



DesIRE



INTEGRATION OF IOT TECHNOLOGIES TO IT PROFESSIONALS EDUCATIONAL PROCESS

**Anzhelika Parkhomenko, Andriy Parkhomenko,
Yaroslav Zalyubovskiy, Marina Kalinina**

**Zaporizhzhya National Technical University,
Ukraine**





Outline

- Motivation for IoT technologies study
- Concept of the SMART LIFE project:
 - Main Stages;
 - Systems Architecture;
 - Subsystems realization and integration based on OpenHAB.
- Conclusions / Work in progress
- Questions





Era of Smart Technologies

CLOUD COMPUTING

ОБЛАЧНЫЕ ТЕХНОЛОГИИ

ОБЛАЧНЫЕ ТЕХНОЛОГИИ

VIRTUAL & AUGMENTED REALITY

ВИРТУАЛЬНАЯ И ДОПОЛНЕННАЯ РЕАЛЬНОСТЬ

INTERNET OF THINGS & SMART HOUSE

УМНЫЙ ДОМ И ИНТЕРНЕТ ВЕЩЕЙ

3D PRINT

3D ПЕЧАТЬ

MULTICOPTERS, ROBOTS, TRANSPORTATION

МУЛЬТИКОПТЕРЫ, РОБОТЫ, ТРАНСПОРТ

WEARABLE ELECTRONICS & NEURAL INTERFACES

НОСИМАЯ ЭЛЕКТРОНИКА И НЕЙРОИНТЕРФЕЙСЫ



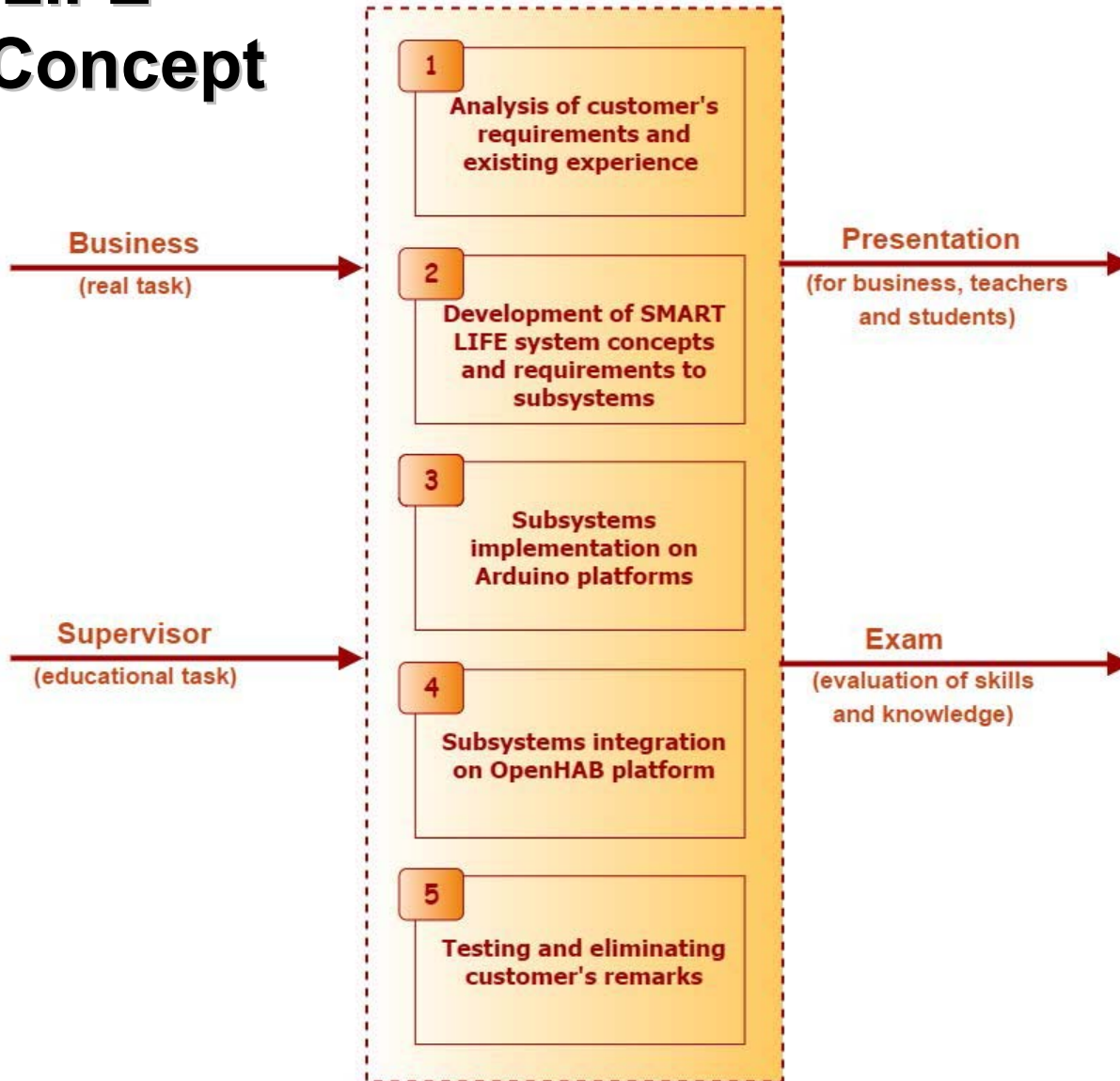


Internet of Things & Smart House



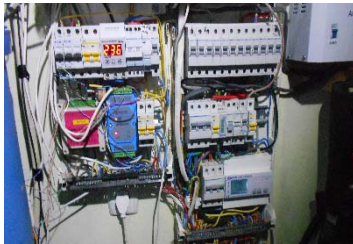


SMART-LIFE Project Concept



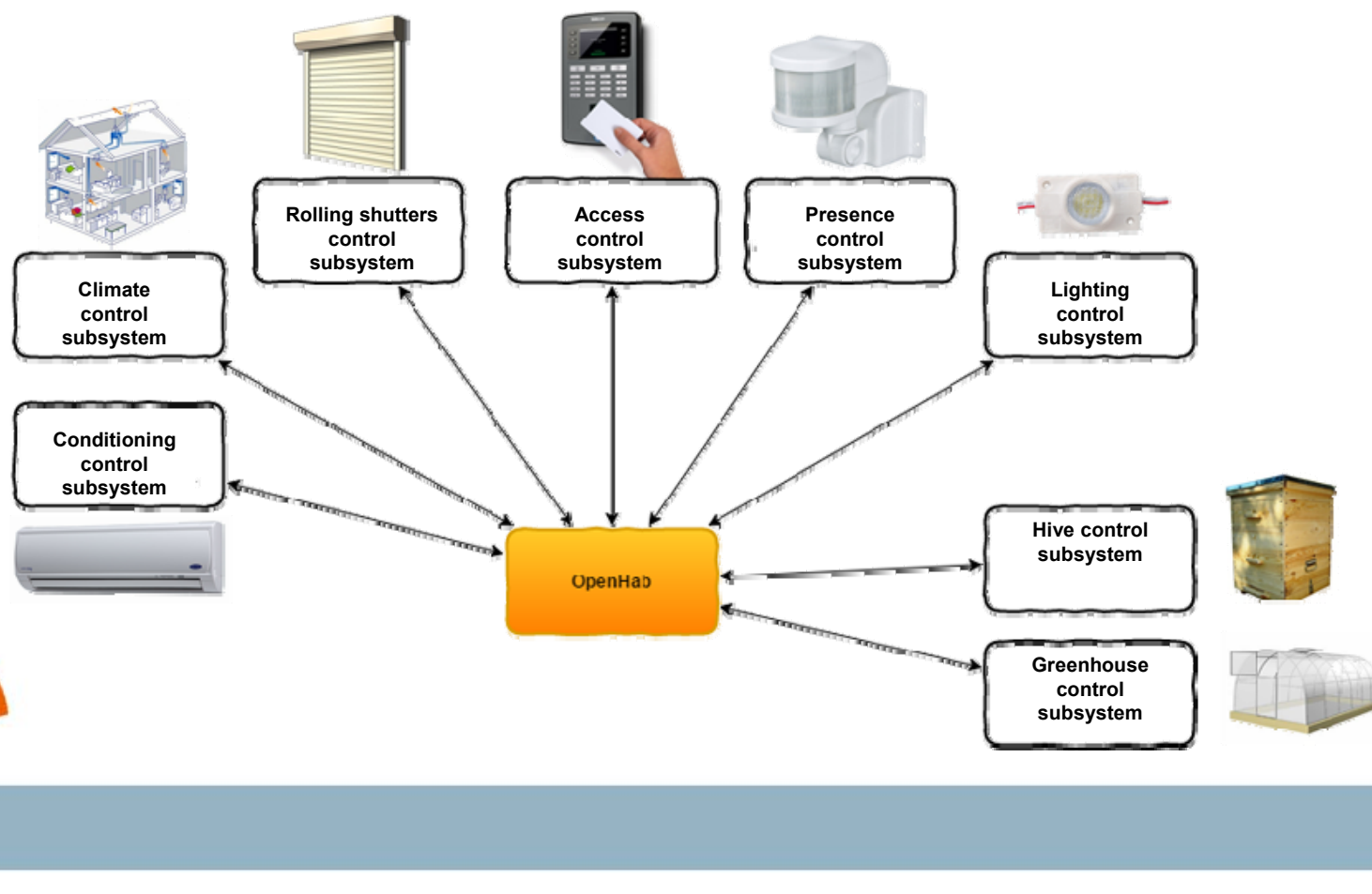


Smart House – real object for students training





SMART-LIFE system architecture





Rolling shutters control subsystem

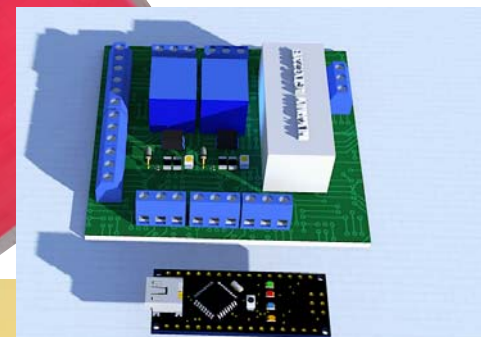
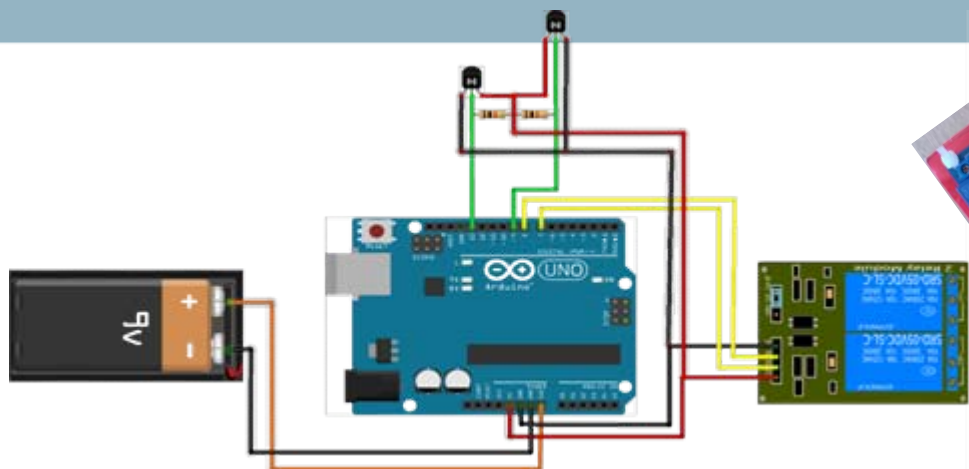


Artem Tulenkov, Master student





Design decisions



fritzing

```
relay_test | Arduino 1.6.9 Hourly Build 2016/03/17 01:33
Файл Процесс Схемы Инструменты Помощь

relay_test

int RelayOpen = 7;
int RelayClose = 4;
int Sens1=0;
char val;
int stay=0;

void setup()
{
  Serial.begin(9600);
  pinMode(RelayOpen, OUTPUT);
  pinMode(RelayClose, OUTPUT);
  digitalWrite(RelayOpen, HIGH);
  digitalWrite(RelayClose, HIGH);
  pinMode(Sens1, INPUT);
}

void loop()
{
  stay=0;
  if(Serial.available() {
    val = Serial.read();
    if(val == 'O'){
      digitalWrite(RelayOpen, LOW);
    }
    if(val == 'M'){
      digitalWrite(RelayOpen, HIGH);
    }
  }
}
```





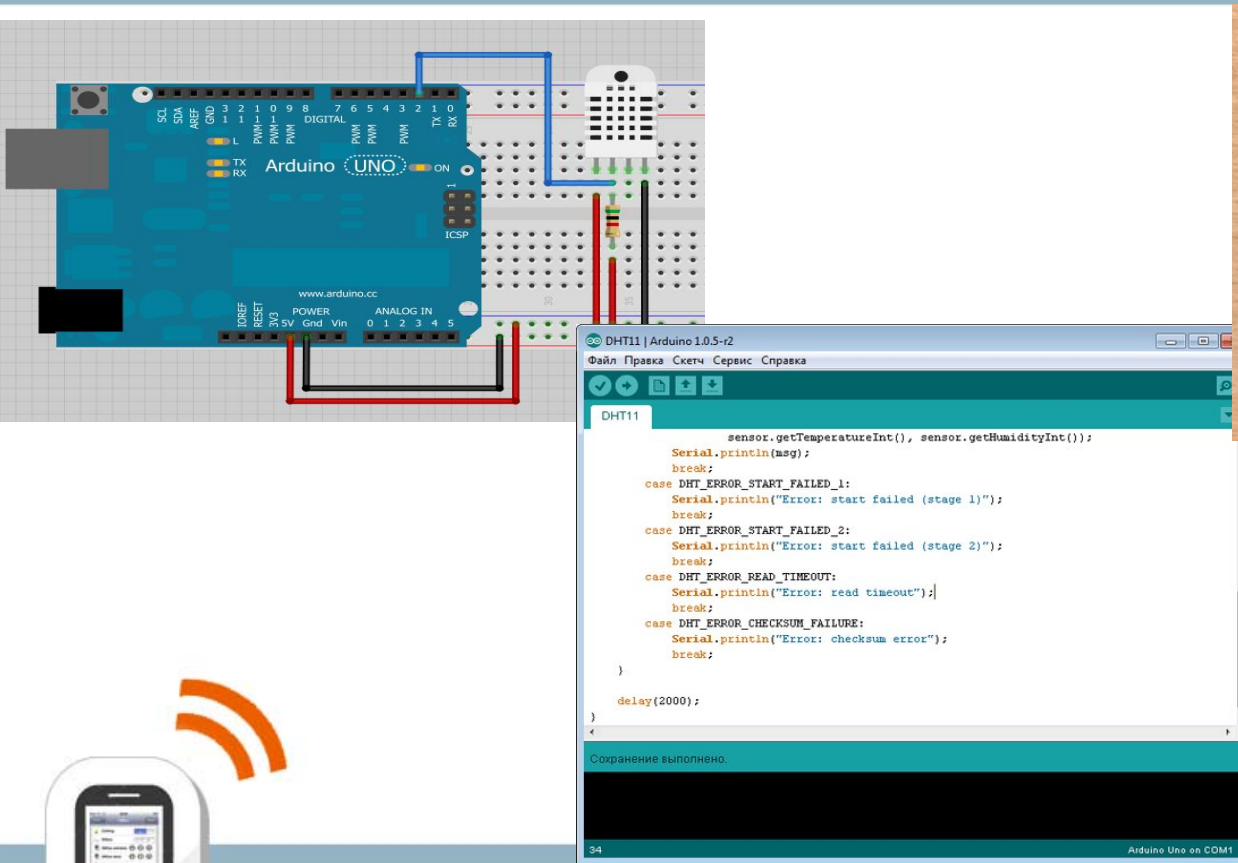
Climate control subsystem

Maxim Reznichenko, Master student





Design decisions



The image shows a top-down view of an Arduino Uno microcontroller board connected to a breadboard. A DHT11 temperature and humidity sensor is mounted on the breadboard. Wires connect the sensor's pins to the Arduino's digital and power pins. Below the breadboard, a screenshot of the Arduino IDE code editor is visible. The code is for a DHT11 sensor and includes error handling for various failure modes.

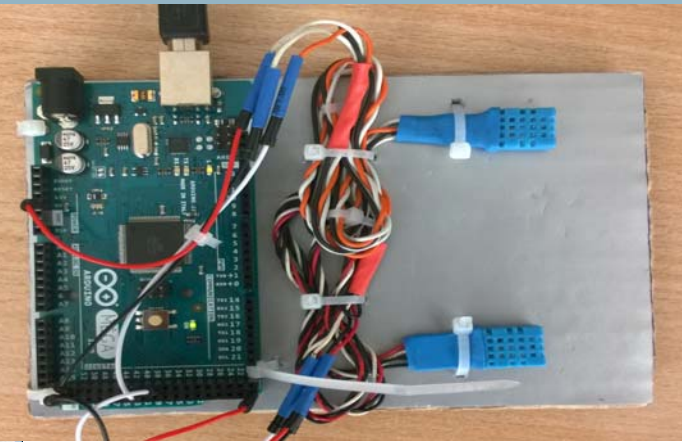
```
DHT11

    sensor.getTemperatureInt(), sensor.getHumidityInt());
    Serial.println(msg);
    break;
case DHT_ERROR_START_FAILED_1:
    Serial.println("Error: start failed (stage 1)");
    break;
case DHT_ERROR_START_FAILED_2:
    Serial.println("Error: start failed (stage 2)");
    break;
case DHT_ERROR_READ_TIMEOUT:
    Serial.println("Error: read timeout");
    break;
case DHT_ERROR_CHECKSUM_FAILURE:
    Serial.println("Error: checksum error");
    break;
}

delay(2000);
}

Сохранение выполнено.

34 Arduino Uno on COM1
```





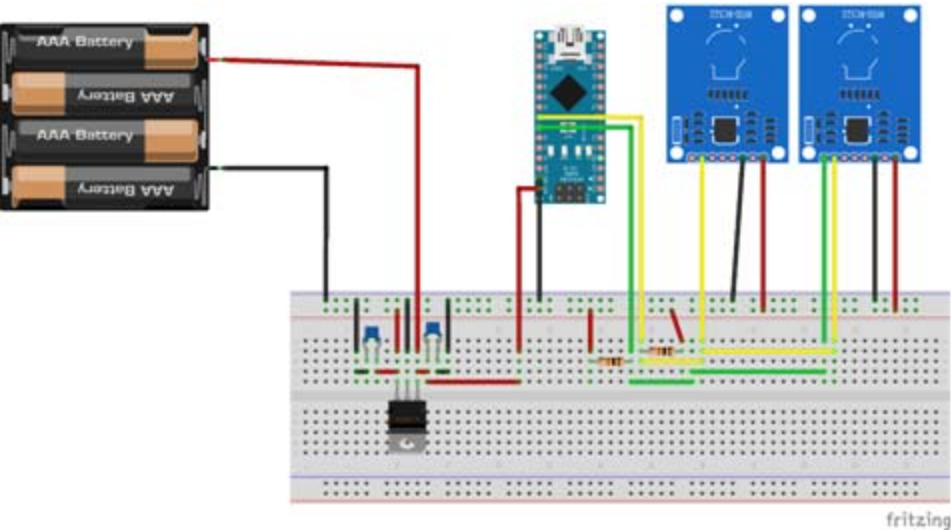
Access control subsystem

Dmytro Kravchenko, Olexiy Kravchenko, Bachelor students





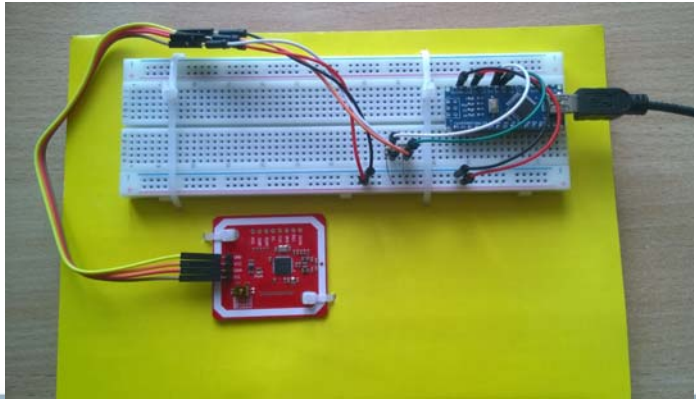
Design decisions



```
readRFID/Arduino 1.6.7
GnuPlot: GnuPlot - File Open - Help
readRFID
// Serial.println(overmodecard) + "\nRF, UID");
// configure board to read RFID tags
nfc.NCConfigure();
// Serial.println("Waiting for an ISO14443A Card ....");
}

void loopFunction ()
uint8_t success;
uint8_t uid[] = { 0, 0, 0, 0, 0, 0 }; // Buffer to store the returned UID
uint8_t uidLength;
// Length of the UID (4 or 7 bytes depending on ISO14443A card type)
// Wait for an ISO14443A type cards (Mifare, etc.). When one is found
// "uid" will be populated with the UID, and uidLength will indicate
// if the uid is 4 bytes (Mifare Classic) or 7 bytes (Mifare Ultralight)
success = nfc.readPassiveTargetID(PN532_RFID_TAG_ISO14443A, uid, uidLength);

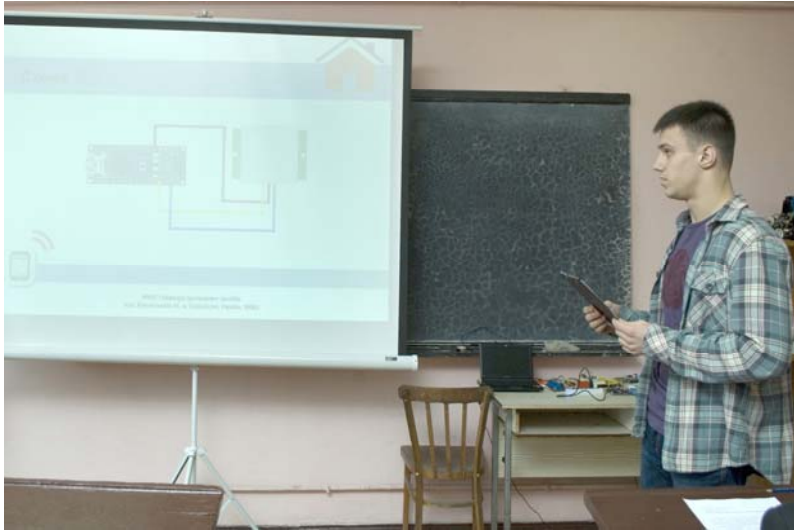
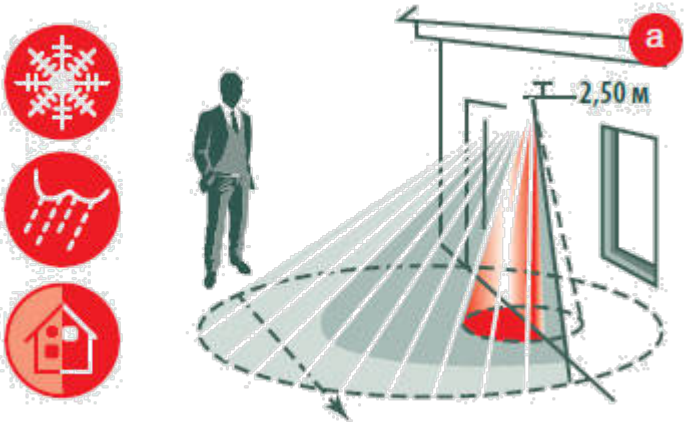
if (success) {
// Display some basic information about the card
// Serial.println("Found an ISO14443A card!");
// Serial.println(" UID Length: ");Serial.print(uidLength, DEC);Serial.println(" bytes");
// Serial.println(" UID Value: ");
nfc.PrintHex(uid, uidLength);
Serial.println("");
Serial.flush();
delay(1000);
}
```





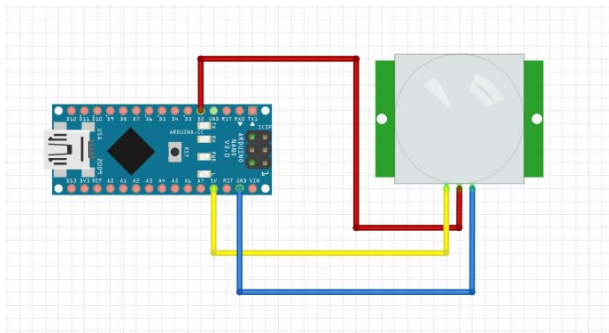
Presence control subsystem

Evgen Grigoriev, Master student



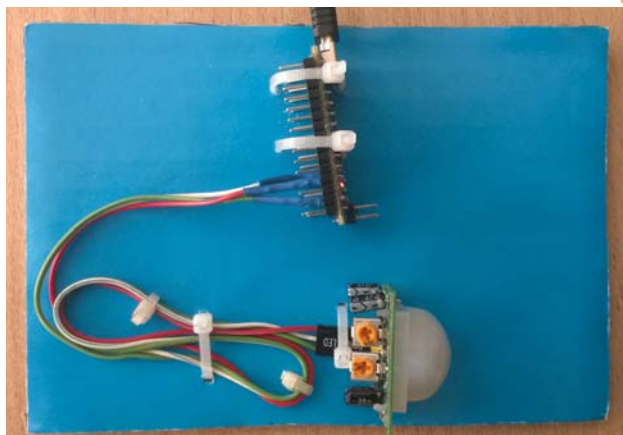


Design decisions



```
PBR | Arduino 1.6.0
Файл Правка Сервис Инструменты Панельца
PBR
//Тестируем на Arduino IDE 1.6.1
//Время калибровки датчика (10-60 сек. по датешту)
int calibrationTime = 30;
//Время, в которое был принят сигнал отсутствия датчика(100)
long waitTime int lowIn;
//Время, после которой датчик считается оповещен
long waitTime int pulse = 5000;
//Max. false - значит датчик уже обнаружен, true - уже известно, что датчик нет
boolean lockLow = true;
//Max. Communication - необходимость изменить время начала отсутствия датчика
boolean takeLowTime;
int pinIn = 2; //Канал подключения PIR датчика
int ledPin = 13; //Канал оповещения датча

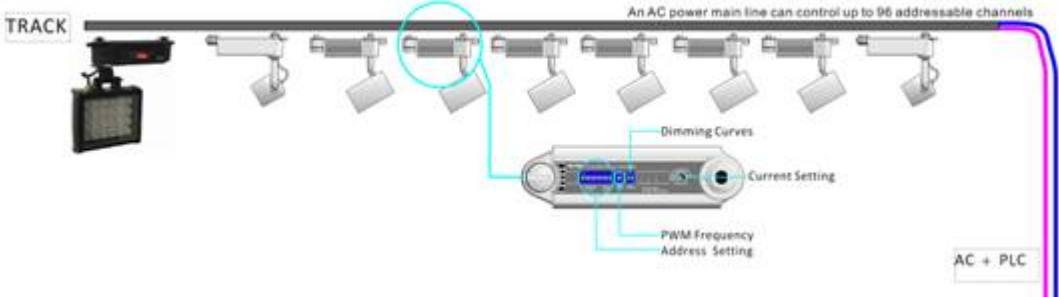
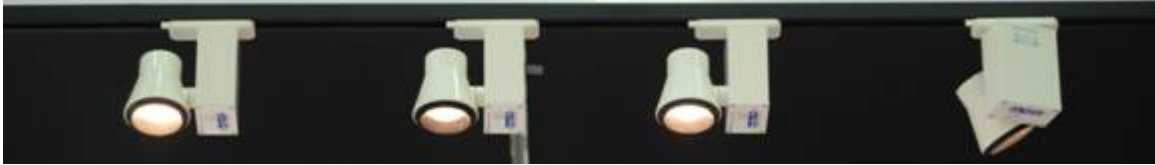
void setup()
{
  Serial.begin(9600);
  pinMode(pinIn, INPUT);
  pinMode(ledPin, OUTPUT);
  digitalWrite(pinIn, LOW);
  //Дать датчику время на инициализацию
  Serial.println("calibration");
  for(int i = 0; i < calibrationTime; i++)
  {
    //Serial.println("");
    digitalWrite(ledPin, HIGH);
    delay(1000);
  }
}
```





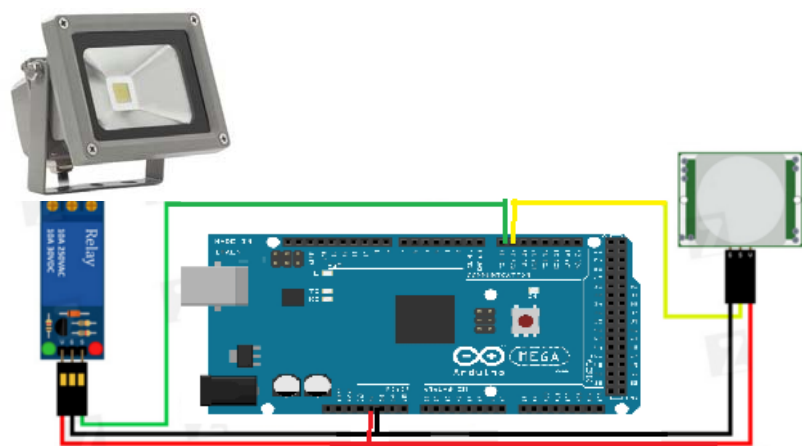
Lighting control subsystem

Olexiy Gaman, Bachelor student





Design decisions



```
light | Arduino 1.6.8
Файл Правка Скетч Инструменты Помощь

light

int ldr = 0; // змінна з номером піна до якого підключено датчик освітлення
int Led = 6; // змінна з номером піна до якого підключено світлодіод
void setup() {
  pinMode(Led, OUTPUT); //встановлюємо світлодіод як вихід
  Serial.begin(9600);
}
void loop() {
  if (analogRead(ldr) > 503) // перевірка порогового значення освітлення
    digitalWrite(Led, LOW); // якщо показник освітлення >503, то вимкнути світло
  else
    digitalWrite(Led, HIGH); // якщо показник освітлення < 503, то увімкнути світло
}
```

Компіляція завершена

Скетч використовує 2 150 байт (6%) пам'яті пристрою. Всього доступно 32 768 байт. Глобальні змінні використовують 188 байт (9%) динамічної пам'яті.

14 Arduino/Genuino Uno на COM1

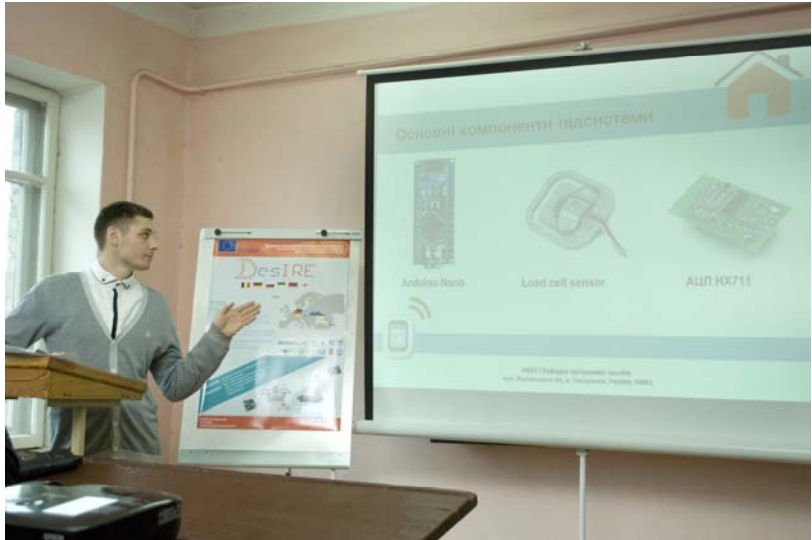




Hive control subsystem



Semen Misura, Master student





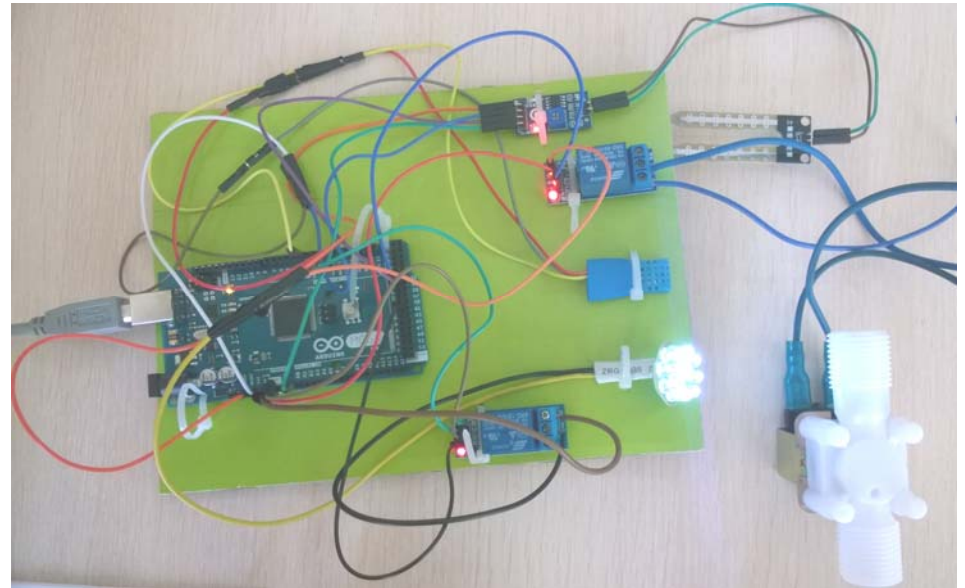
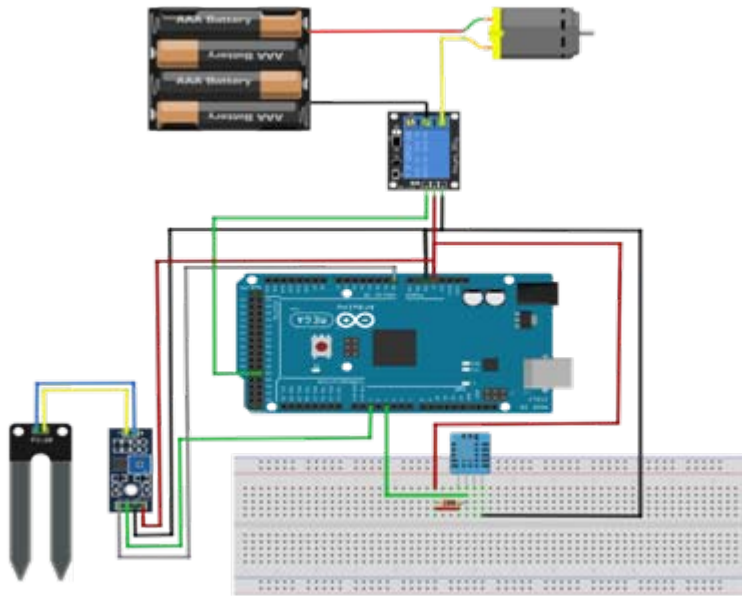
Greenhouse control subsystem

Mikita Shvidkiy, Bachelor student





Design decisions





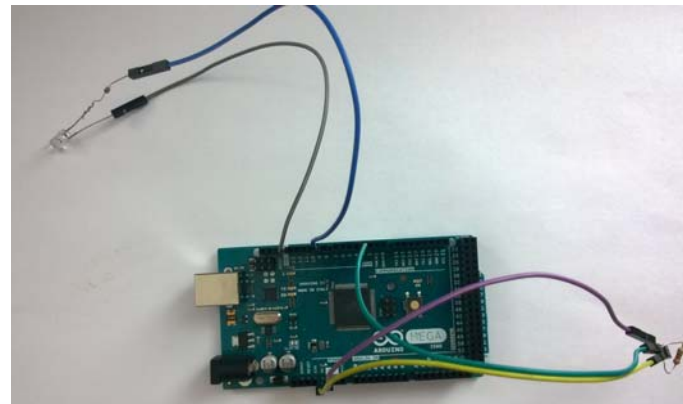
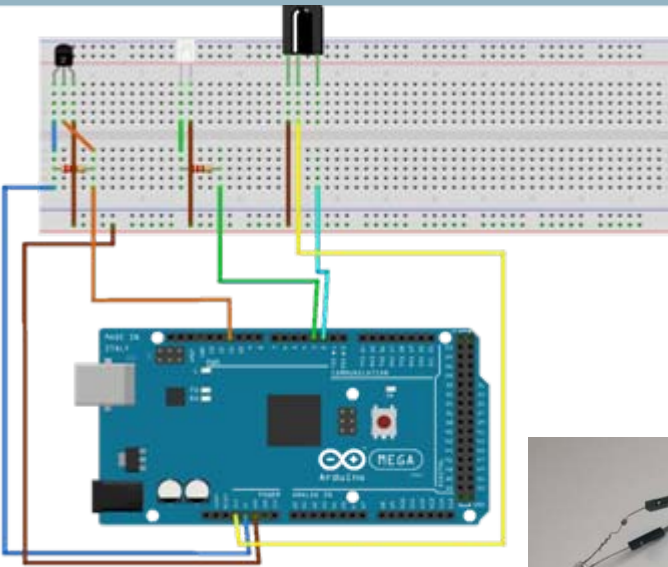
Conditioning control subsystem

Olexander Goloviznin, Specialist student





Design decisions



```
sketch Apr06 g
// Arduino Duemilanove
#define IRpin_PIN D10
#define IRpin 2
#endif

#define MAXPULSE 45000
#define RESOLUTION 20 // Timing resolution

int ButtonPin = 4;
int ButtonState = 0;
uint16_t pulse[200][2]; // 100 pairs at
uint16_t currentpulse = 0;
uint16_t sendpulse = 0;

void setup(void) {
  pinMode(ButtonPin, INPUT);
  digitalWrite(ButtonPin, HIGH);
  Serial.begin(9600);
  Serial.println("Infrared Receive Key...");
}

void loop(void) {
  ButtonState = digitalRead(ButtonPin);
  if (ButtonState==LOW) { // currentpulse != 0)
    if (ButtonState==LOW)
    {
      Serial.println("InReady to decode IR");
      delay(400);
      uint16_t highpulse, lowpulse;
      highpulse = lowpulse = 0;
      while (IRpin_PIN & (1 << IRpin))
      {
        highpulse++;
        delayMicroseconds(RESOLUTION);
        if ((highpulse == MAXPULSE) && (currentpulse != 0))
        {
          printpulse();
          sendpulse=currentpulse;
          currentpulse=0;
          return;
        }
      }
      pulse[currentpulse][0] = highpulse;
      while (! (IRpin_PIN & _BV(IRpin)))
      {
        // pin is still LOW
        lowpulse++;
        delayMicroseconds(RESOLUTION);
        if ((lowpulse == MAXPULSE) && (currentpulse != 0))
        {
          printpulse();
          sendpulse=currentpulse;
          currentpulse=0;
          return;
        }
      }
      pulse[currentpulse][1] = lowpulse;
      currentpulse++;
    }
  }
}

void printpulse() {
  Serial.println("Pulse:");
  for (int i = 0; i < 200; i++)
  {
    Serial.print(pulse[i][0]);
    Serial.print(" ");
    Serial.print(pulse[i][1]);
    Serial.println();
  }
}

void sendpulse() {
  Serial.println("Send pulse:");
  for (int i = 0; i < 200; i++)
  {
    Serial.print(sendpulse[i][0]);
    Serial.print(" ");
    Serial.print(sendpulse[i][1]);
    Serial.println();
  }
}
```





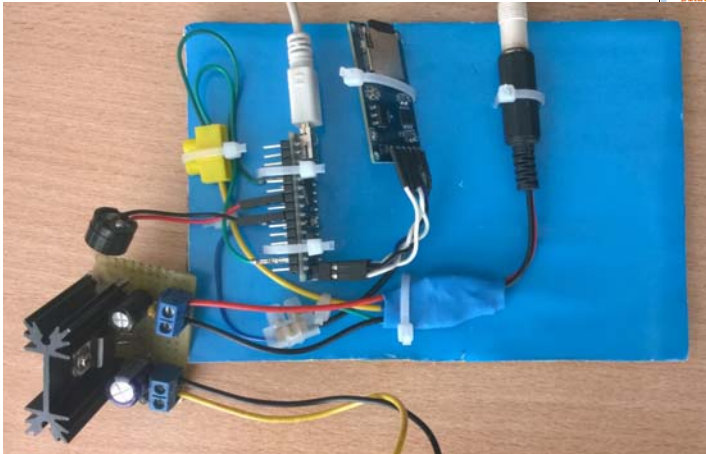
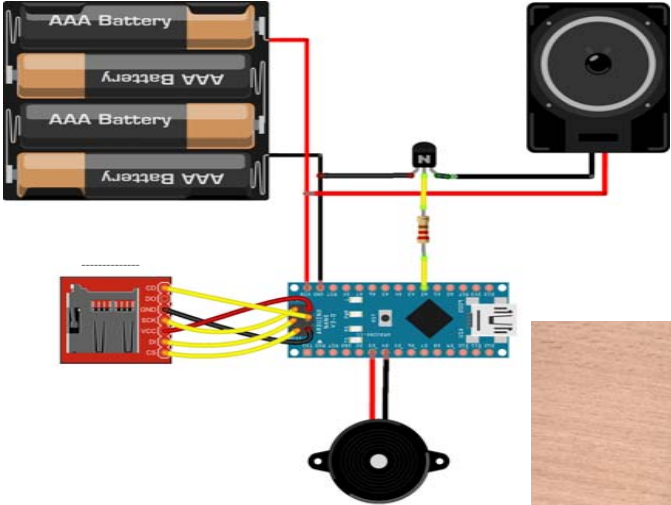
Sound notification subsystem

Artem Tulenkov, Master student





Design decisions



```
Wavino | Arduino 1.6.9 Hourly Build 2016/03/17 01:33
Файл Правка Скетч Инструменты Помощь

Wavino $ TMRpcm.h

#include <TMRpcm.h>
#include <SPI.h>
#include <SD.h>
#define SD_ChipSelectPin 4
#include <TMRpcm.h>

TMRpcm tmrpcm;
int SW1;
int SW2;
int SW3;
int SW4;

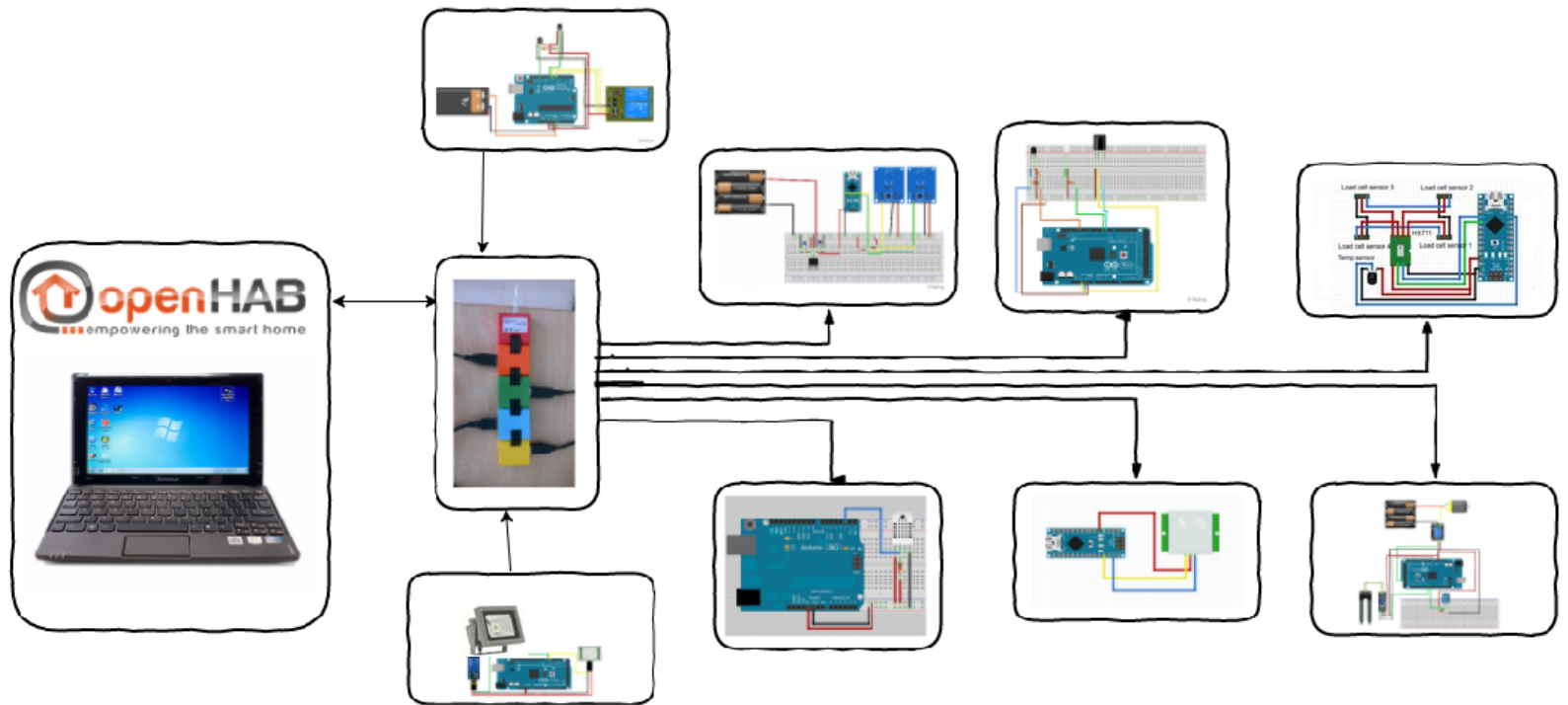
void setup() {
  pinMode(14, INPUT);
  pinMode(15, INPUT);
  pinMode(16, INPUT);
  pinMode(17, INPUT);

  tmrpcm.speakerPin = 9;
}
```



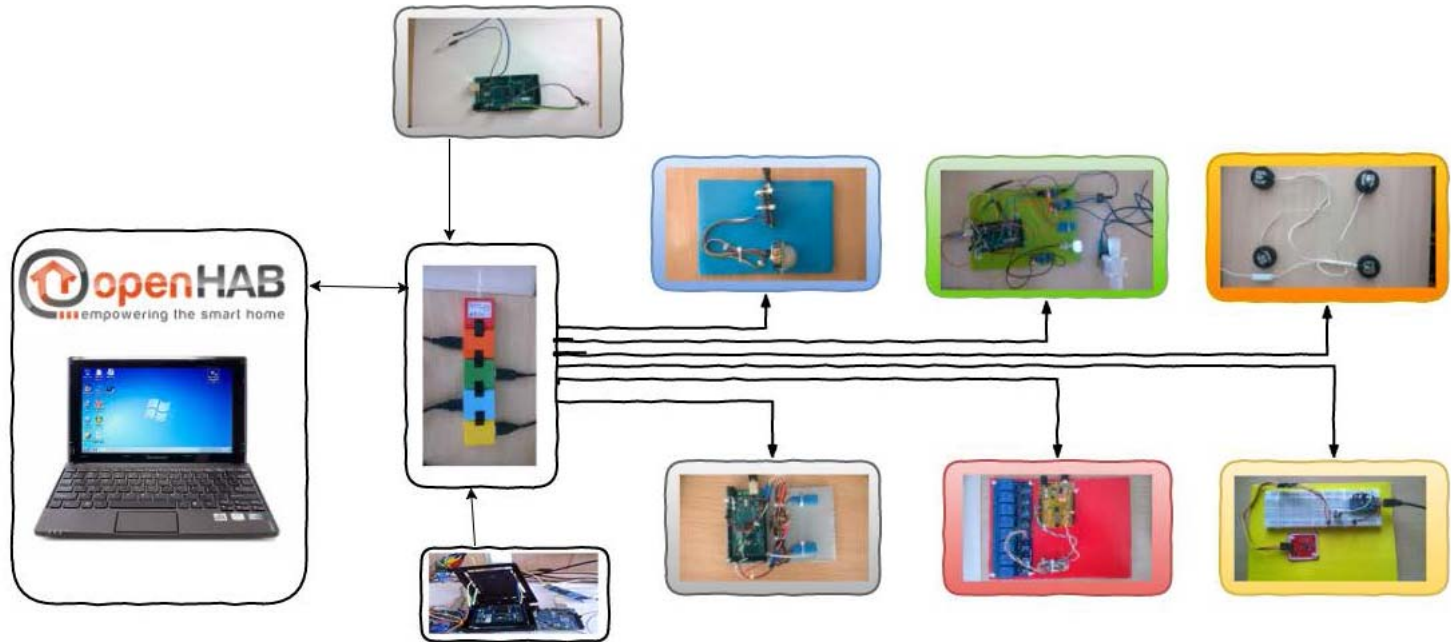


Subsystems integration based on OpenHab (Open Source Home Automation Bus)





Systems prototype testing





OpenHab Interface

The screenshot displays the OpenHab web interface in a browser window. The address bar shows the URL: localhost:8080/openhab.app?sitemap=demo#_Home. The interface is divided into several sections:

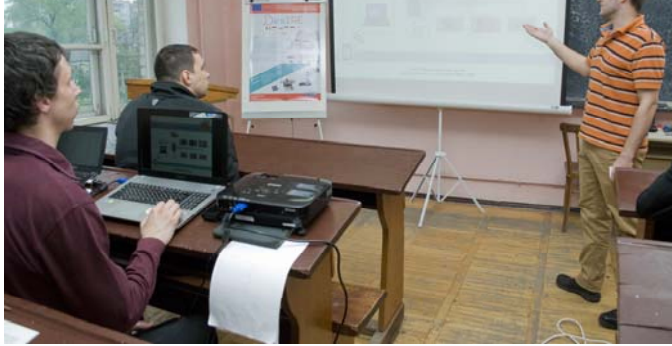
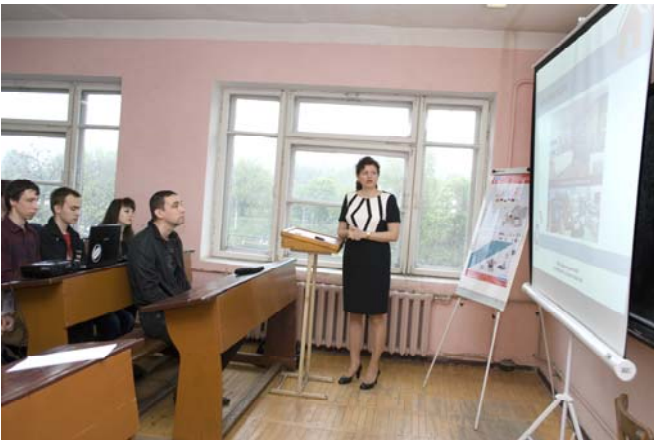
- Статус Дома (Home Status):** A vertical list of controls including: Присутствие (Presence), Освещение (Lighting), Дверь (Door), Роллеты (Roller shutters), and Кондиционер (Air conditioner).
- Статус Теплицы (Greenhouse Status):** A vertical list of controls including: Полив в теплице (Greenhouse watering), Обогрев в теплице (Greenhouse heating), and Вентиляция в теплице (Greenhouse ventilation).
- Улица (Street):** A section for outdoor weather data with two rows: Температура (Temperature) and Влажность (Humidity), both showing a dash (-).
- Улей (Beehive):** A section for beehive data with two rows: Вес улья (Beehive weight) showing a dash (-), and Температура в улье (Beehive temperature) showing a dash (-).
- Теплица (Greenhouse):** A section for greenhouse data with three rows: Влажность почвы (Soil humidity), Влажность воздуха (Air humidity), and Температура воздуха (Air temperature), all showing a dash (-).

At the bottom right of the interface, the copyright notice reads: ©2010-2015 openHAB.org.





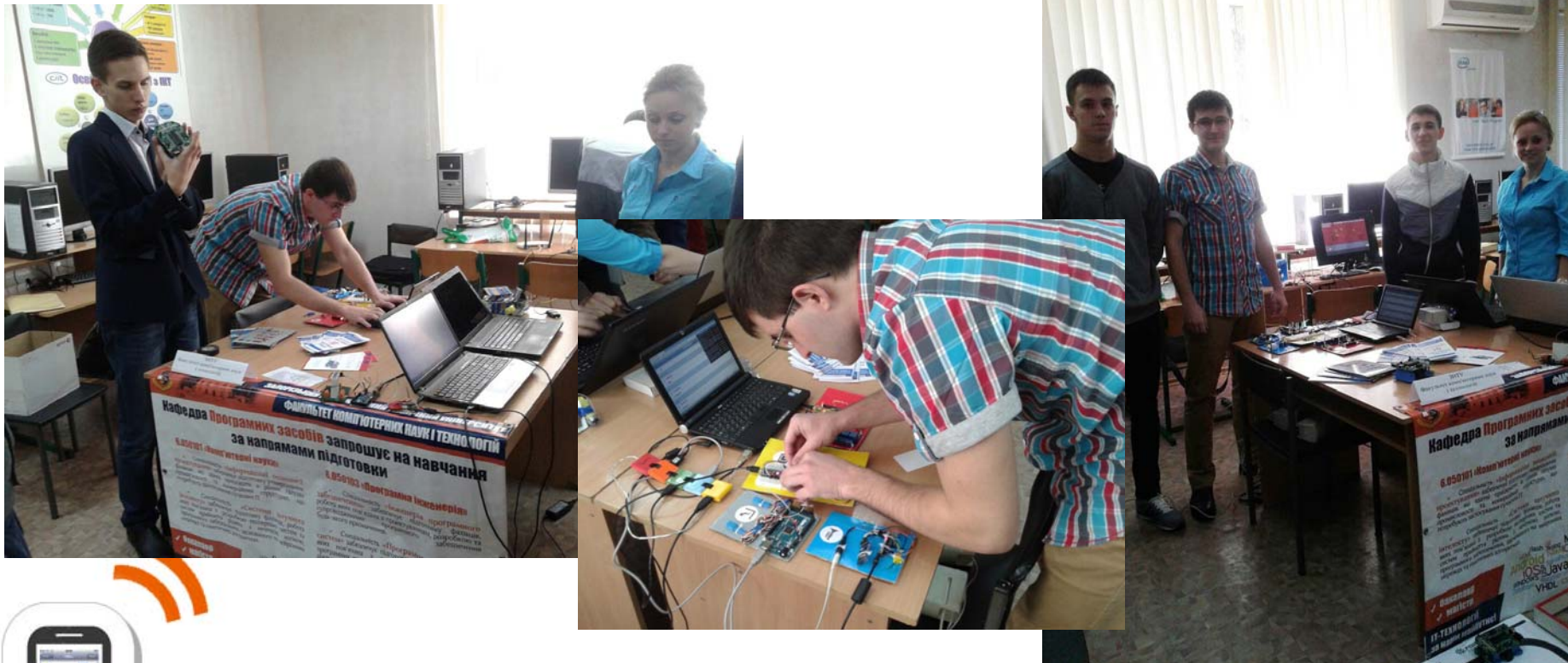
SMART LIFE project team at the conference “ZNTU Science Week» April, 2016



<http://www.zntu.edu.ua/integraciya-tehnologiy-internet-things-v-proces-pidgotovky-it-fahivciv>



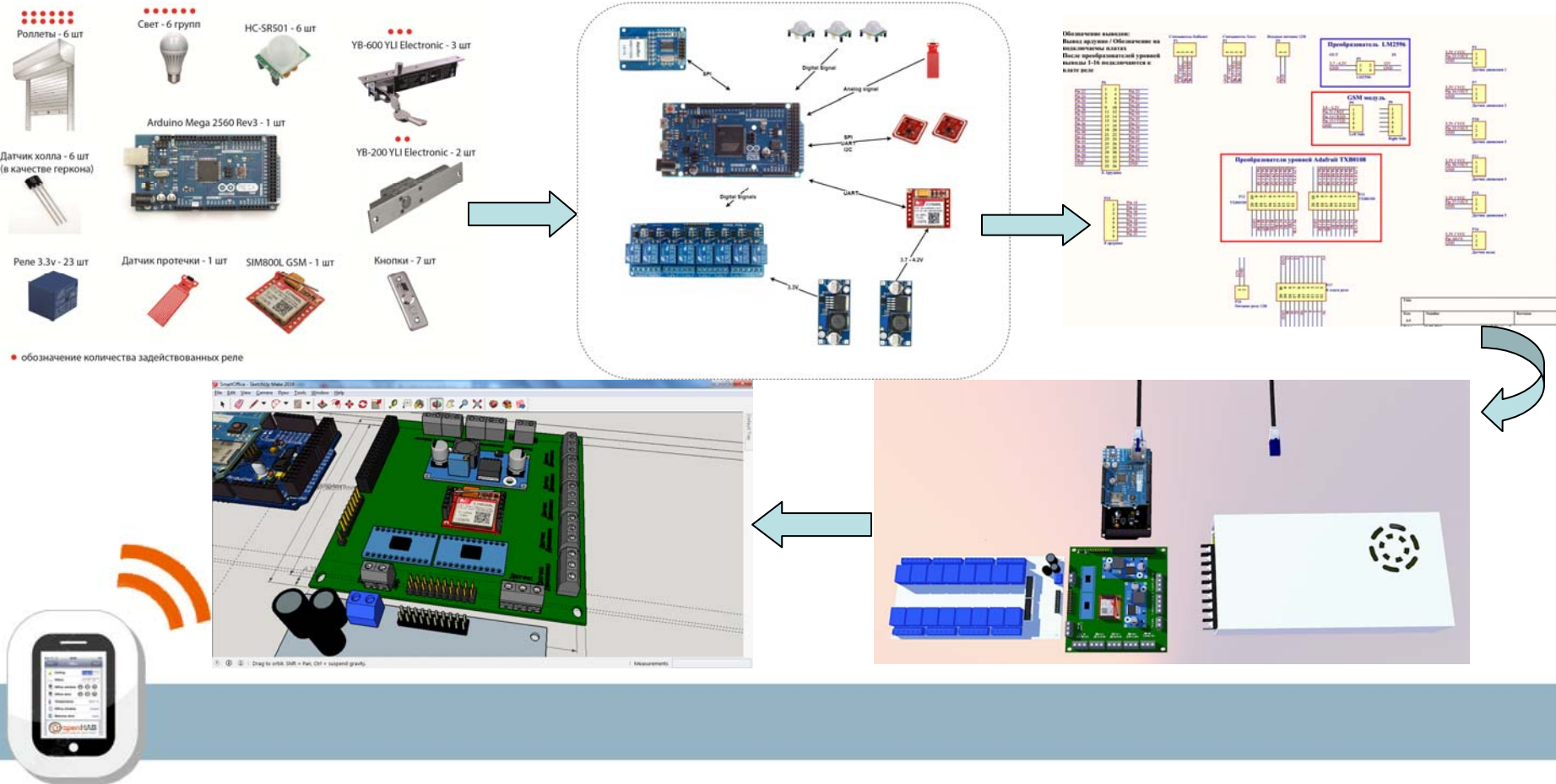
SMART LIFE project team at the exhibition “STEM-education of Zaporizhzhya region – 2016»



<http://www.zntu.edu.ua/stem-osvita-zaporizkogo-krayu-2016>

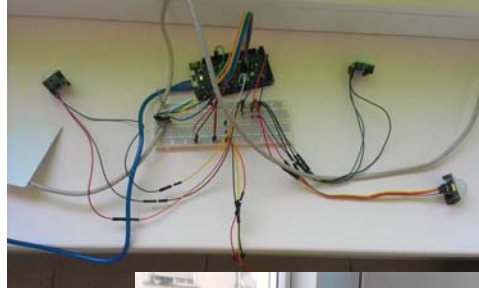


Work in progress: Smart Office for VANDERSLAB Company



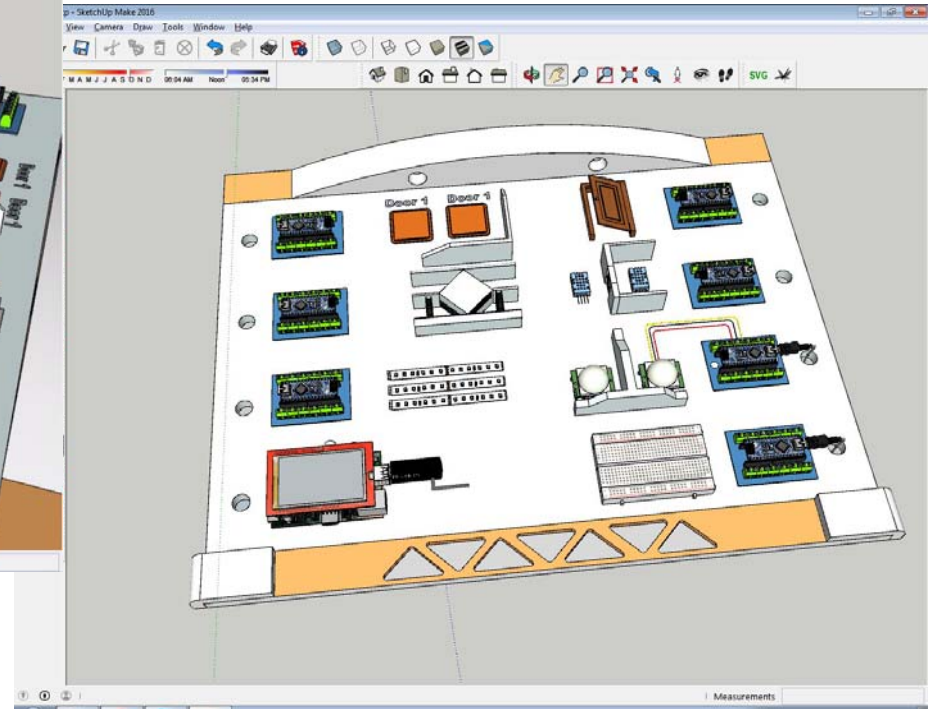
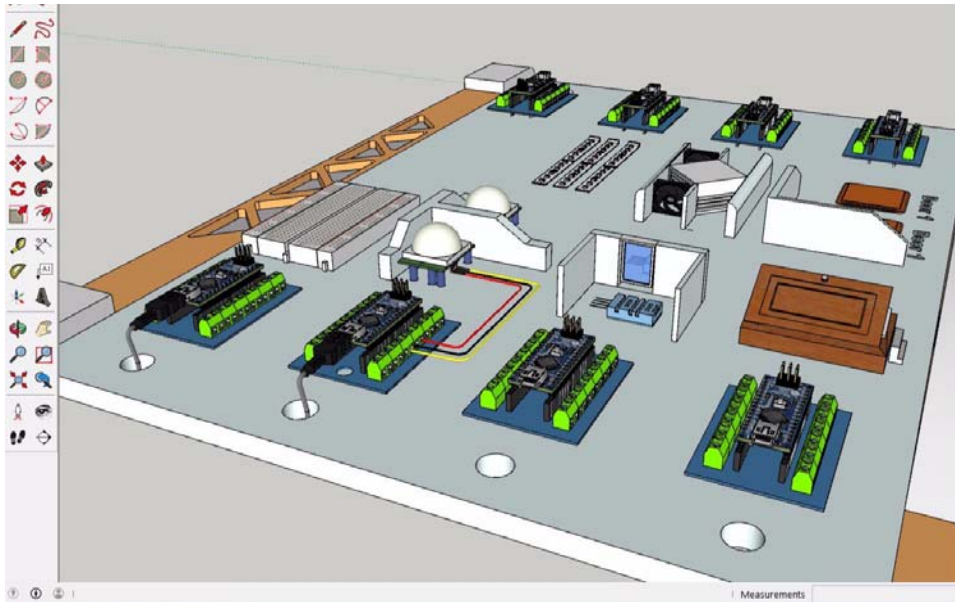


Work in progress: Smart Office for VANDERSLAB Company





Work in progress : Remote Laboratory Smart House



Conclusions



Implementation of real projects gives students invaluable practical experience and knowledge, motivation to research, to work in team, to communicate with the customer, to present the results of their work to an audience.





THANK YOU FOR YOUR ATTENTION!

