**IDEA:** i-LOCKER

**i stand for Integrated / Intelligent Invention**

**Locker is a place keep the parcels safe and well-arranged**

**INTRODUCTION**

1. Opening : Related ideas to the real problems issue ( a situation faced by students in Cau Giay District)

-The COVID-19 pandemic and ensuing stay-at-home orders, lockdown measures, and general reluctance to shop in-person led to a further increase in online shopping. Based on online news (VIETNAM PLUS) it stated that in Hanoi the online shopping will still be booming this year, even if the COVID-19 pandemic is over, experts have said. Massive growth in the digital economy due to ecommerce. The Vietnamese market is also expected to quadruple to 39 billion USD by 2025 from 13 billion USD last year. Suitable for future development.

Vietnam's economic growth slowed down due to the Covid-19 pandemic, but in 2021, the digital economy in general and e-commerce in particular was a bright spot. According to a report by Google, Temasek and Bain&Company, the total merchandise value (GMV) of the e-commerce industry in Vietnam grew dramatically from 8 billion USD in 2020 to 13 billion USD in 2021.

The figures clearly demonstrate that Vietnam's e-commerce industry has successfully overcome difficulties caused by Covid-19 and is thriving in the context of the "new normal", experts said. This development was even more impressive than in the pre-pandemic period.

**What is an i-Locker?**

i-Locker is an automatic unlocking locker to place the parcels from courier services.

**OBJECTIVES?**

To enhance security and privacy for the student’s parcels

To save time

To keep the environment of university a well-arranged and tidy

To identify the

To prevent the parcel from swapping between students.

**OPERATION?**

1. A smart locker that consists of 100 sections with a single 3x4 matrix keypad which has a lock system.
2. The shipper from courier services will put the parcels in a lockers section which limits one section per receiver.
3. The shippers will generate a new passcode through an application for every tracking number. Then, provide the receivers( students) with a passcode via email. Email is the best platform to send the passcode as it will receive to the right owner of the ecommerce account.
4. The passcode will expire when the student collects and the new passcode will be generated for the next parcel.
5. The passcode will expire if the students do not collect the parcels after 24 hours and the shipper will collect the parcels and bring them back to the hub.
6. The operation will repeat.

**Improvement can be made to this project for the future?**

link library: <https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json>

Code:

//BY ARDUICREATE

//INCLUDE THESE LIBRARYS

#include <Keypad.h>

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

//#include <SoftwareSerial.h>

//SoftwareSerial mySerial(2, 3);

LiquidCrystal\_I2C lcd(0x27, 16, 2);

#define Password\_Lenght 5 // Give enough room for six chars + NULL char

char Data[Password\_Lenght]; // 6 is the number of chars it can hold + the null char = 7

char codeLocker1[Password\_Lenght] ="1111";

char codeLocker2[Password\_Lenght]= "2222";

char codeLocker3[Password\_Lenght]= "3333";

char codeLocker4[Password\_Lenght]= "4444" ;

byte data\_count = 0, master\_count = 0;

bool Pass\_is\_good;

char customKey;

const byte ROWS = 4;

const byte COLS = 4;

char hexaKeys[ROWS][COLS] = {

{'1', '2', '3', 'A'},

{'4', '5', '6', 'B'},

{'7', '8', '9', 'C'},

{'\*', '0', '#', 'D'}

};

byte rowPins [ROWS] = {8, 7, 6, 5}; //pins of the keypad

byte colPins [COLS] = {4, 3, 2, 1};

Keypad myKeypad = Keypad( makeKeymap(hexaKeys), rowPins, colPins, ROWS, COLS);

void setup()

{

Serial.begin(9600);

Serial.begin(115200);

Wire.begin(19,18);

lcd.init();

lcd.backlight();

// codeLocker1[Password\_Lenght] = "2222";

// codeLocker2[Password\_Lenght] = "2222";

// codeLocker3[Password\_Lenght] = "3333";

// codeLocker4[Password\_Lenght] = "4444";

}

void loop()

// Serial.println("initilize");

// Serial.println(codeLocker1[Password\_Lenght]);

// char codeLocker1[Password\_Lenght]= "1234";

// codeLocker1 = "123456";

{

lcd.setCursor(0, 0);

lcd.print("Enter Password");

customKey = myKeypad.getKey();

if (customKey) // makes sure a key is actually pressed, equal to (customKey != NO\_KEY)

{

Data[data\_count] = customKey; // store char into data array

lcd.setCursor(data\_count, 1); // move cursor to show each new char

lcd.print(Data[data\_count]); // print char at said cursor

data\_count++; // increment data array by 1 to store new char, also keep track of the number of chars entered

}

if (data\_count == Password\_Lenght - 1) // if the array index is equal to the number of expected chars, compare data to master

{

lcd.clear();

lcd.setCursor(1, 0);

lcd.print("Pass is");

if (!strcmp(Data, codeLocker1)) // equal to (strcmp(Data, Master) == 0)

{

lcd.setCursor(0, 1);

lcd.print("CORRECT-Locker1");

openLocker1();

}

else if (!strcmp(Data, codeLocker2)){

lcd.setCursor(0, 1);

lcd.print("CORRECT-Locker2");

openLocker2();

}

else if (!strcmp(Data, codeLocker3)){

lcd.setCursor(0, 1);

lcd.print("CORRECT-Locker3");

openLocker3();

}

else if (!strcmp(Data, codeLocker4)){

lcd.setCursor(0, 1);

lcd.print("CORRECT-Locker4");

openLocker4();

}

else {

lcd.setCursor(0, 1);

lcd.print("INCORRECT");

wrongCode();

}

delay(1000);// added 1 second delay to make sure the password is completely shown on screen before it gets cleared.

lcd.clear();

clearData();

}

}

void clearData()

{

while (data\_count != 0)

{ // This can be used for any array size,

Data[data\_count--] = 0; //clear array for new data

}

return;

}

void wrongCode()

{

Serial.println("lock close");

delay(2000);

}

void openLocker1()

{

Serial.println("locker1 open");

char changeLocker1[Password\_Lenght] = "1234";

Serial.print("change codeLocker1 to: ");

Serial.println(changeLocker1);

delay(2000);

}

void openLocker2()

{

Serial.println("locker2 open");

delay(2000);

}

void openLocker3()

{

Serial.println("locker3 open");

delay(2000);

}

void openLocker4()

{

Serial.println("locker4 open");

delay(2000);

}

//THANKS FOR DOWNLOADING .......

apply the random instead of “1234”

void assign(char \*pw){

char changeCode[5] = "1234";

for ( int i =0; i<4; i++){

pw[i] = changeCode[i];

}

}

apply random password

char C[3];

void I2C(int);

void assign(char \*pw){

int randNumber;

char changeCode[5];

randNumber = random(1000);

I2C(randNumber);

changeCode[1] = C[0];

changeCode[2] = C[1];

changeCode[3] = C[2];

for ( int i =1; i<4; i++){

pw[i] = changeCode[i];

}

}

void I2C(int Number){

int Num0 = Number%10;

int Num1 = (Number/10)%10;

int Num2 = Number/100;

Serial.println(Num0);

Serial.println(Num1);

Serial.println(Num2);

C[2] = Num0+48;

C[1] = Num1+48;

C[0] = Num2+48;

Serial.println(C);

}