

Vaccination Tracking System Using ML Algorithm

Shivani Shinde¹, Anagha Jadhav², Manisha Gore³, Prof. Rohini S. Jadhav⁴

Students, Department of Computer Engineering^{1,2,3,4}

Smt. Kashibai Navale College Engineering, Pune, Maharashtra, India

Savitribai Phule Pune University, Pune, Maharashtra, India

Abstract: *Coronavirus disease 2019 (COVID-19) known to originate from Wuhan city in China in November 2019 and was declared a pandemic by the in January 2020 World Health Organization (WHO). COVID-19 is known to be a highly infectious virus. Infected individuals do not initially exhibit symptoms, while some remain asymptomatic. We proposed a system where we can get to know the vaccinated persons information and keep track on percent of peoples vaccinated. The system is build using php and Python with machine learning techniques. The proposed system uses machine learning techniques of Support Vector Machine (SVM) for getting the vaccine survey status.*

Keywords: Vaccination, Survey, COVID-19, Support Vector Machine

I. INTRODUCTION

Mankind has observed various pandemics throughout the history where some of them were disastrous .We are observing very tough time to fight with an invisible enemy the novel corona virus. So far, the infectious disease corona virus, which has killed millions of people all over the country and is contently taking people under its arrest. A coronaviruses family of virus that causes illness of respiratory disease. Digital tracking of health status and immunization is one which can be used to receive messages about the vaccinations that have to be taken. India is home to a populace of 1.36 billion, which is around 18 percent of the world's all-out populace. It is provoking task to inoculate masses at an incredibly enormous scope. Billions of individuals thirstily look for vaccination over the upcoming months. Poorly managed supplying systems will result in high or uncalled-for vaccine wastage rates, stock-outs, or improper management of waste, leading to crucial operational program prices and a negative impact on public health. Hence Vaccination Tracking play's a significant role

II. LITERATURE SURVEY

Due to the devastating outbreak of the global pandemic COVID19, gadgets enabled by the Internet of Things (IoT) are on the trend [6]. Early identification of contagious people is one of the key targets of all nations. For example, temperature detector smart thermometers are used. Detection of respiratory systems The WHOOP Wearable Wrist Health Monitoring System [7] is proposed by Central Queensland University Australia. They proposed a self-identified system that would collect patient data, but this method is specifically concerned with the analysis of improvements in the patient's respiratory system. In our proposed method, we primarily deal with more symptoms, such as heartbeat, temperature, and respiratory systems.

A similar device is also introduced for raspberry Pi flu sense [8] that uses Artificial intelligence to get the number of people in a room. After that, it compares coughing and other symptoms to diagnose flu, but it is lacking from other symptoms of COVID-19, e.g. respiratory system, etc. A lot of vaccine manufacturing laboratories work day and night to find the right formula for this crisis, but it's a time consuming operation. Vaccination needs time so that we can rely on other HI-TECH IT strategies to reduce the rate of transmission of the disease and save human lives as much as possible. With the rapid integration of artificial intelligence technology in epidemiology and health logistics, it is important to investigate how and what importance remote processing and computing technology can be reconfigured for the practice and field of global health [9].

Other governments of all countries are looking for devices that can collect reliable information, e.g. Smart thermometers [10] that can collect data sent to a server are being used to detect fever. Besides, AI-enabled fever detection cameras are also used but there is a shortage of literary devices that can cover the highest possible symptoms of COVID-19. Also, data from COVID-19 patients is collected around the globe and many datasets are available for analysis.

However, these datasets help obtain some reliable results after applying machine learning algorithms. To make an analysis, a cloud-based platform is needed that can be further extended with edge computing to perform an analysis efficiently. Edge computing [11] is very effective when the proposed system has to monitor contagious or quarantine people at a geographically different location.

To improve performance, edge-based analysis is carried out in the proposed model to achieve results efficiently. Google and APPLE are releasing Bluetooth-based applications [12]. But these kinds of systems will rely on Bluetooth, which could have problems with ranges. To resolve these concerns, we suggest GPRS-based identification and group alerting of suspicious individuals. In [13] the authors present an IoT based smart edge framework for remote health monitoring. The system has been implemented with some wearable vital sensors to transmit data to two new software engines called Rapid Active Summarization for effective prognosis (RASPRO) and Criticality Measurement Index (CMI). The system was reviewed with precision (0.87), recall (0.83), and F1-score (0.85). Alibaba Cloud has made accessible AI computing power for public research institutions to develop new drugs and vaccines. Hence cloud-based solutions are the fundamental building block for epidemic monitoring.

III. OBJECTIVES

- To study the current situation of pandemic caused by coronavirus.
- To guarantee all gets the vaccine as soon as possible.
- To create graph according to the data registered on portal.
- To keep track on the vaccinated person and non-vaccinated person.
- To maintain the data properly.

IV. SOFTWARE REQUIREMENT SPECIFICATION (SRS)

This software requirement specification (SRS) report expresses complete description about proposed System. This document includes all the functions and specifications with their explanations to solve related problems.

4.1 User Classes and Characteristics

- Basic knowledge of using computers is adequate to use this application.
- Knowledge of how to use a mouse or keyboard and internet browser is necessary.
- The user interface will be friendly enough to guide the user.

4.2 Assumptions and Dependencies

A. Assumptions

1. All the software such as python, etc are installed and running on the computers.
2. The cluster of nodes is formed and running.
3. Dependencies:
4. It is assumed that user know his/her tasks in organizations.

V. FUNCTIONAL REQUIREMENT

5.1 System Feature 1 (Functional Requirement)

- Functional requirements describe features, functioning, and usage of a product/system or software from the perspective of the product and its user.
- Functional requirements are also called as functional specifications were synonym for specification is design. Provide User friendly Interface and Interactive as per standards.

5.2 Non-Functional Requirement

5.2.1 Performance Requirements

A. High Speed

- System should process requested task in parallel for various action to give quick response. Then system must wait for process completion.

**B. Accuracy**

- System should correctly execute process, display the result accurately. System output should be in user required format.

5.2.2 Safety Requirements

- The data safety must be ensured by arranging for a secure and reliable transmission media. The source and destination information must be entered correctly to avoid any misuse or malfunctioning.
- Password generated by user is consisting of characters, special character number so that password is difficult to hack. So, that user account is safe.

5.2.3 Security Requirements

- Secure access of confidential data (user's details). Information security means protecting information and information systems from unauthorized access, use, disclosure, disruption, modification or destruction. The terms information security, computer security and information assurance are frequently incorrectly used interchangeably. These fields are interrelated often and share the common goals of protecting the confidentiality, integrity and availability of information; however, there are some subtle differences between them.

5.2.4 Software Quality Attributes

1. Runtime System Qualities: Runtime System Qualities can be measured as the system executes.
2. Functionality: The ability of the system to do the work for which it was intended.
3. Performance: The response time, utilization, and throughput behavior of the system. Not to be confused with human performance or system delivery time.
4. Security: A measure of system's ability to resist unauthorized attempts at usage or behavior modification, while still providing service to legitimate users.
5. Availability: (Reliability quality attributes falls under this category) the measure of time that the system is up and running correctly; the length of time between failures and the length of time needed to resume operation after a failure.
6. Usability: The ease of use and of training the end users of the system. Sub qualities: learn ability, efficiency, affect, helpfulness, control.
7. Interoperability: The ability of two or more systems to cooperate at runtime.

VI. SYSTEM REQUIREMENT**6.1 Software Requirements (Platform Choice)**

- Tools
 - Python IDE
- Programming Language
 - Python, PHP
- Software Version
 - Python 3.5

6.2 Hardware Requirements

- Processor
 - Pentium IV/Intel I3 core
- Speed
 - 1.1 GHz
- RAM
 - 512 MB (min)
- Hard Disk
 - 20GB



- Keyboard
 - Standard Keyboard
- Mouse
 - Two or Three Button Mouse
- Monitor
 - LED Monitor

VII. IMPLEMENTATION DETAILS OF MODULE

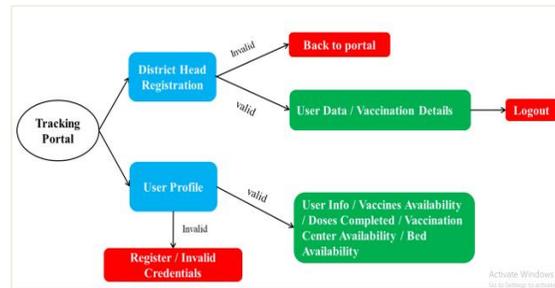


Figure: System Architecture

The proposed system undergoes some modules such as:

1. Preprocessing
2. Feature Extraction
3. Classification

7.1 Data preprocessing

It is a technique used in data mining that involves transforming raw data into an understandable format. The data is cleansed through processes such as filling in missing values, smoothing the noisy data, or resolving the inconsistencies in the data. As it contains some missing value, the dataset is cleaned, and decimal values are converted into proper float values

7.2 Data Splitting

The new dataset is split into two, training set and testing set. The splitting is done in an 80-20 ratio. 80% of the dataset is taken as the Training Set which is used to train the model. The remaining 20% becomes the Test Set which is used to test the model, to analyze its accuracy. The testing set is never used for training, which could otherwise lead to over fitting the mode

7.3 Feature Selection

The data features that used to train machine learning models have a huge influence on the performance of the model. Irrelevant or partially relevant features can negatively impact model performance

7.4 Classification

The model is trained by fitting the training set to the classifier model. The classifier model upon testing, classifies the air quality into good or bad. The classifications are fairly close to the testing set.

VIII. RISK MANAGEMENT

8.1 Overview of Risk Mitigation, Monitoring, Management

A. Risk Management Organizational Role

Each member of the organization will undertake risk management. The development team will consistently be monitoring their progress and project status as to identify present and future risks as quickly and accurately as possible. With this said, the members who are not directly involved with the implementation of the product will also need to keep

their eyes open for any possible risks that the development team did not spot. The responsibility of risk management falls on each member of the organization, while William Lord maintains this document.

B. Business Impact Risk

- Amount and quality of documentation that must be produced and delivered to customer the customer will be supplied with a complete online help file and users manual for Game Forge. Coincidentally, the customer will have access to all development documents for Game Forge, as the customer will also be grading the project
- • Governmental constraints in the construction of the product none known.
- Costs associated with late delivery Late delivery will prevent the customer from issuing a letter of acceptance for the product, which will result in an incomplete grade for the course for all members of the organization.
- Costs associated with a defective product Unknown at this time.

C. Customer Related Risks

- Have you worked with the customer in the past? Yes, all team members have completed at least one project for the customer, though none of them have been to the magnitude of the current project.
- Does the customer have a solid idea of what is required? Yes, the customer has access to both the System Requirements Specification, and the Software Requirements Specification.
- Will the customer agree to spend time in formal requirements gathering meetings to identify project scope? Unknown. While the customer will likely participate if asked, the inquiry has not yet been made.

D. Process Risks

- Does senior management support a written policy statement that emphasizes the importance of a standard process for software development? N/A. PA Software does not have a senior management. It should be noted that the structured method has been adopted. At the completion of the project, it will be determined if the software method is acceptable as a standard process, or if changes need to be implemented.
- Has your organization developed a written description of the software process to be used on this project? Yes.
- Are staff members willing to use the software process? Yes. The software process was agreed upon before development work began.
- Is the software process used for other products? N/A. PA Software has no other projects currently.

E. Technical Issues

- Are facilitated application specification techniques used to aid in communication between the customer and the developer? The development team will hold frequent meetings directly with the customer. No formal meetings are held (all informal). During these meetings the software is discussed and notes are taken for future review.
- Are specific methods used for software analysis? Special methods will be used to analyze the software progress and quality. These are a series of tests and reviews to ensure the software is up to speed. For more information, see the Software Quality Assurance and Software Configuration Management documents.
- Do you use a specific method for data and architectural design? Data and architectural design will be mostly object oriented. This allows for a higher degree data encapsulation and modularity of code.

F. Technology Risk

- Is the technology to be built new to your organization? No
- Does the software interface with new or unproven hardware? No
- Is a specialized user interface demanded by the product requirements? Yes

8.2 Development Environment Risks

- Is a software project management tool available? No. No software tools are to be used. Due to the existing deadline, the development team felt it would be more productive to begin implementing the project than trying to

learn new software tools. After the completion of the project software tools may be implemented for future projects.

IX. PROJECT SCHEDULE

9.1 Project Task

- Getting Basic Knowledge of python.
- Going through the previous existing system.
- Looking for required libraries in python.
- Collecting Dataset.
- Training Model using ML algorithm.
- Building GUI for better outlook.
- Dividing the assign task among group members.

9.2 Time Line Chart

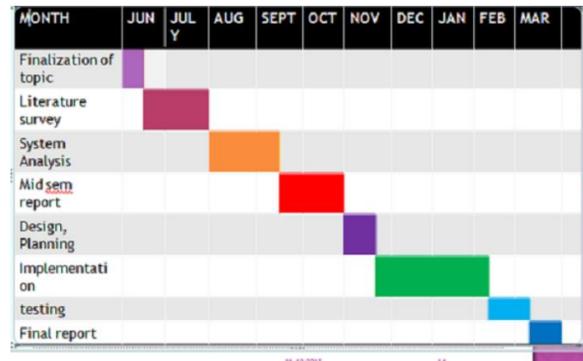


Figure: Time Line Chart

X. TOOLS AND TECHNOLOGY USED

10. Python

Python is an interpreted, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace. It provides constructs that enable clear programming on both small and large scales. VanRossum led the language community until July 2018. Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming.

Python features a comprehensive standard library, and is referred to as "batteries included". Python interpreters are available for many operating systems. CPython, the reference implementation of Python, is open-source software and has a community based development model. Python and CPython are managed by the non-profit Python Software Foundation.

Python is a general-purpose object-oriented programming language with high-level programming capabilities. It has become famous because of its apparent and easily understandable syntax, portability and easy to learn. Python is a programming language that includes features of C and Java. It provides the style of writing an elegant code like C, and for object-oriented programming, it offers classes and objects like Java.

- Python was developed in the late eighties, i.e., late 1980's by Guido van Rossum at the National Research Institute for Mathematics and Computer Science in the Netherlands as a successor of ABC language capable of exception handling and interfacing.
- Python is derived from programming languages such as ABC, Modula 3, small talk, Algol-68. Van Rossum picked the name Python for the new language from a TV show, Monty Python's Flying Circus.

- Python page is a file with a .py extension that contains could be the combination of HTML Tags and Python scripts.
- In December 1989 the creator developed the 1st python interpreter as a hobby and then on 16 October 2000, Python 2.0 was released with many new features. On 3rd December 2008, Python 3.0 was released with more testing and includes new features.
- Python is an open source scripting language., which means that anyone can download it freely from www.python.org and use it to develop programs. Its source code can be accessed and modified as required in the project. Python is one of the official languages at Google.

10.1. Features of Python

1. Easy to Learn and Use. Python is easy to learn and use.
2. Expressive Language. Python language is more expressive means that it is more understandable and readable.
3. Interpreted Language.
4. Cross-platform Language.
5. Free and Open Source.
6. Object-Oriented Language.
7. Extensible.
8. Large Standard Library

10.2 Algorithm

10.2.1 SVM

A support vector machine (SVM) is a supervised machine learning algorithm that can be used for both classification and regression purposes. SVM are mostly used in classification problems. SVM are founded on the idea of finding a hyperplane that best divides a dataset into two classes. Support vectors are the data points nearest to the hyperplane, the points of a data set that, if deleted, would alter the position of the dividing hyperplane. Because of this, they can be considered the critical elements of a data set.

The distance between the hyperplane and the nearest data point from either set is known as the margin. The aim is to choose a hyperplane with the greatest possible margin between the hyperplane and any point within the training set, giving a higher chance of new data being classified correctly. Examples of SVM boundaries. Selecting best hyperplane for our classification. We will show data from 2 classes.

The classes represented by triangle and circle. Case 1 Consider the case in Fig 1, with data from 2 different classes. Now, we wish to find the best hyperplane which can separate the two classes. Please check Fig 1. On the right to find which hyperplane best suit this use case. In SVM, we try to maximize the distance between hyperplane nearest data point.

This is known as margin. Since 1st decision boundary is maximizing the distance between classes on left and right. So, our maximum margin hyperplane will be "1st".

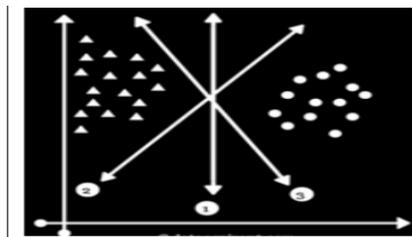


Figure 10.2: case 1

Case 2 Consider the case in Fig 2, with data from 2 different classes. Now, we wish to find the best hyperplane which can separate the two classes. As data of each class is distributed either on left or right. Our motive is to select hyperplane which can separate the classes with maximum margin. In this case, all the decision boundaries are separating classes but only 1st decision boundary is showing maximum margin between

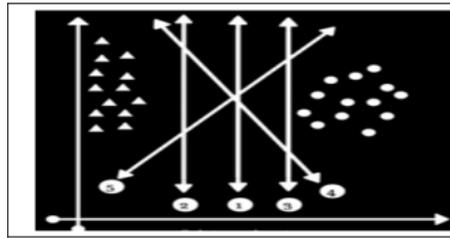


Figure 10.2: case 2

Case 3: Consider the case in Fig 3, with data from 2 different classes. Now, we wish to find the best hyperplane which can separate the two classes. Data is not evenly distributed on left and right. Some of the are on right too. You may feel we can ignore the two data points above 3rd hyperplane but that would be incorrect. SVM tries to find out maximum margin hyperplane but gives first priority to correct classification. 1st decision boundary is separating some from but not all. It's not even showing good margin. 2nd decision boundary is separating the data points similar to 1st boundary but here margin between boundary and data points is larger than the previous case. 3rd decision boundary is separating all from all classes. So, SVM will select 3rd hyperplane.

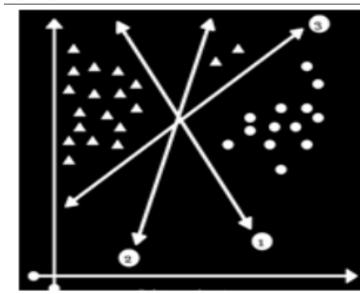


Figure 10.3: case 3

Case 4: Consider the figure 4, we will learn about outliers in SVM. We wish to find the best hyperplane which can separate the two classes. Data is not evenly distributed on left and right. Some of them are on right too. In the real world, you may find few values that correspond to extreme cases i.e., exceptions.

These exceptions are known as Outliers. SVM have the capability to detect and ignore outliers. In the image, 2 are in between the group of . These are outliers. While selecting hyperplane, SVM will automatically ignore these and select best-performing hyperplane. 1st 2nd decision boundaries are separating classes but 1st decision boundary shows maximum margin in between boundary and support vectors.

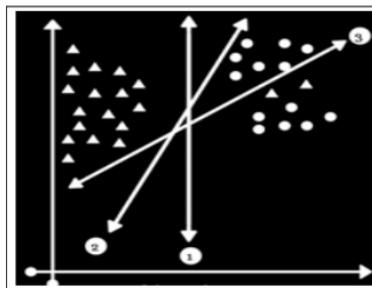


Figure 10.4: case 4

Case 5: We will learn about non-linear classifiers. Please check the figure 5 on right. It's showing that data can't be separated by any straight line, i.e, data is not linearly separable. SVM possess the option of using Non-Linear classifier. We can use different types of kernels like Radial Basis Function Kernel, Polynomial kernel etc. We have shown a decision boundary separating both the classes. This decision boundary resembles a parabola

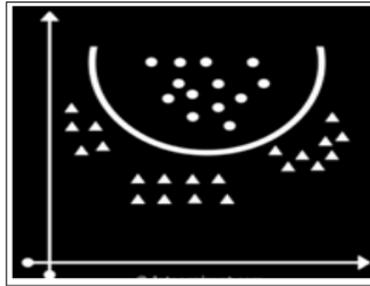


Figure 10.5: case 5

10.2.2 Random Forest

Random forest (RF) is the ensemble classifier, which collects the results of many decision trees by majority vote. In ensemble learning, the results of multiple classifiers are brought together, and a single decision is made on behalf of the community.

Each decision tree in the forest is created by selecting different samples from the original data set using the bootstrap technique. Then, the decisions made by many different individual trees are subject to voting and present the class with the highest number of votes as the class estimate of the committee. In the RF method, trees are created by CART (classification and regression trees) algorithms and boot bagging combination method.

The data set is divided into training and test data. From the training data set, samples are selected as bootstrap (re-sampled and sampled) technique, which will form trees (in a bag) and data that will not build trees (out of the bag). 1/3 of the training set is divided into data that will not form trees, and 2/3 of them will be data that will build trees.

- Step-1: Select random K data points from the training set
- Step-2: Build the decision trees associated with the selected data points (Subsets)
- Step-3: Choose the number N for decision trees that you want to build
- Step-4: Repeat Step 1 2

10.2.3 KNN

K-Nearest Neighbors (KNN) KNN is an effective supervised learning method for many problems including security techniques [16]. K-nearest neighbor is based on the clustering of the elements that have the same characteristics; it decides the class category of a test example based on its k neighbor that is near to it. The value of k in the KNN depends on the size of dataset and the type of the classification problem. Fig. 1 shows that KNN classifies the target based on its neighbors. KNN is explained as follows:

Find the nearest elements from the test data as to training data K based on Euclidean distance to calculate the distance. For two elements in k dimensional space, $a = [a_1, a_2, \dots, a_k]$ and $y = [b_1, b_2, \dots, b_k]$, the Euclidean distance based on the two elements can be computed After collecting the k-nearest neighbors, the majority of the k-nearest neighbors will be considered as a class for the test data.

XI . TEST PLANS

A system should always be tested thoroughly before implementing it, as regards its individual programs. This is because implementing a new system is a major job which a lot of man hours and a lot of other resources, so an error not detected before implementation may cost a lot. Effective testing early in the process translates directly into long term cost saving from reduced number of errors.

This is also necessary because in some cases, a small error is not detected and corrected before installation, which may explode into much larger problem. Programming and testing is followed by the stage of installing the new computer based system.

Actual implementation of the system can begin at this point using either a parallel or a direct changeover plan, or some blend of two. Testing and implementation of firefighting robot controlled using android application is carried out as below.

Software testing is a critical element of Software Quality Assurance (SQA) and represents the ultimate review of specification, design and coding. The purpose of product testing is to verify and validate the various work products viz. units, integrate unit, final product to ensure that they meet their respective requirements.

11.1 Testing Procedure

Software Testing is the critical element of the Software Quality Assurance and represents the ultimate review of specification, design and coding. Testing is the process of checking whether software works according to the specification. Testing will be performed by running the program using the test data. Testing is vital to the success of the system. It will also test whether the system identifies the problem correctly. System is tested by following steps:

Unit Testing:

Each program is tested individually using dummy records to see whether that program produce satisfactory reports.

Sequential Testing:

The program, whose output will affect the processing done by another program, will be tested using dummy records. Testing: The system is corrected in such a way that it does not affect the forced system failure. This testing is done with low volumes of data.

11.2 Test Strategy

The test strategy consists of a series of different tests that will fully exercise the system. The primary purpose of the test is to uncover the system limitations. Following are the several tests that will be conducted:

11.2.1 Unit Testing

Testing conducted to verify the implementation of the design for one software element (e.g., unit, module) is called unit testing. The purpose of unit testing is to ensure that the program logic is complete and correct and ensuring that the component works as designed. In this module, each unit will go through Unit testing after the completion of the module. The bugs in module testing will be reported in Test Log document and will be reported to the developers. After fixing the bug successfully, one more iteration of module testing (Regression Testing) is done. This process is repeated till all critical test cases pass.

11.2.2 Integration Testing

Testing conducted in which software elements, hardware elements, or both are combined and tested until the entire system has been integrated. The purpose of integration testing is to ensure that design objectives are met and ensures that the software, as a complete entity, complies with operational requirements.

This type of testing will be done after all module test cases are passed through module testing, security testing, performance testing, user interface testing and regression testing

11.2.3 Performance Testing

In developing the system, we are going to use Java which will reduce the response time. In Performance Testing, We are going to test Response time for each Screen. It is a type of non-functional testing.

Performance testing is testing that is performed; to determine how fast some aspect of a system performs under a particular workload. It can serve different purposes like it can demonstrate that the system meets performance criteria. It can compare two systems to find which performs better.

Or it can measure what part of the system or workload causes the system to perform badly. This process can involve quantitative tests done in a lab, such as measuring the response time or the number of MIPS (millions of instructions per second) at which a system functions.

11.2.4 Regression Testing

Testing done to ensure that, the changes to the application have not adversely affected previously tested functionality. Here testing will take care of the test cases passed during the first module testing will not be affected in the subsequent rounds of module testing.

11.3 Test Cases

The listed tests were conducted in the software at the various developments stages. Unit testing was conducted. The errors were debugged and regression testing was performed. The integration testing will be performed once the system is integrated with other related systems like Inventory, Budget etc.

Once the design stage was over the Black Box and White Box Testing was performed on the entire application. The results were analyzed and the appropriate alterations were made. The test results proved to be positive and henceforth the application is feasible and test approved.

XII. CONCLUSION

The main purpose of this project is to automate the manual work of the covid-19 vaccination done. By this system the tracking of vaccination done in villages can be easily possible. Also the record will be maintained at District level. By using this, the record will remain more secure and track of vaccination done weekly in villages will be possible. The proposed system uses machine learning techniques of Support Vector Machine (SVM) for getting the vaccine survey status.

REFERENCES

- [1]. Bhand Vishal Dnyaneshwar, "COVID-19 Vaccine Supply Chain Management and Logistics in India", International Research Journal of Engineering and Technology, 2021
- [2]. Ashishsrivastva and NirajTiwari, "COVID-19 DATA TRACKER ", International Research Journal of Engineering and Technology 2020
- [3]. S Joselena Percy Jehane and D Janani, "Digital Tracking of Children Health Status and Immunization Services", International Research Journal of Engineering and Technology, 2020
- [4]. ApoorvaShete, RohanPradyuman, "Sentiment Analysis of COVID-19 Vaccine Tweets", International Research Journal of Engineering and Technology, 2021
- [5]. Sanjib k Deka and SubhasishGoswami, "A Blockchain Based Technique for Storing Vaccination Records", IEEE 2021
- [6]. M. UsmanAshraf "Performance And Power Efficient Massive Parallel Computational Model For HPC Heterogeneous ExascaleSystems." IEEE Access, Volume 6, April 2018, Pages 23095-23107.
- [7]. Le Hoang Son, Sudan Jha "Collaborative Handshaking Approaches Between Internet of Computing and Internet Of Things Towards A Smart World: A Review From 2009–2017." Telecommunication Systems, Volume 70.4, April 2019, Pages 617- 634.
- [8]. Li Bai, Dawei Yang "Chinese Experts' Consensus on the Internet of Things Aided Diagnosis and Treatment of Coronavirus Disease 2019 (COVID- 19)" Clinical.eHealth, Volume 3, 2020.
- [9]. Researchers to study if a startup's wrist-worn wearable can detect early COVID19 respiratory issues.
- [10]. <https://techcrunch.com/2020/04/01/researchers-to-study-if-startups-wrist-worn-wearable-can-detect-early-covid-19-respiratory-issues/>