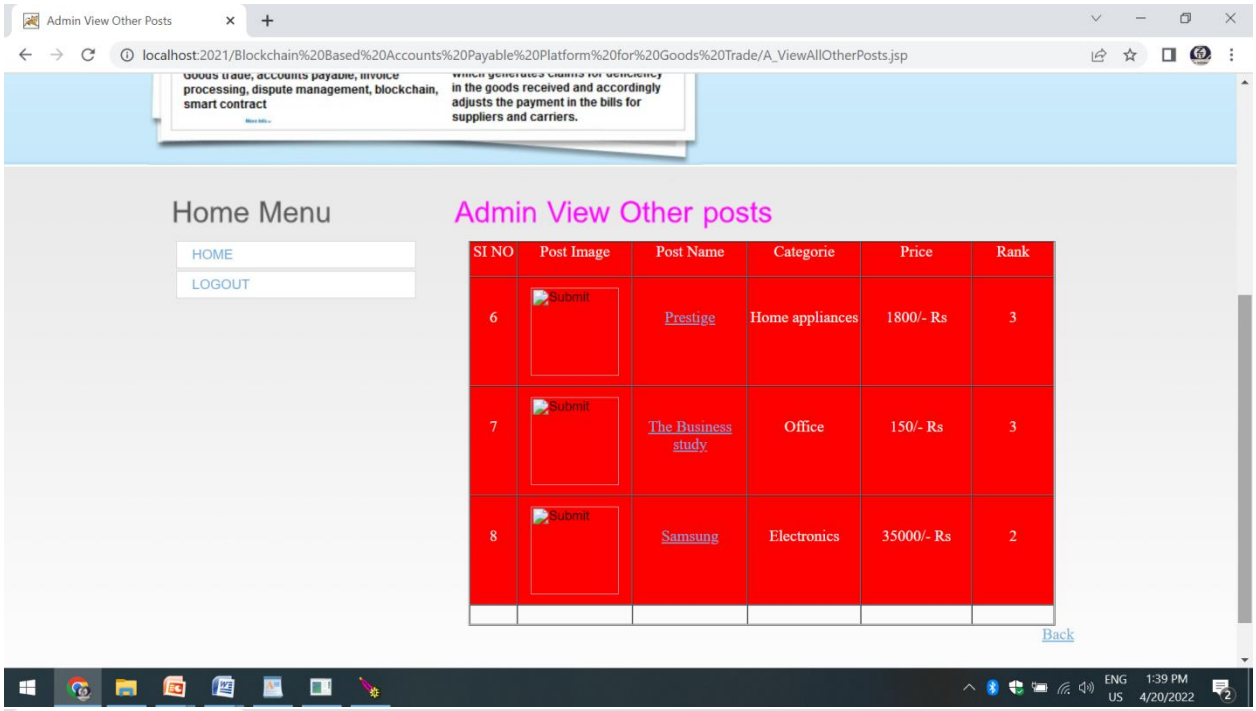

Admin Main
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/A_Main.jsp
Home Menu
List All Users And Authorize
Add Category
Add Sub-Category
Add Mobile Product Posts
Add Other Product Posts
View All Mobile Products
View All Other Products
View Purchased Products
Find Total Bill on Purchased Products
List All Reviewed On Mobile Posts
List All Reviewed On Other Posts
List All Search History
View All Users product searched
View Mobile rank in chart
View Other Products rank in chart
View product search ratio in chart
LogOut
Wel-Come Seller
BEST SELLER
ENG US
1:33 PM
4/20/2022

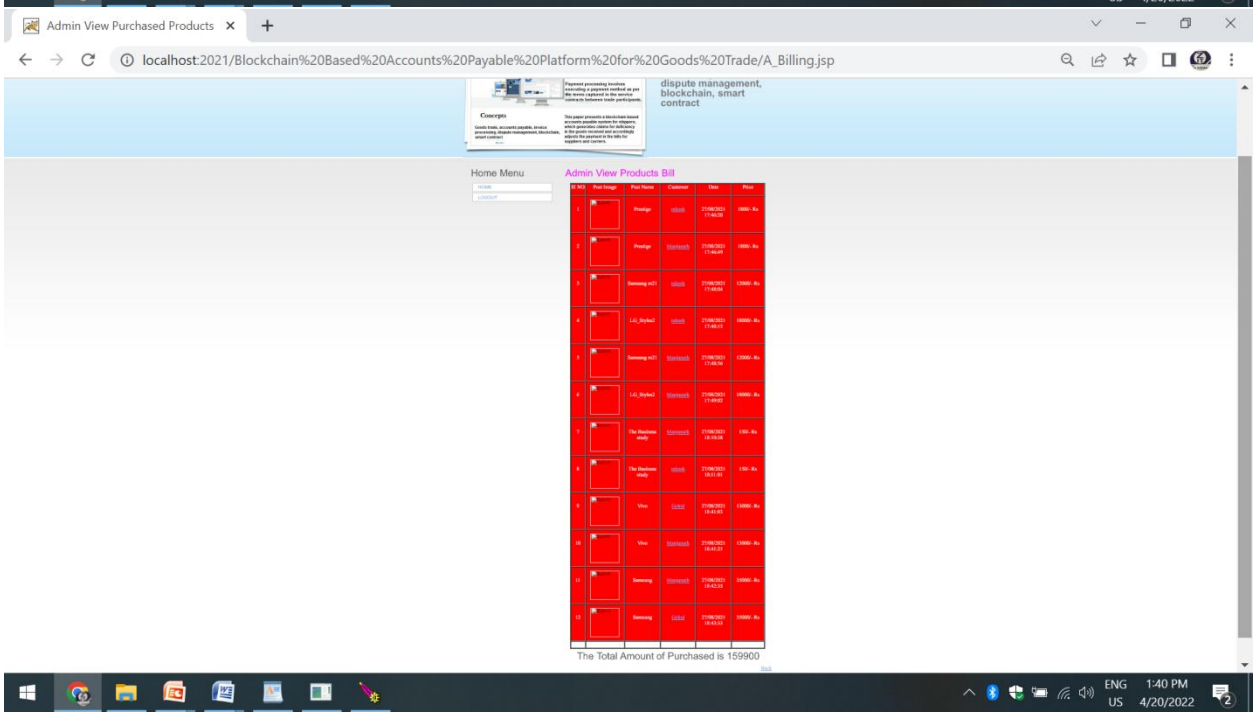

Admin Add Category
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/A_AddCategory.jsp
Read me first...
Payment processing involves executing a payment method as per the terms captured in the service contracts between trade participants.
Concepts
Goods trade, accounts payable, invoice processing, dispute management, blockchain, smart contract
This paper presents a blockchain-based accounts payable system for shippers, which generates claims for deficiency in the goods received and accordingly adjusts the payment in the bills for suppliers and carriers.
processing,
dispute management,
blockchain, smart
contract
Home Menu
HOME
LOGOUT
Admin Add Category
Categorie
laptop
Add
Back
ENG
US
1:36 PM
4/20/2022


Admin Add sub-Category
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/A_AddSubCategory.jsp
Read me first...
Payment processing involves executing a payment method as per the terms captured in the service contracts between trade participants.
Concepts
Goods trade, accounts payable, invoice processing, dispute management, blockchain, smart contract
This paper presents a blockchain-based accounts payable system for shippers, which generates claims for deficiency in the goods received and accordingly adjusts the payment in the bills for suppliers and carriers.
processing,
dispute management,
blockchain, smart
contract
Home Menu
HOME
LOGOUT
Admin Add Sub-Category
Categorie
Electronics
Sub-Category
Mobiles
Add
Back
ENG
US
1:36 PM
4/20/2022


Admin Add Mobile Posts
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/A_AddMobileProd.jsp
Home Menu
HOME
LOGOUT
Admin Add Sub-Category Status
Select the Categorie
Electronics
Select the Sub-Categorie
mobile
Mobile Name
Iphone 13pro
Price
120000
Desplay Size
6'
Color
silver
OS
ios
Internal Memory
256GB
External Memory
0
Mobile Processer
Icloud
Description
iPhone 13 Pro
ENG
US
1:38 PM
4/20/2022

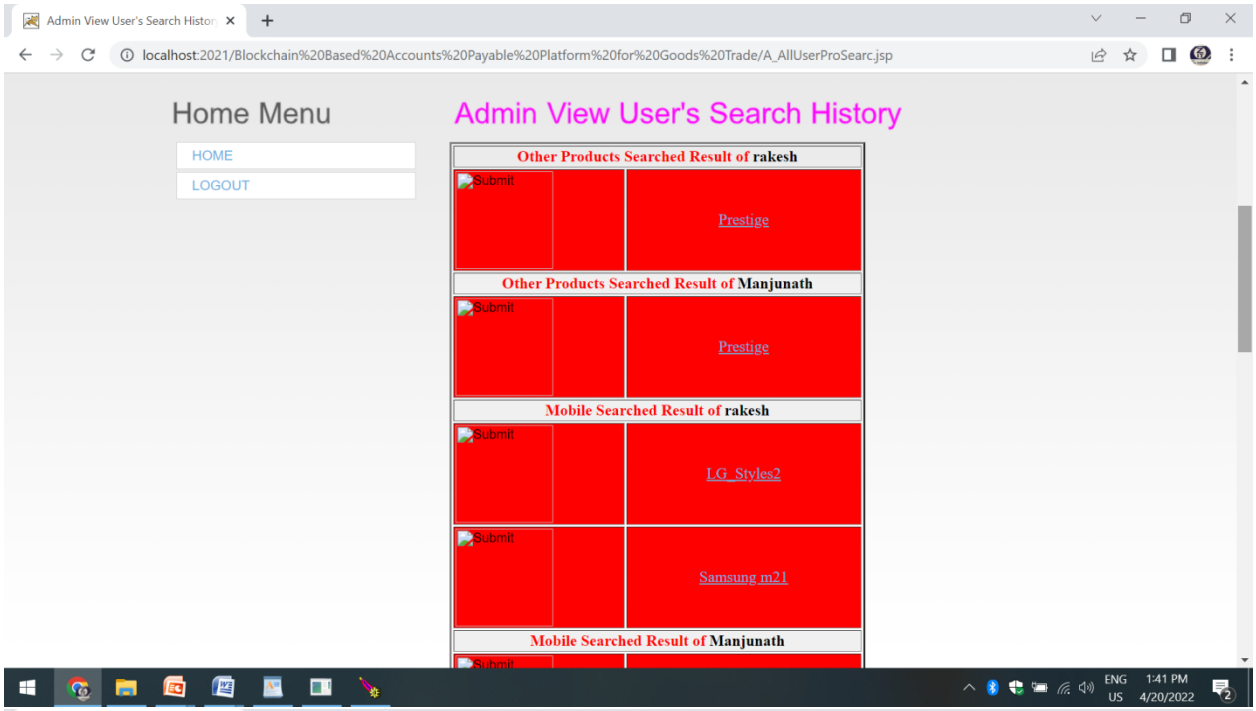

Admin View Mobile Posts
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/A_ViewAllMobPosts.jsp
Home Menu
HOME
LOGOUT
Admin View Mobile posts
SI NO
Post Image
Post Name
Categorie
Price
Rank
8
Submit
Samsung m21
Electronics
12000/- Rs
5
7
Submit
LG_Styles2
Electronics
18000/- Rs
4
9
Submit
Vivo
Electronics
13000/- Rs
2
10
Submit
Iphone 13pro
Electronics
120000/- Rs
0
ENG
US
1:38 PM
4/20/2022

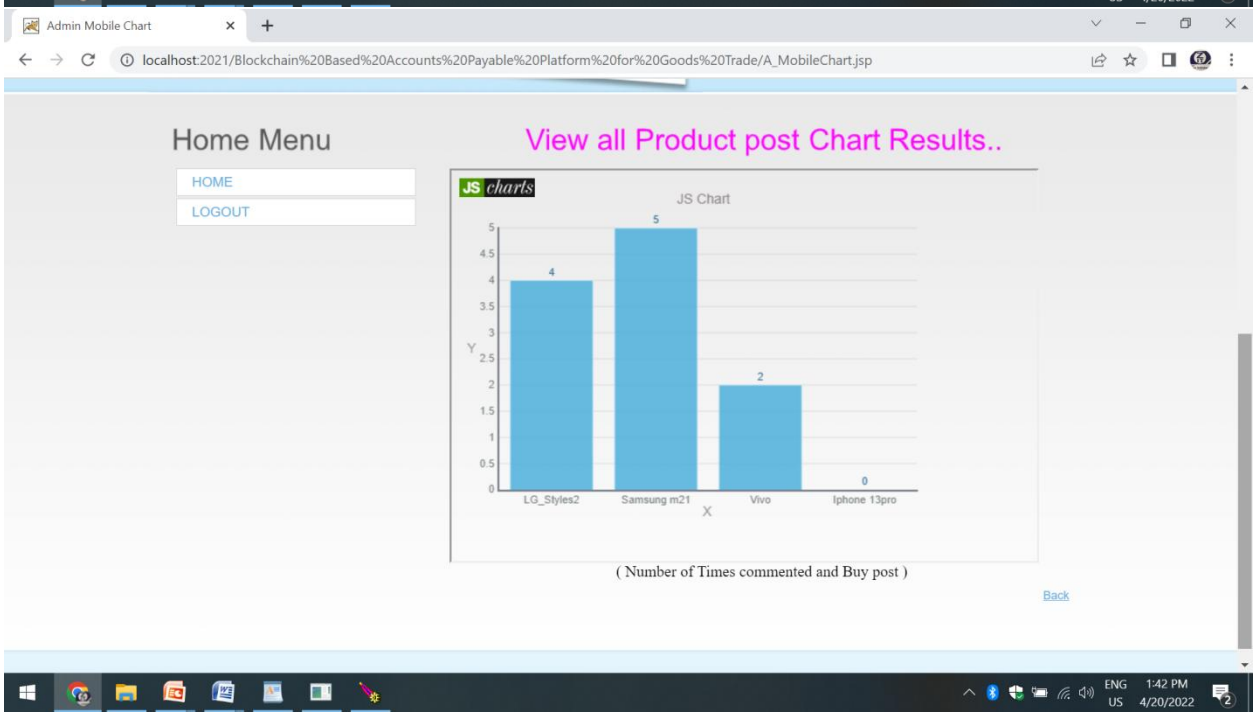

Admin View Other Posts
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/A_ViewAllOtherPosts.jsp
processing, dispute management, blockchain, smart contract
in the goods received and accordingly adjusts the payment in the bills for suppliers and carriers.
Home Menu
HOME
LOGOUT
Admin View Other posts
SI NO
Post Image
Post Name
Categorie
Price
Rank
6
Submit
Prestige
Home appliances
1800/- Rs
3
7
Submit
The Business study
Office
150/- Rs
3
8
Submit
Samsung
Electronics
35000/- Rs
2
Back
ENG US
1:39 PM
4/20/2022

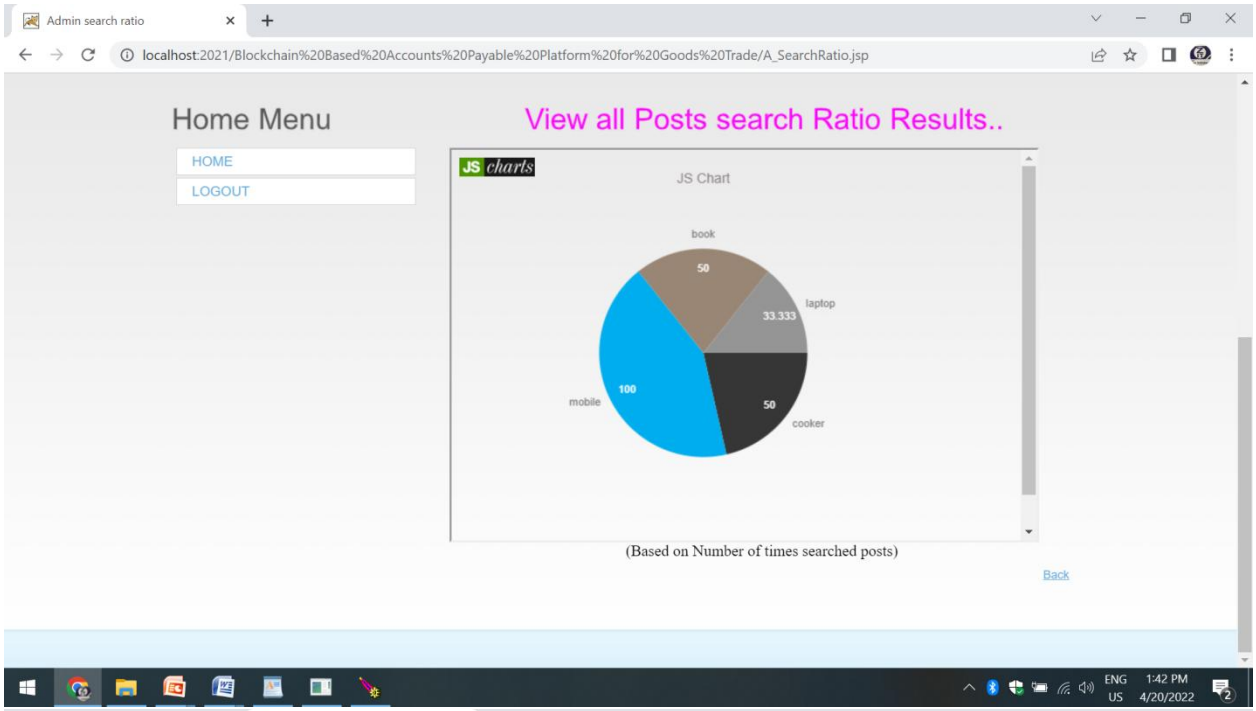

Admin View Purchased Products
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/A_Billing.jsp
dispute management, blockchain, smart contract
Home Menu
Admin View Products Bill
The Total Amount of Purchased is 159900
ENG US
1:40 PM
4/20/2022

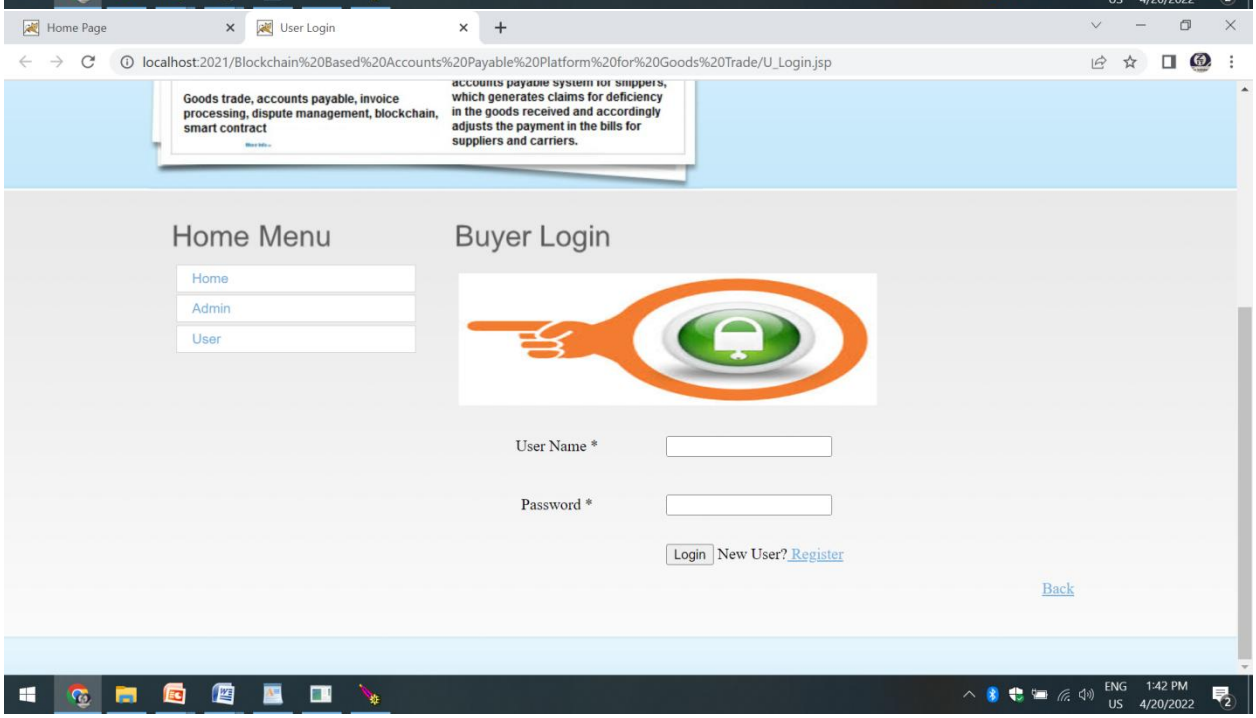

Admin Reviews Mobile Posts
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/A_ViewReviewsMob.jsp
Home Menu
HOME
LOGOUT
Admin Reviews on Mobile posts
SI NO
Post Image
Post Name
Categorie
Price
Comments
7
Submit
LG_Styles2
Electronics
18000/- Rs
Reviews
8
Submit
Samsung m21
Electronics
12000/- Rs
Reviews
9
Submit
Vivo
Electronics
13000/- Rs
Reviews
10
Submit
Iphone 13pro
Electronics
120000/- Rs
Reviews
Back
ENG
US
1:41 PM
4/20/2022

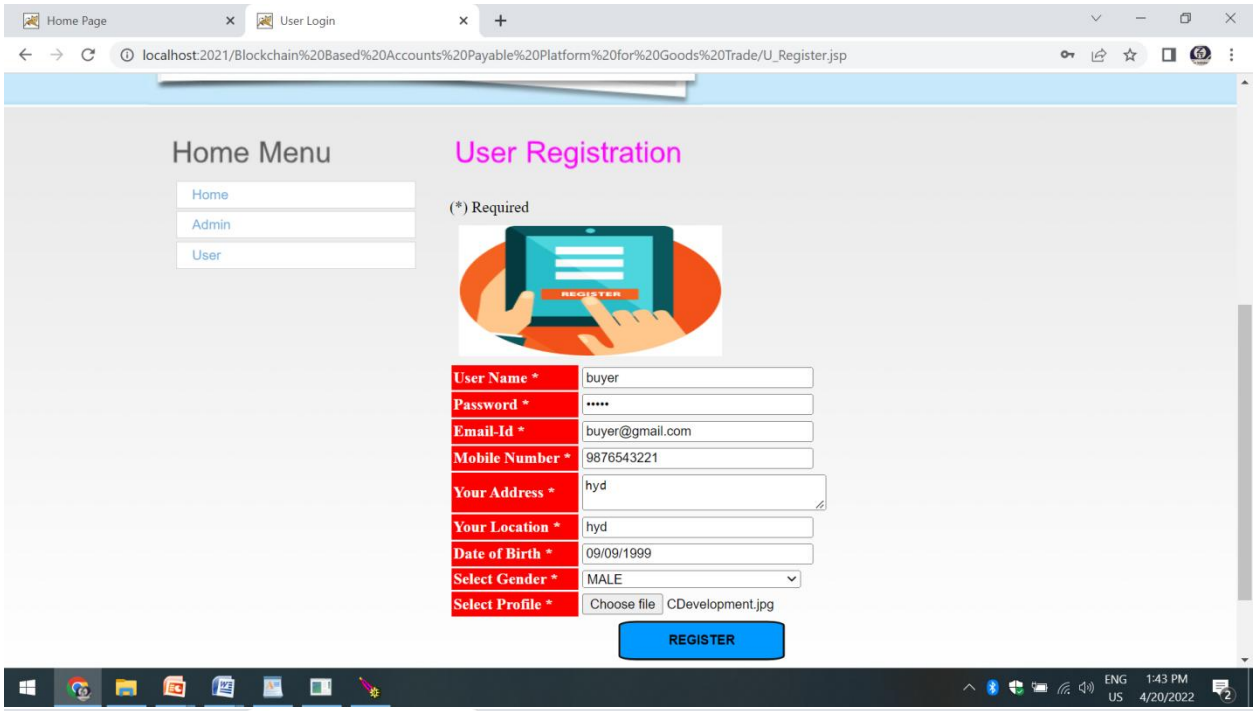
Admin View Reviews Products
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/MReviews.jsp?name=Samsung%20m21
Read me first...
Payment processing involves executing a payment method as per the terms captured in the service contracts between trade participants.
Concepts
Goods trade, accounts payable, invoice processing, dispute management, blockchain, smart contract
This paper presents a blockchain-based accounts payable system for shippers, which generates claims for deficiency in the goods received and accordingly adjusts the payment in the bills for suppliers and carriers.
processing,
dispute management,
blockchain, smart
contract.
Home Menu
HOME
LOGOUT
Admin View Mobile Samsung m21
SI NO
Post Image
Post Name
Comment
Commented By
Date
1
Submit
Samsung m21
It is good product to buy
rakesh
27/08/2021 18:12:22
Back
ENG
US
1:41 PM
4/20/2022

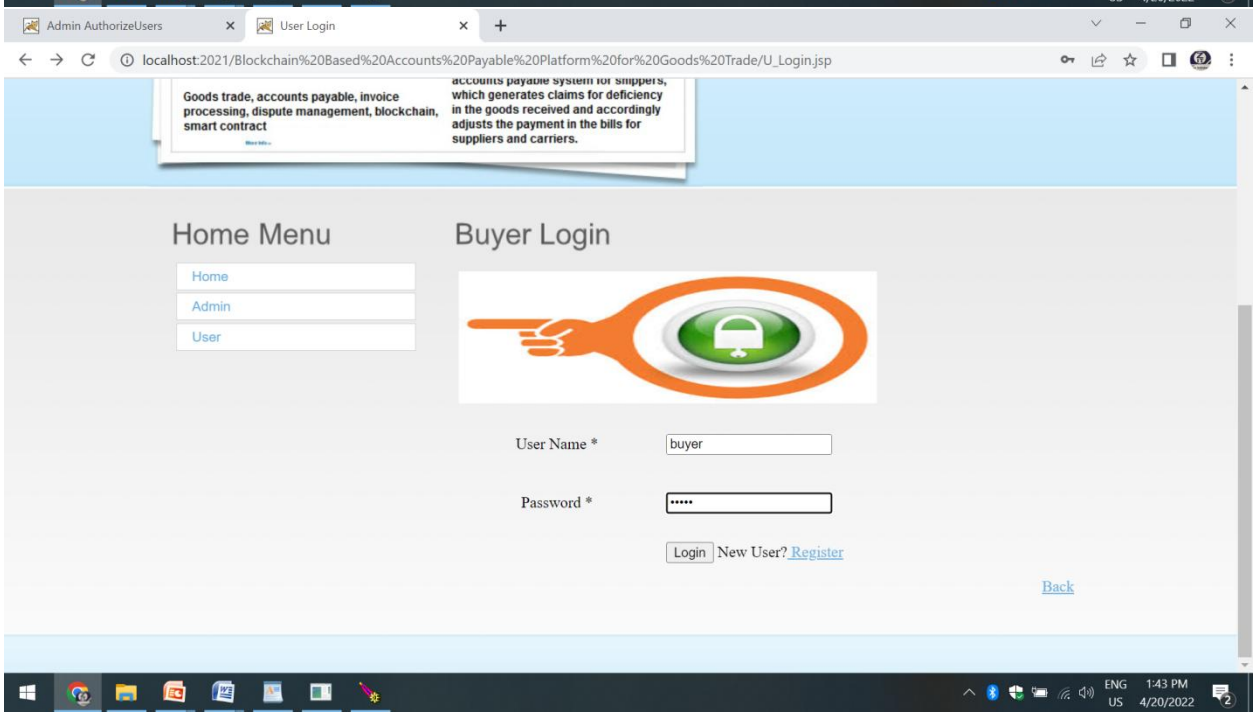

Admin View User's Search Histor
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/A_AllUserProSearc.jsp
Home Menu
HOME
LOGOUT
Admin View User's Search History
Other Products Searched Result of rakesh
Submit
Prestige
Other Products Searched Result of Manjunath
Submit
Prestige
Mobile Searched Result of rakesh
Submit
LG_Styles2
Submit
Samsung m21
Mobile Searched Result of Manjunath
Submit
ENG
US
1:41 PM
4/20/2022


Admin Mobile Chart
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/A_MobileChart.jsp
Home Menu
HOME
LOGOUT
View all Product post Chart Results..
JS charts
JS Chart
5
4
2
0
Y
X
LG_Styles2
Samsung m21
Vivo
Iphone 13pro
(Number of Times commented and Buy post)
Back
ENG
US
1:42 PM
4/20/2022


Admin search ratio
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/A_SearchRatio.jsp
Home Menu
HOME
LOGOUT
View all Posts search Ratio Results..
JS charts
JS Chart
book
50
laptop
33.333
mobile
100
50
cooker
(Based on Number of times searched posts)
Back

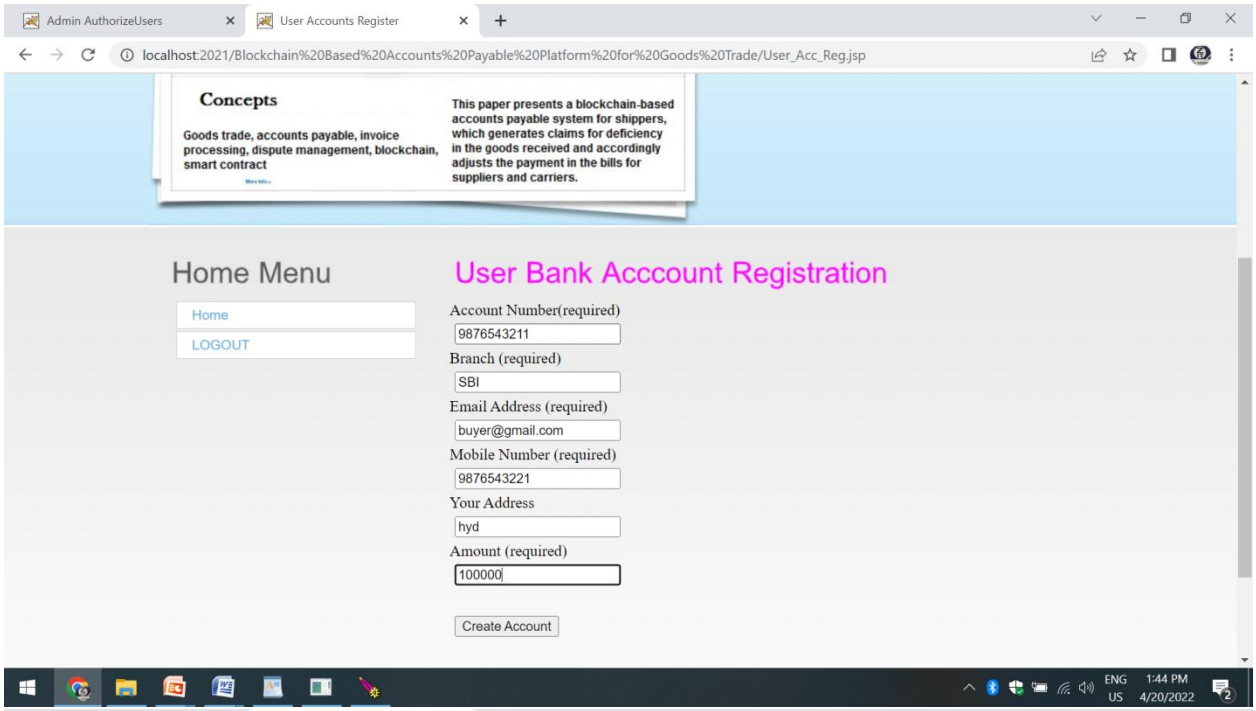

Home Page
User Login
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/U_Login.jsp
Goods trade, accounts payable, invoice processing, dispute management, blockchain, smart contract
which generates claims for deficiency in the goods received and accordingly adjusts the payment in the bills for suppliers and carriers.
Home Menu
Home
Admin
User
Buyer Login
User Name *
Password *
Login
New User? Register
Back

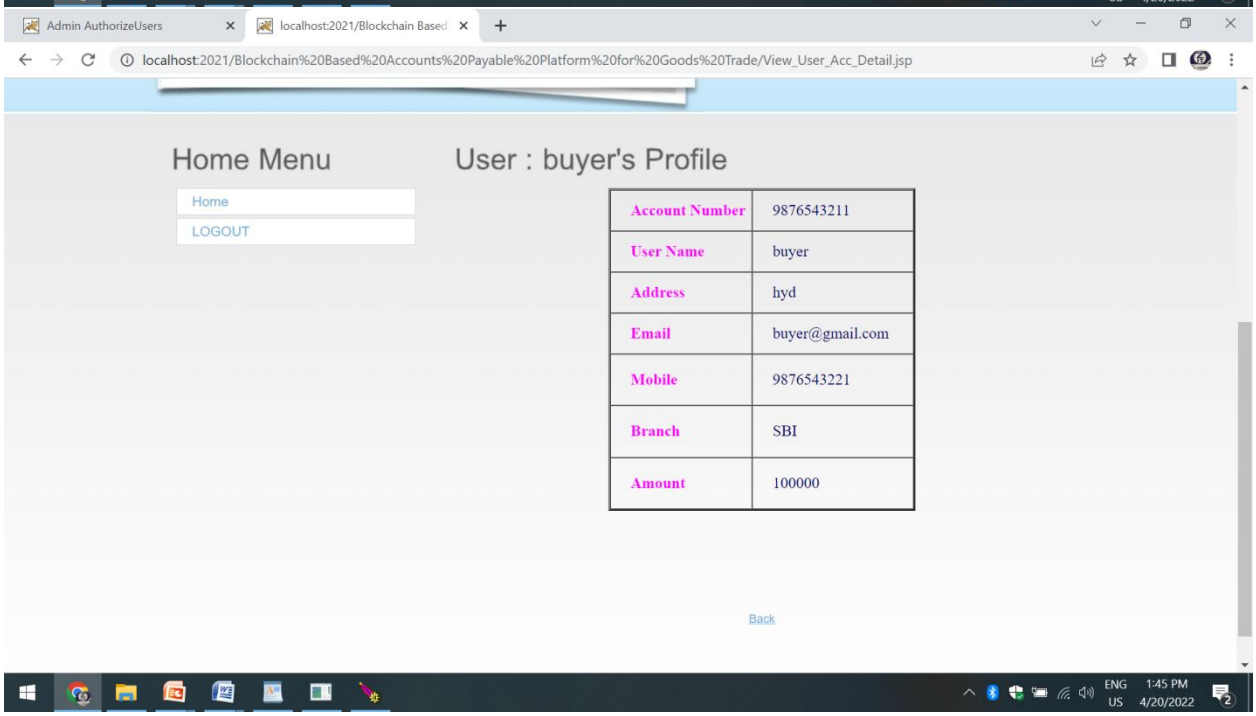

Home Page
User Login
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/U_Register.jsp
Home Menu
Home
Admin
User
User Registration
(*) Required
REGISTER
User Name * buyer
Password * •••••
Email-Id * buyer@gmail.com
Mobile Number * 9876543221
Your Address * hyd
Your Location * hyd
Date of Birth * 09/09/1999
Select Gender * MALE
Select Profile * Choose file CDevelopment.jpg
REGISTER
ENG US 1:43 PM 4/20/2022


Admin AuthorizeUsers
User Login
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/U_Login.jsp
Goods trade, accounts payable, invoice processing, dispute management, blockchain, smart contract
which generates claims for deficiency in the goods received and accordingly adjusts the payment in the bills for suppliers and carriers.
Home Menu
Home
Admin
User
Buyer Login
User Name * buyer
Password * •••••
Login New User? Register
Back
ENG US 1:43 PM 4/20/2022

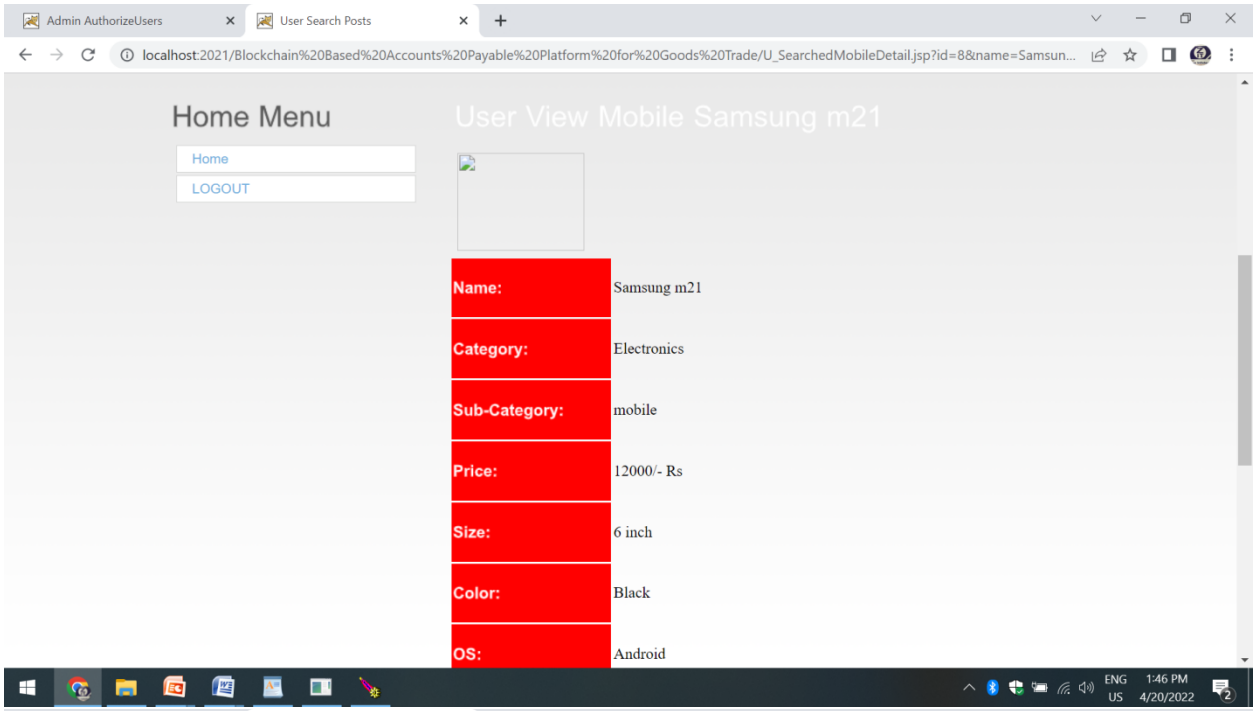

localhost
User Main
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/U_Main.jsp
Goods trade, accounts payable, invoice processing, dispute management, blockchain, smart contract
which generates claims for deficiency in the goods received and accordingly adjusts the payment in the bills for suppliers and carriers.
Home Menu
WelCome Buyer buyer
My Profile
Account Management
Search Mobile Products
Search Other Products
View My Search History
View All Products
View Top K Mobile Products
View Top K Other Products
LogOut
ENG IN 2:53 PM 4/20/2022

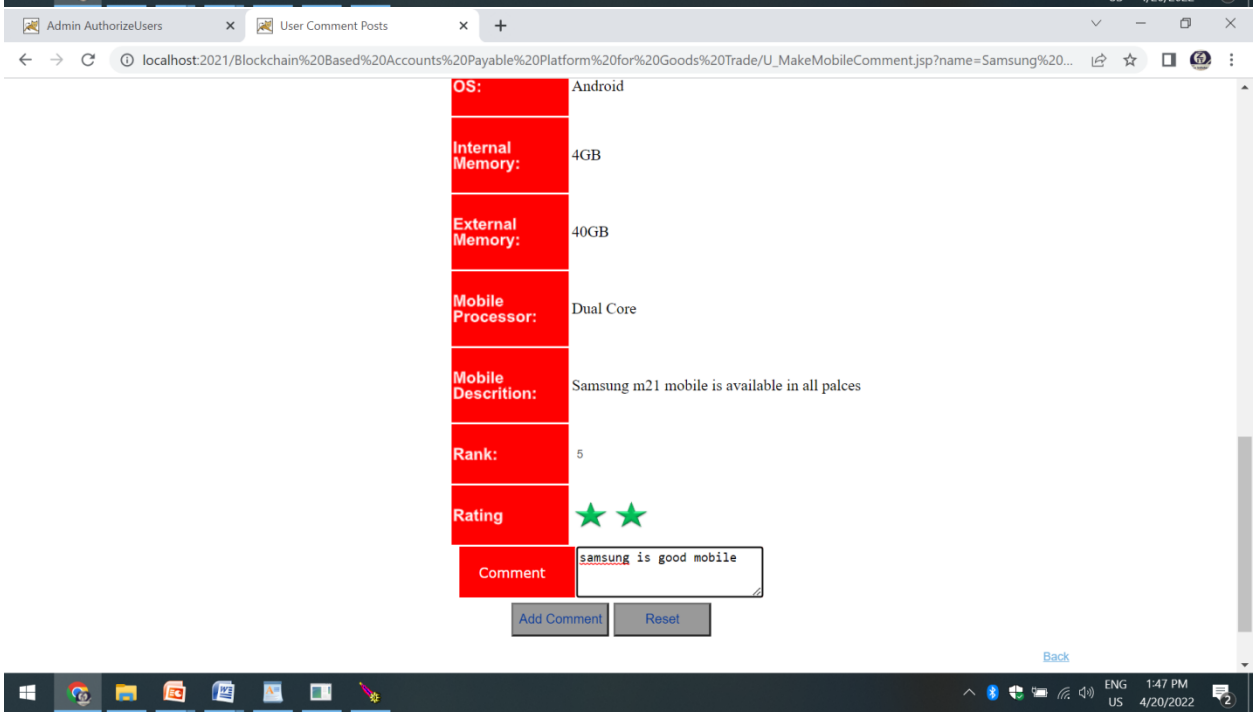
Admin AuthorizeUsers
User Account Management
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/Account_Management.jsp
for Goods Trade
Read me first...
Payment processing involves executing a payment method as per the terms captured in the service contracts between trade participants.
Concepts
Goods trade, accounts payable, invoice processing, dispute management, blockchain, smart contract
This paper presents a blockchain-based accounts payable system for shippers, which generates claims for deficiency in the goods received and accordingly adjusts the payment in the bills for suppliers and carriers.
payable, invoice processing, dispute management, blockchain, smart contract
Home Menu
Home
LOGOUT
WELCOME TO ACCOUNT MANAGMENT mr/ms.buyer
Create Account
Add Money
View Account
Ministatement
Back
localhost:2021/.../User_Acc_Reg.jsp
ENG US 1:44 PM 4/20/2022

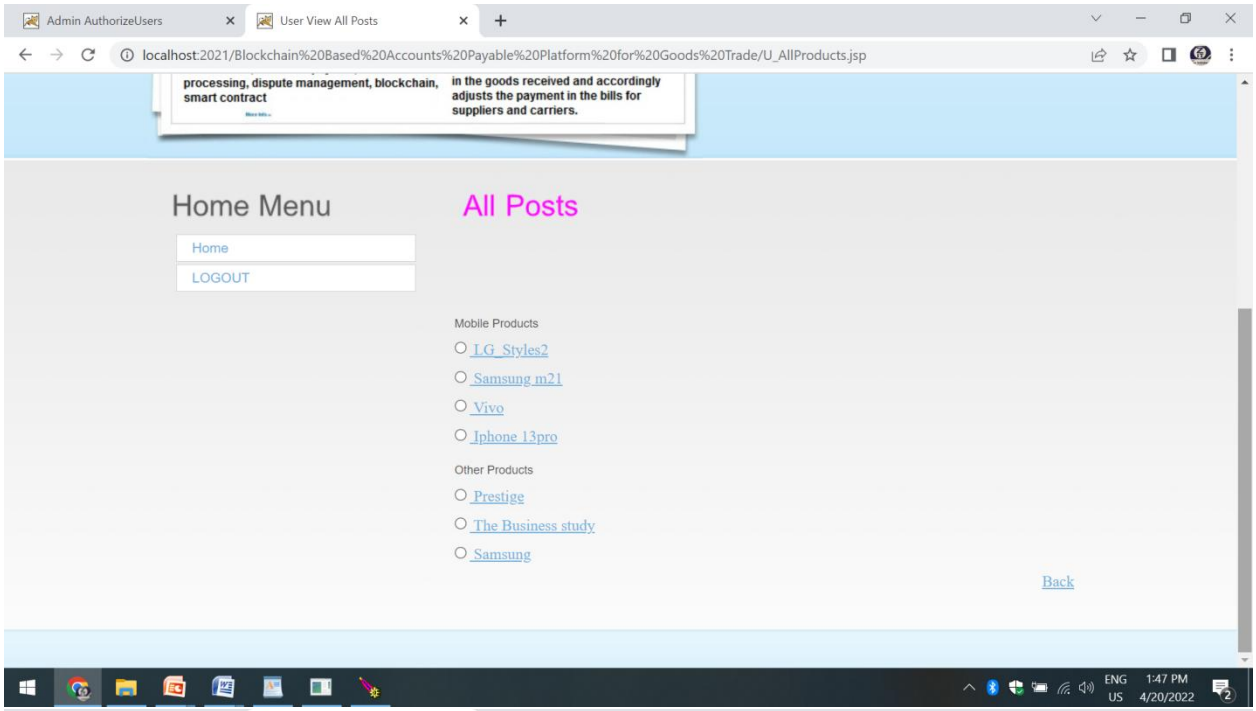

Admin AuthorizeUsers
User Accounts Register
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/User_Acc_Reg.jsp
Concepts
Goods trade, accounts payable, invoice processing, dispute management, blockchain, smart contract
This paper presents a blockchain-based accounts payable system for shippers, which generates claims for deficiency in the goods received and accordingly adjusts the payment in the bills for suppliers and carriers.
Home Menu
Home
LOGOUT
User Bank Acccount Registration
Account Number(required)
9876543211
Branch (required)
SBI
Email Address (required)
buyer@gmail.com
Mobile Number (required)
9876543221
Your Address
hyd
Amount (required)
100000
Create Account
ENG US
1:44 PM
4/20/2022

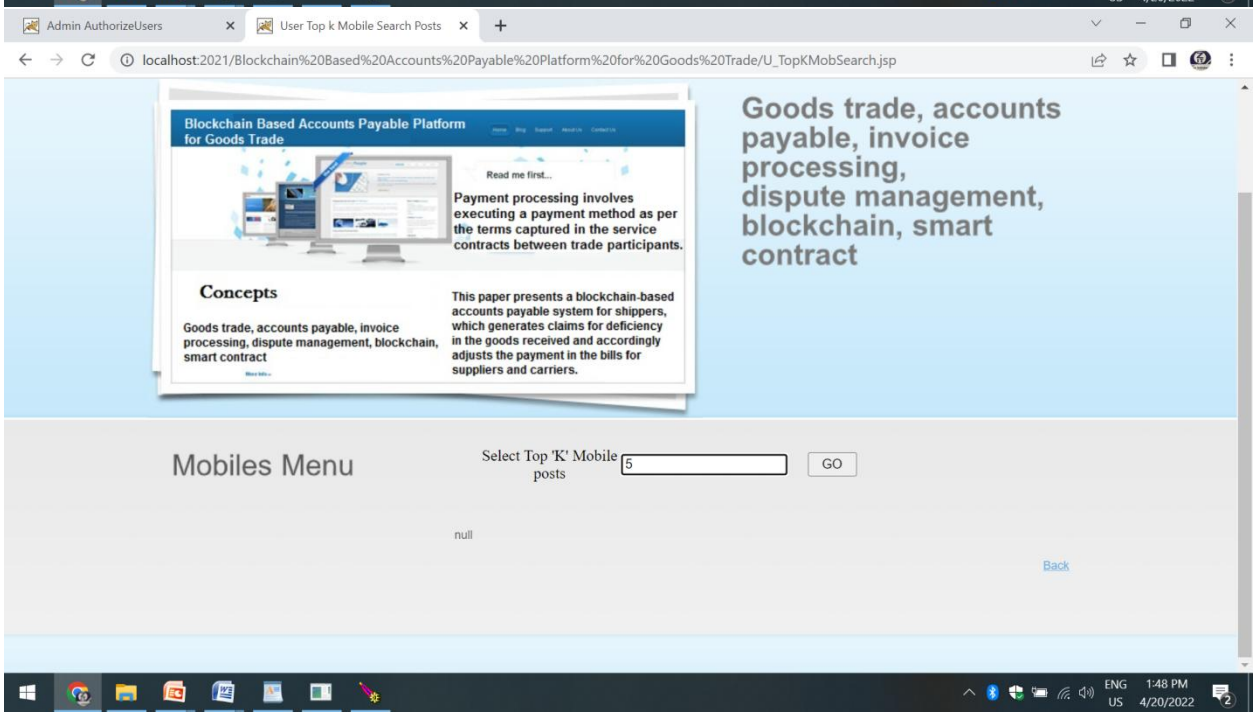

Admin AuthorizeUsers
localhost:2021/Blockchain Based
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/View_User_Acc_Detail.jsp
Home Menu
Home
LOGOUT
User : buyer's Profile
Account Number 9876543211
User Name buyer
Address hyd
Email buyer@gmail.com
Mobile 9876543221
Branch SBI
Amount 100000
Back
ENG US
1:45 PM
4/20/2022

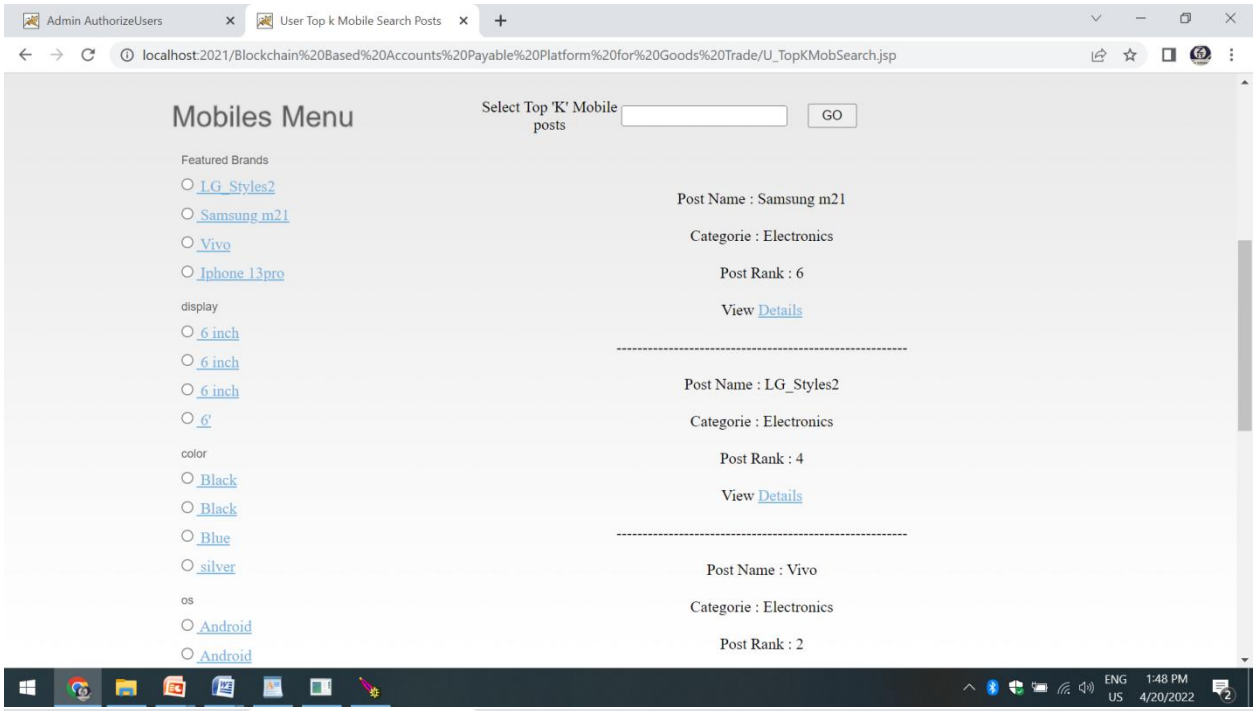

Admin AuthorizeUsers
localhost:2021/Blockchain Based
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/U_SearchMobile.jsp
Read me first...
Payment processing involves executing a payment method as per the terms captured in the service contracts between trade participants.
Concepts
Goods trade, accounts payable, invoice processing, dispute management, blockchain, smart contract
This paper presents a blockchain-based accounts payable system for shippers, which generates claims for deficiency in the goods received and accordingly adjusts the payment in the bills for suppliers and carriers.
processing, dispute management, blockchain, smart contract.
Mobile Menu
Search Mobile Posts
Enter Fuzzy Keyword
samsung
GO
(searching content Based on Post Description)
Back
Admin AuthorizeUsers
localhost:2021/Blockchain Based
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/U_SearchMobile.jsp
Mobile Menu
Search Mobile Posts
Featured Brands
LG_Styles2
Vivo
Iphone 13pro
display
6 inch
6 inch
6'
color
Black
Blue
silver
os
Android
Android
ios
internaml memory
Enter Fuzzy Keyword
GO
(searching content Based on Post Description)
Submit
Mobile Name : Samsung m21
Mobile Rank : 5
View Details
search ratio=1:4
Back

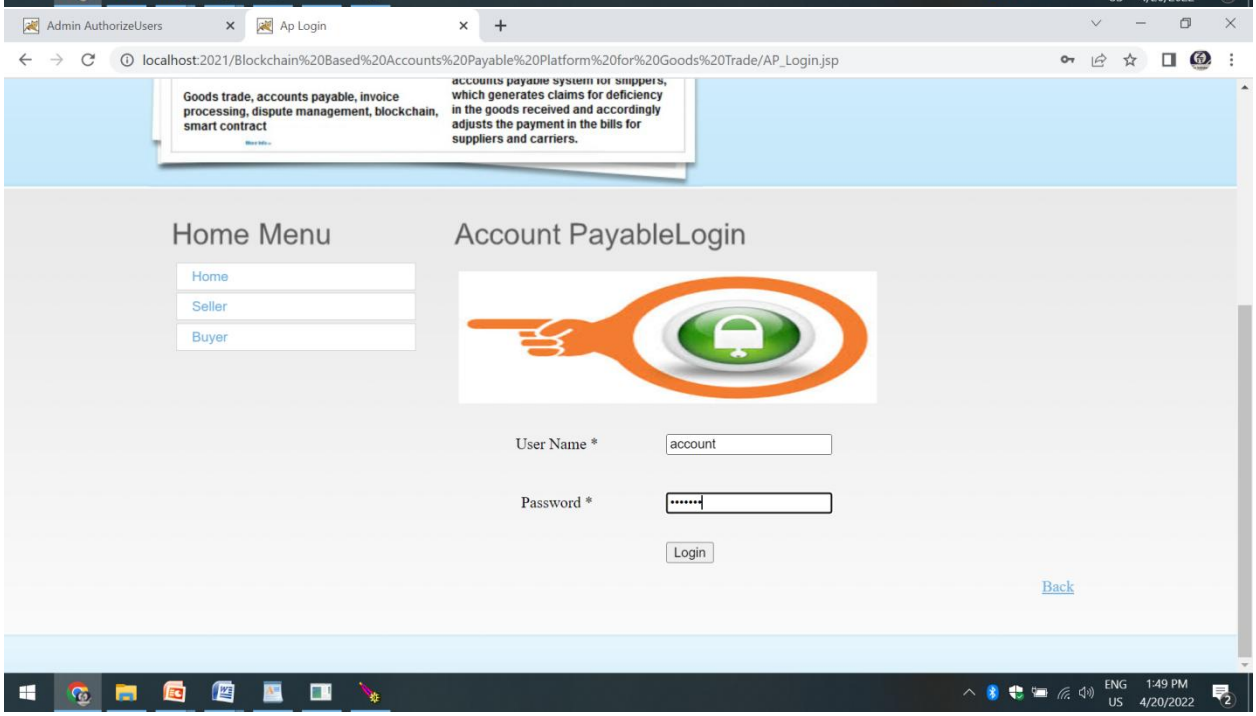

Admin AuthorizeUsers
User Search Posts
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/U_SearchedMobileDetail.jsp?id=8&name=Samsun...
Home Menu
Home
LOGOUT
User View Mobile Samsung m21
Name: Samsung m21
Category: Electronics
Sub-Category: mobile
Price: 12000/- Rs
Size: 6 inch
Color: Black
OS: Android
ENG US 1:46 PM 4/20/2022

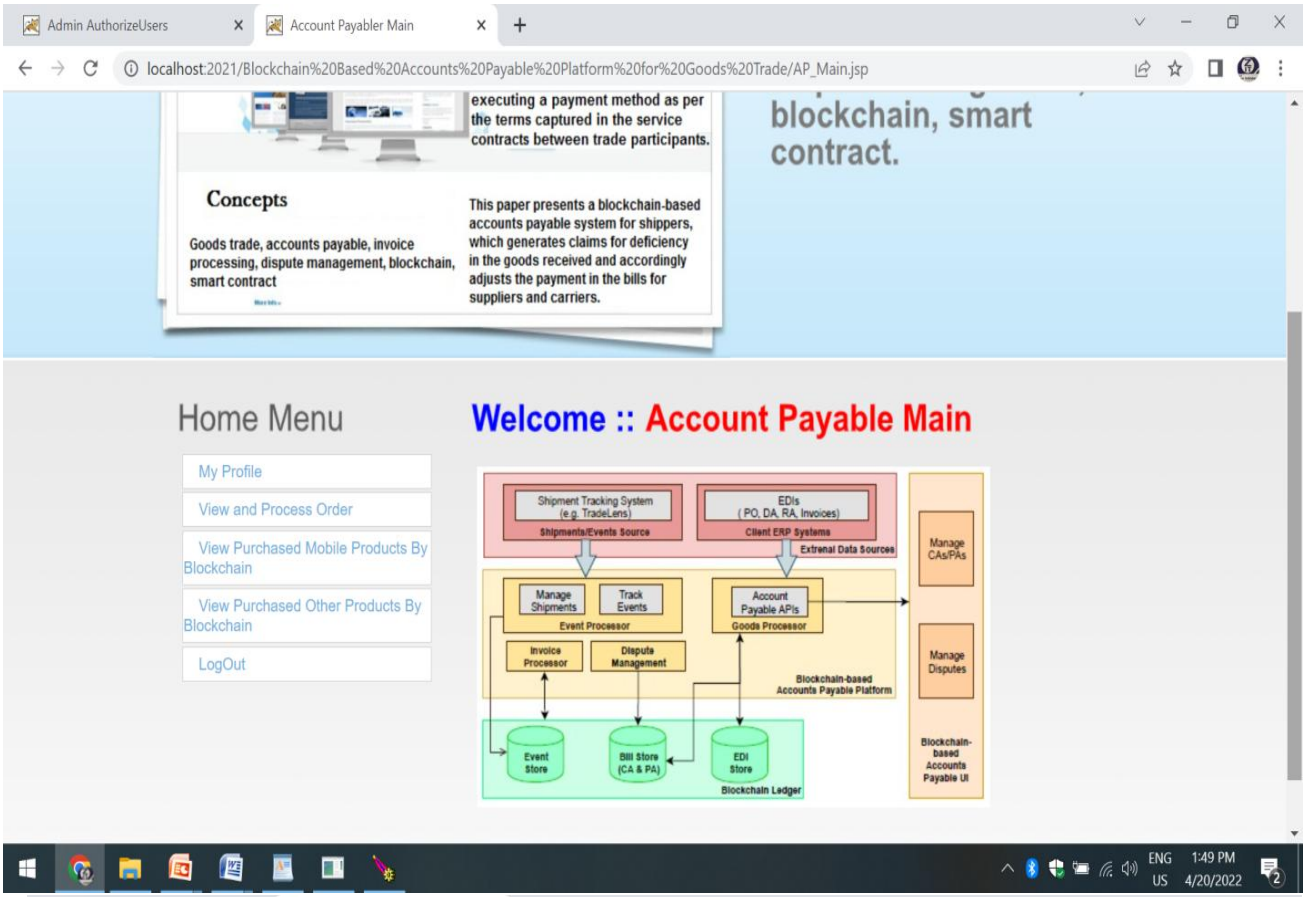

Admin AuthorizeUsers
User Comment Posts
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/U_MakeMobileComment.jsp?name=Samsung%20...
OS: Android
Internal Memory: 4GB
External Memory: 40GB
Mobile Processor: Dual Core
Mobile Descrition: Samsung m21 mobile is available in all palces
Rank: 5
Rating
Comment
samsung is good mobile
Add Comment
Reset
Back
ENG US 1:47 PM 4/20/2022

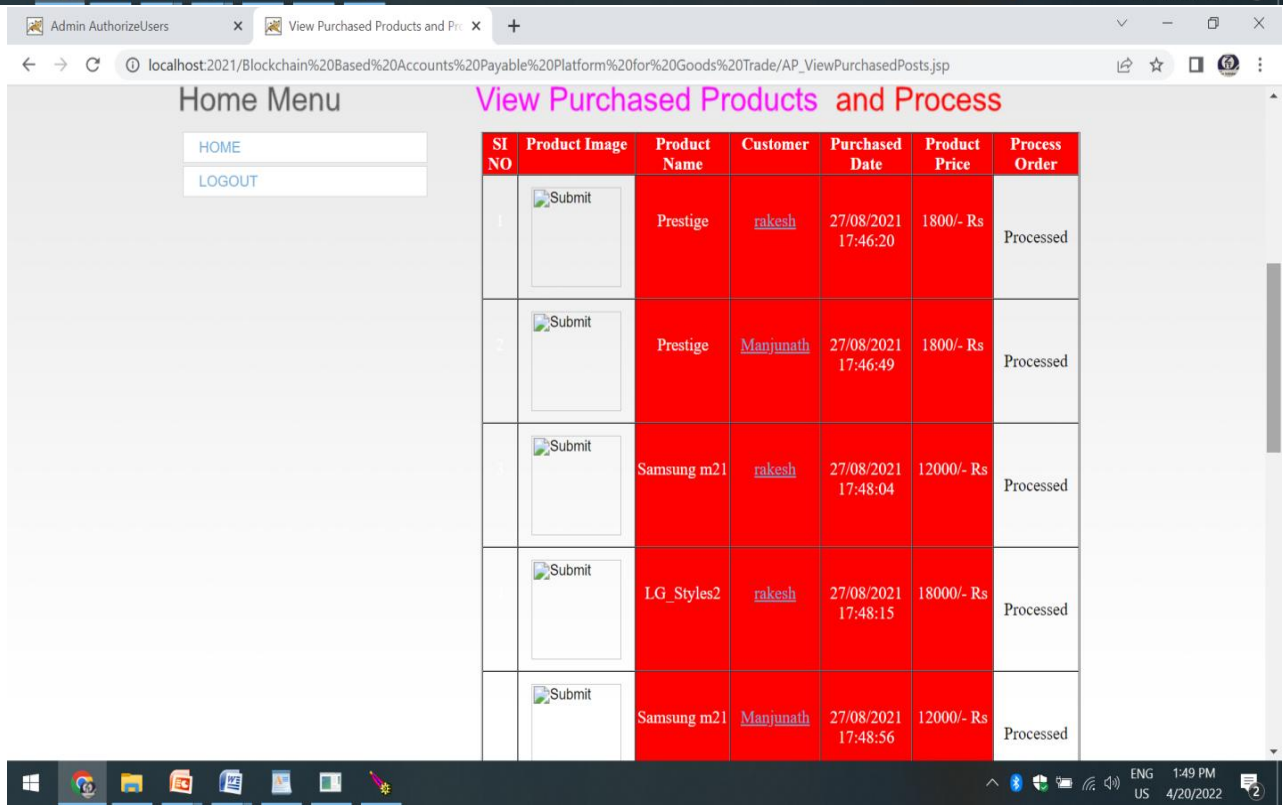

Admin AuthorizeUsers
User View All Posts
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/U_AllProducts.jsp
processing, dispute management, blockchain, smart contract
in the goods received and accordingly adjusts the payment in the bills for suppliers and carriers.
Home Menu
All Posts
Home
LOGOUT
Mobile Products
LG_Styles2
Samsung m21
Vivo
Iphone 13pro
Other Products
Prestige
The Business study
Samsung
Back
ENG US
1:47 PM
4/20/2022

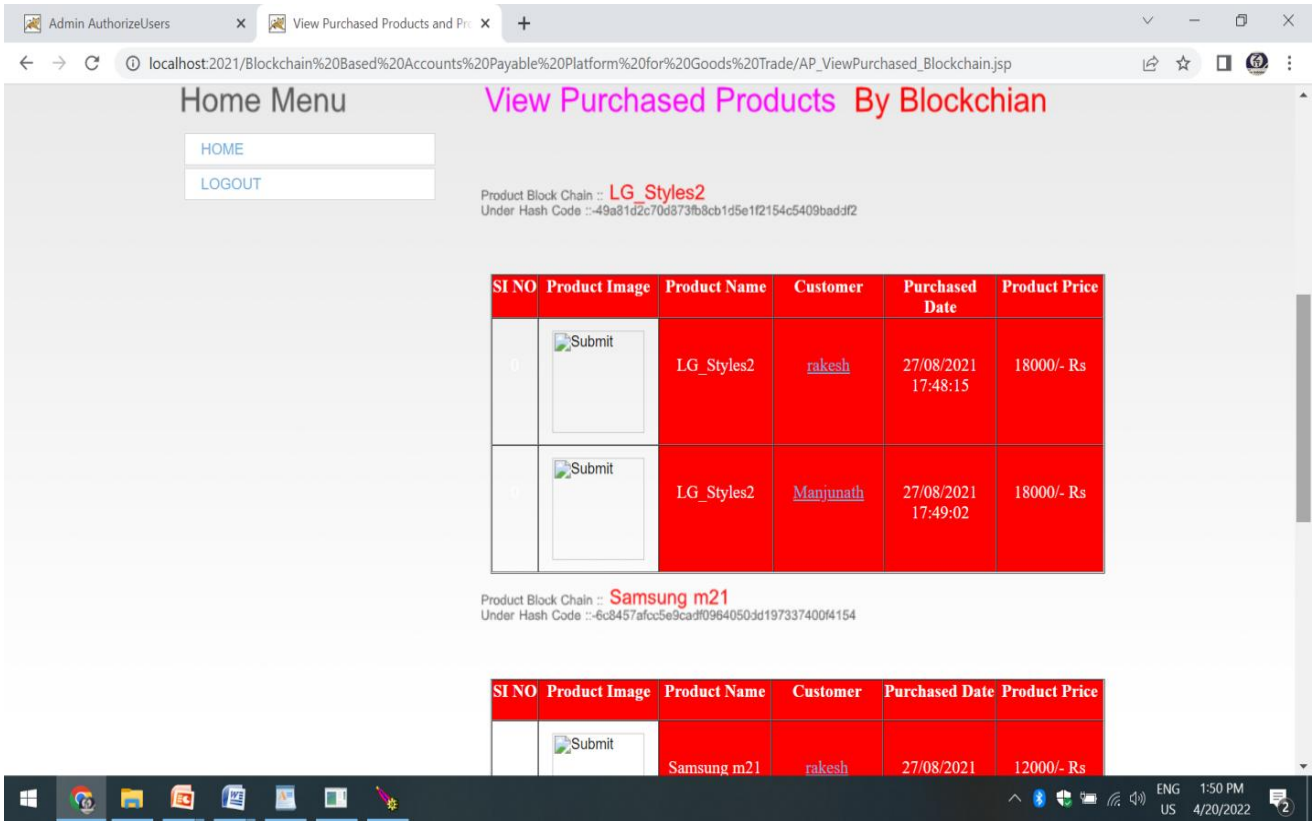

Admin AuthorizeUsers
User Top k Mobile Search Posts
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/U_TopKMobSearch.jsp
Blockchain Based Accounts Payable Platform for Goods Trade
Read me first...
Payment processing involves executing a payment method as per the terms captured in the service contracts between trade participants.
Concepts
Goods trade, accounts payable, invoice processing, dispute management, blockchain, smart contract
This paper presents a blockchain-based accounts payable system for shippers, which generates claims for deficiency in the goods received and accordingly adjusts the payment in the bills for suppliers and carriers.
Goods trade, accounts payable, invoice processing, dispute management, blockchain, smart contract
Mobiles Menu
Select Top 'K' Mobile posts
5
GO
null
Back
ENG US
1:48 PM
4/20/2022

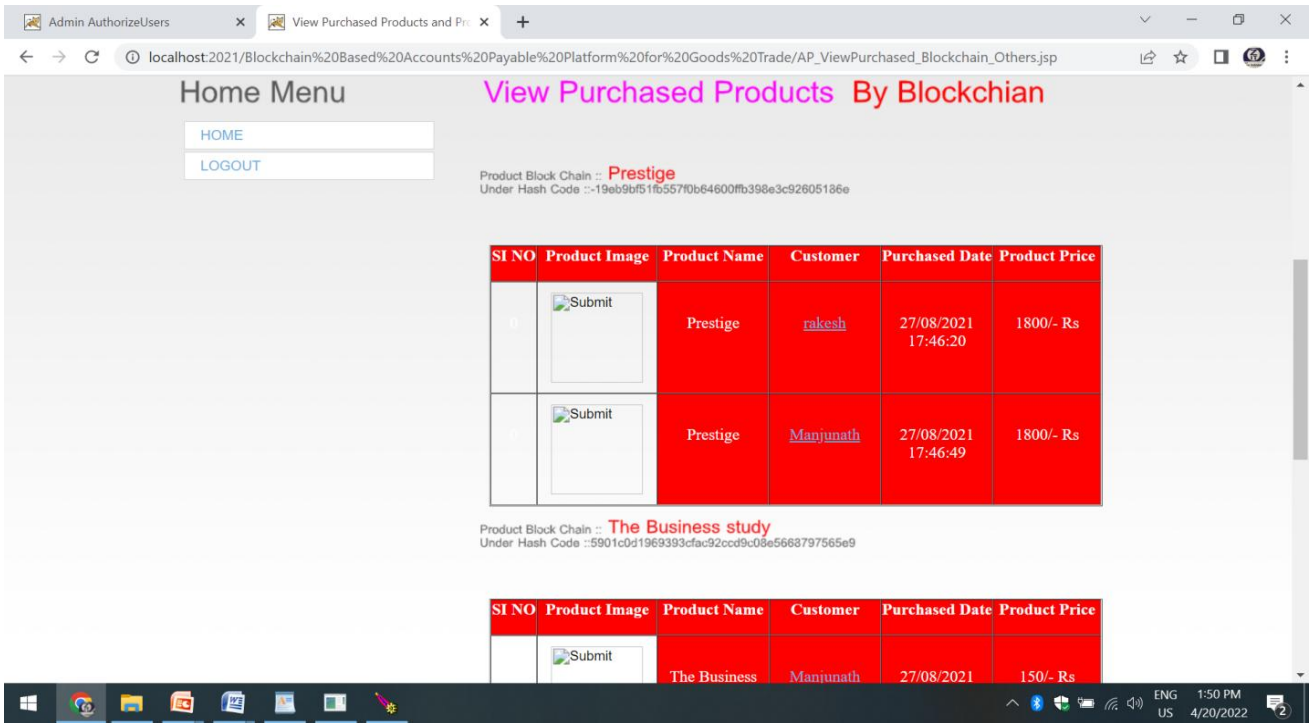

Admin AuthorizeUsers
User Top k Mobile Search Posts
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/U_TopKMobSearch.jsp
Mobiles Menu
Select Top 'K' Mobile posts
GO
Featured Brands
LG_Styles2
Samsung m21
Vivo
Iphone 13pro
display
6 inch
6 inch
6 inch
6'
color
Black
Black
Blue
silver
os
Android
Android
Post Name : Samsung m21
Categorie : Electronics
Post Rank : 6
View Details
Post Name : LG_Styles2
Categorie : Electronics
Post Rank : 4
View Details
Post Name : Vivo
Categorie : Electronics
Post Rank : 2
ENG US 1:48 PM 4/20/2022


Admin AuthorizeUsers
Ap Login
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/AP_Login.jsp
Goods trade, accounts payable, invoice processing, dispute management, blockchain, smart contract
which generates claims for deficiency in the goods received and accordingly adjusts the payment in the bills for suppliers and carriers.
Home Menu
Home
Seller
Buyer
Account PayableLogin
User Name *
account
Password *
Login
Back
ENG US 1:49 PM 4/20/2022


Admin AuthorizeUsers
Account Payabler Main
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/AP_Main.jsp
executing a payment method as per the terms captured in the service contracts between trade participants.
blockchain, smart contract.
Concepts
Goods trade, accounts payable, invoice processing, dispute management, blockchain, smart contract
This paper presents a blockchain-based accounts payable system for shippers, which generates claims for deficiency in the goods received and accordingly adjusts the payment in the bills for suppliers and carriers.
Home Menu
My Profile
View and Process Order
View Purchased Mobile Products By Blockchain
View Purchased Other Products By Blockchain
LogOut
Welcome :: Account Payable Main
Shipment Tracking System (e.g. TradeLens)
Shipments/Events Source
EDIs (PO, DA, RA, Invoices)
Client ERP Systems
External Data Sources
Manage Shipments
Track Events
Event Processor
Account Payable APIs
Goods Processor
Invoice Processor
Dispute Management
Blockchain-based Accounts Payable Platform
Manage CAs/PAs
Manage Disputes
Blockchain-based Accounts Payable UI
Event Store
Bill Store (CA & PA)
EDI Store
Blockchain Ledger
ENG US
1:49 PM
4/20/2022


Admin AuthorizeUsers
View Purchased Products and Pr
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/AP_ViewPurchasedPosts.jsp
Home Menu
HOME
LOGOUT
View Purchased Products and Process
SI NO
Product Image
Product Name
Customer
Purchased Date
Product Price
Process Order
Submit
Prestige
rakesh
27/08/2021 17:46:20
1800/- Rs
Processed
Submit
Prestige
Manjunath
27/08/2021 17:46:49
1800/- Rs
Processed
Submit
Samsung m21
rakesh
27/08/2021 17:48:04
12000/- Rs
Processed
Submit
LG_Styles2
rakesh
27/08/2021 17:48:15
18000/- Rs
Processed
Submit
Samsung m21
Manjunath
27/08/2021 17:48:56
12000/- Rs
Processed
ENG US
1:49 PM
4/20/2022


Admin AuthorizeUsers
View Purchased Products and Pr
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/AP_ViewPurchased_Blockchain.jsp
Home Menu
HOME
LOGOUT
View Purchased Products By Blockchian
Product Block Chain :: LG_Styles2
Under Hash Code ::-49a81d2c70d373fb8cb1d5e1f2154c5409badcf2
SI NO
Product Image
Product Name
Customer
Purchased Date
Product Price
Submit
LG_Styles2
rakesh
27/08/2021 17:48:15
18000/- Rs
Submit
LG_Styles2
Manjunath
27/08/2021 17:49:02
18000/- Rs
Product Block Chain :: Samsung m21
Under Hash Code ::-6c8457afcc5e9cadf0964050dd197337400f4154
SI NO
Product Image
Product Name
Customer
Purchased Date
Product Price
Submit
Samsung m21
rakesh
27/08/2021
12000/- Rs
ENG US
1:50 PM
4/20/2022


Admin AuthorizeUsers
View Purchased Products and Pr
localhost:2021/Blockchain%20Based%20Accounts%20Payable%20Platform%20for%20Goods%20Trade/AP_ViewPurchased_Blockchain_Others.jsp
Home Menu
HOME
LOGOUT
View Purchased Products By Blockchian
Product Block Chain :: Prestige
Under Hash Code ::-19eb9bf51fb557f0b64600ffb398e3c92605186e
SI NO
Product Image
Product Name
Customer
Purchased Date
Product Price
Submit
Prestige
rakesh
27/08/2021 17:46:20
1800/- Rs
Submit
Prestige
Manjunath
27/08/2021 17:46:49
1800/- Rs
Product Block Chain :: The Business study
Under Hash Code ::5901c0d1969393cfac92ccd9c08e5668797565e9
SI NO
Product Image
Product Name
Customer
Purchased Date
Product Price
Submit
The Business
Manjunath
27/08/2021
150/- Rs
ENG US
1:50 PM
4/20/2022

6. TESTING STRATEGY

SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the

Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

TYPES OF TESTS

6.1 Unit Testing:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

6.2 Integration Testing

Integration testing addresses the issues associated with the dual problems of verification and program construction. After the software has been integrated a set of high order tests are conducted. The main objective in this testing process is to take unit tested modules and builds a program structure that has been dictated by design.

The following are the types of Integration Testing:

1. Top Down Integration

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

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

2. Bottom-up Integration

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

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

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

Functional test

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

Functional testing is centered on the following items:

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

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

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

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

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

System Testing

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

White Box Testing

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

Black Box Testing

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

6.3 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

OTHER TESTING METHODOLOGIES

User Acceptance Testing

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

Output Testing

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

Validation Checking

Validation checks are performed on the following fields.

Text Field:

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

Numeric Field:

The numeric field can contain only numbers from 0 to 9. An entry of any character flashes an error messages. The individual modules are checked for accuracy and what it has to perform. Each module is subjected to test run along with sample data. The individually tested modules are integrated into a single system

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

USER TRAINING

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

MAINTAINENCE

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

CONCLUSION

In this paper, we discussed the need for a block chain based accounts payable system that eliminates the process redundancies (accounts payable vs. accounts receivable), enables efficient invoice processing, and reduces the amount of time spent in reconciling disputes between the transacting participants in goods trade (domestic and global). We provided details of our block chain-based accounts payable system supporting trusted invoice processing and transparent dispute resolution. The claims against a PO are generated under four different categories and aggregated to produce the claim advice. The payment advices for supplier and carriers are generated by using the invoices for supplier and carriers respectively together with the claim advice generated by our accounts payable system and real time events from Trade Lens. The computed CA and PAs go through the reconciliation process before the payment gets processed by the shipper. We have realized the system proposed in this paper using Hyper ledger Fabric as the underlying block chain platform (however any other permissioned block chain platform can also be used here) and a cloud micro services architecture. We showcased the performance of different smart contract modules (i.e., transactions) supported by our system using a representative goods trade ecosystem. The results show that its practical to deploy one such system in real-world customer environments. We are currently experimenting with the use of the proposed system in collaboration with the participants of the Trade Lens platform.

REFERENCES

[1] “Letter of credit,” https://en.wikipedia.org/wiki/Letter of credit/.

[2] “Invoice discounting,” https://en.wikipedia.org/wiki/Factoring (finance).

[3] “Reverse factoring (supply chain finance),” https://en.wikipedia.org/wiki/
Reverse factoring.

[4] “GS1 XML standards 3.2,” https://www.gs1.org/standards/gs1-xml/3-2/.

[5] “TradeLens,” https://www.tradelens.com.

[6] K. Narayanam, S. Goel, A. Singh, Y. Shrinivasan, S. Chakraborty,
P. Selvam, V. Choudhary, and M. Verma, “Blockchain based e-invoicing
platform for global trade,” in 2020 IEEE International Conference on
Blockchain (Blockchain), 2020, pp. 385–392.

[7] http://ibm.biz/BlockchainSolutionFromWalmartCanadaAndDLTLabs. [8] S. E. Chang, H. L. Luo, and Y. Chen, “Blockchain-enabled trade
finance innovation: A potential paradigm shift on using letter of credit,”
Sustainability, vol. 12, no. 1, p. 188, 2020.

[9] J. Chiu and T. V. Koeppl, “Blockchain-based settlement for asset
trading,” The Review of Financial Studies, vol. 32, no. 5, pp. 1716–
1753, 2019.

[10] A. Bogucharskov, I. Pokamestov, K. Adamova, and Z. N. Tropina,
“Adoption of blockchain technology in trade finance process,” Journal
of Reviews on Global Economics, vol. 7, pp. 510–515, 2018.