

ESCUELA POLITÉCNICA SUPERIOR DE MONDRAGON UNIBERTSITATEA
MONDRAGON UNIBERTSITATEKO GOI ESKOLA POLITEKNIKOA
MONDRAGON UNIVERSITY FACULTY OF ENGINEERING

Partial release of the project

GRADO EN INGENIERÍA EN INFORMÁTICA
INFORMATIKAKO INGENIARITZA GRADUA
DEGREE IN COMPUTER ENGINEERING

Título del Trabajo *Lanaren izenburua* Project Topic

**RUNTIME VERIFICATION FOR SPATIO-TEMPORAL PROPERTIES (WITH
AGGREGATED OPERATORS)**

Autor *Egilea* Author

OIHANA GARCIA ANACABE

Curso *Ikasturtea* Year

2021/2022

Nombre y apellidos del autor

Egilearen izen-abizenak

Author's name and surnames

GARCIA ANACABE, OIHANA

Nombre y apellidos del/los director/es del trabajo

Zuzendariaren/zuzendarien izen-abizenak

Project director's name and surnames

EZIO BARTOCCI

ILLARRAMENDI, MIREN

Lugar donde se realiza el trabajo

Lana egin deneko lekua

Company where the project is being developed

TU WIEN

Curso académico

Ikasturtea

Academic year

2021/2022

Table of contents

1. Description.....	2
2. Competences	2
3. Objectives	3
4. Product specifications and requirements	3
4.1. Resources and materials	3
4.2. Tests and trials.....	5
4.3. Conditions for the implementation of the project	5
4.4. Legal aspects	5
5. Project information	6
5.1. Department	6
5.2. Tasks	6
5.2.1. Use case	7
5.2.2. Communication and monitor	8
6. Problems and solutions	8
7. Conclusions and evaluation of the tasks carried out.....	8
8. Future developments of the project.....	9
9. Contributions of the traineeships to studies.....	9
10. Bibliography	10

1. Description

IoT (Internet of Things) is the connection of millions of smart devices and sensors connected to the internet. This field is growing rapidly, it is estimated that by the end of 2022 there will be 42.56 billion connected devices. [1] These sensors and devices collect large amount of information, this project consists of an implementation of a service to monitor all this data at runtime.

The data will be collected both across space and time. In Cyber Physical Systems (CPS) physical systems are monitored and/or controlled by a computational core. Interacting with networks, they measure the evolution in time of physical quantities detected by sensors. [2] For this project, IoT sensors (Thingy52) and a monitor (MoonLight) are already provided. The resources will be studied and manipulated and, for the communication of these components, a middleware will be implemented.



Figure 1-A Project outline

This monitor will be capable of monitoring at runtime, analysing the runtime behaviour of the system to detect if it satisfies or violates certain properties. For the monitoring of spatio-temporal properties logic-based specification languages such as STREL will be used. STREL permits to specify the requirements and to monitor them over a spatio-temporal trace.

2. Competences

Competences to be acquired in the Bachelor's Degree Final Project:

Table 1 Competences

Code	Description
G3I406	<p><i>Ariketa original bat indibidualki egitea eta unibertsitateko epaimahai baten aurrean aurkeztu eta defendatzea; ariketa Ingeniaritza Informatikoaren arloko proiektu bat izango da, izaera profesionalakoa, eta bertan ikasketetan barneratutako konpetentziak sintetizatu eta integratuko dira.</i></p> <p>Individually carry out an original exercise and present and defend it in front of a university examining board; the exercise will consist of a computer engineering project of a professional nature in which the competences included in the studies will be synthesised and integrated.</p>
T1IT04	<p><i>Gaitasuna sistema, zerbitzu edo aplikazio informatikoak garatu eta exekutatzeko hardware eta software plataformak definitzeko, ebaluatzeko eta aukeratzeko.</i></p> <p>Ability to define, evaluate and select hardware and software platforms for the development and implementation of computer systems, services or applications.</p>
T1IT06	<p><i>Gaitasuna hardwarea, softwarea eta sareak integratzen dituzten sistema edo arkitektura informatiko zentralizatu edo banatuak sortzeko eta garatzeko.</i></p> <p>Ability to create and develop centralised or distributed computer systems or architectures integrating hardware, software and networks.</p>

3. Objectives

The objective of this project is to implement a demonstration methodology to monitor spatio-temporal properties using logic-based specification languages. To achieve this objective there are some crucial steps to do first:

- ☑ Analyse and comprehend STREL and the MoonLight monitor
- ☑ Identify the use case and data to monitor
- ☑ Set up a MQTT broker for the communication
- ☑ Establish an interface for the monitor and continuous streams
- ☑ Set up a converter from MQTT-data to moonlight-input streams
- ☑ Feed live data into moonlight and monitor.

4. Product specifications and requirements

In this chapter the specifications and requirements of the project will be explained:

4.1. Resources and materials

The resources used for this project are the following ones:

Nordic Thingy52

The Nordic Thingy52 is an easy-to-use prototyping platform. This device has different hardware specifications, built around nRF52832 Bluetooth 5 SoC:

- Arm Cortex M4 32-bit, 64 MHz
- Configurable RGB LED and button
- Environmental sensors: Temperature, humidity, air pressure, air quality (CO2 and TVOC), colour and light intensity
- Nine-axis motion sensing: Tap detection, orientation, step counter, quaternions, Euler angles, rotation matrix, gravity vector, compass heading, raw accelerometer, gyroscope, and compass data
- Sound: speaker for playing prestored samples, tones, or sound streamed over BLE (8 bit 8 kHz LoFi) and microphone streaming (ADPCM compressed 16 bit 16 kHz)
- Secure over-the-air device firmware upgrade (OTA DFU)
- Battery, rechargeable, 1440 mAh
- Low power consumption
- 2.4 GHz transceiver, Bluetooth Low Energy
- Near field communication (NFC) support



Figure 4-A Nordic Thingy52

Segger JLink

SEGGER J-Links are the most widely used line of debug probes on the market. Unparalleled performance, an extensive feature set, many supported CPUs and compatibility with popular environments.

- Programmer
- Debugger



Figure 4-B J-Link EDU Mini and it's cables

ESP 01

Espressif's ESP8266 ESP-01 delivers highly integrated Wi-Fi SoC solution with efficient power usage, compact design and reliable performance in the Internet of Things industry. The features are the following ones:

- Integrated low power 32-bit MCU
- Integrated TCP/IP protocol stack
- 802.11 b/g/n WiFi 2.4 GHz, support WPA/WPA2
- Support STA/AP/STA+AP operation modes
- 10-bit ADC, SDIO 2.0, (H) SPI, UART, I2C, I2S, IR
- Remote Control, PWM, GPIO
- Deep sleep power < 10uA, Power down leakage current < 5 uA
- Wake up and transmit packet in < 2 ms
- Standby power consumption of < 1.0 mW (DTIM3)
- +20 dBm output power in 802.11b
- Operating temperature -40C – 125C

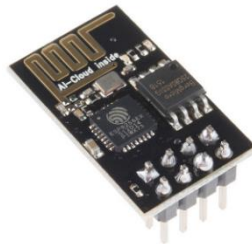


Figure 4-C ESP8266EX ESP-01 WiFi board

Zephyr

The Zephyr Project is a scalable open-source real-time operating system (RTOS) supporting multiple hardware architectures (more than 350 boards), optimized for resource-constrained devices, and built with security in mind.

The Zephyr projects are CMake-based. CMake is an open-source, cross-platform family of tools designed to build, test and package software.



Figure 4-D Zephyr and CMake logos

MoonLight

"MoonLight is a light-weight Java-tool for monitoring temporal, spatial and spatio-temporal properties of distributed complex systems, as Cyber-Physical Systems and Collective Adaptive Systems"¹

- Supports the specification of properties written with the Reach and Escape Logic (STREL)
- Implemented in Java
 - Features MATLAB interface
 - Python Interface under development

IntelliJ IDEA

IntelliJ IDEA is an Integrated Development Environment (IDE) for JVM languages. It provides clever code completion, static code analysis and refactorings, doing the routine and repetitive tasks automatically.

Programming Languages

- Java11 is the language used in the monitor and middleware
- C/C++ is the language used for the Thingy52

Communication protocol MQTT

MQTT (Message Queuing Telemetry Transport) is a lightweight publish/subscribe protocol that transports messages between devices. This protocol is designed for IoT communication and low-bandwidth environments, that makes it suitable for this project.

Bluetooth Low Energy

Bluetooth Low Energy (Bluetooth LE) is a wireless personal area network technology. Compared to Classic Bluetooth, Bluetooth LE is intended to provide considerably reduced power consumption and cost while maintaining a similar communication range.

4.2. Tests and trials**Gradle**

Gradle is a build automation tool that will be used to compile, build, run tests and more.

SonarQube

SonarQube will be used to analyse, detect bugs, vulnerabilities etc.

Trials

The monitoring method will be analysed, studied and evaluated to establish a grade to the achieved results.

4.3. Conditions for the implementation of the project

The development of the project may include the use of a server or additional sensors.

4.4. Legal aspects**General Data Protection Regulation (GDPR)**

General Data Protection Regulation (GDPR) is a regulation on data protection and privacy in the European Union (EU). IoT devices tend to process personal data so data protection should be included into any IoT solution.

¹ MoonLight: <https://github.com/MoonLightSuite/MoonLight>

5. Project information

In this section more specific information about the project is explained. The project duration is of eight months, from November 2021 to June 2022. To achieve the objective, the project has been divided in some tasks and scheduled to manage the work. The following Gantt chart describes the organization of the project.

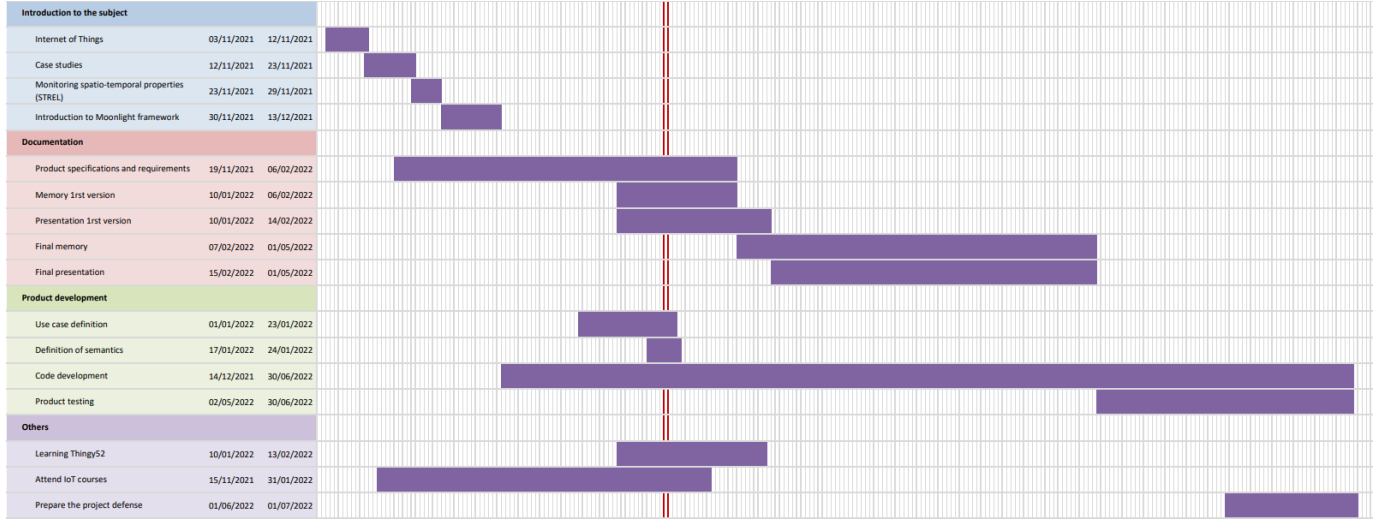


Figure 5-A Gantt chart: project schedule

5.1. Department

The project is held in Technische Universität Wien (TU Wien), Austria's largest research and educational institution in the field of technology and natural sciences. The group is the Cyber-Physical System (CPS) department.

"Cyber-Physical Systems are engineering, physical and biological systems whose operations are integrated, monitored, and/or controlled by a computational core. Components are networked at every scale. Computing is deeply embedded into every physical component, possibly even into materials. The computational core is an embedded system, usually demands real-time response, and is most often distributed. The behavior of a cyber-physical system is a fully-integrated hybridisation of computational (logical) and physical action."

(Helen Gill, US National Science Foundation)

This project is closely related to some aspects of the definition. The IoT devices are in the physical part where they are spatially distributed and networked. One main task of the project is to connect the sensors with the monitor so they can share information (i.e., networking). Finally, this data will be sent to MoonLight to monitor everything in real-time.

5.2. Tasks

The following sections will explain the work done until now and the designed path to follow. The development of the work is being uploaded to two different public repositories².

² GitHub repositories: <https://github.com/oihanagarciaa/GBL-IoTMonitoringSTREL> & <https://github.com/ennioVisco/MoonLight>

5.2.1. Use case

In this world there are many activities or events where this project might suit. IoT devices are growing up and so is the importance of monitoring/controlling. Cyber-physical systems can benefit several industries and organizations (e.g., smart cities, automotive, agriculture, health care). The use case chosen for this project is the next one:

Domotics / Smart Home Automation

A smart home is an accommodation provided with heating, lighting and electronic devices that can be monitored and controlled remotely. Through the IoT, it is possible to control and monitor the temperature, humidity, and all home devices from anywhere in the world. In this way, with a proper monitoring and controlling, the user can minimize the wastage of electrical power and increase the comfort and contentment [3].

In this use case, the project will focus on the CPS department.

- The Spatial Model will be constructed based on the map which can be found on the website of TU Wien:



Figure 5-B Map of the department [4]

- Environmental data will be collected in the department:
 1. Temperature
 2. Humidity
 3. Air pressure
 4. Air quality (CO₂ and TVOC)
 5. Light intensity
 6. Sound
- The monitoring is specification-based monitoring. To specify this spatio-temporal behaviours in a formal and human-understandable specification language, the logic used is STREL. This is the first approach of the specification properties of this project:
 - The environment conditions must be adequate [5]:
 1. The temperature must be lower than 22°C and higher than 15°C
 2. Colour must be neutral/warm white
 3. CO₂ less than X
 4. If the sound in one room is higher than X, the sound in the adjacent rooms should be less than X

5.2.2. Communication and monitor

In this section the project will be explained more deeply.

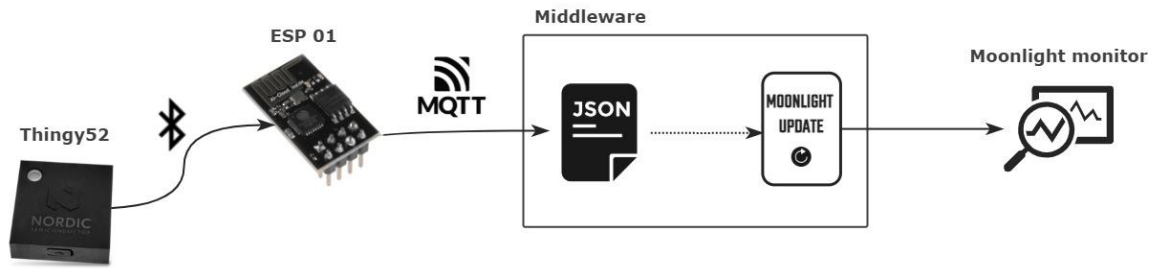


Figure 5-C Project communication overview

In the physical part there are two types of devices. First, the Thingy52's sensors collect data from the environment. Then, the information is sent in a JSON file using Bluetooth Low Energy. The ESP01 receives the file, and it publishes to the MQTT broker, where the middleware is consuming the messages. The environment variables do not change frequently, for this reason the sensor will not be uploading the data frequently neither.

Regarding the monitoring segment, the middleware is responsible of converting the JSON files to a moonlight signal. The monitor will consist in an online monitoring (i.e., the monitoring is performed incrementally). When a JSON file arrives to the middleware, it is converted to an "update", the data class used in the online monitoring. After the preprocessing, Moonlight can monitor and check if the properties are satisfied. Moonlight must be able to get continuous streams, the implementation of an interface will be done to do that.

6. Problems and solutions

Throughout the project there are things that have taken longer than expected and vice versa. In the work process I found some hitches that where no big issue and the next problem:

Table 2 Project problem and solution

Problem	I started setting up the Zephyr development environment on Windows, but I got an error that I could not fix. Zephyr SDK is not available on Windows, although there is a tutorial to set up everything on Windows, I kept hitting obstacles and the initialization process was dragging on.
Solution	I used my Linux Virtual Box Machine, where everything went more smoothly and I achieved to set up the environment and flash some projects to the Thingy52.

7. Conclusions and evaluation of the tasks carried out

While the use case was being decided, I started to develop the moonlight interface and the middleware. For now, they receive some numbers from an online broker and monitor whether they satisfy some test specifications. Apart from setting up the base, these first steps have helped me to understand the classes, methods and the workflow of the monitor. Now that I have the overview of the project defined, I have just started developing the Thingy52.

In conclusion, I have the base and the structure of the project well defined. This should make the development to go faster to achieve the objectives.

8. Future developments of the project

At the moment I will start with the Thingy52 and go climbing step by step until I reach all the objectives of the project. During this process I will have to do some things that are not explained previously in the document and are remarkable:

- **Set up the mosquito broker:** Mosquitto is a lightweight open source message broker. This is a suitable broker for this project and it's IoT sensors.
- **Wiener Linien use case:** Wiener Linien is the company that runs most of the public transit network in Vienna. They have an API that makes the public data accessible in real-time. This data is not only interesting to monitor, but also provides additional value to the project. It will be good to work with different data and to see that with minor changes the monitor can handle a completely different use case. Apart from that, there are more diverse specifications to use.

9. Contributions of the traineeships to studies

This project is helping to improve several aspects of my studies. On the one hand, I am working with different hardware, until now, I did not have the opportunity to use many devices. On the other hand, I am learning more about mathematical logic and expressions that I did not learn in the university.

Furthermore, I am understanding better the communication protocols and I am gaining experience while implementing this project. The programming language that I am mainly using is Java, a language that I already knew but I am still learning. The other language I am using is C++ a language that I did not know much about.

10. Bibliography

- [1] IHS, “Forbes,” 2016. [Online]. Available:
<https://www.forbes.com/sites/louiscolumbus/2016/11/27/roundup-of-internet-of-things-forecasts-and-market-estimates-2016/#6a558beb292d>.
- [2] L. Nenzi, E. Bartocci, L. Bortolussi, M. Loreti and E. Visconti, “Monitoring Spatio-Temporal Properties (Invited Tutorial),” in *Runtime Verification*, Springer International Publishing, 2020.
- [3] K. Diponkar, K. Md, D. Tushar, M. Abdullah and M. Ahmmad, “Smart Home Automation System Using on IoT,” *International Journal of Scientific & Engineering Research*, vol. 11, pp. 697-701, 06 2020.
- [4] TUWien and indrz, “TU Wien maps,” [Online]. Available: <https://tuw-maps.tuwien.ac.at/?q=DE0364#map>.
- [5] E. Sander, A. Caza and P. Jordan, “The physical work environment and its relationship to stress,” pp. 268-284, 06 2019.
- [6] H. Gill, US National Science Foundation, 2006.