CSE 3105/ CSE 3137

OBJECT ORIENTED ANALYSIS AND DESIGN

FALL 2020

COURSE PROJECT: *<Media Application Browser>*

System Design Document

Group 13

Muratcan Erek – 180316042

Emre Çakmak– 180316021

İsmet Dirilen– 180316031

Ali Osman Beker – 18031510

Mehmet Çağlar – 16031026

17 January 2021

Table of Contents

1 Introduction

It is an application that works with a cloud service logic that does not require an additional download within the application, which is constantly connected to the internet due to the project structure. We will design the application that serves by connecting to the servers it has, with the necessary architectural style. The aim here is to explain the working and service logic of the application as clearly as possible and to open the functions in this scheme.

1.1 Purpose of the System

It is aimed to design an easier and better quality music application platform for increasing music listeners.

1.2 Design goals

Reliability: My music app should be reliable with Original music.

Modifiability: My music should be easily replaceable by different playlists.

Security: Playlists I create should be secure so my Playlists must not be allowed to be edited by other users Current Software Architecture.

Performance: My Music should allowed to run on two different platforms simultaneously, and thanks to special data structures, the transition between music is fast.

Usability: My Music should allows multiple users to access an account with user request, and my Music should supports multi-platform(mobil - web vb).

Supportability: It should be Allowing new music and new lists to be added to my Music without disturbing the existing my Music layout

2 Current Software Architecture

Our app includes a mod, top, playlist that takes into account usability, ease of access, etc. Spotify has these software architectures, just like our application. Another feature of our application is that it can suggest songs similar to the type of songs we listen to. This architecture appeals to users' tastes. We come across this architecture in daily life on YouTube

3 Proposed Software Architecture

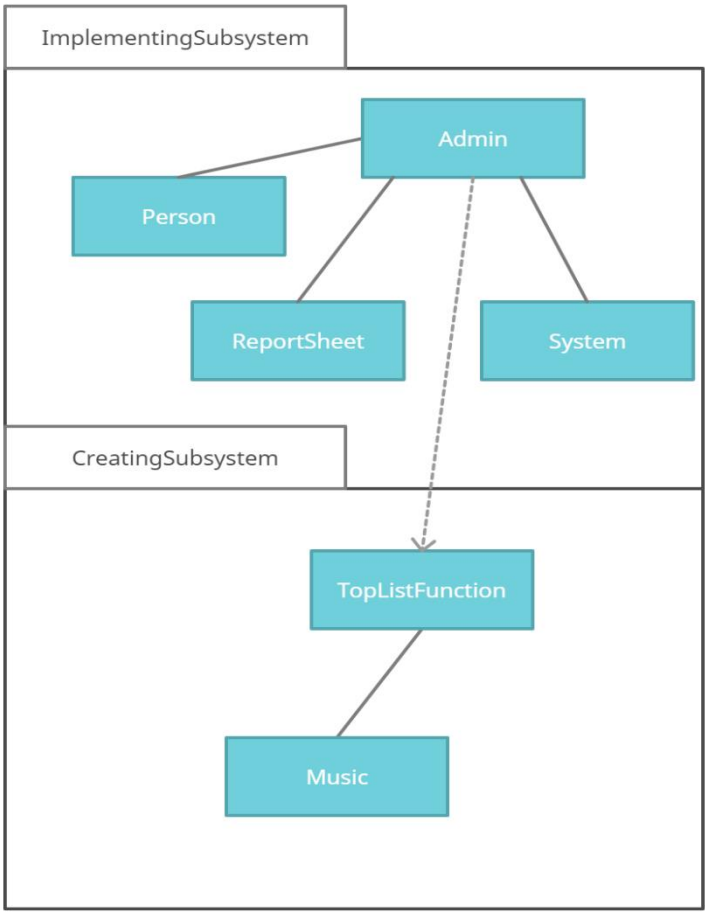
3.1 Subsystem decomposition

Portability: can be integrated with other operating system

Maintainability:
When the system is turned off, the user can continue from where she left off

Modifiability:

It can change the interface according to user request



LAYER 2

LAYER 1

CLOSED ARCHITECTURE OPAQUE LAYERING

ImplementingSubsystem The implementinsubsystem is obliged to evaluate the messages on the reportsheet and forward it to the toplist function of the creatingsubsystem.

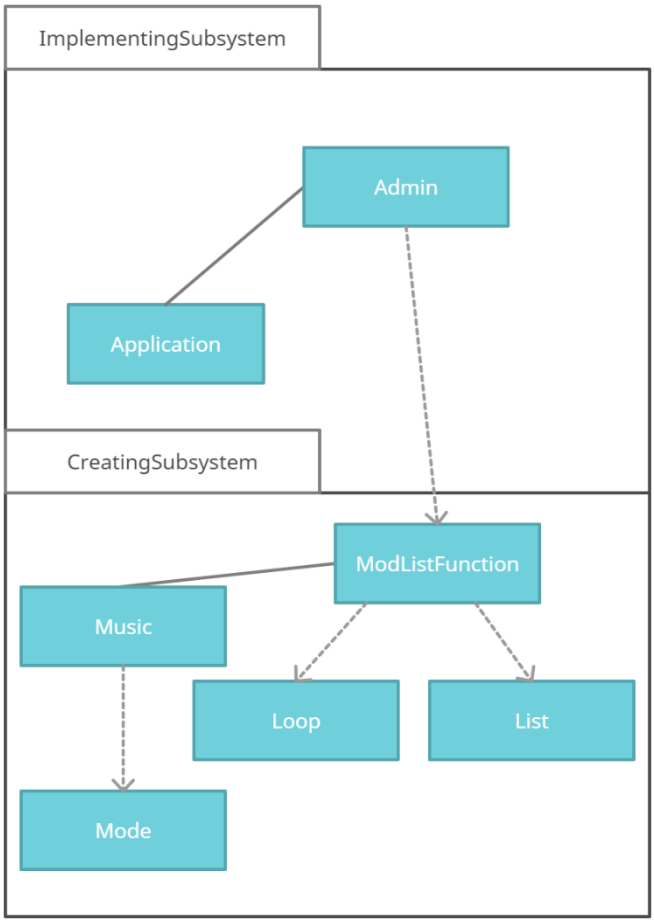
CreatingSubsystem Creatingsubsyem, fulfilling the request received by implementing subsystem, responsible for creating, and also creates its architecture by connecting music with sequences.

Portability: can be integrated with other operating system

Maintainability:
When the system is turned off, the user can continue from where she left off

Modifiability:

It can change the interface according to user request



LAYER 2

LAYER 1

CLOSED ARCHITECTURE OPAQUE LAYERING

ImplementingSubsystem The implementing subsystem responsible for updating and stabilizing the application also transmits to evaluate report via the link between creatingsubsystems.

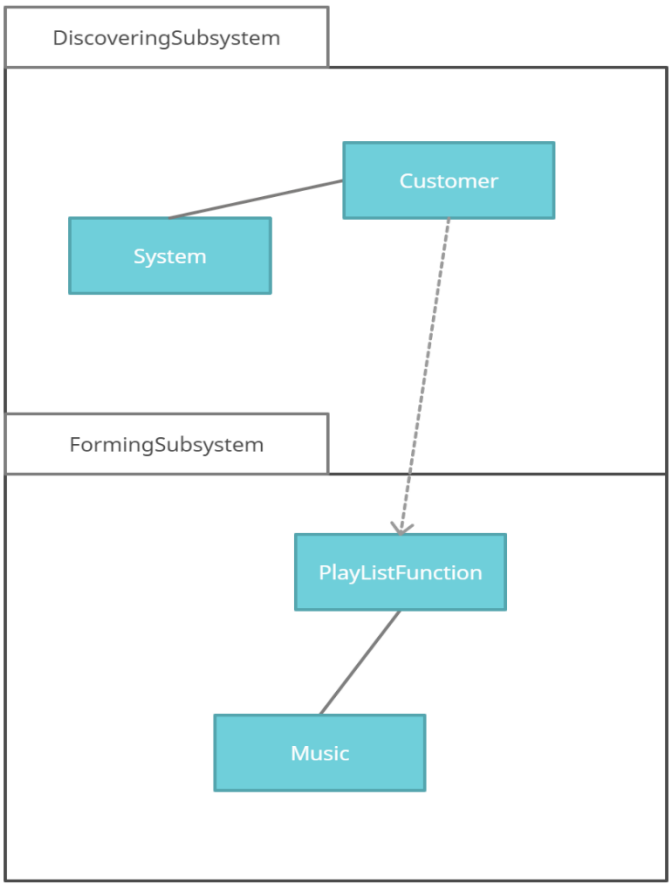
CreatingSubsytem Creatingsubsyem is responsible for creating the requests by fulfilling the requests by the application subsystem and combining the same type of music with sequences to create its architecture.

Portability: can be integrated with other operating system

Maintainability:
When the system is turned off, the user can continue from where she left off

Modifiability:

It can change the interface according to user request



LAYER 2

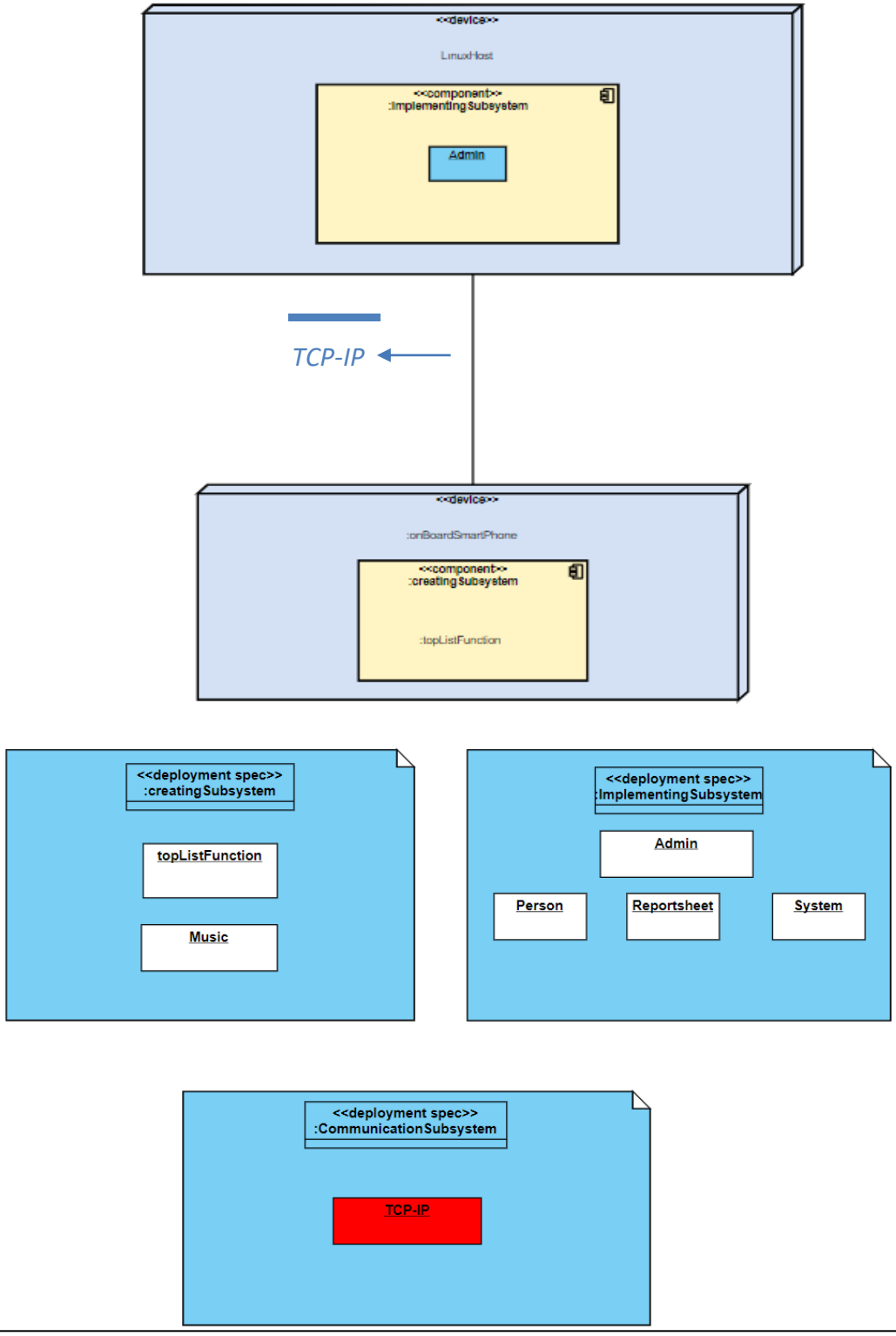
LAYER 1

CLOSED ARCHITECTURE OPAQUE LAYERING

DiscoveringSubsystem DiscoveringSubsystem is responsible for the discovery process by navigating the system and also directs the playlist function in the formingsubsystem

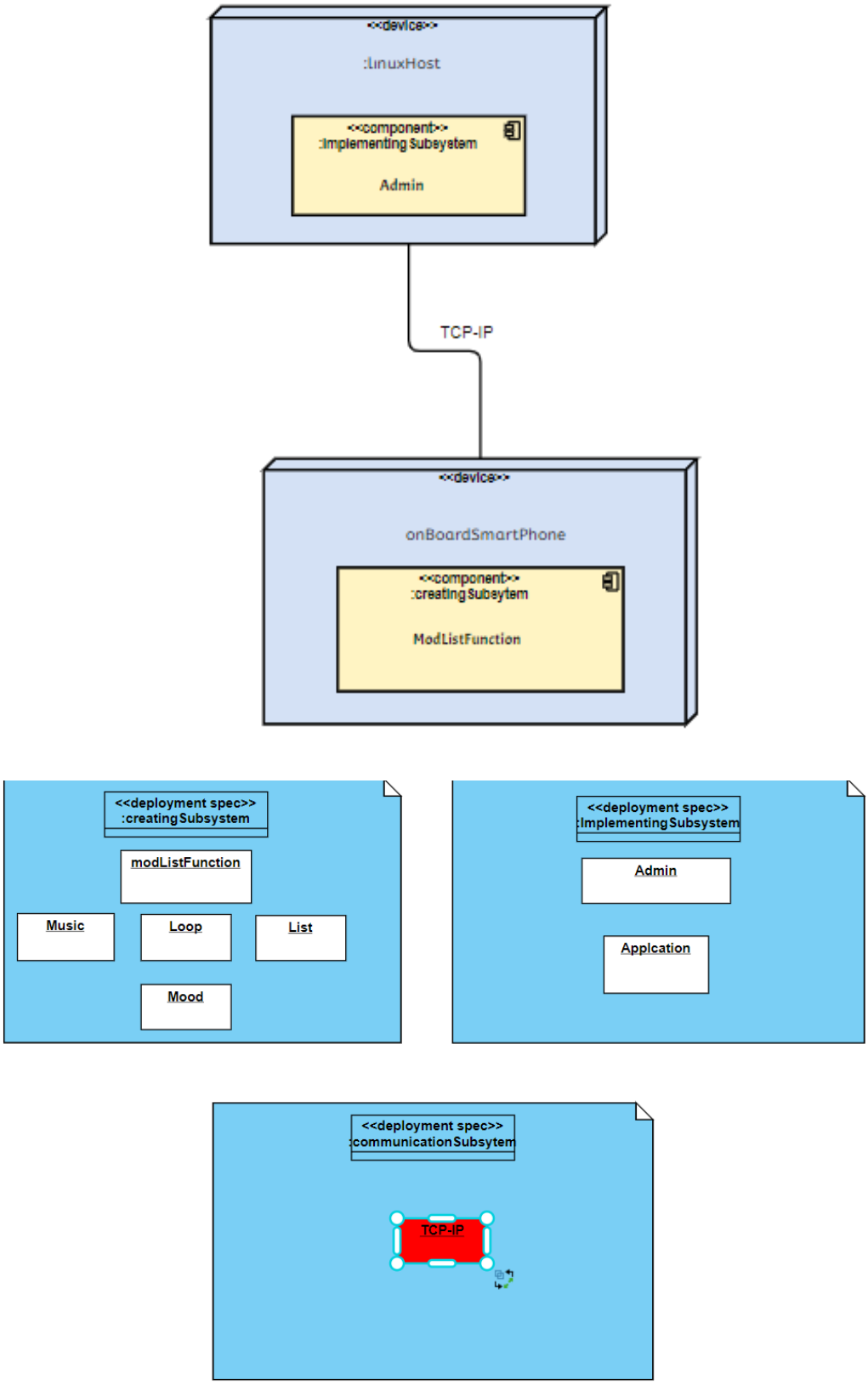
FormingSubsystem Formingsubsytem is responsible for shaping the music series by fulfilling the request of discoveringsubsystem.

3.2 Hardware/software mapping



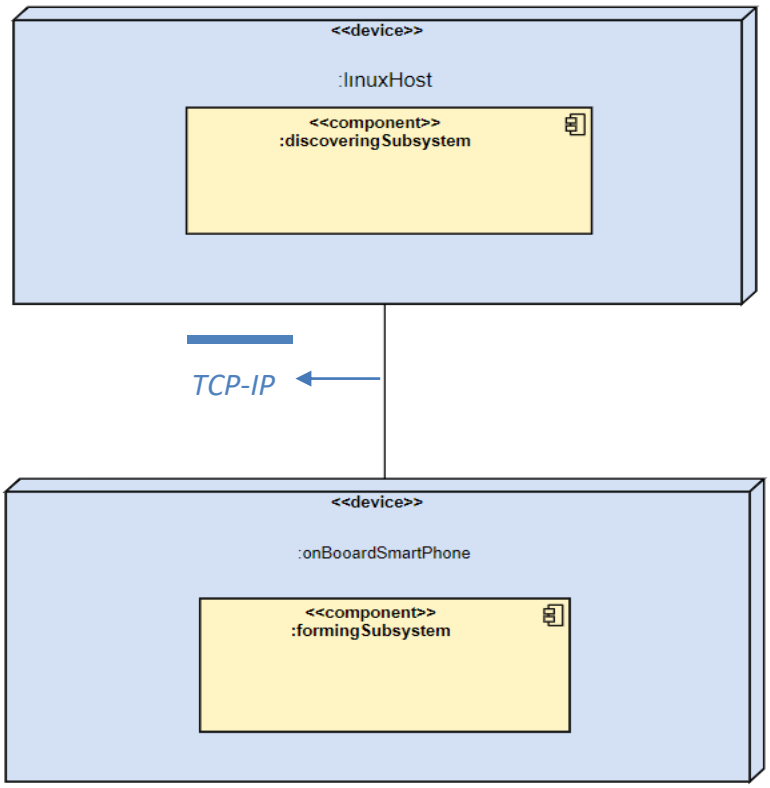
***communicationSubsytem*:** Communication subsystem is responsible for moving objects from implementing subsystem to creating subsystem

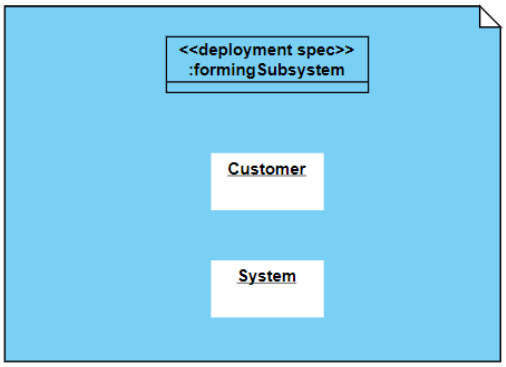
TCP-IP: TCP / IP Reference Model. Application Layer: Communication between processes and applications on different servers is provided at the application layer.

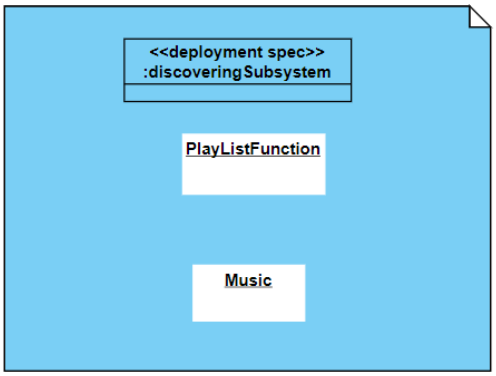


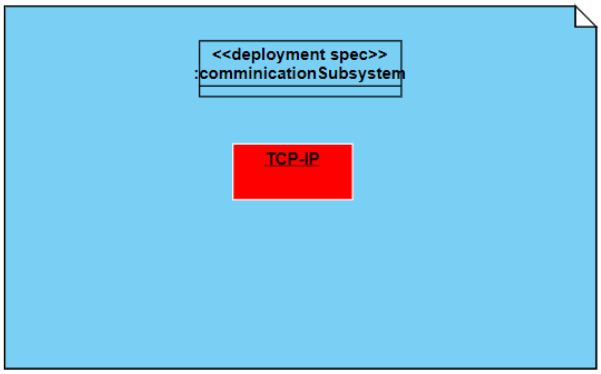
***communicationSubsytem*:** Communicationsubsystem is responsible for moving objects from implementing subsystem to creating subsystem

TCP-IP: TCP / IP Reference Model. Application Layer: Communication between processes and applications on different servers is provided at the application layer.







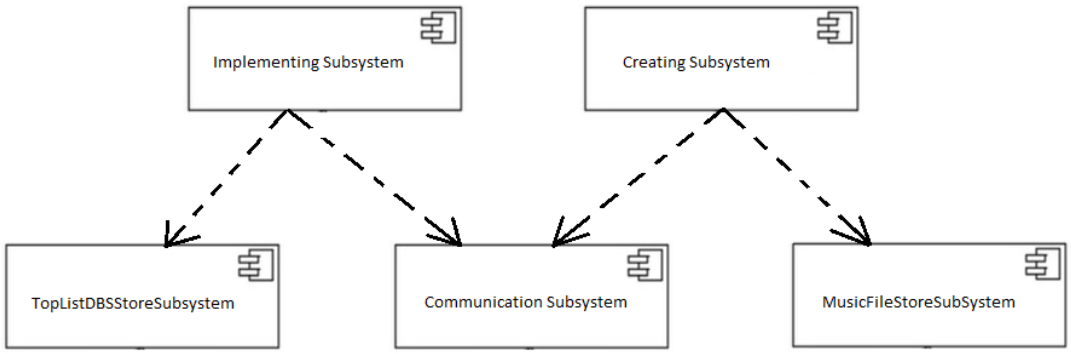


***communicationSubsytem*:** Communication subsystem is responsible for moving objects from discovering subsystem to forming subsystem

TCP-IP: TCP / IP Reference Model. Application Layer: Communication between processes and applications on different servers is provided at the application layer.

3.3 Persistent data management

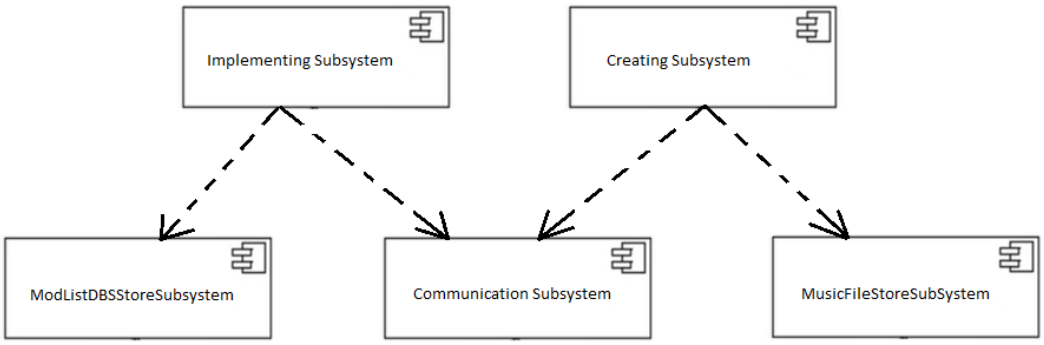
TOPLİST



MusicFileStoreSubSystem MusicFileStoreSubSystem is responsible for listing the most selected music in the application, also this function is responsible for storing the music in a way that is listed only when the application is closed, this subsystem only supports sequential storage and loading of all music.

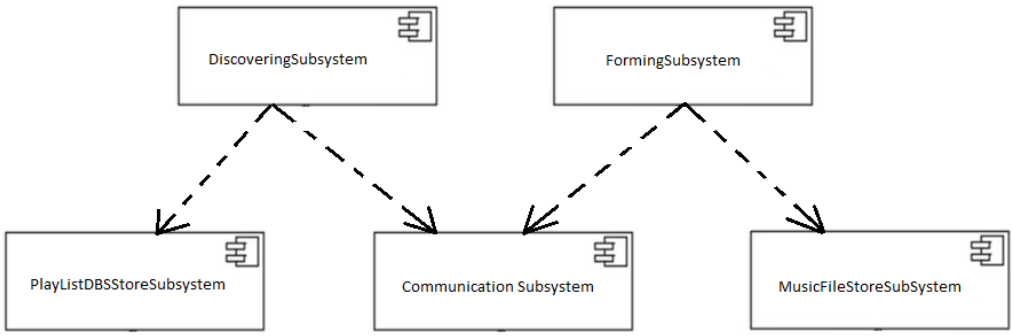
TopListDBSStoreSubsystem Stores for ımplementingsubsystem in a database and supports multiple access in this subsystem

MODLİST



MusicFileStoreSubSystem MusicFileStoreSubSystem is responsible for listing the music listened to in the application by mode, also this function is responsible for storing the music to be listed when the application is closed

ModListDBSStoreSubSystem Stores Modlists for ImplementingSubSystem in a database and this subsystem supports multiple access



MusicFileStoreSubSystem MusicFileStoreSubSystem is is responsible for listing the music selected in the application, also this function is responsible for storing the music to be listed when the application is closed.this subsystem supports sequential storage and loading of all music.

PlayListDBSStoreSubsSystem Stores Playlist-selected music for DiscoveringSubsystem in a database, and this subsystem supports multiple Access.

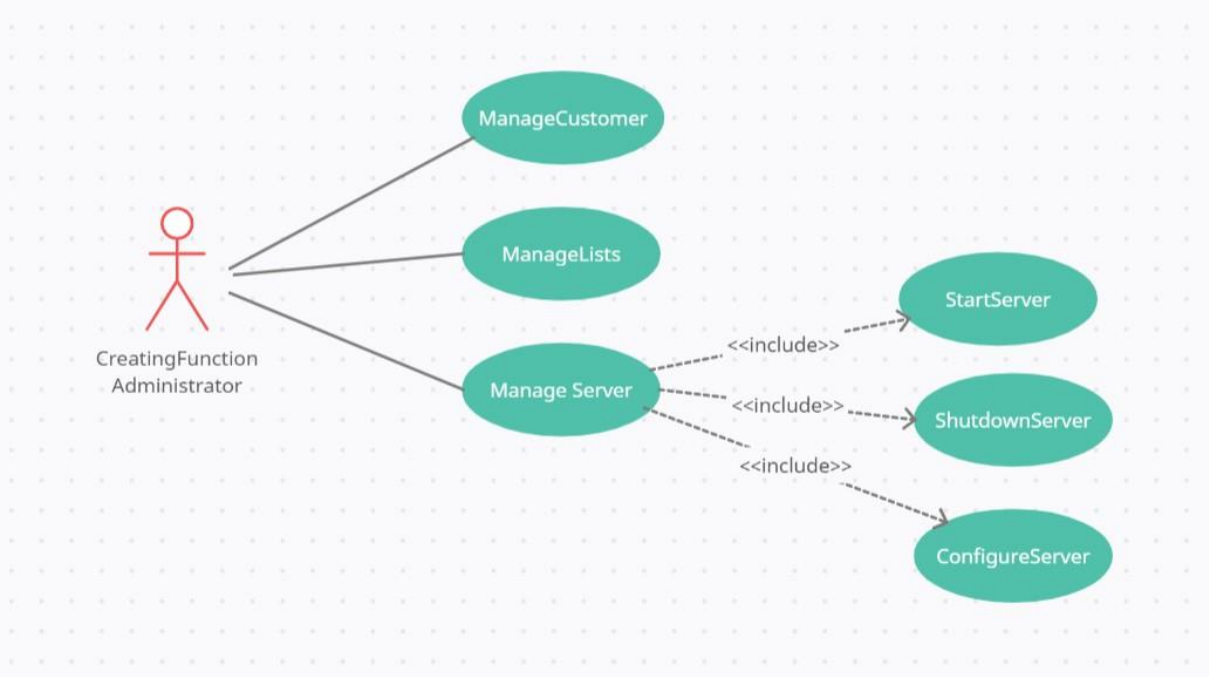
3.4 Access control and security

	PLAYLIST	MODLIST	TOPLIST	REPORT
ADMIN	<<include>> <<extend>> Create() Find()	<<includes>> Names() Create() Run() Add()	<<extend>> Create(); Run(); Titles	<<include>> <<extend>> View();
USERS	Select() Add() Start()	Collect() Add() Start()	List() Put()	

Users Authentication is required when registering and logging in to our application. Username and password are required to login to your application. In the username and password section, we can use your username and e-mail address that we use in our daily lives. For ease of login, we can check the remember me section just below the password and username section, so that when we exit, we can log into our application with one click without entering our password and username.

CommucationSubsytem commucationSubsytem uses the customer associated with the lists being creating for selecting a sign and encrypting the communicaation reportsheet.

3.5 Boundary conditions



ManageCustomer: Controls Users who log in to the system

ManageLists: Provides manangement of the list in the system(adding and removing lists)

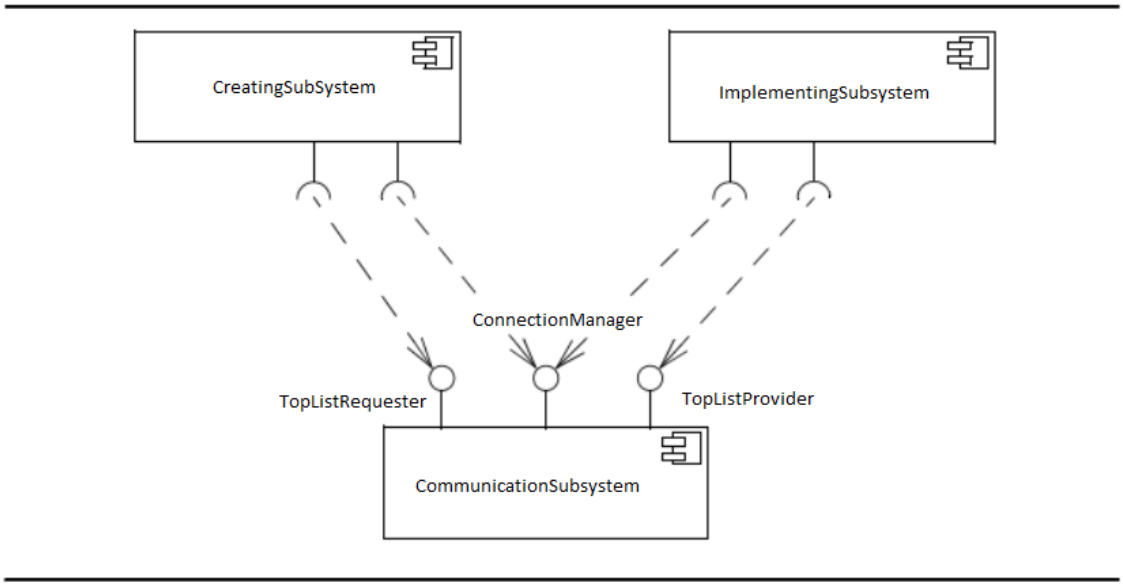
ManageServer: Performs the opening,closing and editing of the server

StartServer: Responsible for starting the server

ShutdownServer: Responsible for proper shutdown of the server

ConfigureServer: It performs the shaping process on the server

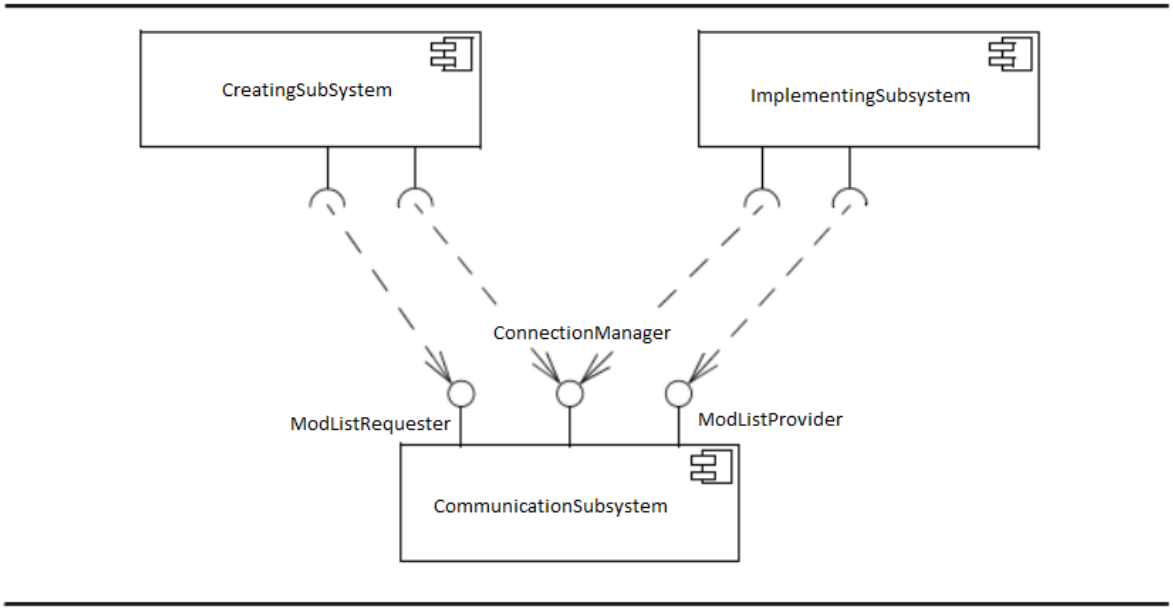
4 Subsystem Services



TopListProvider TopListProvider is responsible for present the evaluated report to the CommunicationSubsystem

TopListRequester TopListRequester is responsible for providing the request service from CommunicationSubsystem

CommunicationSubsystem Provides connection manager service to creatingsubsystem and implementingsubsystems



ModListProvider ModListProvider is responsible for present the evaluated report to the CommunicationSubsystem

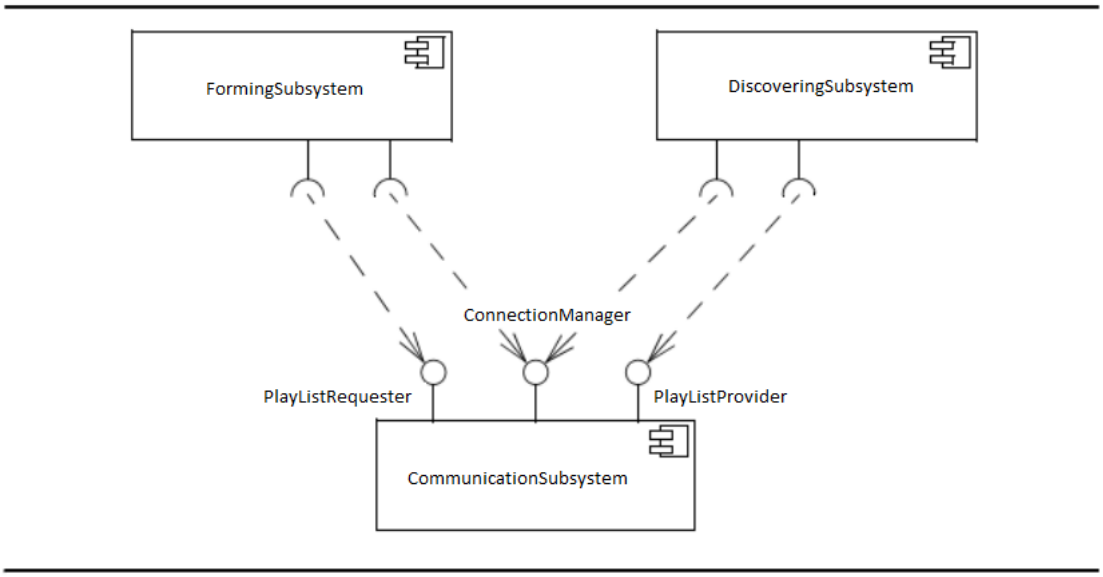
---

ModListRequester ModListRequester is responsible for providing the request service from CommunicationSubsystem

---

CommunicationSubsystem Provides connection manager service to creatingsubsystem and implementingsubsystems

---



PlayListProvider PlayListProvider is responsible to directs the function in DiscoveringSubsytem to communicationSubsystem

---

PlayListRequester PlayListRequester is responsible for providing the request service from CommunicationSubsystem

---

CommunicationSubsystem Provides connection manager service to creatingsubsystem and implementingsubsystems

---

5 Glossary

TCP/IP It is one of the transport layer protocols of the protocol suite.

Spec It is meaning specification

6 References

- Lecture Videos(on Microsoft Teams)

- Slides(on Microsoft Teams)