



MINI - PROJECT REPORT ON

“SECURE SERVER MANAGEMENT”

BY

1. HARSHWARDHAN MEHROTRA (2193113) MITU19BTCS0172
2. VIREN GHUIKHEDKAR (2193087) MITU19BTCS0100
3. SOHAM DHANDE (2193246) MITU19BTCS0085
4. SHREYANSH DUBEY (2193240) MITU19BTCS0089

Under the Guidance of
Dr. NILESH MARATHE

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

MIT SCHOOL of Engineering

Loni Kalbhor Pune

M.I.T. SCHOOL OF ENGINEERING
DEPARTMENT OF COMPUTER ENGINEERING
LONI – KALBHOR PUNE

CERTIFICATE



This is to certify that the Mini- Project report entitled

“SECURE SERVER MANAGEMENT”

submitted by

1. HARSHWARDHAN MEHROTRA (2193113) MITU19BTCS0172
2. VIREN GHUIKHEDKAR (2193087) MITU19BTCS0100
3. SOHAM DHANDE (2193246) MITU19BTCS0085
4. SHREYANSH DUBEY (2193240) MITU19BTCS0089

is a record of bonafide work carried out by them, under my guidance, in partial fulfillment of the requirement for the Second Year of Engineering(Computer) at M.I.T. School of Engineering, Pune under MIT Art, Design & Technology University.

Date:

Place:

Dr. Nilesh Marathe

Guide,

Department of CSE
M.I.T. School of Engineering

Dean Engineering,

Head , Department of CSE
MIT School of Engineering

ACKNOWLEDGEMENT

We owe a debt of sincere gratitude, and respect to our guide and mentor Dr. NILESH MARATHE, Professor, MIT ADT UNIVERSITY, PUNE for his sagacious guidance, vigilant supervision and valuable critical appreciation throughout this project work.

INDEX

SR NO.	CONTENTS	PAGE NO.
1	ABSTRACT	5
2	1.INTRODUCTION 2.PROBLEM DEFINITION 3.WORKING 3.FEATURES OF PROJECT 4.PLATFORM/TECHNOLOGY 5.FLOWCHART 6.OUTPUT 7.CODE	6 7 8 9 10-11 12-13 14-15 16
3	LITERATURE SURVEY	17-20
4	CONCLUSION	21
6	REFERENCES	22

ABSTRACT

“Secure Server Management” is a Web Application that assists the work from home situation of a system administrator by keeping all the data related to the server he/she is managing in one place while keeping all the servers with their data secure. Our application makes the process more secure by making the servers accessible using the application only, so as to reduce the possibility of the servers being available to brute force attacks. This will reduce their hassle of keeping track of which password and key are for which server. This helps overall in preparing the admin in managing servers thus providing better network control and along with that server’s hostname, user_id and password keys are encrypted with AES(256) and stored in KeePass database. Being a web application built with Django, the software has the advantage of being portable and usable anywhere. Users can log in anywhere at any time and monitor their network control. Our dashboard provides 3 essential domain options to work with, firstly to provide their IP address of the system, which is safely stored in our database so that we can locate and provide output regarding that server, next there is the monitoring tab with the help of which one can monitor their CPU usage and memory used by their server system and based on that the amount of data being transferred can be maintained, as not to overload the server. Our website provides an additional SSH terminal that can be used to remotely execute code on the server.

INTRODUCTION

Companies depend on their server infrastructure for most IT functions, including data storage, hosting websites, emails, and applications. While many businesses have shifted to cloud services using servers located in enormous, distant data centres, a significant number of organizations still have in-house servers or use a hybrid environment of in-house and cloud services to host server data requiring management. Managing a server, whether in-house or in the clouds, means staying on top of hardware, software, security, and backups.

Server management is the process of monitoring and maintaining servers to operate at peak performance. Server management also encompasses the management of hardware, software, security, and backups. The primary goals of an effective server management strategy are to:

- Minimize—and hopefully eliminate— server slowdowns and downtime
- Build secure server environments
- Ensure servers continue to meet the needs of an organization as it evolves.

PROBLEM STATEMENT

In this pandemic everyone has to work home and the work of a system administrator gets extremely difficult as he/she might have to manage many servers and also manage their passwords and keys.

We can ease this process of server management by making all the servers accessible using one application with one click login process at the same time providing more security to the servers by becoming the only passage to connect to the servers.

WORKING

FRONT END:->

The user is greeted with a splash screen stating the name of the application 'SERVER MANAGEMENT' as soon as they open the application. The next Activity contains the login /registration page where the users can sign up with new credentials for the first time, or identify and authorize themselves with their credentials. Once logged in, the Activity houses the buttons for allowing the users to give 3 options, server management using SSH, resource monitoring (In App), and file management. The front-end was made keeping in mind the ease of use and instructiveness of the website so that users can easily navigate within our website. Tools and language used to build the front end were as follows - HTML5, CSS3, and Chart-Js.

BACK END:->

Once the admin has entered their server information, the data is stored in our encrypted database and then data regarding the server is retrieved when the user wants to access the server. Upon the request the app can provide a SSH terminal for their server and they can remotely execute code or collect some data using the terminal using a service called webssh written in python.

FEATURES

The aim of this project is to improve the work from home situation of a system administrator by keeping all the data related to the server he/she is managing in one place. This will reduce their hassle of keeping track of which password and key is for which server. Server management is the process of monitoring and maintaining servers to operate at peak performance. Server management also encompasses the management of hardware, software, security, and backups. The primary goals of an effective server management strategy are to:

Primary goals:

- Minimize—and hopefully eliminate— server slowdowns and downtime
- Build secure server environments
- Ensure servers continue to meet the needs of an organization as it evolves.

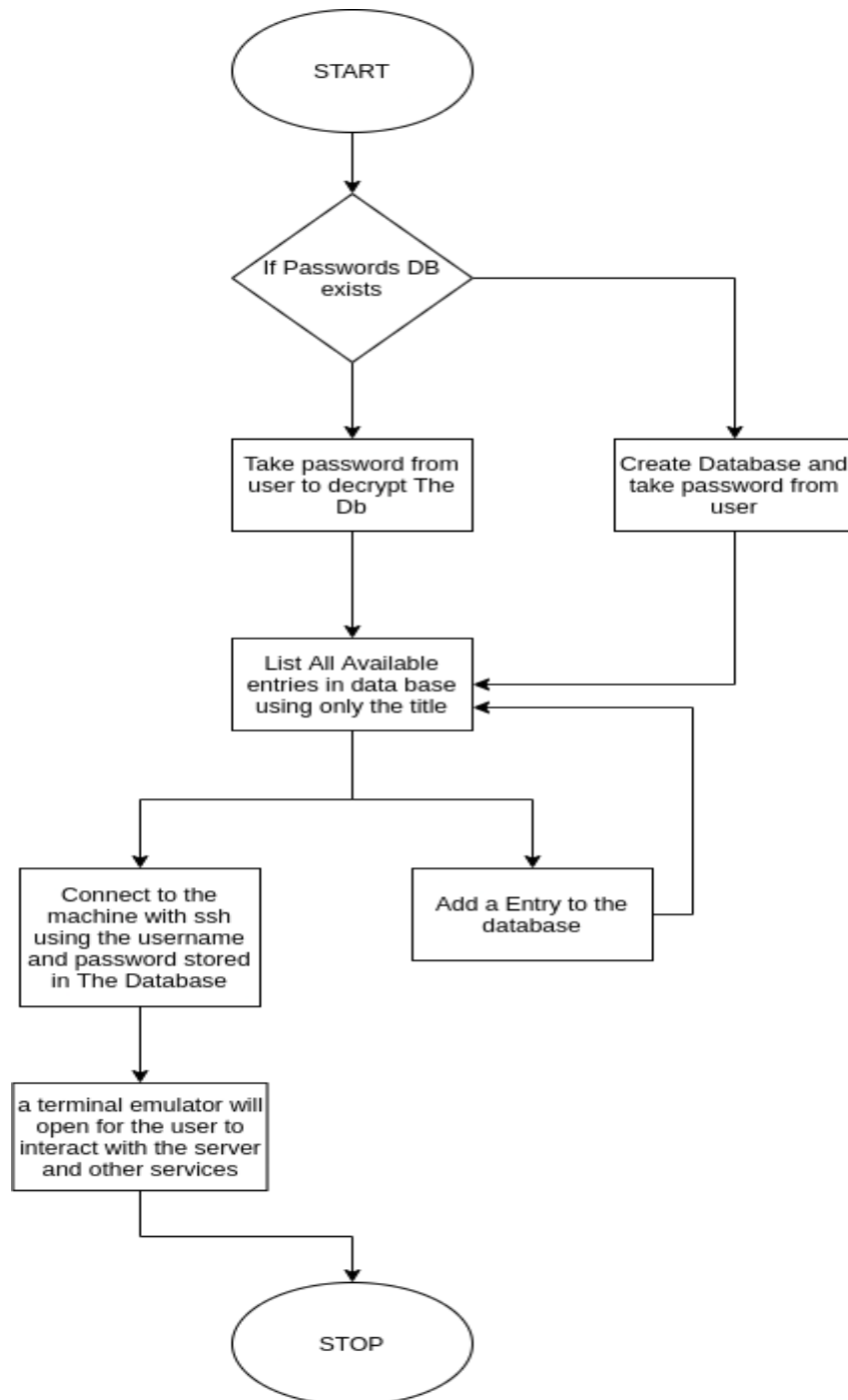
In this Pandemic everyone had to work from home and the work of a system administrator get extremely difficult as he might have to manage many servers and also manage there passwords and keys, all the server's hostname, user_id and password keys are encrypted with AES(256) and stored in a KeePass database.

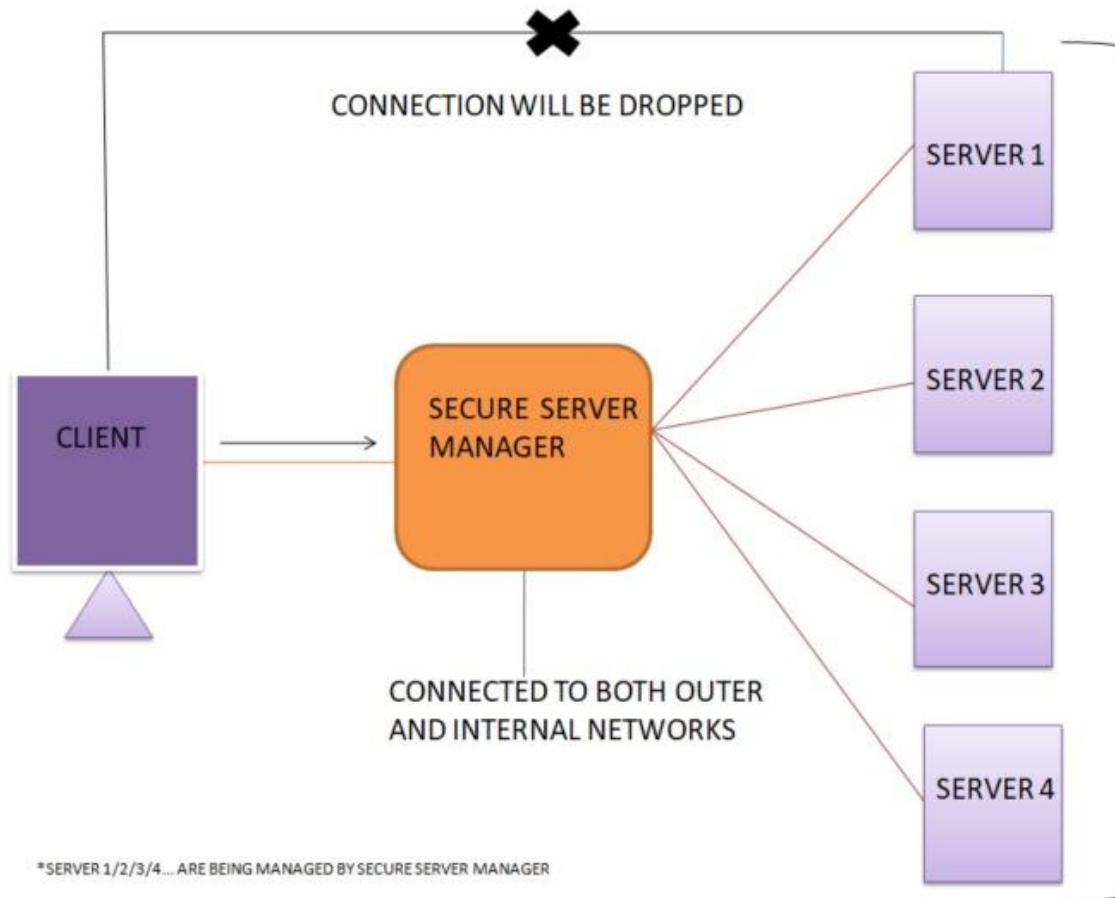
We can ease this process of server management by making all the servers accessible using one application with One Click Login process.

PLATFORM AND TECHNOLOGY USED

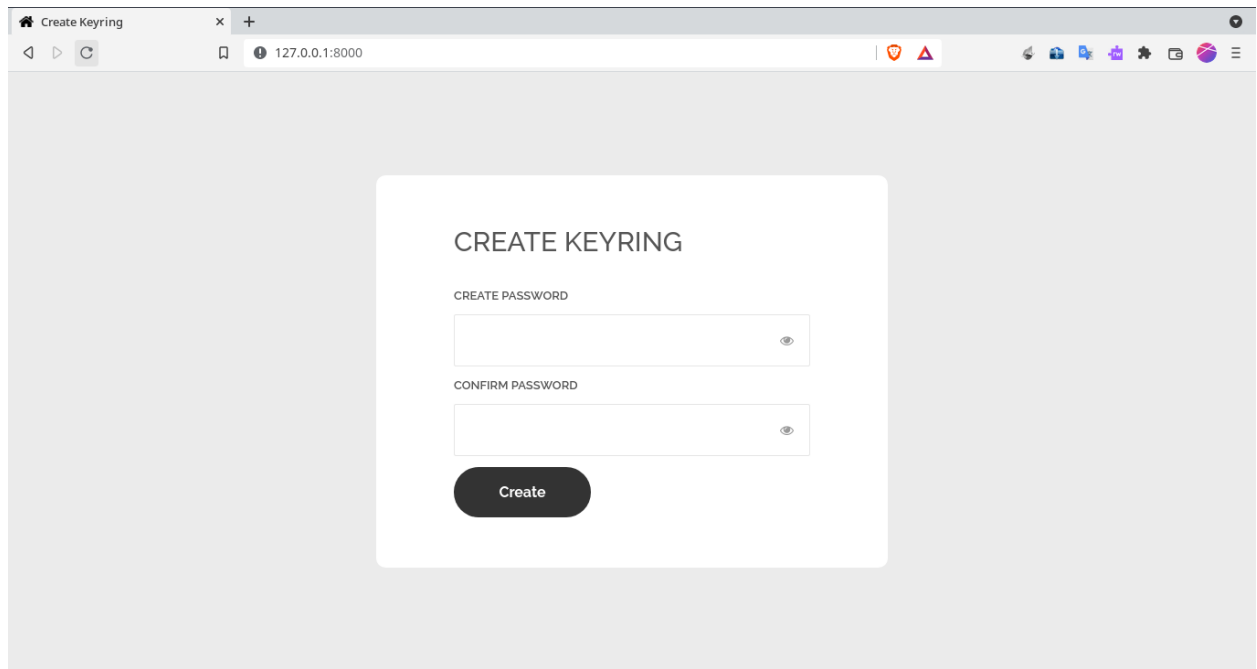
SR. NO.	TOOL	DESCRIPTION
1.	Webssh	This is used to connect to the server using ssh over the web.
2.	Django	We have used it as the framework for our web application.
3.	Pykeepass	It is a python library which helps us to create and make changes to the encrypted KeePass Database. Whenever a new entry is to be added or updated pykeepass helps us to do those changes in the KeePass database.
4.	ChartJs	This library helped us to create the CPU usage and memory Usage charts in the monitoring tab.
5.	Github	For sharing and committing changes on source file

FLOWCHART





OUTPUT



A screenshot of a web browser window titled "Create Keyring". The address bar shows "127.0.0.1:8000". The page features a central white card with the heading "CREATE KEYRING". Below the heading are two password input fields: "CREATE PASSWORD" and "CONFIRM PASSWORD", each with a toggle icon on the right. A dark "Create" button is positioned at the bottom of the card.

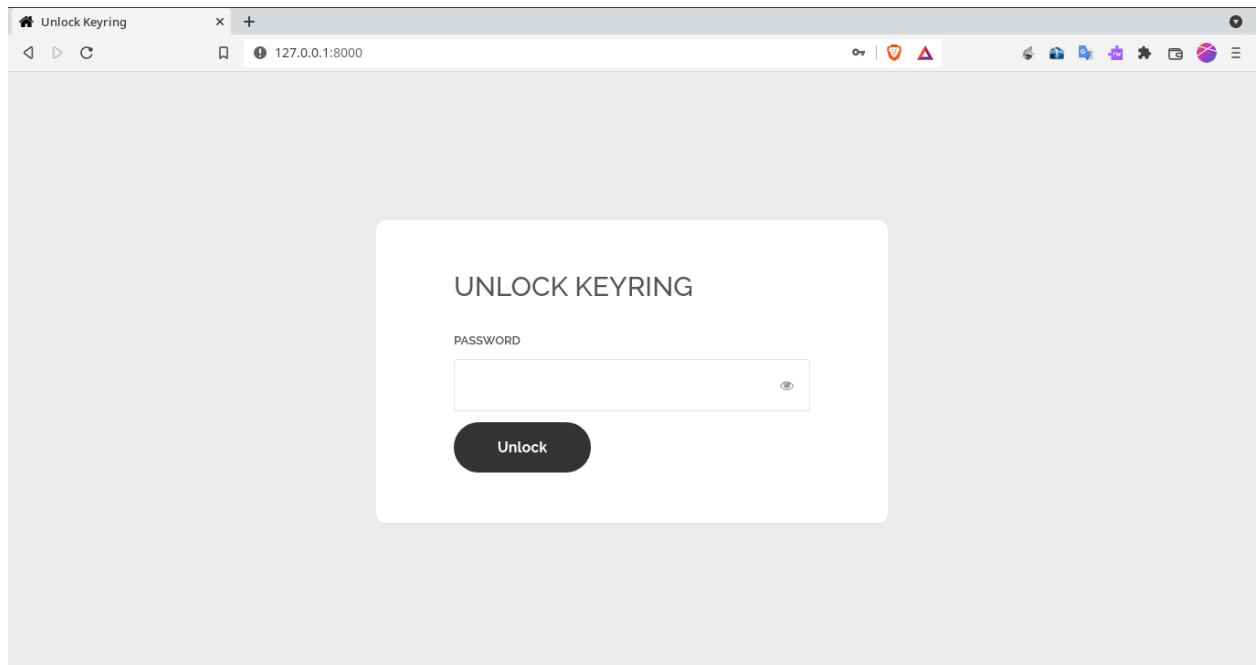
Create Keyring

CREATE KEYRING

CREATE PASSWORD

CONFIRM PASSWORD

Create



A screenshot of a web browser window titled "Unlock Keyring". The address bar shows "127.0.0.1:8000". The page features a central white card with the heading "UNLOCK KEYRING". Below the heading is a single password input field labeled "PASSWORD" with a toggle icon on the right. A dark "Unlock" button is positioned at the bottom of the card.

Unlock Keyring

UNLOCK KEYRING

PASSWORD

Unlock

Server Manager

127.0.0.1:8000/dash/?server=192.168.0.183

#

Terminal

Monitoring

Keyring

```
Linux tweety 6.10.63-v7l+ #1459 SMP Wed Oct 6 16:41:57 BST 2021 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Nov 22 23:25:39 2021 from 192.168.0.110
Welcome to fish, the friendly interactive shell
harsh@tweety ~->
```

Server Manager

127.0.0.1:8000/monitoring/?server=192.168.0.183

#

Terminal

Monitoring

Keyring

13 %

CPU Usage
No. of Cores:4

47 %

Memory Usage
Used:1.76GB Free: 1.99GB
Total:3.75

Terminal

Monitoring

Keyring

	hostname	username	password	privatekey
	192.168.0.183	harsh		
	123	harsh		
	1234	123		
	a	a		
	192.168.0.219	aastha		

Add New Key

CODE:

https://github.com/harsh18262/server_manager_django

Literature Survey:

Sr No.	Referred Paper/Journal	Year of Publication	Summary of paper referred
1	Design and Implementation of Cloud Server Remote Management System	2016	This paper helped us to understand how and what data should be taken from a server for monitoring
2	Management Server	2014	This journal helped us to understand what all should be present in a server management software
3	Client Server	2019	In this survey, we present a detailed report for the client-server based system, highlighting its key concepts, architectural principles, and state-of-the-art implementation as well as research challenges. This paper aims to provide a better awareness of the design challenges of a client-server based system and identify essential research guidelines.

4	Django	2020	Django provides a wide range of features and functionalities. The administration interface provided by Django is one of the coolest things. It's truly simple to create and it's really one of the key advantages when using the framework.
5	Security credentials and their distribution	2019	The text will give an introduction about credentials and their security. And it will present some methods for securing credentials information and existing security credentials system. Finally, it will show the detailed analysis of the network application, it is including the developer and customer point of view, specification and the implementation of it along with the technologies used to create the application.

6	AN EFFECTIVE MECHANISM FOR SECURING AND MANAGING PASSWORD USING AES-256 ENCRYPTION & PBKDF2	2021	The data is encrypted using AES-256 encryption algorithm and PBKDF2 which is the current industry standard, additionally, the username and password are encrypted with a key generated from the user's master password, ensuring data security, and the password manager can be integrated into the browser as an extension, ensuring high compatibility and ease of use for end users.
---	---	------	---

7	Enabling SSH Protocol Visibility in Flow Monitoring	2019	<p>Enabling SSH Protocol Visibility in Flow Monitoring</p> <p>The network flow monitoring has evolved to collect information beyond the network and transport layers, most importantly the application layer information. This information is used to improve network security and performance by enabling more precise performance analysis and intrusion detection. In this paper, we contribute to this effort by extending flow monitoring with information from the SSH protocol.</p>
8	Distributed SSH key management	2018	<p>Distributed SSH key management with proactive RSA threshold signatures</p> <p>In this paper we present ESKM - a distributed enterprise SSH key manager. ESKM is a secure and fault-tolerant logically-centralized SSH key manager. ESKM leverages k-out-of-n threshold security to provide a high level of security. SSH private keys are never stored at any single node, not even when they are used for signing. On a technical level, the system uses k-out-of-n threshold RSA signatures, which are enforced with new methods that refresh the shares in order to achieve proactive security and prevent many side-channel attacks.</p>

9	Secure shell	2020	<p>In this paper we will get idea about Secure Shell i.e Secure Shell gives Associate in open convention. Secure Shell clients/server arrangements provide shell for command, transfer of file for TCP/IP applications. Software purchasers and servers area unit develop local Windows usage that provide a selection of SSH.</p>
10	HTTPS Interception	2016	<p>In this paper, they present a comprehensive study on the prevalence and impact of HTTPS interception. First, they showed that web servers can detect interception by identifying a mismatch between the HTTP User-Agent header and TLS client behavior. Here they characterize the TLS handshakes of major browsers and popular interception products, which they use to build a set of heuristics to detect interception and identify the responsible product.</p>

CONCLUSION

To conclude we are making an application where a system administrator can store their server login details securely and access them easily without the hassle of remembering them. This will improve their way of life a bit.

Will improve work from home to smoothen up the server access process. Will be a single platform for all kinds of server management needs. This application can further be expanded to assess its admins the overall behaviour, data transfer patterns, their strengths and weaknesses, loopholes, etc in order to provide them a detailed analysis.

REFERENCES

- (i)
Design and Implementation of Cloud Server Remote Management System-<https://ieeexplore.ieee.org/abstract/document/7518445?section=abstract>
- (ii)
Management-Server:<https://www.sciencedirect.com/topics/computer-science/management-server>
- (iii)
The Security Impact of HTTPS Interception
Zakir Durumeric*, Zane Ma†, Drew Springall*, Richard Barnes‡, Nick Sullivan§,
Elie Bursztein¶, Michael Bailey†, J. Alex Halderman*, Vern Paxson||
- (iv)
A RESEARCH ON SECURE SHELL (SSH) PROTOCOL
K.Sivaraman
Assistant Professor, Dept. of CSE, Bharath University,
Chennai, Tamil Nadu, India,

Annexure:

Annexure I: Form A-Title Approval

Annexure II: Form B-Market and financial feasibility

Annexure III: Literature survey paper

Annexure IV: Project Tracker Sheet

CSE Department, MIT School of Engineering, MIT ADT University, Pune

THANK YOU