

Detail Design

TEAM 2 – NOT YET

Sure-Park Reservation System for Driver

Sure-Park Manager System for Attendant

Revision History

Version	Date	Description	Author
0.1	2016.06.23	Design reconstruction	Byounghoon (Beney) Kim

Terms and acronyms

Driver package: SurePark reservation system module for drivers.

Attendant package: SurePark manager system module for facility attendant.

References

[1] [TEAM2_DOC_03] Architecture Document

1. INTRODUCTION

1.1 SCOPE OF DOCUMENT

This document describes SurePark reservation system and manager which is designed by Not Yet Team.

1.2 PURPOSE OF DOCUMENT

The purpose of this document are as below.

- describing how SurePark reservation system and manager system are structured
- describing how SurePark reservation system and manager system works.

1.3 DESIGN RECONSTRUCTION

After architectural design, architect design the overall architecture and interface between each network elements, called sub-system. Also, these two sub-system is assigned to me (who is downstream designer and coding engineer). Actually, I implemented it without documenting detail design. Hence, after implementing done, I reconstruct it.

2. STATIC PERSPECTIVE

As described in [1] [TEAM2_DOC_03] Architecture Document, SurePark reservation system and manager system use Communication Manager - implemented as communication libraries (libcomm and libchannel-biz). They hide all network related complex things and provide abstract channels for business logic, such as login, making reservation ..., so on. Figure 1. shows that attendant and driver packages use libchannel_biz package.

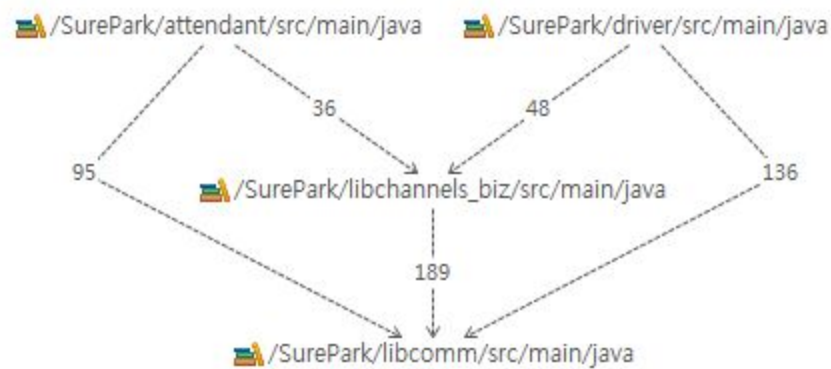


Figure 1. Module dependency (Static Perspective, generated by STAN)

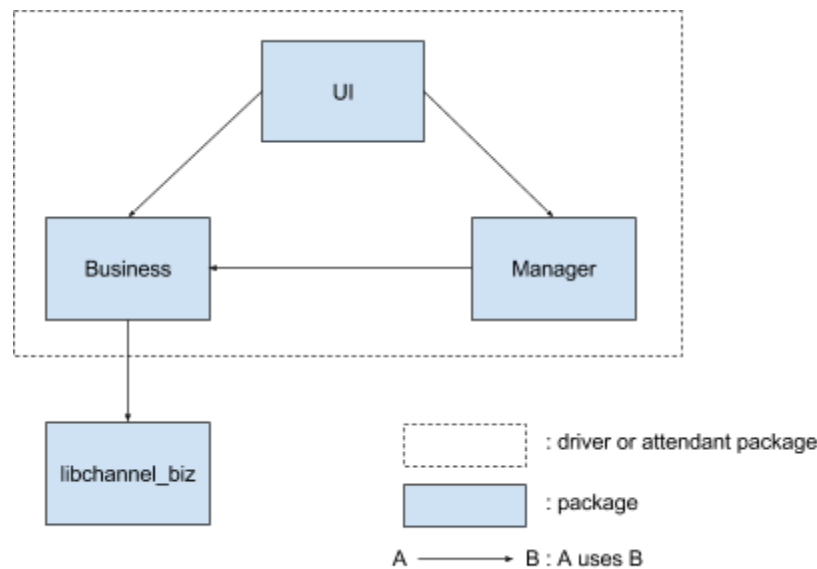


Figure 2. Module dependency (Static Perspective, generated by STAN)

Figure 2 shows decomposition of driver or attendant package. They have exactly same structure. First, business package interacts with libcomm_biz package. Hence, it is responsible for making channels for

specific purpose and communicating with other network elements. The response from other network element is passed to UI package using callback object. For this, UI package registers (use relationship) callback object for each request or notifying message. UI package is responsible for UI interface for driver and attendant. Manager package is comprised of NetworkManager, ScreenManager, SessionManager and TaskManager.

- NetworkManager is responsible for network connection handling.
- ScreenManager is responsible for screen switching.
- SessionManager is responsible for storing session information such as login result, reversion information, ..., so on.
- TaskManager is a job scheduler to process requests which gets from UI package.

3. DYNAMIC PERSPECTIVE

Figure 3 shows how SurePark reservation system and manager system works. UI Task is a thread which receive inputs from user interface. If there is an action from driver which is need to be handled by SurePark processing system, it requests to handle it to TaskManager thread with registering callback object. Then, TaskManager create a Thread which is handle the request. For example, Login Task thread is created to login into SurePark processing system and Make Reservation Task thread is created to make a reservation with SurePark processing system. Please note that there are more Tasks which is not presented in Figure 2. Each task communicate with SurePark processing system using channel as defined interface. Then, there is response or notification message from network, each Task execute callback object's method to send the received message to UI thread. Finally, UI thread handle it and can store some useful data in SessionManager.

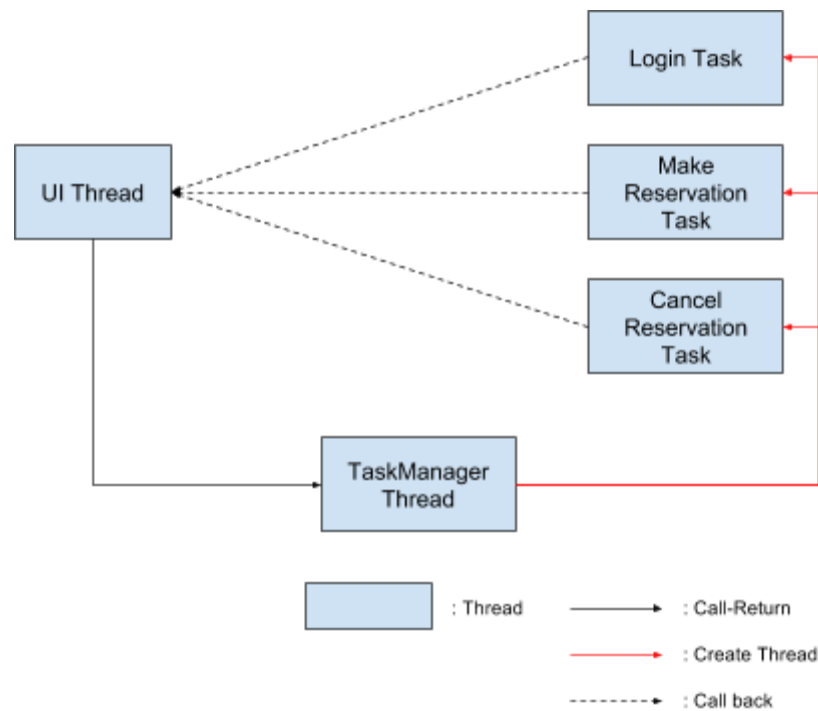


Figure 3. Module dependency (Static Perspective, generated by STAN)

4. PHYSICAL PERSPECTIVE

Physical perspective is trivial. SurePark reservation is running on a laptop. Sure manager system is also running on the other laptop. Both systems can be running on same laptop, but usually different. Both are connected to publish/subscribe broker over WiFi.