



A

PROJECT REPORT ON

“ Virtual mouse using hand gesture detection”

SUBMITTED TO,

SHIVAJI UNIVERSITY, KOLHAPUR

FOR THE PARTIAL FULFILLMENT OF

BACHELOR OF COMPUTER APPLICATION BCA III (SEM VI)

SUBMITTED BY,

Mr. LAWAND SANKET SUHAS

Mr. ATTAR AMAN MALIK

Mr. DHANE ANKIT VINOD

Miss. SONAWANE SANIKA KAMALAKAR

UNDER THE GUIDANCE OF,

Mrs. T.N.SHEVTE

THROUGH

THE DIRECTOR,

RAYAT SHIKSHAN SANSTHA'S

“KARMVEER BHAURAO PATIL INSTITUTE OF

MANAGEMENT

STUDIES AND RESEARCH CENTER

VARYE, SATARA.”

YEAR

2022-2023



**Rayat Shikshan Sanstha's
Karmaveer Bhaurao Patil Institute of Management Studies And
Research, Satara.**

INSTITUTE RECOMMENDATION

This is to certify that, **Mr. LAWAND SANKET SUHAS, Mr. ATTAR AMAN MALIK, Mr. DHANE ANKIT VINOD, Miss. SONAWANE SANIKA KAMALAKAR**, are the bonafide student of this institute studying in BCA III(Semester-VI). They have completed the project report entitled “Virtual mouse using hand gesture detection” for the partial fulfilment of the programme for the academic year 2022-2023.

To the best of my knowledge and belief, this is original work and is not submitted earlier anywhere for any purpose.

Place: Satara

Date:

Dr. B. S. Sawant

(Director)



**Rayat Shikshan Sanstha's
Karmaveer Bhaurao Patil Institute of Management Studies And
Research, Satara.**

Certificate

This is to certify that project report entitled “Virtual mouse using hand gesture detection” submitted by **Mr. LAWAND SANKET SUHAS**. For the partial fulfilment of the requirement of the degree “BACHELOR OF COMPUTER APPLICATION” (Semester-VI) programme for the academic year 2022-2023. To the best of my knowledge and belief, this is original work and is not submitted earlier anywhere for any purpose.

Place: Satara

Date:

Mrs. T.N.Shevte

(Project Guide)

External Examiner

Dr. R. D. Kumbhar

(H.O.D)



**Rayat Shikshan Sanstha's
Karmaveer Bhaurao Patil Institute of Management Studies And
Research, Satara.**

Certificate

This is to certify that project report entitled “Virtual mouse using hand gesture detection” submitted by **Mr. ATTAR AMAN MALIK**. For the partial fulfilment of the requirement of the degree “BACHELOR OF COMPUTER APPLICATION” (Semester-VI) programme for the academic year 2022-2023. To the best of my knowledge and belief, this is original work and is not submitted earlier anywhere for any purpose.

Place: Satara.

Date

Mrs. T.N.Shevte

(Project Guide)

External Examiner

Dr. R. D. Kumbhar

(H.O.D)



**Rayat Shikshan Sanstha's
Karmaveer Bhaurao Patil Institute of Management Studies And
Research, Satara.**

Certificate

This is to certify that project report entitled "Virtual mouse using hand gesture detection" submitted by **Mr. DHANE ANKIT VINOD**. For the partial fulfilment of the requirement of the degree "BACHELOR OF COMPUTER APPLICATION" (Semester-VI) programme for the academic year 2022-2023. To the best of my knowledge and belief, this is original work and is not submitted earlier anywhere for any purpose.

Place: Satara.

Date:

Mrs. T.N.Shevte

(Project Guide)

External Examiner

Dr. R. D. Kumbhar

(H.O.D)



**Rayat Shikshan Sanstha's
Karmaveer Bhaurao Patil Institute of Management Studies And
Research, Satara.**

Certificate

This is to certify that project report entitled “Virtual mouse using hand gesture detection” submitted by **Miss. SONAWANE SANIKA KAMALAKAR**. For the partial fulfilment of the requirement of the degree “BACHELOR OF COMPUTER APPLICATION” (Semester-VI) programme for the academic year 2022-2023. To the best of my knowledge and belief, this is original work and is not submitted earlier anywhere for any purpose.

Place: Satara.

Date:

Mrs. T.N.Shevte
(Project Guide)

External Examiner

Dr. R. D. Kumbhar
(H.O.D)

DECLARATION

To
THE DIRECTOR,
KARMAVEER BHAURAO PATIL
INSTITUTE OF MANAGEMENT STUDIES AND RESEARCH,
VARYE, SATARA.

Respected sir,

We undersigned hereby declare that the project report entitled “Virtual mouse using hand gesture detection” has developed under the guidance of Mrs. T.N.Shevte and submitted to Shivaji University, Kolhapur for the partial fulfilment of the required of the degree “BACHELOR OF COMPUTER APPLICATION” (Semester-VI) which is written and submitted by us is our original work.

The content of this project report we have not copied from any source. We understand that any copying is able to punish in a way that university authorities deem fit.

Place: Satara

Date:

Yours Sincerely,

1. **Mr. LAWAND SANKET SUHAS.**
2. **Mr. ATTAR AMAN MALIK.**
3. **Mr. DHANE ANKIT VINOD.**
4. **Miss. SONAWANE SANIKA KAMALAKAR.**

ACKNOWLEDGEMENT

We take this opportunity to thank all those who have directly inspired us and helped us towards successful completion of this project report. We are grateful to our team for their highly encouraging and Co-operative attitude.

We express our sincere thanks to director Dr. B. S. Sawant, KBPIMSR, Satara for their encouragement throughout the project.

We heartily thank to our guide and HOD for encouragement and providing us with the best resources for completion of our project.

We are thankful to our project guide Mrs. T.N.Shevte for this valuable guidance. Her valuable suggestions and guidance have been very helpful in various phases of the completion of project. We express our sense of gratitude towards our friends and parent for their constant moral during our project report.

Place: Satara

Date:

Yours Sincerely,

Group Members:

- 1. Mr. LAWAND SANKET SUHAS**
- 2. Mr. ATTAR AMAN MALIK**
- 3. Mr. DHANE ANKIT VINOD**
- 4. Miss. SONAWANE SANIKA KAMALAKAR**

INDEX

Chapter No.	Particulars	Page No.
1.	Introduction to project: A. Introduction B. Need and Future Scope of System	10-13
2.	Proposed system A. Objectives- B. System Requirement a. Hardware b. Software	14-16
3.	System Analysis A. System Flow	17-22
4.	System Design A. Input Design B. Output Design	23-37
5.	Source Code	38-39
6.	Conclusion and Future Enhancement A. Conclusion B. Limitations	40-41
7.	Bibliography A. Books B. Web	42-43

Chapter 1

Introduction to project:

A. Introduction

B. Need and Future Scope of System

Introduction

This project is to develop a Virtual Mouse using Hand Gesture Recognition. Hand gestures are the most effortless and natural way of communication. The aim is to perform various operations of the cursor. Instead of using more expensive sensors, a simple web camera can identify the gesture and perform the action. It helps the user to interact with a computer without any physical or hardware device to control mouse operation.

Our proposed Project is the Hand gestures recognition system or Touch less interaction to replace the basic physical interaction pointing devices used in computer systems to reduce the limitations that stay due to the legacy systems such as mouse and Touchpad. The proposed system uses hand gesture, mostly hand region is physically raised within the region of Interest to perform various operations such as some basic and repetitive functions to access various applications like scroll up and down, left click and right click and free movement etc.

NEEDS

- A. The System eliminates the usage of devices, and it improves the human-computer interaction.
- B. It benefits everyone, especially paralysed people who have difficulty using a real mouse.

FUTURE SCOPE

- A. It reduces workspace & burden of extra hardware devices.
- B. It removes the burdens of devices, it brings the user and the workspace more clear.
- C. Less prone to physical damage and mechanical wear & tear.

Chapter 2

Proposed system

- A. Objectives-
- B. System Requirement
 - a. Hardware
 - b. Software

A. OBJECTIVE

- To implement virtual mouse using ML algorithm.
- To control mouse function through the hand.
- To perform particular mouse function such moving and clicking.

System Requirement

The following are minimum hardware and software requirements that should be present to run this project successfully.

- Hardware Requirement:

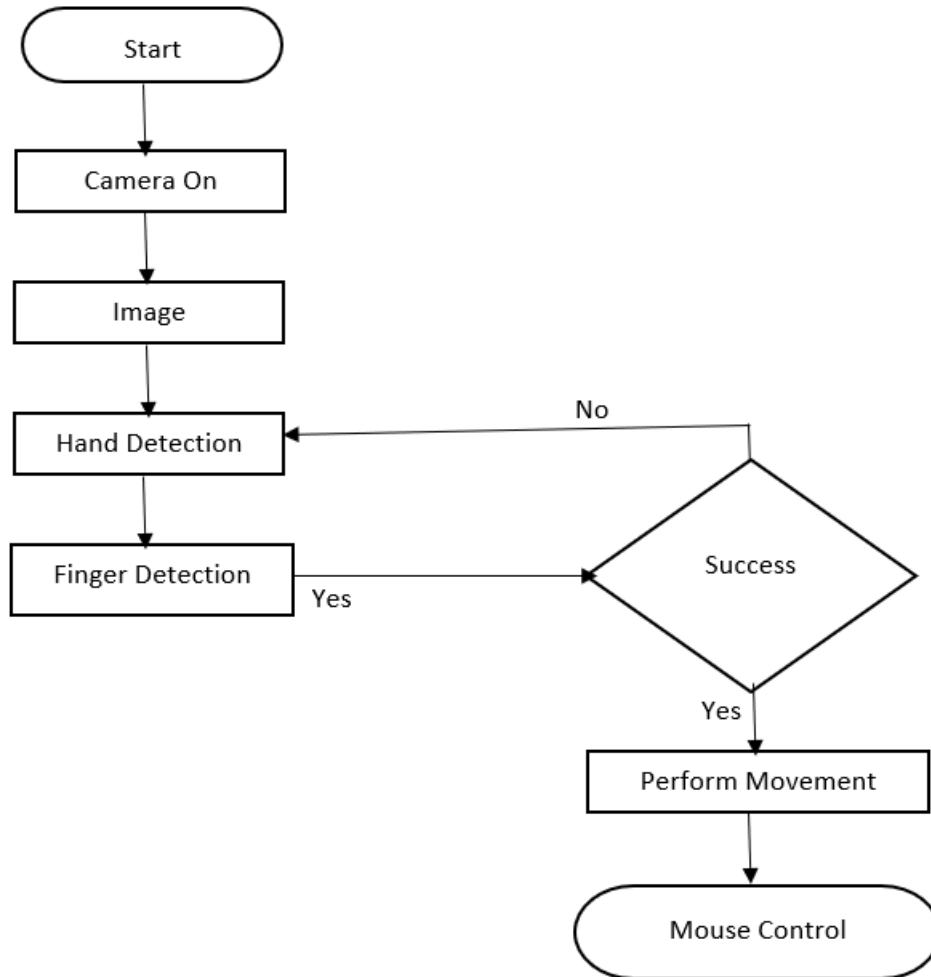
Processor:	Core i3 7 TH Gen
Hard Disk:	1 TB
RAM:	4 GB RAM.
Monitor:	14" Monitor

- Software Requirement:

Operating System:	Windows 10, 11(64 bit)
Programming Language:	Python .
Tool Used	Open CV

Chapter 3
System Analysis :
• **System Flow**

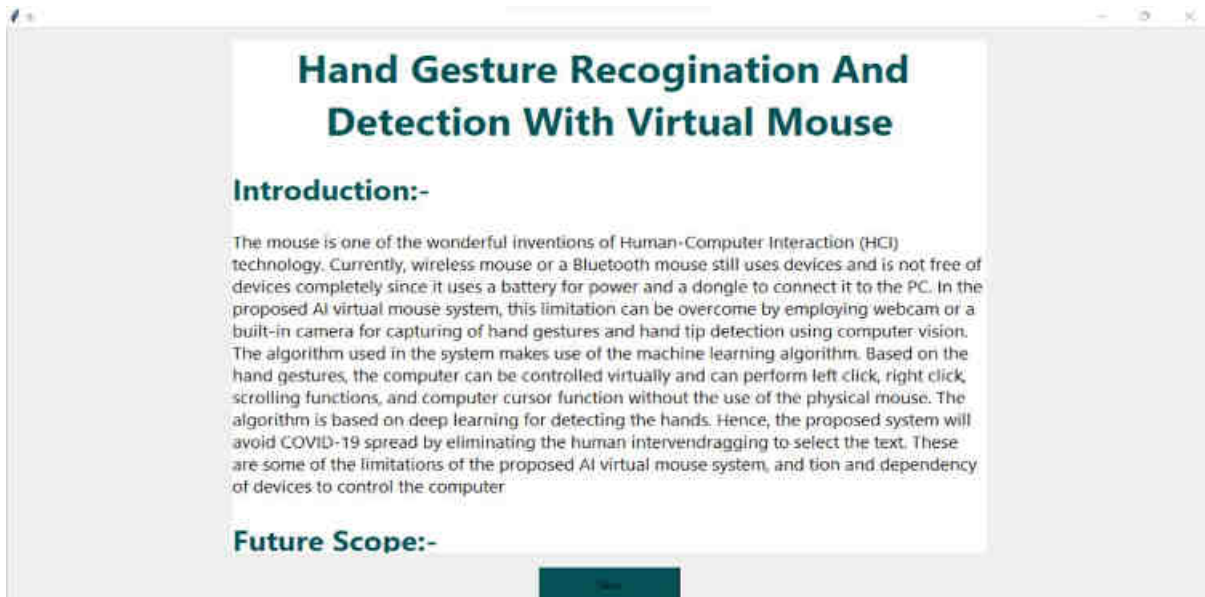
- System Flow :-



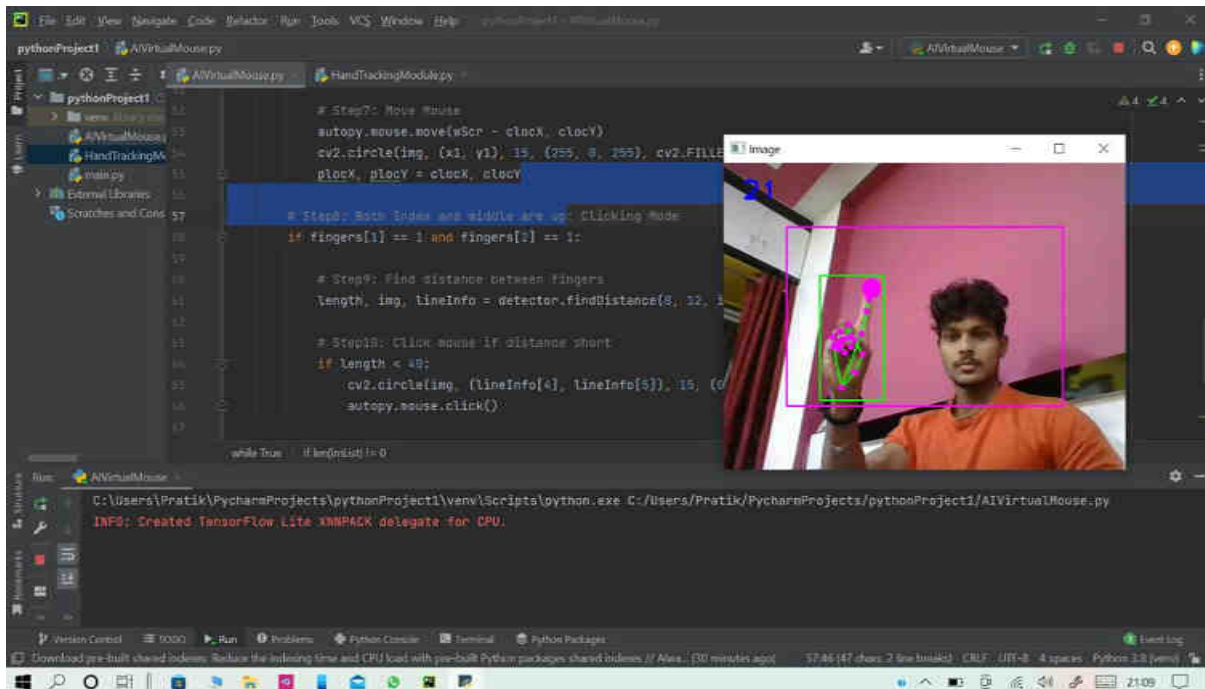
Chapter. 4
System Design

- **Output Design**

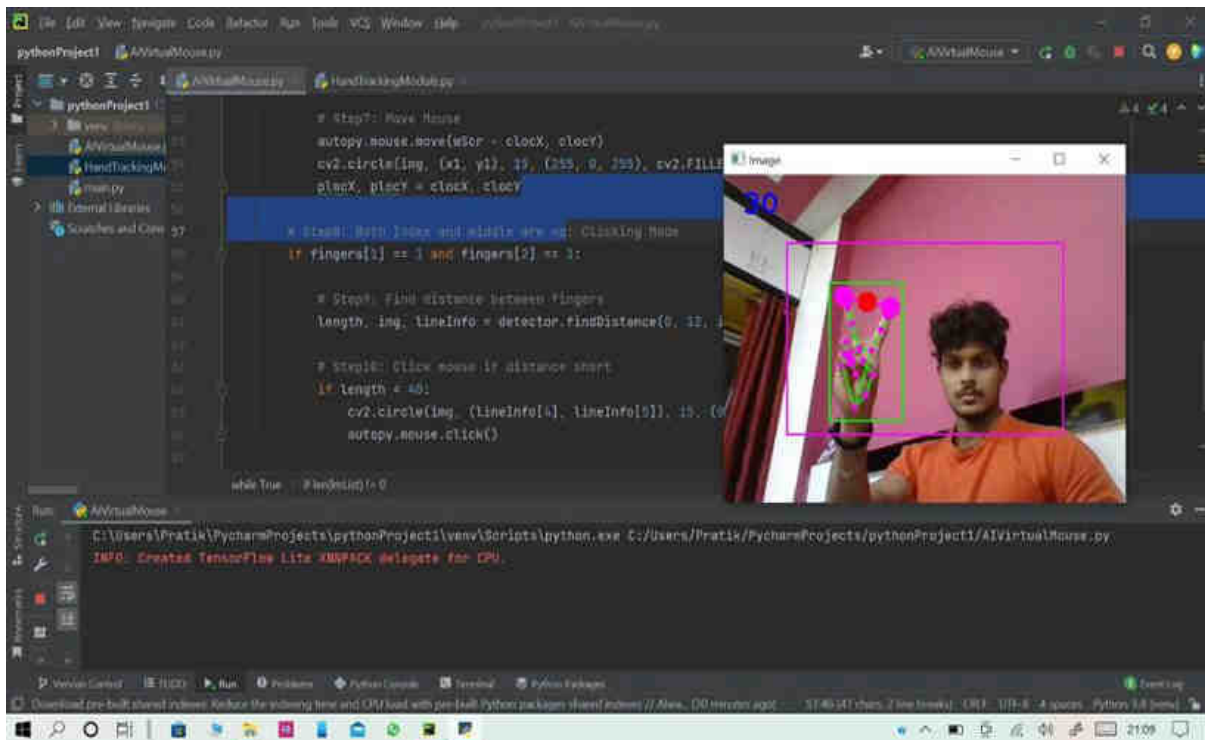
- **Homepage :-**



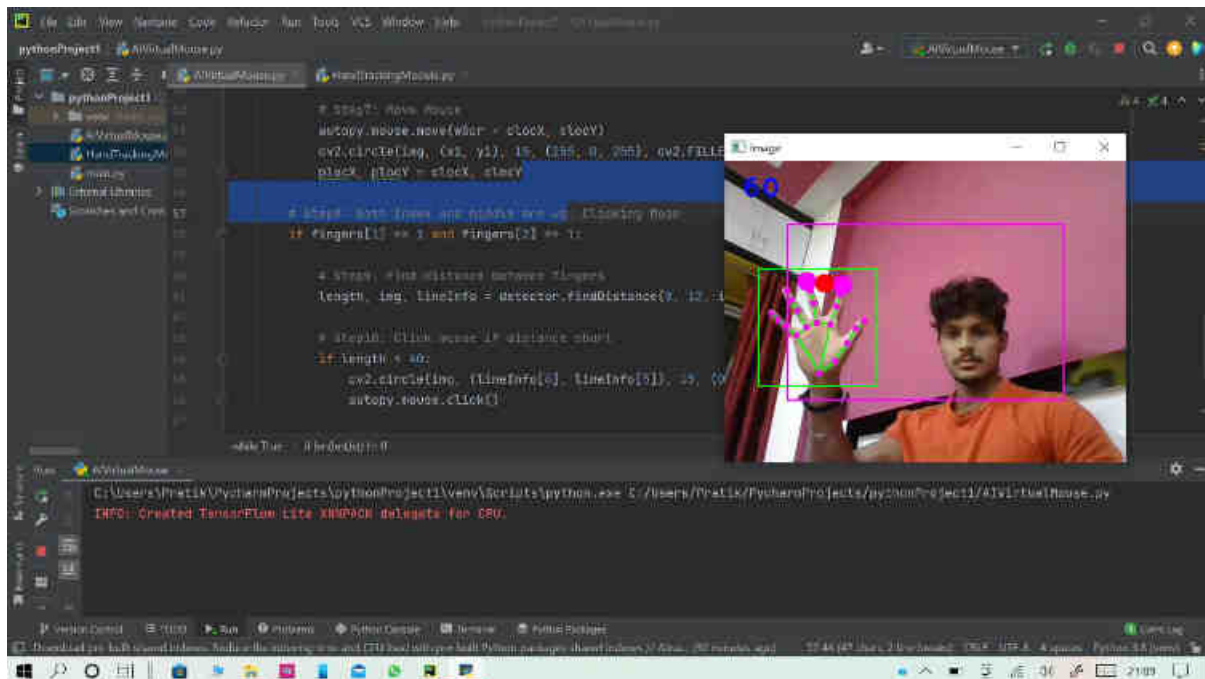
- **Movement of Cursor :-**



- Right Click :-



- Stop The Movement of cursor :-



Chapter. 5

- **Source Code**

- Home Page:-

```
# Import Module
import os
import sys
from tkinter import *
from tkhtmlview import HTMLLabel
from AIVirtualMouse import AIVirtualMouse
root = Tk()

root.geometry("500x500")
def run():
    os.system("AIVirtualMouse.py")
button =Button(root, text="Next",bg ="#045256",comma
nd= AIVirtualMouse , height=2 , width=20)
button.pack(side = "bottom",expand=True)

my_label = HTMLLabel(root, html="""
<div style = "background-color:#FFD181 ; justify:center; height:45px;
width:45px;" >
<div style = "background-color:#FFD181 ">
<div style = "background-color:#FFF ">
<h1 style="font-style: italic; color: #045256 ; text-align: center" >Hand
Gesture Recognition And Detection With Virtual Mouse </h1>
<h2 style="color:#045256;">Introduction:-</h2>
<p style="color:black; margin-right:20px;">The mouse is one of the
wonderful inventions of Human-Computer Interaction (HCI) technology.
Currently, wireless mouse or a Bluetooth mouse still uses devices and is
not free of devices completely since it uses a battery for power and a
dongle to connect it to the PC.
In the proposed AI virtual mouse system, this limitation can be overcome
by employing webcam or a built-in camera for capturing of hand gestures and
hand tip detection using computer vision. The algorithm used in the system
makes use of the machine learning algorithm. Based on the hand gestures,
the computer can be controlled virtually and can perform left click, right
click, scrolling functions, and computer cursor function without the use of
the physical mouse.
The algorithm is based on deep learning for detecting the hands.
Hence, the proposed system will avoid COVID-19 spread by eliminating the
human intervendragging to select the text. These are some of the
limitations of the proposed AI virtual mouse system, and \
tion and dependency of devices to control the computer
<h2 style="color:#045256;">Future Scope:- </h2>
</p>
<p>The proposed AI virtual mouse has some limitations such as small
decrease in accuracy of the right click mouse function and also
the model has some difficulties in executing clicking and
these limitations will be overcome in our future work.
Furthermore, the proposed method can be developed to handle the keyboard
functionalities along with
the mouse functionalities virtually which is another future scope of
Human-Computer Interaction (HCI).</p>
<h2 style="color:#045256">Applications:- </h2>
<p style="float:right">The AI virtual mouse system is useful for many
applications; it can be used to reduce the space for using the physical
mouse, and
it can be used in situations where we cannot use the physical mouse. The
system eliminates the usage of devices, and it improves the human-computer
```

interaction.

Major applications:

- (i) The proposed model has a greater accuracy of 99% which is far greater than the that of other proposed models for virtual mouse, and it has many applications
- (ii) Amidst the COVID-19 situation, it is not safe to use the devices by touching them because it may result in a possible situation of spread of the virus by touching the devices, so the proposed AI virtual mouse can be used to control the PC mouse functions without using the physical mouse
- (iii) The system can be used to control robots and automation systems without the usage of devices
- (iv) 2D and 3D images can be drawn using the AI virtual system using the hand gestures
- (v) AI virtual mouse can be used to play virtual reality- and augmented reality-based games without the wireless or wired mouse devices
- (vi) Persons with problems in their hands can use this system to control the mouse functions in the computer
- (vii) In the field of robotics, the proposed system like HCI can be used for controlling robots
- (viii) In designing and architecture, the proposed system can be used for designing virtually for prototyping

</div>

</div>

```
""", height="65", width="100")
```

```
my_label.pack(padx=15, pady=15)
```

```
root.mainloop()
```

- **Hand Tracking Module:-**

```
import cv2
import mediapipe as mp
import time
import math
import numpy as np

class handDetector():
    def __init__(self, mode=False, maxHands=2, detectionCon=0.5, trackCon=0.5):
        self.mode = mode
        self.maxHands = maxHands
        self.detectionCon = detectionCon
        self.trackCon = trackCon

        self.mpHands = mp.solutions.hands
        self.hands = self.mpHands.Hands(self.mode, self.maxHands, self.detectionCon, self.trackCon)
        self.mpDraw = mp.solutions.drawing_utils
        self.tipIds = [4, 8, 12, 16, 20]

    def findHands(self, img, draw=True):
        imgRGB = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
        self.results = self.hands.process(imgRGB)
```

```

# print(results.multi_hand_landmarks)

if self.results.multi_hand_landmarks:
    for handLms in self.results.multi_hand_landmarks:
        if draw:
            self.mpDraw.draw_landmarks(img, handLms,
self.mpHands.HAND_CONNECTIONS)
        return img

def findPosition(self, img, handNo=0, draw=True):
    xList = []
    yList = []
    bbox = []
    self.lmList = []
    if self.results.multi_hand_landmarks:
        myHand = self.results.multi_hand_landmarks[handNo]
        for id, lm in enumerate(myHand.landmark):
            # print(id, lm)
            h, w, c = img.shape
            cx, cy = int(lm.x * w), int(lm.y * h)
            xList.append(cx)
            yList.append(cy)
            # print(id, cx, cy)
            self.lmList.append([id, cx, cy])
            if draw:
                cv2.circle(img, (cx, cy), 5, (255, 0, 255), cv2.FILLED)

        xmin, xmax = min(xList), max(xList)
        ymin, ymax = min(yList), max(yList)
        bbox = xmin, ymin, xmax, ymax

        if draw:
            cv2.rectangle(img, (xmin - 20, ymin - 20), (xmax + 20, ymax
+ 20), (0, 255, 0), 2)

        return self.lmList, bbox

def fingersUp(self):
    fingers = []
    # Thumb
    if self.lmList[self.tipIds[0]][1] > self.lmList[self.tipIds[0] -
1][1]:
        fingers.append(1)
    else:
        fingers.append(0)
    # Fingers
    for id in range(1, 5):
        if self.lmList[self.tipIds[id]][2] <
self.lmList[self.tipIds[id] -2][2]:
            fingers.append(1)
        else:
            fingers.append(0)
    # totalFingers = fingers.count(1)
    return fingers

def findDistance(self, p1, p2, img, draw=True, r=15, t=3):
    x1, y1 = self.lmList[p1][1:]
    x2, y2 = self.lmList[p2][1:]
    cx, cy = (x1 + x2) // 2, (y1 + y2) // 2

    if draw:

```

```

        cv2.line(img, (x1, y1), (x2, y2), (255, 0, 255), t)
        cv2.circle(img, (x1, y1), r, (255, 0, 255), cv2.FILLED)
        cv2.circle(img, (x2, y2), r, (255, 0, 255), cv2.FILLED)
        cv2.circle(img, (cx, cy), r, (0, 0, 255), cv2.FILLED)
        length = math.hypot(x2 - x1, y2 - y1)

    return length, img, [x1, y1, x2, y2, cx, cy]

def main():
    pTime = 0
    cTime = 0
    cap = cv2.VideoCapture(1)
    detector = handDetector()
    while True:
        success, img = cap.read()
        img = detector.findHands(img)
        lmList, bbox = detector.findPosition(img)
        if len(lmList) != 0:
            print(lmList[4])
            cTime = time.time()
            fps = 1 / (cTime - pTime)
            pTime = cTime
            cv2.putText(img, str(int(fps)), (10, 70), cv2.FONT_HERSHEY_PLAIN,
3,
                        (255, 0, 255), 3)
            cv2.imshow("Image", img)
            cv2.waitKey(1)

if __name__ == "__main__":
    main()

```

- Virtual Mouse :-

```

import cv2
import numpy as np
import HandTrackingModule as htm
import time
import autopy
def AIVirtualMouse():
    wCam, hCam = 640, 480
    frameR = 100
    smoothening = 7

    pTime = 0
    plocX, plocY = 0, 0
    clocX, clocY = 0, 0
    cap = cv2.VideoCapture(0)
    cap.set(3, wCam)
    cap.set(4, hCam)

    detector = htm.handDetector(maxHands=1)
    wScr, hScr = autopy.screen.size()
    print(wScr, hScr)

    while True:

        success, img = cap.read()
        img = detector.findHands(img)

```



```

lmList, bbox = detector.findPosition(img)

if len(lmList) != 0:
    x1, y1 = lmList[8][1:]
    x2, y2 = lmList[12][1:]

    fingers = detector.fingersUp()
    cv2.rectangle(img, (frameR, frameR), (wCam - frameR, hCam -
frameR),
                    (255, 0, 255), 2)

    if fingers[1] == 1 and fingers[2] == 0:

        x3 = np.interp(x1, (frameR, wCam-frameR), (0, wScr))
        y3 = np.interp(y1, (frameR, hCam-frameR), (0, hScr))

        clocX = plocX + (x3 - plocX) / smoothening
        clocY = plocY + (y3 - plocY) / smoothening

        autopy.mouse.move(wScr - clocX, clocY)
        cv2.circle(img, (x1, y1), 15, (255, 0, 255), cv2.FILLED)
        plocX, plocY = clocX, clocY

    if fingers[1] == 1 and fingers[2] == 1:

        length, img, lineInfo = detector.findDistance(8, 12, img)

        if length < 40:
            cv2.circle(img, (lineInfo[4], lineInfo[5]), 15, (0,
255, 0), cv2.FILLED)
            autopy.mouse.click()
            # if fingers[1] == 1 or fingers[0] == 1:
            #     active = 1
            #     # print(mode)
            #     cv2.rectangle(img, (200, 410), (245, 460), (255, 255,
255), cv2.FILLED)
            #     if len(lmList) != 0:
            #         if fingers [0] == fingers[ 1]:
            #             autopy.scroll(300)
            #         if fingers [1] == fingers[1]:
            #             autopy.scroll(300)
            #         elif fingers[0] == fingers[0]:
            #             active = 0

        cTime = time.time()
        fps = 1/(cTime-pTime)
        pTime = cTime
        cv2.putText(img, str(int(fps)), (28, 58), cv2.FONT_HERSHEY_PLAIN,
3, (255, 8, 8), 3)

        cv2.imshow("Image", img)
        cv2.waitKey(1)

```

Chapter .6

Conclusion and Future Enhancement

A. Conclusion

B. Limitations

C. Future Enhancement

A. Conclusion

1. Redeploying the system to control mouse cursor using hand via camera
2. This team is this wrong computer vision algorithm and can do all mouse task
3. This system could be useful in industry level presentation and to reduce workspace
4. Features such as enlarging and shrinking windows closing windows etc by using palm and multiple fingers

B. Limitations

1. The proposed Virtual Mouse has some limitations such as small decrease in accuracy of the right click mouse function
2. The model has some difficulties in executing clicking and dragging to select the text.

B. Future Enhancement

We would improve the performance of the software especially hand tracking in the near future. And we also want to decrease the response time of the software for cursor movement so that it can completely be used to replace our conventional mouse. We are also planning to design a hardware implementation for the same in order to improve accuracy and increase the functionality to various domains such as a gaming controller or as a general purpose computer controller. Other advanced implementation include the hand gesture recognition stage to use the Template Matching method to distinguish the hand gestures. This method requires the use of a machine learning classifier, which takes a considerably long time to train and develop. However, it would allow the use of lots more hand gestures which in turn would allow the use of more mouse functions such as zoom in and zoom out. Once the classifier is well trained, the accuracy of the Template Matching method is expected to be better than the method used in the proposed design.

Chapter. 7

Reference :-

A) Books

B) Web

- **Web:**

1. ▪ www.tripadvisor.com
2. ▪ www.expedia.com
3. ▪ www.kayak.com
4. ▪ www.hotwire.com/Travel-Car-Rentals

- **Books:**

1. Virtual mouse using hand gesture.
2. A review of human–computer interaction and virtual reality research fields in cognitive Info Communications.
3. Cursor control using hand gestures.



A

PROJECT REPORT ON

“Android Based Attendance System”

SUBMITTED TO,

SHIVAJI UNIVERSITY, KOLHAPUR

FOR THE PARTIAL FULLFILMENT OF

BACHELOR OF COMPUTER APPLICATION BCAIII (SEM VI)

SUBMITTED BY,

Manthan Jitendra Pandit

Devraj Nilkanth Shinde

Pradnya Bajirav Gadhawe

UNDER THE GUIDANCE OF,

Mrs. T. N. Shevate

THROUGH

THE DIRECTOR,

RAYAT SHIKSHAN SANSTHA'S

KARMAVEER BHAURAO PATIL INSTITUTE OF

MANAGEMENT STUDIES AND RESEARCH,

VARYE, SATARA.



RAYAT SHIKSHAN SANSTHA'S
KARMAVEER BHAURAO PATIL INSTITUTE OF
MANAGEMENT STUDIES AND RESEARCH, VARYE SATARA.

INSTITUTE RECOMMENDATION

This is to certify that, **Mr. Manthan Pandit, Mr.Devraj Shinde, Miss. Pradnya Gadhave** are the benefited students of this institute studying in **B.C.A.III(SEMESTER-VI)**. They have completed the project report entitled "Android Based Attendance System" for the partial fulfilment of the requirement of the degree "BACHELOR OF COMPUTER APPLICATION (SEMESTER VI)" programme for the academic year 2022-2023.

To the best of my knowledge and belief, the original work and not submitted earlier anywhere for any purpose.

Place: Satara

Date:

Dr. B. S. Sawant
(Director)



RAYAT SHIKSHAN SANSTHA'S
KARMAVEER BHAURAO PATIL INSTITUTE OF
MANAGEMENT STUDIES AND RESEARCH, VARYE SATARA.

CERTIFICATE

This is to certify that, project report entitled “Android Based Attendance System” submitted by **Mr. Manthan Jitendra Pandit**. For the partial fulfilment of the requirement of degree “**BACHELOR OF COMPUTER APPLICATION**” (Semester VI) for the academic year 2022-2023 to the best of my knowledge and belief, the original work and not submitted earlier anywhere for any purpose.

Place: Satara

Date:

Mrs. T. N. Shevate
(Project Guide)

External Examiner

Dr.R.D.Kumbhar
(H.O.D)



RAYAT SHIKSHAN SANSTHA'S
KARMAVEER BHAURAO PATIL INSTITUTE OF
MANAGEMENT STUDIES AND RESEARCH, VARYE SATARA.

CERTIFICATE

This is to certify that, project report entitled “Android Based Attendance System ” submitted by **Mr. Devraj Nilkanth Shinde**. For the partial fulfilment of the requirement of degree “**BACHELOR OF COMPUTER APPLICATION**” (Semester VI) for the academic year 2022-2023 to the best of my knowledge and belief, the original work and not submitted earlier anywhere for any purpose.

Place: Satara

Date:

Mrs. T. N. Shevate
(Project Guide)

External Examiner

Dr.R.D.Kumbhar
(H.O.D)



RAYAT SHIKSHAN SANSTHA'S
KARMAVEER BHAURAO PATIL INSTITUTE OF
MANAGEMENT STUDIES AND RESEARCH, VARYE SATARA.

CERTIFICATE

This is to certify that, project report entitled “Android Based Attendance System” submitted by **Miss. Pradnya Bajirav Gadhave**. For the partial fulfillment of the requirement of degree “**BACHELOR OF COMPUTER APPLICATION**” (Semester VI) for the academic year 2022-2023 to the best of my knowledge and belief, the original work and not submitted earlier anywhere for any purpose.

Place: Satara

Date:

Mrs. T. N. Shevate
(Project Guide)

External Examiner

Dr.R.D.Kumbhar
(H.O.D)

DECLARATION

To,

The Director,

KARMAVEER BHAURAO PATIL

INSTITUTE OF MANAGEMENT STUDIES AND
RESEARCH VARYE, SATARA

Respected Sir,

We undersign hereby declare that the project report entitled “**Android Based Attendance System**” has developed under the guidance of **Mrs. T. N. Shevate** & submitted to Shivaji University, Kolhapur for the partial fulfilment of the requirement of degree “BACHELOR OF COMPUTER APPLICATION”(Semester VI) which is written and submitted by us is our original work. The content of this project we have not copied from any source. We understand that any copying is able to punish in a way that university authorities deem fit.

Place: Satara

Date:

Yours Sincerely,

- 1) **Mr. Manthan Jitendra Pandit**
- 2) **Mr. Devraj Nilkanth Shinde**
- 3) **Miss. Pradnya Bajirav Gadhave**

ACKNOWLEDGMENT

We take this opportunity to thank all those who has directly inspired, director and helped us towards successful completion of this project report. We are grateful to our team for their highly encouraging and co-operative attitude.

We express our sincere thanks to director Dr. B.S. Sawant, KBPIMSR, Satara for his encouragement throughout the project report. We are thankful to our project guide Mrs. T. N. Shevate for this valuable guidance. We express our sense of gratitude towards our friends and parents for their constant moral during our project report.

Place: Satara

Date:

Yours Sincerely,

Group Members

- 1) **Mr. Manthan Jitendra Pandit**
- 2) **Mr. Devraj Nilkanth Shinde**
- 3) **Miss. Pradnya Bajirav Gadhave**

Index

Chapter No	Topic	Page No
1	<u>Introduction to Project</u> A. Introduction B. Existing System C. Need and scope of System	<u>9-12</u>
2	<u>Proposed System</u> A. Advantages B. Objectives C. SRS	<u>13-18</u>
3	<u>System Analysis</u> A. ERD(Entity relational Diagram) B. DFD(Data flow diagram)	<u>19-23</u>
4	<u>System Requirements</u> A. Hardware B. Software	<u>24-25</u>
5	<u>System Design</u> A. Database Designs B. Input Designs C. Output designs	<u>26 -35</u>
6	<u>User Guideline</u> A. Installation process	<u>36-37</u>
7	<u>Source Code</u>	<u>38-55</u>
8	<u>Outputs</u> A. Reports	<u>56-59</u>
9	<u>Conclusion and Suggestions</u> A. Conclusion and suggestions B. Future enhancement C. Bibliography	<u>60-63</u>

Chapter 1

Introduction to Project

- A. Introduction**
- B. Existing System**
- C. Need and Scope of System**
- D. Organizational Profile**

Introduction to project:

A. Introduction:

Android Based Attendance System is a convenient and efficient way for students to track their attendance in college classes. By scanning a QR code provided by their teacher, students can quickly and easily check in to their class. This app eliminates the need for manual sign-sheets and allows faculty to track attendance in real-time. With the QR Attendance College app, both students and faculty can save time and update the attendance process.

A QR code attendance system is a digital method of tracking attendance using QR codes. This system utilizes mobile devices, such as smartphones, to scan QR codes displayed on a screen or printed on a piece of paper. When a code is scanned, the device sends the code's information to a Firebase, which ensures the student's presented in class. After entering in class student scans Bar-code provided to them and marks their attendance in class.

The system can be used in a classroom, conference room, or any other location where attendance needs to be taken. A teacher or administrator can generate a unique QR code for each session, which can be displayed on a screen or printed on a handout. Students can then use their mobile devices to scan the code, which sends their information to a Firebase. The system also allows teachers to generate an attendance report in PDF format at the end of the day.

The system can also be configured to include additional features, such as the ability to track attendance in real-time, generate reports.

Overall, Android Based Attendance System offer a quick and easy way to track attendance and can help improve accuracy and efficiency. It also eliminates manual attendance process and reduce human error.

B. Existing System:

In schooling and academics, the traditional way of taking of attendance is by calling students name or numbers. Teachers will typically call out students' names or numbers and students will respond to confirm that they are present. This method is simple and straightforward, but it can be time-consuming and may not be the most efficient way of taking attendance, especially in larger classes.

C. Need and Scope of System:

Need:

Nowadays life is exceptionally quick and work with the smartphone is fast. The existing framework was physically on paper. So it is time devouring as compared with our framework a man cannot do work as quick as computer.

In existing framework paper work is more where as in our framework on smartphone, so most of the work is done on smartphone and it give offices to alter in the event that any botch happened. So ready to say that in existing framework paper work is more. All the data is on paper in existing framework so, there are conceivable outcomes of misfortune, robbery, burning and harming information on paper.

Where as in smartphone system all information are spared and reinforcement of information is additionally taken in case client need. So there are less chances of information misfortune at that point the existing framework. Existing framework is badly designed for client since it cannot deliver quick and appropriate data in legitimate time and it take long time to donate data to the client. In our framework the data is created exceptionally rapidly and arranged appropriately to the client ask.

Scope:

The scope of a Android-based QR code attendance app for an institute can include the following:

- Attendance tracking: The app can be used to track attendance for classes, lectures, seminars, and other events.
- QR code generation: The app can generate unique QR codes for each class, which can be distributed digitally to students.

- Attendance monitoring: The app can provide real-time monitoring of attendance, allowing institutes to track attendance in near real-time
- Reports and analytics: The app can generate various reports and analytics, such as attendance records and absentees, that can be useful for institutes to track attendance data.
- Contactless attendance tracking: The app allows for contactless attendance tracking.

Chapter 2

Proposed System

- A. Advantages
- B. Objectives
- C. Requirement Gathering
- D. SRS

A. Advantages:

Convenience and ease of use: Android Based Attendance System are quick and easy to use, as all that is required is for the student to scan a code with their smartphone.

Real-time tracking: Attendance data is immediately synced to the system, providing real-time tracking of attendance for the institute.

Reduced human error: Android-based QR code attendance app can reduce human error associated with manual attendance tracking methods such as sign-in sheets.

Increased security: QR codes can be generated and distributed in a secure manner, reducing the risk of attendance fraud.

Cost-effective: Android-based QR code attendance app are relatively inexpensive to set up and maintain, and can save time and resources compared to traditional attendance tracking methods.

B. Objectives:

The main objective of the project on Android Based Attendance System is to manage the Attendance of students. The project is totally built at faculty and Student side. The purpose of the project is to build an application to reduce the manual work of managing the work of marking attendance and taking attendance and manage attendance report.

The main objectives of a Android-based QR code attendance app are:

- To update the attendance process for both students and faculty.
- To remove the need for manual sign-sheets or roll calls.
- To allow faculties to track attendance in real-time.
- To save time and improve efficiency.

- To provide a convenient and easy-to-use solution for tracking attendance.
- To encourage regularity and attendance in college classes.

C. SRS:

The purpose of the Software Requirements Specification (SRS) document in app development with Java is to clearly define the requirements and specifications for the app. The SRS document serves as a guide for the development team, outlining the functional and non-functional requirements for the app, as well as any constraints or limitations that must be considered during the development process.

Introduction:

This project work is undertaken in context of partial fulfilment of BCA-III SEM-VI. We have created an Android based attendance system containing android application for the institute to record attendance. It is a convenient and efficient way for students to track their attendance in college classes. By scanning a QR code provided by their teacher, students can quickly and easily check in to their class. This app eliminates the need for manual sign-sheets and allows faculty to track attendance in real-time.

Scope:

There are potential future scopes, and they will be evaluated based on the market and user demand, as well as the resources and budget available for the project.

1) Overview:

The app is specifically designed for Android devices and utilizes the Java programming language. The app is designed to improve attendance accuracy and reduce administrative workload, and it will be tested for compatibility and performance on Android devices to ensure optimal performance.

2) Overall Description:

A) App Description:

The app displays a QR code on the user's device, which can be scanned by a scanner at the location. The scan confirms the user's attendance and records the time of arrival or departure. The app can also display a record of the user's attendance history and provide real-time updates on attendance status.

B) App Function:

- 1) QR code class creation: The app would allow the user to create a new class by generating a unique QR code that represents that class. The QR code can be scanned by students to join the class.
- 2) Barcode attendance marking: Once the students have joined the class, the app would allow the teacher to mark attendance by scanning the barcode on the student's ID card. The app would then record the student's attendance for that class.
- 3) Attendance records: The app would keep track of attendance records for each class, allowing the teacher to easily view attendance for a specific class or for all classes.
- 4) Real-time attendance tracking: The app would provide real-time attendance tracking, allowing the teacher to see which students are present in the class at any given time.

C) Assumption and Dependencies:**Assumptions:**

The app will be used by teachers and students in a classroom setting.

The app will be accessible on smartphones or tablets.

The app will have the capability to create and scan QR and bar codes.

The app will have the ability to record and store attendance data.

Dependencies:

An internet connection is required to access the app and upload attendance data.

The app will need to be compatible with the device it is being used on.

The app will need to have the necessary permissions to access the device's camera for scanning QR and bar codes.

The app will need to have a user login system to ensure secure access to attendance data.

3) Specific Requirements:

I) External interface requirements:

a. User Interface:

Splash screen: A welcome screen with the app's logo and a loading animation.

Login screen: A screen where the user can enter their login credentials (username and password) to access the app.

Main menu: A screen displaying the main options available in the app, such as creating a new class, marking attendance, and viewing class reports.

Mark attendance screen: A screen where the user can scan a QR code or a barcode to mark a student's attendance. The app will display the student's name and the class they are attending.

Class report screen: A screen where the user can view a report of attendance for a specific class, including the number of students present, absent, and late. The report can be exported as a PDF file for further analysis.

Logout button: A button located in the main menu that allows the user to log out of the app and return to the login screen.

II) Functional Requirements:

QR Code Creation: The app should have the capability to generate unique QR codes for each class, which can be scanned by students to mark their attendance.

Barcode Scanning: The app should have the capability to scan barcodes, which can be used by students to mark their attendance in the class.

Attendance Tracking: The app should be able to track and record the attendance of each student in a class, including the date and time of the attendance.

Student Management: The app should be able to manage student information, including their names, ID numbers, and class schedules.

Reports and Analytics: The app should be able to generate reports and analytics on attendance, class schedules, and student information.

User-friendly Interface: The app should have a user-friendly interface that is easy to navigate and use.

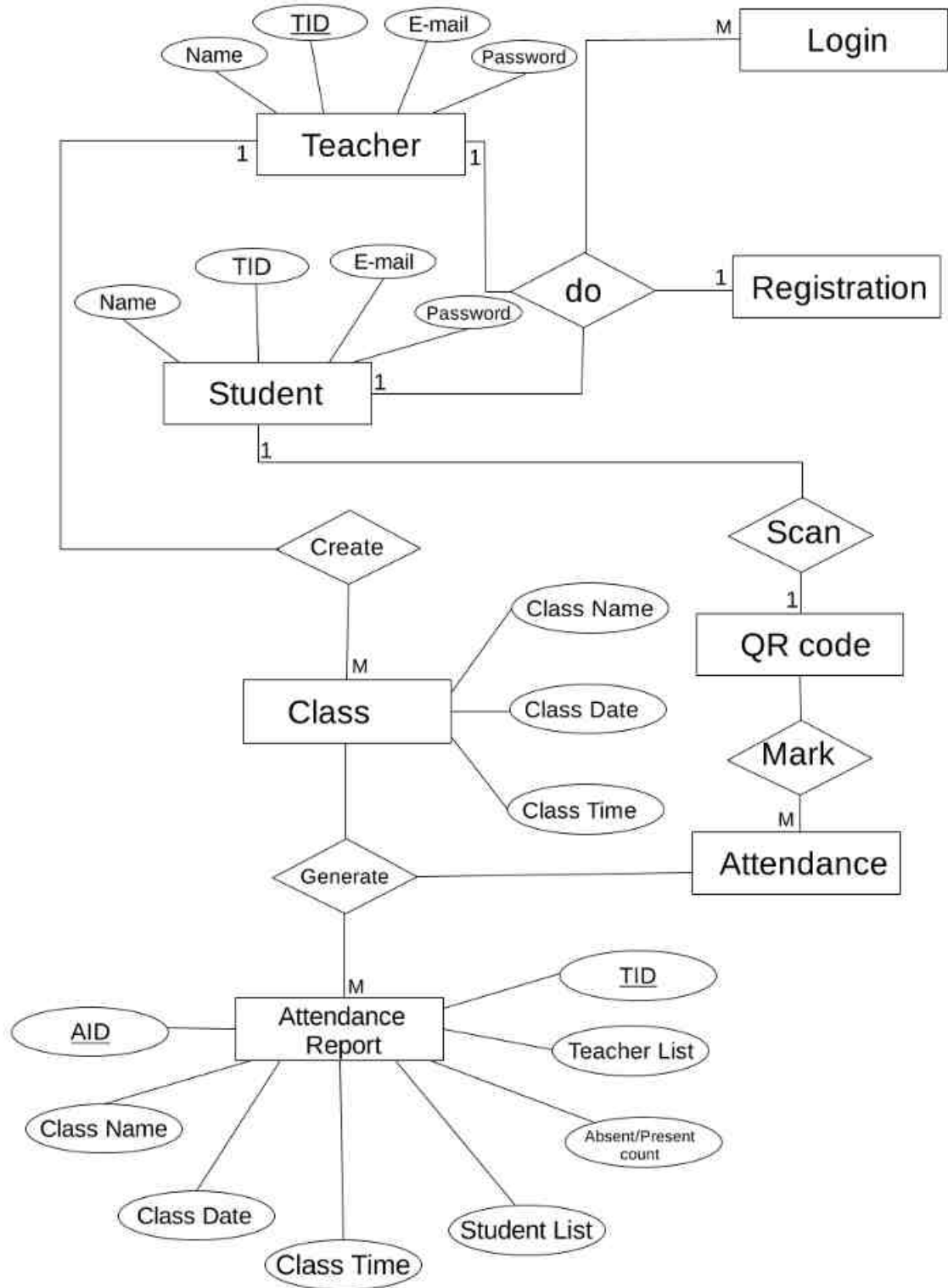
Mobile Compatibility: The app should be compatible with mobile devices, allowing students and teachers to access and use the system on the go.

Chapter 3

System Analysis:

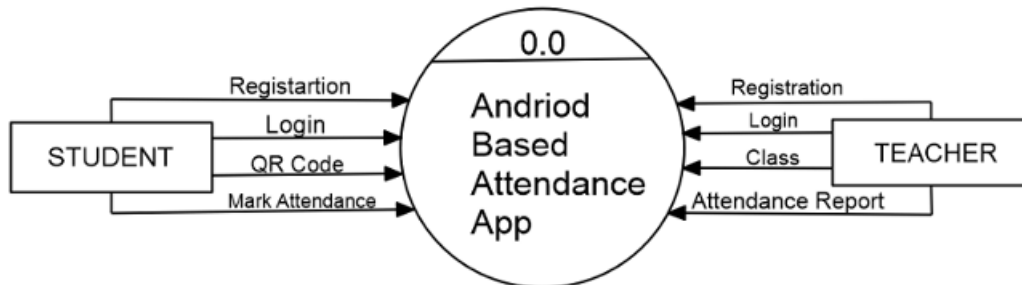
- i. ERD (Entity Relationship Diagram)
- ii. DFD(Data Flow Diagram)

ERD

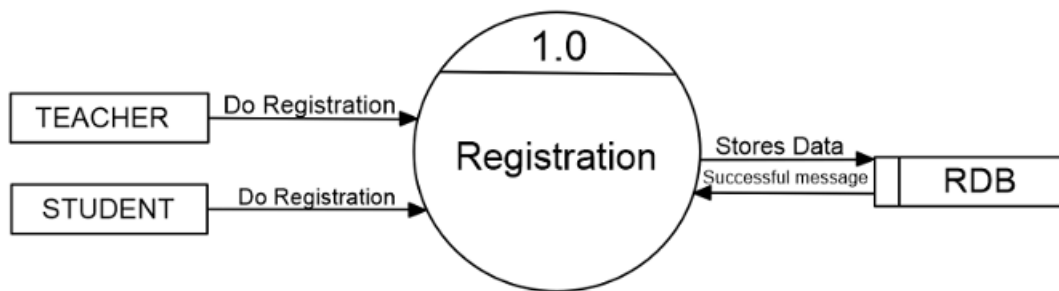


DFD:

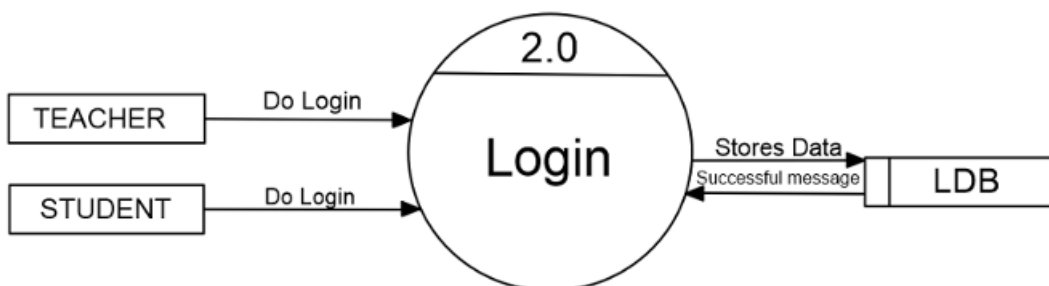
0th Level :



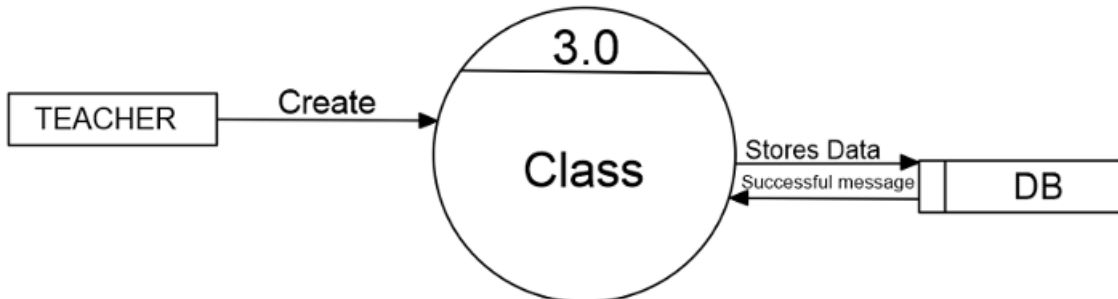
1st Level :



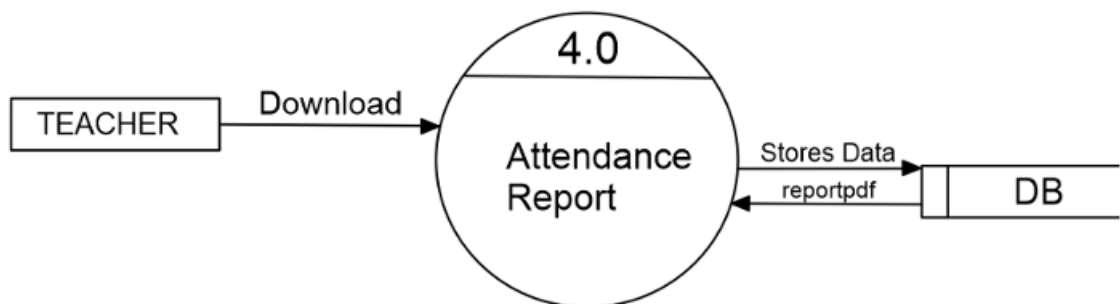
2nd level:



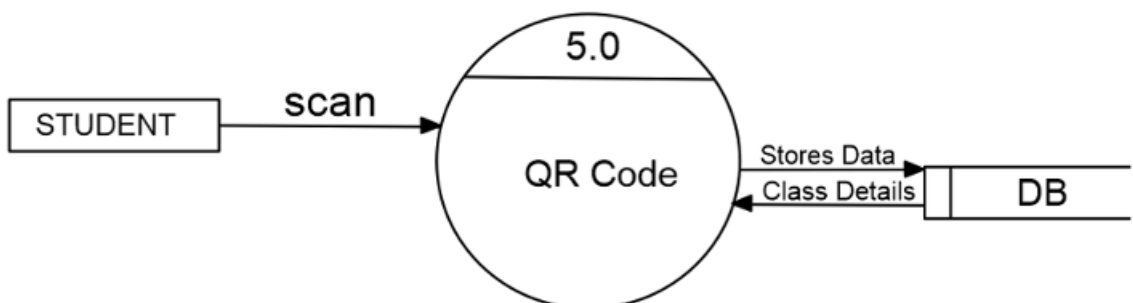
3rd level:



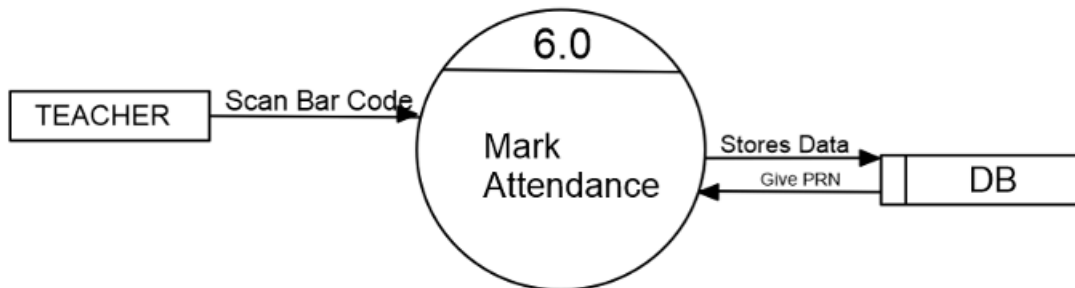
4th level:



5th level:



6th level:



Chapter 4

Implementation:

System Requirement

Hardware

Software

System Requirement

Hardware Requirement:

- Android Smartphone with minimum android 5.1 or later version and good camera quality.
- Stable Internet Connection.

Software Requirement:

- IDE: Android Studio
- Front End: XML
- Back End: Java
- Database: Firebase Realtime Database.

Chapter 5

System Design:

- A. Database Design**
- B. Input Design**
- C. Output Design**

Database Design :**1. Login :****A. Teachers Login:**

Field_Name	Data_Type	Length	Keys	Description
E_mail	Varchar	30	-	Teachers_Email
Password	Varchar	10	-	Password

B. Students Login:

Field_Name	Data_Type	Length	Keys	Description
E_mail	Varchar	30	-	Students_Email
Password	Varchar	10	-	Password

2. Registration :**A. Teachers Registration:**

Field_Name	Data_Type	Length	Keys	Description
T_id	Varchar	10	PK	Registration id
F_Name	Char	30	-	Teacher First Name
L_Name	char	30	-	Teacher Last Name
Email	Varchar	30	-	Teacher's Email
T_Password	varchar	10	-	Password
C_Password	Varchar	10	-	Confirm Password

B. Students Registration:

Field_Name	Data_Type	Length	Keys	Description
S_id	Varchar	10	PK	Registration id
F_Name	Char	30	-	Student First Name
L_Name	char	30	-	Student Last Name
Email	Varchar	30	-	Student's Email
S_Password	varchar	10	-	Password
C_Password	Varchar	10	-	Confirm Password

3. Class :

Field_Name	Data_Type	Length	Keys	Description
Class_Name	Varchar	30	-	ClassName/Subject
Class_Date	Date	-	-	Class Date
Class_Time	Date_Time	-		Class Time

4. Attendance Report:

Field_Name	Data_Type	Length	Keys	Description
A_id	Varchar	10	PK	Attendance List
T_id	Varchar	10	FK	Teachers Id
Class_Name	Varchar	30	-	ClassName/Subject
Class_Date	Date	-	-	Class Date
Class_Time	Date_Time	-	-	Class Time
Ab/Pr_Count	Bool	1	-	Absent/present Count

Input Design:**1. Login And Registration Page:****Registration:****Create Teacher:****Name****Teacher Unique ID****Email****Password**

Create Student:



Name

DEVRAJ

SHINDE

Email

devrajshinde5563@gmail.com

Student PRN

1234567892

Class

BCA-III

Password

.....

1234567892

Submit



Name

MANTHAN

PANDIT

Email

manthanpandit1317@gmail.com

Student PRN

1234567891

Class

BCA-III

Password

.....

1234567891

Submit

Login:

Teacher Login:

Student Login:



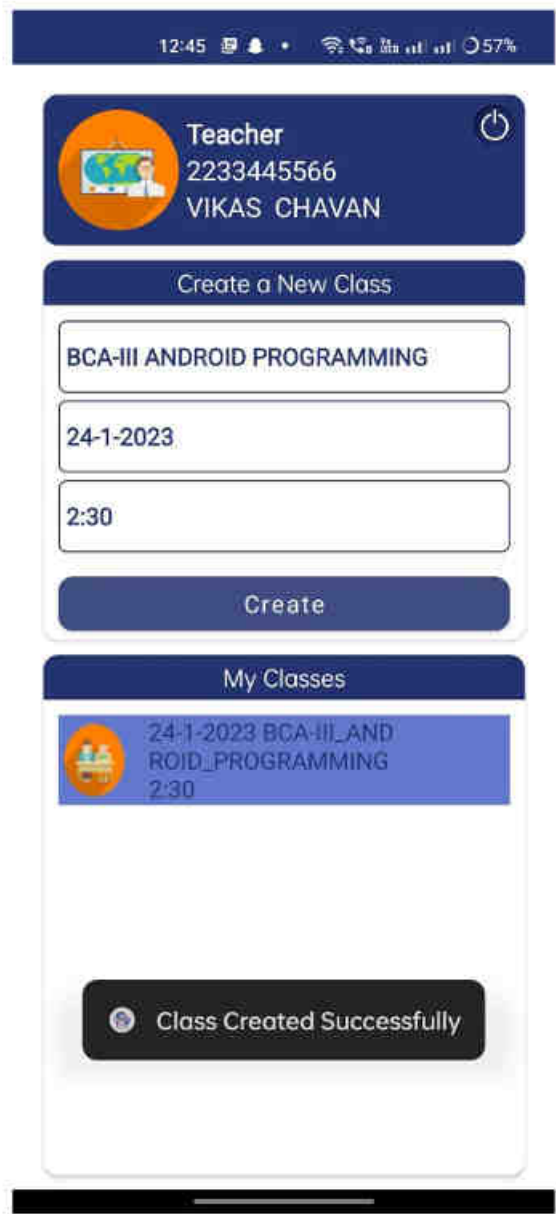
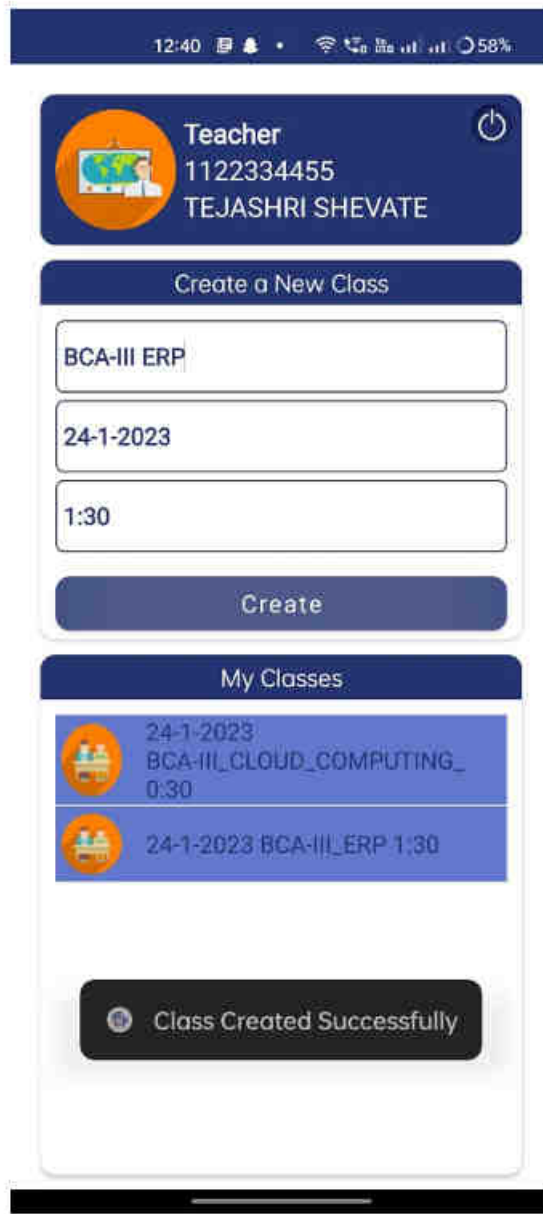
[Forgot Password](#)



[Forgot Password](#)

Attendance Process:

Creating class:



QR code & Bar code Scanning:



Student
1234567891
MANTHAN PANDIT



Important Instructions
First Scan only QR Code and then Scan Identity card
vertically

Attendance Form

BCA-III_CLOUD_COMPUTING_

24-1-2023

0:30

Scan QR Code

1234567891

Scan Identity Card

Mark Attendance

Output Design:

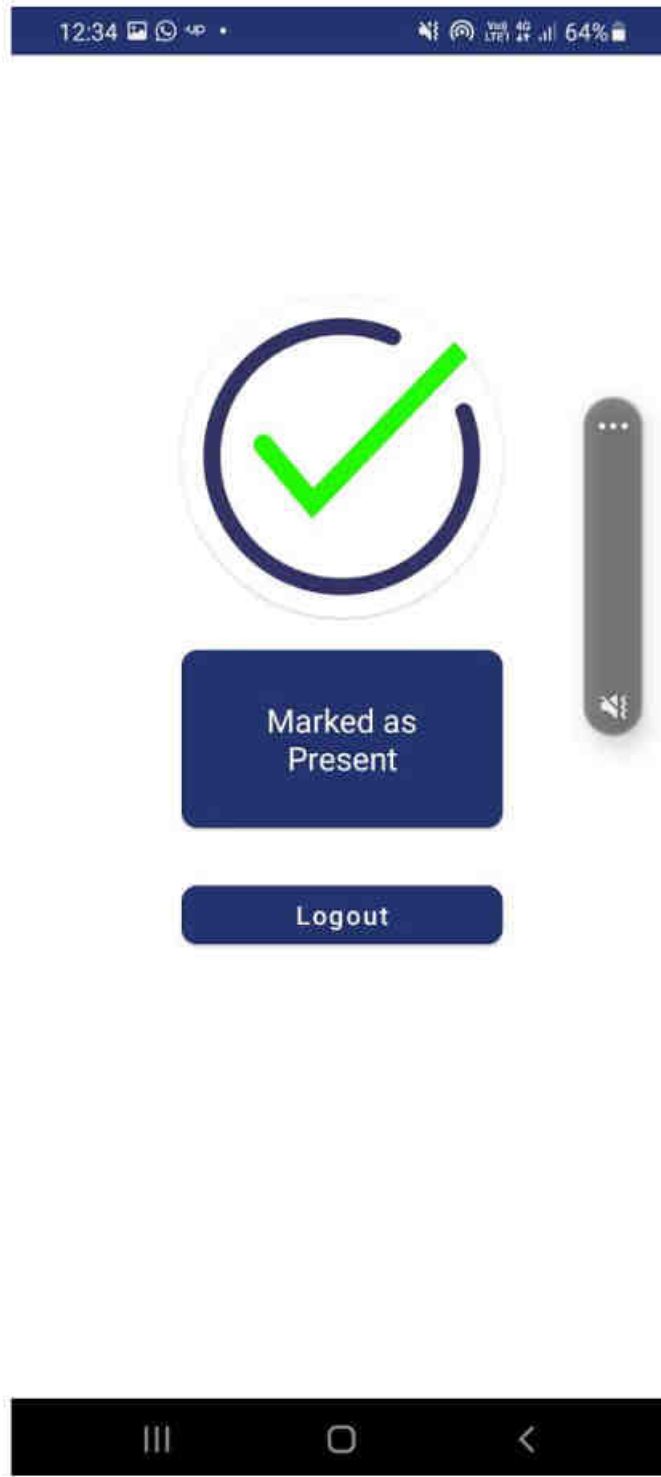
1) Account Created:



Your Profile Created
Successfully!

Login

2) Attendance Marked:



Chapter 6

User Guideline

A. Installation process

Installation process:

- 1) Go to your device's settings and scroll down to the "Security" or "Privacy" section.
- 2) Look for the option to "Allow installation from unknown sources" and toggle it on.
- 3) Download the QR code attendance app from the internet or from a trusted source.
- 4) Once the download is complete, navigate to the location where the app was saved and tap on it to begin the installation process.
- 5) Follow the on-screen instructions to complete the installation.
- 6) Once the installation is complete, you will be able to open the app and use it to track attendance using QR codes.
- 7) After using the app, it is recommended to turn off the "Allow installation from unknown sources" option in the device's settings to increase security.

Chapter 7

Source code:

Source Code:**Attendance Details:**

```
package com.smartattendance;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.graphics.Bitmap;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.ListView;
import android.widget.TextView;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;
import com.google.zxing.WriterException;
import java.util.ArrayList;
import androidmads.library.qrgenerator.QRGContents;
import androidmads.library.qrgenerator.QRGEncoder;

public class AttendanceDetails extends AppCompatActivity {
```

```
    ListView listView;
    String
classTeaName,classTeaUID,classDT,className,classDate,classTime;
    TextView txt_cl_name,
txt_cl_date,txt_cl_time,txt_clPR_cnt,txt_clAB_cnt;
    Button btn_down_sheet;
    ImageView gen_qr;
    ArrayList<String> list = new ArrayList<>();
    DatabaseReference DR1 =
FirebaseDatabase.getInstance().getReference("User").child("Teachers");
    ValueEventListener valueEventListener;
    int ABCCount=0,PRCount=0;

    @Override
    protected void onCreate(Bundle savedInstanceState) {package
com.smartattendance;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.graphics.Bitmap;
import android.os.Bundle;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.ListView;
import android.widget.TextView;
import com.google.firebase.database.DataSnapshot;
```

```
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;
import com.google.zxing.WriterException;
import java.util.ArrayList;
import androidmads.library.qrgenerator.QRGContents;
import androidmads.library.qrgenerator.QRGEncoder;

public class AttendanceDetails extends AppCompatActivity {

    ListView listView;

    String
classTeaName,classTeaUID,classDT,className,classDate,classTime;

    TextView txt_cl_name,
txt_cl_date,txt_cl_time,txt_clPR_cnt,txt_clAB_cnt;

    Button btn_down_sheet;

    ImageView gen_qr;

    ArrayList<String> list = new ArrayList<>();

    DatabaseReference DR1 =
FirebaseDatabase.getInstance().getReference("User").child("Teachers");

    ValueEventListener valueEventListener;

    int ABCCount=0,PRCount=0;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_attendance_details);
    }
}
```

```
gen_qr = findViewById(R.id.gen_qr);
listView = findViewById(R.id.listView);
txt_cl_name = findViewById(R.id.cls_nm_val);
txt_cl_date = findViewById(R.id.cls_dt_val);
txt_cl_time = findViewById(R.id.cls_tm_val);
txt_clPR_cnt = findViewById(R.id.txt_clPR_cnt);
txt_clAB_cnt = findViewById(R.id.txt_clAB_cnt);
btn_down_sheet = findViewById(R.id.btn_down_sheet);
btn_down_sheet.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent = new
Intent(AttendanceDetails.this,PDFGenerator.class);
        intent.putExtra("classTeacherName",classTeaName);
        intent.putExtra("classTeacherUID",classTeaUID);
        intent.putExtra("className",className);
        intent.putExtra("classDate",classDate);
        intent.putExtra("classTime",classTime);
        intent.putExtra("classPresentCount",PRCount);
        intent.putExtra("classAbsentCount",ABCount);
        intent.putExtra("AttendanceList",list);
        startActivity(intent);
    }
});

Bundle extras = getIntent().getExtras();
classTeaName =extras.getString("ClassTeaName");
classTeaUID = extras.getString("ClassTeaUID");
```



```
classDT = extras.getString("ClassDT");

generateQRCode();

gen_qr.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        GotoBigQR();
    }
});

String arr[] = classDT.split(" ");
classDate = arr[0];
className = arr[1];
classTime = arr[2];
txt_cl_date.setText(classDate);
txt_cl_name.setText(className);
txt_cl_time.setText(classTime);

ArrayAdapter adapter = new
ArrayAdapter<String>(this,R.layout.list_item,list);
listView.setAdapter(adapter);

valueEventListener= new ValueEventListener() {
    @Override
    public void onDataChange(@NonNull DataSnapshot snapshot) {
        int AbsentCount=0,PresentCount=0;
        list.clear();
```

```

        for(DataSnapshot snapshot1 : snapshot.getChildren()){
            list.add(snapshot1.getKey()+" "+snapshot1.getValue());
        }

        if(snapshot1.getValue().equals("ABSENT")){AbsentCount++;}

        if(snapshot1.getValue().equals("PRESENT")){PresentCount++;}

        PRCount = PresentCount;
        ABCount = AbsentCount;
        txt_clAB_cnt.setText(String.valueOf(AbsentCount));
        txt_clPR_cnt.setText(String.valueOf(PresentCount));
    }
    adapter.notifyDataSetChanged();
}

@Override
public void onCancelled(@NonNull DatabaseError error) {
}
};

DR1.child(classTeaUID).child("My Classes").child(classDate+"
"+className+"
"+classTime).addValueEventListener(valueEventListener);
}

//QR code Generation
void generateQRCode () {
    QRGEncoder qrgEncoder = new QRGEncoder(classTeaUID+"
"+classDT, null, QRGContents.Type.TEXT, 1000);
    try {
        Bitmap QRBitmap = qrgEncoder.encodeAsBitmap();
    }
}

```

```
        gen_qr.setImageBitmap(QRBitmap);
    } catch (WriterException e) {
        e.printStackTrace();
    }
}

void GotoBigQR(){
    Intent intent = new Intent(AttendanceDetails.this, QRCode.class);
    intent.putExtra("classDT",classDT);
    intent.putExtra("classTeaUID",classTeaUID);
    startActivity(intent);
}

super.onCreate(savedInstanceState);
setContentView(R.layout.activity_attendance_details);
gen_qr = findViewById(R.id.gen_qr);
```

Dashboard Student :

```
package com.smartattendance;

import android.content.Intent;
```

```
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import androidx.annotation.NonNull; package com.smartattendance;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;

import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
```

```
import com.google.firebase.database.ValueEventListener;
import com.google.zxing.integration.android.IntentIntegrator;
import com.google.zxing.integration.android.IntentResult;

public class DashboardStudent extends AppCompatActivity implements
View.OnClickListener{

    EditText ETclassNameDT,ETclassDateDT,ETclassTimeDT,
ETstdPRN;

    String stdPRN,stdFName,stdLName,stdCName;
    TextView txtStdPRN, txtStdName;
    Button btn_scan_qr,btn_scan_ID,btn_mrk_attendance;
    String classDetails,stPRN;
    String CITea,CIDt, CITm, CINm;
    String stdFullName;
    int flag =0;
    ImageView btn_prfl_lgout;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_dashboard_student);

        ETclassNameDT = findViewById(R.id.ETclassNameDT);
        ETclassDateDT = findViewById(R.id.ETclassDateDT);
        ETclassTimeDT = findViewById(R.id.ETclassTimeDT);
        ETstdPRN = findViewById(R.id.ETstdPRN);
        txtStdName = findViewById(R.id.prfl_std_nm);
```

```

txtStdPRN = findViewById(R.id.prfl_std_prn);
btn_scan_qr = findViewById(R.id.btn_scan_qr);
btn_scan_ID = findViewById(R.id.btn_scan_ID);
btn_mrk_attendance = findViewById(R.id.btn_mrk_attendance);
btn_prfl_lgout = findViewById(R.id.btn_prfl_lgout);

//Featching User Data

DatabaseReference DR =
FirebaseDatabase.getInstance().getReference("User").child("Students").c
hild(String.valueOf(LoginStudent.studentdPRN));

DR.addListenerForSingleValueEvent(new ValueEventListener() {
    @Override
    public void onDataChange(@NonNull DataSnapshot snapshot) {
        stdPRN =
snapshot.child("Profile").child("stdPRN").getValue().toString();

        stdFName =
snapshot.child("Profile").child("firstName").getValue().toString();

        stdLName =
snapshot.child("Profile").child("lastName").getValue().toString();

        stdCName =
snapshot.child("Profile").child("stdCourse").getValue().toString();

        stdFullName = stdFName+" "+stdLName;
        txtStdName.setText(stdFName+" "+stdLName);
        txtStdPRN.setText(stdPRN);
    }
    @Override
    public void onCancelled(@NonNull DatabaseError error) {
    }
});

```

```
//Logout
btn_prfl_lgout.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent2 = new Intent(DashboardStudent.this,
MainActivity.class);
        startActivity(intent2);
        finish();
    }
});

//QR Scanning
btn_scan_qr.setOnClickListener(this);
btn_scan_ID.setOnClickListener(this);

//Mark Attendance
btn_mrk_attendance.setOnClickListener(new
View.OnClickListener() {
    @Override
    public void onClick(View v) {
        markAttendance();
    }
});
}

//Initiating Scanner
@Override
```

```
public void onClick(View v) {  
    IntentIntegrator intentIntegrator = new IntentIntegrator(this);  
    intentIntegrator.setPrompt("Scan a barcode or QR Code");  
    intentIntegrator.setOrientationLocked(false);  
    intentIntegrator.initiateScan();  
}
```

Dashboard Teacher:

```
package com.smartattendance;  
  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
import android.app.DatePickerDialog;  
import android.app.TimePickerDialog;  
import android.content.Intent;  
import android.os.Bundle;  
import android.text.InputType;  
import android.view.View;  
import android.widget.AdapterView;  
import android.widget.AdapterView.OnItemClickListener;  
import android.widget.ArrayAdapter;  
import android.widget.Button;  
import android.widget.DatePicker;  
import android.widget.EditText;  
import android.widget.ImageView;  
import android.widget.ListView;
```



```
import android.widget.TextView;
import android.widget.TimePicker;
import android.widget.Toast;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;
import java.util.ArrayList;
import java.util.Calendar;

public class DashboardTeacher extends AppCompatActivity {

    DatePickerDialog picker;
    TimePickerDialog tmPicker;
    TextView txtTeacherUID,txtTeacherName;
    String teaUID, teaFName,teaLName;
    EditText ETclassname,ETclasdate, ETclasstime;
    Button btn_create;
    ImageView btn_prfl_logout;
    static String className, classDate, classTime;
    ListView classListView;

    DatabaseReference DR =
    FirebaseDatabase.getInstance().getReference("User").child("Teachers").c
    hild(String.valueOf(LoginTeacher.teacherUID));

    ValueEventListener
    valueEventListener,classListListner,newClassListner;
```

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_dashboard_teacher);

    ETclassname = findViewById(R.id.ETclassName);
    ETclasdate = findViewById(R.id.ETclassDate);
    ETclasstime = findViewById(R.id.ETclassTime);
    btn_create = findViewById(R.id.btn_QR);
    txtTeacherUID = findViewById(R.id.prfl_tea_prn);
    txtTeacherName = findViewById(R.id.prfl_tea_nm);
    classListView = findViewById(R.id.class_list);
    btn_prfl_logout = findViewById(R.id.btn_prfl_logout);

    //Fetching Profile Details
    valueEventListener= new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            teaUID = snapshot.child("teacherUID").getValue().toString();
            teaFName = snapshot.child("firstName").getValue().toString();
            teaLName = snapshot.child("lastName").getValue().toString();
            txtTeacherName.setText(teaFName+" "+teaLName);
            txtTeacherUID.setText(teaUID);
        }

        @Override
```

```
        public void onCancelled(@NonNull DatabaseError error) {
        }
    };

    DR.child("Profile").addListenerForSingleValueEvent(valueEventListener
    );

    DR.child("Profile").removeEventListener(valueEventListener);

    //Logout
    btn_prfl_logout.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Intent intent2 = new Intent(DashboardTeacher.this,
    MainActivity.class);
            startActivity(intent2);
            finish();
        }
    });

    //Date Picker
    ETclasdate.setInputType(InputType.TYPE_NULL);
    ETclasdate.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            final Calendar calender = Calendar.getInstance();
            int day = calender.get(Calendar.DAY_OF_MONTH);
            int month = calender.get(Calendar.MONTH);
            int year = calender.get(Calendar.YEAR);
```

```
        picker = new DatePickerDialog(DashboardTeacher.this, new
DatePickerDialog.OnDateSetListener() {
            @Override
            public void onDateSet(DatePicker view, int year, int month,
int dayOfMonth) {
                ETclasdate.setText(dayOfMonth+"-"+(month+1)+"-
"+year);
            }
        },year,month,day);
        picker.show();
    }
});

//Time Picker
ETclasstime.setInputType(InputType.TYPE_NULL);
ETclasstime.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        final Calendar calender = Calendar.getInstance();
        int hr = calender.get(Calendar.HOUR_OF_DAY);
        int minutes = calender.get(Calendar.MINUTE);

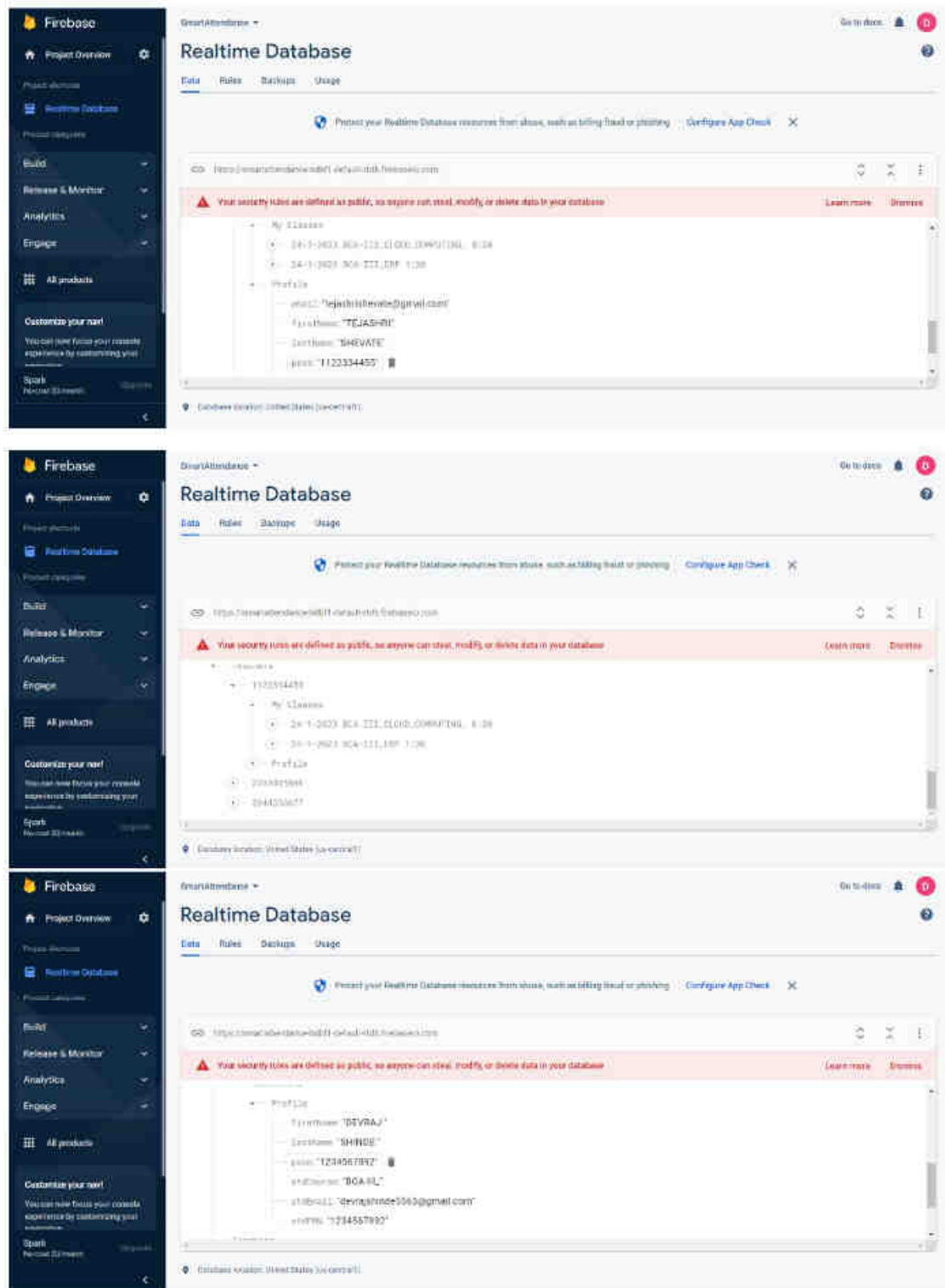
        tmPicker = new TimePickerDialog(DashboardTeacher.this,
new TimePickerDialog.OnTimeSetListener() {
            @Override
            public void onTimeSet(TimePicker view, int hourOfDay, int
minute) {
                ETclasstime.setText((hourOfDay-12)+":"+minute);
```

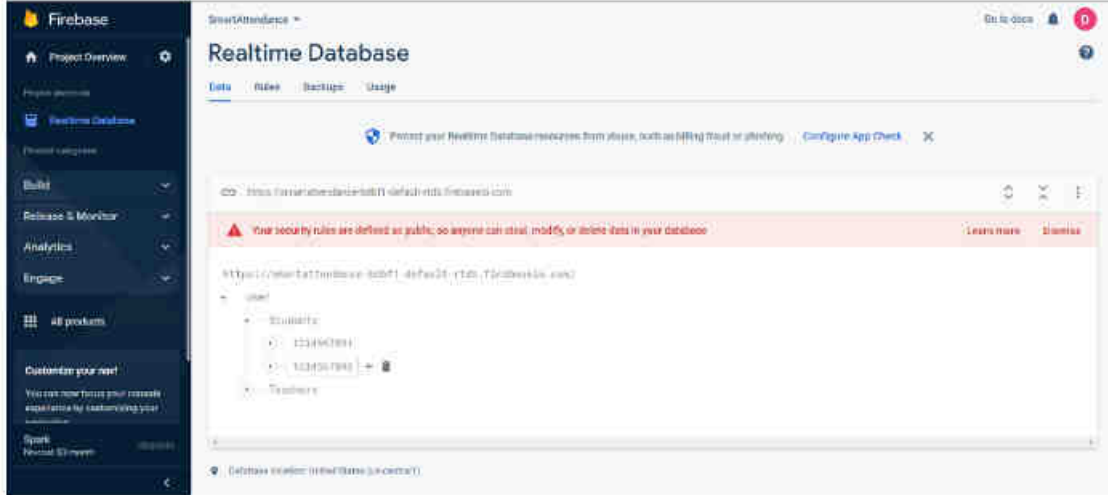
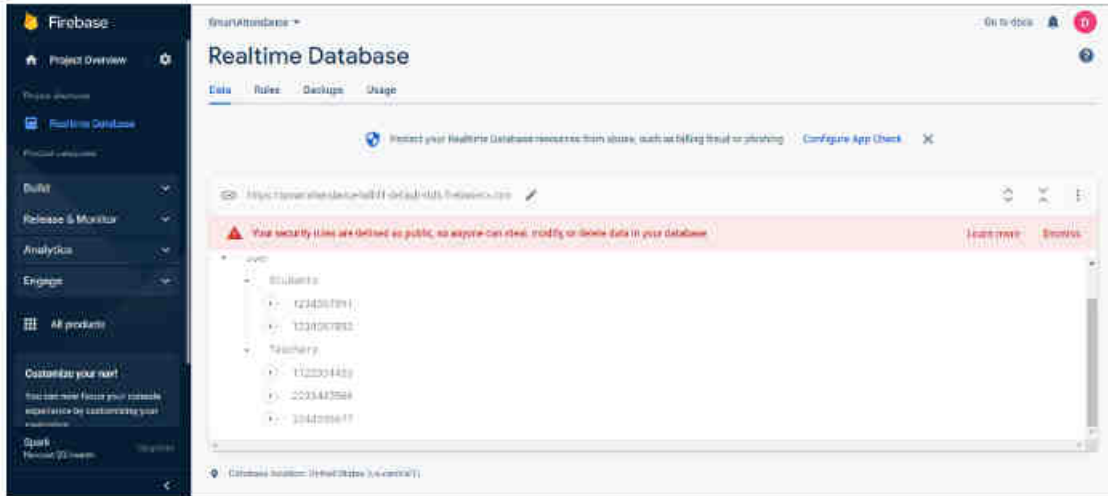
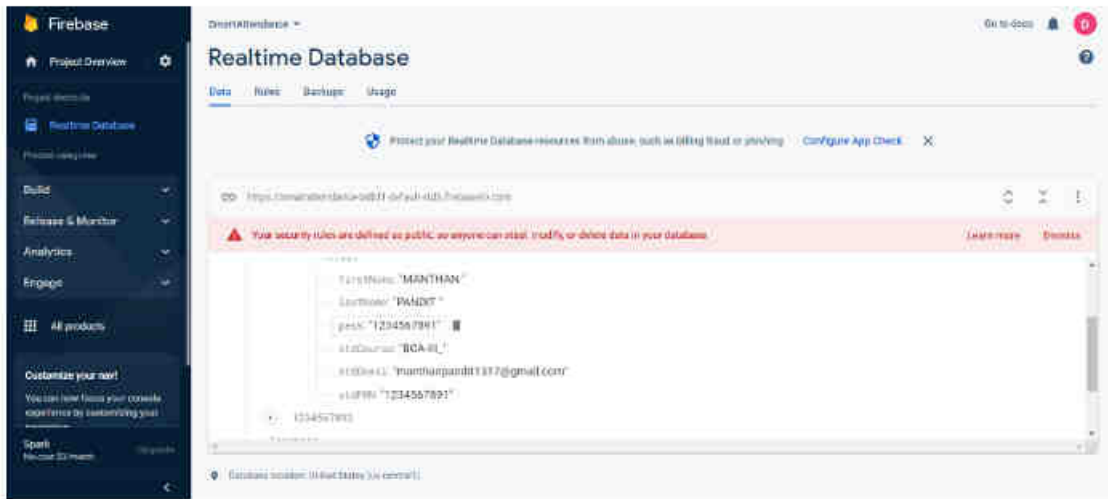
```
        }  
    },hr,minutes,false);  
    tmPicker.show();  
    }  
});
```

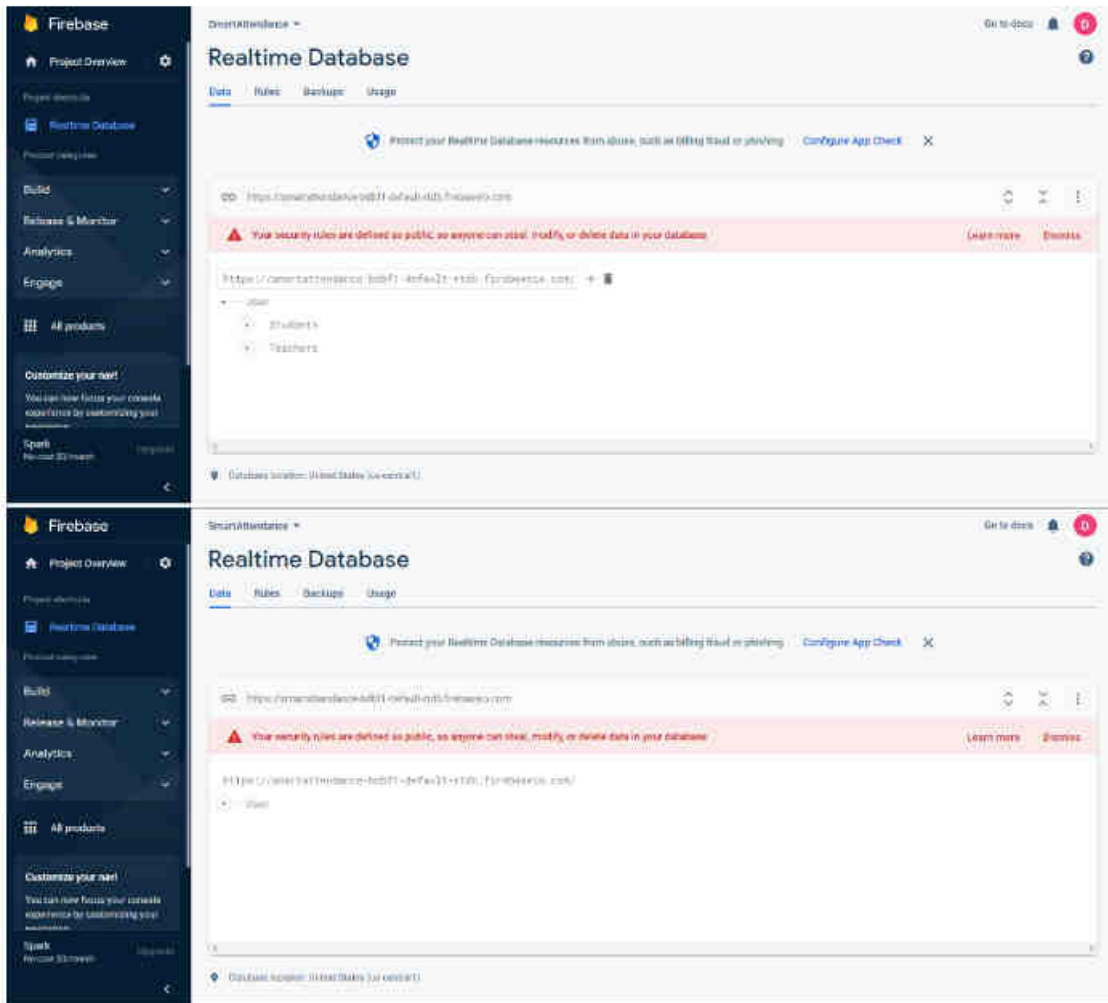
Chapter 8

Reports

Reports:







Chapter 9

Conclusion and Suggestions:

- A. Conclusion and Suggestions**
- B. Future Enhancement**
- C. Bibliography**

Conclusion and Suggestion:

Conclusion:

Android-based QR code attendance app offer a convenient, secure and cost-effective way for institutes to track attendance. The system requires a device with an android operating system, a camera, internet connection and optionally a printer. The software requirements include QR code scanning app, attendance management app, a database and integration with other systems. The system also provides various reports and analytics, real-time tracking, reduced human error and contactless attendance tracking. In addition, institutes can also use additional hardware like barcode scanner, NFC and MDM software for more advanced feature and security.

Suggestion:

- Use a responsive design to ensure the app is compatible with different screen sizes and resolutions on Android devices.
- Consider implementing push notifications to remind students and teachers to record their attendance.
- Use a well-designed user interface that is easy to navigate and understand.
- Ensure the app is optimized for performance and can handle a high volume of attendance records.
- Test the app on different Android devices and platforms to ensure compatibility.
- Consider implementing a feedback system for users to report any issues or suggest improvements.

Future Enhancement:

The Proposed system is Android Based Attendance System. We can enhance this system by including more facilities related to administration and student attendance history. We can provide respective environment for student where he not only can mark his attendance but also authorities can control proxy attendance.

- Automated attendance marking: Automating the process of marking attendance could save time and reduce the workload for teachers.
- Biometric authentication: Using biometric authentication methods such as fingerprint or iris scanning could provide an even more secure and convenient method for students to mark attendance.
- Location-based tracking: Integrating location-based tracking could allow institutes to verify that students are physically present at the event or class location.
- Chatbot integration: A chatbot could be integrated with the app to provide students with quick and easy access to information about attendance and other related matters.
- Parental access: Giving parents the ability to track their child's attendance and performance, and receive notifications, could increase transparency and communication between institutes and parents.
- Platform Independent: App will be available on both Android OS and IOS also.

Bibliography:

References

Google Developers - Firebase Tutorials

Google Developers – Android

Books:

"Android Studio Development Essentials" by Neil Smyth

"Java Programming for Android Developers" by Kyle Mew

"Java Programming"-McGrawHills

Websites:

<https://developer.android.com/>

<https://firebase.google.com/docs/guides>

Android QR Code Scanner App Tutorial Using ZXING Library" by Belal Khan, available at: <https://www.simplifiedcoding.net/android-qr-code-scanner-tutorial/>

"ZXing Android Embedded" by ZXing Team, available at: <https://github.com/zxing/zxing>

