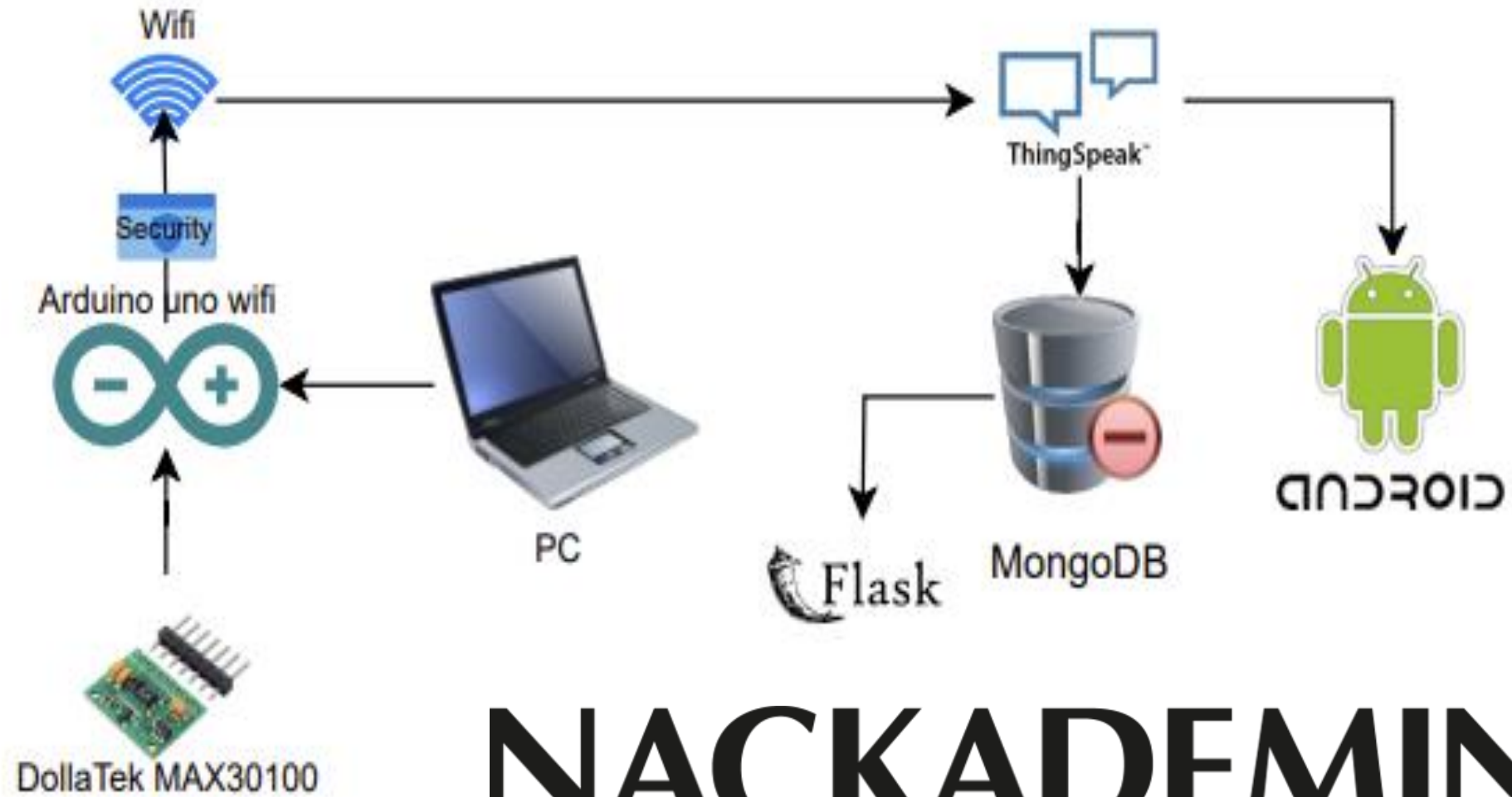


# Examen arbete

Elever: Jekaterina Jelisejeva, George Glor

Koppla dollatek max30100 med Arduino Uno R4 Wifi , ThingSpeak, MongoDB, Flask med Android App.



# NACKADEMIN



# Trello Översikt: Projektstyrning och Arbetsflöde

The screenshot displays a Trello board interface for a project titled "Examen Arbete". The board is organized into four columns representing different stages of the workflow: "To do", "On progress", "Done", and "Test". Each column has a header with a three-dot menu and a "Add a card" button at the bottom. The "Done" column contains five cards: "Projektplan", "Problemformulering", "Skriva arduino kod", "ThingSpeak", and "Flask". The "Test" column contains two cards: "Android APP" and "MongoDB", followed by a "Add a card" button. The top navigation bar includes the board name, a star icon, a "Workspace visible" label, a "Board" dropdown menu, and links to "Power-Ups", "Automation", "Filters", a user profile icon, and a "Share" button.

**Examen Arbete** ☆ Workspace visible Board ▾ Power-Ups ⚡ Automation ≡ Filters GG 👤 Share ⋮

**To do** ...  
+ Add a card 📅

**On progress** ...  
+ Add a card 📅

**Done** ...

- Projektplan
- Problemformulering
- Skriva arduino kod
- ThingSpeak
- Flask

+ Add a card 📅

**Test** ...

- Android APP
- MongoDB

+ Add a card 📅

+ Add another list

## Bibliotek som är viktiga för att läsa av sensorn och koppla upp allt mot ThingSpeak.

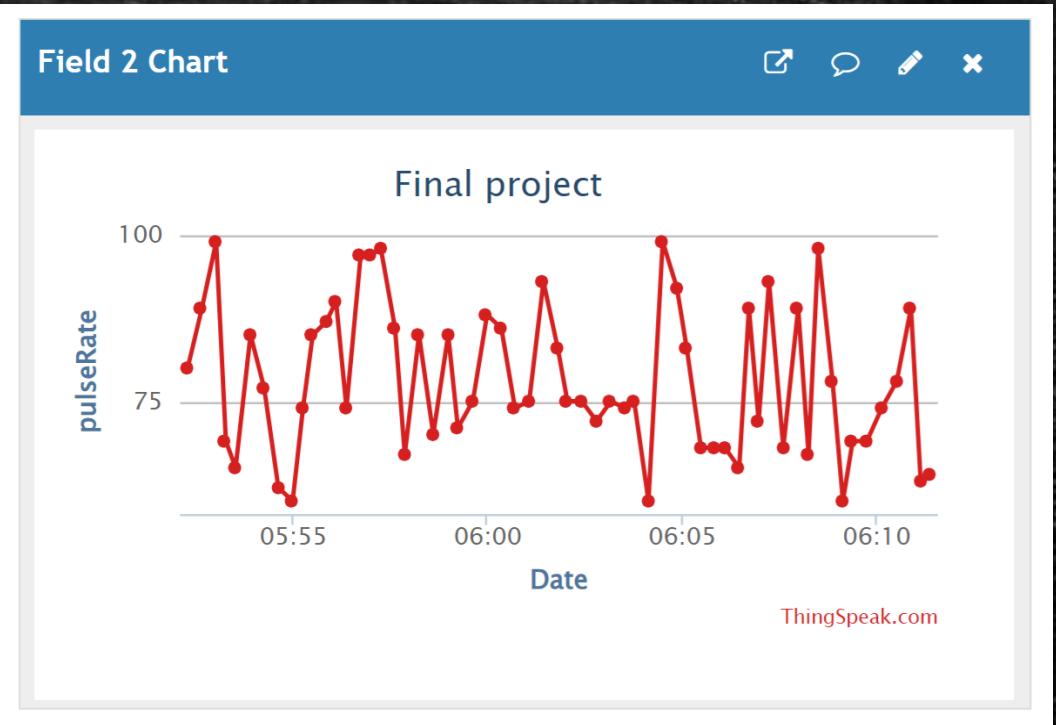
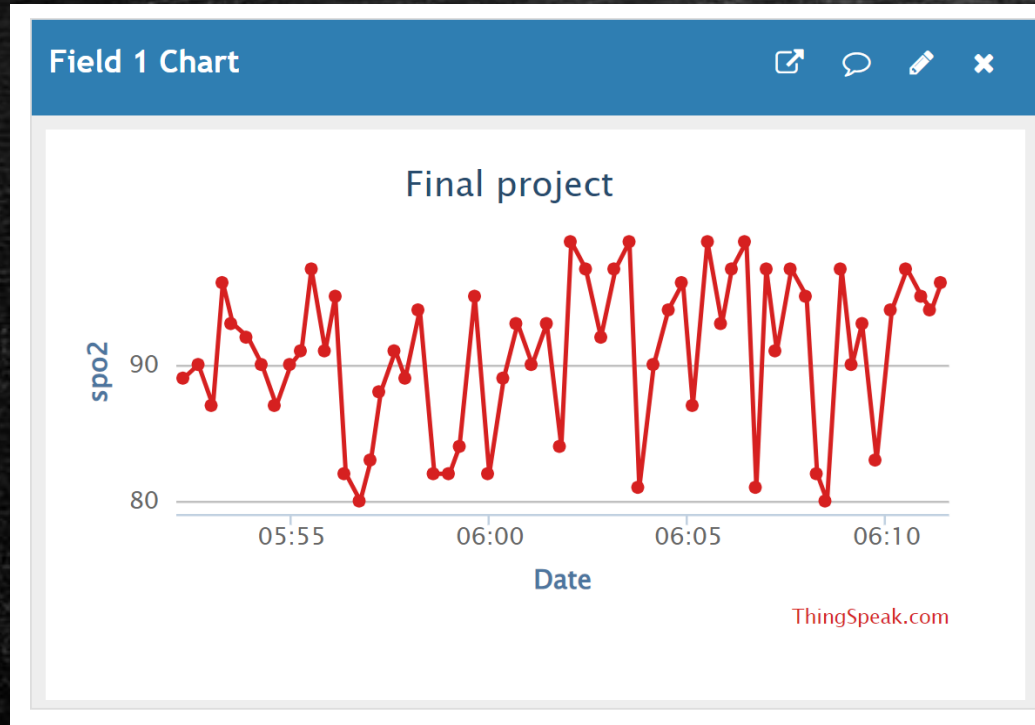
---

```
#include <Wire.h>
#include <MAX30100.h>
#include <WiFiS3.h>
#include "ThingSpeak.h"
#include
<MAX30100_PulseOximeter.h>
```

```
float spo2, pulseRate;
readSensorData(spo2, pulseRate);
Serial.print("SpO2 = ");
Serial.print(spo2);
Serial.println("%");
Serial.print("Pulse Rate = ");
Serial.print(pulseRate);
Serial.println("BPM");
```



# Resultaten i ThingSpeak



# Visualisering av Data med MongoDB: Utforska Representationer av Dina Dokument

Exjobb.HealthData

Documents Aggregations Schema Indexes Validation

Filter ⓘ ⓘ Type a query: { field: 'value' } or [Generate query](#) ↗

ADD DATA EXPORT DATA

```
{
  "_id": ObjectId('65800d7b94c01e128f9b040f'),
  "SpO2": 89,
  "PulseRate": 69,
  "timestamp": "2023-12-18T04:25:56.000+00:00"
}
```

```
{
  "_id": ObjectId('65800d7b94c01e128f9b0410'),
  "SpO2": 90,
  "PulseRate": 68,
  "timestamp": "2023-12-18T04:26:18.000+00:00"
}
```

```
{
  "_id": ObjectId('65800d7b94c01e128f9b0411'),
  "SpO2": 91,
  "PulseRate": 80,
  "timestamp": "2023-12-18T04:26:39.000+00:00"
}
```

```
{
  "_id": ObjectId('65800d7b94c01e128f9b0412'),
  "SpO2": 89,
  "PulseRate": 85,
  "timestamp": "2023-12-18T04:27:02.000+00:00"
}
```

```
{
  "_id": ObjectId('65800d7c94c01e128f9b0413'),
  "SpO2": 84,
  "PulseRate": 89,
  "timestamp": "2023-12-18T04:27:19.000+00:00"
}
```

- `import pymongo`
- `from pymongo import MongoClient`
- `import requests`
- `client = MongoClient(mongodb_uri)`
- `db = client.Exjobb`
- `collection = db.HealthData`



## Oxygen och Puls Statusbedömning: Dynamisk Hälsostatus baserad på Värden

```
String oxygenStatus = "Normal";  
if (oxygen < 90) {  
    oxygenStatus = "Low Risk";  
} else if (oxygen < 80) {  
    oxygenStatus = "High Risk";  
}
```

```
String pulseStatus = "Normal";  
if (pulse < 60 || pulse > 100) {  
    pulseStatus = "Not normal";  
}
```

## Visualisering som hämtas från ThingSpeak.

---


[View Data](#)

Time:

2024-01-11

SpO2 Level: 93

%C (Normal)

Pulse: 69   
(Normal)