

**Status: Draft****Approved by: Mallikarjuna Chirithanal (RBEI/ECP2)****Released by: Binu John****Internal****Document Revision History**

Ver.	Date	Changed by	Modifications
0.1	7/1/2013	Binu John	Initial Version
0.2	19/10/2016	Nagaraj/Thushar	Changed as per the new design

**Abbreviations**

Abbreviations, non-standard terms and acronyms that are used in this document are to be listed here.

ERD	Entity Relationship Diagram
GUI	Graphic User Interface
HLD	High Level Design
OOD	Object Oriented Design
RBEI	Robert Bosch Engineering and Business Solutions Limited
RUP	Rational Unified Process
IPC	Inter Process Communication



# TABLE OF CONTENTS

- 1 INTRODUCTION ..... 3
- 2 DESIGN SCOPE ..... 3
- 3 DESIGN METHODOLOGY ..... 3
- 4 DESIGN NOTATIONS ..... 3
- 5 DESIGN CONSIDERATIONS ..... 3
- 6 DESIGN OVERVIEW ..... 4
- 7 ARCHITECTURE DESIGN ..... 5
  - 7.1 Class Diagram / Module Diagram ..... 5
- 8 USE CASE DESIGN ..... 6
  - 8.1 Use Case Diagram ..... 6
- 9 INTERFACE DESIGN ..... 7
  - 9.1 Component Interface ..... 7
- 10 DATA DESIGN ..... 7
  - 10.1 Data Model ..... 7
- 11 DESIGN ALTERNATIVES ..... 7
- 12 DESIGN FEASIBILITY ..... 7
- 13 DESIGN TOOLS USED ..... 8
- 14 ADDITIONAL HARDWARE AND SOFTWARE REQUIRED ..... 8
- 15 TESTING STRATEGY ..... 8
- 16 RISK IDENTIFICATION ..... 8
- 17 TRACEABILITY MATRIX ..... 8
- 18 REFERENCES ..... 8



# 1 Introduction

This document gives an overview of the HLD for the download/install service (Frome) required for installation of appstore applications in the target based on eCORE framework. The service will be started on-demand and will have the functionality required for the installation of appstore application in the target. The appstore applications don't have dependency on any other applications and hence there is no need of package management. The appstore applications will be packaged with all the required dependencies so that the application can run independently with all the required run time dependencies. The appstore applications will be installed in the /Application/ directory of the target.

## 2 Design Scope

The eCORE Frome is designed with the assumption that there will be separate application and system storage and that the applications don't depend on each other. The design currently doesn't take care of multiuser requirements and the design will be updated when the requirements are finalized. The design supports the installation of appstore applications through USB. The Newport service will keep track of each download request and the download will be restarted in case of system restart or if they are interrupted due to loss of internet connectivity etc.

## 3 Design Methodology

Standard RBEI conventions are to be followed.

## 4 Design Notations

Refer Code guidelines document @ [\\kor123974\ECV\Projects\MM-Demo\Documents\05-Documents\01\\_Project\\_Management\03\\_eCORE\Handbook](\\kor123974\ECV\Projects\MM-Demo\Documents\05-Documents\01_Project_Management\03_eCORE\Handbook)

## 5 Design Considerations

### 1. Functionality:

The user can buy the store applications through the appstore client (mildenhall-appstore) present in the target or the application can be downloaded in the PC (Appstore portal) and can be installed in the target through USB. The download link of the application will be sent to the Frome service and the service downloads the application in the target and installs the application in the /Application/ directory of the target. The status updates received during the download and installation of the application will be broadcast to the appstore client and to the application launcher which are registered with the Frome service for the updates. The appstore client app need not to run in the background and can be killed during the installation process. The application target status namely downloading, installing, installed, un-installed progress status will be updated in the appstore server (Curridge) by the Frome service.

### 2. Reliability:

The service will restart any interrupted download due to system restart or if the internet connectivity gets lost. The target application progress status will be stored in the local database present in the target and the status will be communicated to the appstore server as soon as the internet connectivity is available.

### 3. Usability:

In case of USB appstore app installation a system daemon will run in the background and will check the inserted USB for appstore applications. If the user credentials are valid then the daemon will check if the app is already installed in the target. If the app is not installed or if there is an update available for the installed application then a popup will be displayed to the user to confirm the application installation in the target.

### 4. Efficiency:

The download/installation (Frome) service is independent of the appstore client application and thereby it can run independently. The service can be terminated if there is no request.



5. Portability:  
This requirement is for Linux platform which can be ported on ARM processor and X86.
6. Database design considerations (applicable if database design involved):  
The download requests are stored in the database file and the application installation status will also be stored in the database.
7. Protocols / Notations used by Service:  
DBus protocol will be used for IPC.

## 6 Design overview

The Frome service will expose GDBus interfaces for app download, app installation/uninstallation, and app download status.

The appstore client application and the application launcher app will register with the Frome to get updates related to application download and installation/Uninstall.

The Frome service will update the application progress status to the appstore server (Curridge) and will also store the status in the database.

The appstore client application will also specify the path where the download packages needs to be stored.

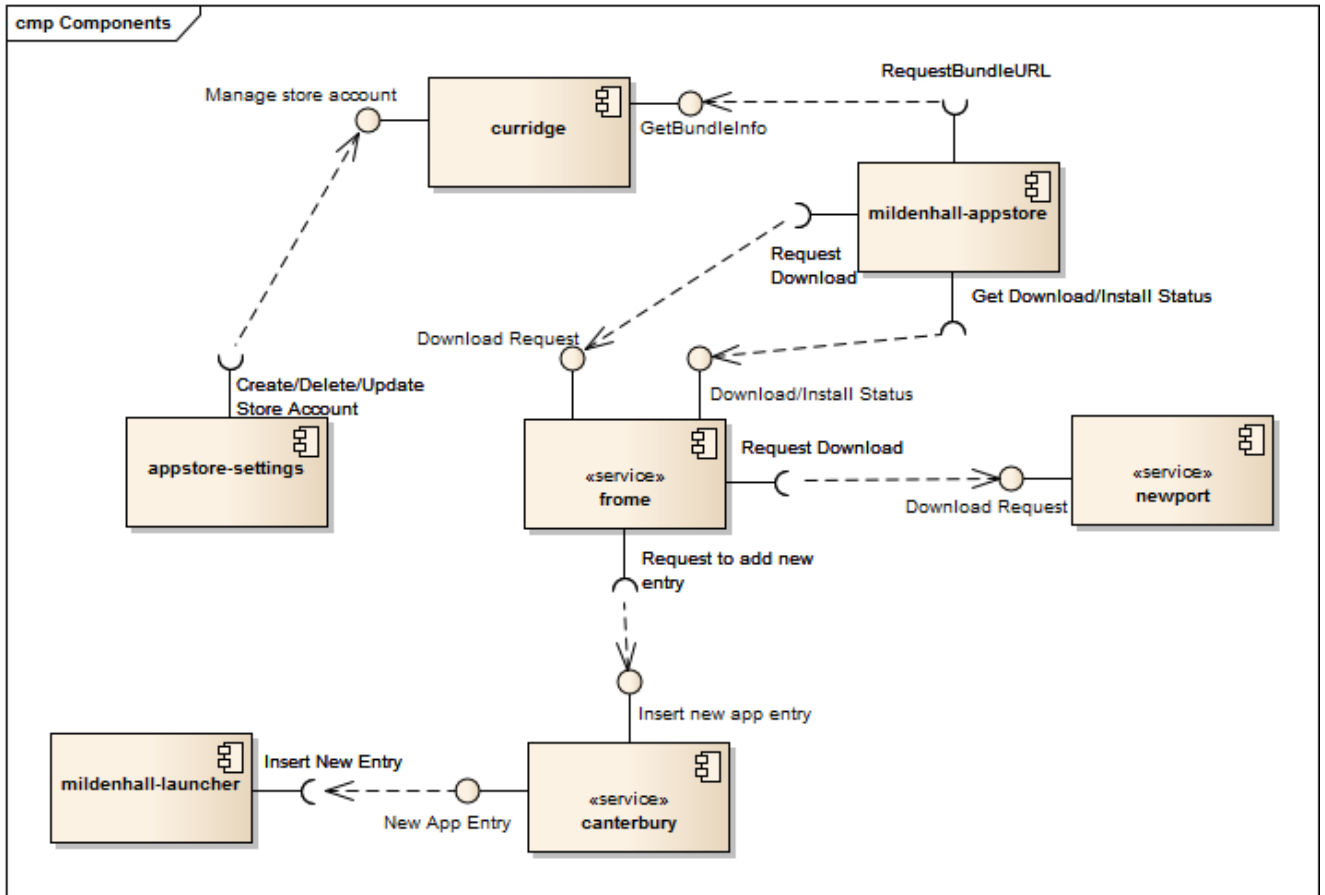
The service will be started on-demand and can terminate once the download requests are processed.

The service takes care of restarting the interrupted downloads without any user intervention.

For rollback feature, two recent versions of the same application will be maintained in the target. The service will also update the target application installation status to the appstore server (Curridge) for the respective target.

Re-installing the application won't download the application package from the appstore server (Curridge) instead the application will be installed from the target.

Please see below the component diagram:



## 7 Architecture Design

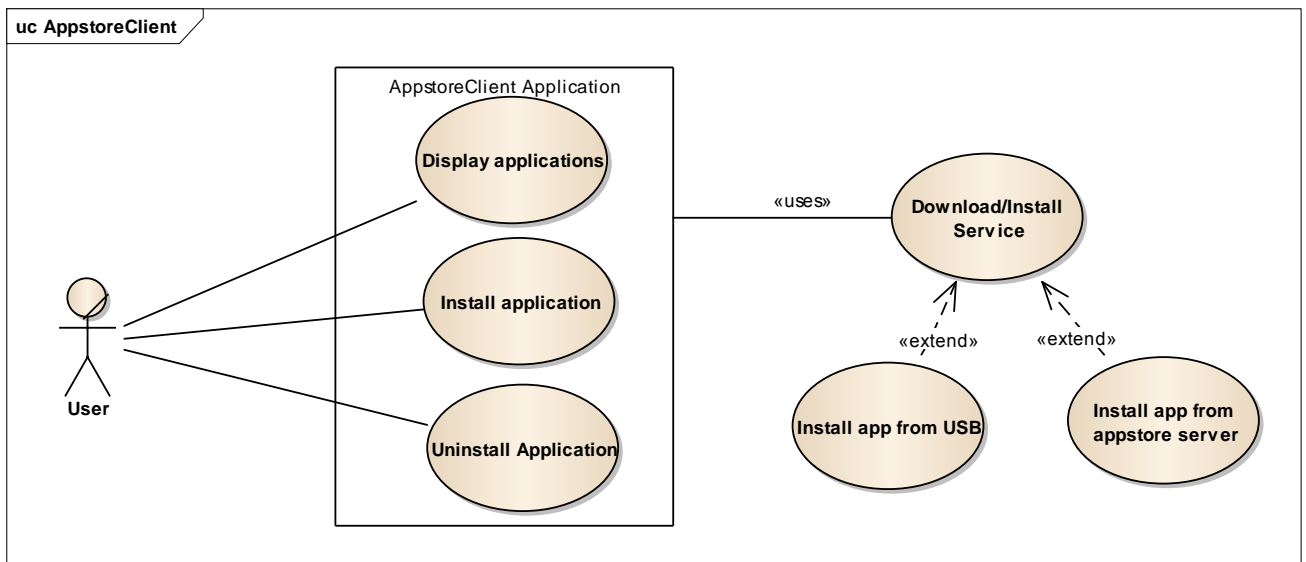
### 7.1 Class Diagram / Module Diagram

Not applicable

## 8 Use case Design

### 8.1 Use Case Diagram

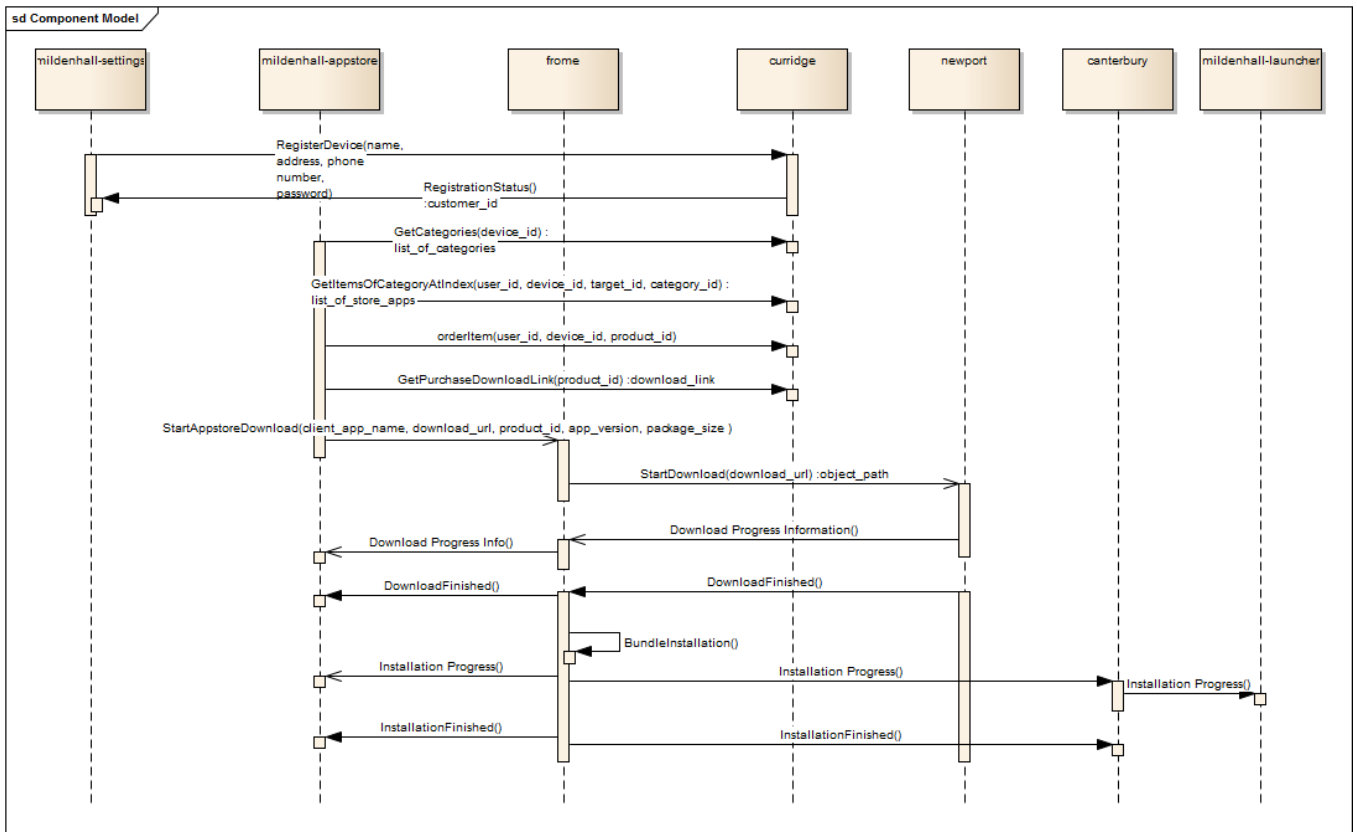
The frome service needs to support the appstore application installation from the USB and also from the appstore server. The service will be used by the appstore client application in the target and also by the application launcher to display the progress to the user.



## 9 Interface Design

### 9.1 Component Interface

The below sequence diagram shows the interaction between the target appstore client application (mildenhall-appstore) and its dependency components like Frome, Curridge, Newport, Canterbury and mildenhall-launcher.



## 10 Data design

### 10.1 Data Model

Not Applicable

## 11 Design Alternatives

Not Applicable

## 12 Design Feasibility

Not Applicable



## 13 Design Tools used

Not Applicable

## 14 Additional Hardware and Software required

Not Applicable

## 15 Testing Strategy

Test Type	Description	Applicability
Stub Code		
Trace Classes		
Simulator		
Target		
...		

## 16 Risk Identification

S. No	Description	Severity (Low/Medium/High)

## 17 Traceability Matrix

Refer traceability matrix document @ [\\kor123974\ECV\Projects\MM-Demo\Documents\05-Documents\01\\_Project\\_Management\03\\_eCORE\Traceability](\\kor123974\ECV\Projects\MM-Demo\Documents\05-Documents\01_Project_Management\03_eCORE\Traceability)

## 18 References

List of all external sources of information referenced in this document.

Sl. No.	Description	Date	Vers.	Location
1.	Bosch-vrns-applications-proposal-0.2.2.pdf			<a href="https://wiki.chaiwala.collabora.co.uk/mediawiki/images/2/27/Bosch-vrns-applications-proposal-0.2.2.pdf">https://wiki.chaiwala.collabora.co.uk/mediawiki/images/2/27/Bosch-vrns-applications-proposal-0.2.2.pdf</a>
2.				
3.				