**Smart**

**Meeting Room**

**for**

# [**Distributed System**](https://github.com/brunocguedes37/DistributedSystem)

**National College of Ireland**

**Project made by:**

Bruno x14104571

Nuth x14101629

GitHub : https://github.com/brunocguedes37/DistributedSystem.git

**Table of Contents**

[Introduction: 1](#_Toc511579836)

[4 complex services: 1](#_Toc511579837)

[SmartDoor GUI and Console display. 2](#_Toc511579838)

[SmartProjector GUI and Console display. 3](#_Toc511579839)

[SmartChair GUI and Console display. 4](#_Toc511579840)

[SmartAssistent Console display. 5](#_Toc511579841)

[Message format Gson 7](#_Toc511579842)

[jmDNS 8](#_Toc511579843)

[User Interface GUI 10](#_Toc511579844)

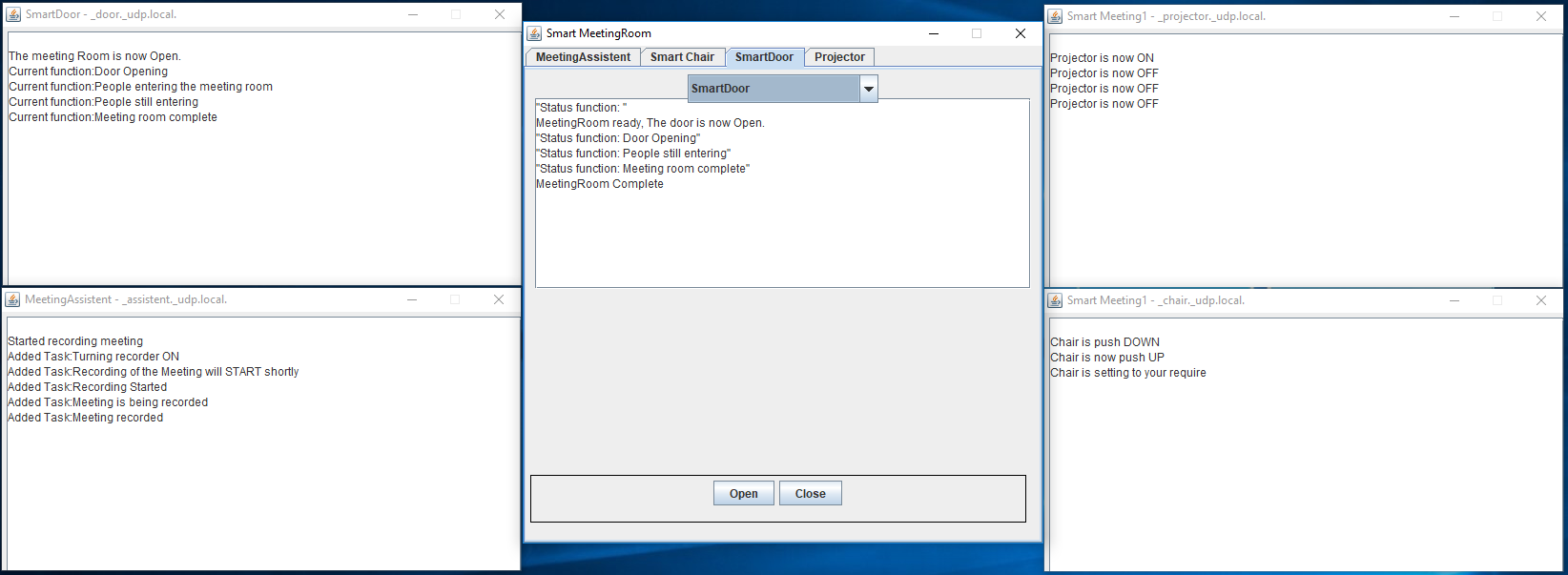
[Error Handling 13](#_Toc511579845)

[Conclusion 15](#_Toc511579846)

# 

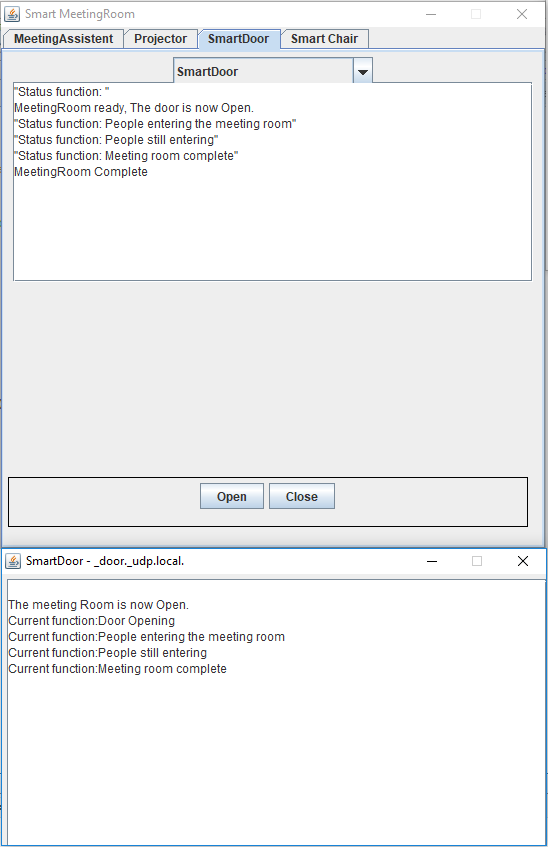
# **Introduction:**

For this project we have created a Smart environment that in our case it is a Smart MeetingRoom, we based our project on the ProjectSample provided adding/creating new code and modifying/adapting the already existent code. For this project new have used NetBeans and GitHub. Your devices must publicize themselves and discover each other using Zeroconf, there is no configuration necessary to utilize the services.

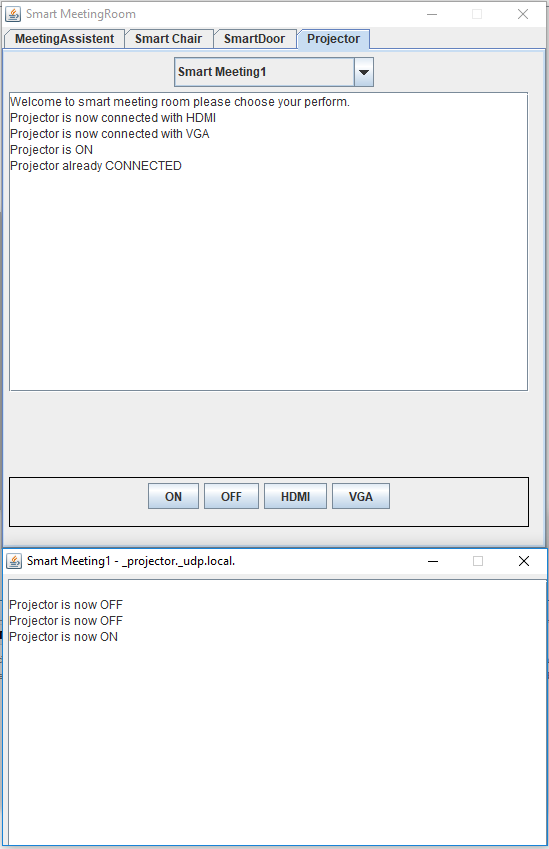


# **4 complex services:**

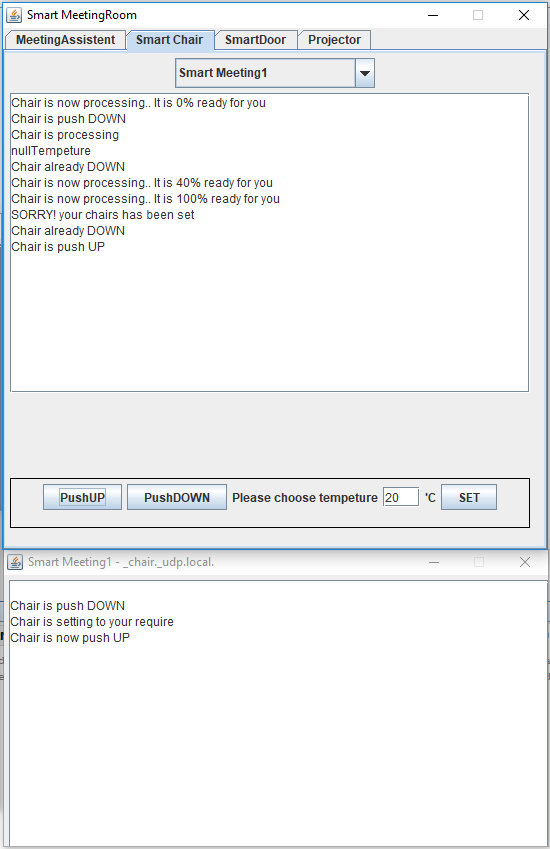
## **SmartDoor GUI and Console display.**



## **SmartProjector GUI and Console display.**



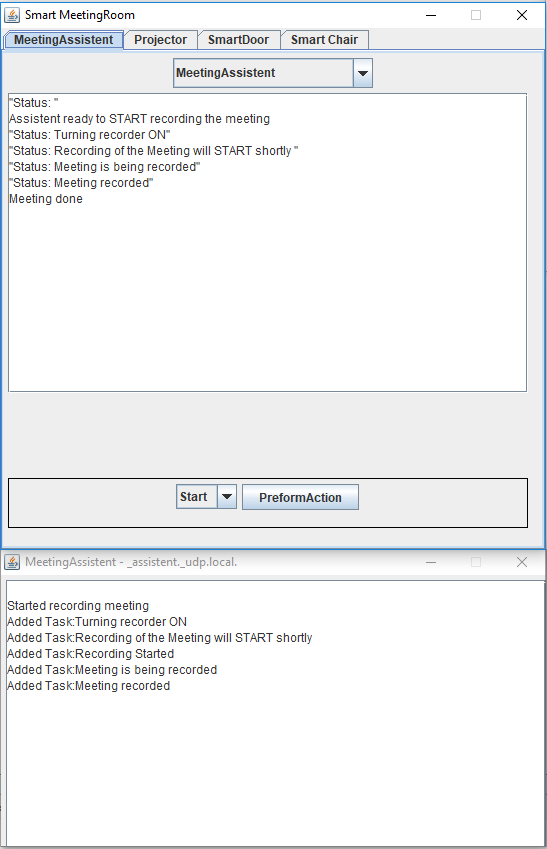
## **SmartChair GUI and Console display.**



## 

## **SmartAssistent Console display.**

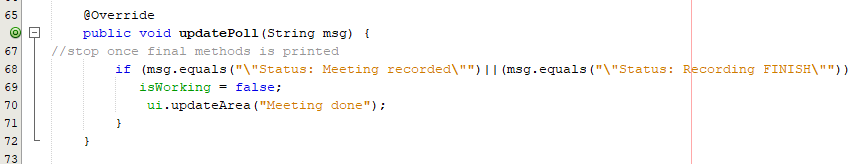
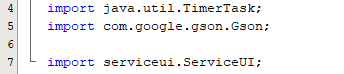
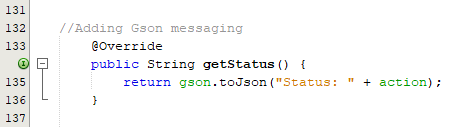
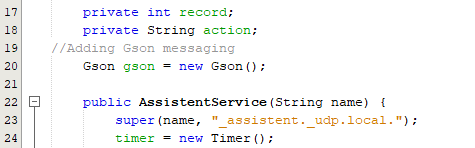
## 

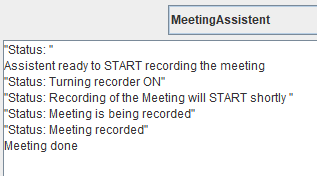
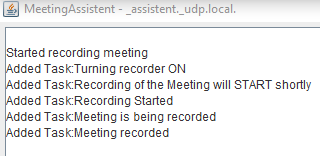


# 

# **Message format Gson**

For our project we decided to use Gson for the messages and for automated conversion to objects. Bellow we have the code we have used and a message display.



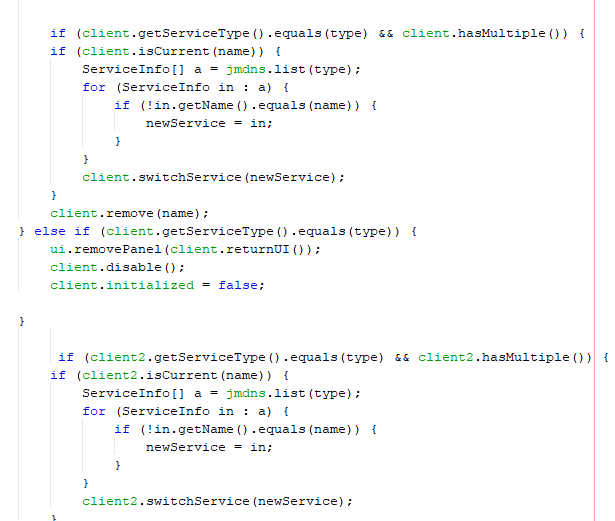
****

# 

# **jmDNS**

For our project we use jmDNS (java multicast DNS) to connect devices. This is a Java implementation of multi-cast DNS and can be used for service registration and discovery in local area networks.

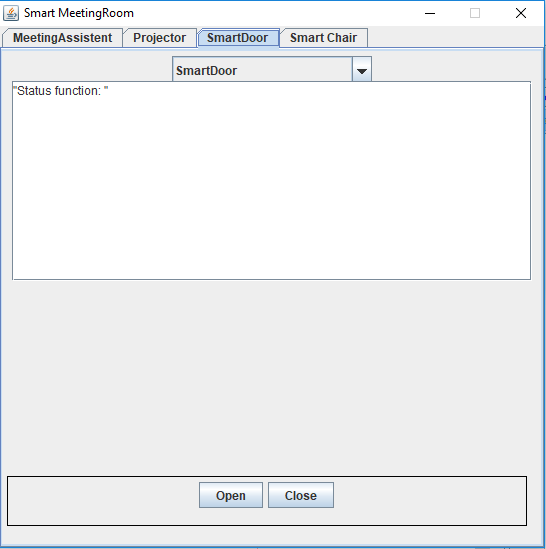
Bellow we have screenshots of the code for this implementation:

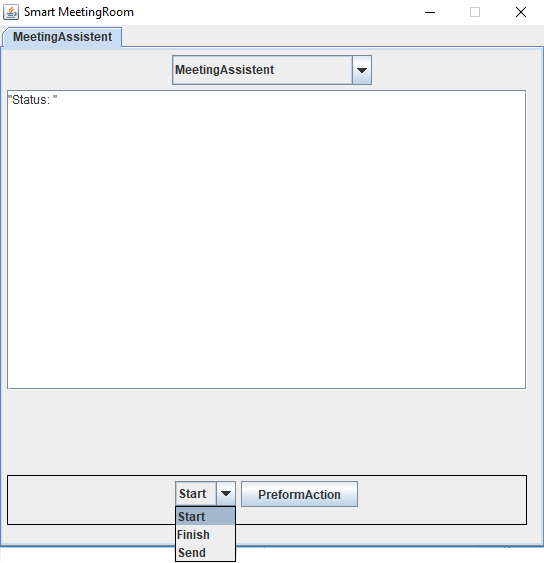


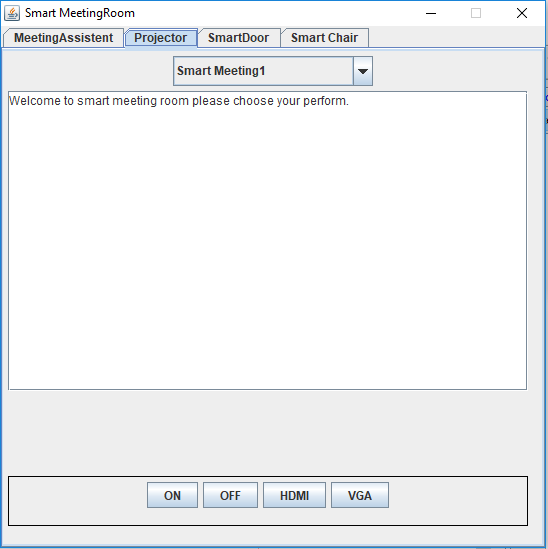
# 

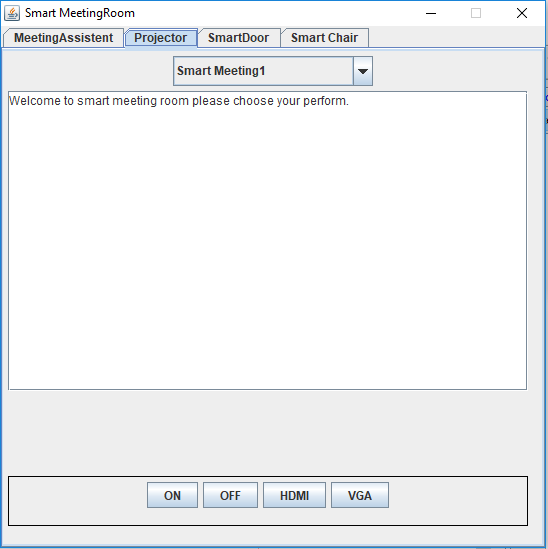
# **User Interface GUI**

Bellow we have screenshots of the 4 different services graphical user interface (GUI), We have used a simple GUI, composed of “Jbuttom”, “JComboBox”, “Jpanels”, “Jtextfield” and “Jlabels”.





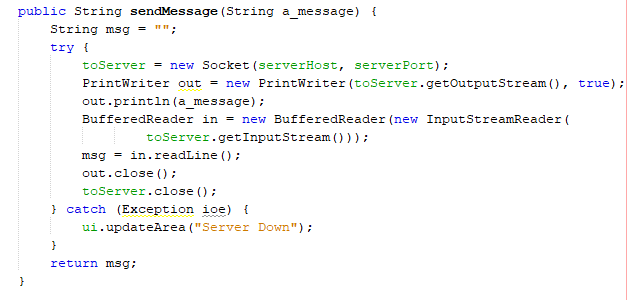
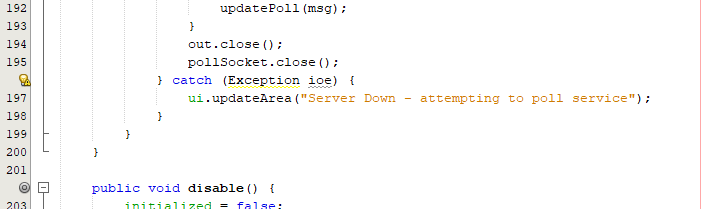


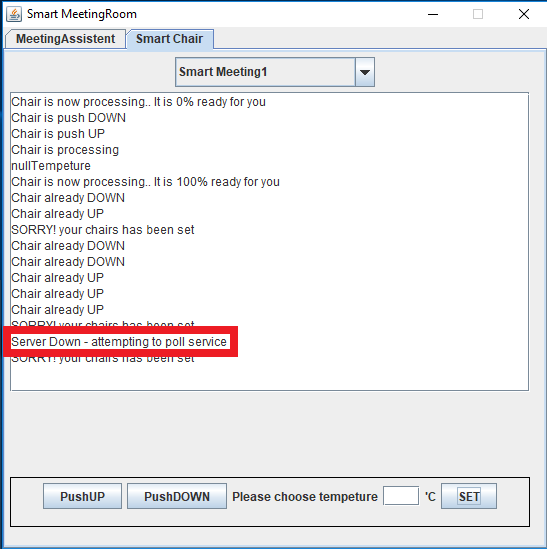


# **Error Handling**

Bellow we have screenshots of the code we have used for ErrorHandling as well as the error messages displayed in the console.

ErrorHandling message, code and display in the console.



# **Conclusion**

This project helped to develop Java skills and to have a better understanding of Internet of Things (IOT)

In smart environments. This was a simple introduction but with complex services and some complexity in the services communication. This open our minds to develop something more complex in the future as we got very interested with IOT as that’s the future.