Smart Sensor Network Based Atm Management System using Lab view

Nagarjuna Telagam, Sunita Panda, Nehru Kandasamy, Menakadevi Nanjundan

Abstract*: This paper examines policy regarding the biometric approaches towards automated teller machine (ATM) for trustworthy and secured transactions. Fingerprint devices for ATM's are mentioned and enforced during this project. Previously the user should carry his Debit card or Credit card whenever he goes to the ATM for his transactions. This is often a time consuming method because the person needs to insert his card in the card reader slot in a correct way and make sure he entered his pin number correctly, sometimes it may block our card if we entered wrong pin number for many times. In our proposed method we are replacing the card reader with Biometric Fingerprint sensor. The project is mainly divided into two cases i.e Authentication through Fingerprint, manual login. Here in this project multi spectral imaging (MSI) sensor is used to reduce the vulnerability of fingerprint sensors to spoof attacks. This sensor has been able to distinguish between a live finger and other soft materials. This project provides a break through against current technology in ATM and able to provide strong protection for upcoming future ATM's.

***Index Terms*: ATM, MSI Sensor, labVIEW, Arduino.**

I. INTRODUCTION

This project explains fingerprint biometric scheme which is fused with the ATM for person identification to increase the security level [1].The ATM management technique is updates using four pin code for increasing the security [2]. At the time of transaction fingerprint image is acquired at the ATM terminal using high resolution fingerprint scanner [3].Here, if the fingerprint is recognized, then it displays the multiple banking screens. The user can choose the bank which we need for transaction. The remaining feature are same [4]. This system model depends on relations among features and deals with automated recognition system which overcomes the present fingerprint models [5]. The Debit card or credit card has inbuilt sensor is used to sense the biometric signature, this signature is being sent to remote reader for biometric signature comparison to validate for original user detection [6]. With NI-VISA driver available, the users can interface to STAR-Dundee Space Wire PCI and cPCI, boards for more security to the system [7]. The projects based on VISA driver receives data only through query buffer of serial ports, so it's impropriate a lot of time of CPU is consumed and not have good real time capability [8]. The Data Acquisition modeling programming is taken as reference from [9]. Serial communication is used at the transmitter section to send data, one bit at a time, through channel to the receiver using labVIEW this programming model is explained in [10]. During authentication, the users fingerprint is scanned to be stored in one of the two buffers and the extracted features compared with the template which is loaded to the other buffer to compare the match before account is verified. [11]. This developed system provides security by providing PIN for authentication [12]. Biometrics are used to provide better secured access to major functioning systems like ATM, cellular phones, cars, laptops, offices, and other things that need authorized access [13]. The users cell phone is used in this project for registration and installed with the OTP generation software [14]. The generated OTP is valid for only a short period of time and is generated by factors that are unique to both, the user and the mobile device itself [15]. The same security token is been used for major functioning systems such as laptops and ATM's to avoid this situation this paper proposes an authentication solution to avoid usage of extra device by re-using existing devices, namely the mobile phone or the SIM cards [16]. VI package manager and the LINX interface for Arduino is used to control boards with using labVIEW software. [17]. This paper explains the sensor based cruise control irrigation motor and sensor working operation is taken from [18]. The usage of labview software and its VI programs are explained in this paper and it has taken as reference [19]. The smart sensor networks are designed using labview software are taken as references [20] and [21]. The virtual and remote laboratory was established and its key features and experiments are designed using USRP devices such as digital audio broadcasting and its bit error rates and its VI programming are taken as references from [22] and concatenated levels of encoding the data is explained in [23] and USRP 2901 based VI programming models are taken as references for this paper [24], [25].

II. IMPLEMENTATION

The figure 2.1 describes the flow of operations of our project. It begins when we press the RUN button in the LabVIEW and ends whenever STOP is encountered.

Revised Manuscript Received on June 28, 2019.

Nagarjuna Telagam, Assistant Professor, ECE Department, GITAM University, Bangalore, India.

Dr. Sunita Panda, Assistant Professor, Department of ECE, GITAM University, Bangalore , India.

Dr. Nehru Kandasamy, ECE Department, Institute of Aeronautical Engineering , Hyderabad, India.

Menakadevin Nanjundan, Assistant Professor, ECE Department, Hindustan College of Engineering and Technology, Coimbatore, India.

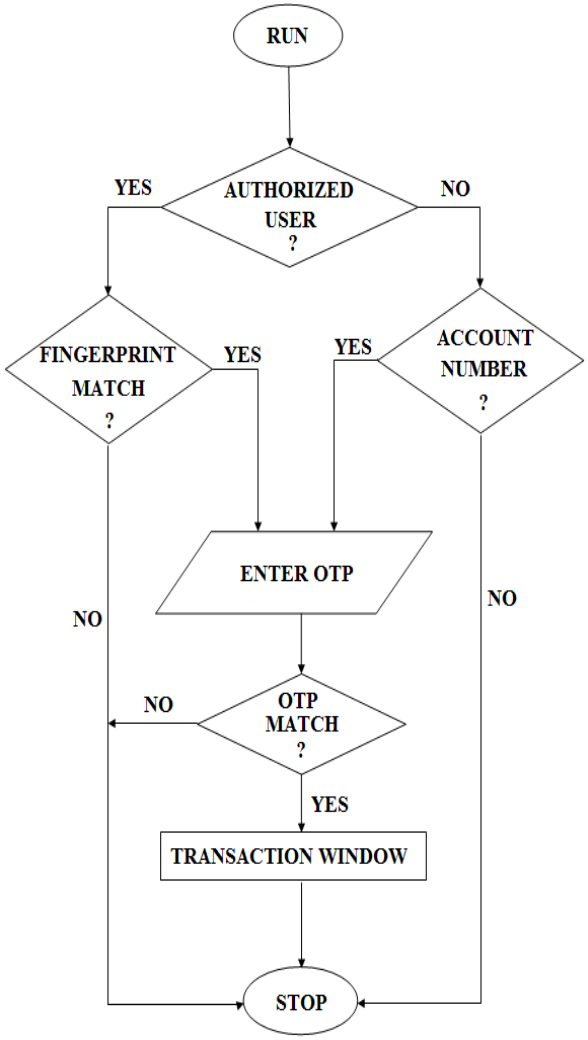

Figure 2.1: Flow chart

In our Project, We are designing two cases for login at ATM. When we start the process it will ask whether user is an authorized user or not. If the user is an authorized one, it will ask the fingerprint of the user. If it is not matched with the database of the users i.e. NO, it will terminate the process. If YES i.e. if the fingerprint is matched with the template in the database, then it will generates a random One Time Password (OTP) and is sent to the user's mobile number obtained from the database. Then a popup will appear on the screen asking user to enter OTP that he/she received. If entered OTP is matched i.e. YES, then user can be allowed to access their account i.e. transaction window will be opened and hence user can access further options such as check balance, deposit some amount or withdraw some amount. If entered OTP doesn't match i.e. NO, then it will STOP the process. In the second case, Account can be accessed through details of the user manually. User has to enter his/her account number. If it is matched with the bank's database i.e. YES, then it will generate random One Time Password (OTP) and is sent to the user mobile number. Within moments a popup will ask the user to enter received One Time Password and verifies it. If it is matched i.e. YES then user can access the transaction window. If the entered OTP is wrong then the process will be terminated i.e. it STOPs. In this project, we proposed a way to login using fingerprint sensor and manually entering account details. We didn't used or coded enrolment logic. The enroll process will be carried out at the bank itself. Before starting the process, we need to make sure that sensor is dust free and all the hardware components are working. To make the front panel look effective, we can choose windows appearance options by clicking the command 'Control+I.' A popup appears and displays specific options which contains window maximization, choose runtime options.

Further we can customize the available options. In LabVIEW, we can password protect the block diagram and it will display only control panel. Only when we type the password, block diagram will be appeared and we can edit that only after providing password. Whenever we place a finger on the sensor's face, the valid fingerprint template is stored into the memory of the sensor at a specific address location as shown in fig 2.2.

In our project we divided the ATM into two cases

1. Fingerprint Login
2. Manual Login

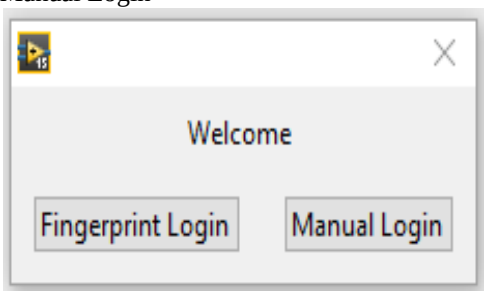

Figure 2.2: Two ways of Login

2.1 Fingerprint Enrollment

In order to verify whether a fingerprint is valid or not, first it has to be enrolled into the sensors memory. This can be done in two ways.

1. Programming in Arduino for enrolment
2. Using SYDemo software

2.2 Fingerprint Verification

To check whether a fingerprint is matched or not with the templates in database of sensor, run the fingerprint file from library as shown.Go to 'files>examples>adafruit_sensor library>fingerprint' Set the baud rate as per the sensor and set the pins properly as per the Hardware. Compile it for any errors, if no errors are found, upload it to UNO board. Open serial monitor and it displays- 'Found fingerprint sensor !','Waiting for valid finger...' Now place valid finger on sensor and remove it and it displays the hash value or address location where your fingerprint is stored. It displays- 'Found ID #x with confidence of xyz.' Here confidence implies the percentage of matching. It displays as 'Found ID #2 with confidence of 136.'

2.3 Fingerprint Id Extraction

Fingerprint verification is done using the Arduino UNO code using the library files. The output of this execution will be a Hash value in the form Fingerprint ID #2 matched with confidence of 132. Here confidence indicates the

percentage of matching. #2 is the address location where the fingerprint template is stored. Using LINX drivers we can interface Arduino code into LabVIEW via Makerhub. It can be opted via 'Tools> Makerhub > LINX > Firmware Wizard.' We need to choose the version of the Arduino we are using, and the port to which board is connected and built the firmware i.e. upload code into LabVIEW as shown in fig 2.3. After uploading the code to LabVIEW, we use serial VISA functions to extract outputs from Arduino code.

Figure 2.3: LINX Build Firmware.

2.4 Code for Fingerprint Extraction

In the first step we used VISA CONFIG to initiate the serial communication. We need to set the exact Baud rate (Symbols/Sec) and COM port. Later we need to read the output from Arduino code and better specify the number of bytes to read. In the project we used Decimal string to Number convertor which converts the string data into number from the specified index value. Using this, we will obtain the account number of the user as described in the figure 2.4.

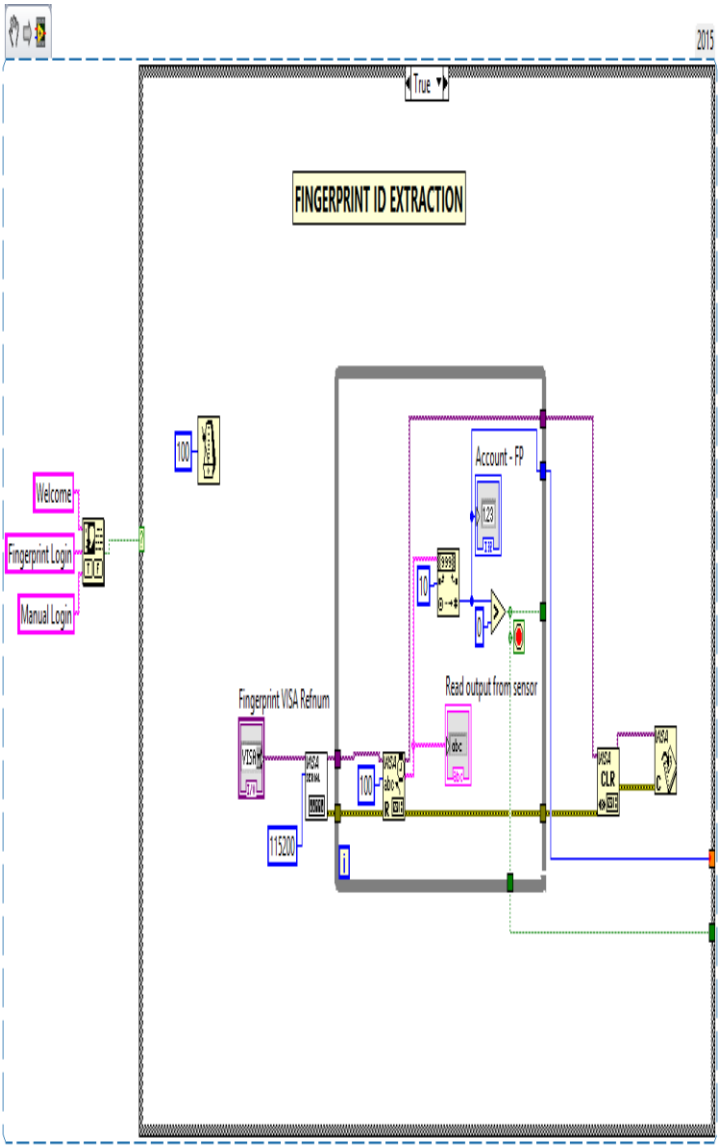

Figure 2.4: Fingerprint ID Extraction

The reading function is kept in while loop since the project involves continuous reading of output from the Arduino board. Once the exact data is read, we need to clear the buffer in which data's been read. And after clearing, serial communication should be closed for next installment of connection. Later from this Account number we need to extract the mobile number as stored in database of the user's information. Once the Account number is extracted, we need to get the mobile number to send the generated OTP from OTP generation.

2.5 Code for Mobile Number Extraction

In this code we used a while loop to automate the whole process. In the first step, we used READ FROM SPREADSHEET function which reads the data from the text file which stored the user's data. The data from spreadsheet is in the form of 2D array, next it is given to Index array with column index as 0. We need to search the 0^{th} column i.e. Account details. If the account is found, we need to display corresponding 1^{st} column value which is mobile umber of the user. Later it is converted to string data using

number to decimal string function as discussed in figure 2.5.

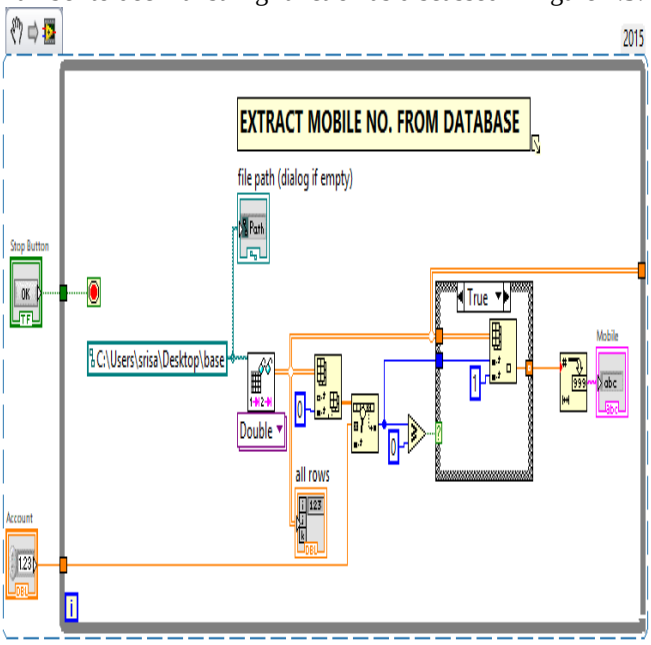

Figure 2.5: Code for Mobile number Extraction

After getting mobile number of the user, we need to do 3 things i.e.

1. Generate an OTP
2. Send OTP to the registered mobile number
3. Open Transaction window

2.6 Otp Generation

After getting account number and mobile number from users list in database, we need to generate an OTP so that it can be to user's mobile. To generate an OTP we need a Random no. generator which can generate a number between 0 to1. We use a multiplier (X) which multiplies the generated number with Nine (9) and the resultant no. is rounded off to the nearest integer with the help of 'Round off to the nearest [ ]'function. Now this value is given to a number to string converter as discussed in figure 2.6.

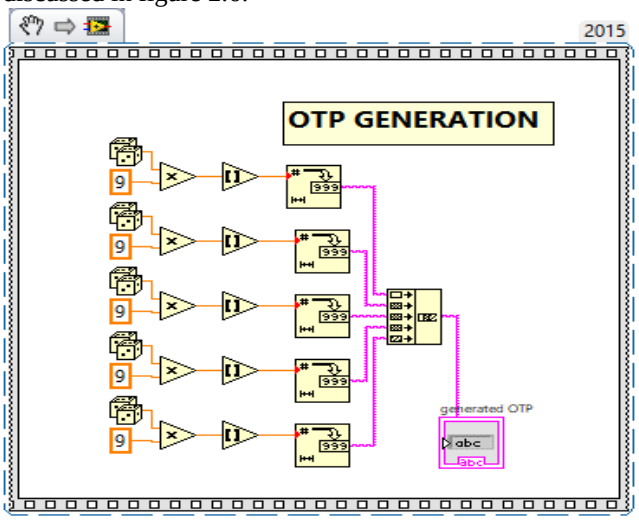

Figure 2.6: Generation of OTP

Using the above logic we generate 5- digit OTP and the resultant number is given to CONCATENATION STRING which convert no. of strings into a sequence of numbers and a '5 DIGIT OTP' is generated. Now generated OTP is sent to the registered mobile number via GSM Module.

2.7 GSM Code to Send Sms

GSM module basically used to send a message to specific mobile number. We use SIM900A module in our project to send generated One Time Password (OTP) to a mobile number specified in the project.We use some kind of commands in order to send messages called AT COMMANDS. These are the set of commands used to send SMS, receive SMS and even call to specific number. These commands are basically executed for the functioning of the GSM module. These can be executed in PUTTY or HYPERTERMINAL software as shown in figure 2.7.

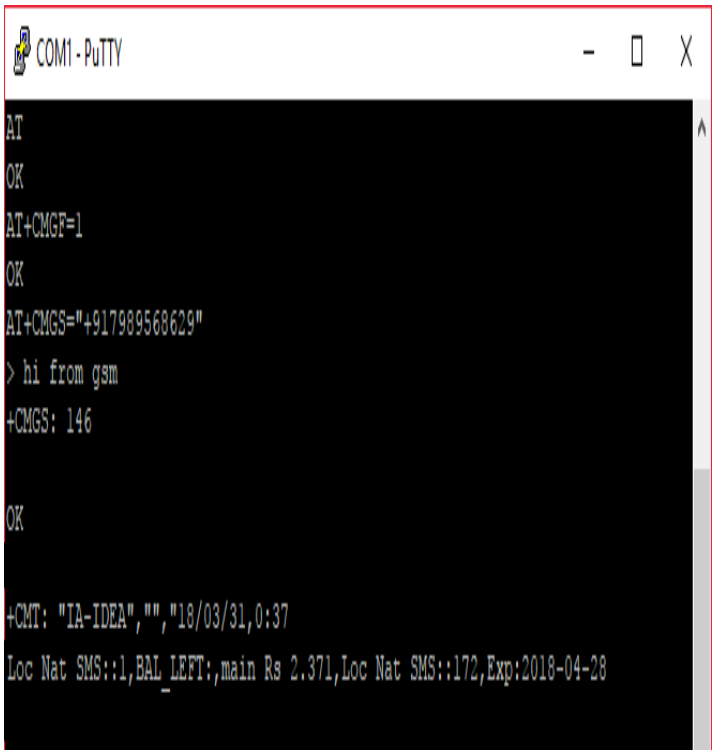

Figure 2.7: AT Commands executed in PUTTY software

2.7.1 Code Explanation

Here each AT command is executed via VISA function.

To configure serial connection, VISA Config. Function is used. It initializes the serial port specified by VISA resource name to the specified settings. Wire a control or constant to the pin according to the code. This is where we specify COM port to which GSM module is connected.Next function is VISA WRITE function using which each AT command is executed. AT+CMGF=1 is the string constant which is written into VISA WRITE function. After it is executed we need to press enter, so the HEX code for Enter command is either Carriage return in string functions or 0D as given in the code.

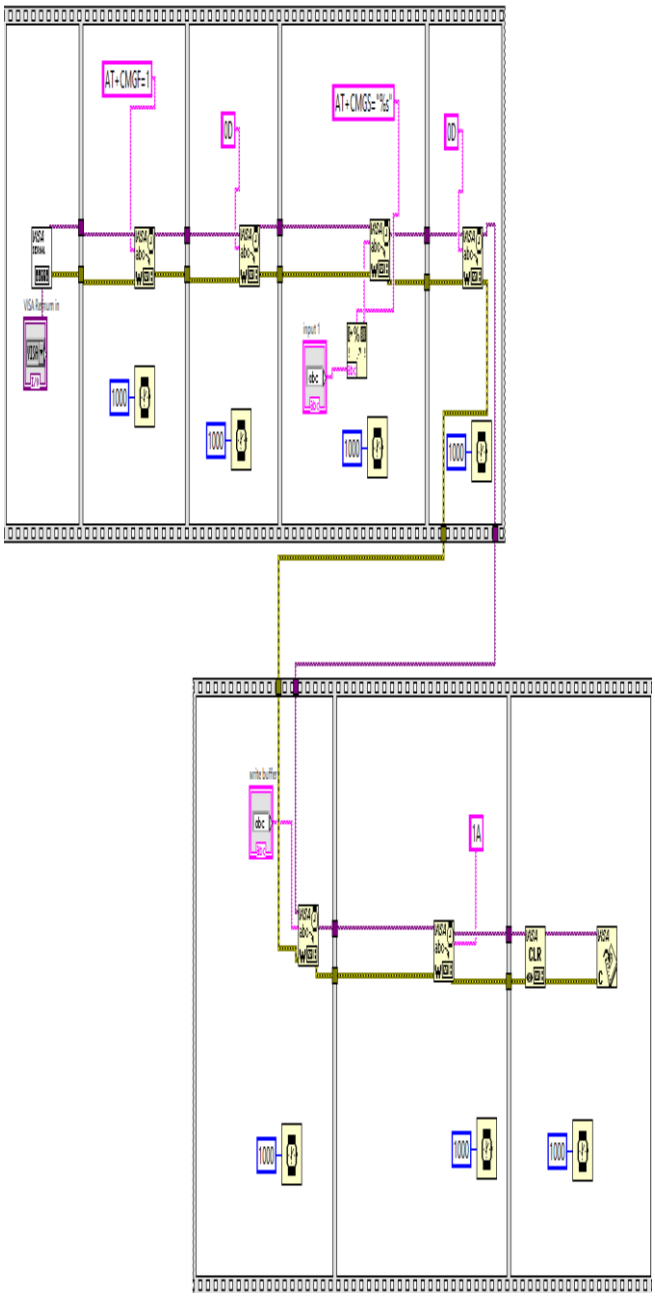

Figure 2.8: GSM code to send SMS – Block Diagram

Next command is to enter mobile number which is given by AT+CMGS=%s string constant. Here format into string function is used which inserts the mobile number in the place of %s. After this again we need to press enter i.e. 0D hex code. After executing this command we need enter our message which is to be delivered. It appears in the form >enter message here. After entering the message we need to press 'Control+Z' in order to send it to the specified number. Since the Hex code for Control+Z is '1A', we placed it as string constant and written and shown in figure 2.8.

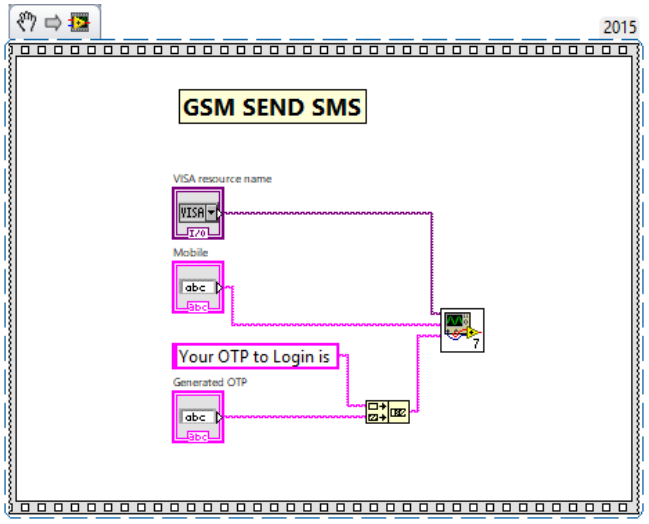

Figure 2.9: Sub VI for GSM code to send SMS

Each block is added with a delay of 1000ms i.e. 1sec which is given for proper execution of the commands. In the last block, VISA CLEAR and VISA CLOSE are used. CLEAR is to empty the buffer where all the commands are executed such that next time it is available to send SMS again. CLOSE is used to terminate the established serial communication at the port.So, in our code, the message to be sent is Your OTP to Login is xxxxx. The xxxxx pattern is generated OTP obtained from OTP generation code. And this code is sent to user's mobile number which is obtained for database as shown in figure 2.9.

2.8 Otp Verification A dialogue box is used in the Flat sequence structure which displays ENTER OTP. After entering the OTP we use a comparator which compares the generated OTP and Received OTP. If the entered OTP is matched then it opens a Transaction Window as shown in figure 2.10.

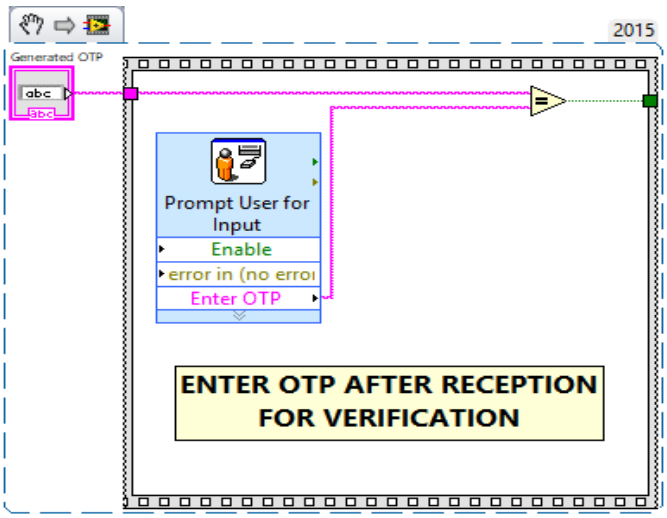

Figure 2.10: OTP verification

2.9 Transaction Window

After, OTP verification, Transaction window will be opened. In the transaction window we use a 3 input dialogue box which displays BALANCE, WITHDRAW and DEPOSIT

as explained in figure 2.11.

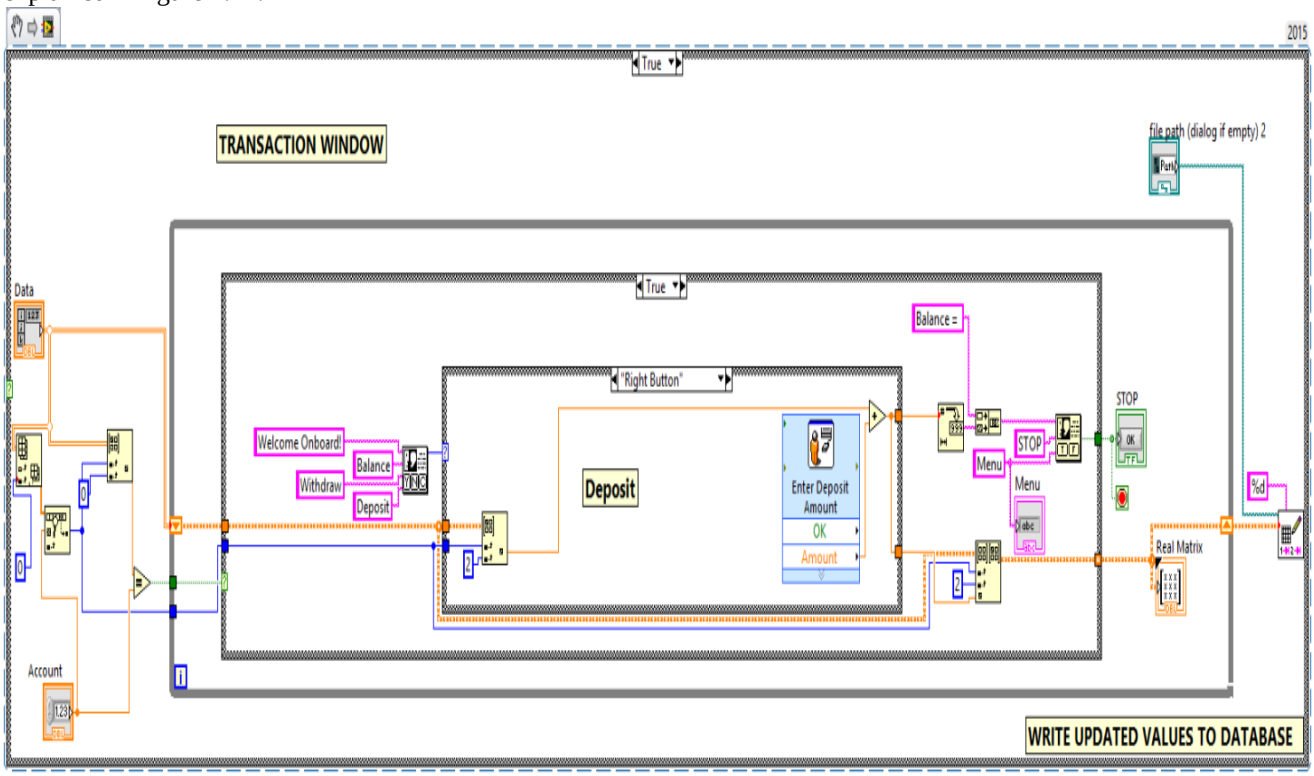

Figure 2.11: Code for Transaction window

2.9.1 Balance

In this it shows the available amount in our account after every transactions. It is updated automatically after every withdrawal or deposit. Here we obtain the updated values and the balance value is obtained using an index array and is given to number to decimal string convertor which is concatenated later with 'BALANCE=' and is given to dialog box which pops up during execution and show the updated balance.

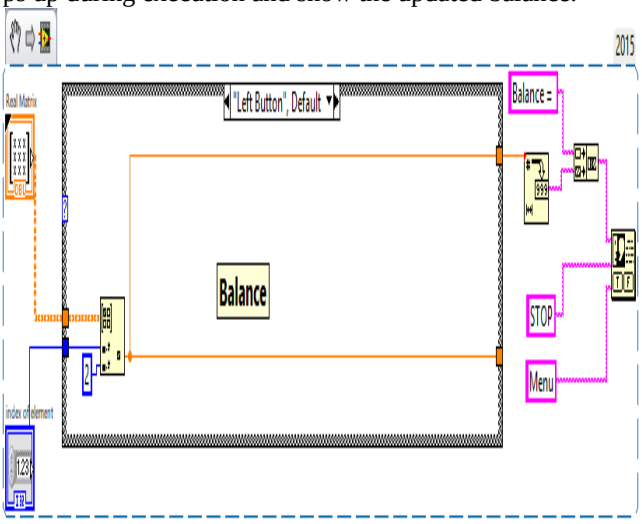

Figure 2.12: Code for Balance window.

2.9.2 Withdraw

Similarly as shown in balance part, we obtain updated balance. Later user will be asked to enter the withdrawal amount. After entering the amount i.e. equal to or less than the balance amount, it is given to subtractor (-) and updated balance will pop up.

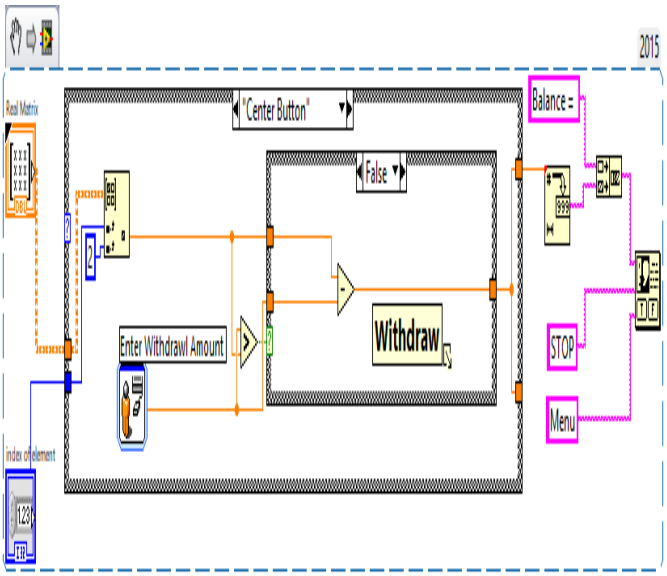

Figure 2.13: Code for withdrawal window.

If the user enters the amount more than the present balance, a pop message with 'Insufficient balance' will be displayed. After every transactions it updates the balance amount.

2.9.3 Deposit

In this we can deposit the amount which does not exceeds the daily limit. After every transaction the deposited amount is updated and it is added to the available balance amount.

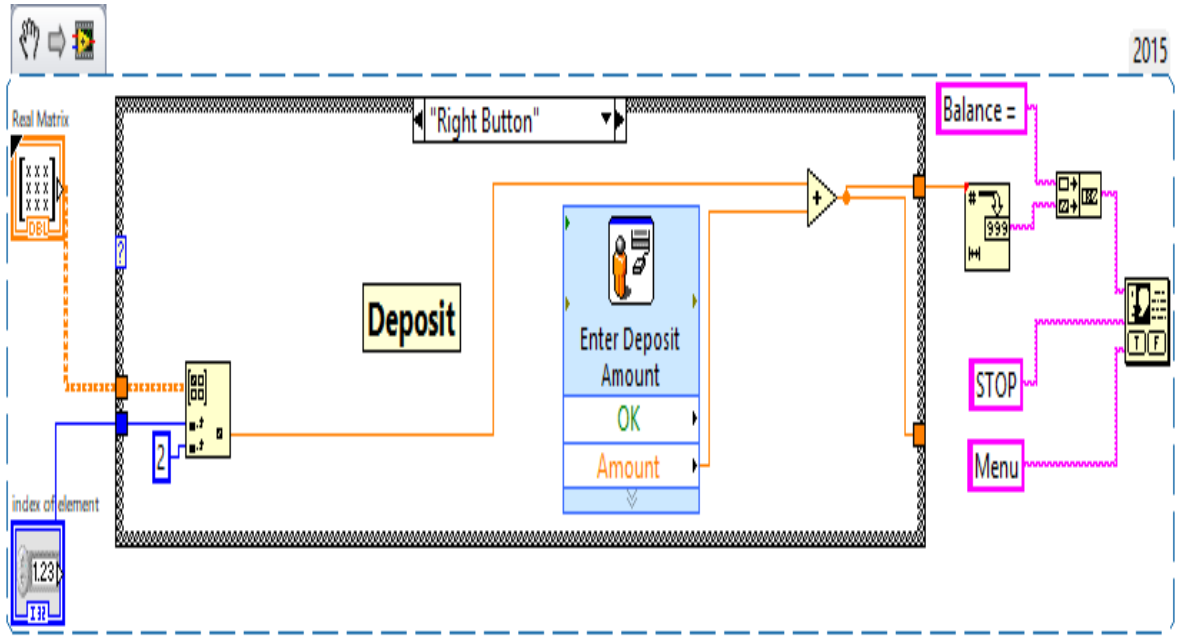

Figure 2.14: Code for Deposit window.

After every transaction we used a 2 button dialogue box which displays STOP and MENU. When we press STOP it aborts the transaction and updates transaction details to the database. If we press MENU, it returns the user to main menu displaying 3 options Balance, Withdraw, Deposit.

2.9.4 Write to Database

Once the data has been updated it should be continuously written to database file. We used 'write to spreadsheet" function to write into a file located by file path control.

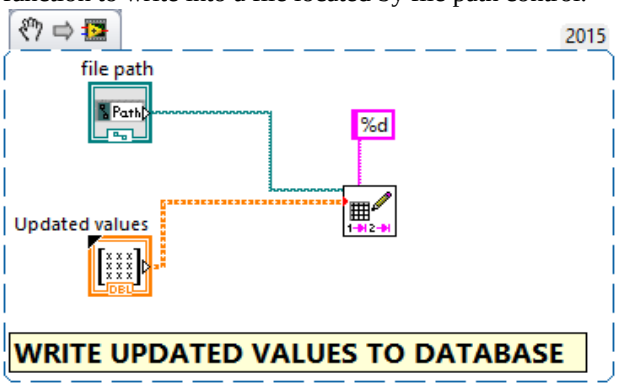

Figure 2.15: Code to update the database after every transaction.

2.10 Manual Login

The second method of login is to enter user details manually. In manual login we need to enter the account no. and if it matches with the number in database, an OTP will be generated and sent to the registered mobile number. Once OTP is received, user has to enter it in the prompt user popup and if it matches with the generated OTP, a transaction window will be displayed. If OTPs doesn't match, process will be terminated indicating 'OTP doesn't match.'Submit your manuscript electronically for review.

III. RESULTS AND DISCUSSIONS

3.1 SETUP

Once the total setup is put together and required drivers are installed in the PC, we need to press RUN. Figure 3.1 and 3.2 shows the total Practical Hardware setup.

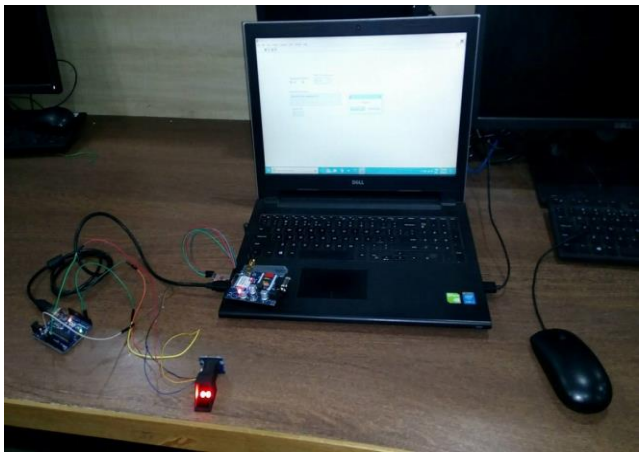

Figure 3.1: Total Setup

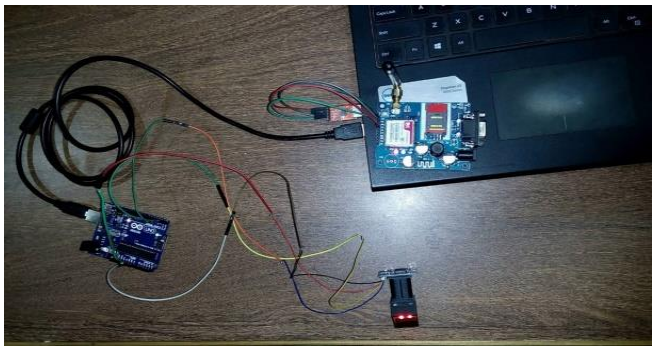

Figure 3.2: Hardware Setup

3.2 LabVIEW CODE

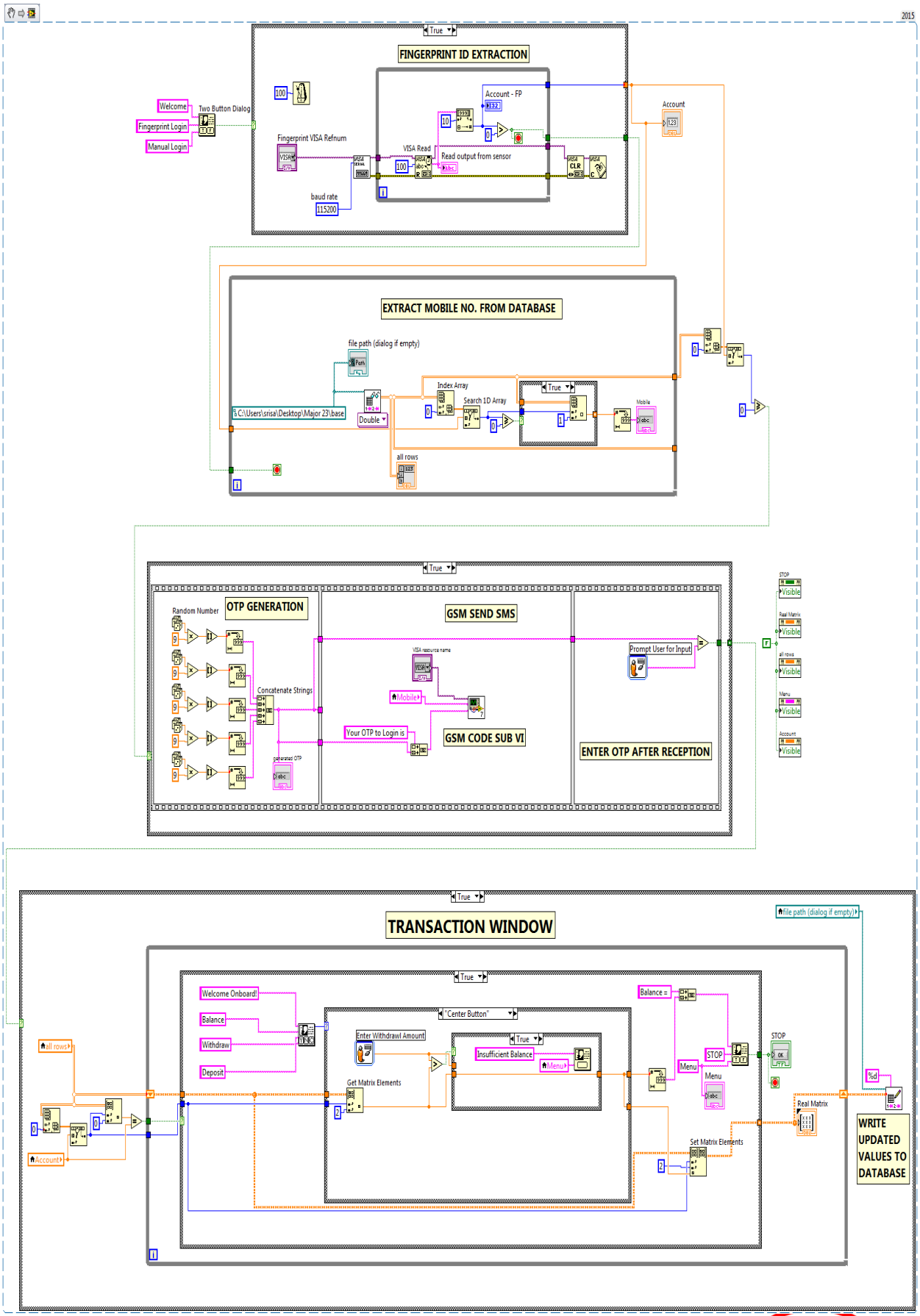

Figure 3.3: LabVIEW Total Logic

After the whole setup we need to run the program. Once we click the run button a pop up will appear on the screen indicating FINGERPRINT LOGIN and MANUAL LOGIN. We need to opt one of it, first we will go with FINGERPRINT LOGIN as shown in figure 3.3.

3.3 Fingerprint Login

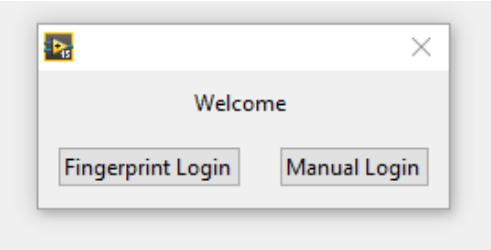

Figure 3.4: Login Pop Up

Now click on the FINGERPRINT LOGIN and put your finger on the sensor, the sensor will scan your finger in less than a second and verifies with the correct finger. If your fingerprint had already been registered in the database then it will return you a HASH VALUE (#3) or the address location of your fingerprint. If account number or fingerprint doesn't match with the ones in database, it displays 'Account not found.'

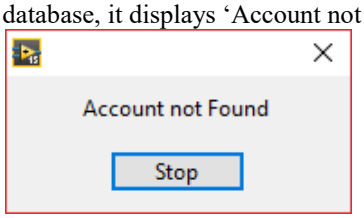

Figure 3.5: Account not found

Now an OTP is sent to the corresponding registered mobile number which is linked with the user account (3). After sending the OTP a pop will appear on the screen with in 5sec displaying ENTER OTP. Enter the received OTP in respective column.

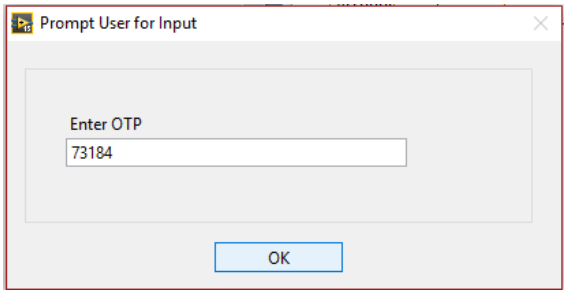

Figure 3.6: Enter OTP to access Your Account

If your OTP is matched then you will directly enter into the transaction window, if it is not matched a pop up will appear on the screen displaying OTP NOT MATCHED and aborts the transaction.

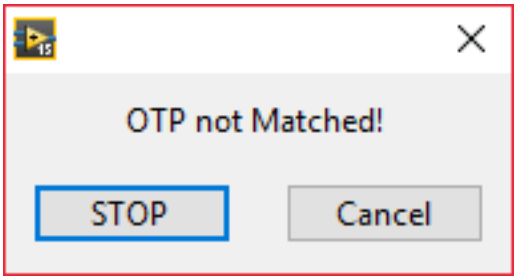

Figure 3.7: OTP not matched

After entering correct OTP, user will be entered into the transaction window a pop up will appear on the screen displaying BALANCE, WITHDRAW, DEPOSIT.

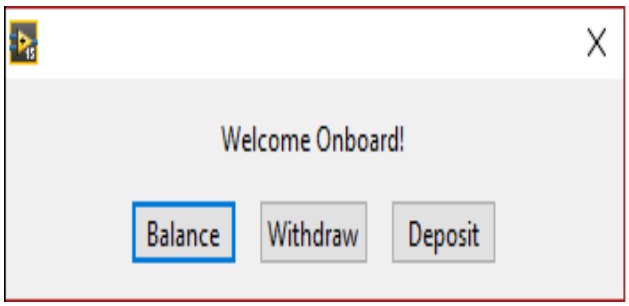

Figure 3.8: Transaction window to choose the operations

If we click on the DEPOSIT it will ask you to ENTER DEPOSIT AMOUNT after adding your amount it will show you the updated Balance.

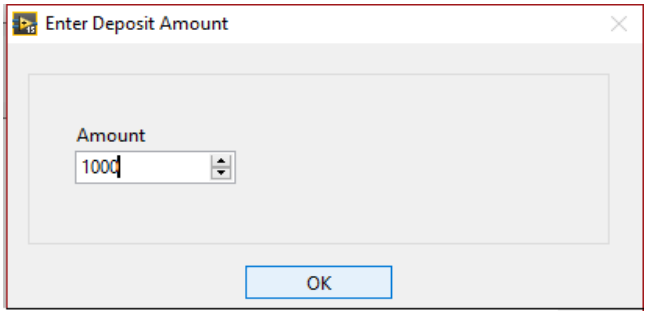

Figure 3.9: Deposit window

If we click on the WITHDRAW it will ask you to ENTER WITHDRAWAL AMOUNT after withdrawing your amount it will show you the updated Balance.

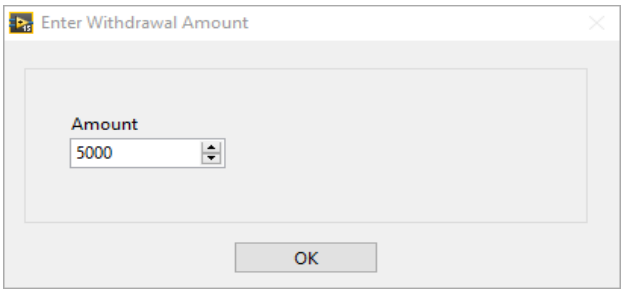

Figure 3.10: Withdrawal window

If user enter more than present balance it displays 'Insufficient balance.'

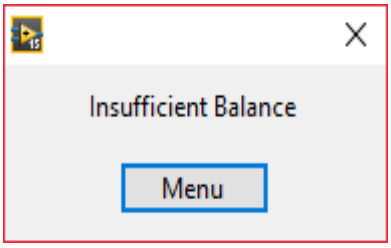

Figure 3.11: Insufficient Balance

If we click on the BALANCE it will show us the available Balance in our Account.

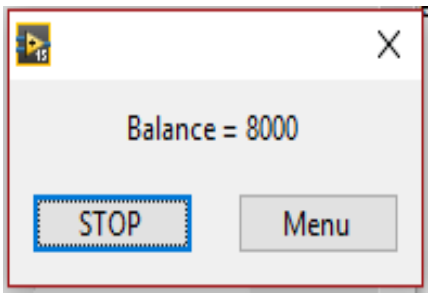

Figure 3.12: Balance Window.

After completing our transaction a pop up will appear on screen displaying STOP and MENU. If you click on the STOP button it will end your transaction, if you click on the MENU it will return you to the Main Menu. All the sequence of steps are designed and explained in figures from 3.4 to 3.12.

3.4 Manual Login

If you are an unauthorized user or you want to access your account through MANUAL LOGIN. You need to enter your account number and then an OTP is sent to the registered mobile number which is linked with the user account. Now after entering the OTP a transaction window will open displaying DEPOSIT, BALANCE, WITHDRAW. After every transaction your amount will be updated in the balance as mentioned above.

IV. CONCLUSION

The proposed project involves ATM management through fingerprint and manual login method in case of emergency access by others. This proposed method is to find a way to replace current model of ATM card and PIN. We even extended our project by generated Random OTP to guarantee high security for the users. One should enroll themselves at banks in order to access at ATMs which is a traditional way of getting card and PIN from banks. We can improve this model even further by replacing biometric with Iris technology which provides a great deal of security for these kind of transactions. We can add few conditions to the login method by implementing a code such that if user enters wrong OTP consecutively for 3 times, their account will be blocked from accessing and in order to unblock, they should provide bank specific reason for the block. We can add GPS module, so that it can send the location of user/culprit or in general where the transaction has been made and at which ATM transaction has been made for increased security. Other than login, we can add the ATM security by implementing an automatic shutter replacing the guard at the door

REFERENCES

1. Onyesolu, Moses Okechukwu, and Ignatius Majesty Ezeani, "ATM security using fingerprint biometric identifer: An investigative study," International Journal of Advanced Computer Science and Application, vol 3, no.4, pp 68-72, 2012.
2. Krishnamurthy, Pennam, and M. Maddhusudhan Redddy, "Implementation of ATM Security by Using Fingerprint recognition and GSM," International Journal of Electronics Communication and Computer Engineering, vol 3, no.1, pp 1-4, 2012.
3. Das, Shimal, and Jhunu Debbarma, "Designing a biometric strategy (fingerprint) measure for enhancing atm security in indian e-banking system," International Journal of Information and Communication Technology Research, vol 1, no.5, 2011.
4. Padmavathi, M., and R. Nagarajan, "Smart Intelligent ATM Using LABVIEW," International Journal of Emerging Technologies in Engineering Research, vol 5, no.5, pp 41-45, 2017.
5. Hrechak, Andrew K., and James A. McHugh. "Automated fingerprint recognition using structural matching." Pattern Recognition , vol 23, no.8, pp 893-904, 1990.
6. Brooks, Juliana HJ, "Method and system for biometric recognition based on electric and/or magnetic characteristics," U.S. Patent No. 6,898,299. 24 May 2005.
7. Mason, Alex, and Steve Parkes, "Using SpaceWire with LabVIEW," SpaceWire Conference (SpaceWire), IEEE, 2014.
8. Liang, Lv Xiangfeng1 Gao Honglin2 Ma, and Wang Xinhua, "Serial communication based on LabVIEW [J]," Foreign Electronic Measurement Technology, vol 12, no.8, 2009.
9. Elliott, Chance, et al, "National instruments LabVIEW: a programming environment for laboratory automation and measurement," JALA: Journal of the Association for Laboratory Automation, vol 12, no.1, pp 17-24, 2007.
10. Machacek, J., and J. Drapela, "Control of serial port (RS-232) communication in LabVIEW," Modern Technique and Technologies, International Conference. IEEE, 2008.
11. Suhag, Sahil, " Biometric Attendance System Using R-305 Sensor and Arduino UNO", Diss. 2016.
12. Kulkarni, Rushikesh, Muzammil Madki, and Tejas Mapari, "CARD-LESS ATM SYSTEM," International Education and Research Journa, vol 2, no.4, 2016.
13. Brijet, Z., B. Santhoshkumar, and N. Bharathi, "Vehicle Anti-Theft System Using Fingerprint Recognition Technique," Journal of Chemical and Pharmaceutical Sciences, ISSN 974: 2115.
14. Acharya, Sagar, Apoorva Polawar, and P. Y. Pawar, "Two factor authentication using smartphone generated one time password," IOSR Journal of Computer Engineering, vol 11, no.2, pp 85-90, 2013.
15. Indu, S., T. N. Sathya, and V. Saravana Kumar, "A stand-alone and SMS-based approach for authentication using mobile phone," Information Communication and Embedded Systems, International Conference on. IEEE, 2013.
16. Rao, T. Venkat Narayana, and K. Vedavathi. "Authentication using mobile phone as a security token." International Journal of Computer Science & Engineering Technology, vol 1, no.9, pp 569-574, 2011.
17. Schwartz, Marco, and Oliver Manickum, " Programming Arduino with LabVIEW," Packt Publishing Ltd, 2015
18. Telagam, Nagarjuna, Menakadevi Nanjundan, Nehru Kandasamy, and Soma Naidu. "Cruise Control of Phase Irrigation Motor Using SparkFun Sensor." *International Journal of Online Engineering (iJOE)* 13, no. 08 (2017): 192-198.
19. Kandasamy, Nehru, Nagarjuna Telagam, V. R. Seshagiri Rao, and T. Arulananth. "Simulation of analog modulation and demodulation techniques in virtual instrumentation and remote lab." *International Journal of Online Engineering (iJOE)* 13, no. 10 (2017): 140-147.
20. Telagam, Nagarjuna, Nehru Kandasamy, and Menakadevi Nanjundan. "Smart Sensor Network Based High Quality Air Pollution Monitoring System Using Labview." *International Journal of Online Engineering (iJOE)* 13, no. 08 (2017): 79-87.
21. Telagam, Nagarjuna, Nehru Kandasamy, Menakadevi Nanjundan, and T. S. Arulanandth. "Smart Sensor Network based Industrial Parameters Monitoring in IOT Environment using Virtual Instrumentation Server." *International Jour-nal of Online Engineering (iJOE)* 13 (2017): 111-119.
22. Nagarjuna Telagam, S.Lakshmi, K.Nehru,"BER Analysis of concatenated levels of encoding in GFDM system using labview", Indonesian journal of electrical engineering, vol 04, issue 1, 2019, pp 77-87.
23. Nagarjuna Telagam, S.Lakshmi, K.Nehru, " Digital audio broadcasting based gfdm transceiver using software defined radio", International journal of Innovative technology and Exploring Engineering, vol 8, issue 5, 2019, pp 273-281.
24. Telagam, Nagarjuna, Shailender Reddy, Menakadevi Nanjundan, and K. Nehru. "USRP 2901 Based MIMO-OFDM Transciever in Virtual and Remote Laboratory." International journal of computer sciences and Engineering, vol 6, issue 7, 2018, pp 1033-1040.
25. Somanaidu, Utlapalli, Nagarjuna Telagam, Nehru Kandasamy, and Menakadevi Nanjundan. "USRP 2901 Based FM Transceiver with Large File Capabilities in Virtual and Remote Laboratory." *International Journal of Online Engineering (iJOE)* 14, no. 10 (2018): 193-200.

AUTHORS PROFILE

Nagarjuna Telagam is currently working in ECE Department, GITAM University, Bangalore, India. His topics of interest are OFDM, GFDM and wireless sensor networks. He published 25 papers in Scopus indexed Journals and 4 SCIE journals. He got CLAD certification from National Instruments.

Sunita Panda is with Electronics and Telecommunication Engineering department, GITAM University, Bangalore campus. Currently she is Assistant Professor, Her areas of interest are soft computing, channel equalization, Digital Signal processing.

Nehru Kandasamy is with Electronics and Communication Engineering, Institute of Aeronatuical Engineering, Hyderabad, India.

Menakadevi Nanjundan is with the Electronics and Communication Engineering Department, Hindustan College of Engineering and Technology, Coimbatore, India.