i2i Systems

innovation to integration

Test Cases
(TC)

Client	All Digital Operators
Project Name	MTH3902

Table of Contents

1. DOCUMENT OVERVIEW 4
1.1. SCOPE & PURPOSE 4
1.2. VERSION NUMBERS 4
1.3. AUDIENCE 4
1.4. DOCUMENT STRUCTURE AND SPECIAL CONVENTIONS 4
1.5. DEFINITIONS, SYMBOLS AND ABBREVIATIONS 5
1.6. REFERENCE DOCUMENTS 6

2. TEST EXECUTION 7
2.1. TEST CASE INDEX 7

3. TEST RESULTS 10
3.1. FINDINGS HISTORY 10

4. APPENDIX - 1 11
4.1. DEFECT SEVERITY LEVEL DEFINITIONS 11

APPENDIX - 2 12

DOCUMENT CONTROL 12

1. Document Overview

1.1. Scope & Purpose

The scope of this document is to provide i2i Internal Acceptance Test Cases and their results on [ZZZ] area which forms a critical part of the [Project_Name] Program/Project of [Customer Name].

1.2. Version Numbers

This Specification has been produced by the i2i Systems and Consultancy (i2i) following 3GPP specifications alike versioning scheme.

The contents of the present document are subject to continuing work and may be changed by i2i. Should the i2i modify the contents of the present document, it will be re-released with an identifying change of **release date** and an increase in **version number** as follows:

Version x.y.z
where:
x **the first digit:**
1 presented for information;
2 presented for approval;
3 or greater indicates approved document under change control.
y **the second digit** is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
z **the third digit** is incremented when editorial only changes have been incorporated in the document.

1.3. Audience

The document may be read by the below audience;

- Business departments
- Business/System analysts
- Solution/System Architects
- Program/Project Managers

1.4. Document Structure and Special Conventions

If the document is complex and specific sections apply to specific reader profile, a brief description of

vital sections contained in the document shall be provided.
Such as;

Section X: Explains the details related to xyz providing diagrams. This section shall be read by Business Analysts
Section Y: Explains the details related to wqv from system perspective. This section shall be read by Solution Architects
Etc.

1.5. Definitions, Symbols and Abbreviations

TERM	DESCRIPTION
API	Application Programming Interface
ASN.1	Abstract Syntax Notation 1. An abstract syntax notation, which is used to describe data and data structure.
BGW	Billing Gateway
BRS	Business Requirements Specification
BSS	Business Support System
CDR	Call Data Record – Records in files generated by external network nodes such as MSC, SDP etc.
DWH	Data Ware House
EDIFACT	Electronic Data Interchange For Administration, Commerce and Transport. EDIFACT indicates a format
EIR	Equipment Identity Register
FMS	Fraud Management System
GUI	Graphical User Interface
HTML	Hyper Text Markup Language.
HTTP	Hyper Text Transfer Protocol. Simple stateless protocol used throughout the internet.
IVR	Interactive Voice Response System
Java	An object orientated programming language which is portable between platforms.
MML	Man Machine Language, specifies the command format API of the HLR
OSS	Operations Support System
SAS	System Architectural Specification
SOAP	Simple Object Access Protocol SOAP is a simple XML-based protocol to let applications exchange information over HTTP
SRS	System Solution Requirements Specification
VAS	Value Added Service

XML	Extensible Markup Language. A widely used standard mark-up language.
XSD	XML Schema Definition. XML Schema is an XML based alternative to DTD. An XML schema describes the structure of an XML document.

1.6. Reference Documents

1. Reference 1
2. Reference 2

2. Test Execution

2.1. Test Case Index

İçinde görünmek üzere ayarlanmış paragraf stillerini kullanmadığınız için içindekiler boş.

1. TC_001: Kafka Message Polling		
Short Description: Test the functionality of polling messages from Kafka `ABMF` topic.		
Scenario Description: This test case will check if the Kafka consumer is able to poll messages from the `ABMF` topic.		
Pre-Condition: Kafka server should be running and the `ABMF` topic should exist with messages.		
Input Data: Kafka properties file with correct configurations		
Triggering the Test Case: • Initialize KafkaMessageConsumer with the `ABMF` topic and Kafka properties. • Call `pollMessage()` method.		
Output/Result Data: The message retrieved from the `ABMF` topic.		
Expected Outputs (Success conditions)	**Passed**	**Finding Id**
• The `pollMessage()` method should return a non-null message.	**YES**	

2. TC_002: Combined ABMF Function		
Short Description: Test the combined functionality of the Account Balance Management Function (ABMF).		
Scenario Description This test case will verify if the ABMF correctly polls messages from Kafka and updates the Oracle database accordingly.		
Pre-Condition: Both Kafka and Oracle database should be running, and the necessary configurations should be in place.		
Input Data: Kafka properties JDBC URL Database username and password Subscriber ID and remaining balance in Kafka message format		
Triggering the Test Case: • Initialize the AccountBalanceManagementFunction with Kafka properties, JDBC URL, username, and password. • Run the ABMF. • Send a message to the Kafka `ABMF` topic with subscriber ID and remaining balance.		
Output/Result Data: Database records showing the updated balance. Logs indicating successful polling and updating operations.		
Expected Outputs (Success conditions)	**Passed**	**Finding Id**
1) The ABMF should correctly poll the message from Kafka and update the subscriber's balance in the Oracle database.	**YES**	

3. Test Results

3.1. Findings History

Defect ID (Mantis ID)	Short Defect Description	Opened Date	Status	Severity	Developer/ Tester
1234567		12.01.2013	Open	Critical	

4. Appendix - 1

4.1. Defect Severity Level Definitions

The severity should describe how critical the problem is. This should not describe how difficult it is to fix the problem.
Generally, there are four level of severity: Critical, High, Medium and Low.

Critical: This defect can be fatal for the execution of the system and can cause loss of business functionality and/or money for the customer. For example, if a user interface that is used to activate contracts is not working due to a software error, then this means loss of business for the customer, and the defect should be fixed urgently. Or, An SQL Error gets severity “High”, because it’s a fatal for the execution. But it might be easy to fix it by correcting the statement.

High/Medium: If a defect doesn’t cause loss of business functionality and/or money for the customer, then this can be considered as “High” or “Medium”. For example, a user interface that is used to activate contracts might populate some wrong data on database, but, the activations can be done properly despite this wrong data. The defect can be corrected within a reasonable time period. In the mean time, wrong data can be corrected with manual operations.

Low: Mostly, cosmetic defects are considered as “Low”. These defects don’t cause any loss of business functionality, money or data. The system can work properly, even if the defect is not fixed forever.

Appendix - 2

Document Control

Superseded Documents

- **N/A**

Change Control & Distribution

Owner	[OWNER NAME]
Reviewer	
Approved By	
Distribution	
File Name	TEST_TestCases_[Area]_[ProjectCode]_Template_v1.1.4

Version History

Version	Change Descriptions	Author	Date
1.0.0	Initial Version	[Author Name]	01/Jan/2014

Approvals

This document requires the following approvals:

Name	Title	Date	Signature	Version