Project Proposal

Piotr Kuderski

Student No.: 22145630

<https://github.com/pkud2022/distributedSystems>

## OVERVIEW

*This document is submitted as an interim submission of CA project of the Distributed Systems module on the Higher Diploma in Computing specializing in Software Development delivered by Yasantha Samarawickrama.*

# Domain description

# You should best describe the overall purpose of the service, explain the functionalities within each service and overall contribution of the service to the application

# [200 words]

# The purpose of the project is to add 3 functionalities to a Smart Office environment, which will contribute to a larger (unspecified) application developed for these environments.

# The functionalities that are intended to be developed are:

# Switching on/off lights in rooms & corridors - client side streaming

# This functionality allows for lights being switched on and off upon the user’s request.

# Sending a subscription-based information/news to user- server side streaming

# This functionality allows for a user to subscribe to the server, which then will provide the user with information/newsletter/memos in a given topic. These for example could be news, canteen lunch menu for the day, changes to office-wide policies, etc.

# Internal office chat - bidirectional streaming where workers can communicate by way of chat.

# This functionality allows for users to communicate in chatrooms removing the need for sending emails or memos to each other – this could be a less formal way of internal communication.

# At this early stage of development it is intended that each service will be in a separate proto file in order to provide better modularity and sooner delivery of features to the customer.

# A simple user interface will be provided via a website.

# Service definition and RPC

# You should explain in detail, with example the request and response for each functionality within the service. Explain in detail the parameters

# [300 words]

# Switching on/off lights in rooms & corridors

# “LightControlService” – the physical rooms and corridors will have to be referred to by an identifying reference, for example “roomId”, “corridorID”.

|  |  |  |
| --- | --- | --- |
| Request | Response | Comments |
| LightControlRequest | LightControlResponse | Request might need a few fields like: roomId/corridorId, turnOn, turnOff.Response might return: “success” or “failed”. |
| LightStatusRequest | LightStatusResponse | Request’s potential fields: roomId/corridorID.Response’s potential fields: “on”, “off”. |
| LightControlAllRequest | LightControlAllResponse | This request might switch on/off all lights linked, with potential fields: turnOn, turnoff. Response: “success” or “failed”. |

# Sending a subscription-based information/news to user

# A user needs to have ability to subscribe to and unsubscribe from the service.

|  |  |  |
| --- | --- | --- |
| Request | Response | Comments |
| SubscribeRequest | NewsResponse | Request will have a boolean field.Response might return: “success” or “failed”. |
| UnsubscribeRequest | UnsubscribeResponse | Request will have a boolean field.Response’s potential fields: “on”, “off”. |
|  | MissiveResponse | The server side will continue sending messages whilst the SubscirbeRequest’s field is true. |

# Internal office chat

# This is a simple functionality where the message sent by one user is then relayed by the server to the reminder of users.

|  |  |  |
| --- | --- | --- |
| Request | Response | Comments |
| SendMessageRequest | SendMessageResponse | Request will require following fields: “userName" (name of the user sending the message), “message”(text being sent).Response is the same as Request in this instance, as it needs to display the message to other users. |