Cover Page

CYBER PREACH

Submitted in partial fulfillment of the requirements for
the award of
Bachelor of Science degree in Computer Science

by

A. MOHAMED SALMAN (39290067)
&
R. HEMNATH (39290035)

DEPARTMENT OF COMPUTER SCIENCE

SCHOOL OF COMPUTING

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI - 600 119

MARCH - 2022

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE
www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of A. Mohamed Salman (39290067) & R. Hemnath (39290035) who carried out the project entitled "CYBER PREACH" under my supervision from_____to _____.

DECLARATION

I_____(Name of the Candidate)_____hereby declare that the Project Report entitled_____ done by me under the guidance of Dr./Prof./Mr./Ms._____(Internal) is submitted in partial fulfillment of the requirements for the award of Bachelor of Science degree in_____.

DATE:

PLACE: **SIGNATURE OF THE CANDIDATE**

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T.Sasikala M.E., Ph.D**, **Dean**, School of Computing **Dr. L. Lakshmanan M.E., Ph.D. ,** and **Dr.S.Vigneshwari M.E., Ph.D. Heads** of the Department of Computer Science and Engineering for providing menecessarysupport and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide Dr./Mr./Ms for his valuable guidance, suggestions and constantencouragementpaved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Departmentof Computer Science and Engineering**who were helpful in many ways for thecompletionof the project.

SYNOPSIS

ABSTRACT:

Technology grow's In day to day life. Some people it's difficult to adapt for modern technology's. Our application make's the work easier for those people whom are lagging in the digital world.

Cyber Preach application provides the basic knowledge about how to use technology via video, theory method and also MCQ's has also been added for a quick knowledge check.

It is classified as a micro framework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third party libraries provide common functions.

Flask python language is adding to the application to improve the use of the system. Mongo Database used for the Application Database

OBJECTIVES:

The main objective of the cyber preach refers to the digital platform that helps the rural people to learn about basics of the digital world.

The purpose of the project is to build an application recognizes the people of the world to survive in the fast –paced digital world.

The main purpose of this project is to educated the people who don't know to use the digital technology

Cyber preach has a easy way of teaching technical knowledge by the method of video, theory, and also an quick knowledge check (MCQ) will also been in our way of teaching.

PLATFORM:

Operating Systems: Microsoft Windows

Technologies Use: Flask python

Front End:

HTML, Css and java script .

Web designing language:

Python

Database :

Mongo Data Base (No SQL)

Microsoft Windows, Linux and Mac

Hardware Requirements:

No hardware is required

Network Connectivity

TABLE OF CONTENTS

CHAPTER 1

1. INTRODUCTION:

Cyber Preach is an automated system that is developed using flask python and Mongo data base. The main purpose of this system is to Improvement of efficiency and effectiveness of rural people. This system is so easy to use that even a newbie can use the features with ease.

In the world of modern technology, there are few people who can't able to synchronized with the digital world. Using modern technology's like internet, phone, computer and social media stuff like this. The cyber preach would be helpful for people to learn about basics of the digital world.

Enhance the quality of learning and teaching. The way cyber preach is teach via video method. We have used some YouTube video script for the teaching method. So, the way of teaching will be easy. Meet the learning style that need for the people.

Learning via video method is will be efficient all though we have added some MQC for a general knowledge. General knowledge check will be evaluates the person is okay with the curriculum that he selected in cyber preach.

Improvement of efficiency and effectiveness of people.

CHAPTER 2

2. LITERATURE SURVEY:

2.1 Student Management System based on .Net Three-layer Structure Student Information Management system has been implemented by (Yang Qingshan, 2010) using .Net Architecture for better standardization, information management, and three layer B/S structure. This architecture improves the security and maintainability.

2.2 Design and implementation of Student Information Management System Similarly simplified friendly interface student management system has been designed and implemented with the use of Visual Basic 6.0 and the SQL Server 2000. This design has a fully functional, flexible and convenient application which provides friendly interface to the users (Zhibing Liu, 2010).

2.3 Student Information Access System via Bluetooth K.A.C. Dushamali (2007) also presented their work and concluded the information can be accessed by teachers and students using Bluetooth. The system consist of a J2SE Server 2 Application, a mobile phone, a Bluetooth dongle, web service which connects the database and the system, a database and a web interface.

2.4 A Project Report on Student Information Management System Kapil Kaushik (2011) also built a student information management system with features like online registration of students, maintenance of students records and searching student records. Their system was built using HTML, JavaScript implementing PHP, MySQL (as back end). They also used Apache HTTP server and Dreamweaver (Kapil Kaushik, 2011).

2.5 Web Based Student Information Management System (SIMS)

S.R.Bharamagoudar, Geeta R.B and S.G.Totad also have built SIMS for maintenance of student information system solving the traditional way of managing the student records. The system was built using html, PHP and MySQL database.

CHAPTER 3

3. METHODOLOGY

-

3.1– Login module

3.2– Home module

3.3- Learn module

- *Internet*
- *Computer*
- *Phone*
- *Social media*

3.4- Quiz module

- *Internet*
- *Mobile*
- *Computer*

3.1-Login module:

Login script is used to provide the authentication for our web pages. the

Script executes after submitting the user login button.

3.2-Home module:

Home page contains LOGO and TAGLINE of our website. It looks attractive to give a good impression for the users.

3.3-Learn module:

- While Opening the learn mode there will be four sections. The Sections are Internet, Phone, Computer and Social Media.
- Each Sections shows us different types of information.
- We can even get the tutorial video to understand easily.
- There will be search box at the top of the learn module page.

3.4-Quiz module:

Quiz module is to gain an extra knowledge about the technology devices.It contains lots of questions about phone, computer and mobile. On choosing any of the section Next page will open in that page there will be questions about the section we have opened, Then It will show us the question, and the answers are in true or false type if we choose correctly it will show the result we have gained at the end

3.5- Technologies used:

FLASK PYTHON
MONGO DATABASE
WIKIHOW
JAVA SCRIPT
HTML
CSS

4 - RESULTS AND DISCUSSIONS:

4.RESULTS:

4.1. Performance Requirements:

The proposed system that we are going to develop will be used as the Chief performance system for providing help to the organization in managing the whole database of the student studying in the organisation. Therefore, it is expected that the database would perform functionally all the requirements that are specified.

4.2. Safety Requirements:

The database may get crashed at any certain time due to virus or operating system failure. Therefore, it is required to take the database backup.

4.3. Security Requirements:

We are going to develop a secured database. There are different categories of users namely administartor. People who will be viewing either all or some specific information form the database.
Depending upon the category of user the access rights are decided. It means if the user is an administrator then he can be able to modify the data, append etc. All other users only have the rights to retrieve the information about database.

4.4 - DISCUSSIONS:

4.4.1) Scope and purpose

The purpose of the design phase is to develop a clear understanding of what the developer want people to gain from his/her project. As you the developer work on the project, the test for every design decision should be "Does this feature fulfill the ultimate purpose of the project?"

A purpose statement affects the design process by explaining what the developer wants the project to do, rather than describing the project itself. The Design Document will verify that the current design meets all of the explicit requirements contained in the system model as well as the implicit requirements desired by the customer.

4.4.2) Overall System Design Objectives

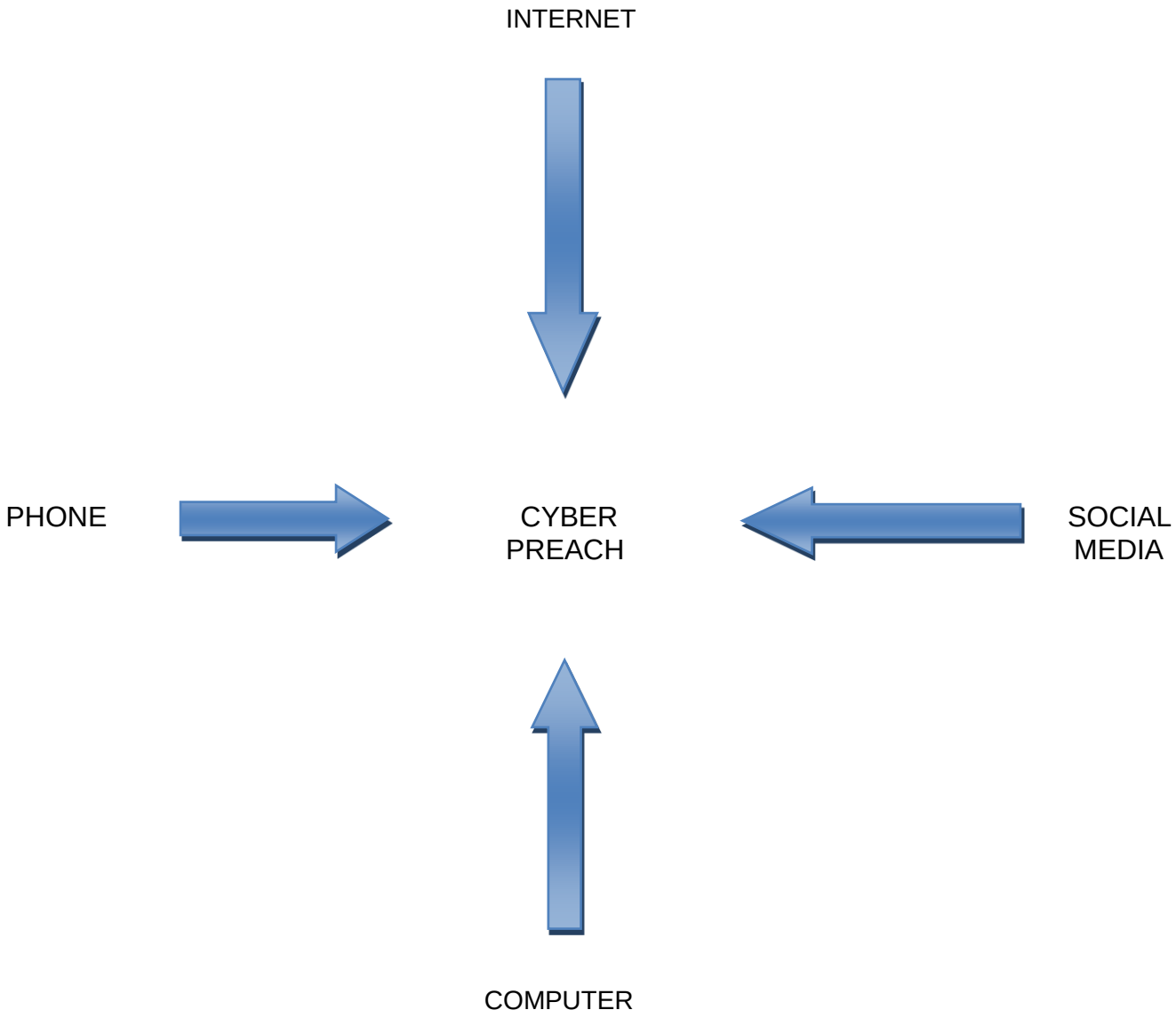
The overall system design objective is to provide an efficient, modular design that will reduce the system's complexity, facilitate change and result in an easy implementation. This will be accomplished by designing strongly cohesion system with minimal coupling. In addition, this document will provide interface design models that are consistent user friendly and will provide straight forward transition through the various system functions.

4.4.3) Structure of Design Document

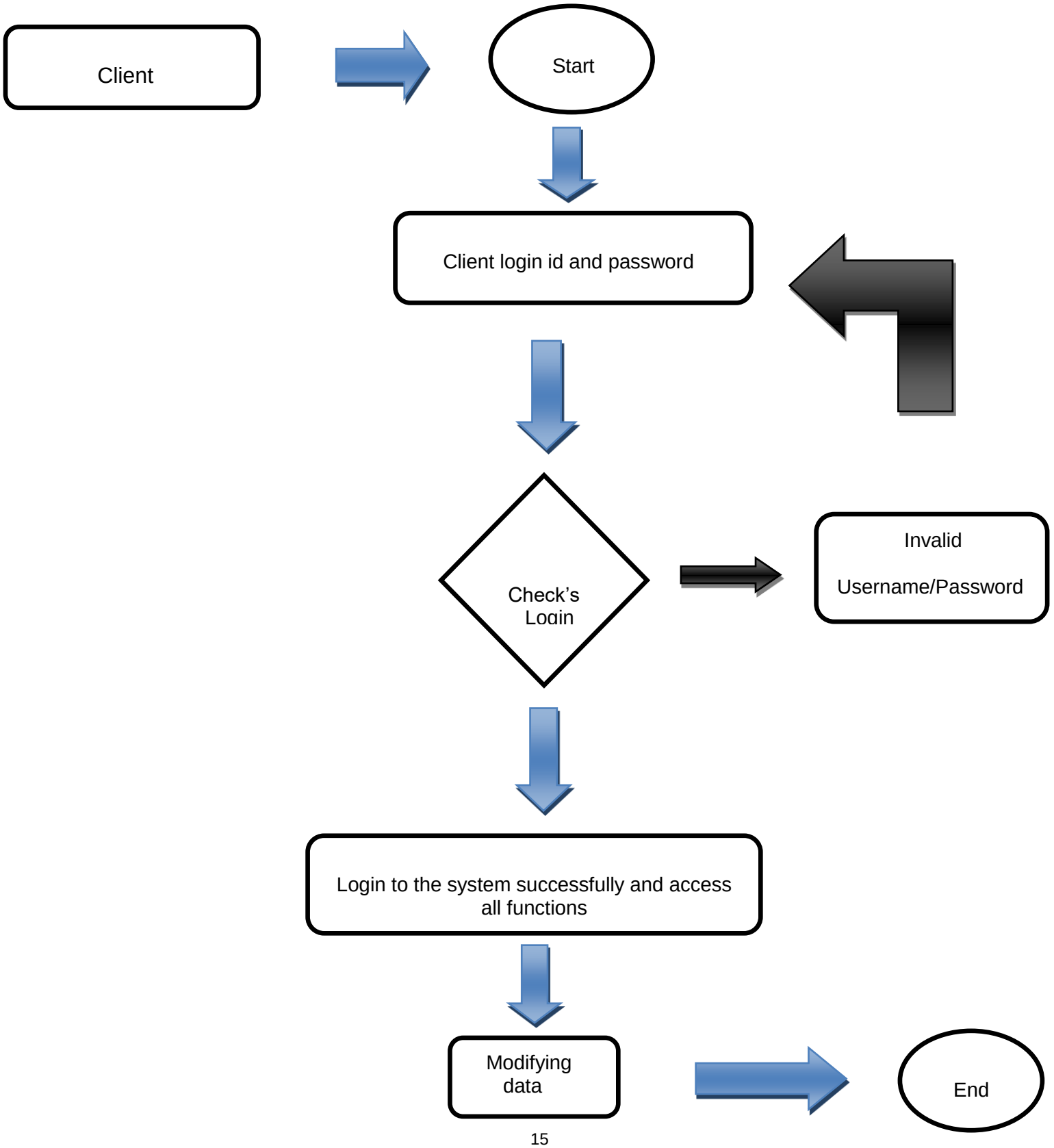
System Architecture Design – The System architecture section has detailed diagram of the system, server and client architecture. *Data Design* – The data Design include an ERD as well as Database design

5 - CONCLUSION AND FUTURE WORK:

5.1. SYSTEM ARCHITECTURE DESIGN:



5.2 - DATA FLOW DESIGN:



5.3 - FINAL CONCLUSION:

Here, I have come to the end of the project on the title of Cyber preach. I tried my best to include all the necessary points that are required related to the given title. Some of the information I wrote in the project's literature review were taken from the internet and I have also referred some books. And this presentation contains information and some concepts of our projects. I do hope that my project will be interesting and may be even knowledgeable."

6 - REFERENCES:

i)PHP book by Vasvani (TMH publications).

ii) Beginning PHP5 by WROX.

iii) www.google.com.

iv) www.wikipedia.com

V) www.w3schools.com

Vi) Informatics practices by Sumita Arora.

Vii) Head First PHP & MySQL by Lynn Beighley and Michael Morrison(O'Reilly)

Viii) Ritchie, Dennis M. (1993). "The Development of the C Language". History of Programming Languages, 2nd Edition. Retrieved 2018-11-11.

IX)"'C' Programming Language: Brian Kernighan - Computerphile". Retrieved 2018-11-11.

X) Paul, Ryan (24 October 2005). "A guided tour of the Microsoft Command Shell". Ars Technica. Retrieved 10 April 2011.

7 - APPENIDICES:

7.1 - SAMPLE CODE:

```
from logging import debug
from flask import
Flask,render_template,url_for,request
from flask_mongoengine import
MongoEngine
from pywikihow import search_wikihow
import contextlib,io,os
from pymongo import MongoClient
import pandas as pd



app = Flask(__name__)

app.config["MONGO_URI"] =
"mongodb+srv://codesploit:codesploit@cl
uster0.xcehq.mongodb.net/test"

client =
MongoClient("mongodb+srv://codesploit:c
odesploit@cluster0.xcehq.mongodb.net/te
st")
dbs = client['Cyberpreach']

class question():
    def __init__(self,question):
```

```
self.question=dbs.quiz.find_one({"name":q
uestion})
        self.quiz=[]
    def fivequestion(self):
        self.quiz=self.question
        return self.quiz

@app.route('/',methods=['GET','POST'])
def index():                              #index
function
    return render_template('index.html')

@app.route('/login',methods=['GET','POS
T'])
def login():
    if(request.method=="POST"):
        email = request.form.get("email")
        passw = request.form.get("pass")
        s=dbs.login.find_one({"email":email,
"pass": passw})
        print(s)
        if(s):
            return
render_template("index.html")
        else:
            print("invalid")

    return render_template('login.html')

@app.route('/signup',methods=['GET','PO
ST'])
def signup():
    if(request.method=="POST"):
        name= request.form.get("name")
        email=request.form.get("email")
        passw=request.form.get("pass")
        cpass=request.form.get("cpass")
        if(passw==cpass):
            x=dbs.login.insert_one({"name":
name, "email": email, "pass": passw})
        if(x):
            return
render_template("login.html")
    return render_template("signup.html")

@app.route('/learn',methods=['GET','POS
T'])
```

```
def learn():
    if(request.method=="POST"):
        try:
            find=request.form.get("searchkey")
            if(find!=""):
                resu=search_wikihow(find,1)
                cont = io.StringIO()
                with
contextlib.redirect_stdout(cont):
                    resu[0].print()
                capt = cont.getvalue()
                print(capt)
                return
render_template("learn.html",output=capt)
        except:
            print("bypass")

    return render_template("learn.html")



@app.route('/quiz/<quizfield>',methods=['
GET','POST'])
def quiz(quizfield):
    Quiz= question(quizfield);
    if(request.method=="POST"):
        marks=0
        for ans in
Quiz.fivequestion()["questions"]:

if(ans['ans']==request.form[ans['name']]):
                marks=marks+1
        print(marks)
        if(marks<7):
            return
render_template("failure.html")
        else:
            return
render_template("success.html")
    questions = Quiz.fivequestion()
    return
render_template("quiz.html",questions=qu
estions)

@app.route('/internet',methods=['GET','P
OST'])
def internet():
```

```
    return render_template("internet.html")


@app.route('/phone',methods=['GET','PO
ST'])
def phone():
    if(request.method=="POST"):
        try:
            x=[]
            var = request.form.get("button")
            print(var)
            coll = dbs[var]
            cur = list(coll.find())
            print(cur[1])
            for i in cur:
                print(i)
            print(x)

            df = pd.DataFrame(cur)
            name_list = df["name"].tolist()
            des_list = df["des"].tolist()
            url_list = df["url"].tolist()
            print(len(name_list))
            return
render_template("phone.html",len=len(na
me_list) ,output=name_list, out1=des_list
,out2 = url_list, out3=var )
        except:
            print("bypass")

    return render_template("phone.html")

@app.route('/video',methods=['GET','POS
T'])
def video():
    if(request.method=="POST"):
        try:
            var = request.form.get("link")
            col_name =
request.form.get("colna")
            var1= request.form.get("vid")
            coll = dbs[col_name]
            cur = list(coll.find({"name":var1}))

            df = pd.DataFrame(cur)
            name_list = df["name"].tolist()
            des_list = df["des"].tolist()
```

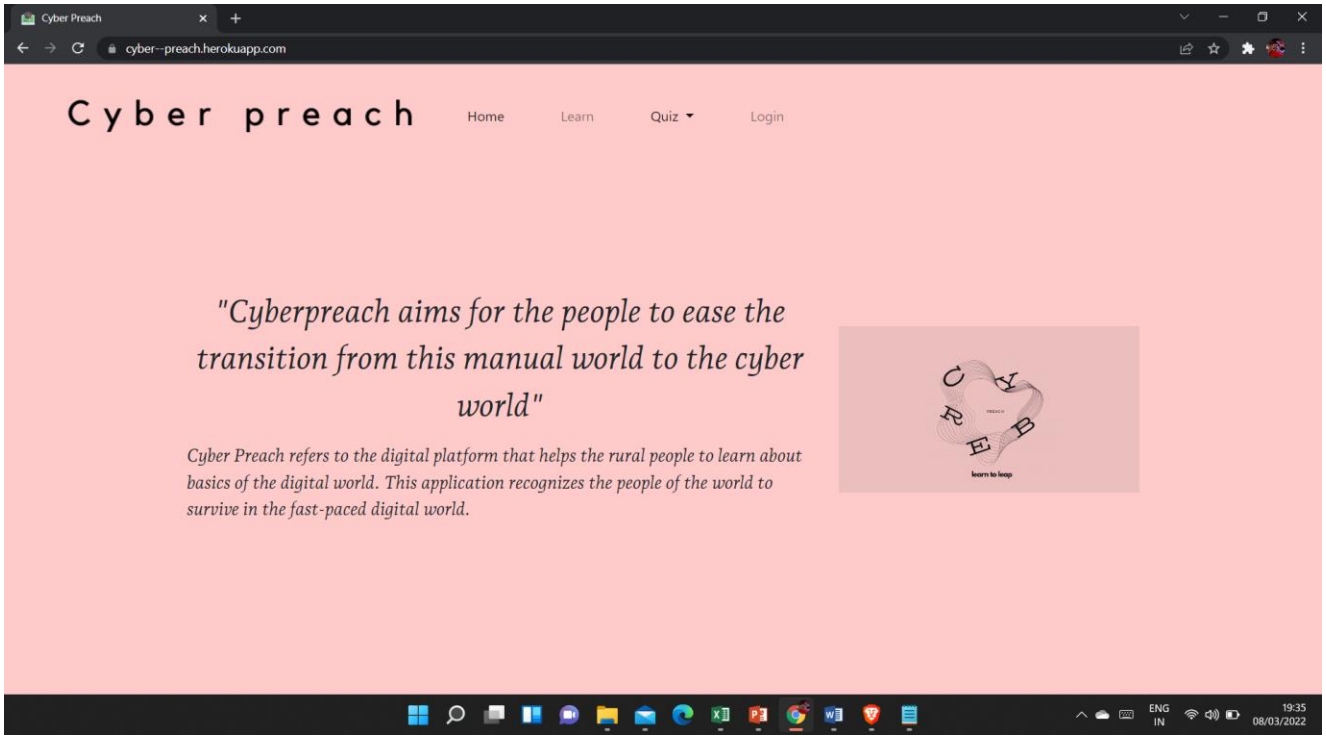
```
            url_list = df["url"].tolist()

            print(cur)
            print(len(name_list))

            seark = des_list[0]
            resu=search_wikihow(seark,1)
            cont = io.StringIO()
            with
contextlib.redirect_stdout(cont):
                resu[0].print()
            capt = cont.getvalue()
            print(capt)

            return
render_template("video.html",
output=name_list, out1=des_list , link=var,
capture = capt)
        except:
            print("by2")

    return render_template("video.html")


if __name__=="__main__":
    app.run(debug=True)
```

No SQL CODE: MongoDB

```
from pymongo import MongoClient
client = MongoClient()
client =
MongoClient('mongodb+srv://codesploit:codesploit@cluster0.xcehq.mongodb.net/test')
db = client['Cyberpreach']
collection = db['quiz']
class question():
    def __init__(self,question):
        self.question=collection.find_one({"name":question})
        self.quiz=[]
    def fivequestion(self):
        self.quiz=self.question
        return self.qui
```

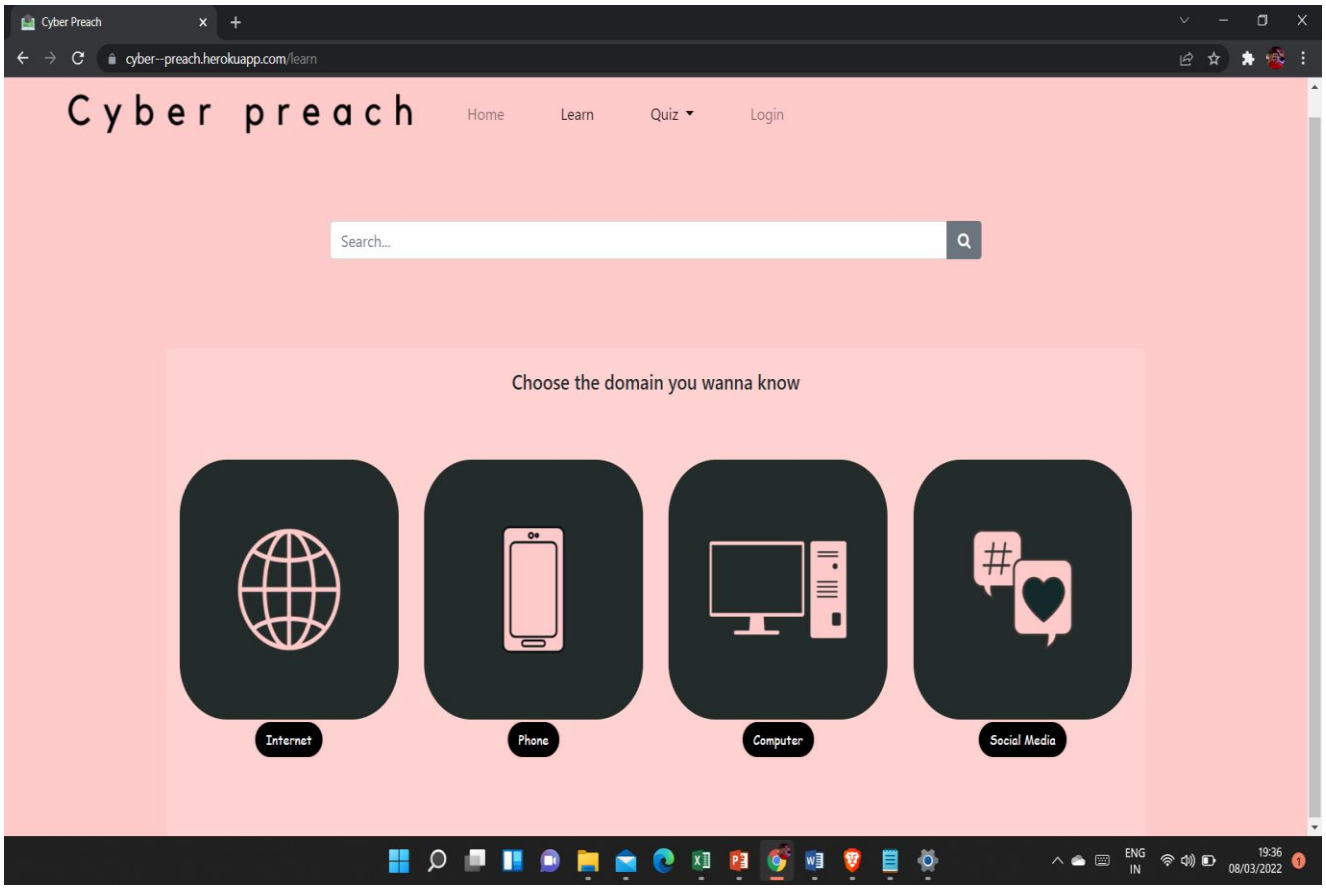
7.2 -

SCREENSHOTS:



Home page of our project cyber preach . Which deiles with the module like Home, Learn, Quiz, Login. The Cyber Preach also have the code “Cyberpreach aims for the people to ease the transition from this manual world to the cyber world”

Cyber Preach refers to the digital platform that helps the rural people to learn about basic of the digital world. This application recognizes the people of the world to survive in the fast – paced digital world.

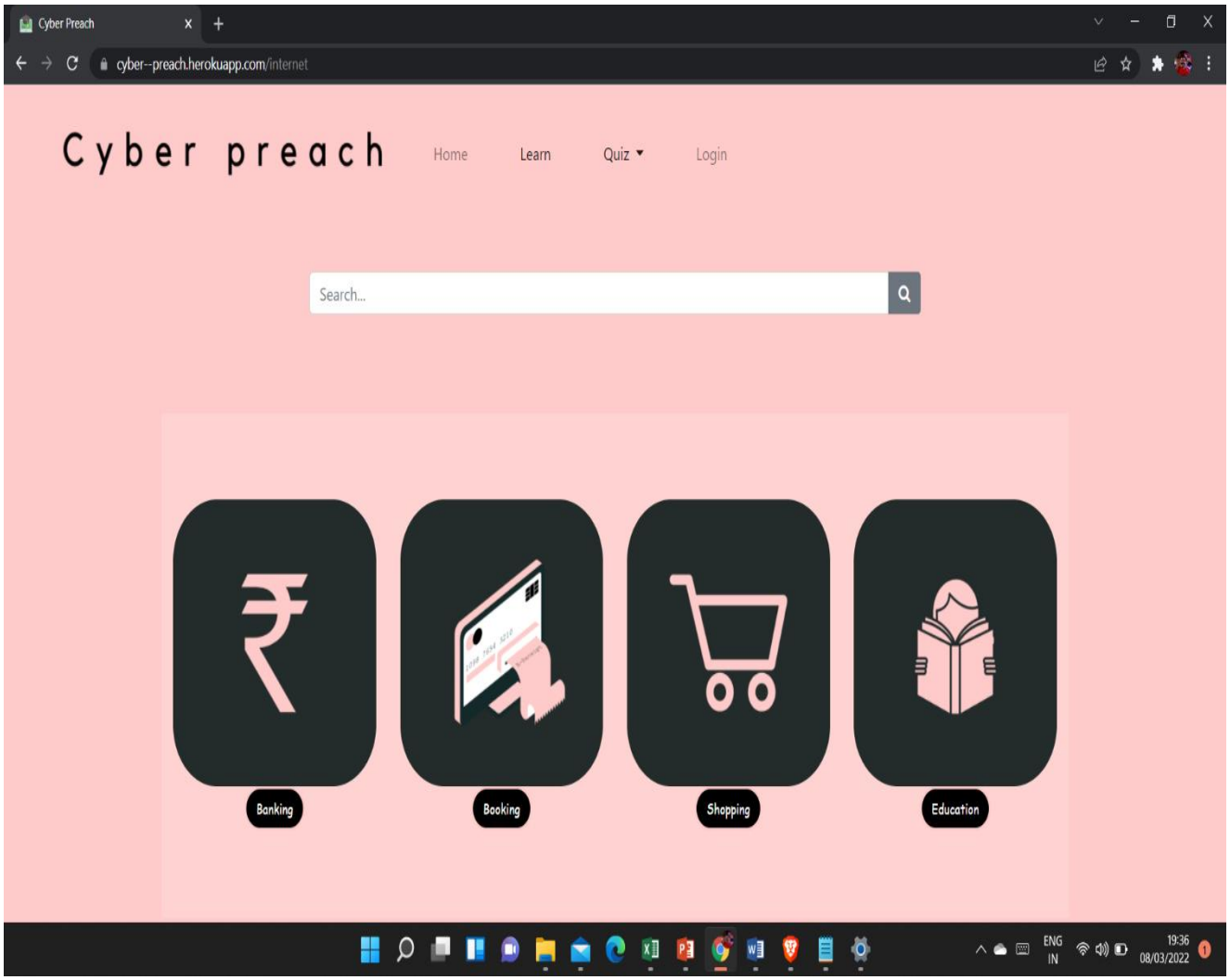


Learn module :

The learn module deals with some sub topics to learn about

- Internet
- Phone
- Computer
- Social Media

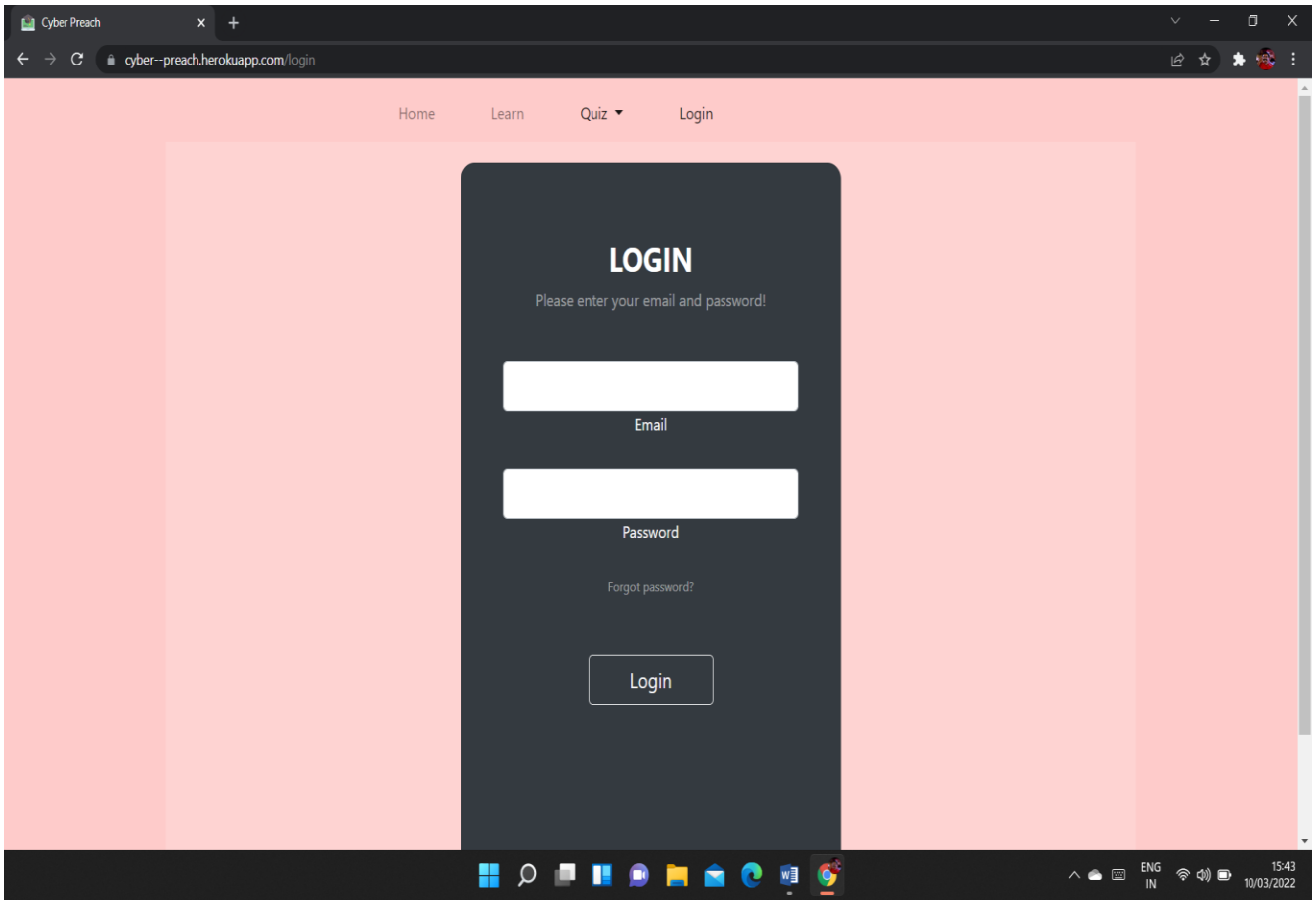
This are the some topics that will be provided by the cyber preach. Cyber Preach refers to the digital platform that helps the rural people to learn about basic of the digital world. This application recognizes the people of the world to survive in the fast – paced digital world



If we select the internet module the server will get you to another page which is shown in the above image

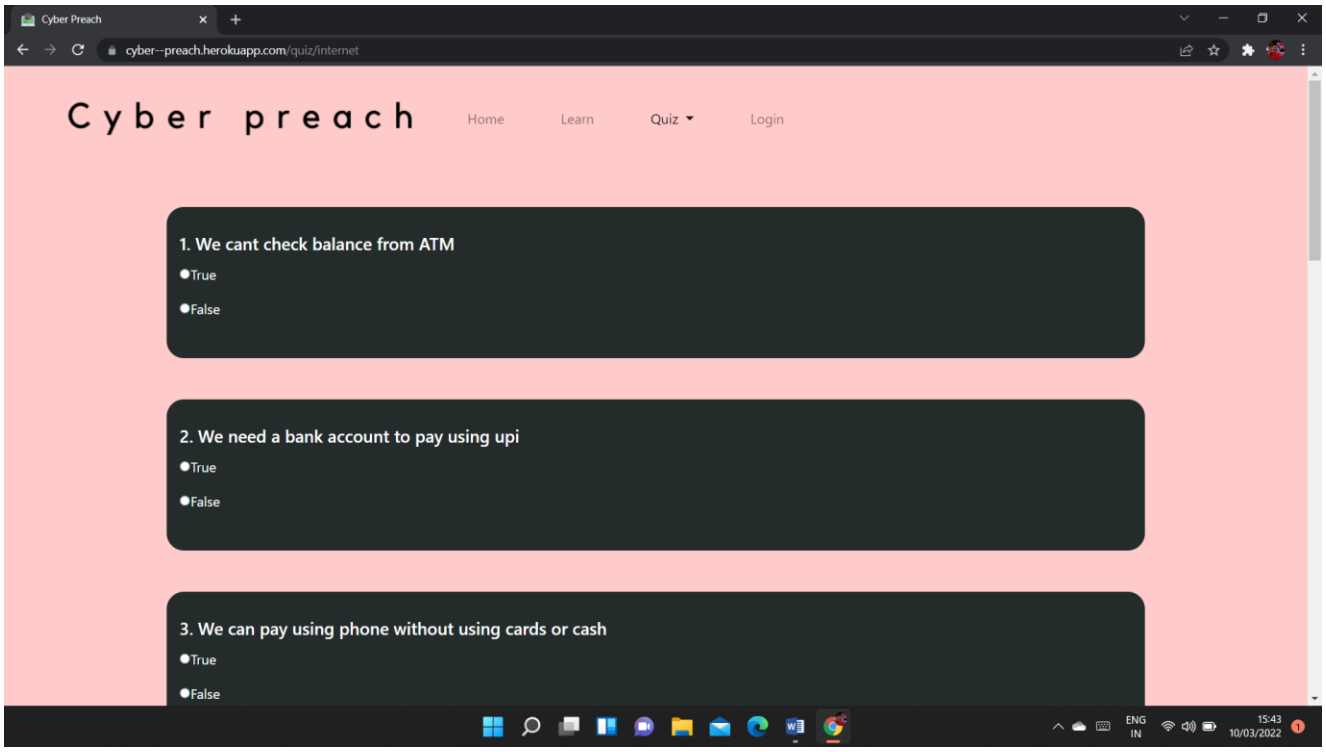
Internet Module :

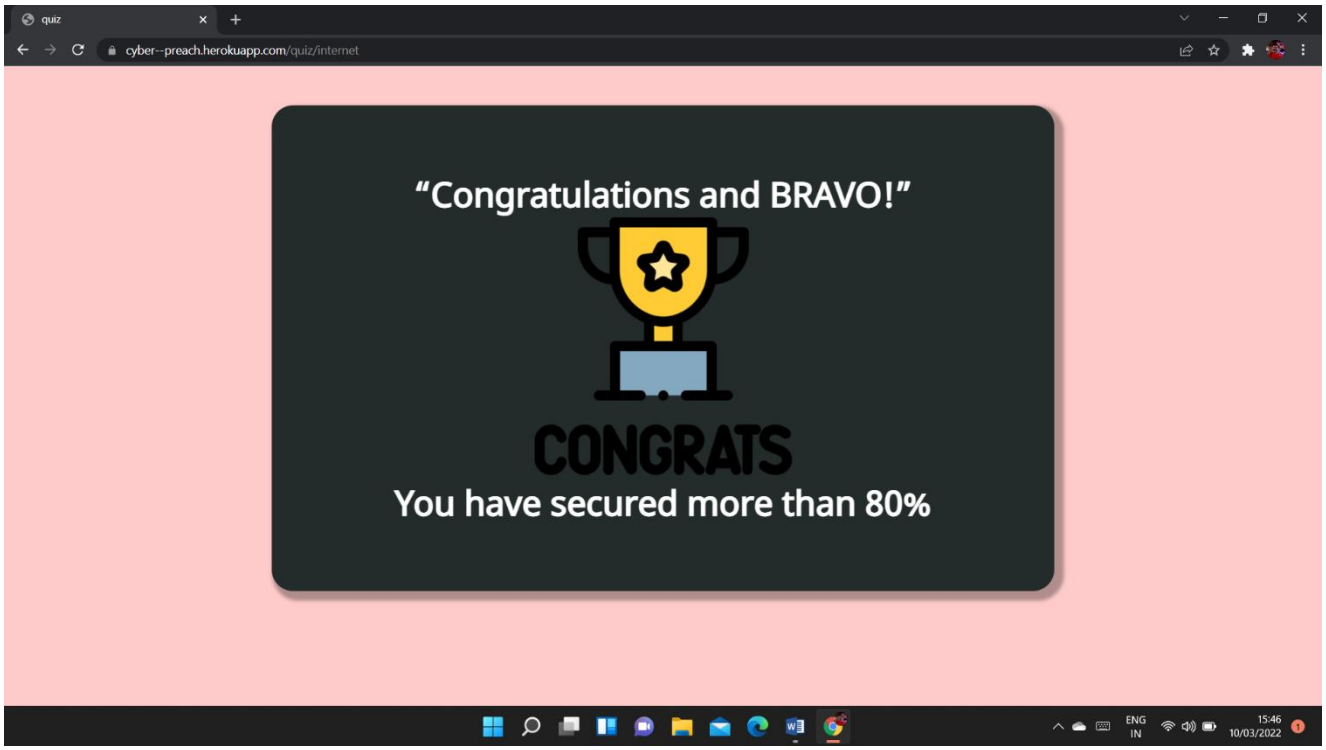
- Banking
- Booking
- Shopping
- Education



This is login module image shown above. The user can login through their email id for their security purpose. As we encrypted the code email login code which is provide by email itself for the startup a web site and login through the email. The code which is provided by the email is free for all. So, we used the same concept of login process for this web site.

If the email id and password is correct it will excute to next step of process.
If the user id and password is wrong the email server will not lest the user to move any more for there in this login process.


Cyber Preach
cyber--preach.herokuapp.com/quiz/internet
Cyber preach
Home
Learn
Quiz
Login
1. We cant check balance from ATM
True
False
2. We need a bank account to pay using upi
True
False
3. We can pay using phone without using cards or cash
True
False
ENG
IN
15:43
10/03/2022


quiz
cyber--preach.herokuapp.com/quiz/internet
“Congratulations and BRAVO!”
CONGRATS
You have secured more than 80%
ENG
IN
15:46
10/03/2022