

MONITOR DE CONSUMO ELÉCTRICO +

Presentación de
Proyecto Integrador
Final Sprint

— Autores:
Benny Ruiz
Erick H Cerecedo
Fernanda Hernández

Análisis



Requerimientos

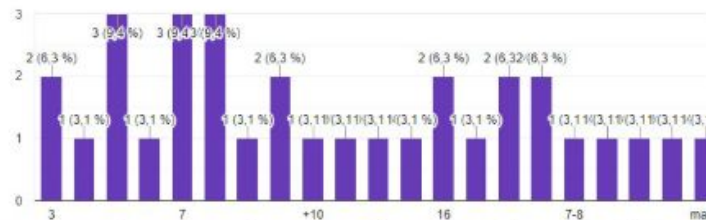
01	Control sobre Descripción breve	02	Interacción de espacios Flexibilidad para el usuario		
03	Usabilidad Facilidad de uso, comprensión y aprendizaje	04	Enfoque al usuario Accesible, útil, valuable	+	+
				×	×
				+	+

Justificación

Entrevistas

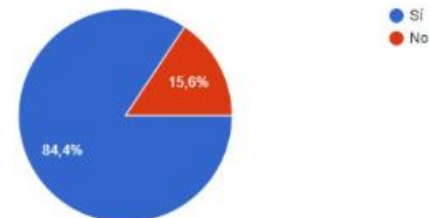
¿Cuántos aparatos electrónicos tienes?

32 respuestas



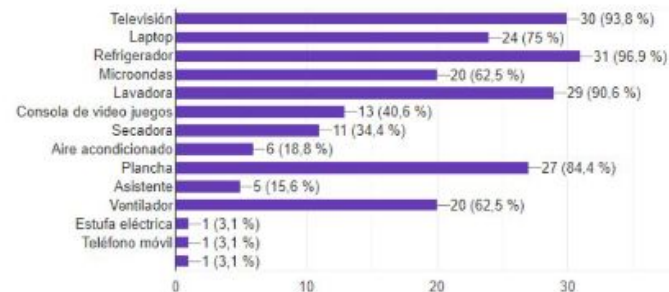
¿Cada que sales te preocupa dejar luces prendidas?

32 respuestas



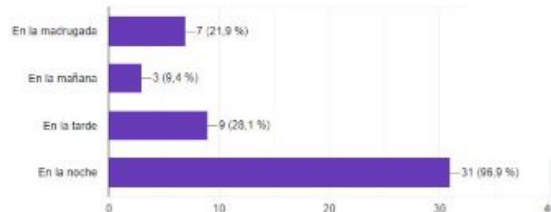
Selecciona lo que tienes:

32 respuestas



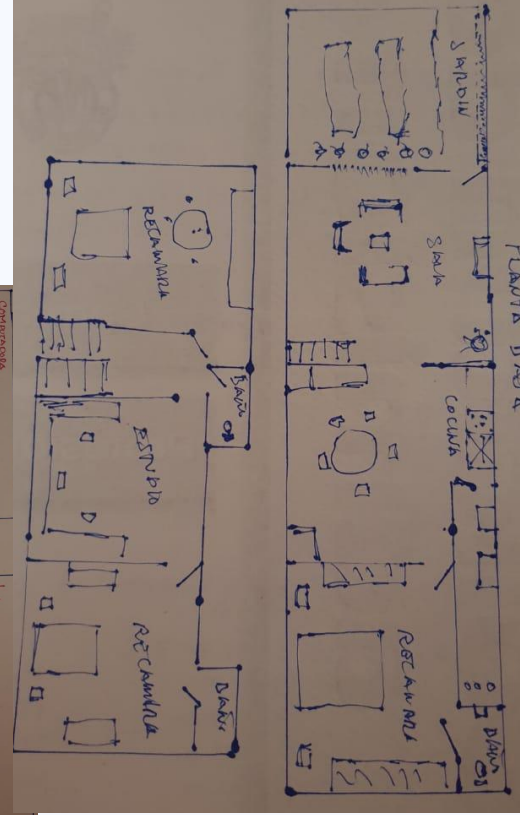
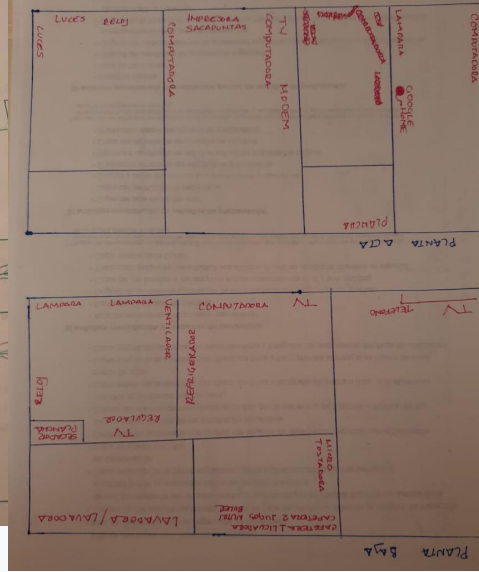
¿Cuánto tiempo usas tus focos?

32 respuestas



Análisis

Empatizar con el usuario



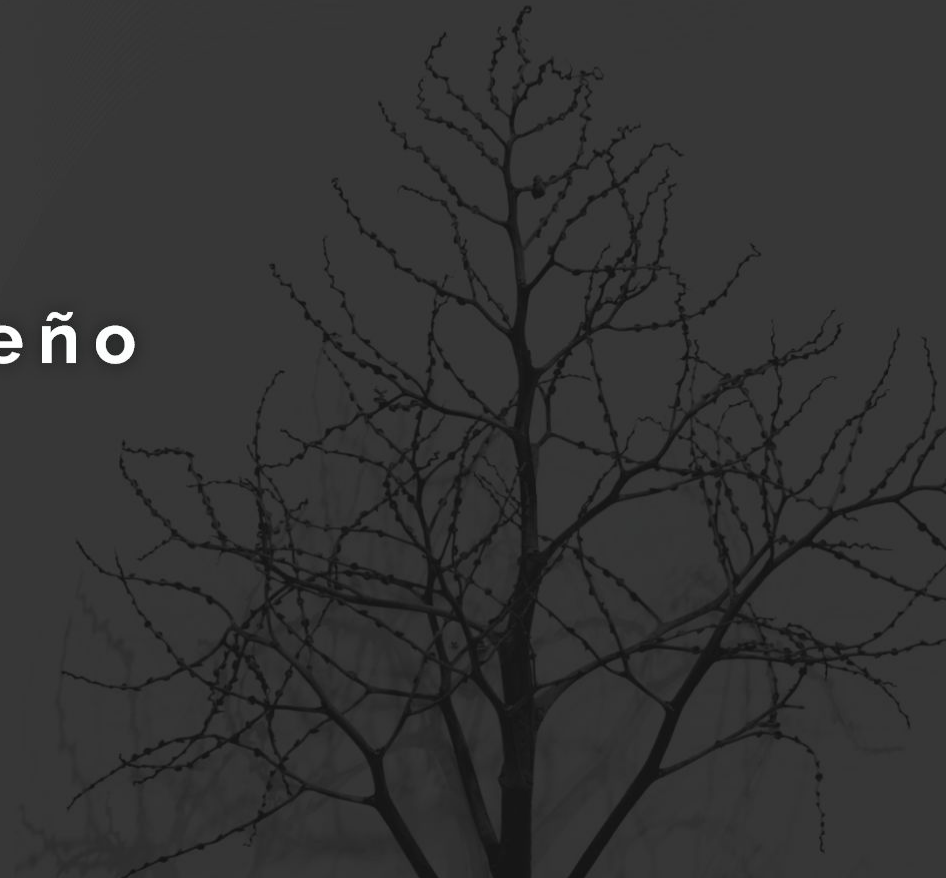
+

+

+

+

Diseño



Diseño

Casos de uso



+

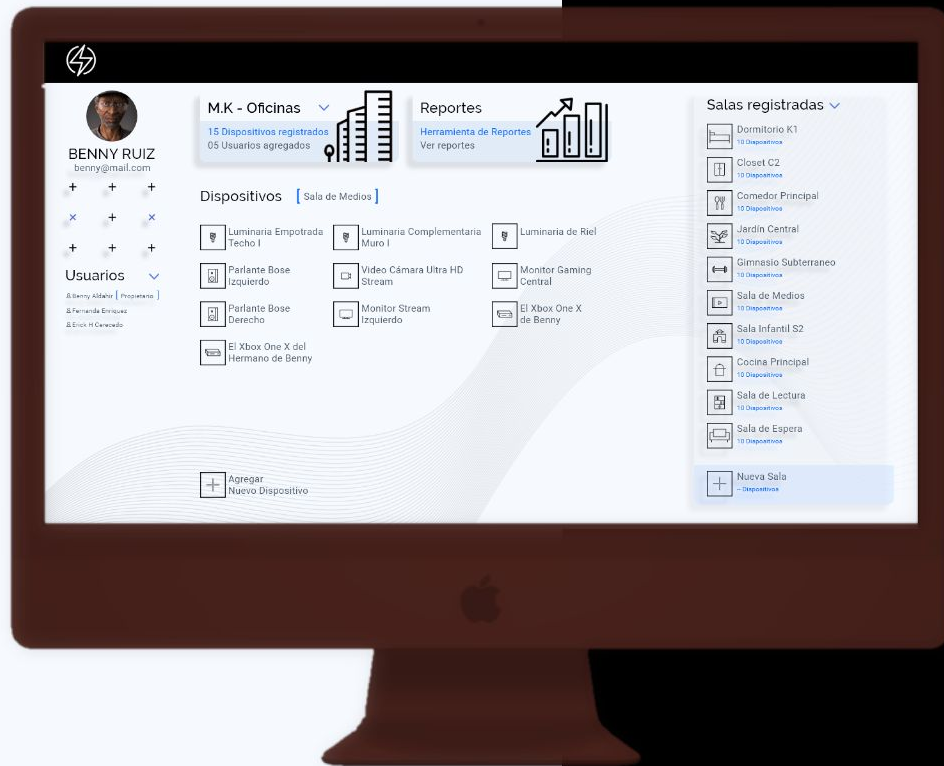
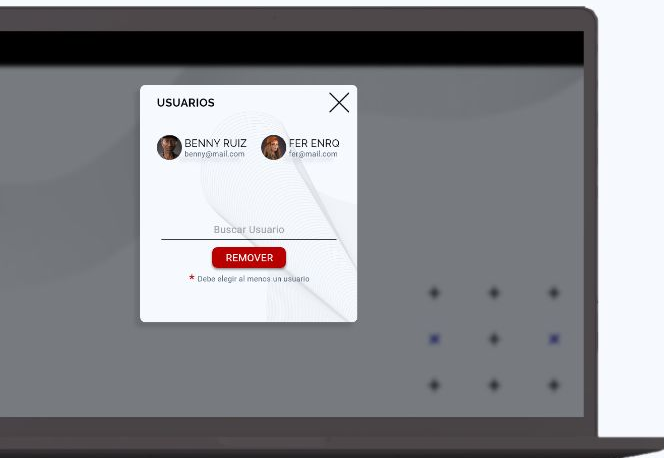
+

+

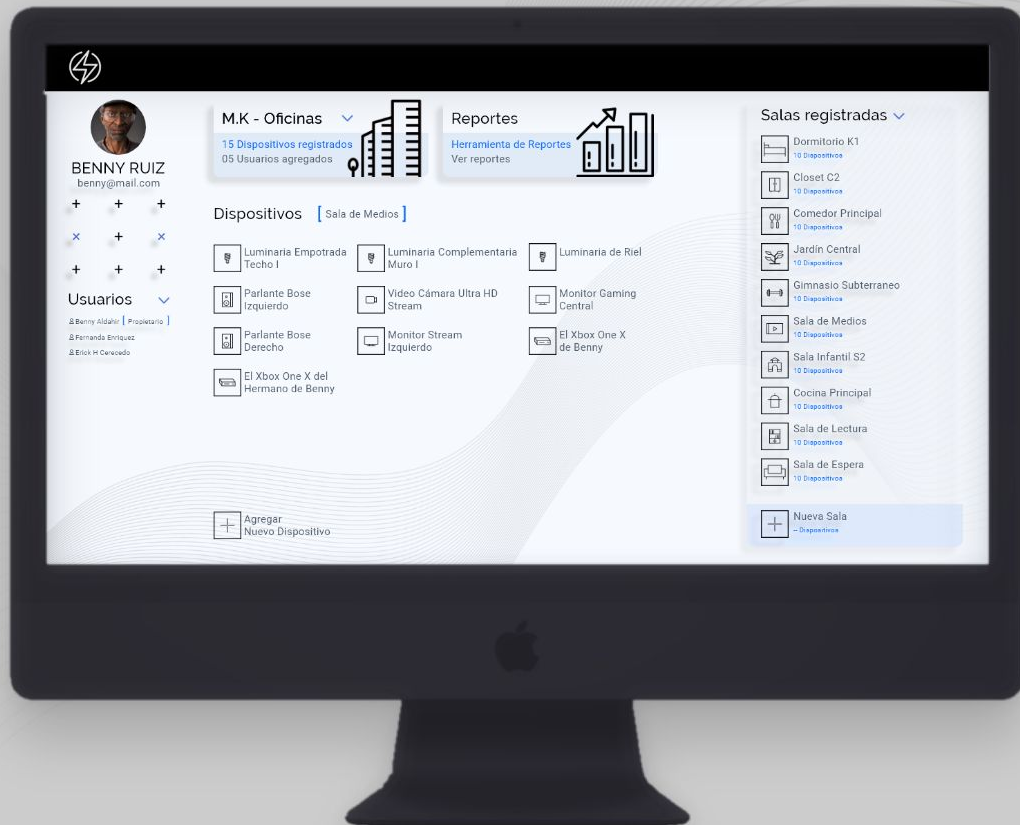
+

Diseño

Mockup's





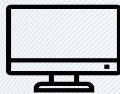


Diseño

Arquitectura



React
Framework



Node.js
Entorno



API
Python



Framework
Flask



Diseño

Arquitectura

Profile

User

List Stage

Stage

ScreenRoom

AddRoom

ListRoom

Room

ScreenDevice

AddRoom

ListDevice

Room

+

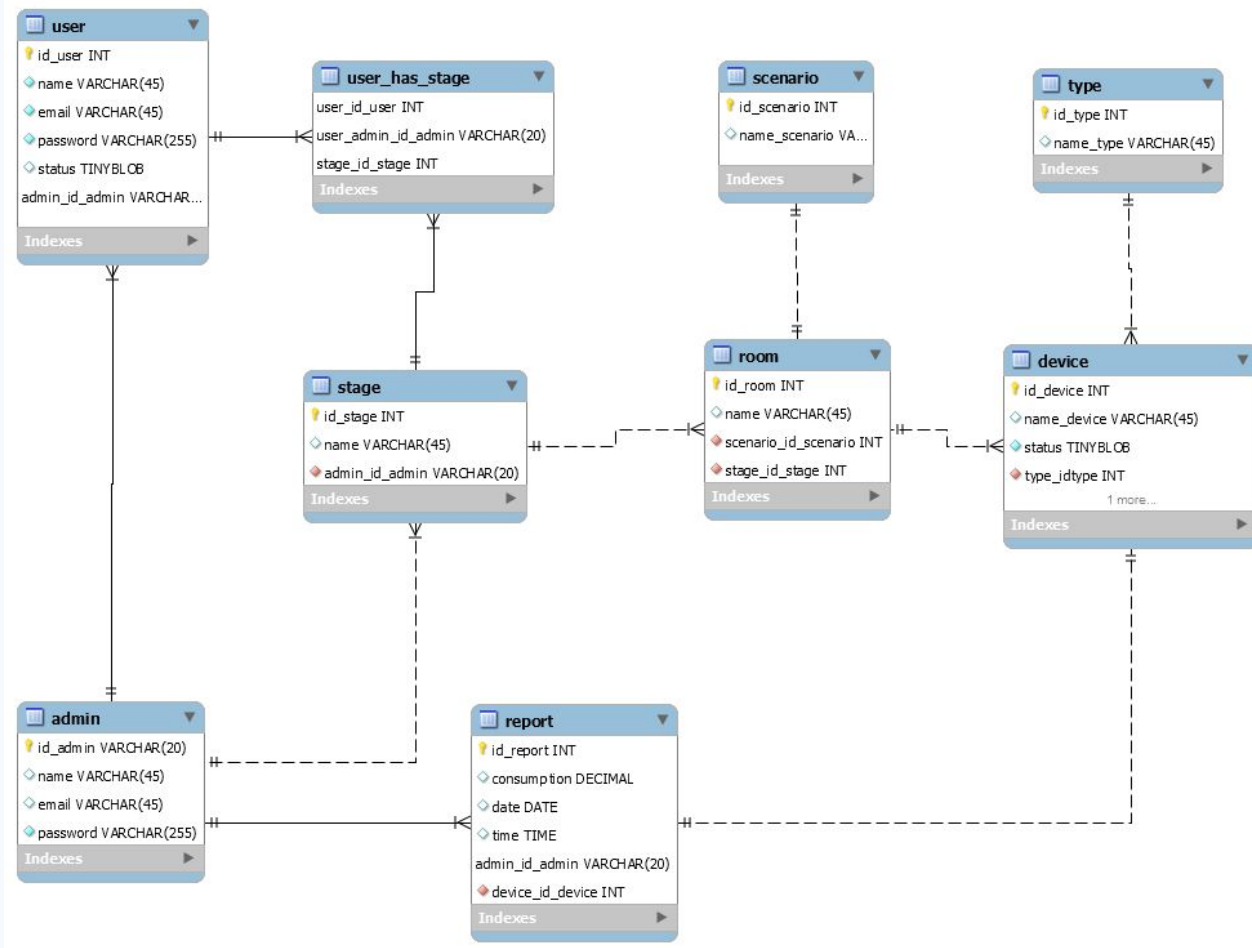
+

+

+

Diseño

Base de datos



Implementación



Implementación

— **Backend**

Implementación

Base de datos



Procedimientos almacenados

Datos por default

```
MariaDB [iotdb]> show tables;
```

Tables_in_iotdb
admin
device
report
room
scenario
stage
type
user
user_has_stage

```
DELIMITER ;;
CREATE DEFINER=`root`@`localhost` PROCEDURE `PasswordAdministrador`(IN password VARCHAR(255))
BEGIN
SELECT * FROM admin WHERE password = password;
END ;;
DELIMITER ;

-- LoginUser
DELIMITER ;;
CREATE DEFINER=`root`@`localhost` PROCEDURE `PasswordUser`(IN password VARCHAR(255))
BEGIN
SELECT * FROM user WHERE password = password;
END ;;
DELIMITER ;

--Insertar un usuario
DELIMITER ;;
CREATE DEFINER=`root`@`localhost` PROCEDURE `insertUser`
(IN name VARCHAR(45), IN email VARCHAR(45), IN encriptado VARCHAR(255), admin VARCHAR(20))
INSERT INTO user
(name, email, password, status, admin_id_admin)
VALUES(name, email, encriptado, TRUE, admin) ;;
DELIMITER ;

--Mostrar todos los usuarios
DELIMITER ;;
CREATE DEFINER=`root`@`localhost` PROCEDURE `showUser`()
BEGIN
SELECT * FROM user where status = 1;
END ;;
```


Implementación

API Rest

- Petición GET
- flask.response(), 200
- JSON
- Encriptación

> Administrador / Usuarios

```
##### /login?email=&password=
##### /insert?name=&email=&password=&admin=
##### /delete?email=
##### /modify?email=&name=&email=
##### /search?email=
##### /showAllUser
```



python



Flask

> Stage

```
##### /insertStage?name=&user=&admin=
##### /deleteStage?id_stage=
##### /searchStage?id_stage=
##### /modifyStage?id_stage=&name=
##### /showAllStage
##### /getStage?id_stage=
##### /stageCountDevices?id_stage=
##### /stageCountUsers?id_stage=
```

> Room

```
##### /insertRoom?name=&id_scenario=&id_stage
##### /deleteRoom?id_room=
##### /searchRoom?id_room=
##### /modify?id_room=&name=&id_scenario=
##### /showAllRoom
##### /countDevices?id_room=
##### /checkRoom?name_room=
##### /getNameRoom?id_room=
```

> Dispositivos

```
##### /insertDevice?name_device=&type_idtype=&id_room=
##### /deleteDevice?id_device=
##### /searchDevice?id_device=
##### /modifyDevice?id_device=&name=&
##### /checkType?id_type=&
##### /showAllDevice
```

A close-up, slightly blurred photograph of a person's hand holding a yellow pen, drawing a diagram on a tablet. The diagram consists of several rectangular boxes connected by lines, representing a system architecture or flowchart. One box on the left contains a smaller box with a grid, and another box on the right contains a bar chart. The background is a plain, light-colored surface.

Implementación

— Frontend

Implementación

— Interfaz



Storybook

+

+

+

+

```
JS Profile.js
JS Room.js
JS Room.stories.js
JS RoomList.js
JS RoomList.stories.js
JS RoomScreen.js
JS RoomScreen.stories.js
JS Stage.js
JS StageList.js
JS User.js
```

```
JS AddDevice.js
JS AddDevice.stories.js
JS AddDeviceList.js
JS AddDeviceList.stories.js
JS AddRoom.js
JS AddRoom.stories.js
JS AddUser.js
JS Device.js
JS Device.stories.js
JS DeviceForm.js
JS DeviceForm.stories.js
JS DeviceFormName.js
JS DeviceFormName.stories.js
JS DeviceFormType.js
JS DeviceFormType.stories.js
JS DeviceList.js
```

Diseño

— Demostración del Proyecto



+

+

+

+

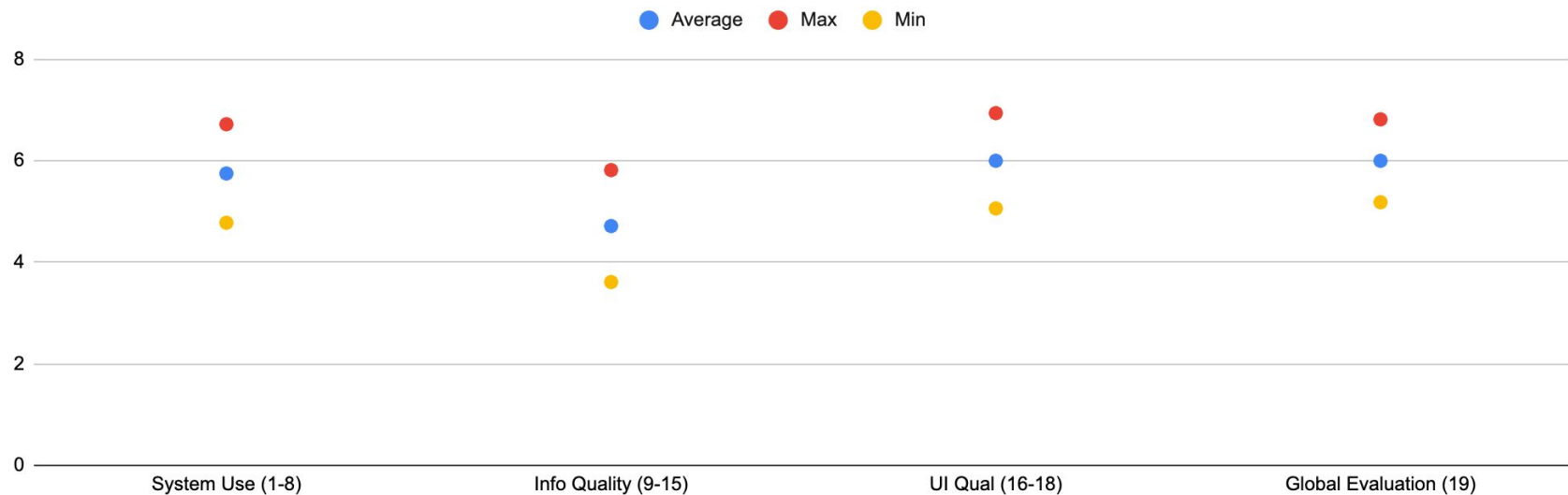
Justificación



Justificación

Evaluaciones

System Use (1-8), Info Quality (9-15), UI Qual (16-18) y Global Evaluation (19)



Opiniones de Los Usuarios.

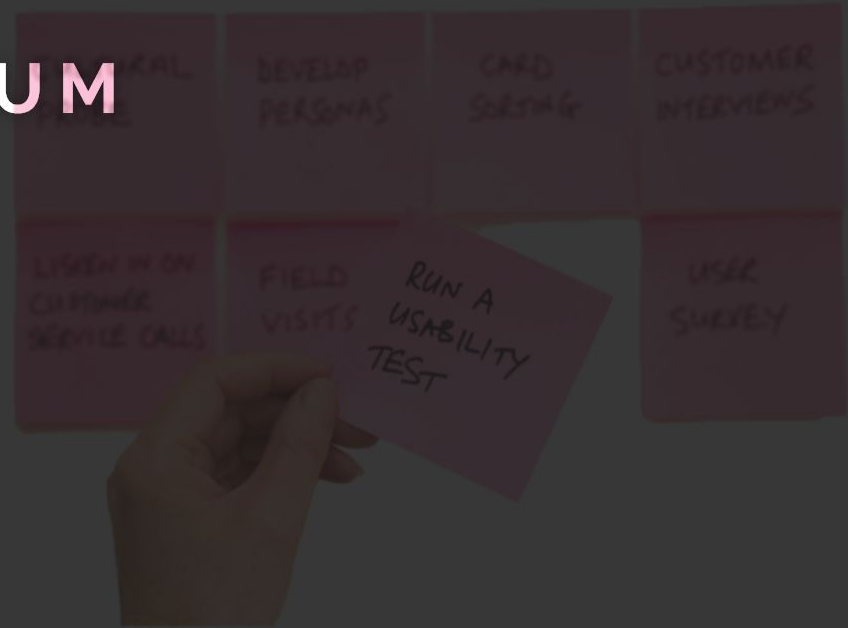
“Excelente trabajo, muy informativo!

Sin duda, algo muy útil para tener más información y control sobre nuestro gasto en energía eléctrica.”

“Felicidades, sin duda un proyecto muy interesante, viable y funcional. Es previsible su éxito por el gran impacto social que representa. Enhorabuena por sus emprendedores.”

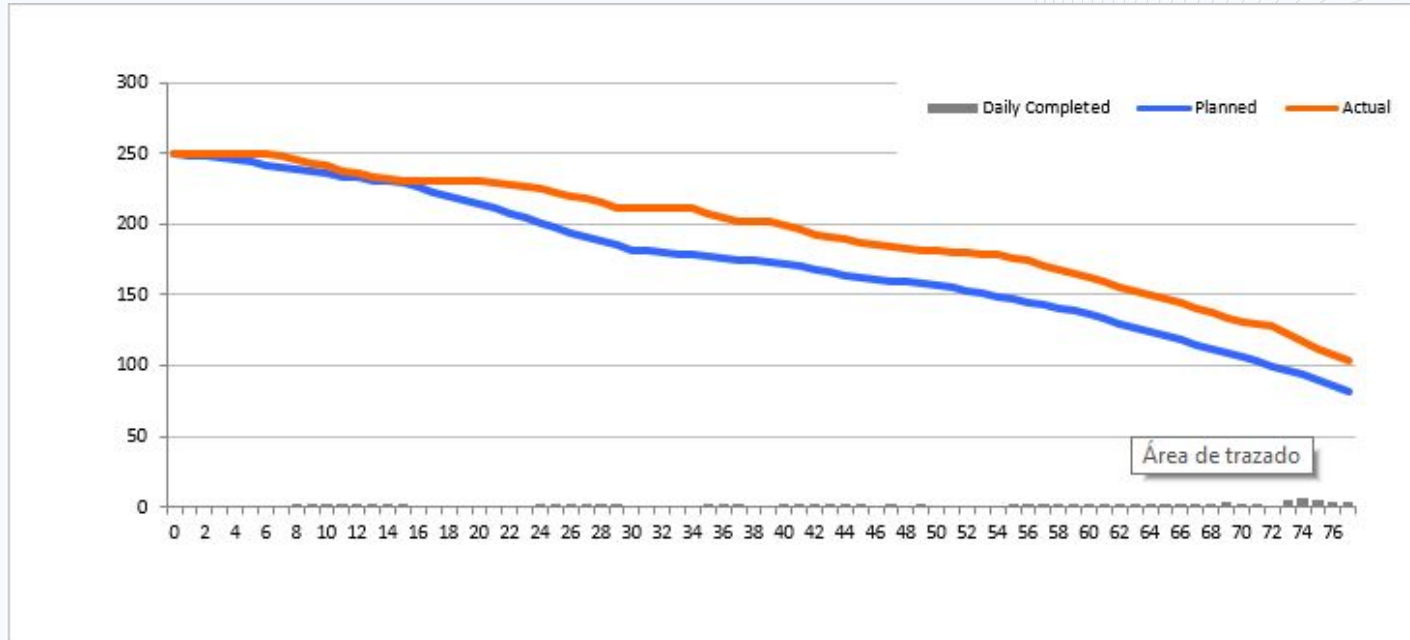
“Es un proyecto interesante. Este proyecto representa una contribución positiva a la red de productos de IoT al promover un consumo responsable de energía eléctrica. Todo cambio empieza por la observación y evaluación de lo que se hace actualmente”

SCRUM



SCRUM

Proceso del proyecto



SCRUM

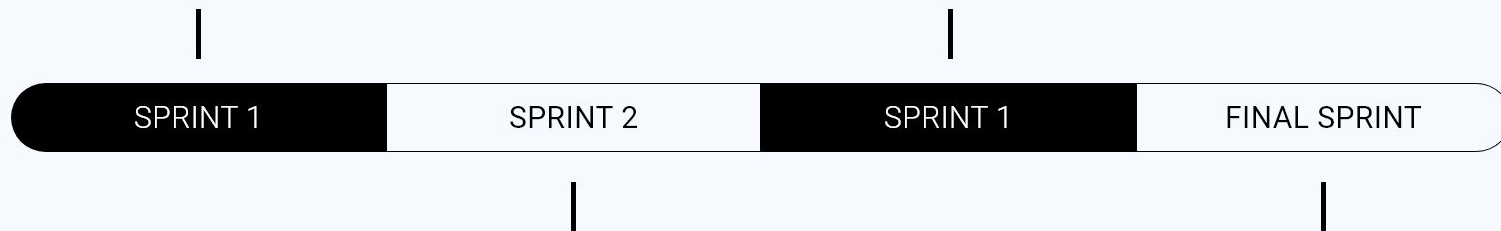
Proceso del proyecto

+

+

+

+



Electric consumption project



Gracias