Internet of Things (IoT)-Based System for Classroom Access Control and Resource Management

(SISGERA)

Gleiston Guerrero-Ulloa Jonathan Villafuerte-Solorzano Michael Yánez
Miguel J. Hornos Carlos Rodriguez-Domínguez

Technical State University of Quevedo, and University of Granada

Córdoba, November 2022

Contents

Introduction

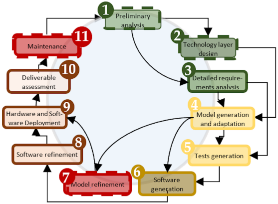
Introduction

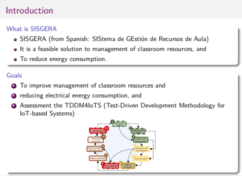
What is SISGERA

- SISGERA (from Spanish: SIStema de GEstión de Recursos de Aula)
- It is a feasible solution to management of classroom resources, and
- To reduce energy consumption.

Goals

1. To improve management of classroom resources and
2. reducing electrical energy consumption, and
3. Assessment the TDDM4IoTS (Test-Driven Development Methodology for IoT-based Systems)





In this work we propose to SISGERA from Spanish: SIStema de GEstión de Recursos de Aula. SISGERA is a prototype of an IoT-based system that was developed following the Test-Driven Development Methodology for IoT-based Systems, as a possible and feasible solution to manage classroom resources and to control access to classrooms. Therefore, the present work has a three goals.

The **first goal** Classroom resource management aims to ensure that classroom resources are switched on for the required time.

And as a consequence, **the second goal** reduce energy consumption. And **last goal** evaluate the TDDM4IoTS.

TDDM4IoTS is a methodology consisting of 11 phases, namely: Stage 1 Preliminary analysis, 2 Technology layer design, 3 Detailed requirement analysis, 4 Model generation and adaptation, 5 Test generation, 6 Software generation, 7 Model refinement, 8 Software refinement, 9 Hardware and software deployment, 10 Deliverable assessment, and 11 Maintenance. It addresses the construction of IoT device and development of both the software for IoT hardware configuration and the software for user interaction.

Goal (1): Improve management of classroom resources

Classroom resources

Current challenges

- The janitorial staff is in charge of opening the doors.
- Each janitorial staff member must control access to approximately 16 classrooms.
- On some days of the week, all classrooms start classes at the same time.
- The professor must record student attendance.
- The professor does not know all his students (at least for the first few class meetings).

Other challenges

- The professor and students may leave the classroom at any time (e.g., when they need to move to practice labs).
- The professor does not have access to the remote controls of the devices.

Classroom resources

Current challenges

- The janitorial staff is in charge of opening the doors.
- Each janitorial staff member must control access to approximately 16 classrooms.
- On some days of the week, all classrooms start classes at the same time.
- The professor must record student attendance.
- The professor does not know all his students (at least for the first few class meetings).

Other challenges

- The professor and students may leave the classroom at any time (e.g., when they need to move to practice labs).
- The professor does not have access to the remote controls of the devices.

The janitorial staff is in charge of opening the doors of approximately sixteen classrooms at the beginning of the shift in the morning and closing them at the end of the shift in the afternoon. They to managing the door locks, they are in charge of the remote controls to manage projectors and air conditioners, so they turn these on at the beginning of the shift and expect to turn them off at the end of the shift.

It is a rule that the classroom must be opened when the professor is ready to enter, which can be a delay or rush for the janitorial staff, as one day all classrooms will be required at the same time by the respective professors to start classes.

When classes start, the professor must register the student's attendance, and from the very first classes, students may be evaluated on their performance, and the professor does not always identify each student, which may lead to some impersonation.

Goal (2): Reducing electrical energy consumption

Reducing electrical energy consumption...

Example classroom FCI-008 class schedule

<table>
<tr><th>Start time to End time</th><th>Monday</th><th>Tuesday</th><th>Wednesday</th><th>Thursday</th><th>Friday</th></tr>
<tr><td>07:30 –0830</td><td rowspan="3">ME9MC1</td><td rowspan="3">ME9MC3</td><td></td><td rowspan="3">ME9MC6</td><td></td></tr>
<tr><td>08:30 –09:30</td><td></td><td></td></tr>
<tr><td>09:30 –10:30</td><td rowspan="3">ME9MC5</td><td rowspan="2">ME9MC4</td></tr>
<tr><td>10:30 –11:30</td><td rowspan="2">ME9MC2</td><td rowspan="2">ME9MC4</td><td rowspan="2">ME9MC2</td></tr>
<tr><td>11:30 –12:30</td><td></td></tr>
<tr><td>12:30 –13:30</td><td></td><td rowspan="2">EE8MC3</td><td rowspan="2">EE8MC1</td><td></td><td>EE8MC5</td></tr>
<tr><td>13:30 –14:30</td><td rowspan="2">EE8MC1</td><td rowspan="3">EE8MC3</td><td>EE8MC4</td></tr>
<tr><td>14:30 –15:30</td><td rowspan="3">EE8MC4</td><td rowspan="3">EE8MC5</td><td rowspan="3">EE8MC6</td></tr>
<tr><td>15:30 –16:30</td><td rowspan="2">EE8MC2</td></tr>
<tr><td>16:30 –17:30</td><td>EE8MC6</td></tr>
</table>

Reducing electrical energy consumption...

Example classroom FCI-008 class schedule

Start time to End time	Monday	Tuesday	Wednesday	Thursday	Friday
07:30 –0830					
08:30 –09:30	ME9MC1	ME9MC3		ME9MC6	
09:30 –10:30					
10:30 –11:30			ME9MC5		ME9MC4
11:30 –12:30	ME9MC2	ME9MC4		ME9MC2	
12:30 –13:30					EE8MC5
13:30 –14:30		EE8MC3	EE8MC1		EE8MC4
14:30 –15:30	EE8MC1			EE8MC3	
15:30 –16:30		EE8MC4	EE8MC5		EE8MC6
16:30 –17:30	EE8MC2			EE8MC6	

The SISGERA is implemented in a classroom at the Faculty of Engineering Sciences FCI. Some courses of its programs have practical hours -laboratory or field- and theoretical hours.

As an example of a classroom use schedule, we provide below the morning shift schedule of the classroom, which is used to teach the ninth module of the Mechanical Engineering degree, is show in green.

However, lessons are given for the eighth module of the Environmental Engineering degree in the same classroom during afternoon shift, which is shown in yellow. The blank spaces show that the classroom is free at that time.

Time without using the equipment

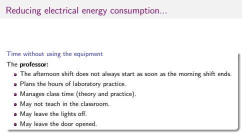
As the travel time that professors need to go to the corresponding classroom is between 5 and 15 minutes, the time that lights, air conditioners and video projectors will remain on without use will be between 10 and 105 minutes per day.

Reducing electrical energy consumption...

Time without using the equipment

The **professor:**

- The afternoon shift does not always start as soon as the morning shift ends.
- Plans the hours of laboratory practice.
- Manages class time (theory and practice).
- May not teach in the classroom.
- May leave the lights off.
- May leave the door opened.



Not to mention the days when students finish their morning classes before the afternoon classes start, for example Monday, Thursday and Friday. That maybe cause lights and equipment to remain on for hours unnecessarily. In addition, the professor plans the hours of laboratory practice, and he/she manages class time theory and practice. In practice time the students have classes in computer labs, applied physics labs or other environments depending on the program, they must leave their usual classrooms, which implies the devices and even the lights could remain on unnecessarily. Our proposal aims to avoid this waste of energy, while extending the lifetime of devices and lamps.

Goal (3): Evaluation of the development methodology

SISGERA (**SIS**tema de **GE**stión de **R**ecursos de **A**ula)

Test-Driven Development Methodology for IoT-Based Systems (TDDM4IoTS [1])

Stage 1: Preliminary Analysis

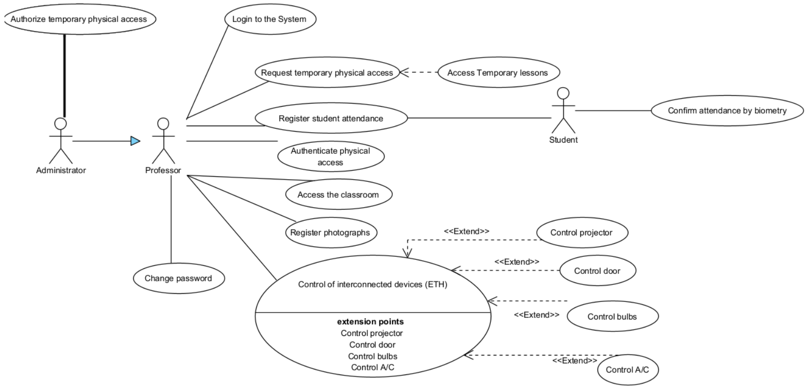


Figure: Use cases diagram of SISGERA.

SISGERA (**SIS**tema de **GE**stión de **R**ecursos de **A**ula)

Test-Driven Development Methodology for IoT-Based Systems (TDDM4IoTS [1])
Stage 1: Preliminary Analysis

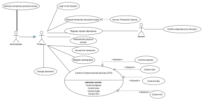


Figure: Use cases diagram of SISGERA.

Preliminary Analysis **Functional requirements** Among the main requirements to be met by SISGERA are:

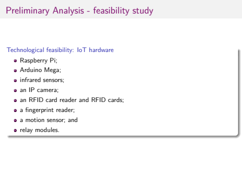
- Control interconnected device: Turns on and off the lights, air conditioner and video projector.
- Physical access control: Verifies the identity of professor through facial recognition, and Unlocks the door to allow the professor to enter.
- Records student attendance by means of RFID identification.
- Verify the identity of students by fingerprinting.
- Unlock the door to let in students arriving after the start time.

Non-functional requirements Regarding the non-functional requirements of the system, it was determined that, as it is a system to be deployed in a classroom, the power supply is guaranteed by the public utility. As for the Internet connection, a permanent Wi-Fi connection is available.

Preliminary Analysis - feasibility study

Technological feasibility: IoT hardware

- Raspberry Pi;
- Arduino Mega;
- infrared sensors;
- an IP camera;
- an RFID card reader and RFID cards;
- a fingerprint reader;
- a motion sensor; and
- relay modules.



All the IoT hardware components necessary to meet the functional requirements exist, and were within our reach.

- Raspberry Pi: for local processing and sending data via Internet.
- Arduino Mega: to treat the signals captured by the sensors and send the data to the Raspberry Pi through the Bluetooth module.
- infrared sensors: turn on and off air conditioner and video projector.
- an IP camera: to recognize people through live video.
- cards and RFID card reader: to identify students when accessing the classroom.
- a fingerprint reader: to identify students more securely when required.
- a motion sensor: to detect people presence inside the classroom; and
- relay modules: to turn on and off lights and equipment.

Preliminary Analysis - feasibility study

Technological feasibility: Software tools and technologies

- Python language
- Java language
- Application Server (GlassFish or Apache Tomcat)
- PostgreSQL

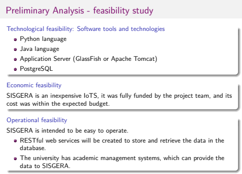
Economic feasibility

SISGERA is an inexpensive IoTS, it was fully funded by the project team, and its cost was within the expected budget.

Operational feasibility

SISGERA is intended to be easy to operate.

- RESTful web services will be created to store and retrieve the data in the database.
- The university has academic management systems, which can provide the data to SISGERA.



In addition, in Technological feasibility have the Software tools and technologies

- Python Language: to facial recognition with OpenCV. Runs on Rapsberry PI
- Java langauge: to construct the Mobile app and Web Services RESTfull
- SISGERA was tested with GlassFish as the application server
- PostgreSQL: Persistence layer

Economic feasibility
SISGERA is an inexpensive IoTS, it was fully funded by the project team, and its cost was within the expected budget.

Operational feasibility
SISGERA is intended to be easy to operate. RESTful web services will be created to store and retrieve the data in the database. The administrator will be able to take photographs of professors for facial recognition, and also feed the system with student and academic planning data.

Stage 2: Technology Layer Design

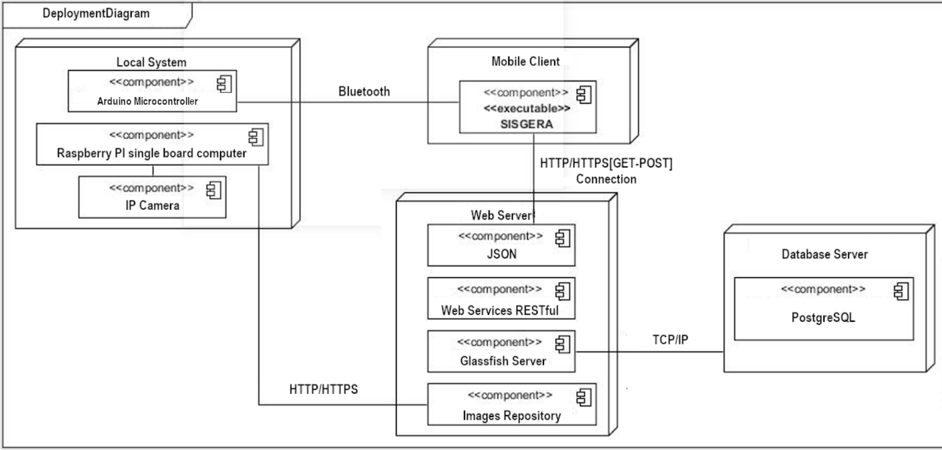


Figure: SISGERA Architecture.

Stage 2: Technology Layer Design

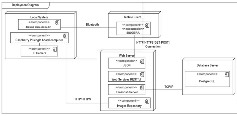


Figure: SISGERA Architecture.

The SISGERA architecture is designed in 4 layers:

- **First layer:** local processing or set of components that could be called the device,
- **Second Layer:** the mobile application with which the professor can interact with the environment,
- **Third layer:** the web server where the application server is installed, which provides the web services, and the storage of the images for facial recognition, and of course
- **Fourth layer:** the data persistence layer that basically consists of a PostgreSQL database server.
- For more details of the **device** and its **components** you can refer to the paper.

Stage 3: Detailed requirement analysis

Elicitation of system requirements

- The development of the system was divided into different deliverables and each of them was analysed.
- Semi-structured use cases were used to collect those requirements.
- The system meets the requirements demanded by the end users.

Semi-structured use cases

Semi-structured use cases are used to describe the flow of actions that are carried out to obtain an outcome of observable value by a particular actor.

Stage 3: Detailed requirement analysis

Elicitation of system requirements

- The development of the system was divided into different deliverables and each of them was analysed.
- Semi-structured use cases were used to collect those requirements.
- The system meets the requirements demanded by the end users.

Semi-structured use cases

Semi-structured use cases are used to describe the flow of actions that are carried out to obtain an outcome of observable value by a particular actor.

Elicitation of system requirements

The development of system was divided into different deliverables. For each one, the requirements were analysed in detail to obtain the tests that the software must pass, thus ensuring that the system meets the requirements demanded by the end users.

Semi-structured use cases were used to collect those requirements. Semi-structured use cases describe the flow of actions that are carried out to obtain an outcome of observable value by a particular actor.

Stage 4: Model Generation and Adaptation

Platform-independent model

IoTS are heterogeneous in nature, so a platform-independent model was obtained before generating a platform-specific model.

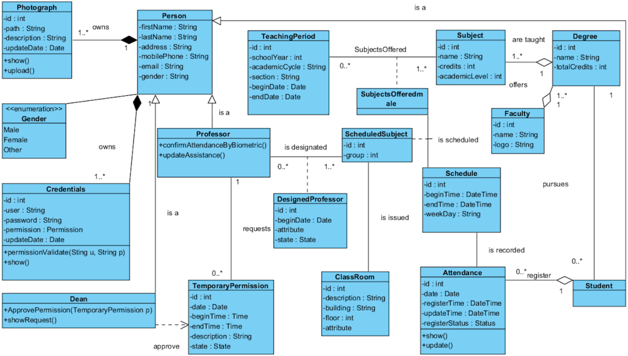


Figure: Class diagram of SISGERA

Stage 4: Model Generation and Adaptation

Platform-independent model

IoTS are heterogeneous in nature, so a platform-independent model was obtained before generating a platform-specific model.

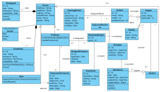


Figure: Class diagram of SISGERA

In this phase, the class diagram was created, considered one of the most important in the construction of systems. It represent the architecture of the system software. With the appropriate tools, it was used to generate the database and part of the necessary code.

Generation and refinement Stages

- **Stage 5: Test generation:** The tests were written considering the different cases that may occur when trying to enter or leave the classroom.
  - The professor is out of time,
  - he/she takes care to turn off the equipment,
  - etc.
- **Stage 6: Software generation**
  - Automatic generation of the software, Visual Paradigm was used
  - Visual Paradigm generate the code for each class with its attributes and methods
  - The code was used in the mobile application
- **Stage 7: Model refinement:** This stage was not necessary
- **Stage 8: Software refinement:** The guidelines for writing clean code were considered for writing the software code.
  - It was carried out after obtaining the mobile application software for SISGERA.
  - The IoT hardware configuration software was not refined.

Generation and refinement Stages

- **Stage 5: Test generation:** The tests were written considering the different cases that may occur when trying to enter or leave the classroom.
  - The professor is out of time,
  - he/she takes care to turn off the equipment,
  - etc.
- **Stage 6: Software generation**
  - Automatic generation of the software, Visual Paradigm was used
  - Visual Paradigm generate the code for each class with its attributes and methods
  - The code was used in the mobile application
- **Stage 7: Model refinement:** This stage was not necessary
- **Stage 8: Software refinement:** The guidelines for writing clean code were considered for writing the software code.
  - It was carried out after obtaining the mobile application software for SISGERA.
  - The IoT hardware configuration software was not refined.

Stage 5: Test generation: The system must respond appropriately to each of the possible cases that may occur. For example: it must not allow a professor to enter if it is not his or her approved schedule. Turn on the lights and equipment when a person enters. Lights and equipment must be turned off when there are no people in the classroom.
Stage 6: Software generation: This process can be manual or automatic. For the automatic generation of the software, Visual Paradigm was used. Visual Paradigm generate each class with its attributes and methods, to be used in the mobile application. 100% of the Arduino code and the rest of the business logic code was written by the development team.
Stage 7: Model refinement: This stage was not necessary because of the size of the project and the developers' mastery of the domain. This is foreseen by the authors of TDDM4IoTS.
Stage 8: Software refinement: This stage was executed only once, after obtaining the mobile application software for SISGERA. It was not necessary to refine the device configuration software, as it was written considering from the beginning the guidelines to obtain a clean code.

Stage 9: Hardware and software deployment

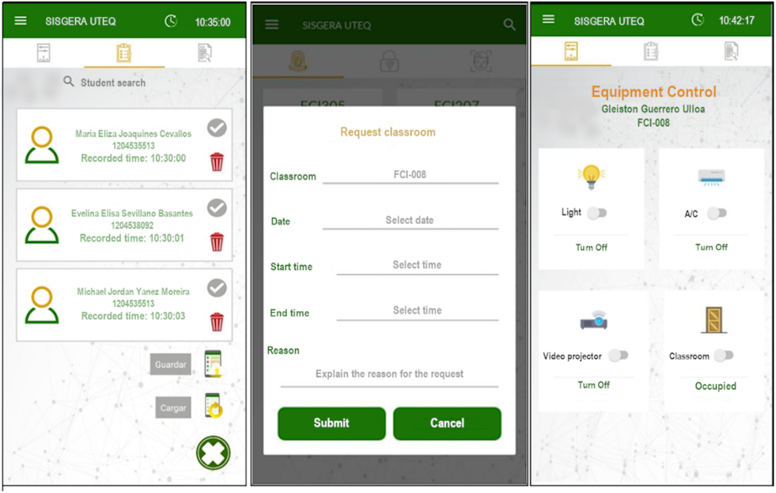


Figure: Screenshots of the SISGERA mobile application.

Stage 9: Hardware and software deployment

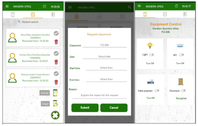


Figure: Screenshots of the SISGERA mobile application.

The device was deployed as determined in the environment analysis. It was deployed in the classroom, connecting directly to the public power supply without the need for a battery, a router was installed to fix the Wi-Fi signal with good quality.

The mobile application was installed on the smartphones of the professors who tested the application. This application has professor and Dean roles. The latter role is the one that authorises the professor to use the requested classroom.

The first screenshot shows list of students who have registered their attendance, the middle screenshot shows a form to request classroom. The user Dean is one who approves his/her request. At the last screenshot shows the menu options for controlling equipment and lights, and unlocking the door.

Assessment and maintenance Stages

Stage 10: SISGERA assessment

The device and the mobile application

- Passed all tests de system acceptation.
- The usability evaluation of SISGERA mobile application.
- After this results, improvements were implemented.
- Those improvements resulted in 100% acceptance by professors.

Stage 11: Maintenance

The Maintenance was not carried out, due to the low complexity of the project. Moreover, it is a prototype that is to be evaluated.

Assessment and maintenance Stages

Stage 10: SISGERA assessment

The device and the mobile application

- Passed all tests de system acceptation.
- The usability evaluation of SISGERA mobile application.
- After this results, improvements were implemented.
- Those improvements resulted in 100% acceptance by professors.

Stage 11: Maintenance

The Maintenance was not carried out, due to the low complexity of the project. Moreover, it is a prototype that is to be evaluated.

Stage 10: SISGERA assessment

The device and the mobile application passed all tests specified in the test generation stage. In addition, the usability evaluation of SISGERA was done along 5 weeks only by professors. Following these results, more professors were involved in the design of the mobile application interfaces, to implement improvements. Those improvements resulted in 100% acceptance by professors.

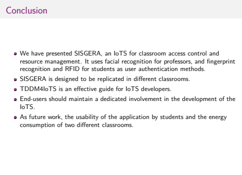
Stage 11: Maintenance

SISGERA is a prototype, but considering the improvements made by the professors' comments, maintenance was not carried out as it is not a production system. However, it is planned to put it into production for a more exhaustive evaluation and to consider the maintenance stage.

Conclusion

Conclusion

- We have presented SISGERA, an IoTS for classroom access control and resource management. It uses facial recognition for professors, and fingerprint recognition and RFID for students as user authentication methods.
- SISGERA is designed to be replicated in different classrooms.
- TDDM4IoTS is an effective guide for IoTS developers.
- End-users should maintain a dedicated involvement in the development of the IoTS.
- As future work, the usability of the application by students and the energy consumption of two different classrooms.



Conclusion

- We have presented SISGERA, an IoTS for classroom access control and resource management. It uses facial recognition for professors, and fingerprint recognition and RFID for students as user authentication methods.
- The combination of RFID authentication and fingerprint recognition as identification methods can be used to deter students from identity theft.
- SISGERA is designed to be replicated in different classrooms as a distributed system.
- The reduction in energy consumption can be calculated according to the time that the equipment are kept off.
- TDDM4IoTS is an effective guide for IoTS developers, especially for those who do not have sufficient experience in IoTS development.
- End-users should maintain a dedicated involvement in the development of the IoTS, especially in the design of its user interfaces. TDDM4IoTS encourages user participation from the earliest stages of IoTS development.

Thank you!