WiPay Architecture and Design Document

Document ID	
Version	Version 0.1
URL	
Originator	Hari Ganti
Approval Date	
Status	Draft

Modification History:

Version	Date	Author	Description
0.1	05/25/2016	Hari Ganti	Initial Version

TABLE OF CONTENTS

1. Introduction

1.1. Overview

WiPay is a secured payment platform which provides seamless access to Merchents for Mobile Operator billing as well as other billing solutions like Credit/Debit cards, Net banking and Wallet services. WiPay platform provides Single API to access multiple billing services (PPD, PPU, Subscriptions etc) across different carriers and billing providers.

1.2. Purpose

Purpose of this Document is to list out different design aspects of the product, flow, Data Design.

1.3. Scope

Scope of this document is limited to Architecture and WiPay Services, Design, Flow and Data modeling. This document shall cover all the aspects which are related to billing. This document shall not cover design aspects related to Content Management, License Management.

1.4. References

- WiPay – Product Requirement Document
- Operator Billing Integration Documents
- PayTM billing Integration Documents
-

1.5. Definitions, Acronyms and Abbreviations

	Definition

1.6. Location of Document

The latest version of this document can be found on the <TBD> Repository.

1.7. Target Audience

This document is intended for Engineering, Product Management, Maintenance, Support and Business Teams to agree on requirements and functionality of WiPay. The reader shall have minimal understanding of Billing services offered by Telecom Providers in India.

1.8. Signoff

Required?	Title	Name
Yes	Product Manager	Hari Ganti
Yes	Engineering Manager	Hari Ganti
Yes	Quality Assurance	Hari Ganti
Optional	Internal Customer	Paramjit Singh
Optional	Business Head	Nitin Goel
Optional	MD	Bobby Prasad
Optional	Maintenance	Balu
Optional	Support	Karan Bhandari

2. System Overview

2.1. Generic Flow

WiPay Clients make billing request though published APIs. Based on the request and network through which the request has come, WiPay systems take appropriate actions as per flow defined by different operators/payment partners. Based on the status of billing, WiPay system informs the clients. Clients shall take appropriate steps like providing the access to service, sending the content etc based on the status of the transaction.

2.2. APIs

WiPay services will be accessed through APIs and Restful APIs are defined in separate document.

2.3. System Description

This Section described different entities which are involved in the flow

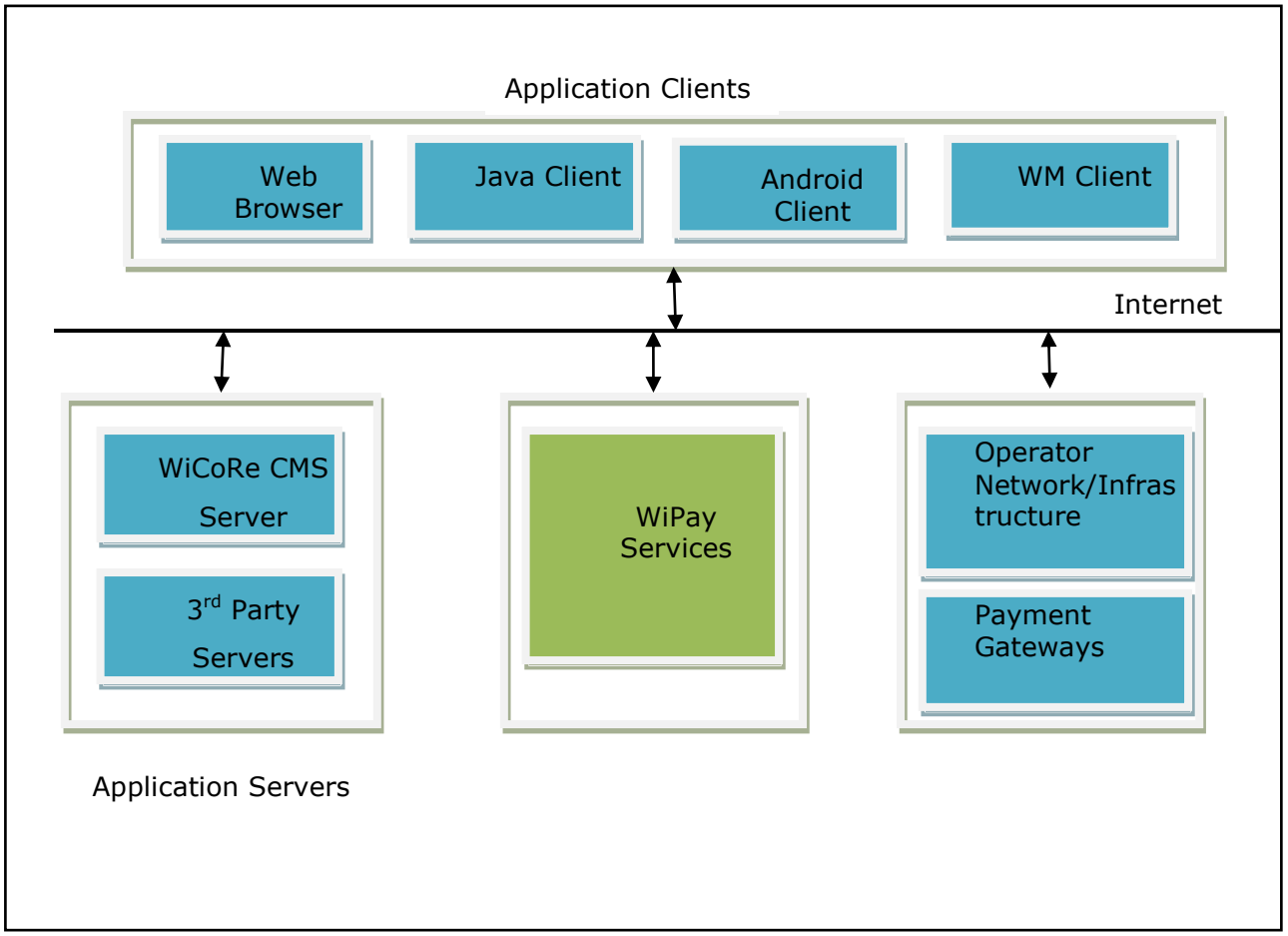


Figure 2-1: System Overview Diagram

2.3.1. Application Clients

Application Clients are responsible for provide the prescribed services to End user. These applications include Games, Content Stores, App Stores or any other Digital content Stores. Applications can be either Browser based (like wap/web Portal) or native applications for different types of devices (Mobile phones, Tablets, PC etc).

2.3.2. Application Servers

Application Servers are responsible for offering/controlling the service behavior. Theser servers are typically Content management systems or License Management systems which offer different types of Digital Content related services to end user. Application Servers and WiPay Services interact through predefined APIs

2.3.3. WiPay Services

WiPay Services act like a payment gateway for Operator billing for mobile payments or with 3rd party payment gateways for Credit Card/Debit Card/Net Banking/Wallet Services. Services offered by WiPay, will be used by Application Servers in order to offer digital content and Services.

2.3.4. Operator Network/Payment Gateways

WiPay Service will make use of services offered by Operator Network/Payment Gateways in order to charge the user for the services requested by user. WiPay Servers interact with Operator Network/Payment Gateways through APIs provided by Operator/Payment Gateway Providers.

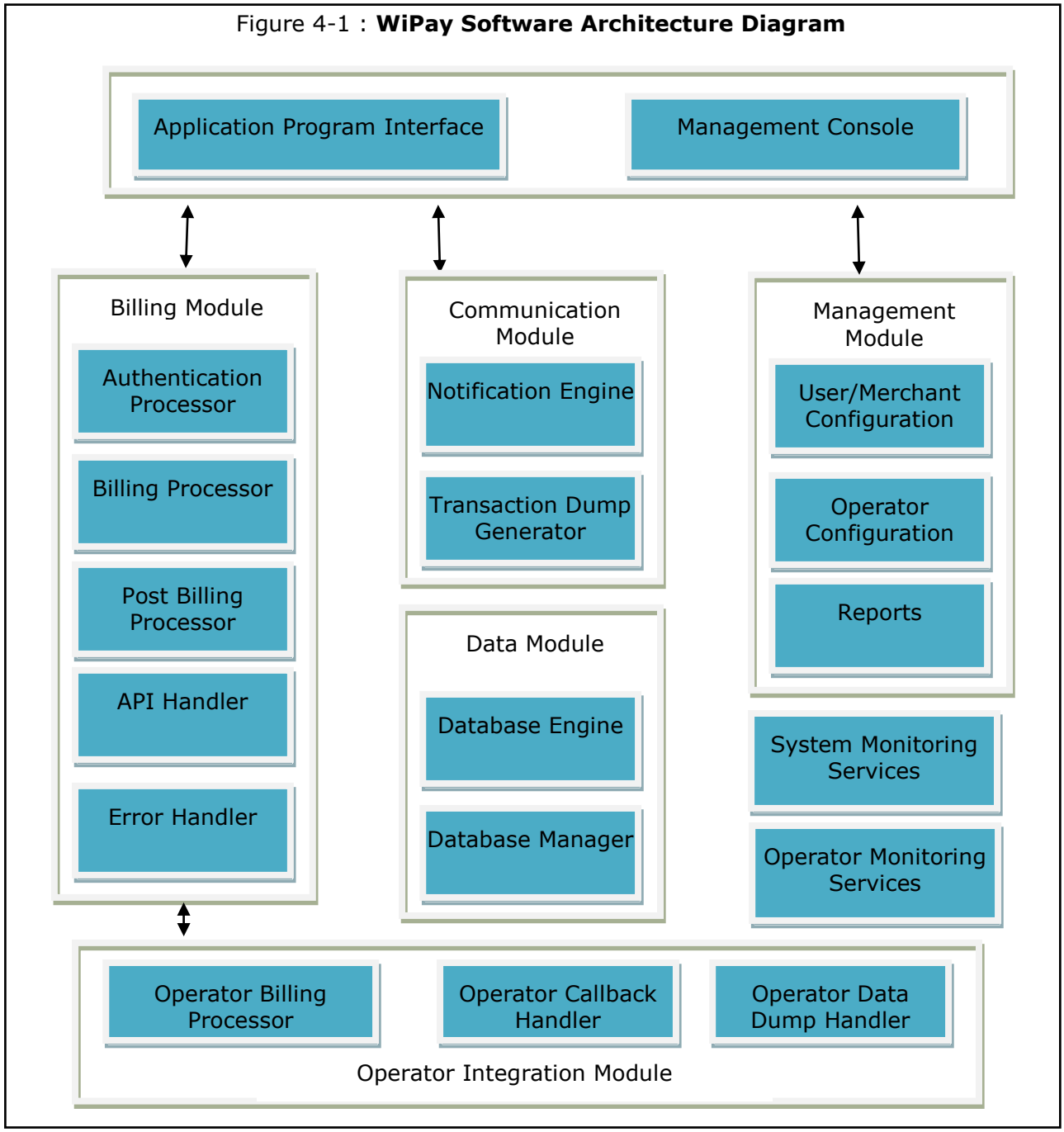
3. Hardware Architecture

TBD

4. Software Architecture

This section identifies and describes different modules of WiPay Product. The design aspects of each of these modules are discussed in later sections.

Figure 4-1 : **WiPay Software Architecture Diagram**



4.1. Application Program Interface (API) Module

In order to avail services offered by WiPay Platform, Merchant' systems will have to use Published APIs. These APIs will enable merchants to fulfill the desired functionality such as Event Billing, Subscription Billing, Unsubscriptions, getting the Subscription Status etc. Detailed list of APIs will be covered in separate document. This document will detail out how each API will be handled internally. These details are covered in Design section. APIs will make use of functionality provided by Billing Module and Communication Module.

4.2. Billing Module

Billing module will have the functionality related to performing the billing operations across different operators and Payment Gateway providers. This module consists of following sub modules

4.2.1. Authentication Processor

Whenever any Application makes a billing related request, the same will be verified by WiPay server by calling "ValidateOrderID" URL. This URL shall be provided by merchant at the time creating the merchant in the system. If this URL returns, SUCCESS, WiPay billing API will proceed further otherwise it'll return an Error message (404 error code). All Errors will be recorded and the data will be used at later stage for block listing the merchant/IP from which invalid requests are getting sent.

4.2.2. Billing Processor

All billing requests will go through the billing processor. Billing processor is responsible for converting the generic parameters to operator specific parameters and passes the request to Operator Billing processor.

4.2.3. Post Billing Processor

Post billing processor is responsible for handling the billing callbacks and status of requests coming from Operator. Post Billing processor receives the requests/status from "Operator's Callback Handler" and processes the request and performs the actions which are to be done like sending the information to appropriate client.

4.2.4. API Handler

All Requests which are coming from APIs will be passed to API handler and API handler either processes the request directly or passes the request to appropriate module. For eg., if the API is related to billing, API handler, after Authenticating the request, will forward the request to billing processor. If the API is related to getting the Status message, the same will be handled by API handler and response will be sent to client directly.

4.2.5. Error Handler

All Errors, which are related to System and billing, will be handled by Error Handler. All modules, while processing the request or while handling the response, handover the error, if occurs to Error Handler. Based on the error, Error Handler takes appropriate actions and sends the response to Clients. Centralized Error Handler is important element in order to achieve uniform functionality across different modules.

4.3. Communication Module

Communication module is responsible for sending asynchronous notifications to external systems. This will also have functionality to sending Raw Data dump, through FTP, to external servers (Merchant's system).

4.3.1. Notification Engine

Notification engine is responsible for sending asynchronous notifications to External systems. These notifications may be through SMS or HTTP based notifications. All Status updates such as New Subscription Activation, Renewals, and deactivations shall be communicated through this Notification Engine. This module will also retries sending the notification if previous attempt fails.

4.3.2. Transaction Dump Generator

In order to maintain the data integrity across WiPay System and Merchants system, Communication Module generates dump periodically and uploads the data into FTP location. Merchants can pick up this dump and process the same. Frequency of this dump generation may vary based on Merchants' need.

4.4. Data Module

Data

4.5. Management Console

Management Console is web based interface provided for Administrators, Merchants to configure the services and access the reports. Management Console will have following modules.

4.5.1. User/Merchant Configuration

This module will be used by WiPay Administrators/support team to create user and Configure Merchant with details. The access for the same shall be only with Administrator

4.5.2. Operator Configuration

This module will be used to configure Telecom Provider/Gateway Provider. All Operator Specific information (such as Billing URLs, Credentials etc) will be entered into the system through this module. Access to this module will be only with Administrators.

4.5.3. Reports Module

Reports Module will have defined set of reports for getting the transaction details, Revenue etc. Each Report will have hierarchy of sub reports.

5. Additional Material

6. Assumptions and Constraints

6.1. Assumptions

ID	Assumptions	Validated (Yes/No/Pending)
A1	None	

6.2. Hardware Constraints

ID	Constraints
HC1	None

6.3. Software Constraints

ID	Software Constraints
SC1	None

7. Documentation

7.1. Architecture, Design and Data Modeling

Software Component Design and Data Modeling documentation is used to record Data Structures, Data Flow between different modules and Operator Integration Framework. This document is used for internal team who is working on WiPay platform.

7.2. API Documentation

Application Program Interface document is intended for WiPay users who will be using published APIs to access platform functionality. This API document shall contain APIs for Making the payments, getting the status of a particular transactions etc. This document shall also contain Message Flow Diagrams and examples for easy understanding. This API document shall be published to all developers who will be developing applications using these APIs

7.3. Administrative Documentation

The Product documentation should be updated to include any new screens and commands resulting from Accessibility changes, including any limitations which the design may implement.

The expected audience for Product Administration does not change.

7.4. End-User Documentation

End user Help for Corporate Edition will need to be updated to accommodate these changes.