

RFID Системa за контрол на достъпа

Йордан Живков Йорданов



July 4, 2019

# RFID – радиочестотна идентификация

## C:\Users\C#\AppData\Local\Microsoft\Windows\INetCache\Content.Word\rf.jpgКакво е RFID?

Технологията за радиочестотна идентификация е подготвена за някои значителни подобрения, стимулирани от това, което според анализаторите ще бъде по-бърз растеж в здравеопазването, търговията на дребно, безопасността на храните и други пазари. Бъдещето на RFID се разраства и разширява, тъй като повече индустрии и компании инвестират в технологията. В резултат на това RFID става все по-рентабилен от всякога за решаване на реални бизнес предизвикателства.

Технологията за [радиочестотна идентификация](https://lowrysolutions.com/solutions/rfid-technology/) ( [RFID](https://lowrysolutions.com/solutions/rfid-technology/) ) използва електромагнитни полета за предаване на данни от RFID етикет към четец, като осигурява точни данни за проследяване в реално време. За разлика от баркод скенери, RFID четците не се насочват директно към етикет, за да получат сканиране.

## Анализи и прогнози

Анализаторите предвиждат експлозия на RFID за последното десетилетие; пазарът обаче нараства бавно и стабилно. Докладът за [проучвания и пазари](http://www.businesswire.com/news/home/20150113005541/en/Research-Markets-Global-Sourcing-RFID-Tags-Market#.VLhyvSfEiLY) обаче предвижда глобално снабдяване с RFID тагове да има сложен годишен темп на растеж от 22,4% до 2018 г. Още един [доклад](http://www.virtual-strategy.com/2015/01/15/smart-label-market-eas-rfid-sensing-esl-worth-10-billion-2020-new-report-marketsandmarket#axzz3OwjneRIh) поставя пазара на интелигентни етикети на 10 милиарда долара до 2020 година.

Някои експерти смятат, че пазарът на пасивни RFID етикети само ще нарасне до почти 7 милиарда етикети през следващата година.

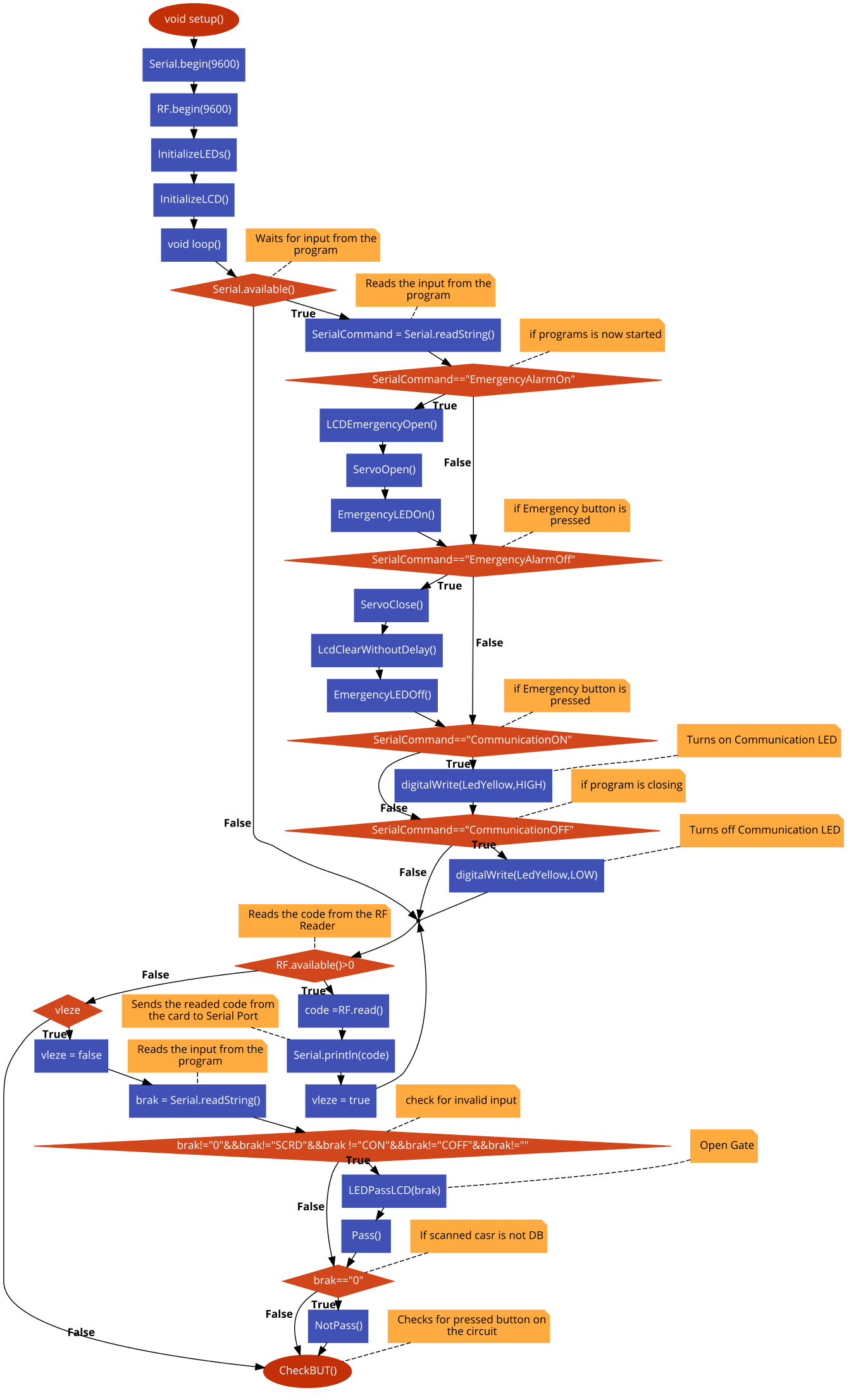
## Бъдещи иновации за RFID

Наред с по-широкото внедряване, новите технологии ще направят RFID по-надежден и рентабилен за по-голям брой приложения.

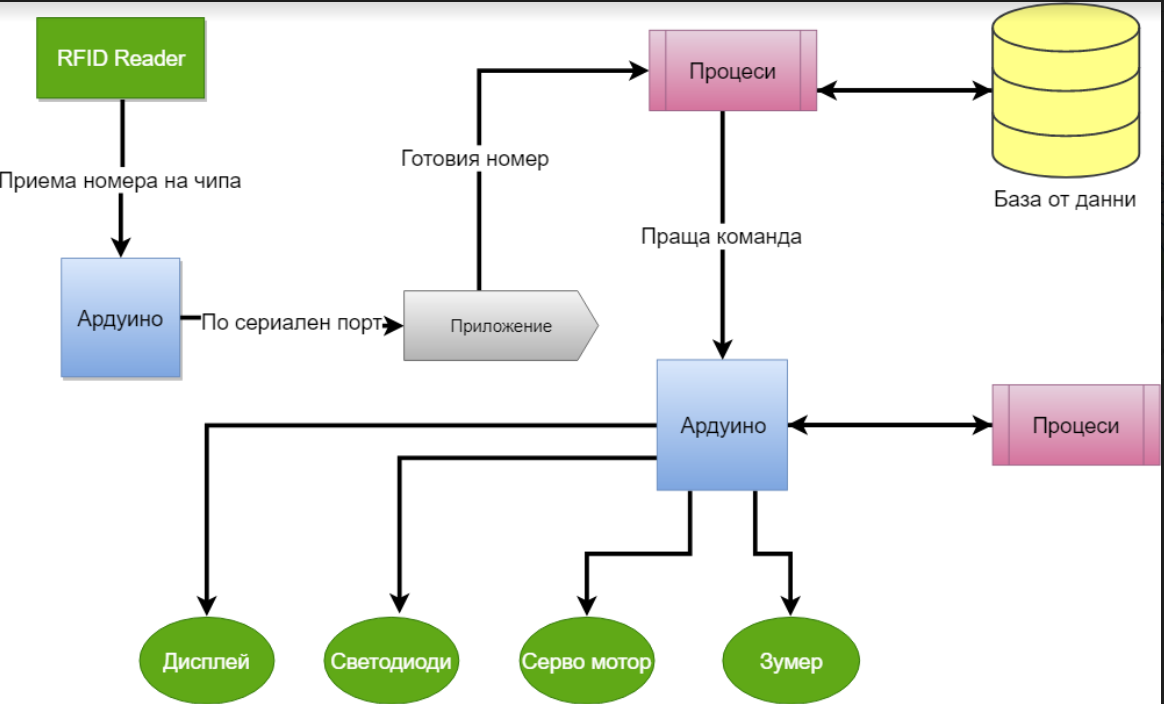


# C:\Users\C#\AppData\Local\Microsoft\Windows\INetCache\Content.Word\ProjectWires.pngЕлектрическа схема

# Блокова схема на програмното осигуряване за хардуера



# Блокова схема на Софтуер и Хардуер



# Employee-Management

1. Тема:

**Employee-Management** – това е проект, който представлява софтуер управяляващ хардуер. Темата на този проект е RFID система за контрол на достъпа. Целта на проекта е да се ограничи достъпа на външни лица до дадена сграда и да се пази в база от данни кой кога е влизал

и колко часа е работил месечно.

## Автор:

* **Йордан Живков Йорданов**, ЕГН: 0143285804, телефон: 088 980 3172, имейл: **dani\_yordani@abv.bg**, училище: ППМГ „Добри Чинтулов“, клас: 11д
* **Ръководител**: **доц. д-р Димитър Минчев,** тел: +359 899 148 872 , имейл: dimitar.minchev@gmail.com , преподавател към Център по информатика и технически науки при Бургаски свободен университет (БСУ).

## Резюме:

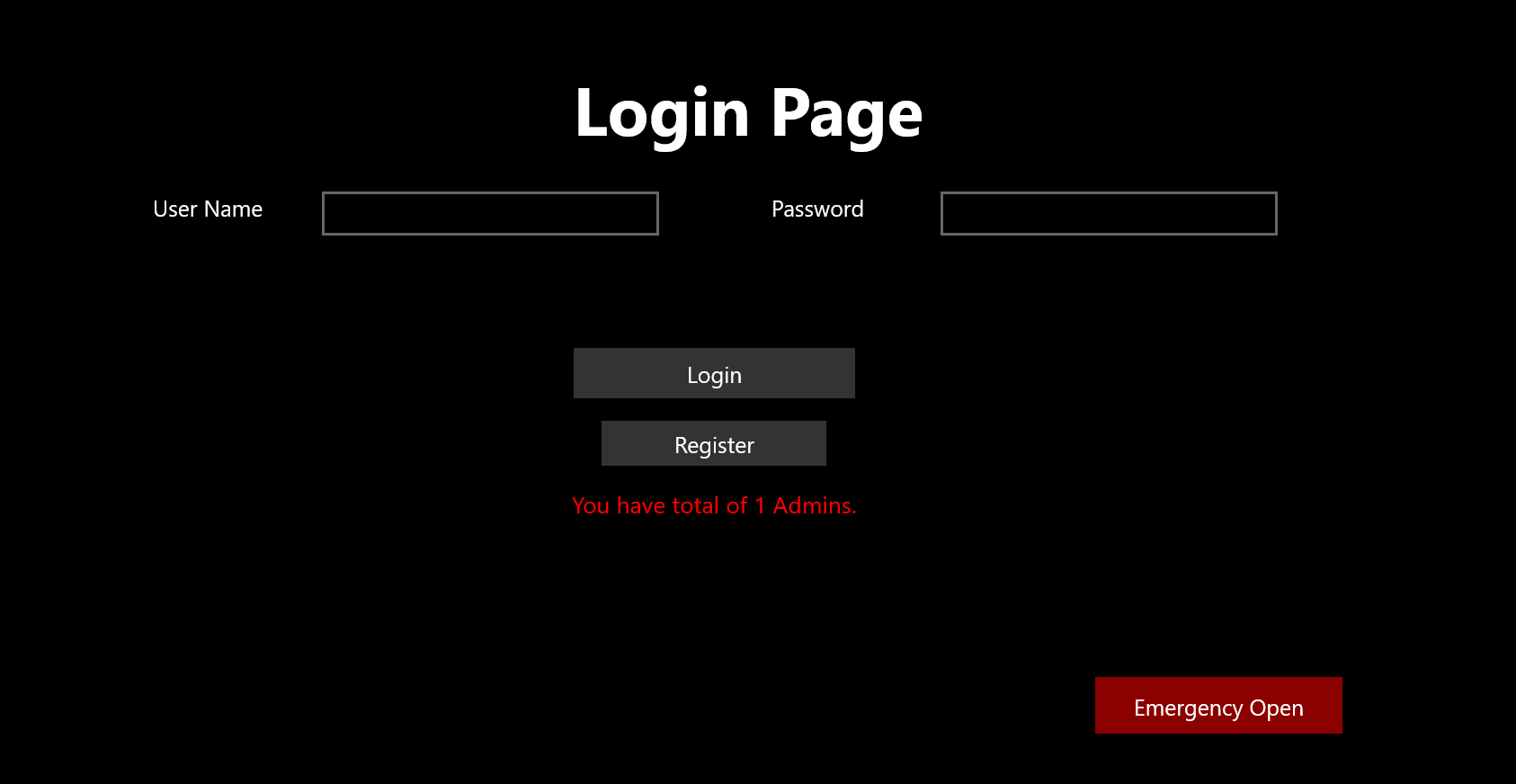
* 1. Пректът Employee-Management се състой от софтуер и хардуер. Софтуерът е разработен на Visual Studio и на платформата UWP (Универсалната Windows платформа) създадена от Microsoft използвайки: C ++ , C # , VB.NET и XAML. Софтуерът има за цел да складира в база от данни всички служители (с тяхната информация) от една компания. Предоставя информация за всеки служител кога е влязъл и напуснал сградата и колко часа е работил. Също пази информаця колко часа е работил служител за целия месец (което улеснява калкулирането му на месечна заплата).
  2. Хардуерът представлява RFID четец които прочита индефициращият код на една карта или чип

и го праща по сериален порт на Софтуерът. Има и бутон които служи за отключване на ключалката ръчно без да се нуждае Хардуерът от връзка с база от данни (Софтуерът). За олесняването на потребителя съм предоставил светодиоди които предоставят информация (кога има комуникация между Хардуер и Софтуер, кога може да влезе и кога неможе ) и също така и Дисплей който предоставя името и фамилията на човека сканирал чипа (Дисплеят показва чрез текст кога ключалката е отворена поради инзвънредна ситуация или ръчно някой я е задействал).

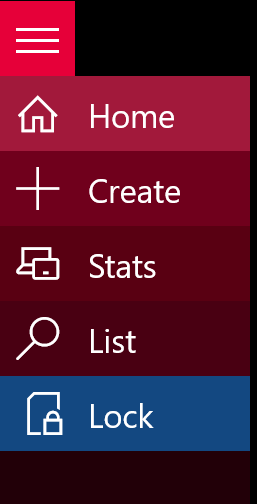
1. Основните етапи в реализирането на проекта са разделени в няколко групи:

* оформяне на основната идея на проекта;
* схема на Хардуерната част
* Реализиране на основната част от Хардуерът
* Създаване на база от данни
* Създаване на страници
* Валидация на страници
* Създаване на логиката по MVC модел.
* Оправяне на конвенциите за писане на код
* Завършване на Хардуерът

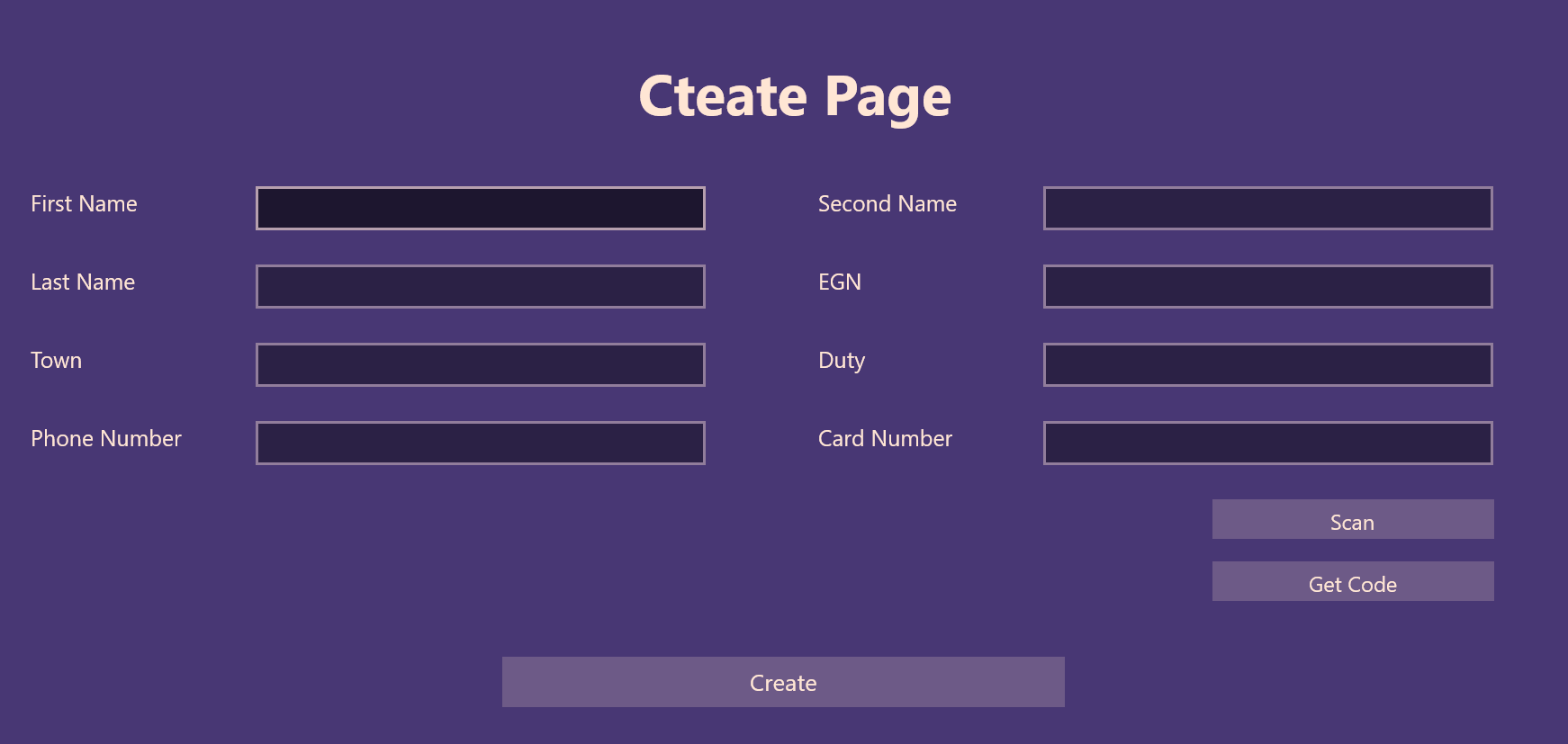
## Описание на Софтуерът



При стартирането на проекта сме посрещнати от Логин страницата която изисква от нас да въведем потребителско име и парола и ни дава възможност да се отправим към Регистър страницата с бутона „Register“ . Текстовото поле под този бутон ни предоставя колко активни потребителя имаме в текущо време. „Emergancy Open“ бутона ни дава опцията да отключим всички ключалки при извънредна ситуация.

Навигационият панел ни предоставя 5 опции за пренасочване към отделните страници.

* Първата ни препраща към главната страниця на приложението.
* Втората ни препраща към страница за създаване на потребители
* Третата ни препраща към страницата която показва кой кога е влизал и излязъл
* Четвъртата но препраща към списък от всички служители
* Петата заключва приложението и го препраща към Логин страницата

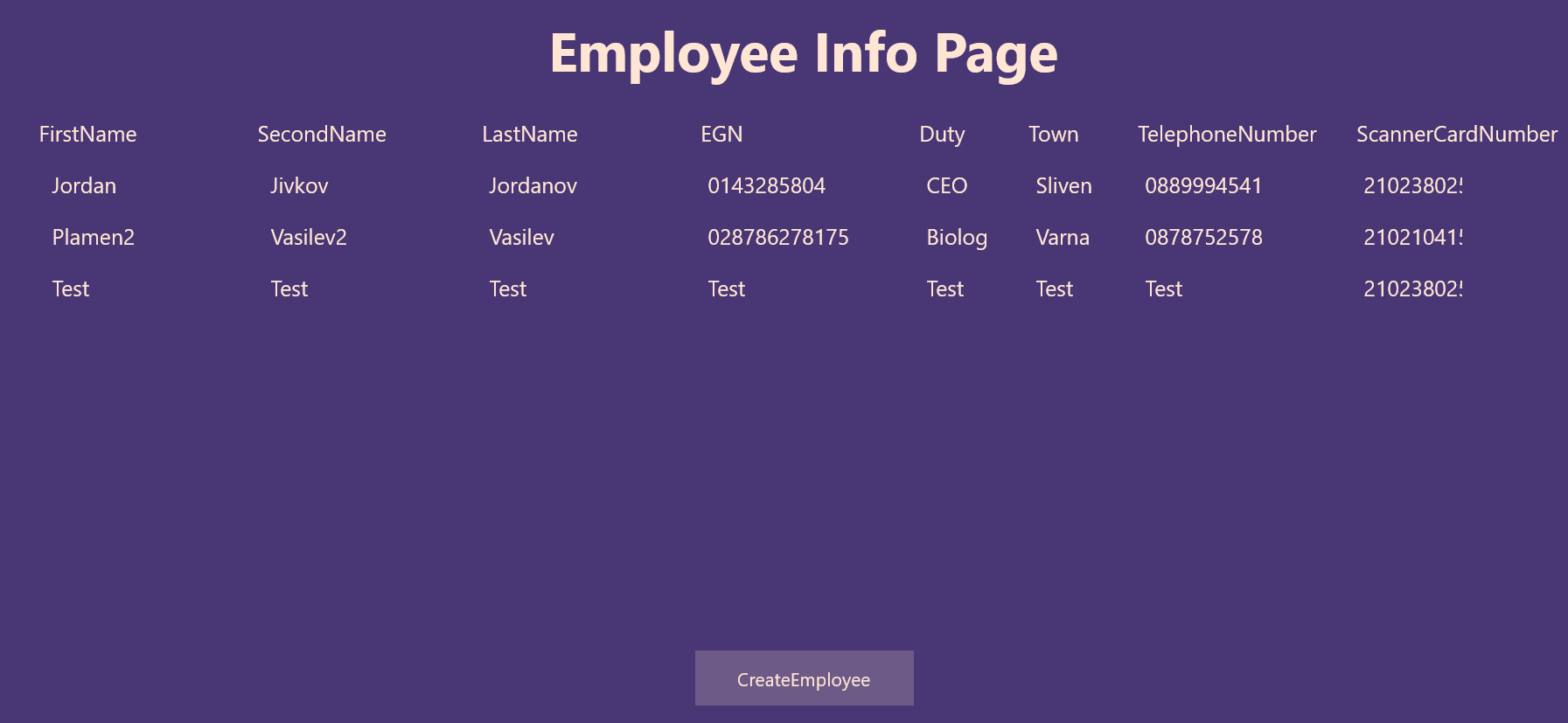


Това е Create страницата която има за предназначение да създава нови служители. Не може да се създаде нов служител дори и ако едно от полетата е непопълнено . Полето „Card Number“ се попълва по следния начин: 1- натиска се бутон „Scan“ след това се сканира чипа който ще пренадлежи на дадения служител и чак след това се натиска бутона „Get Code“ който поставя индефикационият номер на сканираната карта в текстовото поле.

Това е страницата която ни предоставя информация за влизането и излизането на служителите.

Имамеме падащ списък котйто ни предоставя две опции да видим кой и кога е влязъл също колко време е работил (ако е излязъл се разбира колко часа е рабтил) за днеска. А другата опция ни предоставя същите данни като предишната но само че за периода от един месец.

Бутонa „Refresh“ ни обновява списака и го прави актуален.



Тази страница ни предоставя списък с всички регистрирани служители с тяхна най важна информация при натискане на даден служител от списъка страницата ни препраща в „Edit“ страницата.

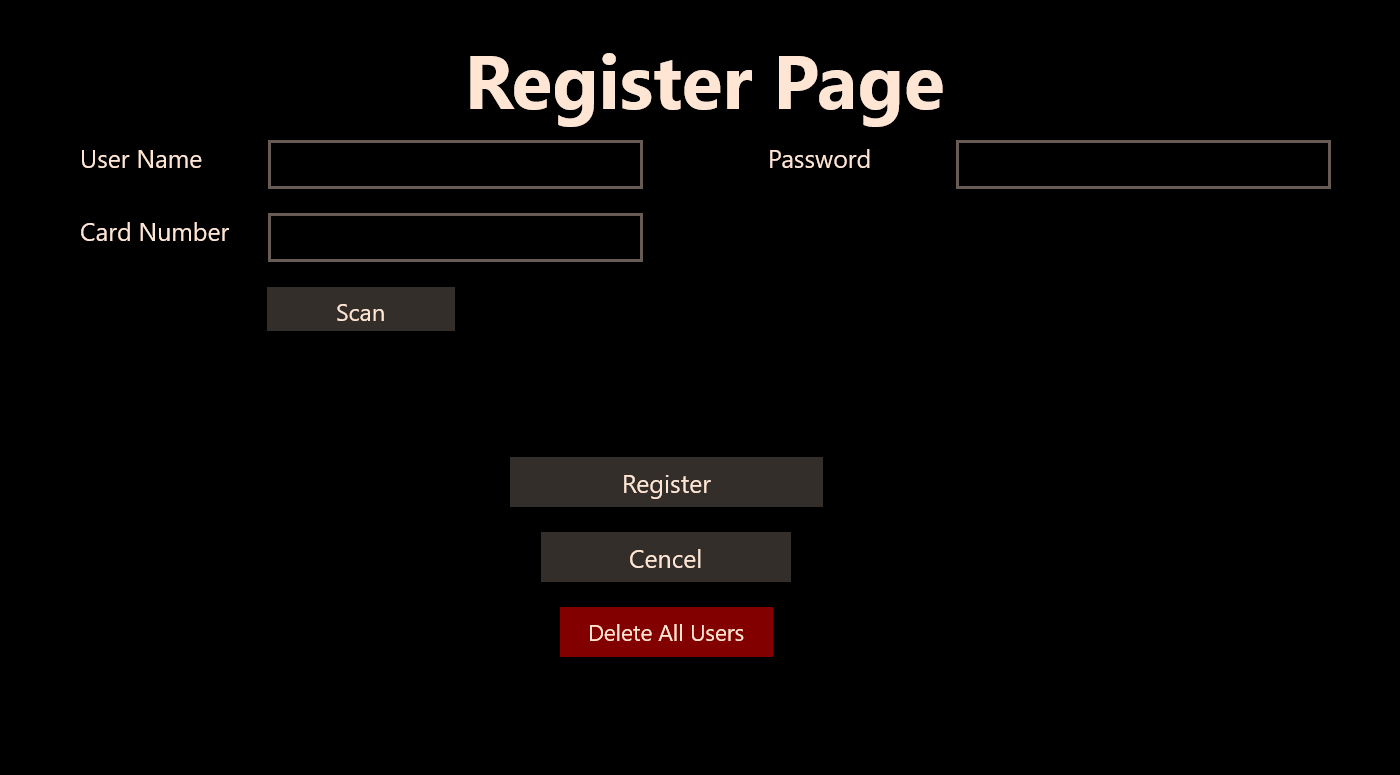
Бутона „Create Empoyee“ ни препраща към „Create“ страницата.

Тази страница ни предоставя възожноста да редактираме всяко от полетата на даден служител.

Бутона „ Edit“ ни потвърждава и запазва всички промени по даден служител

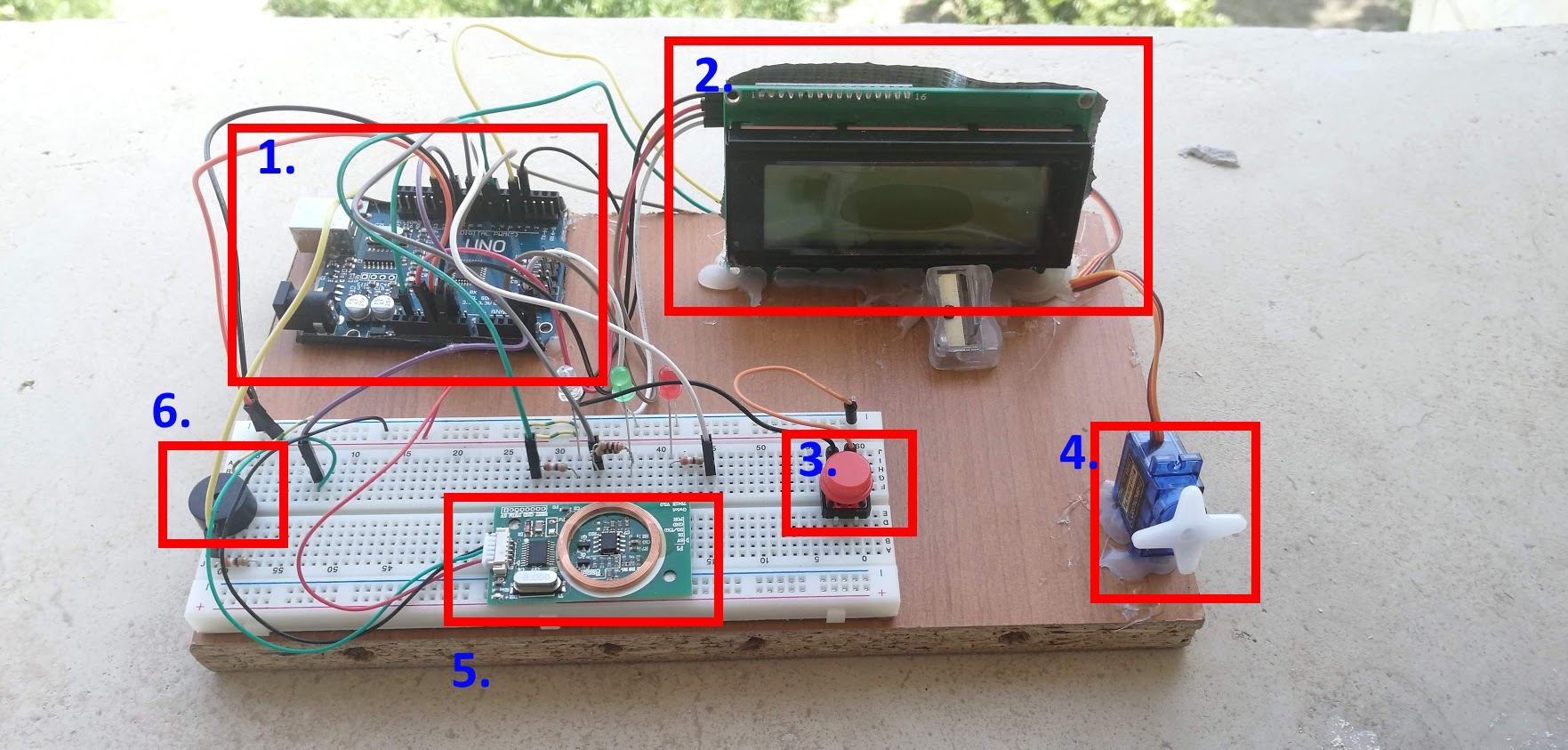
Бутона „Cencel“ ни отказва от всички въведени промени

Бутона “Delete” изтрива служителя от всички таблици но преди да се извърши тази операция ни излиза потвърждаващо поле.



Тази страниця ни предоставя възможноста да създаваме админи които да имат достъп до данните на служителите.

За да създадем админ ние трябва да предоставим в полето “Card Number” индефикационият номер на „Mastar Card“ която я предоставяме ние.

Имаме опциятя да изтрием всички админи регистрирани до момента но преди това ще ни излезе потвърждаващо поле. 

**Съставни части:**

1. **ARDUINO UNO R3 (CH340G)**

* Осъществява комуникацияте между софтуер и хардуер.

1. **Character Display LCD Modules 20x4**

* Насочва потребителя
* Иписва името на потребителя който е сканирал

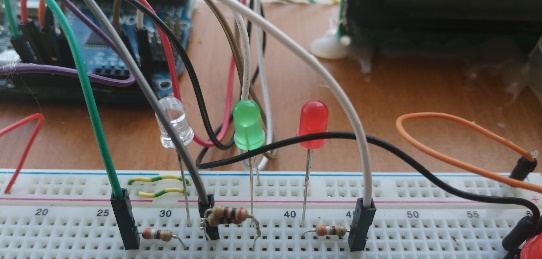
1. **Бутон**

* Ръчно отключва и заключва „ключалката“
* Без да има връзка към програмата

1. **Micro continuous Servo TS90A**

* Играе ролята на ключалка

1. **RFID Reader**

* Прочита номера на картата
* Праща индификационният номер на програмата

1. **Зумер**

* Задейстава се при отваряне и затваряне на ключалката

1. **Ляв светодиод**

* Мига (в синя светлина)
* Индикира за осъществена комуникация между хардуер и софтуер

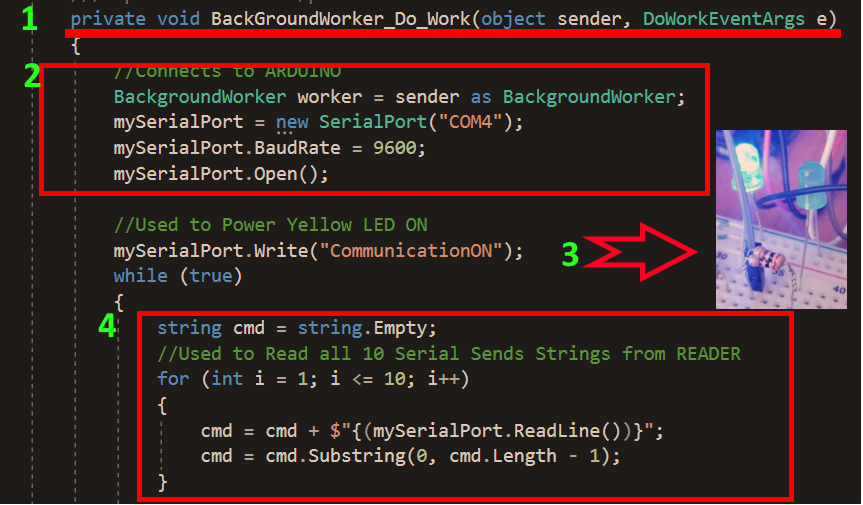
1. **Среден светодиод**

* Светва (в зелена светлина) когато кода на сканираната карта е базата

1. **Десен светодиод**

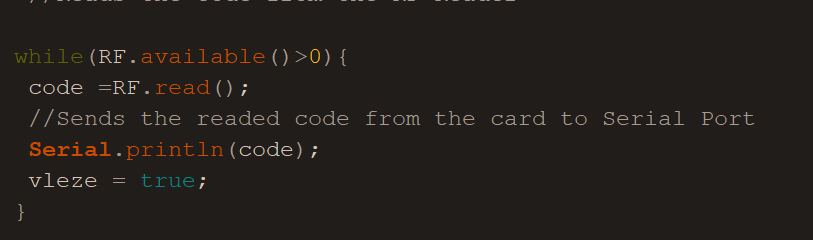
* Светва (в червена светлина) когато кода на сканираната карта не е в базата

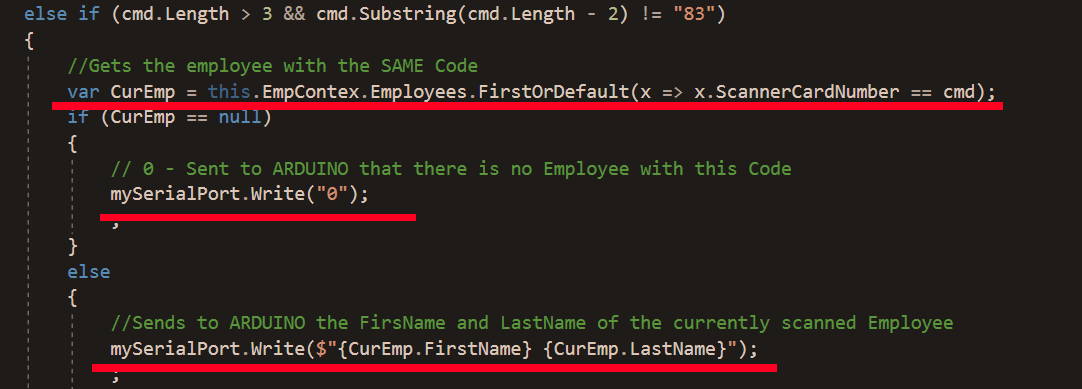
# Описание на функционалноста

Отключване и заключване на ключалка

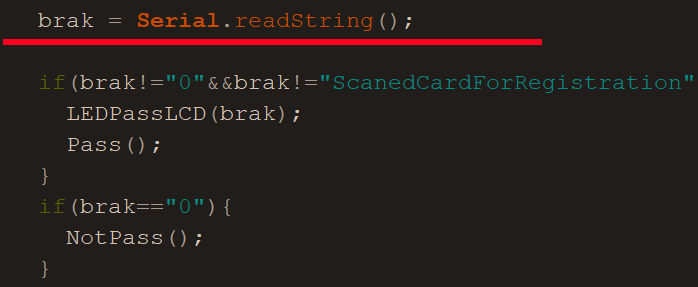
При стартирането на програмата се активира „BackGroundWorker“

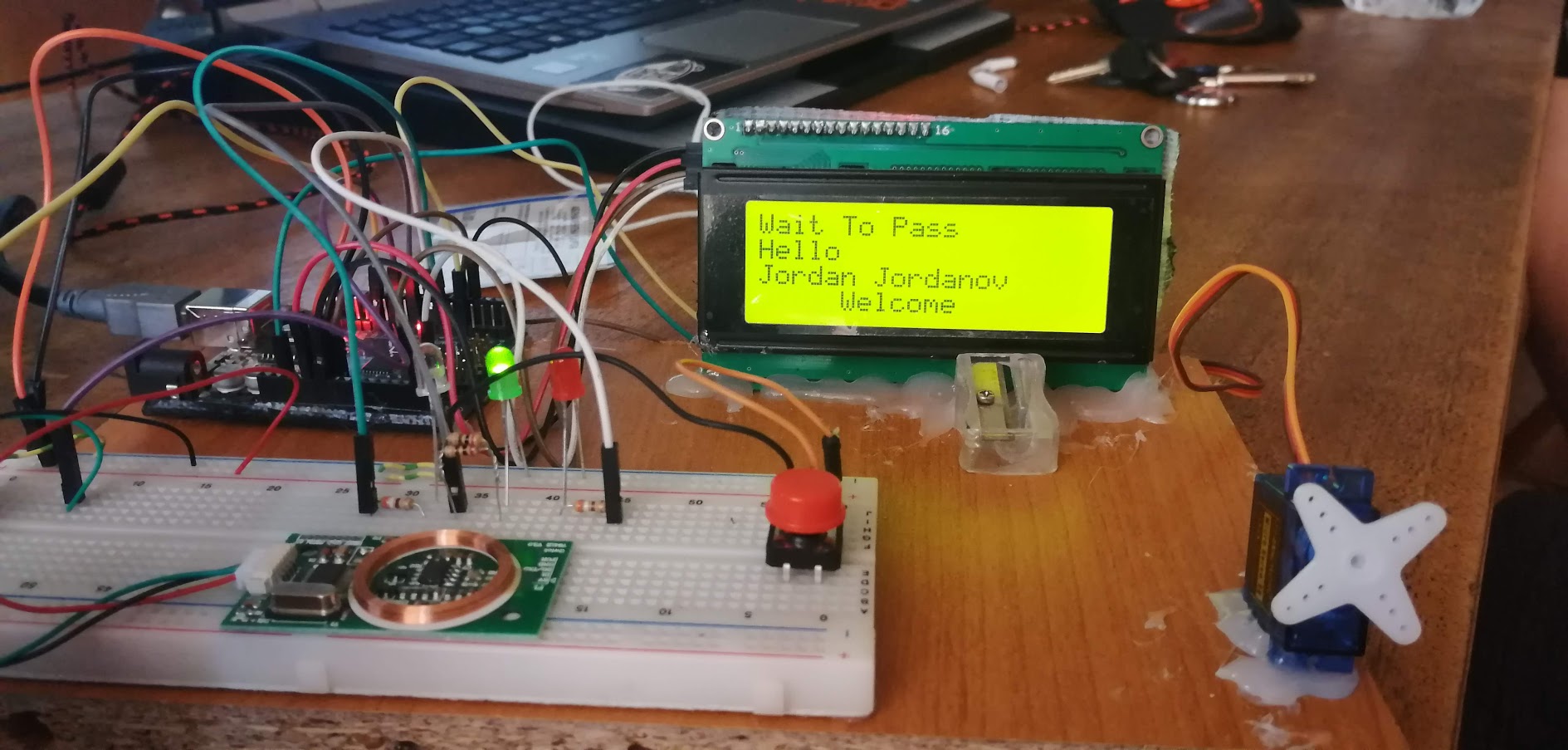
1. Това е метод на „BackGroundWorker“ който се изпълнява през цялато време когато програмата е пусната.
2. Осъществява се връзката по сериален порт между програмата и Ардуиното.
3. Праща се команда по сериален порт на Ардуиното която влючва Светодиод които индикира комуникацията между Софтуер и Хардуер.
4. Цикълът чака индификационият номер на сканираната карта от Четецът и го събита в една променлива.

В ардуиното когато някой доближи карта до четеца се задейства while цикълът и се прочита номерът на картата (разделен на десет части) и се праща на Serial port.

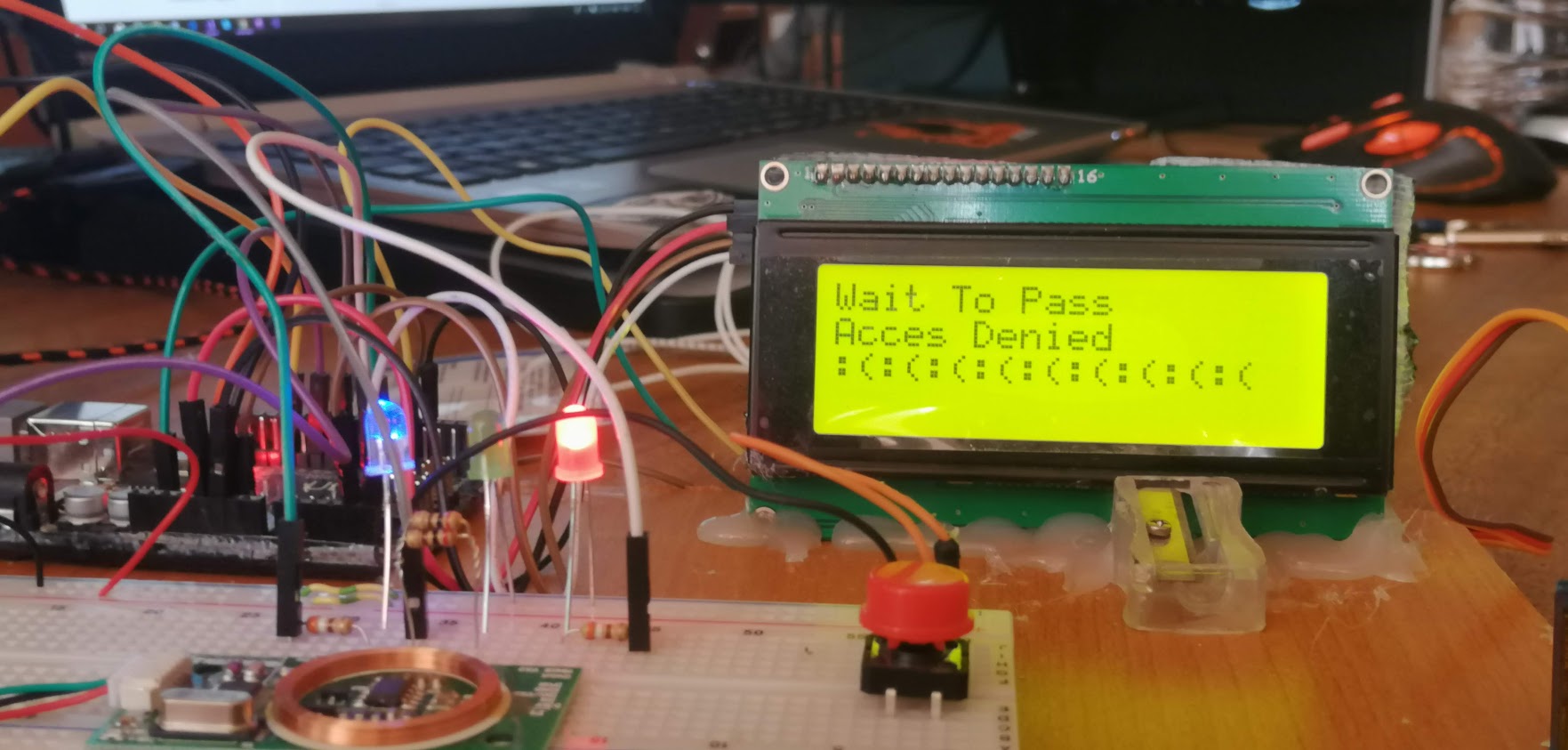


Взема се готовата променлива и се проверява в базата от данни дали има служител с такъв номер. Ако нама такъв ще върне „0“, а ако има ще му върне двете имена по сериален порт на Ардуиното.

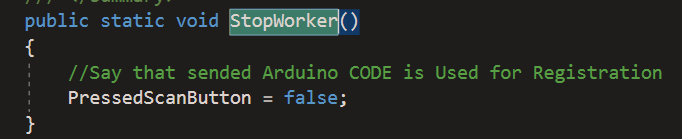
Първо Ардуиното прочита командата пратена по сериален порт от “Visual Studio” и прави нужните проверки

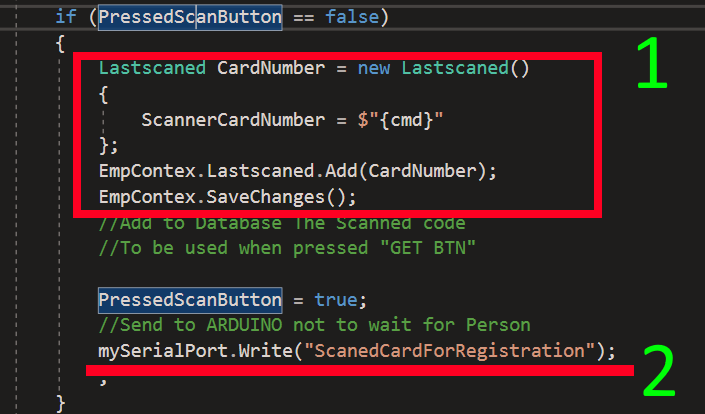
Достъпът приет

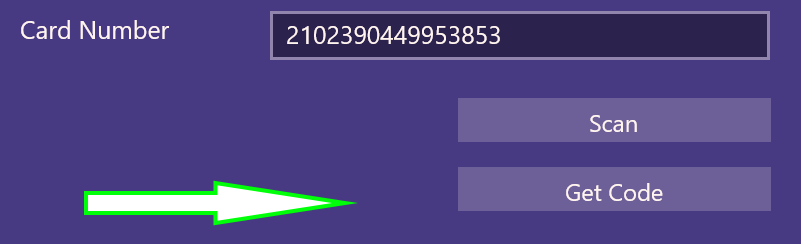
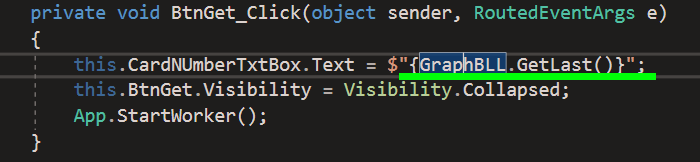
Достъпа отказан

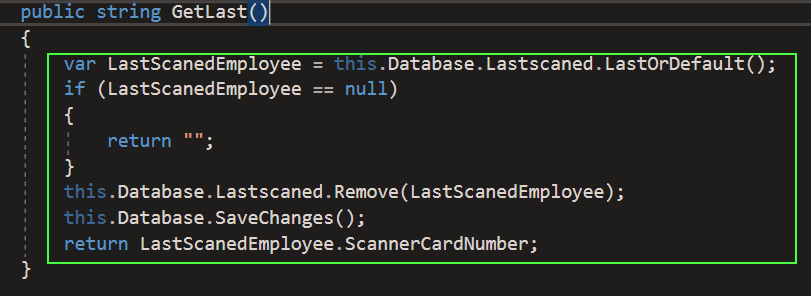


# C:\Users\C#\AppData\Local\Microsoft\Windows\INetCache\Content.Word\2.pngC:\Users\C#\AppData\Local\Microsoft\Windows\INetCache\Content.Word\1.pngНомер на картата на служител

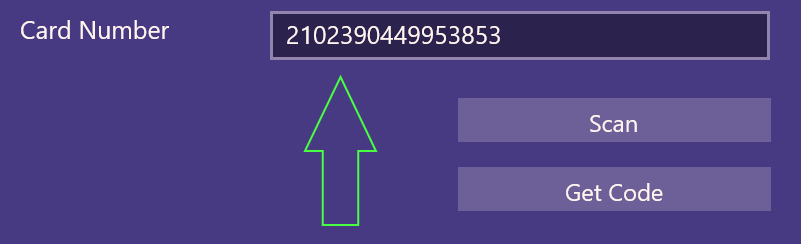


1. Праща в таблицата от базата „Last Scaned“ току що сканирания индефикациония код на сканираната карта
2. Праща команда на Ардуиното (да бездействиа☺)



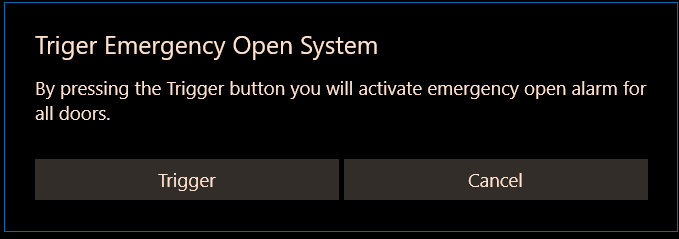


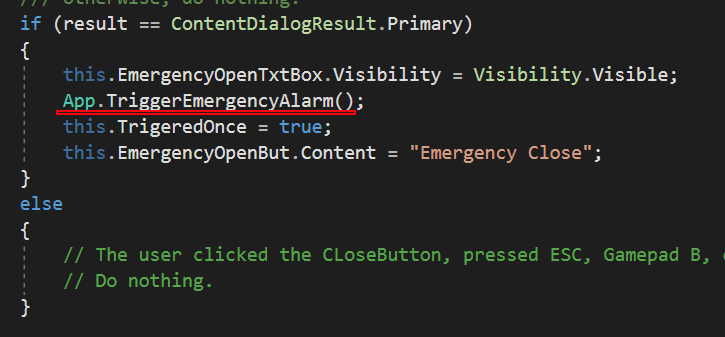
Взема от таблицата (ако потребителя е сканирал нещо ) последния запис и го постявя в текстовот поле след това изтрива записа от таблицата.



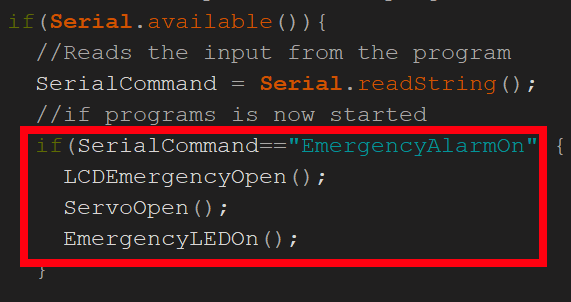
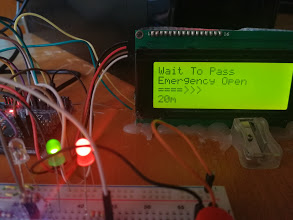
## C:\Users\C#\AppData\Local\Microsoft\Windows\INetCache\Content.Word\a.pngБутон за инзвънредни ситуации

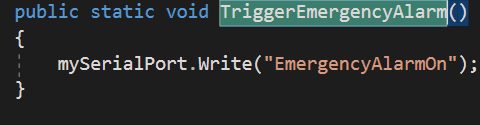
Този бутон има задача да отключи ключалките при инзвънредни ситуации



При потвърждаване натискането на бутона „Emergancy Open“ ако го натискме за първи път се появява прозореца за потвърждаване ако не викаме метода „TriggerEmergencyAlarm“ на статичния клас „App“.

Този статичен метод праща по сериален порт на ардуино следната команда





# Сорс код – Описание на функционалноста

#include <SoftwareSerial.h>

#include <LiquidCrystal\_PCF8574.h>

#include <Wire.h>

#include <Servo.h>

#include <Servo.h>

Servo ServoLock;

const int LedGreen = 8;

const int LedRed = 7;

const int LedYellow = 6;

const int Zoomer = 12;

const int buttonPin = 3;

bool buttonState ;

SoftwareSerial RF(10, 11); // RX, TX

int cardNumber;

bool wasInWhile = false;

String brak;

String SerialCommand;

LiquidCrystal\_PCF8574 lcd(0x27); // set the LCD address to 0x27 for a 16 chars and 2 line display

void setup() {

Serial.begin(9600);

RF.begin(9600);

ServoLock.attach(4);

digitalWrite(Zoomer,LOW);

pinMode(buttonPin, INPUT\_PULLUP);

InitializeLEDs();

InitializeLCD();

}

void loop() {

//Waits for input from the program

if(Serial.available()){

//Reads the input from the program

SerialCommand = Serial.readString();

//if prograServoLock is now started

if(SerialCommand=="EmergencyAlarmOn"){

//Displays on LCD "Emergency Open"

LCDEmergencyOpen();

ServoOpen();

//Power onn both LED

EmergencyLEDOn();

//When you cencel the Emergency

if(SerialCommand=="EmergencyAlarmOff"){

ServoClose();

LcdClearWithoutDelay();

EmergencyLEDOff();

}

if(SerialCommand=="CommunicationON"){

//Turns on Communication LED

digitalWrite(LedYellow,HIGH);

}

//if program is closing

if(SerialCommand=="CommunicationOFF"){

//Turns off Communication LED

digitalWrite(LedYellow,LOW);

}

}

//Reads the cardNumber from the RF Reader

while(RF.available()>0){

cardNumber =RF.read();

//Sends the readed cardNumber from the card to Serial Port

Serial.println(cardNumber);

wasInWhile = true;

}

if(wasInWhile){

wasInWhile = false;

//Reads the input from the program

brak = Serial.readString();

//Checks for false Scanned text

if(brak!="0"&&brak!="ScanedCardForRegistration"&&brak !="CommunicationON"&&brak!="CommunicationOFF"&&brak!=""){

LEDPassLCD(brak);

Pass();

}

if(brak=="0"){

NotPass();

}

}

//Checks for pressed "Manual Opne" button

CheckBUT();

}

//Reads the cardNumber from the RF Reader

while(RF.available()>0){

cardNumber =RF.read();

//Sends the readed cardNumber from the card to Serial Port

Serial.println(cardNumber);

wasInWhile = true;

}

if(wasInWhile){

wasInWhile = false;

//Reads the input from the program

brak = Serial.readString();

//Checks for false Scanned text

if(brak!="0"&&brak!="ScanedCardForRegistration"&&brak !="CommunicationON"&&brak!="CommunicationOFF"&&brak!=""){

LEDPassLCD(brak);

Pass();

}

if(brak=="0"){

NotPass();

}

}

//Checks for pressed "Manual Opne" button

CheckBUT();

}

# З

void Pass(){

digitalWrite(Zoomer,HIGH);

ServoOpen();

LEDPass();

digitalWrite(Zoomer,LOW);

LcdClear();

ServoClose();

}

void NotPass(){

LCDNotPass();

LEDNotPass();

LcdClear();

}

void InitializeLCD(){

Wire.begin();

Wire.beginTransmission(0x27);

lcd.begin(20, 4); // initialize the lcd

lcd.setBacklight(255);

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Scanning...");

}

void InitializeLEDs(){

pinMode(LedGreen,OUTPUT);

pinMode(LedYellow,OUTPUT);

pinMode(LedRed,OUTPUT);

digitalWrite(LedYellow,LOW);

}

void LEDPass(){

digitalWrite(LedGreen,HIGH);

delay(2200);

digitalWrite(LedGreen,LOW);

}

void LEDPass(){

digitalWrite(LedGreen,HIGH);

delay(2200);

digitalWrite(LedGreen,LOW);

}

void EmergencyLEDOn(){

digitalWrite(LedGreen,HIGH);

digitalWrite(LedRed,HIGH);

}

void EmergencyLEDOff(){

digitalWrite(LedGreen,LOW);

digitalWrite(LedRed,LOW);

}

void ServoOpen(){

ServoLock.write(50);

delay(653);

ServoLock.write(90);

}

void ServoClose(){

ServoLock.write(150);

delay(653);

ServoLock.write(90);

}

void CheckBUT(){

//If button is pressed

if(digitalRead(buttonPin)==LOW){

//If button was pressed before

if (buttonState == LOW) {

digitalWrite(LedGreen,HIGH);

// digitalWrite(Zoomer,HIGH);

LCDOpenedByManual();

ServoOpen();

}

else {

// turn LED off:

digitalWrite(LedGreen,LOW);

// digitalWrite(Zoomer,LOW);

LcdClearWithoutDelay();

ServoClose();

}

//Change the button state to opasite

buttonState=!buttonState;

delay(200);

}

}

void LEDNotPass(){

for(int a=0;a<2;a++)

{

digitalWrite(LedRed,HIGH);

digitalWrite(Zoomer,HIGH);

delay(1000);

digitalWrite(Zoomer,LOW);

digitalWrite(LedRed,LOW);

delay(1000);

digitalWrite(Zoomer,LOW);

}

}

void LCDOpenedByManual(){

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Wait To Pass");

lcd.setCursor(0, 1);

lcd.print("Opened By Manual");

}

void LCDEmergencyOpen(){

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Wait To Pass");

lcd.setCursor(0, 1);

lcd.print("Emergency Open");

lcd.setCursor(0, 2);

lcd.print("====>>>");

lcd.setCursor(0, 3);

lcd.print("20m");

}

void LEDPassLCD(String EmpName){

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Wait To Pass");

lcd.setCursor(0, 1);

lcd.print("Hello");

lcd.setCursor(0, 2);

//Prints the name of scanned Eployee

lcd.print(EmpName);

lcd.setCursor(5, 3);

lcd.print("Welcome");

}

void LCDNotPass(){

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Wait To Pass");

lcd.setCursor(0, 1);

lcd.print("Acces Denied");

lcd.setCursor(0, 2);

lcd.print(":(:(:(:(:(:(:(:(:(");

}

void LcdClearWithoutDelay(){

lcd.clear();

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Scanning...");

}

void LcdClear(){

delay(3000);

lcd.clear();

lcd.clear();

lcd.setCursor(0, 0);

lcd.print("Scanning...");

}

# Заключение

## Положителни страни

### Проектът позволява да се следи колко часа работи на ден един служител

### Проектът позволява да се следи колко часа е работил за един месец

### Проектът позволява лесно да се редактират данните на служителите

### Улеснен и опростен дизайн на приложението

### Проектът предоставя опция за инзвънредно отваряне на ключалките

### Проектът предоставя опция за ръчно отваряне на ключалките без да изисква връзка със софтуер

### Проектът ограничава достъпа на външни лица в дадената сграда

### Приложението към проекта неможе да се достъпи от всеки

## Отрицателни страни

### Липса на Bluetooth модул

### Изискването на постоянна връзка Приложение и Хардуер

### Липса на Уеб приложение

Contents

[RFID – радиочестотна идентификация 1](#_Toc13213081)

[Какво е RFID? 1](#_Toc13213082)

[Анализи и прогнози 1](#_Toc13213083)

[Бъдещи иновации за RFID 1](#_Toc13213084)

[Електрическа схема 2](#_Toc13213085)

[Блокова схема на програмното осигуряване за хардуера 3](#_Toc13213086)

[Employee-Management 5](#_Toc13213087)

[1. Тема 5](#_Toc13213088)

[2. Автор: 5](#_Toc13213089)

[3. Резюме: 5](#_Toc13213090)

[4. Основните етапи в реализирането на проекта са разделени в няколко групи 6](#_Toc13213091)

[5. Описание на Софтуерът 6](#_Toc13213092)

[Описание на функционалноста 11](#_Toc13213093)

[Номер на картата на служител 13](#_Toc13213094)

[Заключение 15](#_Toc13213095)

[Положителни страни 24](#_Toc13213096)

[Проектът позволява да се следи колко часа работи на ден един служител 24](#_Toc13213097)

[Проектът позволява да се следи колко часа е работил за един месец 24](#_Toc13213098)

[Проектът позволява лесно да се редактират данните на служителите 24](#_Toc13213099)

[Улеснен и опростен дизайн на приложението 24](#_Toc13213100)

[Проектът предоставя опция за инзвънредно отваряне на ключалките 24](#_Toc13213101)

[Проектът предоставя опция за ръчно отваряне на ключалките без да изисква връзка със софтуер 24](#_Toc13213102)

[Проектът ограничава достъпа на външни лица в дадената сграда 24](#_Toc13213103)

[Приложението към проекта неможе да се достъпи от всеки 24](#_Toc13213104)

[Отрицателни страни 24](#_Toc13213105)

[Липса на Bluetooth модул 24](#_Toc13213106)

[Изискването на постоянна връзка Приложение и Хардуер 24](#_Toc13213107)

[Липса на Уеб приложение 24](#_Toc13213108)

[Бутон за инзвънредни ситуации **Error! Bookmark not defined.**](#_Toc13213109)