Analysis and design

5.1. Introduction

In the previous chapter I have discussed the technologies and methodologies used in solving this problem. In this chapter I will cover the analysis and design process of the new system. First I analyzed the current system and with the knowledge of current system I designed the new system for production confirmation.

5.2. Current system analysis

After the MOS system implementation, Contourline use its own production control system. That system is manual system and it's not capable in tracking the production In/Out real time or update to SAP.

System study by interviews formal / informal, how u gather requirements

5.2.1. Functional and non functional requirements of current system

- **Functional requirements**
    1. Shall be able to create module wise bundle guides
    2. Shall be able to create production orders (sewing) for respective bundle guides.
    3. Shall be able to create material requisitions for sewing orders.
    4. Shall provide facility to issue packing sewing accessories against bundle guide (Sewing Production Order)
    5. Module wise line in operation Shall be confirmed against bundle guide
    6. Module wise line out operation Shall be confirmed against bundle guide

- **Non functional requirements**
    1. System shall provide a facility to find available raw material and cut fabric stock.
    2. Shall provide the order requirement in the system.
    3. Shall provide a report on confirmed quantities in all operations against production order.

5.2.2. Check list for the current system analysis

Check list of the current system analysis is attached below. Use cases and relevant use case descriptors are attached to appendix B. Since my current system is manual system I have not created separate activity diagrams for all sob systems. But I have created use case and use case descriptors for all the sub systems in current process. That exercise helped me to gather good understanding about the current system.

	Use Case	Software Requirement	Use Case Descriptor	Activity Diagram
1	Bundle Guide Creation	1,2	B.1	Page32
2	Material Requisition /Picklist Generation	3	B.2	Page 32
3	Sewing Packing RM issue	4	B.3	Page32
4	Line In Confirmation	5	B.4	Page 32
5	Line out Confirmation	6	B.5	Page 32

Table 5.1 Check list of the current system

5.2.3. Use case diagram for the current system

I have included only the top level use case here. Please refer to the Figure 5.1. There I have identified main sub systems and actors. All the use cases for the sub systems are not included here and they are separately attached to the appendix. (Refer the appendix B.)
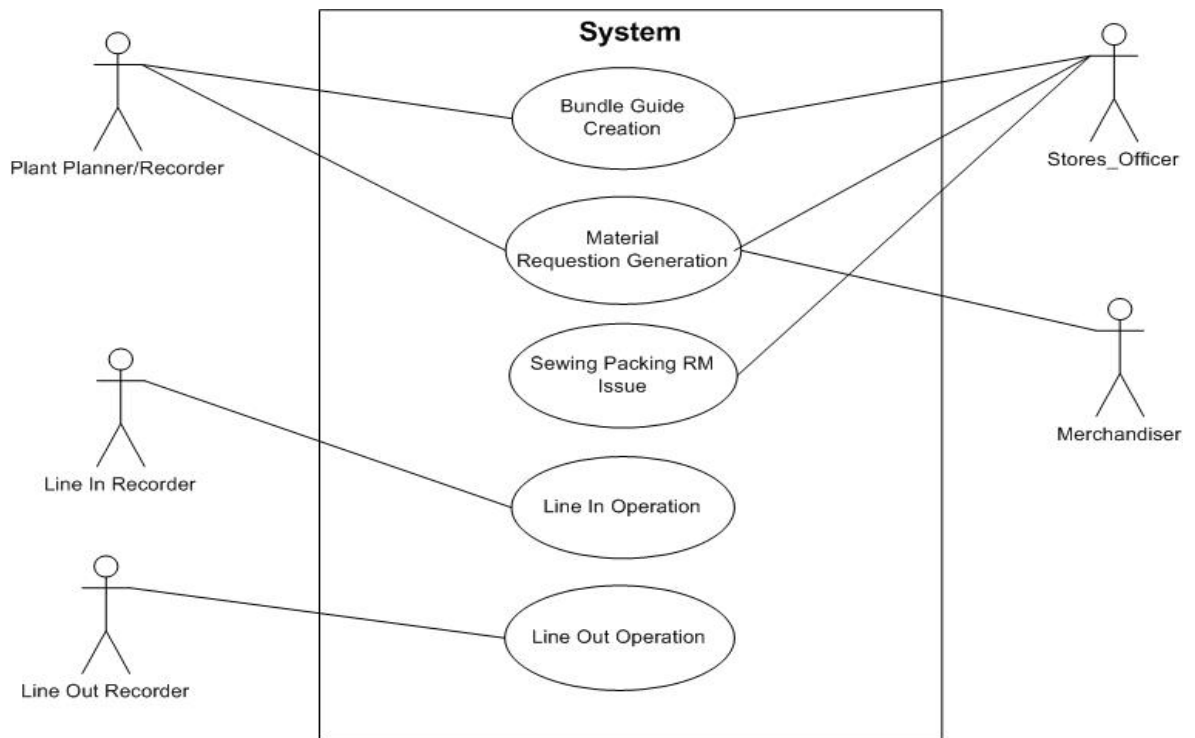
Figure 5.1 Use Case design for the current system

5.2.4. Activity diagram for the current system

Below figure 5.2 depicted the Activity diagram for the current system. It covers all the main activities currently exercising in production control process. Other than to SAP confirmations these activities are performed manually.
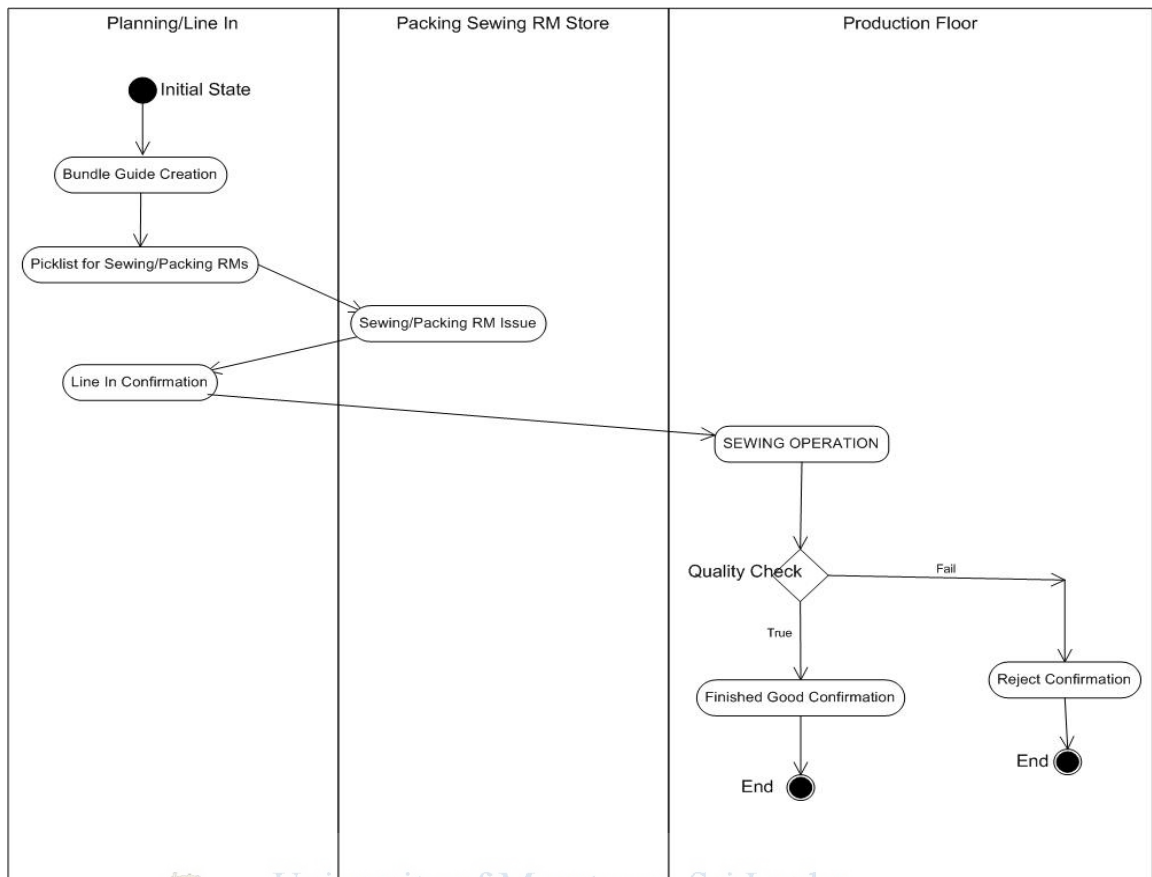
Figure 5.2 Activity diagram for current process

5.3. Proposed system design

I have use UML design methodology to design the new system. With the system knowledge I acquire from the current system analysis proposed system design is carried out.

5.3.1. Functional and non functional requirement of proposed system

- **Functional requirements**
    1. System shall provide a mechanism to capture the Production Confirmation process into SAP in Contourline.
    2. System shall be able to capture the single peace flow in line end (Line Out).
    3. System shall provide a way to create Bundle Guides according to the plant requirements.

32

4. System shall capture the Line In Bundles

5. System shall create a Material Requisition base on the Bundle Guide.

6. System shall automated Sewing RMs issue in Line In

7. System shall capture the Line Out by peaces

8. System shall use the UPC stickers for track Line Out garment wise.

9. System shall display the Line In/Out quantities in display boards.

10. System shall provide a method to update Line In/Out to SAP

11. System shall automated Packing RMs issue in Line Out confirmation.

12. System shall provide a method to enter reject items.

13. System should be able to indicate issues/errors in the process.

14. System should alarm/halted the process when an error/bug found.

- **Non functional requirements**

1. System shall provide a method to mitigate the data upload issues to SAP Server.

2. System shall record all the Line Out scanning.

3. System shall maintain a log of data uploads to SAP in Line Out.

5.3.2. New system architecture

As per the discovered functional and technical requirement the requirement portioning can be done as below figure 5.3. I have grayed the systems and sub systems that are developed out of SAP.
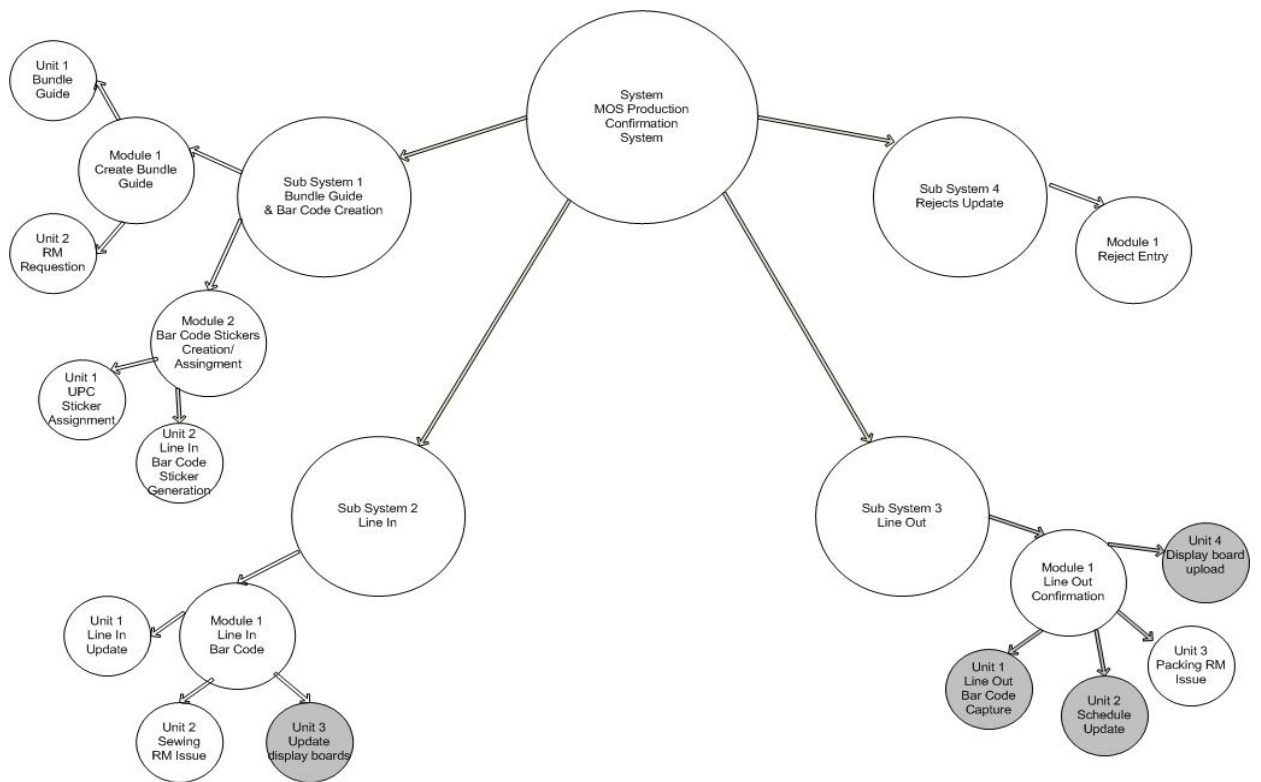
Figure 5.3 Requirement Partitioning design

5.3.3. Proposed system overview

Below in figure 5.4 I have added the proposed system architecture. Local server will be connect with SAP server via RFC calls and IP scanners and display boards are connect to local server via existing network infrastructure.
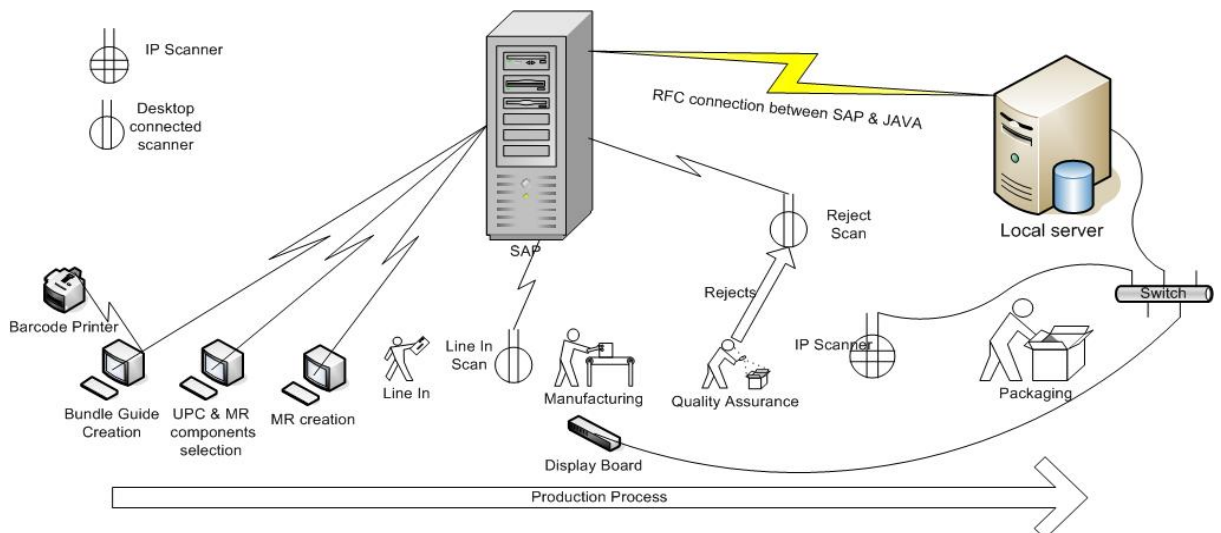

Figure 5.4 System architecture.

5.3.4. Check list of the proposed system design

Following is the proposed systems checklist for the design documents and descriptions. Table 5.2 displays the check list of the proposed system. It covers the initial design documents and design descriptions for the proposed system.

	Use Case	Software Requirement	Use Case Descriptor	Activity Diagram
1	Bundle Guide Creation	3,5	1	
2	Bar Code Creation and UPC sticker Assignment	2,8	2	
3	Line In Confirmation	4,6,9,10	3	
4	Line out Confirmation	7,8,9,10	4	
5	Rejects Confirmation	11	5	

Table 5.2 Checklist for the proposed system design documents

5.3.5. Use case diagram for the proposed system

Below figure 5.5 I have included the top level use case for the proposed system. All the use cases for the sub systems are attached in the reference under appendix C. Compare with the current systems' top level use case, two changes have been done.

In the new system sewing packing RM issue is automated within the Line In/Out confirmations. In proposed system there are two new use cases included. They call as UPC sticker introduction and Rejects update respectively. UPC sticker is used in Line Out confirmation; therefore the system should be updated with all the UPC stickers for the respective Sales Order/Line Item. In the current process Contourline doesn't count the rejects. This issue is eliminated in new system by the reject update use case.
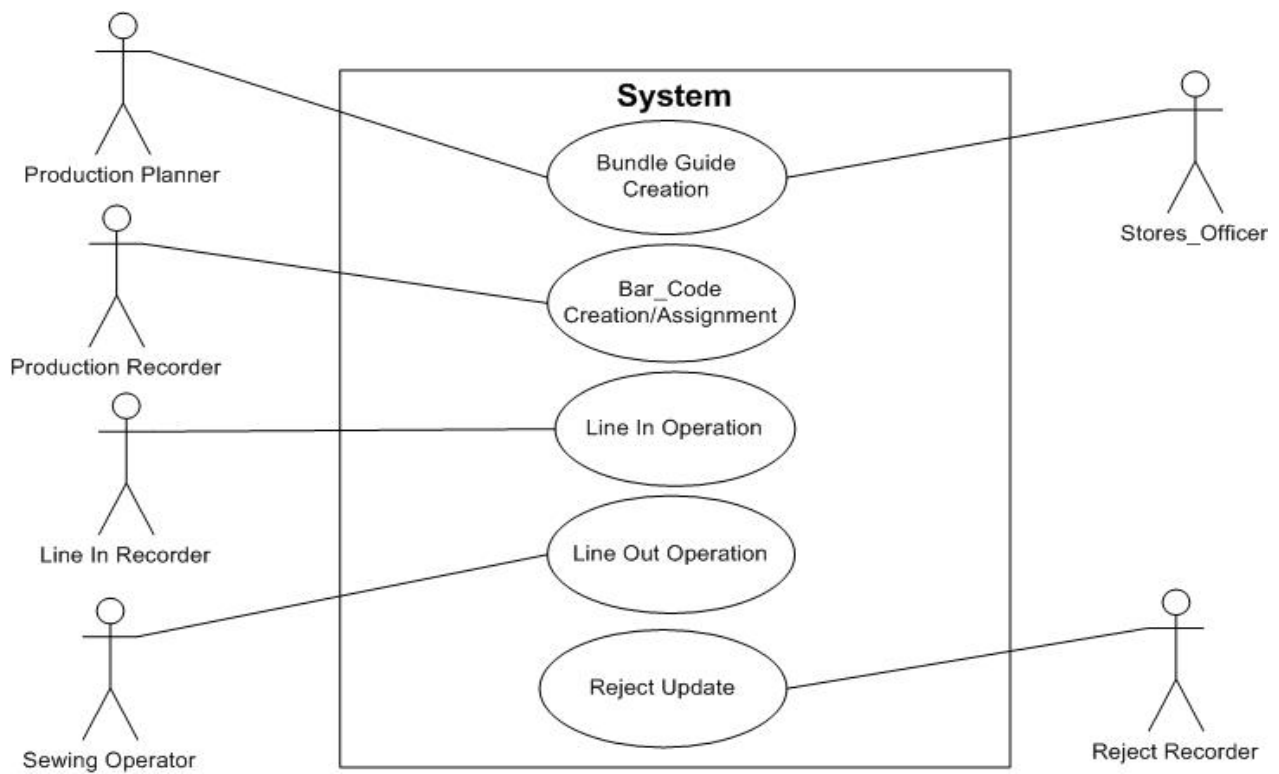
35

Figure 5.5 Top level Use case of the proposed system

5.3.6. Activity diagram for the proposed system

Proposed system top level activity diagram is attached below under figure 5.6. It's just an overview of the proposed system. I have created separate activity diagrams for all the sub systems and they are attached under appendix D.
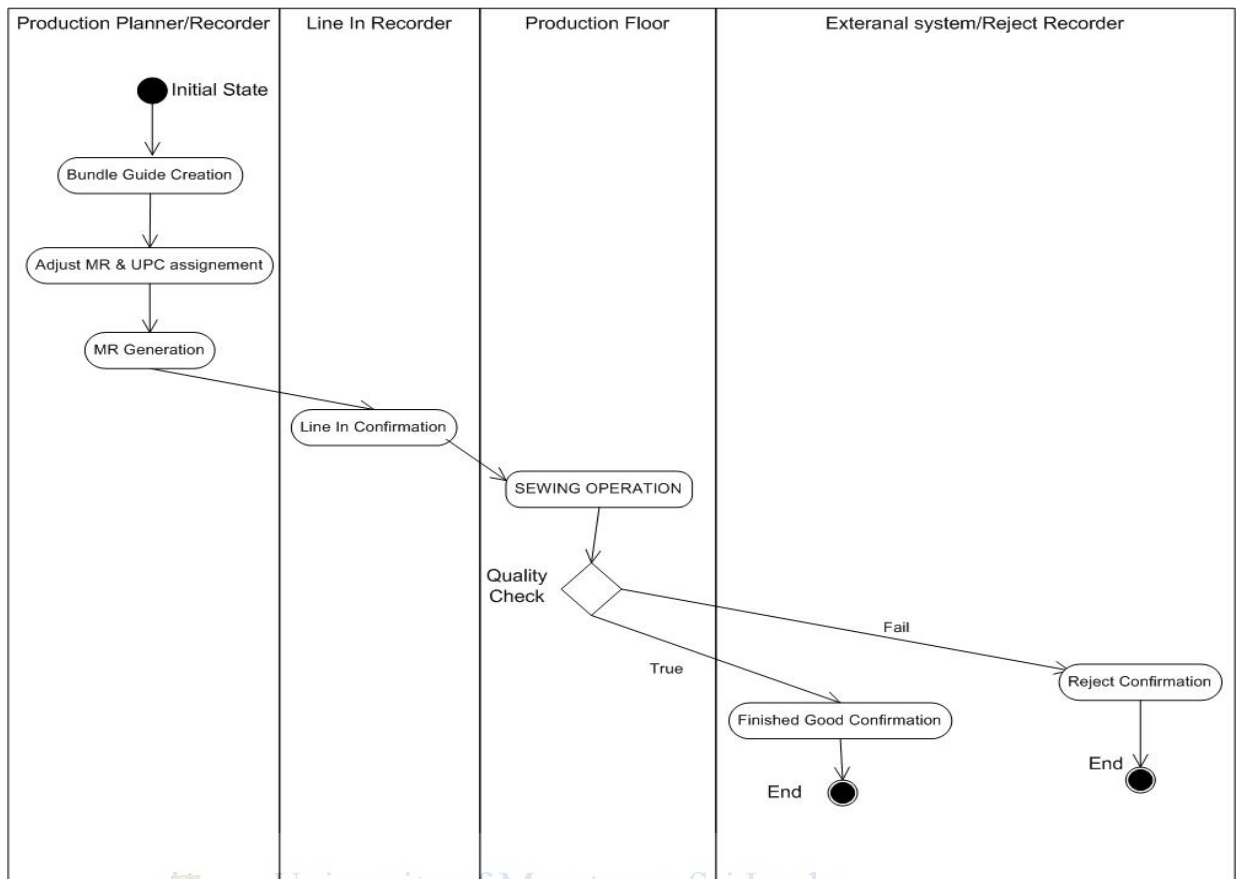
Figure 5.6 Activity diagram for the proposed system

5.3.7. Grammatical analysis

Part of the proposed system is designed in Java we need to identify entities and classes. By analyzing use case descriptors (attached to appendix C) for the proposed system we can find candidate class entities.

Nouns (Class/Attributes)	Verbs (Methods
3.5 Display Board- Line In	Update (Line In)
4.1 IP Scanner	Read Scanned data
	Write to Line Out table
4.2 Schedule Upload Process	Read entries from Line Out Table
	Process the read data
	Upload to SAP
4.5 Display Board – Line Out	Update (Line Out)

Table 5.3 Grammatical analysis

5.3.8. Sequence diagrams for the proposed system

In normal circumstances sequence diagrams supposed to be designed after the classes diagrams designed. But in this case grammatical analysis doesn't have the all the classes I needed. Because most of the Java developments are for automated processes, the detail descriptions are difficult to include in use case descriptors.

Therefore with the instructions of my supervisor I designed the sequence diagrams. By performing that activity I'm able to get the classes as well as sequence diagrams simultaneously.

I have designed use cases only for two sub systems as the Java development done only for those activities.

1. Line In process

   I have considered only activities that are capture within the Java development in the Line In sub system. Figure 5.7 shows the sequence diagram for Line In process which covers the process implement in Java.
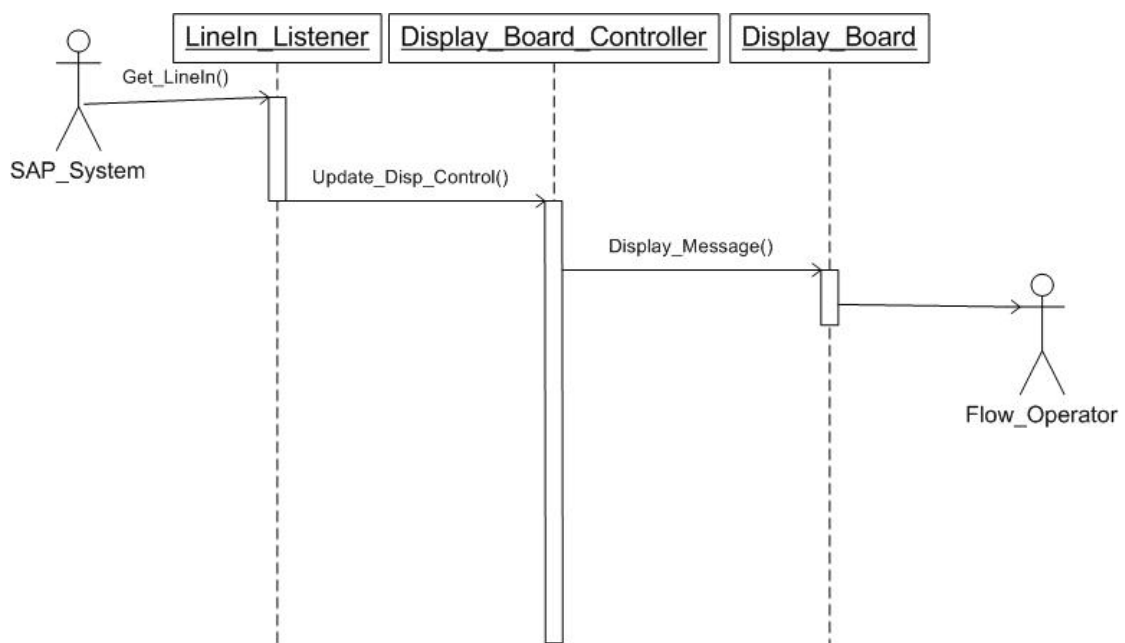

Figure 5.7 Sequence diagram for the Line In operations

2. Line Out process

Most of the areas are coved within the Java development in Line Out. SAP confirmation and RM issue automation are the units that are not cover with in this sequence diagram. Figure 5.8 displays the entity classes and there operations.
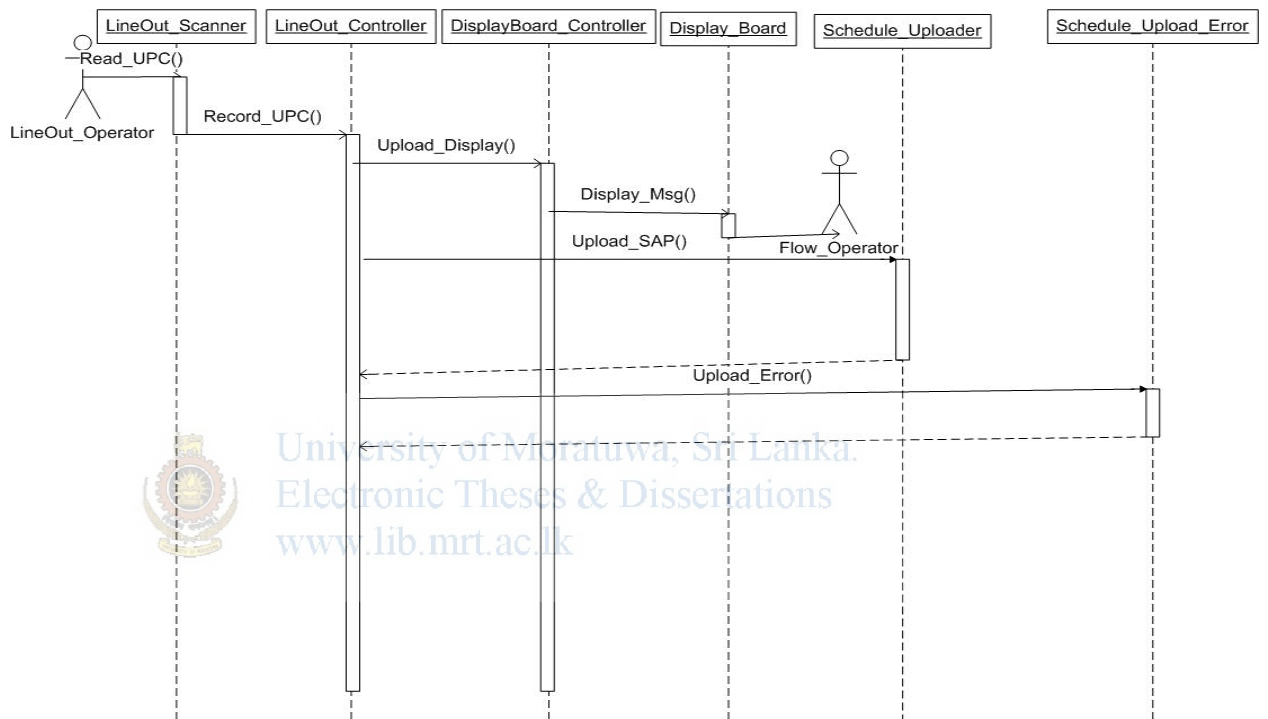
Figure 5.8 Sequence diagram for the Line Out process.

5.3.9. Class diagrams

I have attached the class diagram for the proposed system in figure 5.9. Design approach to the class diagram based on grammatical analysis and sequence diagrams.
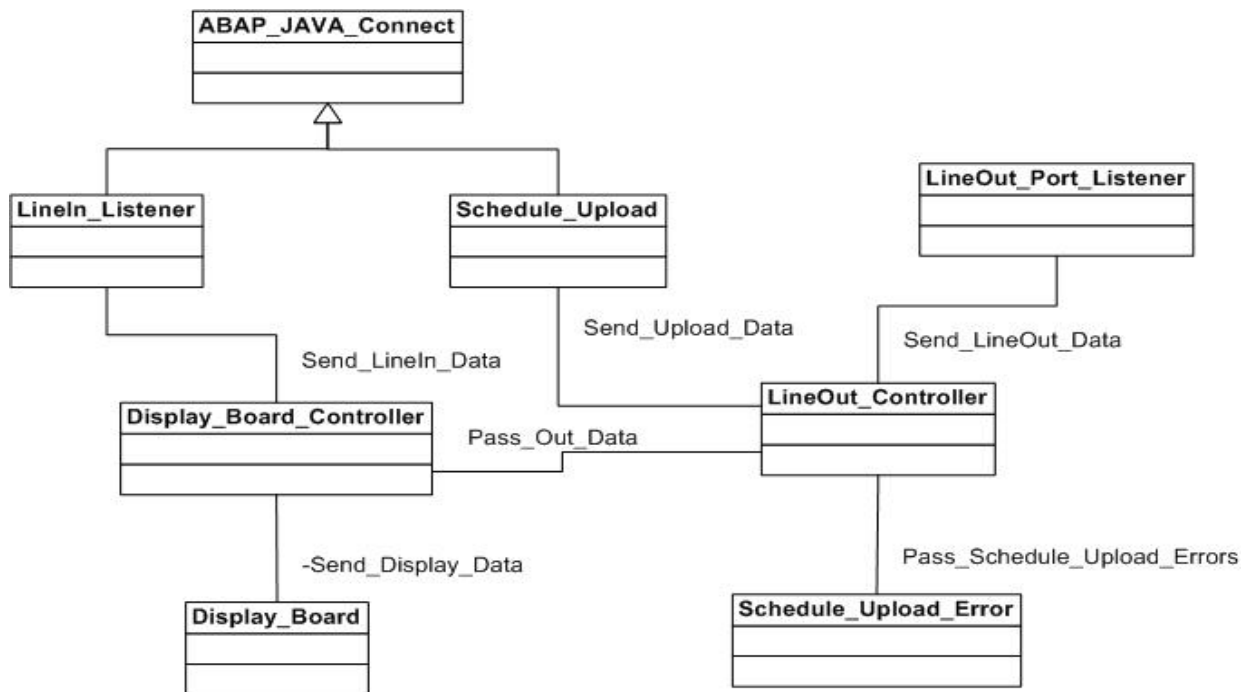
Figure 5.9 Class diagrams for the proposed system

5.3.10. Database design

As per the requirement the database and ER diagram design is carried out. Firstly I carried out the ER diagram design and then the database design.

- ER diagram

  ER diagram designing is very challengeable in this development since it deals with two separate systems. With SAP database arrangement it's difficult to find the pure entities relevant to this development. With the support of my supervisor I was able to identify the entities and design the ER diagram. Figure 5.10 contains the ER diagram for the proposed system. Entities which are in gray color are generated due to this development and others are Standard entities lies within SAP.
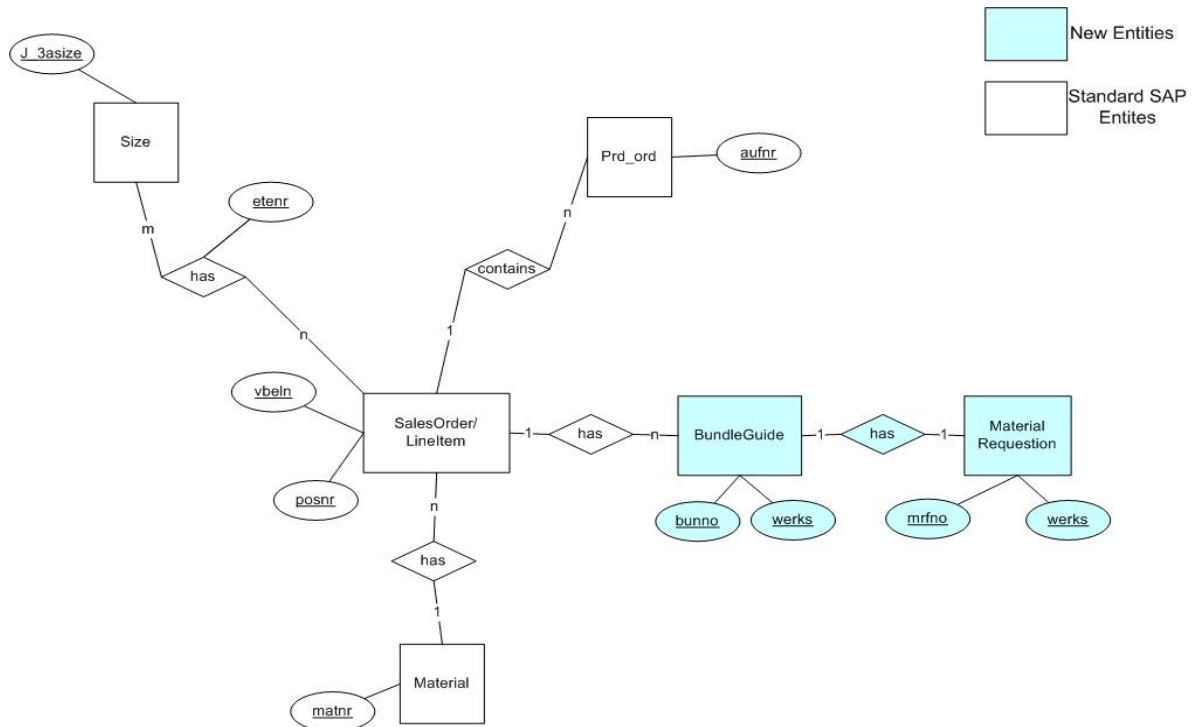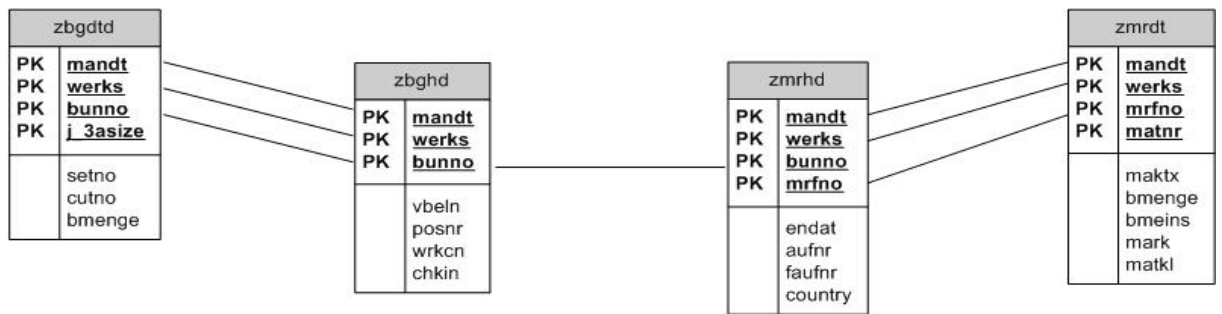
40

Figure 5.10 ER diagram for the proposed system.

- DB diagram

  In the design of the database I have to consider which tables are going into SAP and which are not. The basement of this development is in SAP. All the master data and required information stays within the SAP. So it will be easy to maintain all the required data within the SAP and the transaction data + logs to be maintained out of SAP. As a rule all the SAP tables have to be started their name with letter 'Z'. Figure 5.11 displays the database design. In the top table set provides the entity tables for the new system that are created in SAP. Two tables in bottom left side is transaction data tables created in SAP. Two tables in bottom left of the diagram displayed the logs that are created out of SAP. (SQL server)

41

Entity tables in SAP
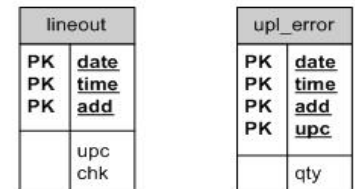

Figure 5.11 Database design for the proposed system

5.3.11.   GUI design

All the user interfaces are designed in SAP. Therefore the standard SAP GUI design methodology is performed. Following SAP standard GUI design methodology is performed in user interface design. Therefore no need of user training for the user interfaces.

1. All the GUI's are designed in SAP
2. Font will be SAP standard font type and size for the screen.
3. Screen Back Ground will be standard SAP background.
4. Text Boxes will be design as per the standard SAP Text boxes.
5. Data Grids will be design as per the standard SAP Data Grids.
6. All the main screens are numbered as 1000,2000,....etc
7. Sub screens are numbered as (1100,1200,….) depend on the main screen
8. All the screen should contain a Title Bar and "PF-Status"
9. Title bar name Format: "TITLE_ScreenNO".
10. PF-Status name Format: "STAT_ScreenNO".

42

11. PF-Status configurations.

In the PF-Status the Standard Tool bar should be as follows:

Figure 5.12 PF – Status Standard bar
"Accordance with the relevance some buttons can be skipped"

In the PF-Status the Applicaton Tool bar should be as follows:

Figure 5.13 PF – Status Application toolbar
"Accordance with the relevance some buttons or whole tool bar can be skipped"

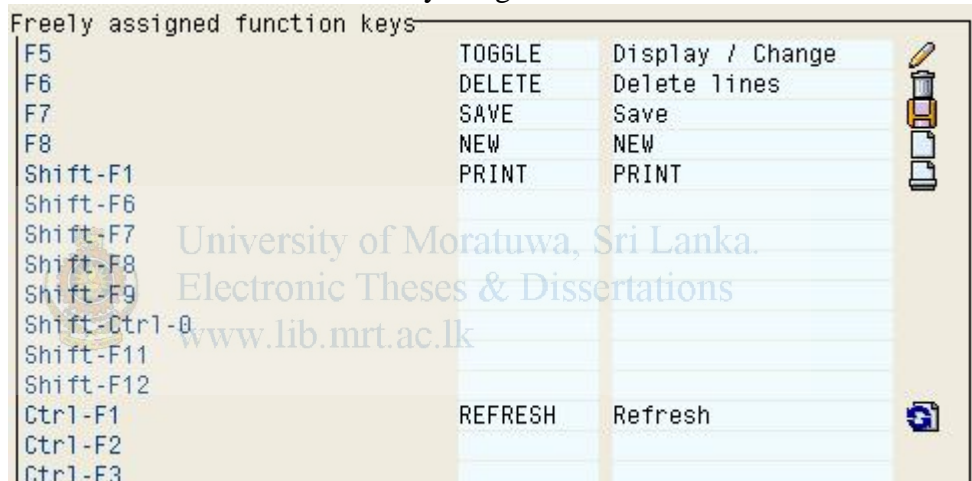In the PF-Status the Function Key assignment should be as follows:

Figure 5.14 PF-Status Function keys
"Accordance with the relevance some of the functions can be skipped"

12. Message Class will be "ZBUNMCLS" for this project.
13. Only following Message types can be used.
    S - SAP standard status message
    W - SAP standard warning message
    E - SAP standard error message

5.4. Design decision

As mention earlier the grammatical analysis is not sufficient in class diagram design. Therefore after having discussion with my supervisor I developed sequence diagrams. From that I manage to get the relevant classes.

When the UPC sticker capture by the system it will be saved directly in a table call lineout. There the date and time of the recording is added to maintain each record unique.

In the schedule upload process system reads UPC sticker codes from lineout table and total the same UPC id's recorded from same module. When that happens it automatically update the field "chk" by a 'X' for distinguish newly uploaded stickers and countered stickers.

5.5. Summary

This chapter discussed full detailed analysis and design methodology of the production confirmation system. Starting with analyzing current system, system analysises goes up to the proposed systems class and database design. Next chapter will cover the implementation details of the system.