



SECURE THE DRUG COMPONENTS USING NAÏVE BAYES AND SVM

A PROJECT REPORT

Submitted by

BHAGYA SREE.C.R [REGISTER NO: 211417104035]

LAVANYA.T [REGISTER NO: 211414104130]

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

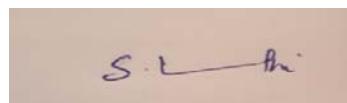
PANIMALAR ENGINEERING COLLEGE, CHENNAI-600123.

ANNA UNIVERSITY: CHENNAI 600 025

AUGUST 2021

BONAFIDE CERTIFICATE

Certified that this project report “**SECURE DRUG COMPONENTS USING NAÏVE BAYES AND SVM**” is the bonafide work of “**BHAGYA SREE.C.R (211417104035), LAVANYA.T (211417104130)**” who carried out the project work under my supervision.



SIGNATURE

**Dr.S.MURUGAVALLI, M.E., Ph.D.,
HEAD OF THE DEPARTMENT**

DEPARTMENT OF CSE,
PANIMALAR ENGINEERING COLLEGE,
NASARATHPETTAI,
POONAMALLEE,
CHENNAI-600 123.

SIGNATURE

**Mrs.M.S.VINMATHI
SUPERVISOR
ASSOCIATE PROFESSOR**

DEPARTMENT OF CSE,
PANIMALAR ENGINEERING COLLEGE,
NASARATHPETTAI,
POONAMALLEE,
CHENNAI-600 123.

Certified that the above candidate(s) was/ were examined in the Anna University Project

Viva-Voce Examination held on...05-08-2021.....

INTERNAL EXAMINER

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

We express our deep gratitude to our respected Secretary and Correspondent **Dr.P.CHINNADURAI, M.A., Ph.D.** for his kind words and enthusiastic motivation, which inspired us a lot in completing this project.

We would like to extend our heartfelt and sincere thanks to our Directors **Tmt.C.VIJAYARAJESWARI, Dr.C.SAKTHIKUMAR, M.E., Ph.D.,** and **Tmt. SARANYASREE SAKTHIKUMAR B.E.,M.B.A.,** for providing us with the necessary facilities for completion of this project.

We also express our gratitude to our Principal **Dr.K.Mani, M.E., Ph.D.** for his timely concern and encouragement provided to us throughout the course.

We thank the HOD of CSE Department, **Dr. S.MURUGAVALLI, M.E.,Ph.D.,** for the support extended throughout the project.

We would like to thank my **Project Guide Mrs.M.S.VINMATHI** and all the faculty members of the Department of CSE for their advice and suggestions for the successful completion of the project.

Finally, we would like to thank our dear Parents and friends for their constant moral support, encouragement and the environment to work efficiently.

**BHAGYA SREE.C.R
LAVANYA.T**

ABSTRACT

In this paper, we propose a framework to secure drug components in the cloud. Specifically, we design for multiple drug formula providers' to use the cloud securely. In our approach, the analyzer trains the drug formulas using Support Vector Machine (SVM) and naïve Bayes. To perform integer and fraction computations in the cloud server, we designed secure computation protocols. We securely train the SVM to privately refresh the selected SVM parameters using the two protocols, which are SVM parameter selection protocol and sequential minimal optimization protocol. We train NB based on Bayes theorem with an assumption of independence among predictors. To determine whether a drug compound is active or inactive in a cloud, the trained SVM and NB classifier is used. Lastly, we prove that the proposed framework achieves the goal that facilitate drug manufacturers to securely outsource their formulas without privacy leakage to unauthorized parties in the cloud for storage and for SVM and NB training.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	4
	LIST OF TABLES	7
	LIST OF FIGURES	8
	LIST OF SYMBOLS, ABBREVIATIONS	9
1.	INTRODUCTION	10
	1.1 Overview	11
	1.2 Problem Definition	11
2.	LITERATURE SURVEY	14
3.	SYSTEM ANALYSIS	27
	3.1 Existing System	28
	3.2 Proposed system	28
	3.3 Requirement Analysis and Specification	30
	3.3.1 Input Requirements	32
	3.3.2 Output Requirements	32
	3.3.3 Functional Requirements	33
	3.4 Hardware Environment	34
	3.5 Software Environment	34
4.	SYSTEM DESIGN	35
	4.1. ER diagram	36
	4.2 Data dictionary	37

CHAPTER NO.	TITLE	PAGE NO.
	4.3 Table Normalization	38
	4.4 Data Flow Diagram	43
	4.5 UML Diagrams	46
5.	SYSTEM ARCHITECTURE	51
	5.1 Architecture Overview	52
	5.2 Module Design Specification	54
	5.3 Program Design Language	55
6.	SYSTEM IMPLEMENTATION	59
	6.1 Client-side coding	60
	6.2 Server-side coding	64
7.	SYSTEM TESTING	75
	7.1 Unit Testing	76
	7.2 Integration Testing	76
	7.3 Test Cases & Reports / Performance Analysis	77
8.	CONCLUSION	80
	8.1 Conclusion and Future Enhancements	81
	APPENDICES	82
	A.1 Sample Screens	83
	A.2 Publications	100
	REFERENCES	101

LIST OF TABLES

TABLE NO.	NAME	PAGE NO.
2.1	Comparison table for Literature Survey	22
4.1	Fields used in forms	37
4.2	First normalized table	39
4.3	Second normalized table	40
4.4	Third normalized table	41
7.1	Test case design	78

LIST OF FIGURES

FIGURE NO.	NAME	PAGE NO.
3.1	Drug Discovery Cycle	29
3.2	Java Working Process	31
4.1	E-R Diagram for drug secure system	36
4.2	Context level Diagram for registration	43
4.3	Level 1 DFD for module Description	44
4.4	Level 2 DFD for module Description	44
4.5	Level 3 DFD for module Description	45
4.6	Use case Diagram for drug secure system	46
4.7	Sequence Diagram for drug secure system	47
4.8	Collaboration Diagram for drug secure system	48
4.9	Activity Diagram for drug secure system	49
4.10	Class Diagram for drug secure system	50
5.1	System Architecture for secure drug component system	53
5.2	2D Diagram for SVM Algorithm	56

LIST OF ABBREVIATIONS

JDK	Java Development Toolkit
DEX	Dalvik Executables
TCP	Transmission Control Protocol
IP	Internet Protocol
HTTP	Hyper Text Transfer Protocol
ADT	Android Development Tool

INTRODUCTION

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW OF THE PROJECT

Drug discovery is the process through which potential new therapeutic entities are identified, using a combination of computational, experimental, translational, and clinical models. Drug discovery is generally the process of identifying one or more active ingredients from traditional remedies, and includes the identification of screening hits, medicinal chemistry and optimization of these hits to increase the affinity, selectivity (to reduce the potential of side effects), bioavailability, and metabolic half-life. Drug discovery and development is a long-term, competitive, expensive and complicated process. Bringing the drugs from the bench to the market, that is, from screening and identification of the drug to its introduction to the market, takes several years of efforts. The process of discovering and developing a new drug involves an intricate interaction between investors, industry, academia, patent laws, regulatory authorities, marketing and the necessity to balance confidentiality with communication. The complete process of presenting a drug to the patients involves four stages- drug discovery, drug development, regulatory review and approval, and marketing.

1.2 PROBLEM DEFINITION OF THE PROJECT

In the drug discovery process, the first step is to identify an appropriate 'drugable' target, which can be a biomolecule or a protein receptor. After the target has been identified, the next step involves target validation and the confirmation. This is followed by testing of the target against different small molecule compounds to identify lead compounds. The lead compounds can be identified by screening a library of compounds through various methods, such as high-throughput screening, de novo synthesis and isolation from the natural products. The drug development phase involves stringent testing and optimization

of the selected compounds to identify the 'drug candidate' which might be most effective in terms of safety, toxicity, dosage, and efficacy. For this purpose, the selected lead compounds are tested in cells (*in vitro*) and in animals (*in vivo*) to study their pharmacodynamics and pharmacokinetic properties. The successful lead candidate must be non-toxic and should be absorbed into the bloodstream, can be distributed to the proper action site in the body, and can be metabolized efficient and effective as well as successfully excreted from the body. This part of the development process is referred to as the 'preclinical phase'. The outcome of the 'clinical trial' decides whether the drug candidate is safer and effective enough in treating the disease. At this point, new drug applications with all the essential evidence, including quality, preclinical and clinical data collected during development of the drug candidate, are submitted to the relevant regulatory authorities. At this point, there may be a chance where the drug information gets disclosed. Due to the significant investments and high commercial values involved in drug discovery, privacy is an important factor .To secure the drug components, a data mining tools is used.

Of the data mining tools, Support Vector Machine (SVM) has a relatively high decision rate and has been widely used in recent times to predict ligand-based chemical compounds in drug discovery. In approaches using SVMs, known drug formulas datasets is used to train the SVM classifier, and the new drug compound visual scanning is done by trained SVM. As privacy is prime, how can we minimize the risk of unauthorized disclosure during the SVM training phase? In this context, when a researcher sends some chemical compounds to the cloud for SVM classification, it is important to ensure that the potential new drug compounds will not be leaked to a third party, such as a competing pharmaceutical corporation. Furthermore, to train the SVM, multiple pharmaceutical corporations may collaborate in order to increase the SVM decision rate. At the same time, these corporations do not wish to reveal their datasets. How to achieve secure SVM

training and decision under multiple data sources without compromising, the privacy of each individual party remains a research and operational challenge. Thus, in this paper, we propose secure drug components using SVM and Naïve Bayes for Securing Drug discovery in the cloud environment. Unlike existing drug discovery frameworks, our framework seeks to achieve the following

- Secure Outsourced Data Storage: The drug formula owner can securely outsource the data (e.g. drug formula) to the cloud for storage without leaking the data to unauthorized third parties.
- Secure Multi-Source SVM Training: The POD allows an authorized model provider to use other drug formula owners' encrypted data to train the SVM on the fly. The model provider can decrypt and obtain the trained model without knowing (contents of) the training dataset.
- Secure SVM Drug Decision: An authorized tester can securely upload his/her drug chemical compounds to the cloud and determine whether the compound is active or not in a privacy-preserving way.

Commercialization is the last phase of drug development process. Once the drug has been approved, it is marketed or commercialized. The drug manufacturers need to submit marketing authorization applications in every country in which they want to sell the drug. As the drug is typically targeted to a very large number of patients, the manufacturer is expected to monitor this stage cautiously and submit reports to the FDA. The reports include evidence for medicine-related problems, e.g., treatment failure, adverse reaction, counterfeit/poor quality medicines, drug interactions, or incorrect use. These reports are significant in terms of generating proof of efficacy that will inspire public confidence and trust.

LITERATURE SURVEY

CHAPTER 2

LITERATURE SURVEY

DEEP LEARNING IN DRUG DISCOVERY AND MEDICINE; SCRATCHING THE SURFACE

The practice of medicine is ever evolving. Diagnosing disease, which is often the first step in a cure, has seen a sea change from the discerning hands of the neighborhood physician to the use of sophisticated machines to use of information gleaned from biomarkers obtained by the most minimally invasive of means. The last 100 or so years have borne witness to the enormous success story of allopathy, a practice that found favor over earlier practices of medical purgatory and homeopathy. Nevertheless, failures of this approach coupled with the omics and bioinformatics revolution spurred precision medicine, a platform wherein the molecular profile of an individual patient drives the selection of therapy. Indeed, precision medicine-based therapies that first found their place in oncology are rapidly finding uses in autoimmune, renal and other diseases. More recently a new renaissance that is shaping everyday life is making its way into healthcare. Drug discovery and medicine that started with Ayurveda in India are now benefiting from an altogether different artificial intelligence (AI)—one which is automating the invention of new chemical entities and the mining of large databases in health-privacy-protected vaults. Indeed, disciplines as diverse as language, neurophysiology, chemistry, toxicology, biostatistics, medicine and computing have come together to harness algorithms based on transfer learning and recurrent neural networks to design novel drug candidates, *a priori* inform on their safety, metabolism and clearance, and engineer their delivery but only on demand, all the while cataloging and comparing omics signatures across traditionally classified diseases to enable basket treatment strategies. This review highlights inroads made and being made in directed-drug design and molecular therapy.[1]

A NOVEL NEUTROSOPHIC WEIGHTED EXTREME LEARING MACHINE FOR IMBALANCED DATA SET

Extreme learning machine (ELM) is known as a kind of single-hidden layer feedforward network (SLFN), and has obtained considerable attention within the machine learning community and achieved various real-world applications. It has advantages such as good generalization performance, fast learning speed, and low computational cost. However, the ELM might have problems in the classification of imbalanced data sets. In this paper, we present a novel weighted ELM scheme based on neutrosophic set theory, denoted as neutrosophic weighted extreme learning machine (NWELM), in which neutrosophic c-means (NCM) clustering algorithm is used for the approximation of the output weights of the ELM. We also investigate and compare NWELM with several weighted algorithms. The proposed method demonstrates advantages to compare with the previous studies on benchmarks.[2]

DRUG DESIGN AND DISCOVERY: PRINCIPLES AND APPLICATIONS

Drug discovery is the process through which potential new therapeutic entities are identified, using a combination of computational, experimental, translational, and clinical models. Despite advances in biotechnology and understanding of biological systems, drug discovery is still a lengthy, costly, difficult, and inefficient process with a high attrition rate of new therapeutic discovery. Drug design is the inventive process of finding new medications based on the knowledge of a biological target. In the most basic sense, drug design involves the design of molecules that are complementary in shape and charge to the molecular target with which they interact and bind. Drug design frequently but not necessarily relies on computer modeling techniques and bioinformatics approaches in the big data era. In addition to small molecules,

biopharmaceuticals and especially therapeutic antibodies are an increasingly important class of drugs and computational methods for improving the affinity, selectivity, and stability of this protein-based therapeutics have also gained great advances [3]. Drug development and discovery includes preclinical research on cell-based and animal models and clinical trials on humans, and finally move forward to the step of obtaining regulatory approval in order to market the drug. Modern drug discovery involves the identification of screening hits, medicinal chemistry and optimization of those hits to increase the affinity, selectivity (to reduce the potential of side effects), efficacy/potency, metabolic stability (to increase the half-life), and oral bioavailability. Once a compound that fulfills all of these requirements has been identified, it will begin the process of drug development prior to clinical trials.[3]

PRIVACY-PRESERVING CLINICAL DECISION SUPPORT SYSTEM USING GAUSSIAN KERNEL-BASED CLASSIFICATION

A clinical decision support system forms a critical capability to link health observations with health knowledge to influence choices by clinicians for improved healthcare. Recent trends towards remote outsourcing can be exploited to provide efficient and accurate clinical decision support in healthcare. In this scenario, clinicians can use the health knowledge located in remote servers via the Internet to diagnose their patients. However, the fact that these servers are third party and therefore potentially not fully trusted raises possible privacy concerns. In this paper, we propose a novel privacy-preserving protocol for a clinical decision support system where the patients' data always remain in encrypted form during the diagnosis process. Hence the server involved in the diagnosis process is not able to learn any extra knowledge about the patient data and results. Our experimental results on popular medical data sets from UCI

database demonstrate that the accuracy of the proposed protocol is up to 97:21% and the privacy of patient data is not compromised.[4]

PRIVACY-PRESERVING MULTI-CLASS SUPPORT VECTOR MACHINE FOR OUTSOURCING THE DATA CLASSIFICATION IN CLOUD

Emerging cloud computing infrastructure replaces traditional outsourcing techniques and provides flexible services to clients at different locations via Internet. This leads to the requirement for data classification to be performed by potentially untrusted servers in the cloud. Within this context, classifier built by the server can be utilized by clients in order to classify their own data samples over the cloud. In this paper, we study a privacy-preserving (PP) data classification technique where the server is unable to learn any knowledge about clients' input data samples while the server side classifier is also kept secret from the clients during the classification process. More specifically, to the best of our knowledge, we propose the first known client-server data classification protocol using support vector machine. The proposed protocol performs PP classification for both two-class and multi-class problems. The protocol exploits properties of Pailler homomorphic encryption and secure two-party computation. At the core of our protocol lies an efficient, novel protocol for securely obtaining the sign of Pailler encrypted numbers.[5]

DRUG DISCOVERY AND DEVELOPMENT: AN INSIGHT INTO PHARMACOVIGILANCE

Drug discovery and development is a long-term, competitive, expensive and complicated process. Bringing the drugs from the bench to the market, that is, from screening and identification of the drug to its introduction to the market, takes several years of efforts [1]. The process of discovering and developing a new drug involves an

intricate interaction between investors, industry, academia, patent laws, regulatory authorities, marketing and the necessity to balance confidentiality with communication [2]. The complete process of presenting a drug to the patients involves four stages- drug discovery, drug development, regulatory review and approval, and marketing.[6]

DRUG DISCOVERY AND DEVELOPMENT: ROLE OF BASIC BIOLOGICAL RESEARCH

This article provides a brief overview of the processes of drug discovery and development. Our aim is to help scientists whose research may be relevant to drug discovery and/or development to frame their research report in a way that appropriately places their findings within the drug discovery and development process and thereby support effective translation of preclinical research to humans. One overall theme of our article is that the process is sufficiently long, complex, and expensive so that many biological targets must be considered for every new medicine eventually approved for clinical use and new research tools may be needed to investigate each new target. Studies that contribute to solving any of the many scientific and operational issues involved in the development process can improve the efficiency of the process. An awareness of these issues allows the early implementation of measures to increase the opportunity for success. As editors of the journal, we encourage submission of research reports that provide data relevant to the issues presented.[7]

MACHINE LEARNING CLASSIFICATION OVER ENCRYPTED DATA

Machine learning classification is used for numerous tasks nowadays, such as medical or genomics predictions, spam detection, face recognition, and financial predictions. Due to privacy concerns, in some of these applications, it is important that the data and the classifier remain confidential. In this work, we construct three major classification protocols that satisfy this privacy constraint: hyperplane decision, Naïve Bayes, and

decision trees. We also enable these protocols to be combined with AdaBoost. At the basis of these constructions is a new library of building blocks, which enables constructing a wide range of privacy-preserving classifiers; we demonstrate how this library can be used to construct other classifiers than the three mentioned above, such as a multiplexer and a face detection classifier. We implemented and evaluated our library and our classifiers. Our protocols are efficient, taking milliseconds to a few seconds to perform a classification when running on real medical datasets.[8]

SAFE DRUG ADMINISTRATION: RIGHT OF EVERY PATIENT

Medication errors are well known problem in hospitals. Registered nurses are accountable for their actions and omissions when administering any medication. To assess and compare the knowledge of nursing personnel regarding injection administration before and after demonstration of safe injection administration techniques. Experimental research approach was used with one group pretest- posttest design. Data was collected from 163 nursing personnel working in different ward of the hospital were which was selected by using total Enumeration sampling technique. The mean with standard deviation post-implementation knowledge score (8.0 ± 1.5) was higher than the mean with standard deviation of pre-implementation knowledge scoring (4.2 ± 1.5) scores. “t” value” calculated between the pre-implementation practice scores and post implementation practice scores with a df (162) was found to be statistically significant at $0.00^{**}(p > 0.05)$. Thus, the study showed that along with theoretical education, special attention must be paid to the practical aspect of education. Demonstration and other such educational programs are very effective methods in updating the knowledge of the health care professionals.[9]

NEEDLE FREE DRUG DELIVERY DEVICES MARKET SIZE, INDUSTRY OUTLOOK AND OPPORTUNITY ANALYSIS REPORT

According to the global Needle-Free Drug Delivery Devices Market report published by Value Market Research, the market is expected to touch USD XX.X MN by 2025, with a CAGR of X.X% growing from valued USD XX.X MN (by revenue) in 2018. This is a tailored made research service providing informative data and various critical aspects of the market such as market outlook, market share, growth, and trends. Further, the report also offers evidence-based information that helps to transform clients business and achieve their business goals. Moreover, the report also highlights the key strategy of top players. Additionally, this report covers a wide spectrum of services such as the latest technology trend, market opportunity analysis, and competitive landscape.[10]

2.11 COMPARISON TABLE

Table 2.1 Comparison Table for Literature Survey

TITLE	AUTHOR	YEAR	INFERENCE	DRAWBACKS
Deep Learning in Drug Discovery and Medicine; Scratching the Surface	Dibyendu Dana , Satishkumar V. Gadhiya , Luce G. St. Surin , David Li , Farha Naaz ,Quaisar Ali	British journal of pharmacology, vol. 162, no. 6, pp. 1239–1249, 2018	Drug discovery and medicine that started with Ayurveda in India are now benefiting from an altogether different artificial intelligence (AI)—one which is automating the invention of new chemical entities and the mining of large databases in health-privacy-protected vaults. Indeed, disciplines as diverse as language, chemistry, toxicology, biostatistics, medicine and computing have come together to harness algorithms based on transfer learning and recurrent neural networks to design novel drug candidates but only on demand.	1)The molecular profile of an individual patient drives the selection of therapy. 2)Post-marketing surveillance data show that its benefit is not experienced by every patient.
A Novel Neutrosophic Weighted Extreme Learning Machine for Imbalanced Data Set	Yaman Akbulut 1 ID , Abdulkadir Sengür 1,* ID , Yanhui Guo 2 and Florentin Smarandache	Journal of chemical information and computer sciences, vol. 44, no. 5, pp. 1630–1638, 2017	It has advantages such as good generalization performance, fast learning speed, and low computational cost. Neutrosophic weighted extreme learning machine (NWELM), in which neutrosophic c-means (NCM) clustering algorithm is used for the approximation of the output weights of the ELM.	1)Problems in the classification of imbalanced data sets. 2)ELM suffers from the presence of irrelevant variables in the large and high dimensional real data set
Drug discovery and developm	Richard C. Mohsa, Nigel H. Greig	IEEE Journal of Biomedic	Better medicines that are iterative improvements on current medications are valuable as they may offer benefits over existing	The process is sufficiently long, complex, and expensive

ent: Role of basic biological research		al and Health Informatics, vol. 20, pp. 655 – 668, 2017	medications in terms of potency, safety, tolerability, or convenience. The drug discovery and development process and thereby support effective translation of preclinical research to humans	so that many biological targets must be considered for every new medicine eventually approved for clinical use and new research tools may be needed to investigate each new target.
Needle Free Drug Delivery Devices Market Size, Industry Outlook and Opportunity Analysis Report 2018-2025	Aniket Sharma	8th ACM SIGSAC symposium on Information, computer and communications security. ACM, 2013, pp. 541– 546.2019	The global Needle-Free Drug Delivery Devices Market report is a tailored made research service providing informative data and various critical aspects of the market such as market outlook, market share, growth, and trends. Moreover, the report also highlights the key strategy of top players. Additionally, this report covers a wide spectrum of services such as the latest technology trend, market opportunity analysis, and competitive landscape	This Needle free drug discovery is yet to be developed and to carry on this discovery we need advanced expensive technology.
Drug Design and Discovery : Principles and Applications	Shu-Feng Zhou , Wei-Zhu Zhong	Drug discovery today, vol. 17, no. 19, pp. 1088– 1102, 2017	Drug discovery is the process through which potential new therapeutic entities are identified, using a combination of computational, experimental, translational, and clinical models. Modern drug discovery involves the identification of screening hits, medicinal chemistry and optimization of those hits to	Despite advances in biotechnology and understanding of biological systems, drug discovery is still a lengthy, costly, difficult,

			increase the affinity, selectivity (to reduce the potential of side effects), efficacy/potency, metabolic stability (to increase the half-life), and oral bioavailability	and inefficient process with a high attrition rate of new therapeutic discovery.
Privacy-Preserving Clinical Decision Support System Using Gaussian Kernel-Based Classification	Yogachandan Rahulamathavan, Suresh Veluru, Raphael	Biomedical and Health Informatics, IEEE Journal of, vol. 18, no. 1, pp. 56–66, 2015	A clinical decision support system forms a critical capability to link health observations with health knowledge to influence choices by clinicians for improved healthcare. Remote outsourcing can be exploited to provide efficient and accurate clinical decision support in healthcare.	Servers are third party and therefore potentially not fully trusted raises possible privacy concerns.
Machine Learning Classification over Encrypted Data	Raphael Bost, Raluca Ada Popa, Stephen Tu	22nd Annual Network and Distributed System Security Symposium, NDSS 2015, San Diego, California, USA, February 8-11, 2015.	Machine learning classification is used for numerous tasks nowadays, such as medical or genomics predictions, spam detection, face recognition, and financial predictions. Due to privacy concerns, in some of these applications, it is important that the data and the classifier remain confidential. Three major classification protocols that satisfy this privacy constraint: hyperplane decision, Naïve Bayes, and decision trees. The basis of these constructions is a new library of building blocks, which enables constructing a wide range of privacy-preserving classifiers.	In the most basic sense, drug design involves the design of molecules that are complementary in shape and charge to the molecular target with which they interact and bind. No security for drug components while train the data set.

Privacy-Preserving Multi-Class Support Vector Machine for Outsourcing the Data Classification in Cloud	Kanapathipillai Cumanan, Muttukrishnan Rajarajan	IEEE on Dependable and Secure Computing, vol. 11, no. 5, pp. 467–479, 2014	<p>Classifier built by the server can be utilized by clients in order to classify their own data samples over the cloud.</p> <p>A privacy-preserving (PP) data classification technique where the server is unable to learn any knowledge about clients' input data samples while the server side classifier is also kept secret from the clients during the classification process</p>	<p>Emerging cloud computing infrastructure replaces traditional outsourcing techniques and provides flexible services to clients at different locations via Internet. This leads to the requirement for data classification to be performed by potentially untrusted servers in the cloud. Releasing the data samples owned by the clients to the cloud raises privacy concerns since the data processed in the cloud are often outsourced to untrusted-third-party-servers.</p>
Machine learning methods in chemoinformatics	J. B. Mitchell	Wiley Interdisciplinary Reviews: Computational	Machine learning algorithms are generally developed in computer science or adjacent disciplines and find their way into chemical modeling by a process of diffusion. This discussion is	Although numerous articles cited herein have compared performances of

		Molecular Science, . 4, no. 5, pp. 468–481, 2014.	<p>concentrate on methods for supervised learning, predicting the unknown property values of a test set of instances, usually molecules, based on the known values for a training set. Particularly relevant approaches include Artificial Neural Networks, Random Forest, Support Vector Machine, k-Nearest Neighbors and naïve Bayes classifiers.</p>	the various machine-learning algorithms used in chemoinformatics, there is no single best method for all problems. The relative abilities of methods will depend on the size and distribution in chemical space of the dataset, the linearity etc.
Computer-aided drug design platform using pymol	M. A. Lill and M. L. Danielson	Journal of computer-aided molecular design, vol. 25, no. 1, pp. 13–19, 2011.	<p>Over the decades, many powerful standalone tools for computer-aided drug discovery have been developed in academia providing insight into protein-ligand interactions. As programs are developed by various research groups, a consistent user-friendly graphical working environment combining computational techniques such as docking, scoring, molecular dynamics simulations, and free energy calculations is needed.</p>	Despite advances in biotechnology and understanding of biological systems, drug discovery is still a lengthy, costly, difficult, and inefficient process with a high attrition rate of new therapeutic discovery.

SYSTEM ANALYSIS

CHAPTER 3

SYSTEM ANALYSIS

Systems design is the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. Systems design could see it as the application of systems theory to product development.

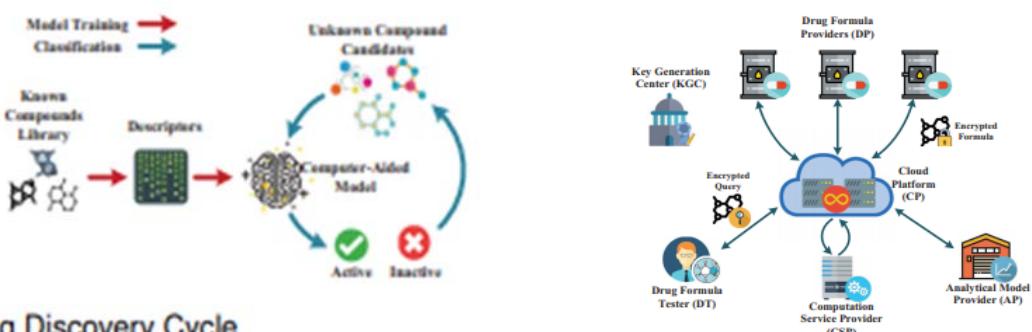
3.1 EXISTING SYSTEM

In this existing system, securing drug components is designed to allow the cloud to securely use multiple drug formula providers' drug formulas to train Support Vector Machine (SVM) provided by the analytical model provider. In this approach, it is designed to secure computation protocols to allow the cloud server to perform commonly used integer and fraction computations. To securely train the SVM, it is designed as a secure SVM parameter selection protocol to select two SVM parameters and construct a secure sequential minimal optimization protocol to privately refresh both selected SVM parameters. The trained SVM classifier can be used to determine whether a drug chemical compound is active or not in a privacy-preserving way. The existing datasets of known drug formulas to train the SVM classifier, and the trained SVM classifier can be used for new drug compound visual scanning. Due to the significant investments and high commercial values involved in drug discovery, privacy is an important factor. When a researcher sends some chemical compounds to the cloud for SVM classification, it is important to ensure that the potential new drug compounds will not be leaked to a third-party, such as a competing pharmaceutical corporation.

3.2 PROPOSED SYSTEM

We propose secure drug discovery components in the cloud environment. Unlike existing drug discovery frameworks, our POD seeks to achieve it efficiently. We are

not using three real time datasets to check the efficiency of potential new drug component. Instead of using existing datasets, we are using another one data-mining algorithm Naïve Bayes (NB). It is a classification technique based on Bayes' Theorem with an assumption of independence among predictors. NB classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature also fast to predict class of test data set and performs well in multi class prediction. These two algorithms such as SVM and NB are used to train the uploaded drug dataset (CSV file). Thus we will get trained data and accuracy for that uploaded dataset. The trained data and accuracy will be sent to the owner from python server. Drug tester will check that new drug component. To check the new drug tester has to send request for accessing the drug component. As privacy is an important factor due to the significant investments and high commercial values involved in drug discovery, even drug tester does not know the contents of that file; they will get the trained data only. Once the Testing is completed, the tester sends result to the admin. If that particular drug component is still retained in the cloud then it is assumed that component is still active and passed the testing successfully. If not, then the drug component is removed from the cloud. Finally, admin will approve the drug component.



1. Drug Discovery Cycle

Figure 3.1 Drug discovery cycle

3.3 REQUIREMENT ANALYSIS AND SPECIFICATION

Requirement Specification is a complete description of the behavior of a system to be developed. It includes a set of use cases that describe all the interactions the users will have with the software. This chapter clearly depicts the software languages used in the system design and the significance of it.

The requirements specification is a technical specification of requirements for the software products. It is the first step in the requirements analysis process it lists the requirements of a particular software system including functional, performance and security requirements. The requirements also provide usage scenarios from a user, an operational and an administrative perspective.

The purpose of software requirements specification is to provide a detailed overview of the software project, its parameters and goals. This describes the project target audience and its user interface, hardware and software requirements. It defines how the client, team and audience see the project and its functionality.

JAVA

Java has been around since 1991, developed by a small team of Sun Microsystems developers in a project originally called the Green project. The intent of the project was to develop a platform-independent software technology that would be used in the consumer electronics industry. The language that the team created was originally called Oak. Web surfing has become an enormously popular practice among millions of computer users. Until Java, however, the content of information on the Internet has been a bland series of HTML documents. Web users are hungry for applications that are interactive, that users can execute no matter what hardware or software platform they are using, and that travel across heterogeneous networks and do not spread viruses to their computers. Java can create such applications. Java byte codes as the machine code instructions for the **Java Virtual Machine (JVM)**. Every Java interpreter, whether it is

a Java development tool or a Web browser that can run Java applets, is an implementation of JVM. That JVM can also be implemented in hardware. Java byte codes help make “write once, run anywhere” possible. Using compiler, compile the Java program into byte codes on any platform that has a Java compiler. The byte codes can then be run on any implementation of the JVM.

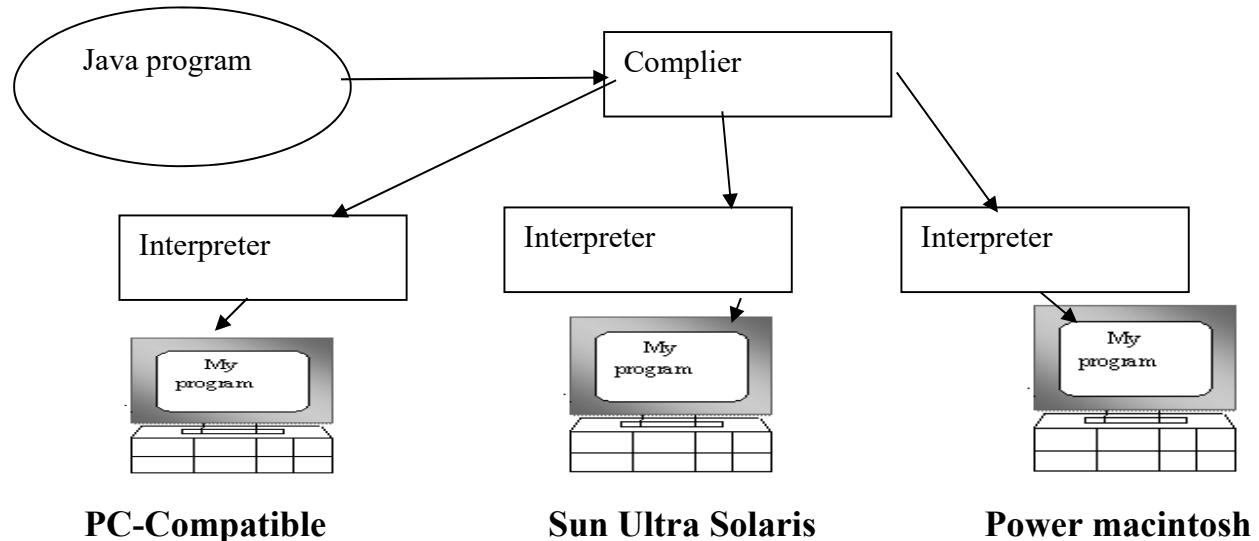


Figure 3.2 Java working process

APACHE TOMCAT SERVER

Apache Tomcat (formerly under the Apache Jakarta Project; Tomcat is now a top-level project) is a web container developed at the Apache Software Foundation. Tomcat implements the servlet and the JavaServer Pages (JSP) specifications from Sun Microsystems, providing an environment for Java code to run in cooperation with a web server. It adds tools for configuration and management but can also be configured by editing configuration files that are normally XML-formatted. Because Tomcat includes its own HTTP server internally, it is also considered a standalone web server.

DATA MINING

In general terms, “**Mining**” is the process of extraction of some valuable material from the earth e.g. coal mining, diamond mining etc. In the context of computer science, “Data Mining” refers to the extraction of useful information from a bulk of data or data warehouses, the information gathered from Data Mining helps to predict hidden patterns, future trends and behaviors and allowing businesses to take decisions. Technically, data mining is the computational process of analyzing data from different perspective, dimensions, angles and categorizing/summarizing it into meaningful information.

Data Mining can be applied to any type of data e.g. Data Warehouses, Transactional Databases, Relational Databases, Multimedia Databases, Spatial Databases, Time-series Databases, World Wide Web.

SVM (Support Vector Machine)

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane. SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.

Naive Bayes algorithm

It is a classification technique based on Bayes' Theorem with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. For example, a fruit may be considered an apple if it is red, round, and about 3 inches in diameter. Even if these features depend on each other or upon the existence of the other features, all of these properties independently contribute to the probability that this fruit is an apple and that is why it is known as 'Naive'.

Naive Bayes model is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods.

Bayes theorem provides a way of calculating posterior probability $P(c|x)$ from $P(c)$, $P(x)$ and $P(x|c)$. Look at the equation below:

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)}$$

$$P(c|X) = P(x_1|c) \times P(x_2|c) \times \dots \times P(x_n|c) \times P(c)$$

Above,

- $P(c|x)$ is the posterior probability of class (c, target) given predictor (x, attributes).
- $P(c)$ is the prior probability of class.
- $P(x|c)$ is the likelihood which is the probability of predictor given class.
- $P(x)$ is the prior probability of predictor.

3.4 HARDWARE ENVIRONMENT

- Hard Disk : 80GB and Above
- RAM : 4GB and Above
- Processor : P IV and Above

3.5 SOFTWARE ENVIRONMENT

- Windows 7 and above(64 bit)
- JDK 1.8
- Python 3.6.3
- Tomcat 9.0.26
- MySQL

3.6 TECHNOLOGIES USED

- JAVA
- Spring Framework

SYSTEM DESIGN

CHAPTER 4

SYSTEM DESIGN

4.1 E-R DIAGRAM

An entity relationship model, also called an entity-relationship (ER) diagram, is a graphical representation of entities and their relationships to each other, typically used in computing about the organization of data within databases or information systems.

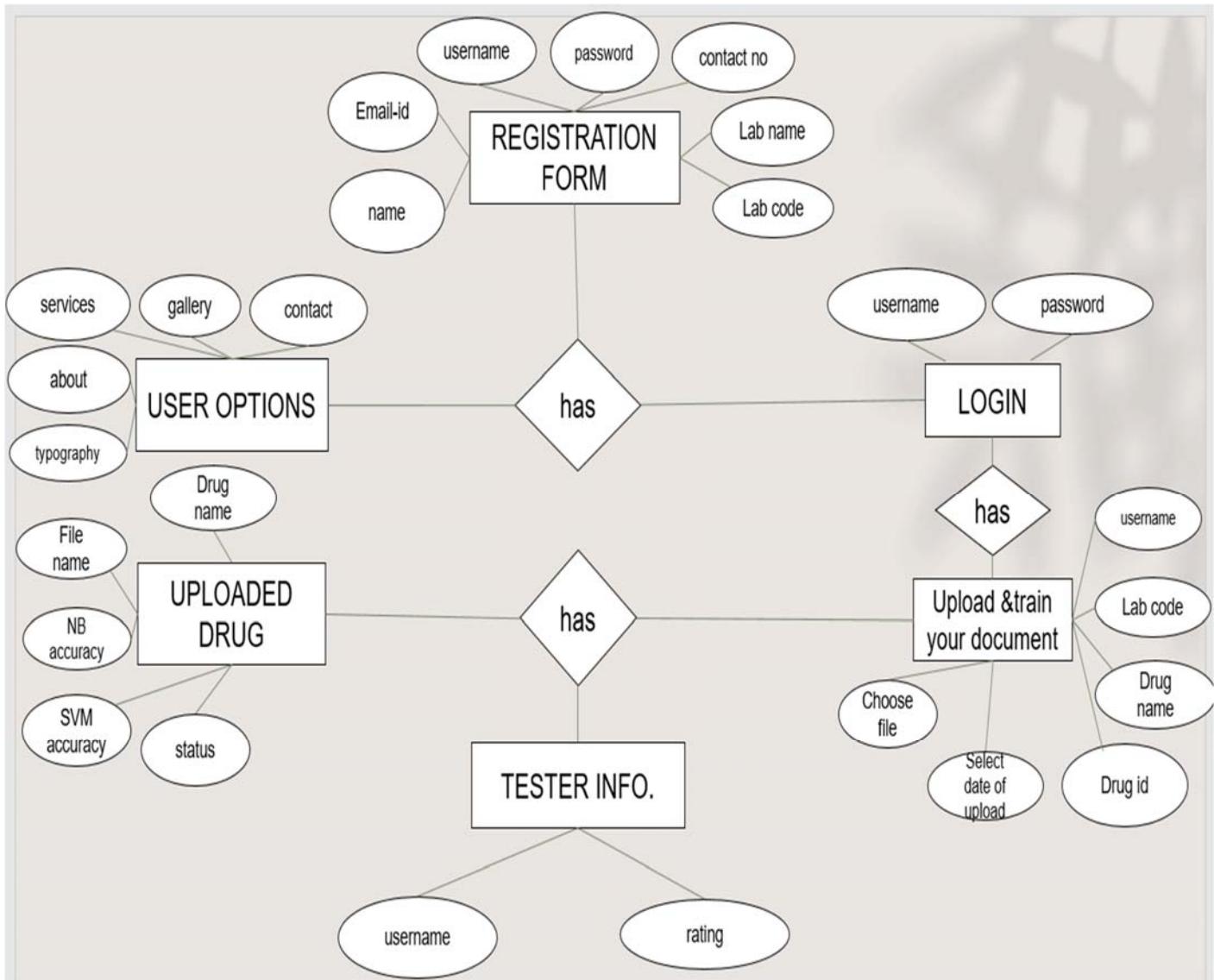


Figure 4.1 E-R Diagram for Drug Secure System

4.2 DATA DICTIONARY

A data dictionary, or metadata repository, as defined as a centralized repository of information about data such as meaning, relationships to other data, origin, usage, and format. It is also defined as a collection of tables with metadata. The term can have one of several closely related meanings pertaining to databases. A document describing a database or collection of databases. An integral component of a DBMS that is required to determine its structure. A piece of middleware that extends or supplants the native data dictionary of a DBMS.

FIELD NAME	DATATYPE	LENGTH	PRIMARY KEY	CONSTRAINTS
Username	Varchar	255	Yes	Primary key
Password	Varchar	255	No	Not null
Name	Varchar	255	No	Not null
Email	Varchar	255	Yes	Primary key
Phone	Varchar	255	Yes	Primary Key
Status	Varchar	255	No	Not null
Labname	Varchar	255	No	Not null
Labcode	Varchar	255	Yes	Primary Key
Drugname	Varchar	255	No	Not null
Drugid	Varchar	255	Yes	Primary Key
Date	Date	255	No	Not null

Table 4.1 Fields used in forms

4.3 TABLE NORMALIZATION

Normalization is a database design technique that reduces data redundancy and eliminates undesirable characteristics like Insertion, Update and Deletion Anomalies. Normalization rules divides larger tables into smaller tables and links them using relationships. The purpose of Normalization in SQL is to eliminate redundant (repetitive) data and ensure data is stored logically. The inventor of the relational model Edgar Codd proposed the theory of normalization of data with the introduction of the First Normal Form, and he continued to extend theory with Second and Third Normal Form. Later he joined Raymond F. Boyce to develop the theory of Boyce-Codd Normal Form.

4.3.1 FIRST NORMAL FORM

First normal form (1NF) is a property of a relation in a relational database. A relation is in first normal form if and only if the domain of each attribute contains only atomic (indivisible) values, and the value of each attribute contains only a single value from that domain .The first definition of the term, in a 1971 conference paper by Edgar Codd, defined a relation to be in first normal form when none of its domains have any sets as elements .These two definitions do not conflict each other. First normal form is an essential property of a relation in a relational database. Database normalization is the process of representing a database in terms of relations in standard normal forms, where first normal is a minimal requirement.

First normal form enforces these criteria:

- 1) Every column/attribute must be unique in each table
- 2) Create a separate table for each set of related data[definition needed]

3)All entries must be single-valued and atomic

LAB NAME	LAB CODE	NAME	EMAIL	CONTACT NO	USER NAME	PASSWORD
Medlife	ML3754	Jack	jackrose@gmail.com	9067395421	Jack99	*****
Medlife	ML3754	Jack	jackrose@gmail.com	9879904366	Jack99	*****
Realab	RL6996	Devmohan	devmohan1995@hotmail.com	8790653428	Dev_mohan	*****
UALab	UAL9395	Leela	leelaram@yahoo.com	7689004503	Leela_ram	*****
UALab	UAL9395	Leela	leelaram@yahoo.com	7996500887	Leela_ram	*****
AccuLab	AL5698	Harsha	harshayadhav@gmail.com	9003216914	harsha	*****
Labplex	LP7812	Neela	Neela23@gmail.com	9345813429	neela	*****
Labplex	LP7812	Neela	Neela23@gmail.com	7890448890	neela	*****

LAB NAME	LAB ID	NAME	EMAIL	CONTACT NO	USER NAME	PASSWORD
Medlife	ML3754	Jack	jackrose@gmail.com	9067395421 9879904366	Jack99	*****
Realab	RL6996	Devmohan	devmohan1995@hotmail.com	8790653428	Dev_mohan	*****
UALab	UAL9395	Leela	leelaram@yahoo.com	7689004503 7996500887	Leela_ram	*****
AccuLab	AL5698	Harsha	harshayadhav@gmail.com	9003216914	harsha	*****
Labplex	LP7812	Neela	Neela23@gmail.com	9345813429 7890448890	neela	*****

Table 4.2 First normalized table

4.3.2 SECOND NORMAL FORM

Second normal form (2NF) is a normal form used in database normalization. 2NF was originally defined by E. F. Codd in 1971. A relation is in the second normal form if it fulfills the following two requirements :

1)It is in first normal form.

2) It does not have any non-prime attribute that is functionally dependent on any proper subset of any candidate key of the relation. A non-prime attribute of a relation is an attribute that is not a part of any candidate key of the relation.

Put simply, a relation is in 2NF if it is in 1NF and every non-prime attribute of the relation is dependent on the whole of every candidate key. Note that it does not put any restriction on the non-prime to non-prime attribute dependency. That is addressed in third normal form.

NAME	EMAIL	CONTACT NO	USER NAME
Jack	jackrose@gmail.com	9067395421	Jack99
Devmohan	devmohan1995@hotmail.com	8790653428	Dev_mohan
Leela	leelaram@yahoo.com	7689004503	Leela_ram
Harsha	harshayadhav@gmail.com	9003216914	harsha
Neela	Neela23@gmail.com	9345813429	neela

User Name	Drug Name	Drug ID	Date of upload	File Upload
Jack99	zincovit	ZC683	05-18-2018	Zincovit.csv
Dev_mohan	multivit	MV395	08-23-2019	Multivit.csv
Leela_ram	truebasic	TB103	10-04-2020	Truebasic.csv
Harsha	matilda	M890	11-17-2020	Matilda.csv
neela	mbvip	MV532	07-04-2020	Mbvip.csv

User Name	Drug Name	DrugID	Date of upload	File Upload	Name	Email	Contact no
Jack99	zincovit	ZC683	05-18-2018	Zincovit.csv	Jack	jackrose@gmail.com	9067395421
Dev_mohan	multivit	MV395	08-23-2019	Multivit.csv	Devmohan	devmohan1995@hotmail.com	8790653428
Leela_ram	truebasic	TB103	10-04-2020	Truebasic.csv	Leela	leelaram@yahoo.com	7689004503
Harsha	matilda	M890	11-17-2020	Matilda.csv	Harsha	harshayadhav@gmail.com	9003216914
neela	mbvip	MV532	07-04-2020	Mbvip.csv	Neela	Neela23@gmail.com	9345813429

Table 4.3 Second normalized table

4.3.3 THIRD NORMAL FORM

Third normal form (3NF) is a database schema design approach for relational databases, which uses normalizing principles to reduce the duplication of data, avoid data anomalies, ensure referential integrity, and simplify data management. Edgar F. Codd, an English computer scientist who invented the relational model for database management, defined it in 1971. A relation is in third normal form, if there is no transitive dependency for non-prime attributes as well as it is in second normal form. A relation is in 3NF if at least one of the following condition holds in every non-trivial function dependency $X \rightarrow Y$:

1. X is a super key.
2. Y is a prime attribute (each element of Y is part of some candidate key).

User Name	Drug Name	DrugID	Date of upload	File Upload	Name	Email	Contact no
Jack99	zincovit	ZC683	05-18-2018	Zincovit.csv	Jack	jackrose@gmail.com	9067395421
Dev_mohan	multivit	MV395	08-23-2019	Multivit.csv	Devmohan	devmohan1995@hotmail.com	8790653428
Leela_ram	truebasic	TB103	10-04-2020	Truebasic.csv	Leela	leelaram@yahoo.com	7689004503
Harsha	matilda	M890	11-17-2020	Matilda.csv	Harsha	harshayadhav@gmail.com	9003216914
neela	mbvip	MV532	07-04-2020	Mbvip.csv	Neela	Neela23@gmail.com	9345813429

Drug Name	Drug ID	Date of upload	File Upload
zincovit	ZC683	05-18-2018	Zincovit.csv
multivit	MV395	08-23-2019	Multivit.csv
truebasic	TB103	10-04-2020	Truebasic.csv
matilda	M890	11-17-2020	Matilda.csv
mbvip	MV532	07-04-2020	Mbvip.csv

User Name	DrugID	Name	Email	Contact no
Jack99	ZC683	Jack	jackrose@gmail.com	9067395421
Dev_mohan	MV395	Devmohan	devmohan1995@hotmail.com	8790653428
Leela_ram	TB103	Leela	leelaram@yahoo.com	7689004503
Harsha	M890	Harsha	harshayadhav@gmail.com	9003216914
neela	MV532	Neela	Neela23@gmail.com	9345813429

Table 4.4 Third normalized table

4.4 DATA FLOW DIAGRAM

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system without going into detail, which can later be elaborated.

4.4.1 CONTEXT LEVEL DIAGRAM

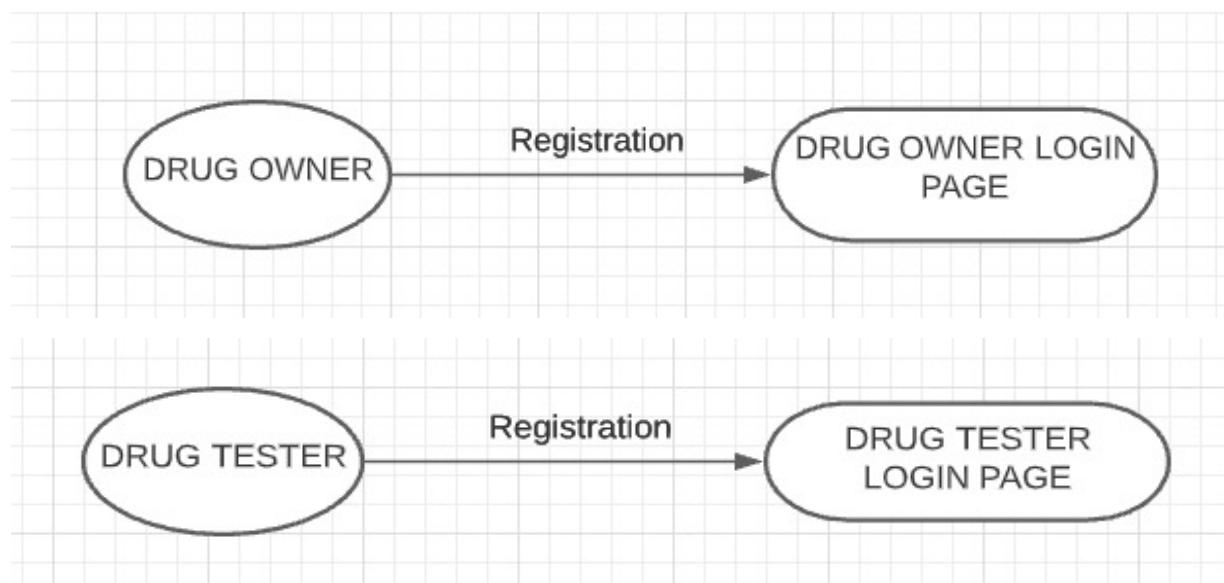
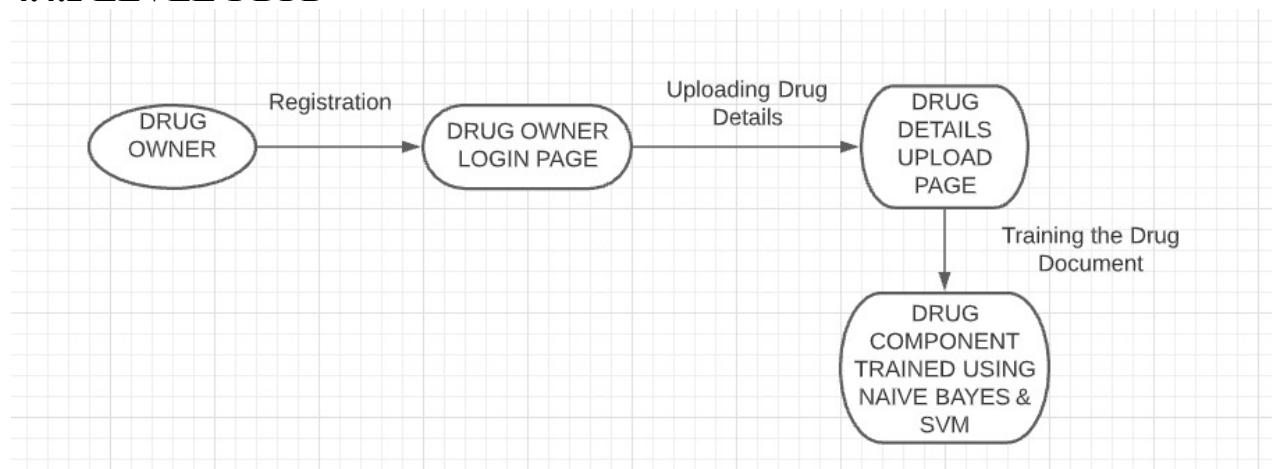


Figure 4.2 Context Level Diagram for registration

4.4.2 LEVEL 1 DFD



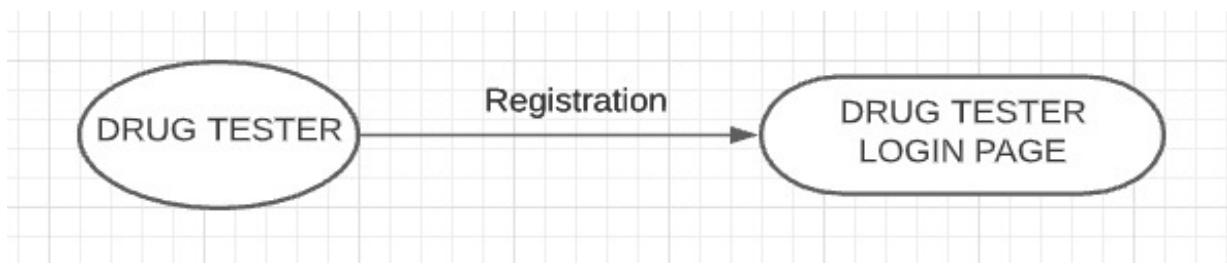


Figure 4.3 Level 1 DFD for module description

4.4.3 LEVEL 2 DFD

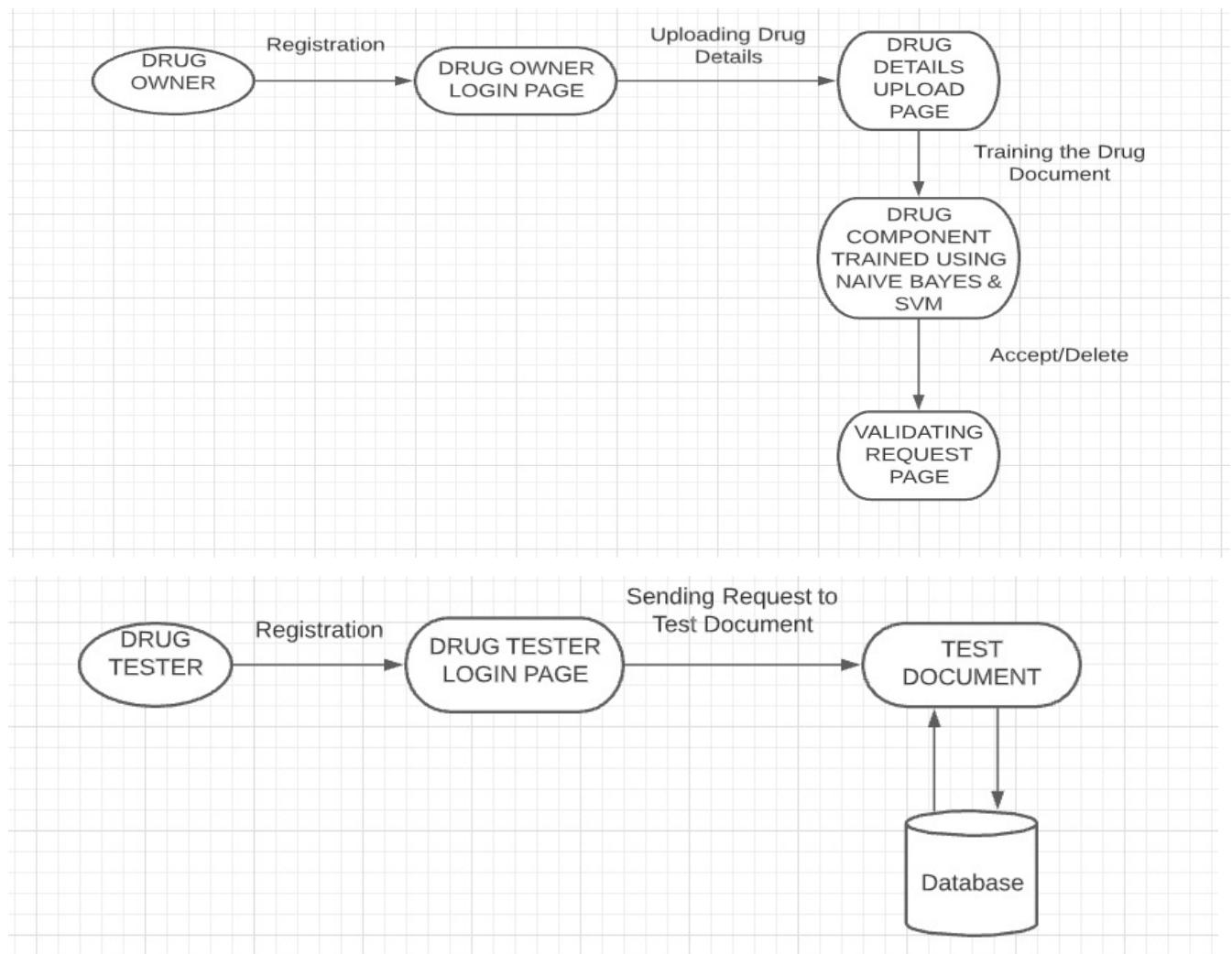


Figure 4.4 Level 2 DFD for module description

4.4.4 LEVEL 3 DFD

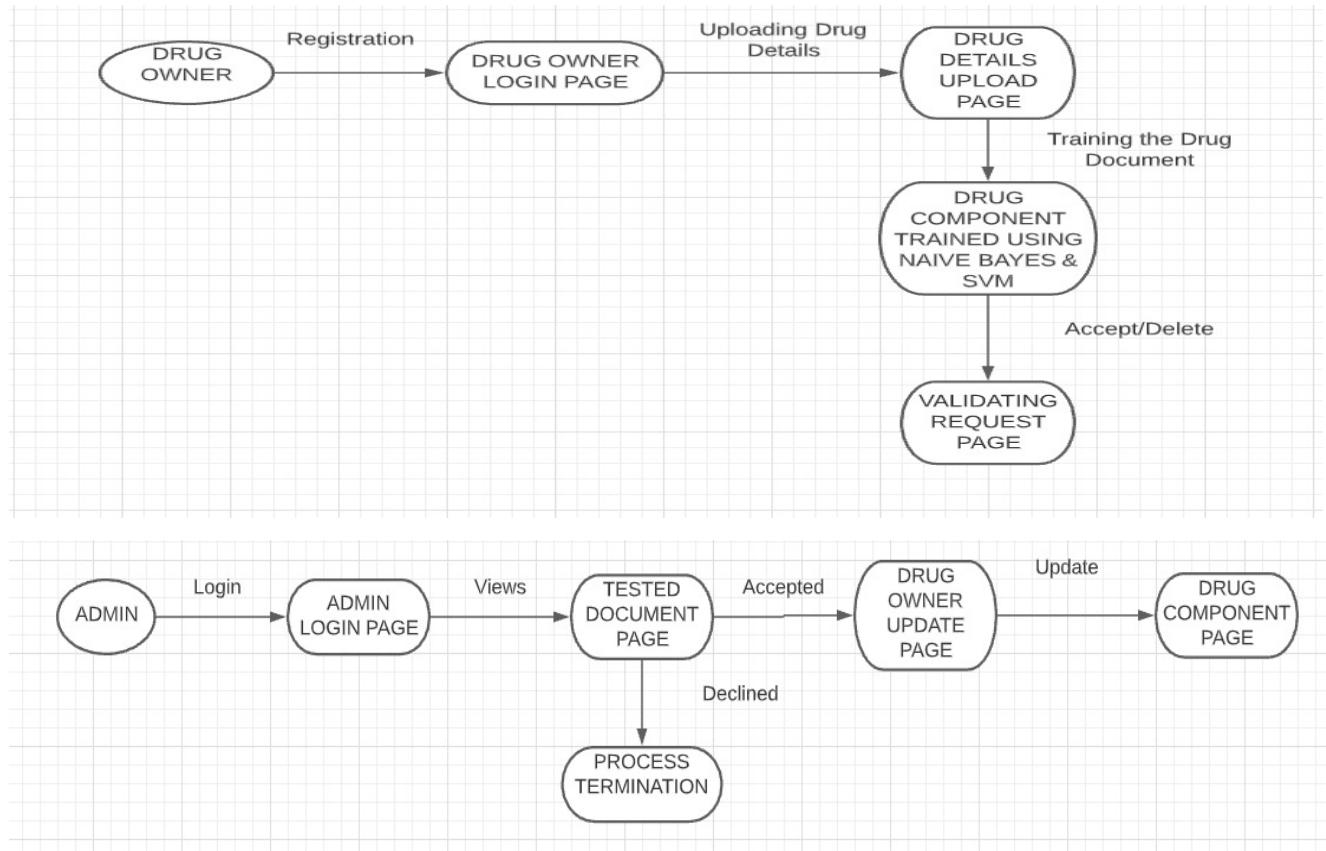


Figure 4.5 Level 3 DFD for overall description

4.5 UML DIAGRAM

4.5.1 USE CASE DIAGRAM

A use case diagram at its simplest is a representation of a user's interaction with the system and depicting the specifications of a use case. A use case diagram can portray the different types of users of a system and the various ways that they interact with the system. This type of diagram is typically used in conjunction with the textual use case and will often be accompanied by other types of diagrams as well.

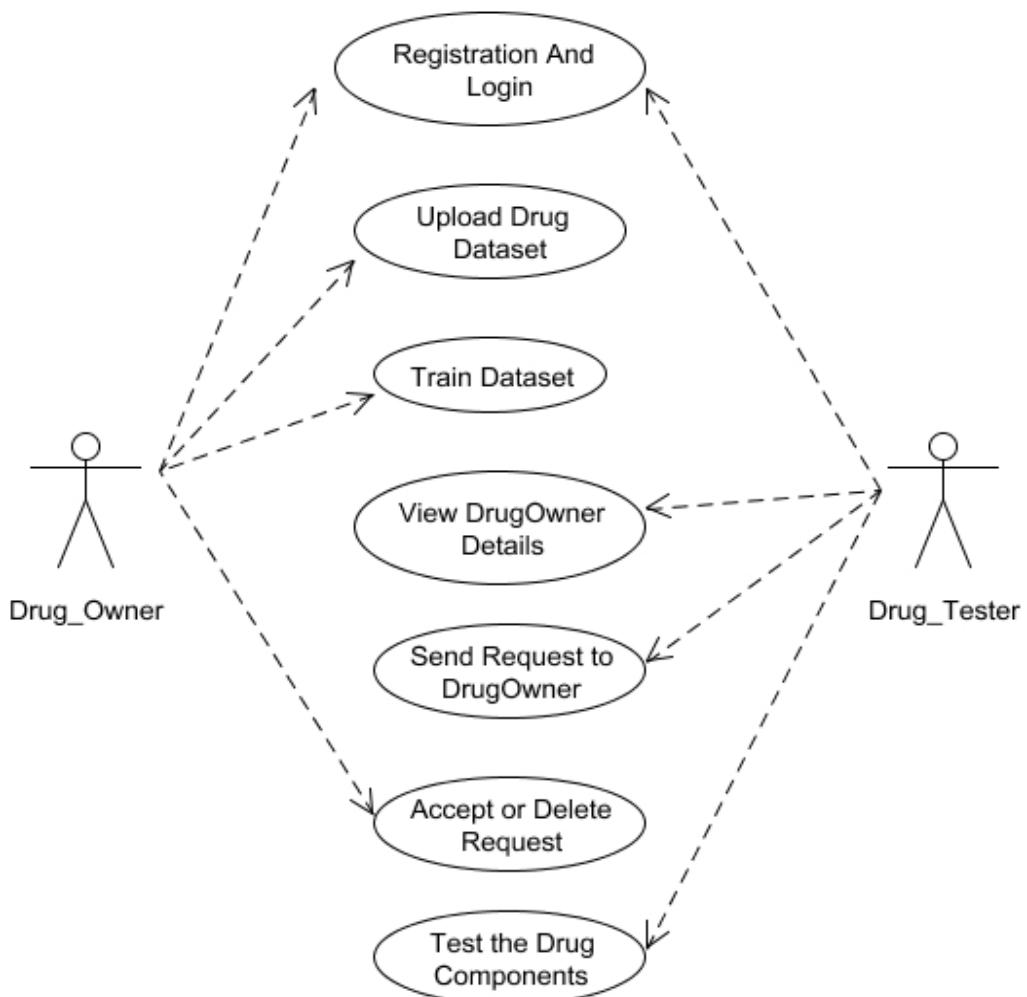


Figure 4.6 Use Case diagram for securing drug components

4.5.2 SEQUENCE DIAGRAM

A sequence diagram is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development.

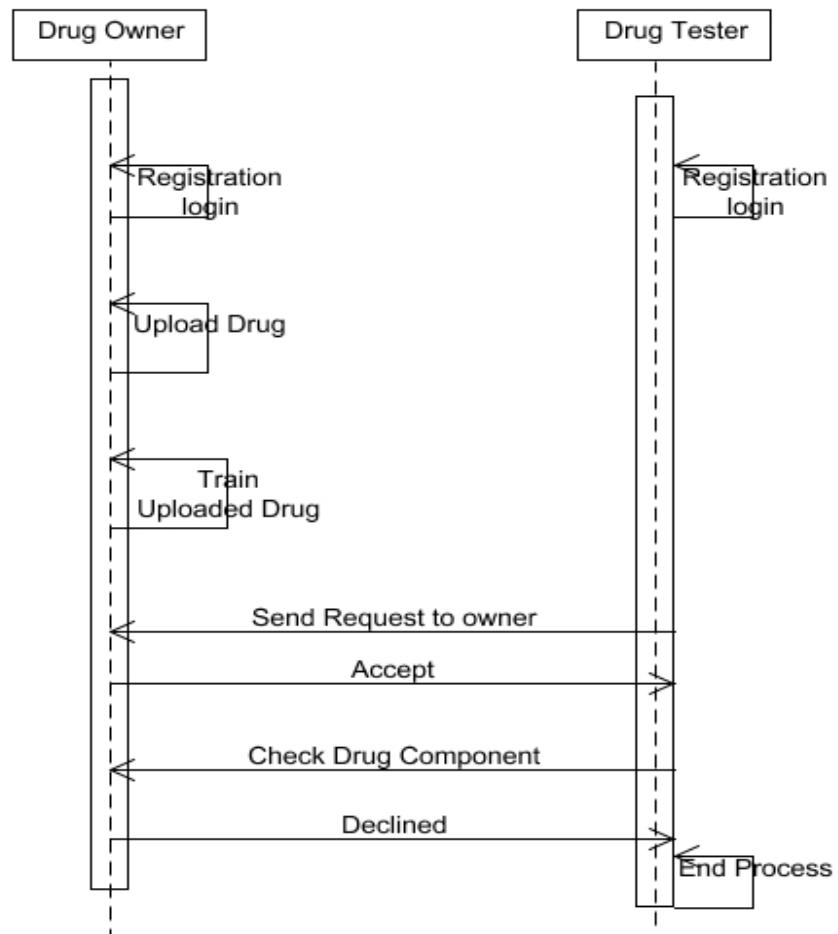


Figure 4.7 Sequence diagram for securing drug components

4.5.3 COLLABORATION DIAGRAM

Collaboration diagrams offer a better view of a scenario than a Sequence diagram when the modeler is trying to understand all of the effects on a given object and are therefore good for procedural design. Objects are connected by links, each link representing an instance of an association between the respective classes involved. The link shows messages sent between the objects, and the type of message passed.

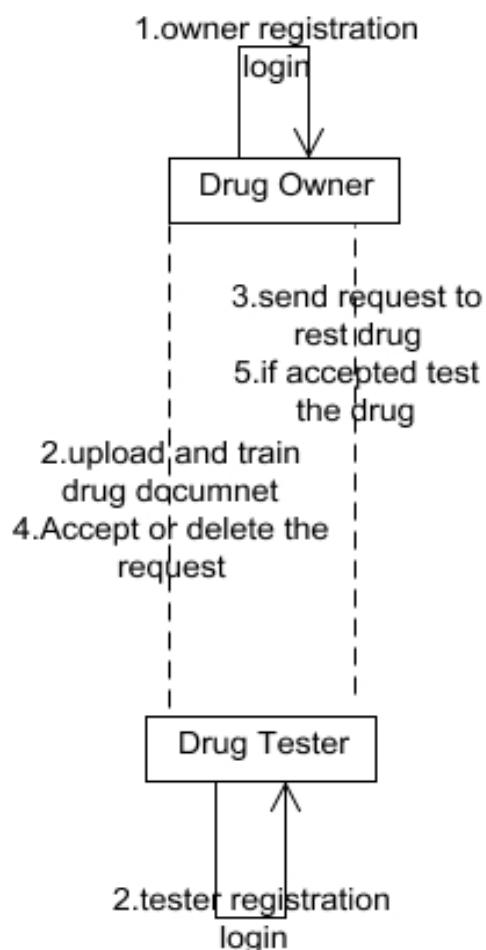


Figure 4.8 Collaboration diagram for securing drug components

4.5.4 ACTIVITY DIAGRAM

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.



Figure 4.9 Activity diagram for securing drug components

4.5.5 CLASS DIAGRAM

Class Diagram provides an overview of the target system by describing the objects and classes inside the system and the relationships between them. It provides a wide variety of usages; from modeling the domain-specific data structure to detailed design of the target system. With the share model facilities, you can reuse your class model in the interaction diagram for modeling the detailed design of the dynamic behavior. The Form Diagram allows you to generate diagram automatically with user-defined scope.

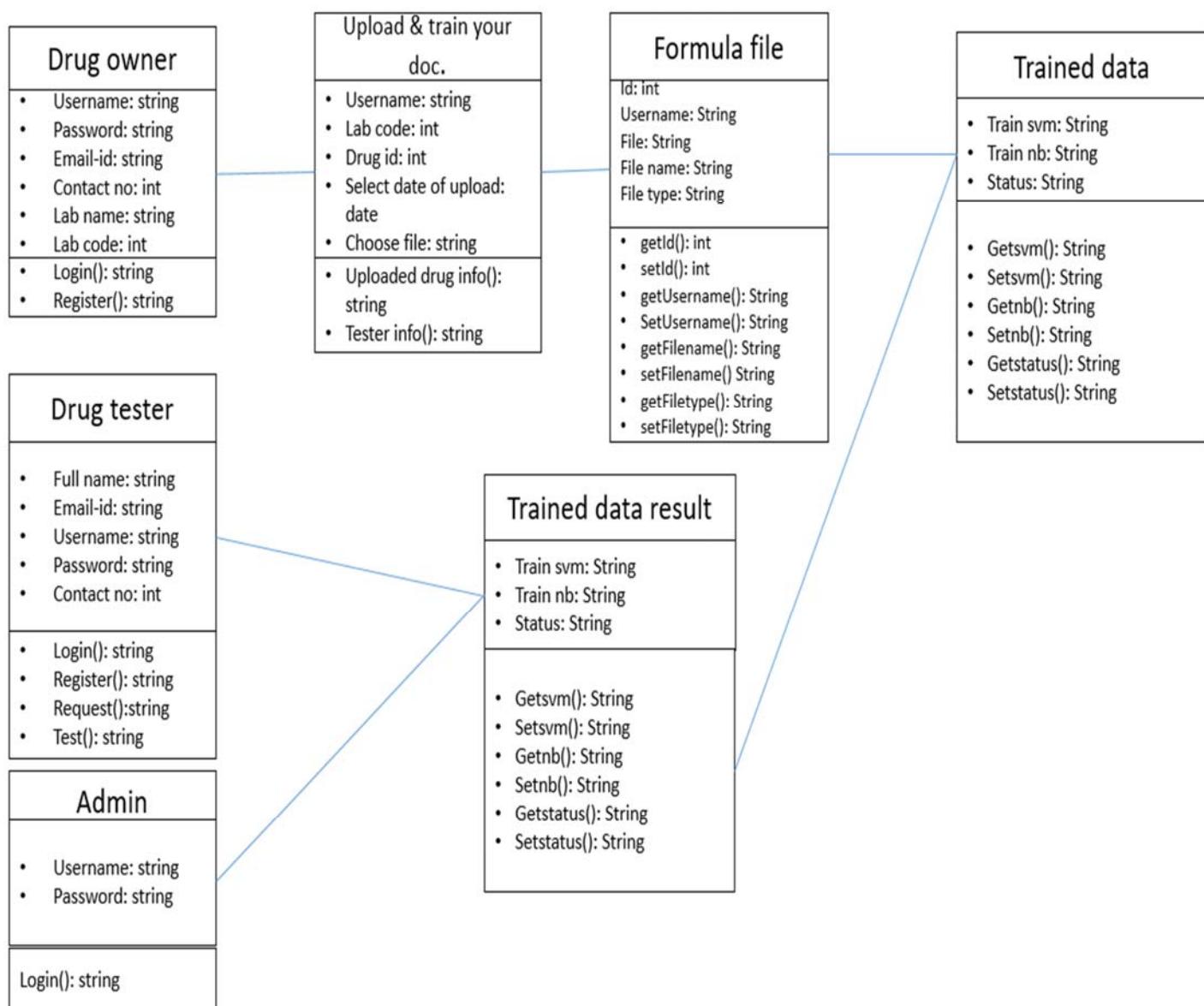


Figure 4.10 Class Diagram for securing drug components

SYSTEM ARCHITECTURE

CHAPTER 5

SYSTEM ARCHITECTURE

5.1 ARCHITECTURE OVERVIEW

LIST OF MODULES

- Drug Owner & Tester Registration
- Drug Component Uploading
- Train dataset
- Drug Testing

This system secure drug components for Secure Drug discovery in the cloud environment. Unlike drug discovery frameworks, the secure drug discovery seeks to achieve it efficiently. We are not using three real time datasets to check the efficiency of potential new drug component. Instead of using existing datasets, we are using another one data-mining algorithm Naïve Bayes (NB). These two algorithms are used to train the uploaded drug dataset (CSV file). In final, we will get trained data and accuracy for that uploaded dataset. Drug tester will check that new drug component. Drug tester does not know the contents of that file; they will get the trained data only. Then they let us know the file was active or not. Finally, admin will approve the drug component.

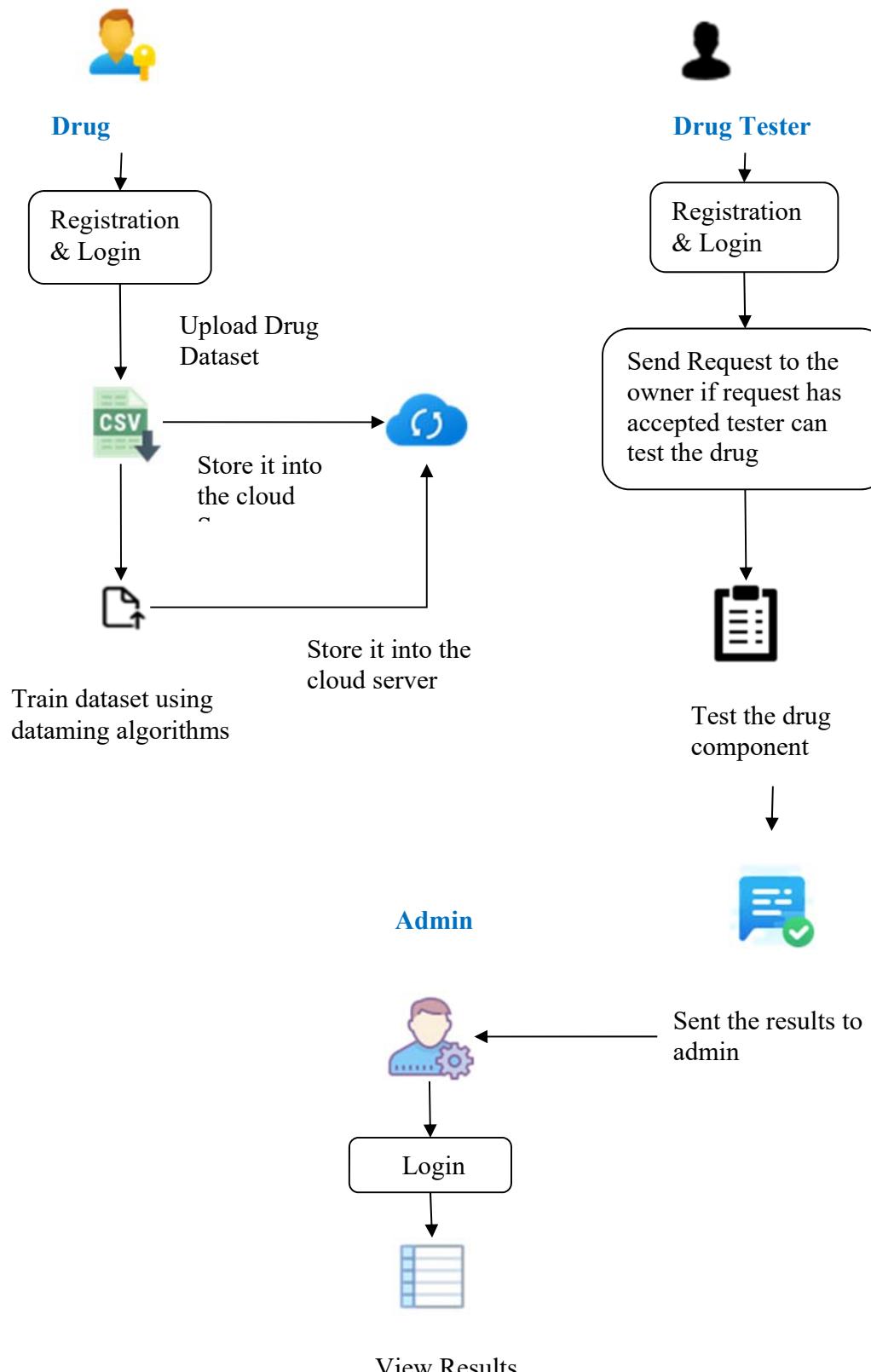


Figure 5.1 System Architecture for secure drug component system

5.2 MODULE DESIGN SPECIFICATION

5.2.1 DRUG OWNER AND TESTER REGISTRATION

Drug Owner will register in the service provider platform. MySQL database is used to store the drugs Meta details. This drug owner registration process will involves few entities such as 1.Name(drug owner's)2.Email(drug owner's)3.Contact number(drug owner's)4.lab name(where the drugs has been discovered)5.lab code(unique number of that lab) for this process.

The drug tester will also be registered in the drug tester registration platform. MySQL database is used to store the drugs Meta details. This drug tester registration process will involves few entities such as 1.Name(drug tester's)2.Email(drug tester's)3.Contact number(drug tester's) and 4.TesterID (unique number of that tester) for this process.

5.2.2 DRUG COMPONENT UPLOADING

Once the Registration is completed, the drug owner should upload the drug component. For uploading the drug component, the owner must provide the Drug Name, Drug Id and Date of Uploading. Now we have to choose the file that contains the drug data sets and drug components. That data set contains the formula and we have to mention the type of class (Class A, Class B). While uploading the file we will read the content and store into the database and store that .csv file in cloud.

5.2.3 TRAIN DATASET

As the drug owner has successfully uploaded the drug components, now we will train the uploaded data using python. For this part, we will use two algorithms, SVM and Naïve Bayes. The trained data and accuracy will be sent to the owner from python server.

5.2.4 DRUG TESTING

As the drug dataset has been trained and uploaded in the cloud, the drug tester now send request to the owner for testing. Only after accepting the request, the test can test the uploaded drug components. Once the Testing is completed, the drug tester sends result to the admin. If that particular drug component is retained in the cloud then it is assumed that component is still active and passed the testing successfully. If not, then the drug component is removed from the cloud.

5.3 PROGRAM DESIGN LANGUAGE

5.3.1 SVM (Support Vector Machine)

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane. SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.

```
class SVM: def fit(self, X, y):  
    n_samples, n_features = X.shape# P = X^T X  
    K = np.zeros((n_samples, n_samples))  
    for i in range(n_samples):  
        for j in range(n_samples):  
            K[i,j] = np.dot(X[i], X[j])P =  
cvxopt.matrix(np.outer(y, y) * K)# q = -1 (1xN)  
    q = cvxopt.matrix(np.ones(n_samples) * -1)# A = y^T  
    A = cvxopt.matrix(y, (1, n_samples))# b = 0  
    b = cvxopt.matrix(0.0)# -1 (NxN)  
    G = cvxopt.matrix(np.diag(np.ones(n_samples) * -1))# 0 (1xN)  
    h = cvxopt.matrix(np.zeros(n_samples))solution =  
cvxopt.solvers.qp(P, q, G, h, A, b)# Lagrange multipliers
```

```

        a = np.ravel(solution['x'])# Lagrange have non zero lagrange
multipliers
        sv = a > 1e-5
        ind = np.arange(len(a))[sv]
        self.a = a[sv]
        self.sv = X[sv]
        self.sv_y = y[sv]# Intercept
        self.b = 0
        for n in range(len(self.a)):
            self.b += self.sv_y[n]
            self.b -= np.sum(self.a * self.sv_y * K[ind[n], sv])
        self.b /= len(self.a)# Weights
        self.w = np.zeros(n_features)
        for n in range(len(self.a)):
            self.w += self.a[n] * self.sv_y[n] * self.sv[n]

def project(self, X):
    return np.dot(X, self.w) + self.b

def predict(self, X):
    return np.sign(self.project(X))

```

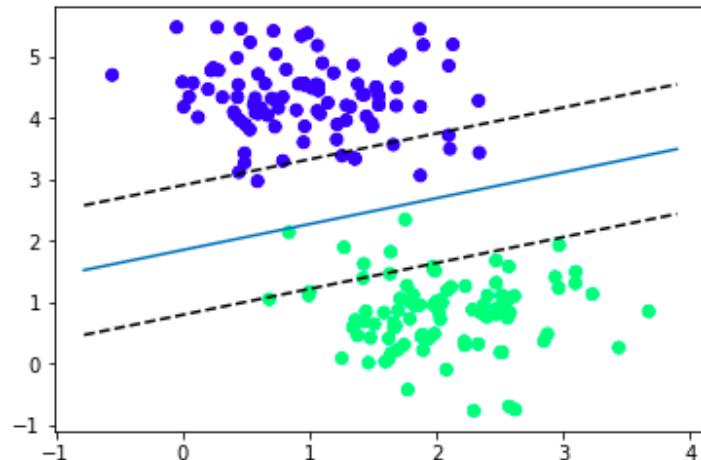


Figure 5.2 2D Diagram of SVM algorithm

5.3.2 Naïve Bayes

It is a classification technique based on Bayes' Theorem with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. For example, a fruit may be considered to be an apple if it is red, round, and about 3 inches in diameter. Even if these features depend on each other or upon the existence of the other features, all of these properties independently contribute to the probability that this fruit is an apple and that is why it is known as 'Naive'.

Bayes theorem:

$$P(c | x) = \frac{P(x | c)P(c)}{P(x)}$$

↑ ↑
Likelihood Class Prior Probability
↓ ↓
Posterior Probability Predictor Prior Probability

$$P(c | \mathbf{x}) = P(x_1 | c) \times P(x_2 | c) \times \dots \times P(x_n | c) \times P(c)$$

Let's understand it using an example. Below we have a training data set of weather and corresponding target variable 'Play' (suggesting possibilities of playing). Now, we need to classify whether players will play or not based on weather condition.

Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Rainy	No
Sunny	Yes
Rainy	Yes
Sunny	No
Overcast	Yes
Overcast	Yes
Rainy	No

Frequency Table		
Weather	No	Yes
Overcast		4
Rainy	3	2
Sunny	2	3
Grand Total	5	9

Likelihood table			
Weather	No	Yes	
Overcast		4	=4/14 0.29
Rainy	3	2	=5/14 0.36
Sunny	2	3	=5/14 0.36
All	5	9	
	=5/14	=9/14	
	0.36	0.64	

Problem: Players will play if weather is sunny. Is this statement is correct?

We can solve it using above discussed method of posterior probability.

$$P(\text{Yes} | \text{Sunny}) = P(\text{Sunny} | \text{Yes}) * P(\text{Yes}) / P(\text{Sunny})$$

Here we have $P(\text{Sunny} | \text{Yes}) = 3/9 = 0.33$, $P(\text{Sunny}) = 5/14 = 0.36$, $P(\text{Yes}) = 9/14 = 0.64$

Now, $P(\text{Yes} | \text{Sunny}) = 0.33 * 0.64 / 0.36 = 0.60$, which has higher probability.

Naive Bayes uses a similar method to predict the probability of different class based on various attributes. This algorithm is mostly used in text classification and with problems having multiple classes.

SYSTEM IMPLEMENTATION

CHAPTER 6

SYSTEM IMPLEMENTATION

6.1 CLIENT-SIDE CODING

```
import org.springframework.stereotype.Controller;
import org.springframework.validation.BindingResult;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.servlet.ModelAndView;
import com.dug.dao.OwnerDao;
import com.dug.model.OwnerModel;
import com.dug.model.testerModel;

@Controller
public class TesterController {
    @Autowired
    private OwnerDao dao;

    @RequestMapping(value = "/testerInfo/testerView/{testernname}/{username}",
method = RequestMethod.GET)
    public ModelAndView tester(@PathVariable("testernname") String testernname,
                               @PathVariable("username") String username) {
        ModelAndView model = new ModelAndView();
        List<testerModel> tester = dao.getTester(testernname);
        List<testerModel> list = dao.getTester();
        List<OwnerModel> owner = dao.findbyid(username);
        if (tester != null) {
            model.addObject("msg", tester);
            model.addObject("owner", owner);
            model.setViewName("testerslist");
        } else {
            model.addObject("msg1", list);
            model.setViewName("testerPage");
        }
        return model;
    }
}
```

```

    @RequestMapping(value
"/testerInfo/testerView/{testernname}/accept/{username}/{testernname}",method
RequestMethod.GET)

    public ModelAndView acceptRequest(@PathVariable("username") String
username,
        @PathVariable("testernname") String testernname) {
    ModelAndView model = new ModelAndView();
    String status="Accepted";
    int count=dao.updateOwner(username, status);
    int counter=dao.updateTester(testernname, status);

    if(count>0) {
        model.setViewName("Success");

    }
    return model;
}

    @RequestMapping(value
"/testerInfo/testerView/{testernname}/delete/{username}/{testernname}",method
RequestMethod.GET)

    public ModelAndView deleteRequest(@PathVariable("username") String
username,
        @PathVariable("testernname") String testername) {
    ModelAndView model = new ModelAndView();
    String status="No action Taken";
    String status1="No action";
    int count=dao.updateOwner(username, status);
    int counter=dao.updateTester(testernname, status1);

    if(count>0) {
        model.setViewName("Awesome");

    }
    return model;
}

```

```

package com.drug.controller;
import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.util.ArrayList;
import java.util.List;
import org.apache.http.HttpResponse;
import org.apache.http.NameValuePair;
import org.apache.http.client.HttpClient;
import org.apache.http.client.entity.UrlEncodedFormEntity;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.impl.client.DefaultHttpClient;
import org.apache.http.message.BasicNameValuePair;
@SuppressWarnings("deprecation")
public class PythonCall {

    @SuppressWarnings("deprecation")
    public List<String> executeMultiPartRequest(String urlString, String path)
throws Exception {

    String output = null;
    List<String> respons = new ArrayList<String>();
    HttpClient client = new DefaultHttpClient();
    HttpPost postRequest = new HttpPost(urlString);
    ArrayList<NameValuePair> postParameters;
    try {

        System.out.println("=====path=====" + path);

        // Send request

        postParameters = new ArrayList<NameValuePair>();
        postParameters.add(new BasicNameValuePair("path", path));
        postParameters.add(new BasicNameValuePair("courseid",
// courseid));
// postParameters.add(new BasicNameValuePair("activityid", actid));

        postRequest.setEntity(new
UrlEncodedFormEntity(postParameters, "UTF-8"));
        HttpResponse response = client.execute(postRequest);
}

```

```

        System.out.println("-Response----"
+ response.getStatusLine().getStatusCode());
        // Verify response if any
        if (response != null) {

            BufferedReader br = new BufferedReader(new
InputStreamReader((response.getEntity().getContent())));
// Getting

            System.out.println("Output from Server .... \n");
            while ((output = br.readLine()) != null) {
                respons.add(output);
                System.out.println(output); // Instead of this, you could
append all your response to a StringBuffer
// and use
`toString()` to get the entire JSON response as a String.
                // This string json response can be parsed using any
json library. Eg. GSON from
                // Google.

            br.close();

            return respons;
        }
    } else {
        return null;
    }
} catch (Exception ex) {
    return null;
}
return respons;

}
}

```

6.2 SERVER-SIDE CODING

```
package com.drugTest.controller;
import java.io.InputStream;
import java.util.List;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.servlet.ModelAndView;

import com.drugTest.dao.testerDao;
import com.drugTest.model.FormulaFile;
import com.drugTest.model.OwnerModel;
import com.drugTest.model.Requestmodel;
import com.drugTest.model.Traineddata;

@Controller
public class DrugOwnerController {
    @Autowired
    private testerDao dao;

    @RequestMapping(value = "/ownerDetails/{username}/{username1}/{status}",
method = RequestMethod.GET)
    public ModelAndView owner(@PathVariable("username") String username,
@PathVariable("username1") String username1,
        @PathVariable("status") String status) {

        ModelAndView model = new ModelAndView();
        List<OwnerModel> owner = dao.getOwnerList(username);
        List<OwnerModel> list = dao.getOwnerList();
        if (owner != null) {
            model.addObject("msg", owner);
            model.addObject("username", username1);
            model.addObject("status", status);
            model.setViewName("ownerDetails");
        } else {
            model.addObject("msg1", list);
            model.setViewName("ownerPage");
        }
    }
}
```

```

        return model;
    }

    @RequestMapping(value = "/fileDetails/{username}/{status}", method =
RequestMethod.GET)
    public ModelAndView file(@PathVariable("username") String username,
@PathVariable("status") String status) {
        ModelAndView model = new ModelAndView();
        OwnerModel owner = new OwnerModel();
        owner.setUsername(username);
        owner.setTesternname(status);

        String result = dao.testerstatus(owner);

        System.out.println("----result----" + result);

        List<FormulaFile> file = dao.getFileList(username);
        List<OwnerModel> list = dao.getOwnerList();
        if (file != null) {
            model.addObject("msg", file);
            model.addObject("result", result);
            model.setViewName("fileDetails");
        } else {
            model.addObject("msg1", list);
            model.setViewName("ownerPage");
        }
        return model;
    }

    @RequestMapping(value =
"ownerDetails/{username}/{username1}/request/{username1}/{username}", method =
RequestMethod.GET)
    public ModelAndView request(@PathVariable("username1") String testernname,
                               @PathVariable("username") String username) {
        ModelAndView model = new ModelAndView();
        String status = "Request pending";

        Requestmodel request = new Requestmodel();
        request.setUsername(username);
        request.setTesternname(testernname);
        request.setStatus(status);

```

```

// OwnerModel owner=new OwnerModel();
// owner.setUsername(username);
// owner.setStatus(status);
// owner.setTesternname(testernname);

int result = dao.uodateStatus(username, status, testernname);
int count = dao.updateStatus(testernname, status);

System.out.println("-----" + count);

int counter = dao.saveRequest(request);
if (count > 0) {
    model.addObject("msg", "success");
    model.setViewName("Success");

} else {
    model.addObject("msg", "failed");
    model.setViewName("ownerPage");

}

return model;
}

@RequestMapping(value
"fileDetails/{username}/testFile/{username}/{drugname}",      method      =
RequestMethod.GET)
public      ModelAndView      testing(ModelAndView      model,
@PathVariable("drugname") String drugname,
@PathVariable("username") String username) {
String status="checked";
String statuss="Active";

List<Traineddata> dataList=dao.getDataList(username);
List<FormulaFile> fileList=dao.getFileList(drugname);
for (FormulaFile formulaFile : fileList) {
for (Traineddata traineddata : dataList) {
    String dnb=formulaFile.getTrain_nb();
    String tnb=traineddata.getTrain_NB();
}
}
}

```

```

        String dvm=formulaFile.getTrain_svm();
        String tvm=traineddata.getTrain_SVM();
        if(dataList.isEmpty()) {
            model.addObject("msg","Cannot Find the Data");
            model.setViewName("fileDetails");

        }else if (dnb!=null || dvm!=null || tnb!=null || tvm!=null) {

            int result=dao.formulaStatus(drugname, status);
            int count=dao.dataStatus(drugname, statuss);
            System.out.println("-----");

            model.addObject("msg","success");
            model.setViewName("checked");

        }

    }

    return model;
}

}

package com.gts.controller;
import java.util.List;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.servlet.ModelAndView;
import com.gts.dao.AdminDao;
import com.gts.model.Traineddata;

@Controller
public class AdminController {
    @Autowired

```

```

private AdminDao dao;
@RequestMapping(value = "/adminlogin",method = RequestMethod.POST)
public ModelAndView admin(ModelAndView model,
@RequestParam("username") String username,
    @RequestParam("password") String password) {
    System.out.println("===="+username+"===="+password);

    if(username.equalsIgnoreCase("admin")&&password.equalsIgnoreCase("admin"))
    {
        List<Traineddata> list=dao.getAllData();
        model.addObject("msg", list);
        model.setViewName("dashboard");
    }else {
        model.setViewName("index");
    }
    return model;
}

@RequestMapping(value      =      "/accept/{drugname}",method      =
RequestMethod.GET)
public ModelAndView OK(ModelAndView
model,@PathVariable("drugname")String drugname) {
    String status="Accepted";
    int result=dao.accept(drugname, status);
    int ok=dao.ok(drugname, status);
    model.setViewName("dashboard");
    return model;
}

@RequestMapping(value      =      "/decline/{drugname}",method      =
RequestMethod.GET)
public ModelAndView NO(ModelAndView
model,@PathVariable("drugname")String drugname) {
    String status="Declined";
    int result=dao.decline(drugname, status);
    int notok=dao.notok(drugname, status);
    model.setViewName("dashboard");
    return model;
}

```

```
}
```

```
package com.gts.model;
public class FormulaFile {
    private int id;
    private String username;
    private String labcode;
    private String drugname;
    private String drugid;
    private String date;
    private String filename;
    private String file;
    private String filetype;
    private String train_svm;
    private String train_nb;
    private String status;
```

```
    public String getStatus() {
        return status;
    }
```

```
    public void setStatus(String status) {
        this.status = status;
    }
```

```
    public String getTrain_svm() {
        return train_svm;
    }
```

```
    public void setTrain_svm(String train_svm) {
        this.train_svm = train_svm;
    }
```

```
    public String getTrain_nb() {
        return train_nb;
    }
```

```
public void setTrain_nb(String train_nb) {
    this.train_nb = train_nb;
}

public FormulaFile() {

}

public String getDrugid() {
    return drugid;
}

public void setDrugid(String drugid) {
    this.drugid = drugid;
}

public int getId() {
    return id;
}

public void setId(int id) {
    this.id = id;
}

public String getUsername() {
    return username;
}

public void setUsername(String username) {
    this.username = username;
}

public String getLabcode() {
    return labcode;
}
```

```
public void setLabcode(String labcode) {  
    this.labcode = labcode;  
}  
  
public String getDrugname() {  
    return drugname;  
}  
  
public void setDrugname(String drugname) {  
    this.drugname = drugname;  
}  
  
public String getDate() {  
    return date;  
}  
  
public void setDate(String date) {  
    this.date = date;  
}  
  
public String getFilename() {  
    return filename;  
}  
  
public void setFilename(String filename) {  
    this.filename = filename;  
}  
  
public String getFile() {  
    return file;  
}  
  
public void setFile(String file) {  
    this.file = file;  
}  
  
public String getFiletype() {  
    return filetype;  
}
```

```

public void setFiletype(String filetype) {
    this.filetype = filetype;
}

}

package com.gts.model;
public class Traineddata {
    private int id;
    private String username;
    private String drugname;
    private String filename;
    private String accuracy_nb;
    private String accuracy_svm;
    private String Train_SVM;
    private String Train_NB;
    private String status;

    public String getStatus() {
        return status;
    }
    public void setStatus(String status) {
        this.status = status;
    }
    public int getId() {
        return id;
    }
    public void setId(int id) {
        this.id = id;
    }
    public String getUsername() {
        return username;
    }
    public void setUsername(String username) {
        this.username = username;
    }
    public String getDrugname() {
        return drugname;
    }
    public void setDrugname(String drugname) {
        this.drugname = drugname;
    }
}

```

```

    }
    public String getFilename() {
        return filename;
    }
    public void setFilename(String filename) {
        this.filename = filename;
    }

    public String getAccuracy_nb() {
        return accuracy_nb;
    }
    public void setAccuracy_nb(String accuracy_nb) {
        this.accuracy_nb = accuracy_nb;
    }

    public String getAccuracy_svm() {
        return accuracy_svm;
    }
    public void setAccuracy_svm(String accuracy_svm) {
        this.accuracy_svm = accuracy_svm;
    }
    public String getTrain_SVM() {
        return Train_SVM;
    }
    public void setTrain_SVM(String train_SVM) {
        Train_SVM = train_SVM;
    }
    public String getTrain_NB() {
        return Train_NB;
    }
    public void setTrain_NB(String train_NB) {
        Train_NB = train_NB;
    }
}

package com.drugTest.model;

public class Requestmodel {
    private String testername;
    private String username;
}

```

```
private String status;

public Requestmodel() {

}

public String getTestername() {
    return testername;
}

public void setTestername(String testername) {
    this.testername = testername;
}

public String getUsername() {
    return username;
}

public void setUsername(String username) {
    this.username = username;
}

public String getStatus() {
    return status;
}

public void setStatus(String status) {
    this.status = status;
}

}
```

SYSTEM TESTING

CHAPTER 7

SYSTEM TESTING

7.1 UNIT TESTING

Unit Testing is a level of the software testing process where individual units/components of a software/system are tested. The purpose is to validate that each unit of the software performs as designed. The goal of unit testing is to isolate each part of the program and show that the individual parts are correct. In our application, the modules are taken separately and tested. In the module of registration each input is denoted with specify constraint and the violation of this constraint will display error message. Thus, it verify the unit testing. Likewise, the other modules are tested separately to test their functionality. The developer usually performs unit testing.

7.2 INTEGRATION TESTING

Integration testing is a systematic technique for construction the program structure while at the same time conducting tests to uncover errors associated with interfacing. i.e., integration testing is the complete testing of the set of modules, which makes up the product. The objective is to take untested modules and build a program structure tester should identify critical modules. Critical modules should be tested as early as possible. One approach is to wait until all the units have passed testing, and then combine them and then tested. This approach is evolved from unstructured testing of small programs. Another strategy is to construct the product in increments of tested units. A small set of modules are integrated together and tested, to which another module is added and tested in combination. And so on. The advantages of this approach are that, interface dispenses can be easily found and corrected.

The major error that was faced during the project is linking error. When all the modules are combined, the link is not set properly with all support files. Then we checked out for interconnection and the links. Errors are localized to the new module and its intercommunications. The product development can be staged, and modules integrated

in as they complete unit testing. Testing is completed when the last module is integrated and tested.

7.3 TEST CASES & REPORTS

Testing is a process of executing a program with the intent of finding an error. A good test case is one that has a high probability of finding an as-yet –undiscovered error. A successful test is one that uncovers an as-yet- undiscovered error. System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently as expected before live operation commences. It verifies that the whole set of programs hang together. System testing requires a test consists of several key activities and steps for run program, string, system and is important in adopting a successful new system. This is the last chance to detect and correct errors before the system is installed for user acceptance testing. The software testing process commences once the program is created and the documentation and related data structures are designed. Software testing is essential for correcting errors. Otherwise the program or the project is not said to be complete. Software testing is the critical element of software quality assurance and represents the ultimate the review of specification design and coding. Testing is the process of executing the program with the intent of finding the error.

TEST CASE DESIGN

A test case is a detailed procedure that fully tests a feature or an aspect of a feature. Whereas the test plan describes what to test, a test case describes how to perform a particular test. You need to develop a test case for each test listed in the test plan. The below table shows the possible Inputs and their corresponding Expected and Obtained output for the given modules with their status.

Table 7.1: Test Case Design

**TABLE 7.1.1 DRUG OWNER LOGIN TEST REPORT
MODULE : DRUG OWNER AND TESTER REGISTRATION
SCREEN: (2) LOGIN PAGE FOR DRUG OWNER**

S.NO	ACTION	INPUT	EXPECTED OUTPUT	ACTUAL OUTPUT	TEST RESULT
1.	Enter the <u>username,password</u> for registration.	Username:XXX Password: ***	XXX ***	XXX ***	passed
2.	Compare username and password with registered field	Username:XXX Password: ***	It Redirects to drug component uploading page.	It redirects to drug component uploading page.	passed
3.	Compare username and password with registered field	Username:XXy Password: ***	Invalid username and password.	Invalid username and password	passed

**TABLE 7.1.2 DRUG OWNER REGISTRATION TEST REPORT
MODULE: DRUG OWNER AND TESTER REGISTRATION
SCREEN: (3) REGISTRATION PAGE FOR DRUG OWNER**

S.NO	ACTION	INPUT	EXPECTED OUTPUT	ACTUAL OUTPUT	TEST RESULT
1.	Drug Owner should provide name, mail id ,contact no,Lab Name and Lab Id.	Name:Jayalakshmi Mail id: jay@gmail.com Contact no:9876543210 Lab Name:ABC Lab ID:1234	These details will be stored in drug Owner database.	These details will be stored in drug owner database.	Passed.
2.	Drug Owner should provide name, mail id ,contact no,Lab Name and Lab Id.	Name:Jayalakshmi Mail id: jay@gmail.com Contact no:9876543210 Lab Name:ABC Lab ID:1234	Please match the requested format for Mail Id.	Please match the requested format for Mail Id.	passed
3.	Drug Owner should provide name, mail id ,contact no,Lab Name and Lab Id.	Name:Jayalakshmi Mail id: jay@gmail.com Contact no:98765432 Lab Name:ABC Lab ID:1234	Please match the requested format for Contact number.	Please match the requested format for Contact number.	passed

TABLE 7.1.3 UPLOADING DRUG COMPONENT TEST REPORT
MODULE : DRUG COMPONENT UPLOADING
SCREEN: (4) DRUG FILE UPLOADING PAGE

S.NO	ACTION	INPUT	EXPECTED OUTPUT	ACTUAL OUTPUT	TEST RESULT
1.	Drug Owner should provide drug name, drug id, date of upload and drug component file.	Drug Name:XYZ Drug ID:12345 Date of Upload:MM-DD-YYYY File:drug1.csv	These details will be stored in drug Owner database.	These details will be stored in drug owner database.	Passed.
2.	Drug Owner should provide drug name, drug id , date of upload and drug component file.	Drug Name:Zincovit Drug ID:1234 Date of Upload:22-06-2020 File:drug1.csv	Please match the requested format for Date of upload.	Please match the requested format for Date of upload.	passed
3.	Drug Owner should provide drug name, drug id , date of upload and drug component file.	Drug Name:Zincovit Drug ID:1234 Date of Upload:22-06-2020 File:drug1.csv	Please match the requested format for File upload.	Please match the requested format for file upload.	Passed.

TABLE 7.1.4 DRUG TESTER REGISTRATION TEST REPORT
MODULE : DRUG OWNER AND TESTER REGISTRATION
SCREEN: (8) REGISTRATION PAGE FOR DRUG TESTER

S.NO	ACTION	INPUT	EXPECTED OUTPUT	ACTUAL OUTPUT	TEST RESULT
1.	Drug Tester should provide name, mail id ,contact no and Tester Id.	Name:Saiveena Mail id: saiveena@gmail.com Contact no:9876543210 Tester ID:1234	These details will be stored in drug Owner database.	These details will be stored in drug owner database.	Passed.
2.	Drug Owner should provide name, mail id ,contact no and Tester Id.	Name:Saiveena Mail id: saiveenagmailcom Contact no:9876543210 Tester ID:1234	Please match the requested format for Mail Id.	Please match the requested format for Mail Id.	passed
3.	Drug Owner should provide name, mail id ,contact no and Tester Id.	Name:Saiveena Mail id: saiveena@gmail.com Contact no:98765432 Tester ID:1234	Please match the requested format for Contact number.	Please match the requested format for Contact number.	passed

CONCLUSION

CHAPTER 8

CONCLUSION

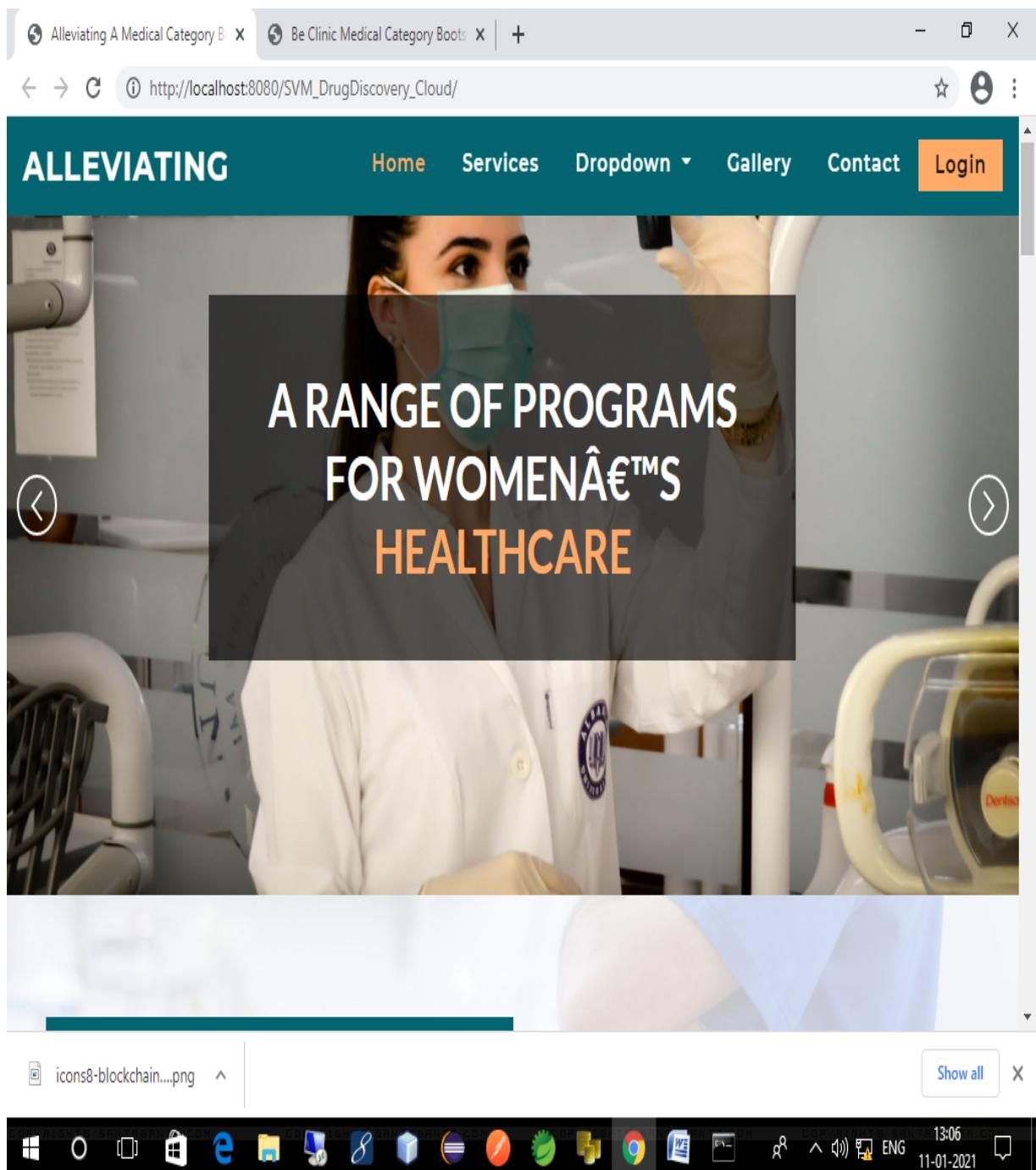
8.1 CONCLUSION AND FUTURE ENHANCEMENTS

This project focuses on securing the drug components in the cloud. Privacy is a major factor in drug discovery as it involves significant investment and high commercial values. Drug discovery is a long term expensive process. Bringing the drugs from the bench to the market involves a lot of threat when stored in the cloud. Thus, we proposed securing the drug components for drug discovery in the cloud. Securing the drug components is designed to facilitate drug manufacturers to securely outsource their formulas to the cloud for storage and SVM and NB training. These two algorithms such as SVM and NB are used to train the uploaded drug dataset (CSV file). As a result, we receive trained data and accuracy for that uploaded dataset. The trained model could be used for authorized client's compound classification in a privacy-preserving way. Specifically, we designed a secure domain transformation protocol and several basic secure computation components for secure outsourced computation across different parties. We also built two key secure components (i.e. secure parameter selection and secure sequential minimal optimization) to achieve privacy-preserving SVM and NB training in drug discovery. We will be extending our approach to support more sophisticated data mining method in order to support very large dataset in drug discovery.

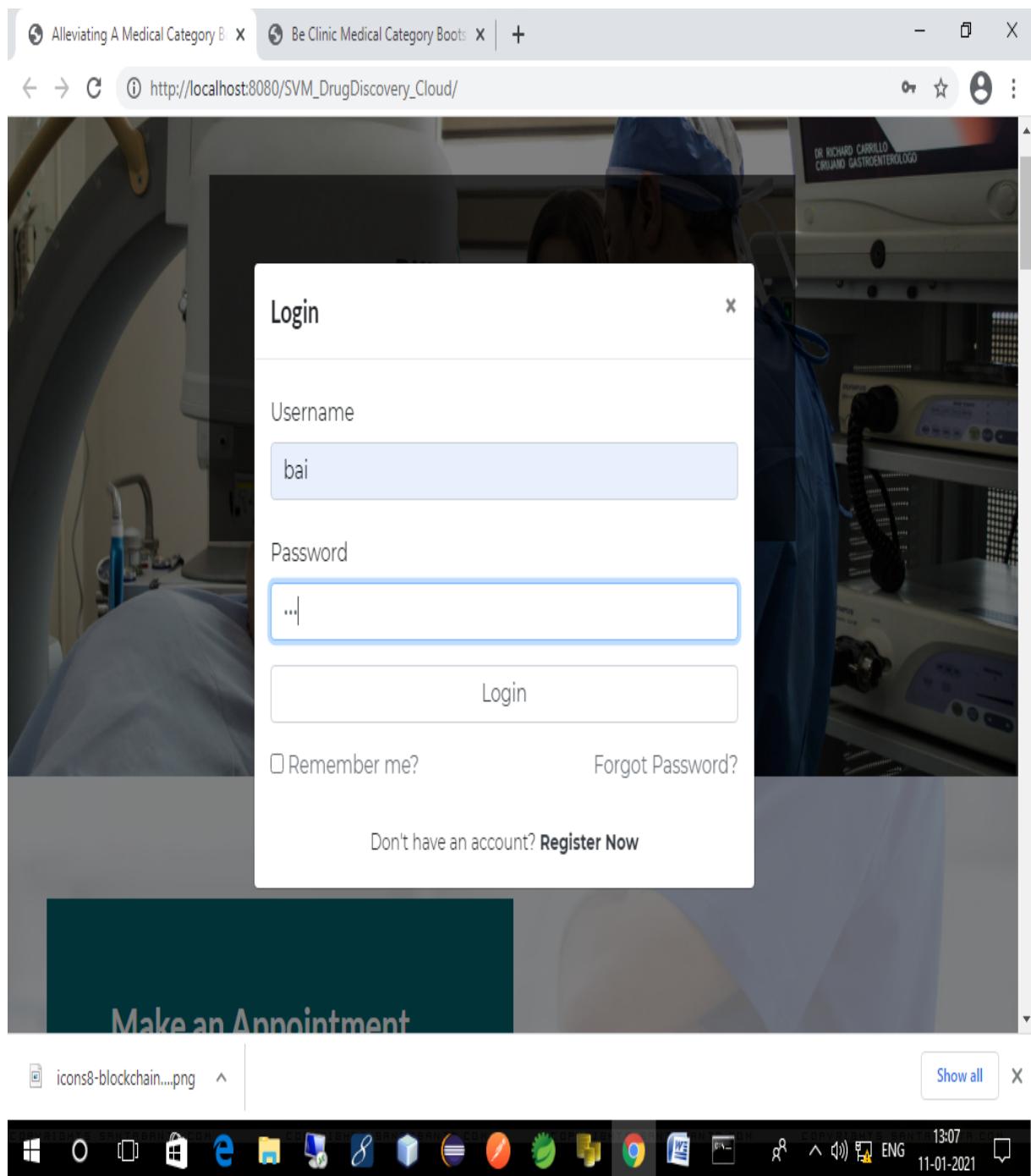
APPENDICES

A.1 SAMPLE SCREENS

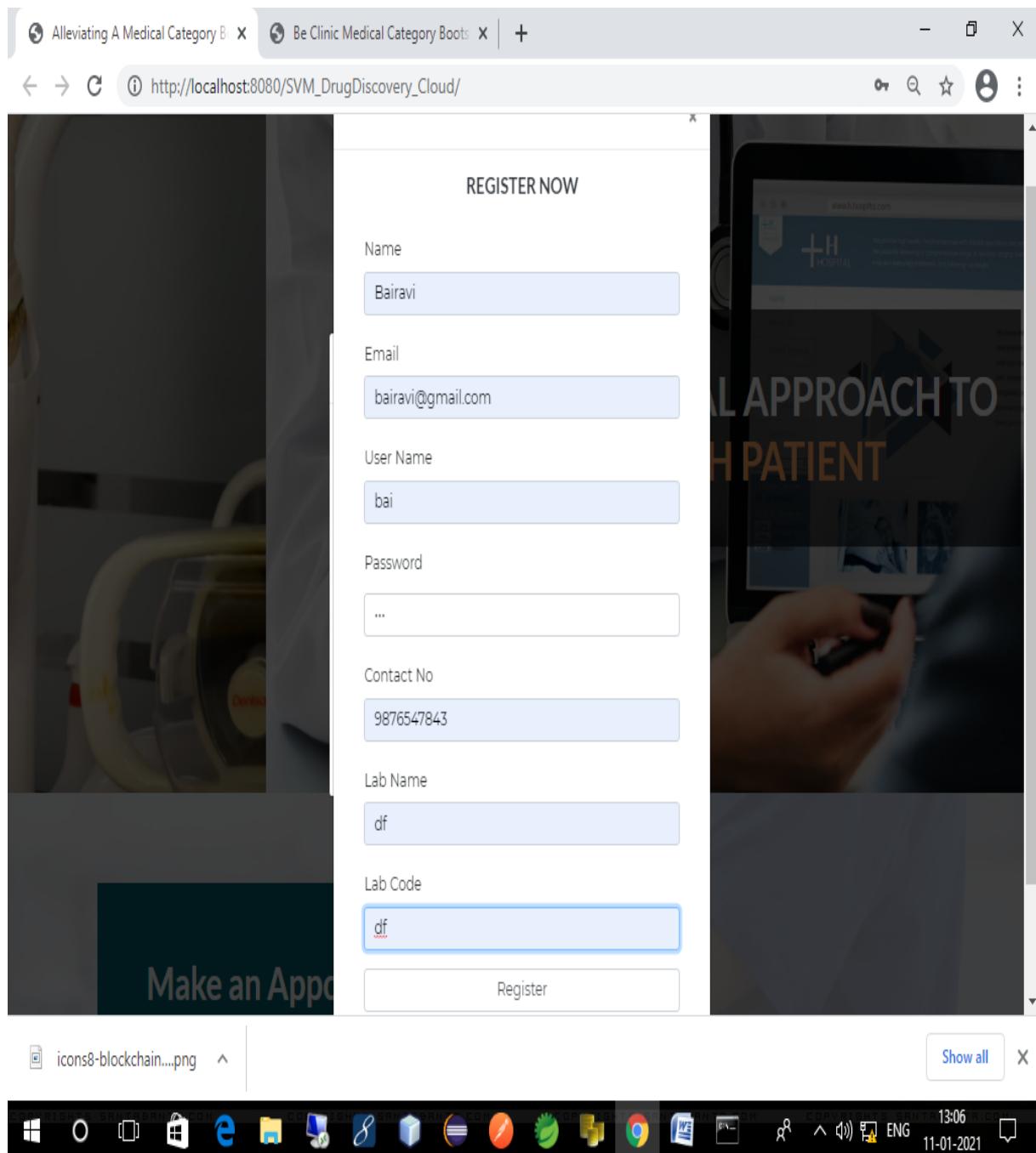
(1) HOME PAGE FOR DRUG OWNER



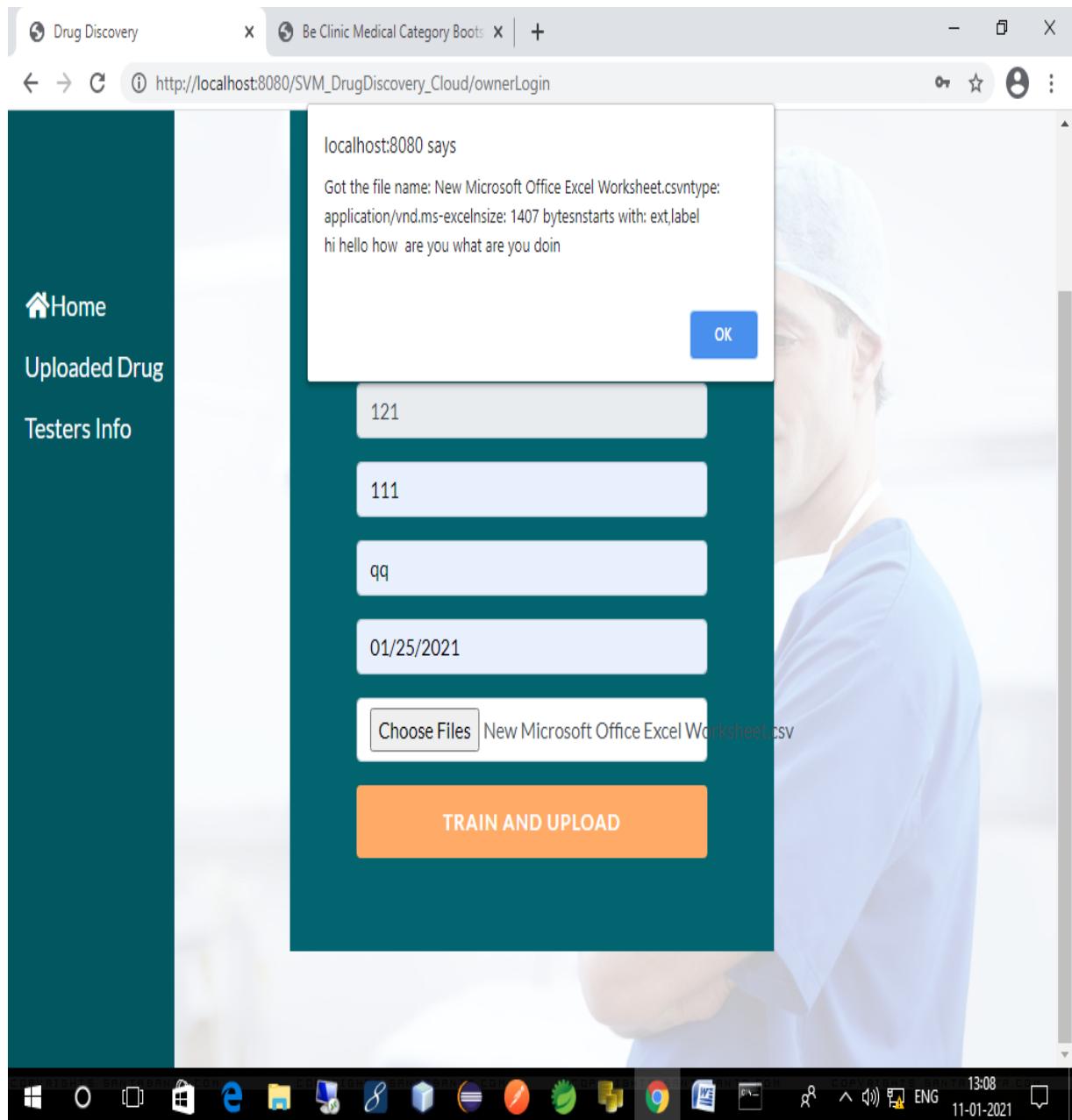
(2) LOGIN PAGE FOR DRUG OWNER



(3) REGISTRATION PAGE FOR DRUG OWNER



(4) DRUG FILE UPLOADING PAGE



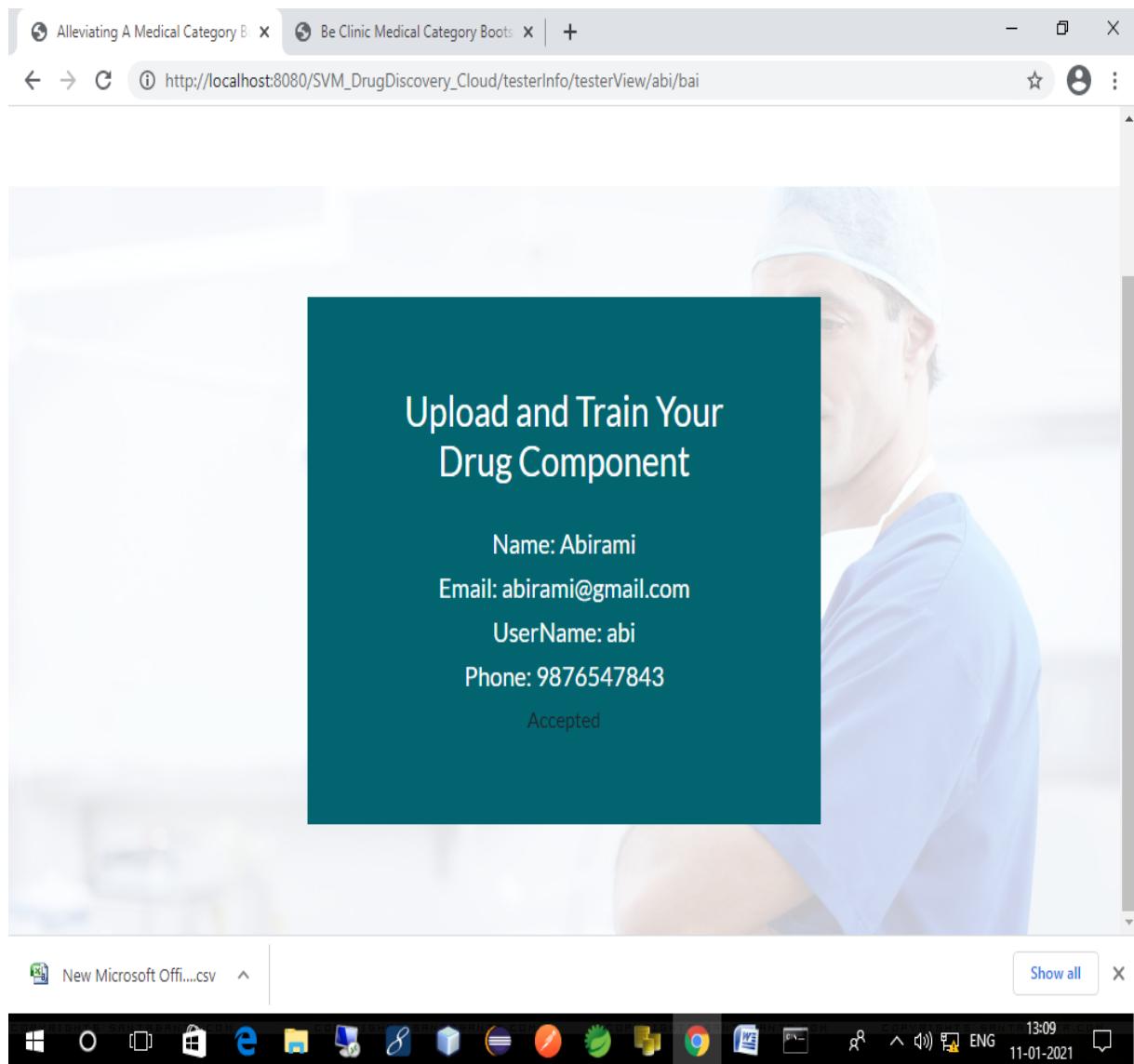
(5) ACCURACY DETECTION

The screenshot shows a web browser window with two tabs open: "Alleviating A Medical Category B" and "Be Clinic Medical Category Boots". The main content area displays the following table:

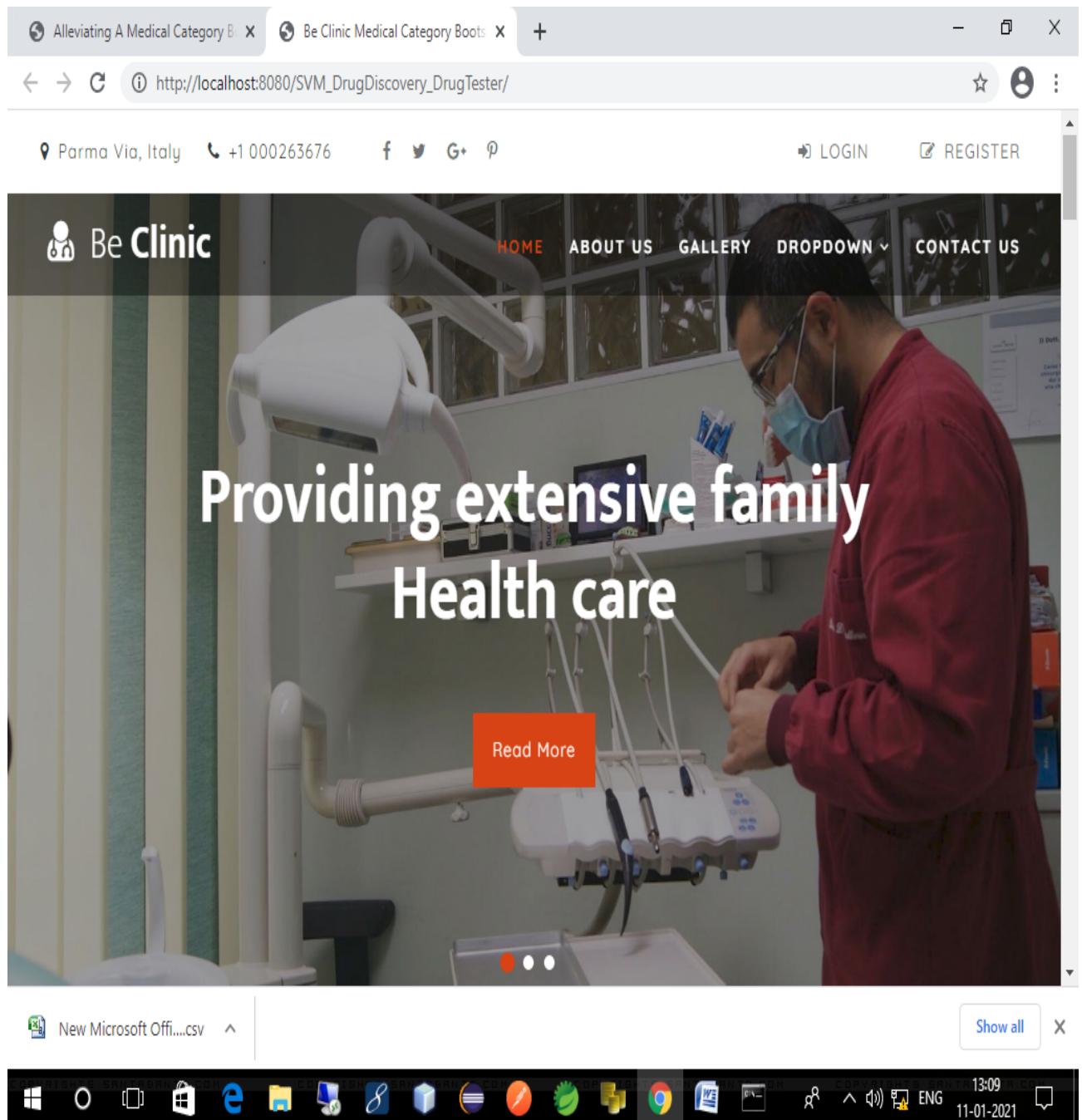
DrugName	FileName	NB Accuracy	SVM Accuracy	Status
111	New Microsoft Office Excel Worksheet.csv	53.84615384615385	53.84615384615385	

The browser's address bar shows the URL http://localhost:8080/SVM_DrugDiscovery_Cloud/findFile/bai. The system tray at the bottom right shows the date and time as 11-01-2021 13:08.

(6) STATUS PAGE



(7) HOME PAGE FOR DRUG TESTER



(8) REGISTRATION PAGE FOR DRUG TESTER

REGISTER NOW

Full Name
Abirami

Email
abirami@gmail.com

User Name
abi

Password

Contact
9876547843

REGISTER

By clicking Register, I agree to your terms

New Microsoft Offi....csv Show all X

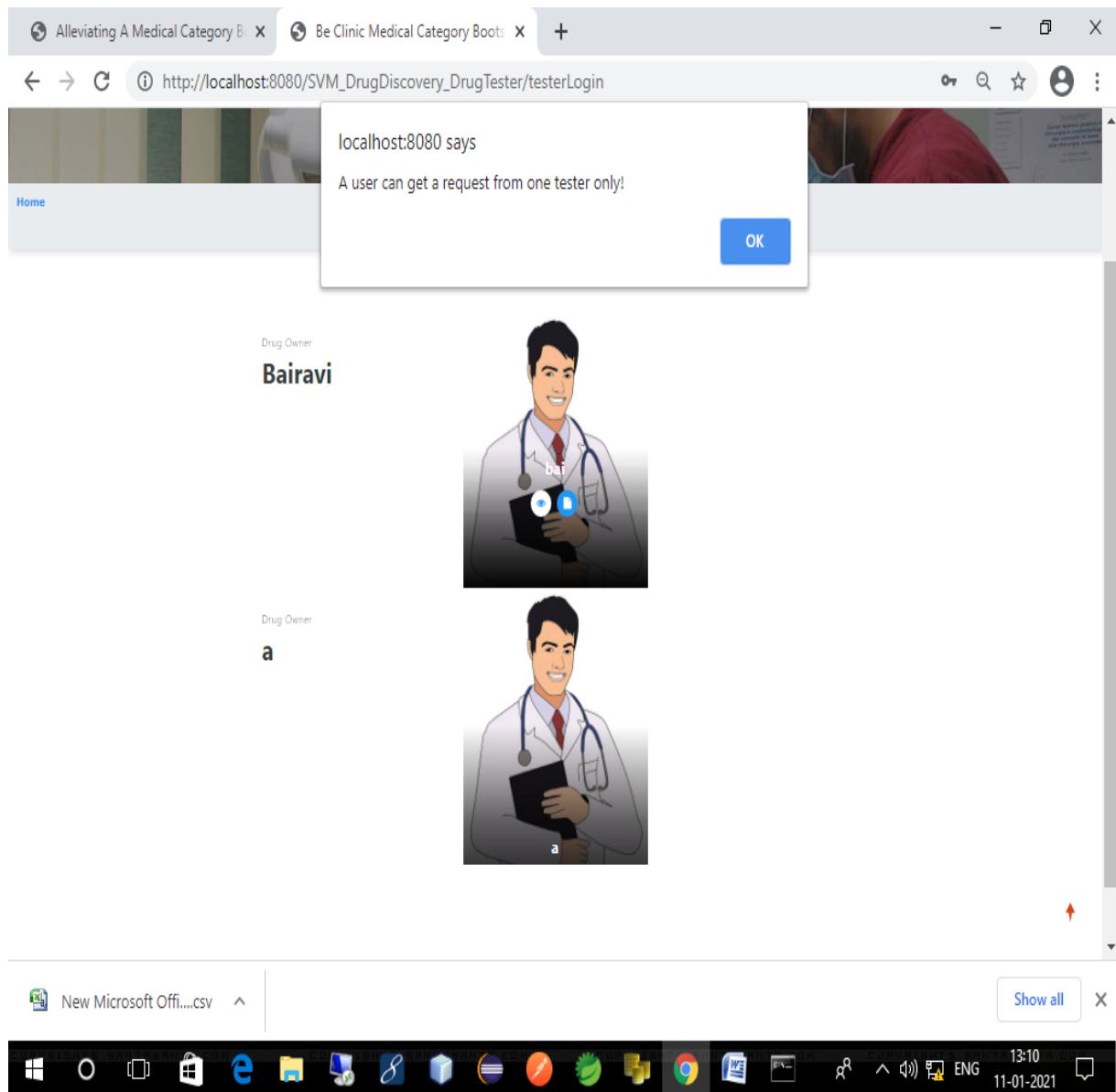
13:09 ENG 11-01-2021

(9) LOGIN PAGE FOR DRUG TESTER

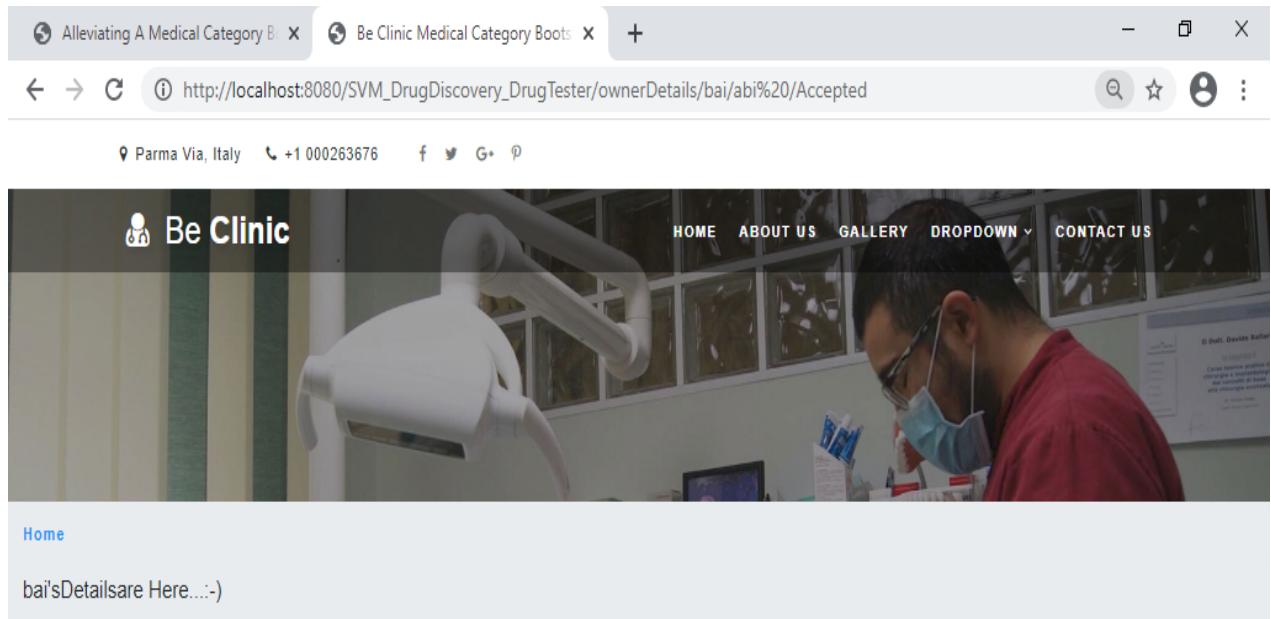
The screenshot shows a web browser window with the following details:

- Address Bar:** http://localhost:8080/SVM_DrugDiscovery_DrugTester/login.jsp
- Content Area:**
 - Header:** LOGIN NOW
 - User Name:** abi
 - Password:** ***
 - Login Button:** LOGIN
 - Link:** Don't have an Account?
- Background Image:** A medical-themed image featuring a doctor's hands on a keyboard and a stethoscope.
- Bottom Navigation:** Shows the Windows taskbar with various pinned icons (File Explorer, Edge, File Manager, etc.) and system status indicators (13:10, ENG, 11-01-2021).

(10) REQUEST PAGE

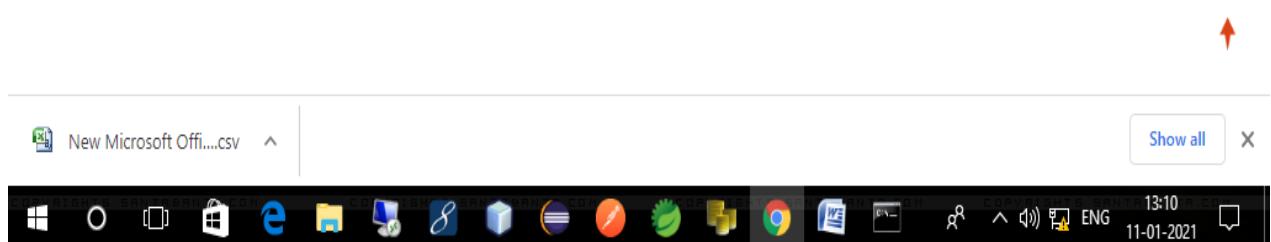


(11) STATUS MODULE



The screenshot shows a web browser window with two tabs: "Alleviating A Medical Category" and "Be Clinic Medical Category Boots". The active tab displays the URL http://localhost:8080/SVM_DrugDiscovery_DrugTester/ownerDetails/bai/abi%20/Accepted. The page content includes a header with a doctor icon and the text "Be Clinic", followed by navigation links: HOME, ABOUT US, GALLERY, DROPODOWN ▾, and CONTACT US. Below the header is a large image of a dental procedure. The main content area contains the text "Home" and "bai'sDetailsare Here...:-)".

Name	Email	UserName	LabName	LabCode	Action	Status
Bairavi	bairavi@gmail.com	bai	ASC	121	Request Accepted	You can Test the drug component



(12) TESTING PAGE

The screenshot shows a web browser window with two tabs open. The active tab is titled "Be Clinic Medical Category Boots" and displays a testing page for a file named "bai/abi". The page includes a header with a logo, navigation links (HOME, ABOUT US, GALLERY, DROPODOWN, CONTACT US), and social media links. Below the header is a large image of a dental or medical procedure. The main content area shows "Uploaded File Details" and a table with one row of data:

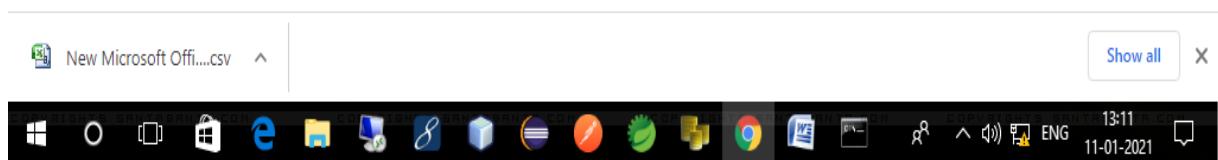
UserName	Drug_id	Drug_name	File_name	Actions
bai	qq	111	New Microsoft Office Excel Worksheet.csv	Test

At the bottom of the browser window, there is a taskbar with various icons and system status information.

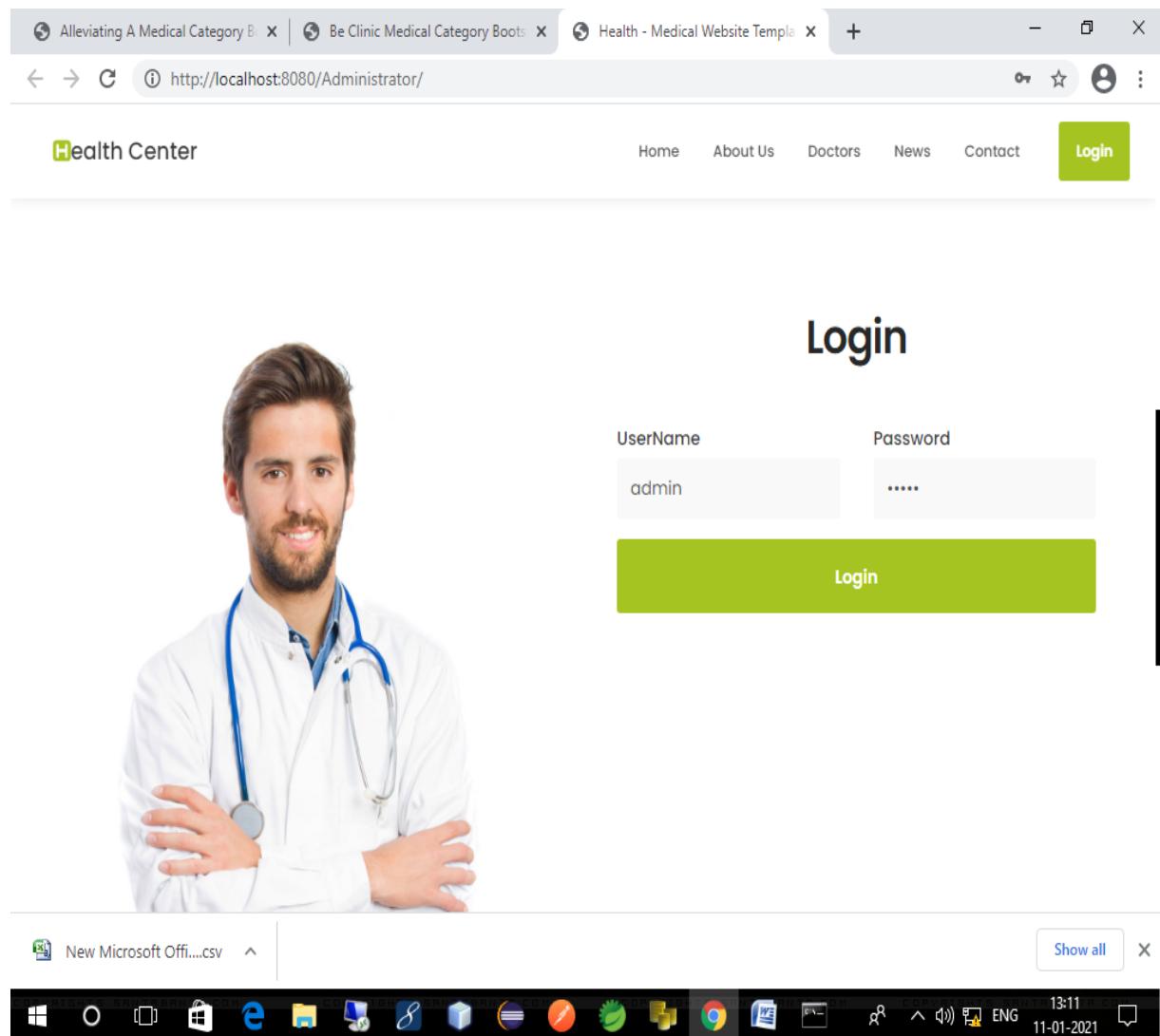
(13) DRUG TESTED STATUS PAGE

The screenshot shows a web browser window with two tabs open: "Alleviating A Medical Category" and "Be Clinic Medical Category". The active tab is "Be Clinic Medical Category". The URL in the address bar is http://localhost:8080/SVM_DrugDiscovery_DrugTester/fileDetails/bai/abi. The page content includes a header with "Be Clinic" logo, navigation links for HOME, ABOUT US, GALLERY, DROPODOWN, and CONTACT US, and a main image of a medical professional in a red shirt and mask. Below the image, there are links for "Home" and "Uploaded File Details". A table titled "Uploaded File Details" lists one item:

UserName	Drug_id	Drug_name	File_name	Actions
bai	qq	111	New Microsoft Office Excel Worksheet.csv	Checked



(14) LOGIN PAGE FOR ADMIN



(15) ADMIN APPROVAL PAGE

The screenshot shows a web browser window with three tabs open: "Alleviating A Medical Category B", "Be Clinic Medical Category Boots", and "Health - Medical Website Template". The current page is "http://localhost:8080/Administrator/adminlogin". The header includes a phone number (010-060-0160), operating hours (6:00 AM - 10:00 PM Mon-Fri), and an email address (info@company.com). The main content area is titled "Accepted Drug Components Details" and contains a table:

DrugOwner	Drug Name	NB Accuracy	SVM Accuracy	Conclusion
bai	111	53.84615384615385	53.84615384615385	Approve Decline

At the bottom, there is a taskbar showing various icons and system status: New Microsoft Offi...csv, Show all, Windows Start button, Task View, File Explorer, Edge, File Manager, Taskbar settings, Google Chrome, File Explorer, Taskbar settings, 13:12, ENG, 11-01-2021.

(16) STATUS MODULE

The screenshot shows a web browser window with three tabs open:

- Alleviating A Medical Category B
- Be Clinic Medical Category Boots
- Health - Medical Website Templa

The address bar shows the URL: http://localhost:8080/SVM_DrugDiscovery_Cloud/findFile/bai.

The main content area displays the "ALLEVIATING" logo and a user icon with the word "Home".

A sidebar on the left contains links for "Home" and "Testers Info".

The main content area has a table titled "Uploaded Drug" with the following columns: DrugName, FileName, NB Accuracy, SVM Accuracy, and Status.

	DrugName	FileName	NB Accuracy	SVM Accuracy	Status
111	New Microsoft Office Excel Worksheet.csv		53.84615384615385	53.84615384615385	Update your details into Clinic DB

The taskbar at the bottom shows various icons and system status information, including the date (11-01-2021), time (13:12), and language (ENG).

(17) INCLUDING DRUG DETAILS PAGE

Drug Discovery Be Clinic Medical Category Boots Health - Medical Website Templa

http://localhost:8080/SVM_DrugDiscovery_Cloud/findFile/fullDetails/FullDrugDetails

ALLEVIATING

User Actions Home

Home
Uploaded Drug
Testers Info

Drug Details

Details added successfully

111

Ingredients

Strength

Dosage Form

UPLOAD DETAILS

11-01-2021 13:12

A.2 PUBLICATIONS

Lavanya T , Bhagya sree C R , Vinmathi M S , Dr Kavitha Subramani ,“Secure the drug components using Naïve Bayes and SVM”, International Journal of Creative Research Thoughts (IJCRT) , ISSN : 2320-2882, 2021

REFERENCES

REFERENCES

- [1] Dibyendu Dana , Satishkumar V. Gadhiya , Luce G. St. Surin , David Li , Farha Naaz ,Quaisar Ali,“Deep Learning in Drug Discovery and Medicine; Scratching the Surface”, The British journal of pharmacology, vol. 162, no. 6, pp. 1239–1249, 2018
- [2] Yaman Akbulut 1 ID , Abdulkadir , Sengür 1,* ID , Yanhui Guo 2 and Florentin Smarandache, “A Novel Neutrosophic Weighted Extreme Learning Machine for Imbalanced Data Set ”,Journal of chemical information and computer sciences, vol. 44, no. 5, pp. 1630–1638, 2017
- [3] Richard C. Mohsa, Nigel H. Greig, “Drug discovery and development: Role of basic biological research,” IEEE Journal of Biomedical and Health Informatics, vol. 20, pp. 655 – 668, 2017
- [4] Aniket Sharma, “Needle Free Drug Delivery Devices Market Size, Industry Outlook and Opportunity Analysis Report 2018-2025 ”
- [5] Shu- Feng Zhou , Wei-Zhu Zhong,” Drug Design and Discovery: Principles and Applications ”, Drug discovery today, vol. 17, no. 19, pp. 1088–1102,2017.
- [6] Y.Rahulamathavan, S. Veluru, R. C.-W. Phan, J.A. Chambers, and M. Rajarajan, “Privacy-preserving clinical decision support system using Gaussian kernel-based Classification,” Biomedical and Health Informatics, IEEE Journal of, vol.18, no. 1, pp.56-66,2015.

- [7] R. Bost, R. A. Popa, S. Tu, and S. Goldwasser, “Machine learning classification over encrypted data,” in 22nd Annual Network and Distributed System Security Symposium, NDSS 2015, San Diego, California, USA, February 8-11, 2015.
- [8] Y. Rahulamathavan, R. C.-W. Phan, S. Veluru, K. Cumanan, and M. Rajarajan, “Privacy-preserving multi-class support vector machine for outsourcing the data classification in cloud,” IEEE on Dependable and Secure Computing, vol. 11, no. 5, pp. 467–479, 2014.
- [9] J. B. Mitchell, “Machine learning methods in chemoinformatics,” Wiley Interdisciplinary Reviews: Computational Molecular Science, . 4, no. 5, pp. 468–481, 2014.
- [10] M. A. Lill and M. L. Danielson, “Computer-aided drug design platform using pymol,” Journal of computer-aided molecular design, vol. 25, no. 1, pp. 13–19, 2011.
- [11] G. Cano, J. Garcia-Rodriguez, A. Garcia-Garcia, H. Perez-Sanchez, J. A. Benediktsson, A. Thapa, and A. Barr, “Automatic selection of descriptors using random forest: Application to drug discovery,” Expert Systems with Applications, vol. 72, pp. 151–159,
- [12] Liu, K.-K. R. Choo, R. H. Deng, R. Lu, and J. Weng, “Efficient and privacy-preserving outsourced computation of rational numbers,” IEEE Journal of Biomedical and Health Informatics, vol. 20, pp. 655 – 668, 2016.

- [13] X. Liu, R. Choo, R. Deng, R. Lu, and J. Weng, “Efficient and privacy-preserving outsourced calculation of rational numbers,” IEEE Transactions on Dependable and Secure Computing, 2016.
- [14] B. K. Samanthula, H. Chun, and W. Jiang, “An efficient and probabilisticbit decomposition,” in Proceedings of the 8th ACM SIGSAC symposium on Information, computer and communications security. ACM, 2013.
- [15] X. Liu, R. H. Deng, K.-K. R. Choo, and J. Weng, “An efficient privacy preserving outsourced calculation toolkit with multiple keys,” IEEE Transactions Information Forensics and Security, vol. 11, no. 11, pp. 2401–2414, 2016.
- [16] J. Platt, “Sequential minimal optimization: A fast algorithm for training support vector machines,” 1998.
- [17] R. Todeschini and V. Consonni, Handbook of molecular descriptors. John Wiley & Sons, 2008, vol. 11.
- [18] S. Kamara, P. Mohassel, and M. Raykova, “Outsourcing multiparty ,” IACR Cryptology ePrint Archive, vol. 2011, p. , 2011. [Online]. Available: <http://eprint.iacr.org/2011/272>
- [19] A. Peter, E. Tews, and S. Katzenbeisser, “Efficiently outsourcing multiparty computation under multiple keys,” Information Forensics and Security, IEEE Transactions on, vol. 8, no. 12, pp. 2046–2058, 2013.

- [20] Y. Xue, Z.-R. Li, C. W. Yap, L. Z. Sun, X. Chen, and Y. Z. Chen, “Effect of molecular descriptor feature selection in support vector machine classification of pharmacokinetic and toxicological properties of chemical agents,” *Journal of chemical information and computer sciences*, vol. 44, no. 5, pp. 1630–1638, 2004.
- [21] D. E. Knuth, “Seminumerical algorithm (arithmetic) the art of computer programming vol. 2,” 1981.
- [22] H. Yu, J. Vaidya, and X. Jiang, “Privacy-preserving SVM classification on vertically partitioned data,” in *Advances in Knowledge and Data Mining, 10th Pacific-Asia Conference, PAKDD 2006, Singapore, April 9-12, 2006, Proceedings*, 2006, pp. 647–656.
- [23] H. Yu, X. Jiang, and J. Vaidya, “Privacy-preserving SVM using nonlinear kernels on horizontally partitioned data,” in *Proceedings of the 2006 ACM Symposium on Applied Computing (SAC), Dijon, France, April 23-27, 2006*, 2006, pp. 603–610.
- [24] J. Vaidya, H. Yu, and X. Jiang, “Privacy-preserving SVM classification,” *Knowl. Inf. Syst.*, vol. 14, no. 2, pp. 161–178, 2008.
- [25] T. Joachims, “Making large-scale svm learning practical,” *Technical Report,SFB 475: Komplexitätsreduktion in Multivariaten Datenstrukturen, Universitäat Dortmund, Tech. Rep.*, 1998.

[26] R. Burbidge, M. Trotter, B. Buxton, and S. Holden, “Drug design by machine learning: support vector machines for pharmaceutical data analysis,” *Computers & chemistry*, vol. 26, no. 1, pp. 5–14, 2001.