National College of Ireland

SMART SURVEILLANCE/SECURITY

Higher Diploma in Science in Computing
Distributed Systems (HDCSDEV_INT)
March 2023

Flávio Pereira de Carvalho
x 21189056

Contents

1 Introduction

The goal of this project is to offer a specialized and efficient Smart Surveillance/Security service by utilizing remote procedure call (RPC) methods to control and monitor access for different types of places.

To be able to monitor access from a location, we need to divide the location into areas and define a level of access for each of them. From this, we need to create users with names and IDs, which will then have the release of access to the areas defined by the client.

With this efficient system, it will be possible to control the access of several places, being possible to restrict access in places that need more security and attention, lock and unlock doors remotely, as well as reporting unauthorized access attempts and even invasions. To ensure greater efficiency, the system will report invasions to the site manager and can send a message for help to the local police, being a great differential for this service and making access control more efficient and secure.

2. Service 1: ListControl

The list control service will allow the client to add, remove, and edit a user access list. Through this service, he can insert and remove users from the access list, as well as edit the level of access permission of an already registered user.

2.1 Methods

2.1.1 RPC Method 1: CreateUser

This method allows the client to add a user and set their access level on-premises. Therefore, the customer can restrict the user's access to certain areas, having greater security in place.

rpc CreateUser (CreateUserRequest) returns (ConfirmationCreateUserResponse){}

2.1.2 RPC Method 2: DeleteUser

This method allows the customer to remove a user who was previously entered, such as an employee who no longer works in the company.

rpc DeleteUser (DeleteUserRequest) returns (ConfirmationDeleteUserResponse){}

2.1.3 RPC Method 3: ModifyUserAccessLevel

Modifying the access level is essential for a surveillance/security system, as the client may need to increase or decrease the access of a determined user on site. This service is called bidirectional streaming RPC.

rpc ModifyUserAccessLevel (ModifyUserAccessLevelRequest) returns (ConfirmationModifyUserAccessLevelResponse){}

3 Service 2: AccessMonitoring

AccessMonitoring is a service that allows the customer to check the level of access of a user to allow or reject the entry of an area, in addition to allowing the activation of audible alarms in case of identification of invasion or unauthorized entry into a location.

3.1 Methods

3.1.1 RPC Method 1: CheckUsersAccessPermission

The access check is the best way to avoid access to areas not allowed and with this method the client can check if a user has permission to access a particular place and thus prevent or allow the entry of it.

rpc BidiCheckUsersAccessPermission (stream CheckUsersAccessPermissionRequest) returns (stream CheckUsersAccessPermissionResponse){}

3.1.2 RPC Method 2: SetPerimeterLevel

In this method, the client can set the permission level in certain areas, which will then be used to determine in which area a user has access permission.

rpc SetPerimeterLevel (SetPerimeterLevelRequest) returns (SetPerimeterLevel Response){}

3.1.3 RPC Method 3: RingAudibleAlarm

In case of unauthorized access identification or intrusion, this method will allow the customer to activate a system of audible alarms, alerting them about an invasion in a certain area.

rpc RingAudibleAlarm (RingAudibleAlarmRequest) returns (RingAudibleAlarm Response){}

4 Service 3: MonitoringCenter

With this modern service, the customer can receive alert messages of invasion or unauthorized access to a certain area, lock and unlock doors remotely and still send a request for help to the police.

4.1.1 RPC Method 1: AlertMessages

In case of unauthorized access identification or intrusion, this method will allow the customer to activate a system of audible alarms, alerting them about an invasion in a certain area.

rpc AlertMessages (AlertMessagesRequest) returns (stream AlertMessagesResponse){}

4.1.2 RPC Method 2: ControlDoorsRemotely

Through this method, the client can lock or unlock doors and gates remotely. This can prevent someone from accessing without permission or even prevent the exit of an attacker who was detected on the spot.

rpc BidiControlDoorsRemotely (stream ControlDoorsRemotely Request) returns (stream ControlDoorsRemotelyResponse){}

4.1.3 RPC Method 3: InvasionPoliceReport

This method will be responsible for sending a request for help to the police in case of invasion or unauthorized access to a certain place. Helping to preserve the safety of the site.

rpc DeleteUser (InvasionPoliceReportRequest) returns (stream ConfirmationInvasionPoliceReportResponse){}

5 RPC types used in this project's implementation

The three RPC types used in this project's implementation are:

5.1.1 Unary RPCs:

2.1.1 RPC Method 1: CreateUser

2.1.2 RPC Method 2: DeleteUser

2.1.3 RPC Method 3: ModifyUserAccessLevel

3.1.2 RPC Method 2: SetPerimeterLevel

3.1.3 RPC Method 3: RingAudibleAlarm

5.1.2 Server Streaming RPCs:

4.1.3 RPC Method 3: InvasionPoliceReport

4.1.1 RPC Method 1: AlertMessages

5.1.3 Bidirectional Streaming RPCs:

3.1.1 RPC Method 1: CheckUsersAccessPermission

4.1.2 RPC Method 2: ControlDoorsRemotely