

Oracle Banking APIs

Data Protection Guide

Release 19.2.0.0.0

Part No. F26907-01

December 2019

ORACLE®

Data Protection Guide
December 2019

Oracle Financial Services Software Limited
Oracle Park
Off Western Express Highway
Goregaon (East)
Mumbai, Maharashtra 400 063
India

Worldwide Inquiries:

Phone: +91 22 6718 3000

Fax: +91 22 6718 3001

www.oracle.com/financialservices/

Copyright © 2019, Oracle and/or its affiliates. All rights reserved.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are “commercial computer software” pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate failsafe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

This software or hardware and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Table of Contents

1. Objective and Scope	4
1.1 Background	4
1.2 Objective	4
1.3 Scope	4
2. Personally Identifiable Information (PII)	5
3. Flow of PII Data	8
4. Administration of PII Data	10
4.1 Extracting PII data	10
4.1.1 Data stored in OBAPI	10
4.1.2 Data stored outside OBAPI	12
4.2 Deleting or Purging PII data	12
4.2.1 Using User Interface	13
4.2.2 Using purge procedures	13
4.2.3 Manual truncation of data from backend	14
4.3 Masking of PII data	18
5. Access Control for Audit Information	20
6. User exporting the PII data	27
7. Third Party Consents	30
8. Device ID Consents	36

1. Objective and Scope

1.1 Background

OBAPI is designed to help banks respond strategically to today's business challenges, while also transforming their business models and processes to reduce operating costs and improve productivity across both front and back offices. It is a one-stop solution for a bank that seeks to leverage Oracle Fusion experience across its core banking operations across its retail and corporate offerings.

OBAPI provides a unified yet scalable IT solution for a bank to manage its data and end-to-end business operations with an enriched user experience. It comprises pre-integrated enterprise applications leveraging and relying on the underlying Oracle Technology Stack to help reduce in-house integration and testing efforts.

In order to provide these services OBAPI needs to acquire, use or store personally identifiable information (PII). In some cases, OBAPI may be owner of the PII data and in some other cases OBAPI might just acquire and use this data for providing required services to the customer.

1.2 Objective

By the very nature of PII data, it is necessary for the Bank to be aware of the information being acquired or used or stored by OBAPI. This knowledge will enable the Bank to take necessary measures and put apt policies and procedures in place to deal with PII data. In some of the geographies Bank might need to comply with local laws and regulations for dealing with PII data. This document attempts to provide necessary information so as to enable the Bank to do so.

1.3 Scope

This document is intended for technical staff of the Bank as well as administration users of the Bank and provides information about following aspects of the PII data.

- Identifies what PII data is acquired, used or stored in OBAPI
- Process to extract PII data from OBAPI
- Process to purge and delete the PII data from OBAPI

Out of scope

This document does not intend to suggest that OBAPI is out of box compliant with any local laws and regulations related to data protection. The purpose of this document is to provide information about PII data dealt with in the system so that the Bank can put in place appropriate processes to comply with laws and regulations of the land.

2. Personally Identifiable Information (PII)

Personally identifiable information (PII) is any data that could potentially identify a specific individual. Any information that can be used to distinguish one person from another and can be used to de-anonymizing anonymous data can be considered PII.

OBAPI needs to acquire, use or store some PII data of the customers of the Bank in order to perform its desired services. This section declares the PII data captured by OBAPI so that the Bank is aware of the same and adopts necessary operational procedures and checks in order to protect PII data in the best interest of its customers.

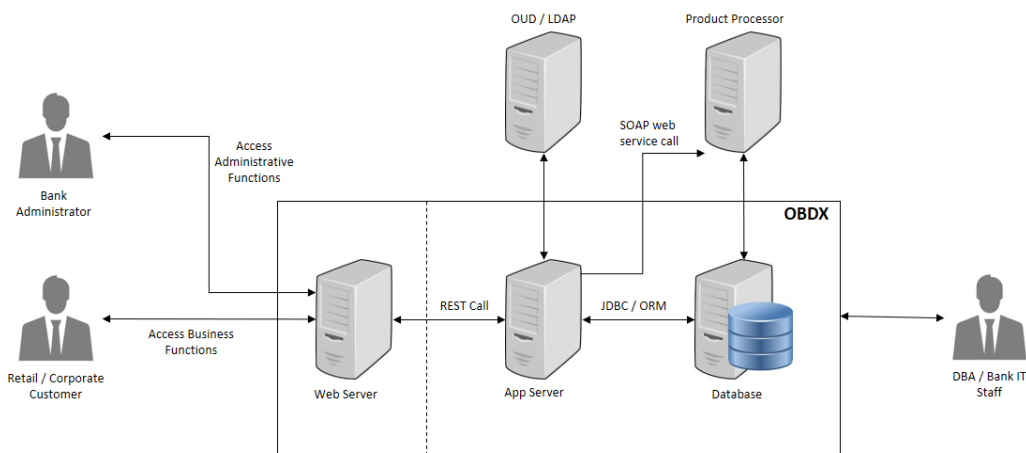
Fields	OBAPI 19.1
Bank account information	Yes
Beneficiaries	Yes
Biometric records	No
Birthplace	No
Bonus	No
Country, state, or city of residence	Yes
Credit card numbers	Yes
Criminal record	No
Date of birth	Yes
Digital identity	No
Disability leave	No
Driver's license number	Yes
Education history	No
Email address	Yes
Emergency contacts	No
Employee ID	Yes
Ethnicity	No
Financial information and accounts	Yes
Fingerprints	No
Full name	Yes
Gender	Yes

Fields	OBAPI 19.1
Genetic information	No
Health information (including conditions, treatment, and payment)	No
Healthcare providers and plans	No
Personal/office telephone numbers	Yes
IP address	No
Job title	Yes
Login name	Yes
MAC address	Yes
Marital status	Yes
Military rank	No
Mother's maiden name	No
National identification number	Yes
Passport number	Yes
Performance evaluation	No
Personal phone number	Yes
Photographic images	No
PIN numbers	Yes
Political affiliations	No
Property title information	No
Religion	No
Salary	Yes
Screen name	No
Sexual life	No
Social security number	Yes
Taxpayer information	Yes

Fields	OBAPI 19.1
Union membership	No
Vehicle registration number	Yes
Work telephone	Yes
Citizenship Number	No
Geo-Location	No
Product has Customer defined fields	No
Mobile Subscriber Identifier (IMSI)	No
Surname	Yes
First name	Yes

3. Flow of PII Data

This section depicts the flow 'personally identifiable information' (PII) within the OBAPI system in the form of a data flow diagram.



The Bank Administrator is Bank's employee who is performing administrative functions using OBAPI. As part of these, he will be dealing with PII data. An example is that the Administrator creates Retail and Corporate users in OBAPI and while creating users he/she enters user information such as first name, last name, email address, mobile number, correspondence address etc.

Retail / Corporate Customer is Bank's customer who is accessing the online banking features. As part of this he/she will be able to see his/her accounts, balances, beneficiaries, transactions, profile details etc. Note that OBAPI also supports onboarding of new users. The system captures some user information such as first name, last name, email address, mobile number, correspondence address and financial information such as income profile.

DBA / Bank IT Staff is Bank's employee who is not a user of OBAPI but has access to the database that stores OBAPI bank end data or the server environments on which OBAPI is deployed.

Web server typically contains static web content such as styling information (CSS), Javascript resources, images, static HTMLs etc. Web server passes the REST service calls to Application server.

Application (App) Server is the server on which OBAPI services are deployed. This server performs required processing on the service calls. It does use the database for retrieval or storage of data. It can also connect to external user credential store (such as OUD or Open LDAP). It can also connect to core product processor to enquiring CIF or Account related data or for posting any transactions initiated by the Retail or Corporate customer.

Database is the persistence store for OBAPI. It can contain master configuration data, user data and transactional data.

ODU / LDAP represents the external user credentials store. OBAPI does not maintain user credentials locally but depends on external specialized software to do that. An example can be Oracle Unified Directory (ODU) or Open LDAP.

Product Processor is the core banking solution which actually processes actual banking transactions. OBAPI connects to the product processor to fetch data such as CIFs or Accounts or transactions. It also connects to the product processor to post new transaction initiated by Retail or Corporate customer.

4. Administration of PII Data

This section provides information about doing administrative tasks on PII data. This includes retrieval, modification, deletion or purging of such data.

4.1 Extracting PII data

OBAPI stores some PII data in its database and it also accesses data stored or owned by external systems such as OUD / LDAP or product processor.

4.1.1 Data stored in OBAPI

This section provides information about the tables that store PII data. This information is useful for the Bank to extract PII information.

PII Data	Table
Bank account information	DIGX_AC_ACCOUNT_NICKNAME DIGX_AM_ACCOUNT_ACCESS DIGX_AM_ACCOUNT_EXCEPTION
Beneficiaries	DIGX_PY_PAYEEGROUP DIGX_PY_PAYEE DIGX_PY_DOMESTIC_UK_PAYEE DIGX_PY_INTERNAL_PAYEE DIGX_PY_DEMANDDRAFT_PAYEE DIGX_PY_INTNATNL_PAYEE_BNKDTLS DIGX_PY_DOMESTIC_INDIA_PAYEE DIGX_PY_PEERTOPEER_PAYEE DIGX_PY_INTERNATIONAL_PAYEE DIGX_PY_DOMESTIC_SEPA_PAYEE DIGX_PY_GLOBAL_PAYEE
Country, state, or city of residence	DIGX_OR_APPLICANT, DIGX_OR_APPLICANT_ADDRESS DIGX_UM_USERPROFILE
Credit card numbers	DIGX_CD_CREDITCARD_MASTER DIGX_CD_SUPP_CARD_RELATION
Date of birth	DIGX_OR_APPLICANT DIGX_UM_USERPROFILE
Driver's license number	DIGX_OR_APLT_IDNT

PII Data	Table
Email address	DIGX_OR_APPLICANT_CONTACT DIGX_OR_EMAIL_VERIFICATION (used only for email verification, data is purged once email is verified) DIGX_UM_USERPROFILE
Employee ID	DIGX_OR_APLT_EMPT
Financial information and accounts	Only financial information(Income, Asset, expense, Liability) DIGX_OR_APLT_FIN_INCM, DIGX_OR_APLT_FIN_AST, DIGX_OR_APLT_FIN_EXP, DIGX_OR_APLT_FIN_LIB
Full name	DIGX_OR_APPLICANT DIGX_UM_USERPROFILE
Gender	DIGX_OR_APPLICANT
Personal/office telephone numbers	DIGX_OR_APPLICANT_CONTACT DIGX_UM_USERPROFILE
Job title	DIGX_OR_APLT_EMPT DIGX_UM_USERPROFILE
Login name	DIGX_UM_USERAPPDATA DIGX_UM_USERPARTY_RELATION USERS GROUPMEMBERS DIGX_UM_USERPROFILE DIGX_AM_ACCOUNT_ACCESS

PII Data	Table
MAC Address	DIGX_AUDIT_LOGGING
Marital status	DIGX_OR_APPLICANT
National identification number	DIGX_OR_APLT_IDNT
Passport number	DIGX_OR_APLT_IDNT
Personal phone number	DIGX_OR_APPLICANT_CONTACT
PIN numbers	DIGX_OR_APPLICANT_ADDRESS
Salary	DIGX_OR_APLT_FIN_INCM, DIGX_OR_APLT_EMPT
Social security number	DIGX_OR_APLT_IDNT
Taxpayer information	DIGX_OR_APLT_IDNT
Vehicle registration number	DIGX_OR_APLT_IDNT
Work telephone	DIGX_OR_APPLICANT_CONTACT
Surname	DIGX_OR_APPLICANT DIGX_UM_USERPROFILE
First name	DIGX_OR_APPLICANT DIGX_UM_USERPROFILE

Please note that OBAPI provides user interface to access most of this data. The data will be accessible to you only if you have required roles and policies mapped to your OBAPI login. For example, an Administrator user can see retail user's profile only if he is entitled by a policy to access this information.

4.1.2 Data stored outside OBAPI

OBAPI can store user information in external systems such as OUD or LDAP. OBAPI provides screens for fetching this data. Please refer to the 'User Management' section of the Core user manual of OBAPI. Web help is also available at <https://docs.oracle.com/cd/F25153-01/webhelp/OBDX.htm#obdx/core/coreintro.htm>

Also note that the data can be accessed directly from the external system i.e. OUD, Open LDAP or the Product Processor. These details are outside the scope of this document. Please refer to the manual of corresponding software for more details.

4.2 Deleting or Purging PII data

There are two ways in which PII data can be deleted or purged from the system.

4.2.1 Using User Interface

The information created in (or owned by) OBAPI can be deleted from its user interface. For example, a retail user can delete the beneficiaries he/she has maintained. Please refer to 'Manage Payee' section of following user manual for more details.

https://docs.oracle.com/cd/F25153-01/webhelp/OBDX.htm#obdx/retail/payments/managepayees.htm%3FTocPath%3DOracle%2520Banking%2520Digital%2520Experience%7CRetail%2520Servicing%7CPayments%7C_____1

Note that user's data such as CIF or account number is not owned by OBAPI and hence it cannot be deleted from OBAPI. However information such as account access granted to a particular user can be modified or deleted by the bank administrator. Please refer to 'Party Account Access' and 'User Account Access' sections of the Core user manual for more details.

https://docs.oracle.com/cd/F25153-01/webhelp/OBDX.htm#obdx/core/accountaccess/acctaccesintro.htm%3FTocPath%3DOracle%2520Banking%2520Digital%2520Experience%7CCore%7CAccount%2520Access%7C_____0

4.2.2 Using purge procedures

OBAPI provides some out of the box purge procedure that can be used to purge the data. Otherwise the DBA / IT staff can prepare similar procedures to purge required data. However note that it is not recommended to purge or delete any data stored in OBAPI tables without doing detailed impact analysis. Please also note that the purge jobs are useful typically for purging old data. They may not be useful for purging data of a specific customer.

Procedure name –

DIGX_USER_PII_DATA_PURGE.sql

Procedure input parameter –

User Id (unique identifier of user) which is to be purged.

Description -

DIGX_USER_PII_DATA_PURGE will permanently purge the user and all the PII data associated with the user from all the database tables of OBAPI.

It must be noted that once user is purged then associated PII data and user cannot be retrieved under any circumstances.

Associated table –

This table holds data of table names and field names of tables containing User Id. Procedure fetches data from table DIGX_UM_USERS_ASSOCIATIONS and deletes all the PII data related to the provided User Id

Steps to run -

Run the procedure with providing User Id as input parameter.

4.2.3 Manual truncation of data from backend

In scenarios where OBAPI does not have user interface to remove customer data and scheduled purge option is not useful, then data needs to be purged using SQL scripts. Below section provides some queries that can be used for such a purging. This option must be used with utmost care and proper impact analysis must be done before using these scripts.

PII Data	Table	Script
For modules other than Origination: Personal information of user including Country, state, or city of residence, Date of birth, Email address, Employee ID, Full name, Gender, Personal/office telephone numbers, Login name, Work telephone, First Name, Surname	USERS GROUPMEMBERS DIGX_UM_USERPROFILE DIGX_UM_USERAPPDATA DIGX_UM_USERPARTY_RELATION DIGX_UM_REGISTRATION	<pre> delete from digx_um_userparty_relation where user_id = '<USER IDENTIFIER>'; delete from digx_um_userappdata where id = '<USER IDENTIFIER>'; delete from DIGX_UM_USERPROFILE where U_NAME = '<USER IDENTIFIER>'; delete from GROUPMEMBERS where G_MEMBER = '<USER IDENTIFIER>'; delete from USERS where U_NAME = '<USER IDENTIFIER>'; </pre>
Bank Account Information	DIGX_AC_ACCOUNT_NICKNAME DIGX_AM_ACCOUNT_ACCESS DIGX_AM_ACCOUNT_EXCEPTION	<pre> delete from DIGX_AC_ACCOUNT_NICKNAME where USER_ID = <USER IDENTIFIER>; delete from DIGX_AM_ACCOUNT_EXCEPTION where ACCOUNT_ACCESS_ID in (select ACCOUNT_ACCESS_ID from DIGX_AM_ACCOUNT_ACCESS where ACCESS_LEVEL = 'USER' and USERID = <USER IDENTIFIER>); delete from DIGX_AM_ACCOUNT_ACCESS </pre>

PII Data	Table	Script
		where ACCESS_LEVEL = 'USER' and USERID = <USER IDENTIFIER>;
Beneficiaries	DIGX_PY_PAYEEGROUP DIGX_PY_PAYEE DIGX_PY_DOMESTIC_UK_PAYEE DIGX_PY_INTERNAL_PAYEE DIGX_PY_DEMANDDRAFT_PAYEE DIGX_PY_INTNATNL_PAYEE_BNKD TLS DIGX_PY_DOMESTIC_INDIA_PAYEE DIGX_PY_PEERTOPEER_PAYEE DIGX_PY_INTERNATIONAL_PAYEE DIGX_PY_DOMESTIC_SEPA_PAYEE	delete from DIGX_PY_INTERNAL_PAYEE where PAYEE_ID in (select PAYEE_ID from DIGX_PY_PAYEE where CREATED_BY = <USER IDENTIFIER>); delete from DIGX_PY_DOMESTIC_UK_PAYEE where PAYEE_ID in (select PAYEE_ID from DIGX_PY_PAYEE where CREATED_BY = <USER IDENTIFIER>); delete from DIGX_PY_DEMANDDRAFT_PAYEE where PAYEE_ID in (select PAYEE_ID from DIGX_PY_PAYEE where CREATED_BY = <USER IDENTIFIER>); delete from DIGX_PY_INTNATNL_PAYEE_BNKD TLS where PAYEE_ID in (select PAYEE_ID from DIGX_PY_PAYEE where CREATED_BY = <USER IDENTIFIER>); delete from DIGX_PY_INTERNATIONAL_PAYEE where PAYEE_ID in (select PAYEE_ID from DIGX_PY_PAYEE where CREATED_BY = <USER IDENTIFIER>); delete from DIGX_PY_DOMESTIC_INDIA_PAYEE E where PAYEE_ID in (select PAYEE_ID from DIGX_PY_PAYEE where CREATED_BY = <USER IDENTIFIER>); delete from

PII Data	Table	Script
		<pre> DIGX_PY_PEERTOPEER_PAYEE where PAYEE_ID in (select PAYEE_ID from DIGX_PY_PAYEE where CREATED_BY = <USER IDENTIFIER>); delete from DIGX_PY_DOMESTIC_SEPA_PAYEE where PAYEE_ID in (select PAYEE_ID from DIGX_PY_PAYEE where CREATED_BY = <USER IDENTIFIER>); delete from DIGX_PY_PAYEE where CREATED_BY = <USER IDENTIFIER>; delete from DIGX_PY_PAYEEGROUP where CREATED_BY = <USER IDENTIFIER>; </pre>
Credit Card Information	DIGX_CD_CREDITCARD_MASTER DIGX_CD_SUPP_CARD_RELATION	<pre> delete from DIGX_CD_SUPP_CARD_RELATION where PRIMARYCARDNUMBER in (select ID from DIGX_CD_CREDITCARD_MASTER where PARTY_ID = <PARTY IDENTIFIER>); delete from DIGX_CD_CREDITCARD_MASTER where PARTY_ID = <PARTY IDENTIFIER>; </pre>
Party/User Information in Originations	DIGX_OR_APPLICANT DIGX_OR_APPLICANT_ADDRESS DIGX_OR_APLT_IDNT DIGX_OR_APPLICANT_CONTACT DIGX_OR_EMAIL_VERIFICATION DIGX_OR_APLT_EMPT	<pre> delete from DIGX_OR_APLT_FIN_INCM where APPLICANT_ID = '<APPLICANT IDENTIFIER>'; delete from DIGX_OR_APLT_FIN_AST where APPLICANT_ID = '<APPLICANT IDENTIFIER>'; delete from DIGX_OR_APLT_FIN_EXP where </pre>

PII Data	Table	Script
	DIGX_OR_APLT_FIN_INCM DIGX_OR_APLT_FIN_AST DIGX_OR_APLT_FIN_EXP DIGX_OR_APLT_FIN_LIB	<pre> APPLICANT_ID = '<APPLICANT IDENTIFIER>'; delete from DIGX_OR_APLT_FIN_LIB where APPLICANT_ID = '<APPLICANT IDENTIFIER>'; delete from DIGX_OR_APLT_EMPT where APPLICANT_ID = '<APPLICANT IDENTIFIER>'; delete from DIGX_OR_APLT_IDNT where APPLICANT_ID = '<APPLICANT IDENTIFIER>'; delete from DIGX_OR_APPLICANT_CONTACT where APPLICANT_ID = '<APPLICANT IDENTIFIER>'; delete from DIGX_OR_EMAIL_VERIFICATION where SUBMISSION_ID = '<SUBMISSION IDENTIFIER>'; delete from DIGX_OR_APPLICANT_ADDRESS where APPLICANT_ID = '<APPLICANT IDENTIFIER>'; delete from DIGX_OR_APPLICANT where PARTY_ID = '<PARTY IDENTIFIER>'; </pre>

4.3 Masking of PII data

OBAPI framework provides a facility to mask user sensitive information before showing on the screen. Masking is a process in which only some portion of the data is displayed to the user while remaining portion of the data is either skipped or is replaced with hash characters such as '*'. Main purpose of masking is to avoid a possibility of 'over the shoulder' stealing of sensitive information. However it is also used so that the clear text sensitive information is not logged in system logs.

A typical example of masking is the account numbers. When OBAPI is invoked that contains Account number in the response, the API will always give masked value. So complete clear text account number is never displayed on the screen.

OBAPI provides masking for following fields out of the box.

Sr. No	Field Name
1	Party Identifier
2	Account Number (Includes current account, saving account, deposit, loan account)
3	Debit Card Number
4	Credit Card Number
5	Mobile/phone number
6	E-mail ID
7	Social Security Number
8	Submission Identifier
9	Application Identifier

OBAPI framework also provides a provision in which any field other than the ones mentioned in the above table can also be masked as per the requirement. This can be achieved by following steps:

1. Create a complex datatype in OBAPI. This datatype must extend `com.ofss.digx.datatype.complex.MaskedIndirectedObject`
2. Define a 'masking qualifier' and a 'masking attribute'
3. Configure this masking qualifier and masking attribute in `DIGX_FW_CONFIG_ALL_B`. An example of the configurations for account number mask is given below

```
INSERT INTO digx_fw_config_all_b (PROP_ID, CATEGORY_ID, PROP_VALUE,
FACTORY_SHIPPED_FLAG, PROP_COMMENTS, SUMMARY_TEXT, CREATED_BY,
CREATION_DATE, LAST_UPDATED_BY, LAST_UPDATED_DATE, OBJECT_STATUS,
OBJECT_VERSION_NUMBER)
```

```
VALUES (*.account_id, 'Masking', 'AccountNumberMasking<', 'Y', null, null, 'ofssuser', sysdate,
'ofssuser', sysdate, 'A', 1);
```

```
INSERT INTO digx_fw_config_all_b (PROP_ID, CATEGORY_ID, PROP_VALUE,
FACTORY_SHIPPED_FLAG, PROP_COMMENTS, SUMMARY_TEXT, CREATED_BY,
CREATION_DATE, LAST_UPDATED_BY, LAST_UPDATED_DATE, OBJECT_STATUS,
OBJECT_VERSION_NUMBER)
```

```
VALUES ('AccountNumberMasking', 'MaskingPattern', 'xxxxxxxxxxxNNNN', 'Y', null, null,
'ofssuser', sysdate, 'ofssuser', sysdate, 'A', 1);
```

With above steps, the OBAPI framework will make sure to mask the data of this data type during serialization phase in the REST tier.

The masking pattern can contain following characters

1. N – Original character in the data will be retained
2. H – Original character in the data will be skipped
3. * (Or any other placeholder character) – Original character in the data will be replaced with this character

5. Access Control for Audit Information

OBAPI provides mechanism for maintaining audit trail of transactions / activities done by its users in the system. This audit trail is expected to be used for customer support, dispute handling. It can also be used for generating some management reports related to feature usage statistics etc.

From a data protection perspective it is worth noting that the audit trail contains

PII data in the form of transactional data as well as usage trends or statistics. Hence it is necessary for the Bank to put in place appropriate access control mechanisms so that only authorized Bank employees get access to this data. OBAPI provides comprehensive access control mechanism that the Bank can leverage to achieve this.

This access control can be achieved using the role based transaction mapping. This section focuses specifically from data protection aspect. You are requested to go through the user manual for 'Role Transaction Mapping' before reading further in this section. As an example, we have considered a use case where the Bank wants to restrict access to 'Audit Log' feature so that only the permitted set of administration users will be able to access audit of the users. Please note that same process can be applied to other services that deal with PII data. For example, same process can be used for restricting access to user management functions.

Check the 'out of box' access granted

There are two ways to check the Audit Information

- Maintenance
- Utilization

Maintenance (Performed by system admin)

1. Log in using Authadmin credentials.
2. Go to tab Role Transaction Mapping.
3. Find application role named "AuditAdmin" or "AuthAdmin".

The screenshot shows the 'Role Transaction Mapping' page in a web browser. The browser address bar shows the URL: `Not secure | mum00chx:3333/index.html?module=authadmin&page=application-role-base`. The page header includes a search icon, a mail icon, and a user profile for 'Welcome, Administrator User' with a last login of '05 Sep 10:52 AM'. The main content area is titled 'Role Transaction Mapping' and features a 'User Type' dropdown set to 'All'. Below this is an 'Application Role Name' search field with 'Search', 'Cancel', and 'Clear' buttons. The 'Application Role Details' section has two tabs: 'Internal' (selected) and 'External'. Under the 'Internal' tab, there is a table listing roles:

Role Name	Display Name
Administrator	
AdminMaker	AdminMakerDisplayName
AdminChecker	AdminCheckerDisplayName
AuthAdmin	AuthAdminDisplayName
payment	asa
AuditAdmin	AuditDisplayName

To the right of the table is a 'Note' box with a blue icon of a notepad. The note text reads: 'An Application may have several Application Roles for different type of users under different User Segments i.e. Retail, Corporate and Admin. These roles can be defined for internal as well as for external Access points and various transactions needs to be mapped to it. Click below to create an Application Role and map it to various transactions for selected access points.' Below the note is a 'Create' button. The browser's taskbar at the bottom shows several open applications, including Slack, Inbox, Trunk, Middleware, Oracle SQL Developer, and the current Role Transaction Mapping page.

4. Click on AuditAdmin and click on edit symbol as shown.

The screenshot shows the 'Role Transaction Mapping' page for the 'AuditAdmin' role. The form contains the following fields and options:

- Application Role Name: AuditAdmin
- Description: AuditDisplayName
- User Type: Administrator
- Access Type: INTERNAL
- Module Name: (Empty text box)
- Map Transactions to Touch Points:
 - Mobile Application
 - Mobile (Responsive)
 - Internet
 - Missed Call Banking
 - SMS Banking
 - Wearables
 - Snapshot
 - Siri/Chatbot
- Transactions:
 - Perform:
 - Approve:
 - View:

Buttons at the bottom: Delete, Cancel, Back. A callout box with a pencil icon and the text 'Click here to edit transaction mapping' points to the edit icon in the top right corner of the form area.

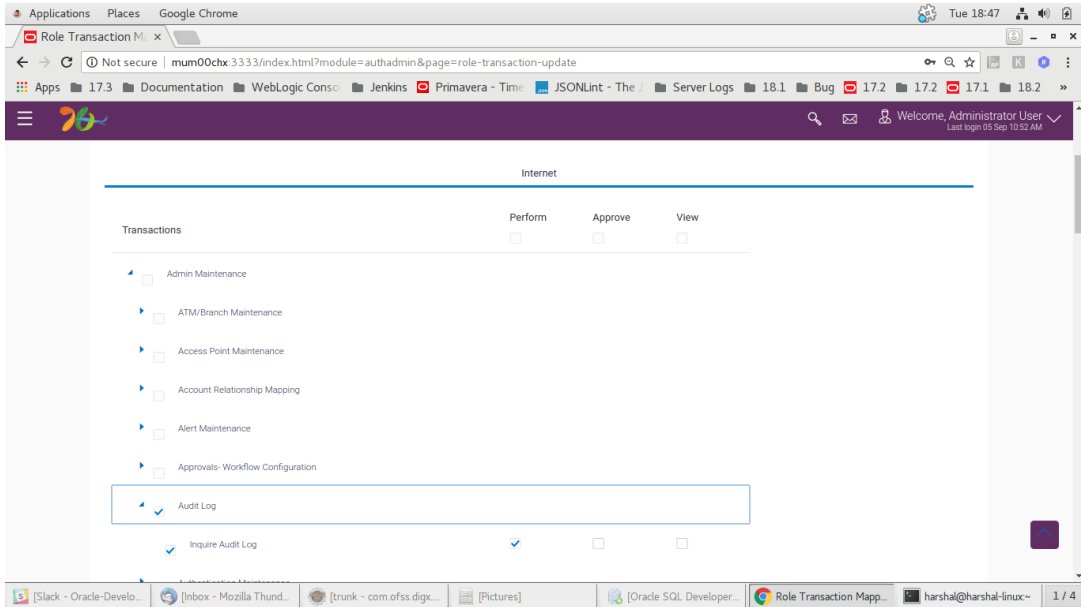
5. Assign module name "Admin Maintenance" and check "Internet".

The screenshot shows the 'Role Transaction Mapping' page for the 'AuditAdmin' role after updates. The form contains the following fields and options:

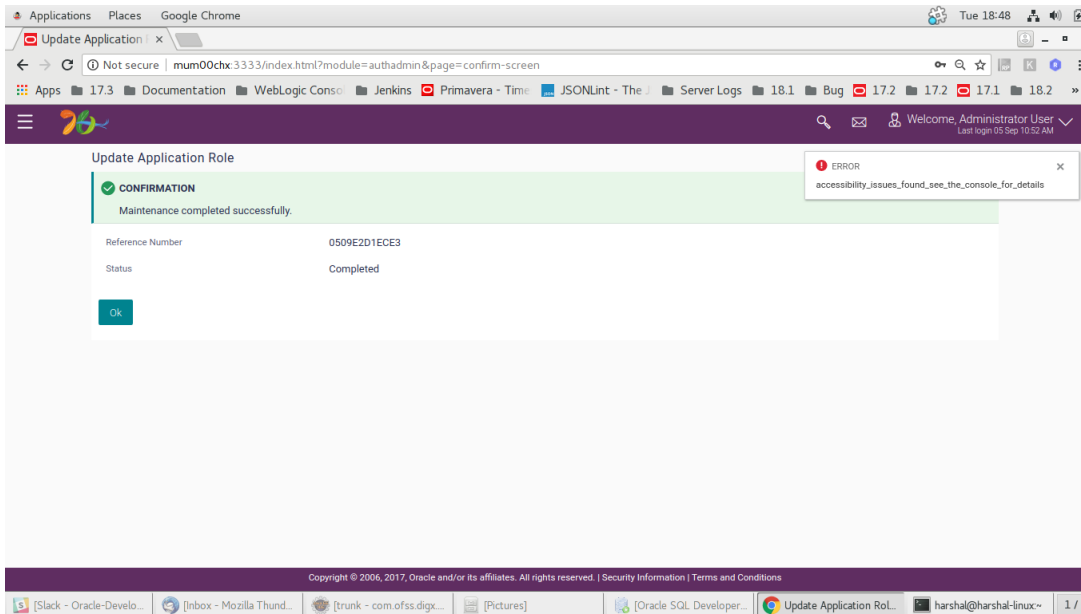
- Application Role Name: AuditAdmin
- Description: AuditDisplayName
- User Type: Administrator
- Access Type: INTERNAL
- Module Name: Admin Maintenance
- Map Transactions to Touch Points:
 - Mobile Application
 - Mobile (Responsive)
 - Internet
 - Missed Call Banking
 - SMS Banking
 - Wearables
 - Snapshot
 - Siri/Chatbot

Buttons at the bottom: Next, Back.

6. Under Admin maintenance give access of Module name Audit log to it and click save.

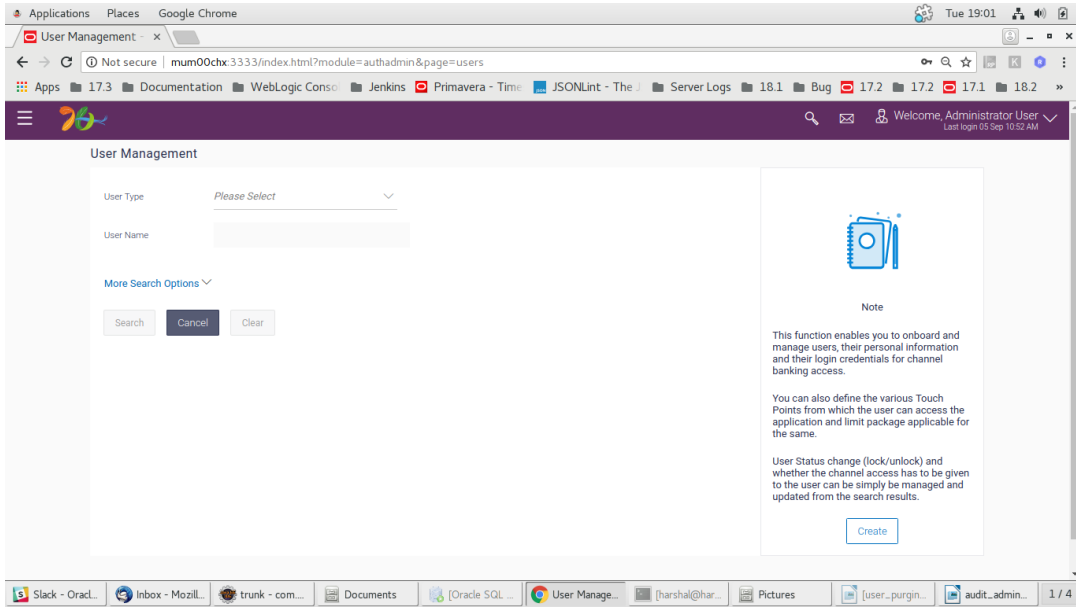


7. Submit.

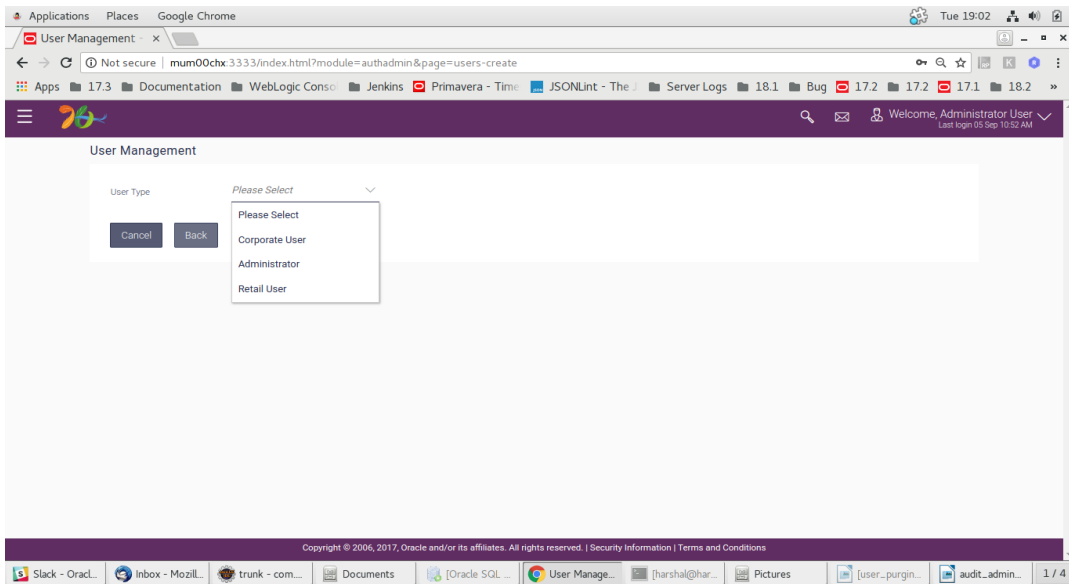


Utilization

1. Go to user management.
2. Click "Create" user.



3. Select Administrator.



4. Fill necessary details.

Applications Places Google Chrome Tue 19:03

User Management - x

Not secure | mum00chx3333/index.html?module=authadmin&page=users--create

Apps 17.3 Documentation WebLogic Conso Jenkins Primavera - Time JSONLint - The J Server Logs 18.1 Bug 17.2 17.2 17.1 18.2

Welcome, Administrator User
Last login 05 Sep 10:52 AM

User Management

User Type: Administrator

Organization: Oracle

Manager: ABC

Employee Number: 121212

User Name: AuditAdminUser Available

Title: Mr

First Name: AuditAdminUser

Middle Name:

Last Name: AuditAdminUser

Date of Birth: 04 Sep 2018

Slack - Oracl... Inbox - Mozill... trunk - com... Documents [Oracle SQL... User Manage... [harshal@har... Pictures [user...purgin... audit_admin... 1 / 4

5. Select AuditAdmin or Authadmin as an application role.

Applications Places Google Chrome Tue 19:03

User Management - x

Not secure | mum00chx3333/index.html?module=authadmin&page=users--create

Apps 17.3 Documentation WebLogic Conso Jenkins Primavera - Time JSONLint - The J Server Logs 18.1 Bug 17.2 17.2 17.1 18.2

Welcome, Administrator User
Last login 05 Sep 10:52 AM

Address Line 4

Country: India

City: mumbai

Zip Code: 123

Roles

AdminMaker AdminChecker AuthAdmin payment

AuditAdmin

Select Touch Points

Mobile Application Mobile (Responsive) Internet

Missed Call Banking SMS Banking Wearables Snapshot

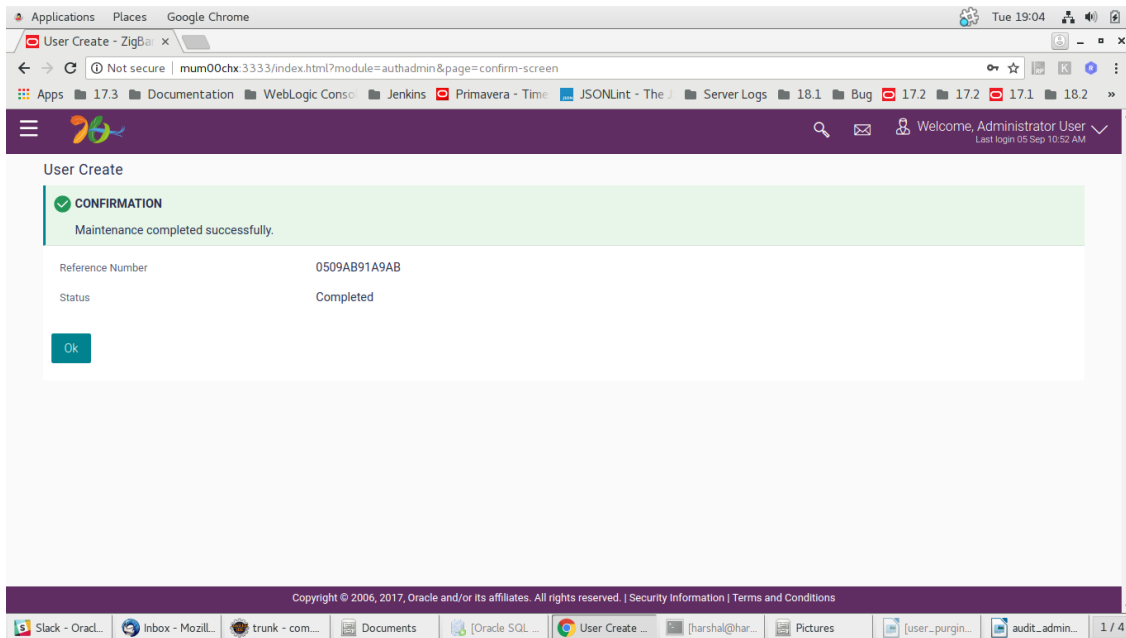
Siri/Chatbot

Add Accessible Entity

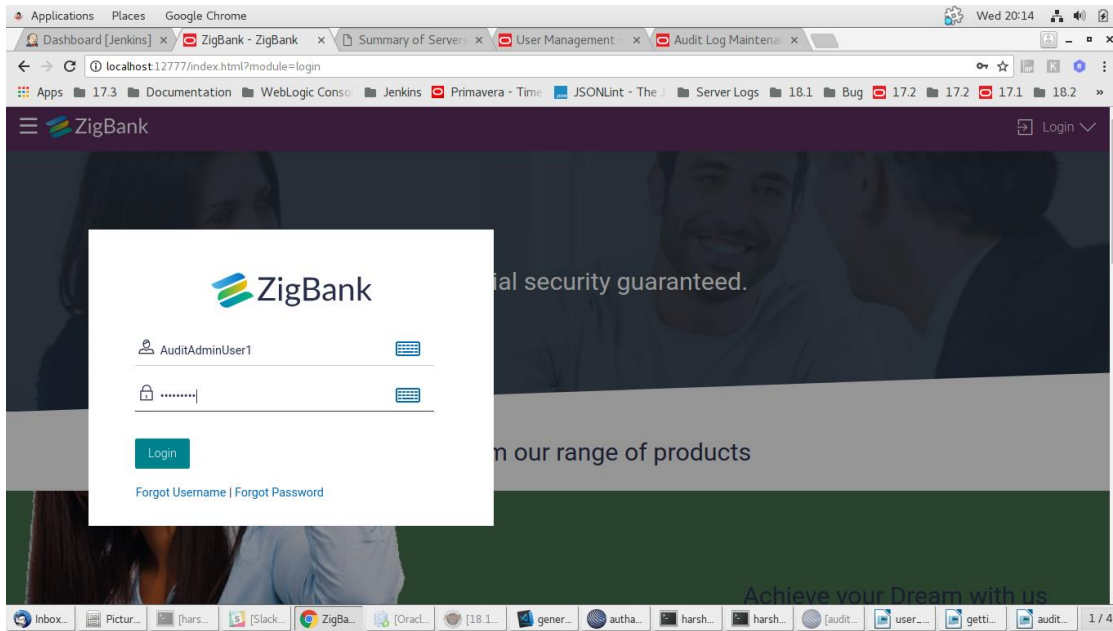
Save Cancel Back

Slack - Oracl... Inbox - Mozill... trunk - com... Documents [Oracle SQL... User Manage... [harshal@har... Pictures [user...purgin... audit_admin... 1 / 4

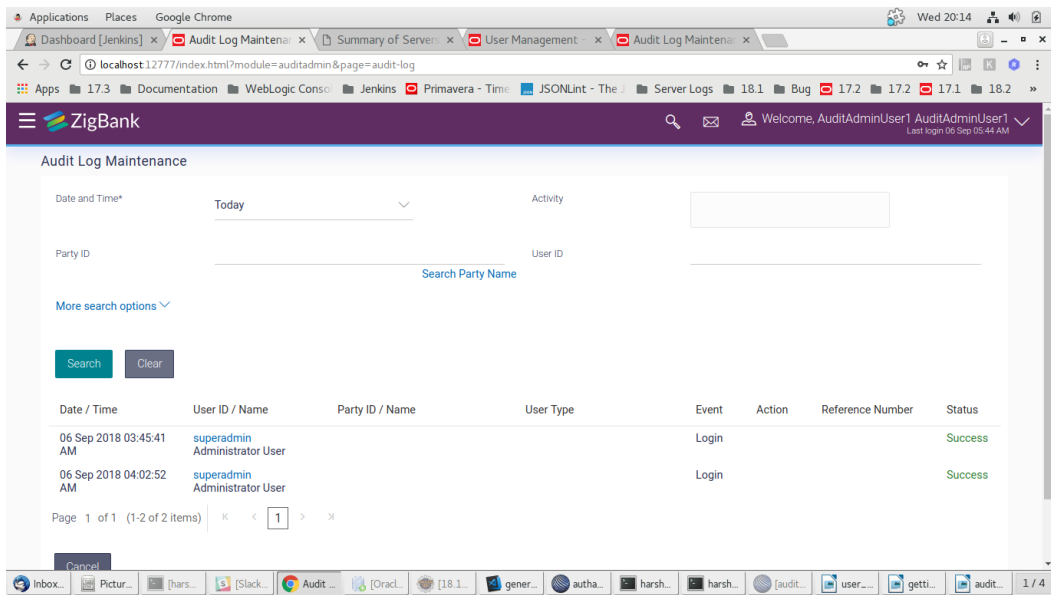
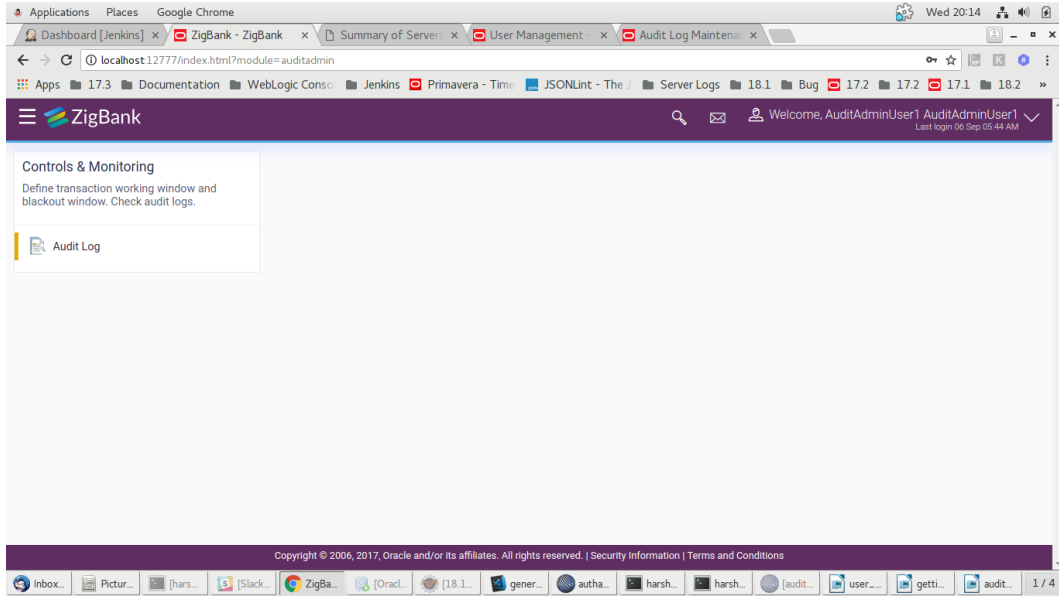
6. Submit



7. Log in using created user.



8. User can access audit log.

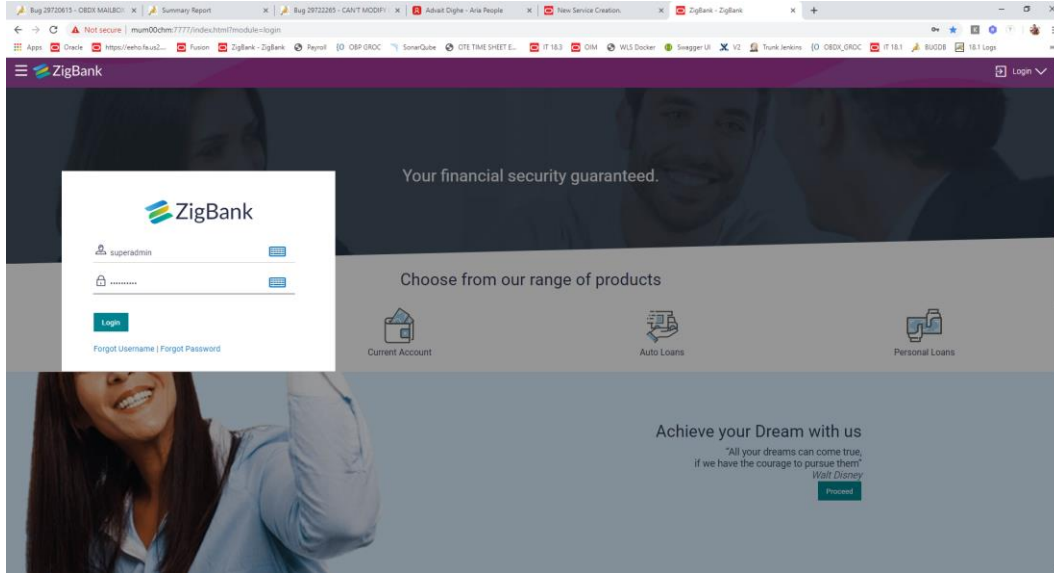


6. User exporting the PII data

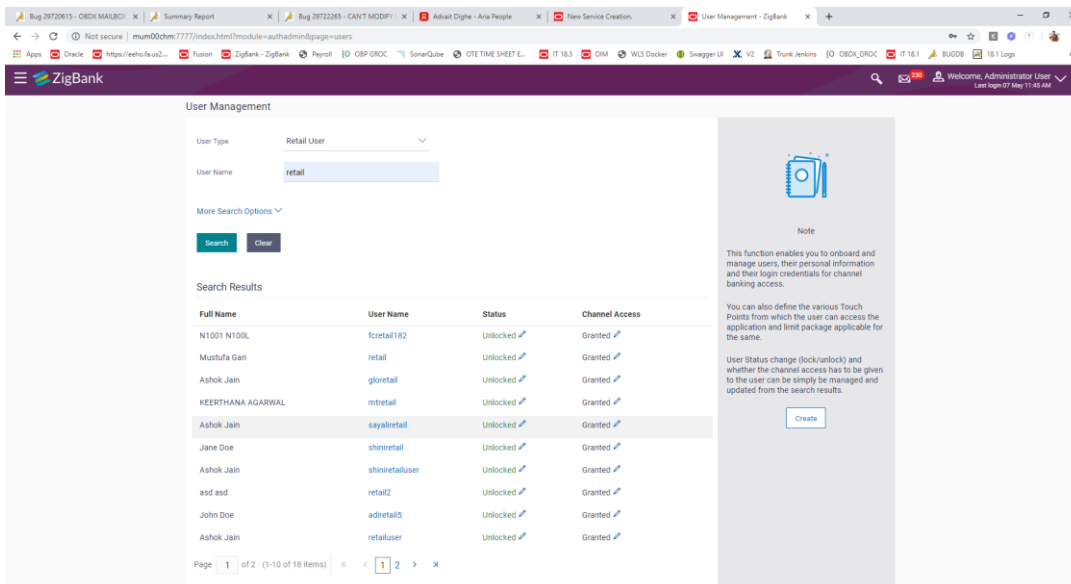
This functionality will allow to download of user wise PII in CSV formats.

6.1 Administrator

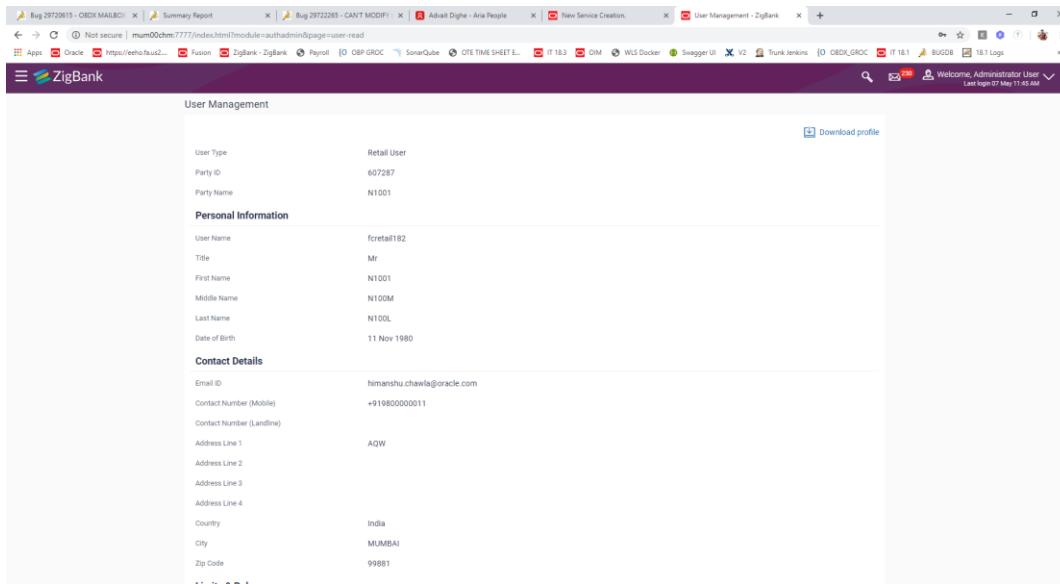
1. Login as administrator



2. Click on “User Management” and search for any user (Corporate User/ Administrator / Retail User) then clicked on the any “User Name” from the list of search users.

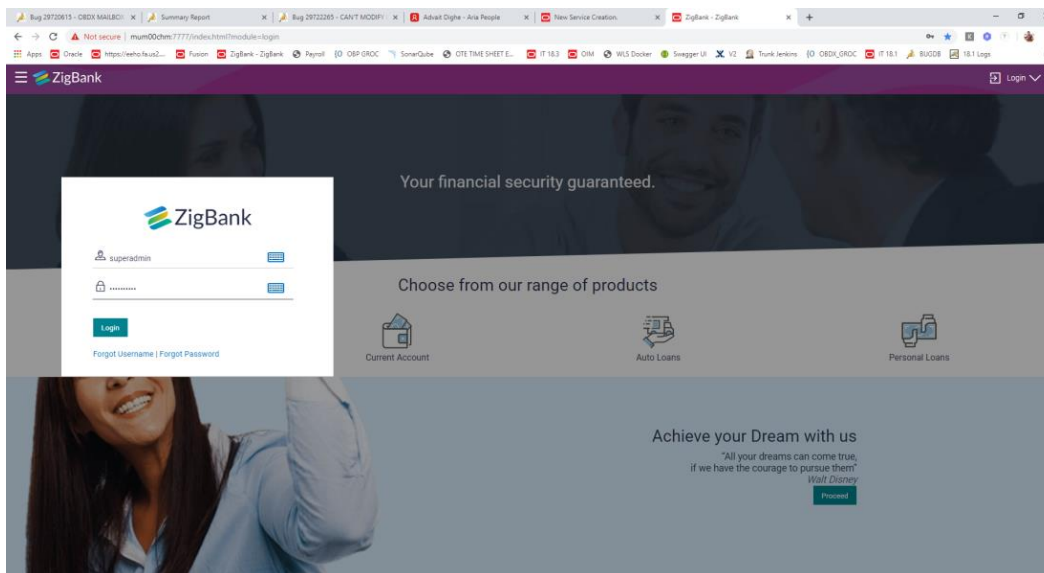


3. Clicked on the “Download profile” link.

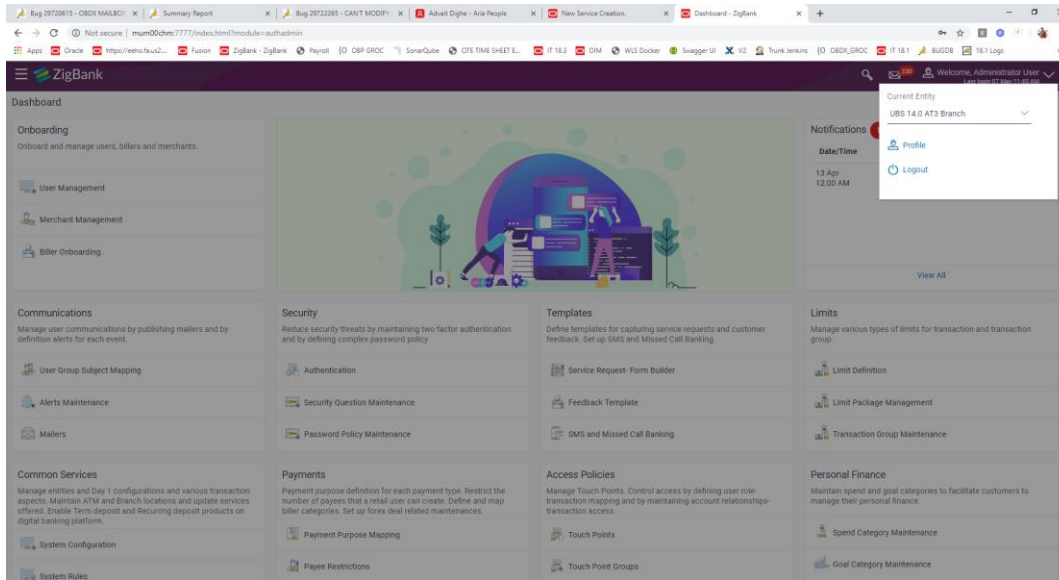


6.2 Business User

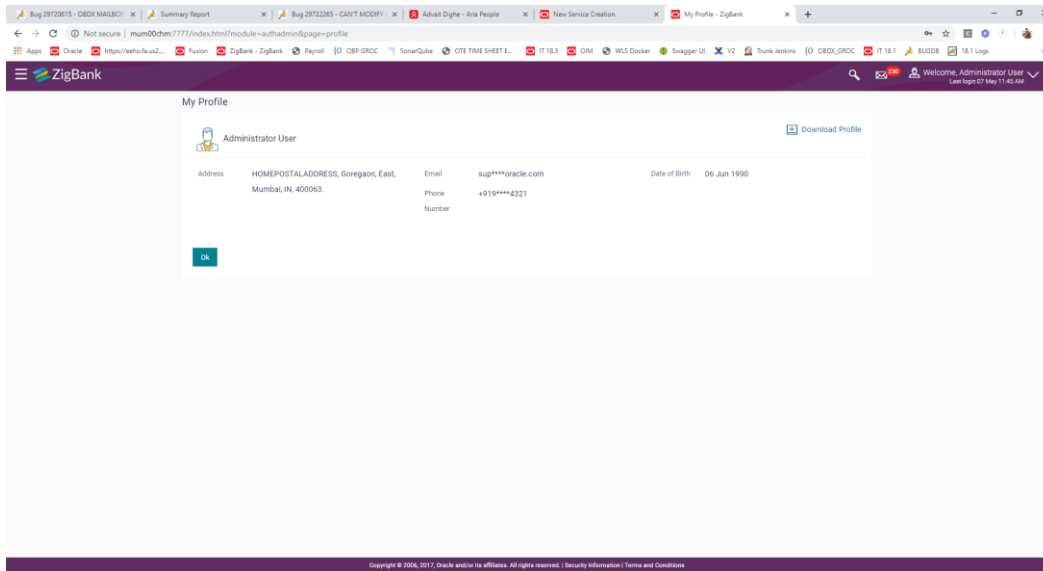
1. Login as Business User (Retail/Corporate/Admin)



2. Clicked on “Profile”



3. Clicked on “Download Profile”



7. Third Party Consents

This option enables the user to manage the access provided to third party application(s). The user can define the fine-grained entitlements i.e. account level access along with a set of transactions for the third party. The user can disable the access for a specific third party application whenever required.

Note: Only those third party applications for which the user has registered and given rights to access his/her accounts for inquiries and transactions, will appear on this page.

How to reach here:

Dashboard > Toggle Menu > Account Settings > My Preferences > Third Party Application OR

Dashboard > My Profile > Profile > Third Party Application

Third Party Apps

ZigBank | Welcome, Ashok Jain | Last login 29 Jun 06:47 PM

Third Party Consents

MODEL Solutions **epay**

Application Access: Granted

Current & Savings | Term Deposits | Loans

xxxxxxxxxxxx0035 - Savings Account Class 1

- Map All Transactions
- CASA Inquiries
 - CASA Interest Certificate
 - Party CASA Interest Certificate
 - Inquire Sweep-In Instruction
- Sweep-In Instruction
 - Create Sweep-in Instruction
 - Delete Sweep-in Instruction
- Loans
 - Loan Settlement
- CASA
 - Replace Debit card
 - Request DC Limit Change
 - Reset Debit Card Pin
 - Stop/Unblock Cheque
 - Allow International Transaction on DC
 - Request Debit Card Pin
 - Validate Card Details
 - Block Debit Card
 - Cheque Book Request
 - E-Statement Subscription
 - Demand Deposit Electronic Statement Download
 - List Demand Deposit Electronic Statement
 - Apply Debit Card
 - Request Demand Deposit Statement
- Term Deposits - Financial
 - New Deposit
 - TD Top UP
- All Inquiry Transactions
 - CASA Inquiries

xxxxxxxxxxxx0037 - Savings Account Class 1

xxxxxxxxxxxx0046 - Savings Account Class 1

[Edit](#) [Cancel](#)

[Back To Dashboard](#)

Copyright © 2006, 2017, Oracle and/or its affiliates. All rights reserved. | Security Information | Terms and Conditions

Field Description

Field Name	Description
Third Party Application Name	The names of the third party applications are displayed. Select a third party application to define access to the application.

Field Name	Description
Application Access	The option to define whether access for the application is to be provided or not. If access is granted, then the user can revoke access and if it was revoked, then the user can grant access whenever required.
Current and Savings/ Term Deposits/ Loans and Finances	Select a product to define account and transaction level access to the third party.

1. Select the third party application for which you wish to define fine grained access.
2. The system will display the list of accounts under each of the account types along with the transactions
3. Click **Edit** to modify account and transaction access. The **Third Party Consents – Edit**
4. Screen with values in editable form appears.
OR
Click **Cancel** to cancel the operation and to navigate back to the Dashboard.
OR
Click **Back** to Dashboard to go to the Dashboard.

Third Party Apps – Edit

The screenshot shows the 'Third Party Consents' page in the ZigBank interface. The header includes the ZigBank logo, a search icon, an envelope icon, and a user profile for 'Mary Doe' with the last login time '12 Jun 04:41 PM'. The left sidebar contains navigation options: Profile, Primary Account Num..., Alerts/Notifications, Third Party Apps (selected), Security and Login, and Settings. The main content area is titled 'Third Party Consents' and features the logos for 'MODEL Solutions' and 'epay'. An 'Application Access' toggle is currently turned on, labeled 'Granted'. Below this, there are three tabs: 'Current and Savings', 'Term Deposits', and 'Loans and Finances'. The 'Current and Savings' tab is active, showing a list of consented items for the account 'xxxxxxxxxxxx020 - Savings Account - Regular'. These items include: 'Map All Transactions', 'CASA Inquiries' (with sub-items 'CASA Interest Certificate' and 'Party CASA Interest Certificate'), 'CASA' (with sub-item 'E-Statement Subscription'), 'Payments' (with sub-items 'Domestic Payment', 'International Payout', 'Instruction Cancellation', 'International Draft', 'External Transfer', 'Self Transfer', 'Bill Payment', 'Internal Transfer', 'Domestic Draft', and 'PeerToPeer Transfer'), and 'All Inquiry Transactions' (with sub-items 'Payments Inquiries' and 'CASA Inquiries'). There are also sub-items 'Demand Deposit Electronic Statement Download', 'List Demand Deposit Electronic Statement', and 'Request Demand Deposit Statement'. A second account, 'xxxxxxxxxxxx018 - Savings Account - Regular', is listed below but is not selected. At the bottom of the main content area are three buttons: 'Save', 'Back', and 'Cancel'. A 'Back To Dashboard' link is located at the bottom left of the page. The footer contains the copyright notice: 'Copyright © 2006, 2017, Oracle and/or its affiliates. All rights reserved. | Security Information | Terms and Conditions'.

Field Description

Field Name	Description
Third Party Application Name	The names of the third party applications are displayed. Select a third party application to define access to accounts and transactions.
Application Access	The option to define whether access for the application is to be provided or not.

Field Name	Description
Current and Savings/ Term Deposits/ Loans and Finances	Select a product to define account level access to the third party.
Accounts	All the accounts of the user are displayed under the respective account type.
Transactions	Once you select an account, all the transactions through which the account can be accessed are displayed. Select any or all transactions to provide account access for the transactions to the third party application.

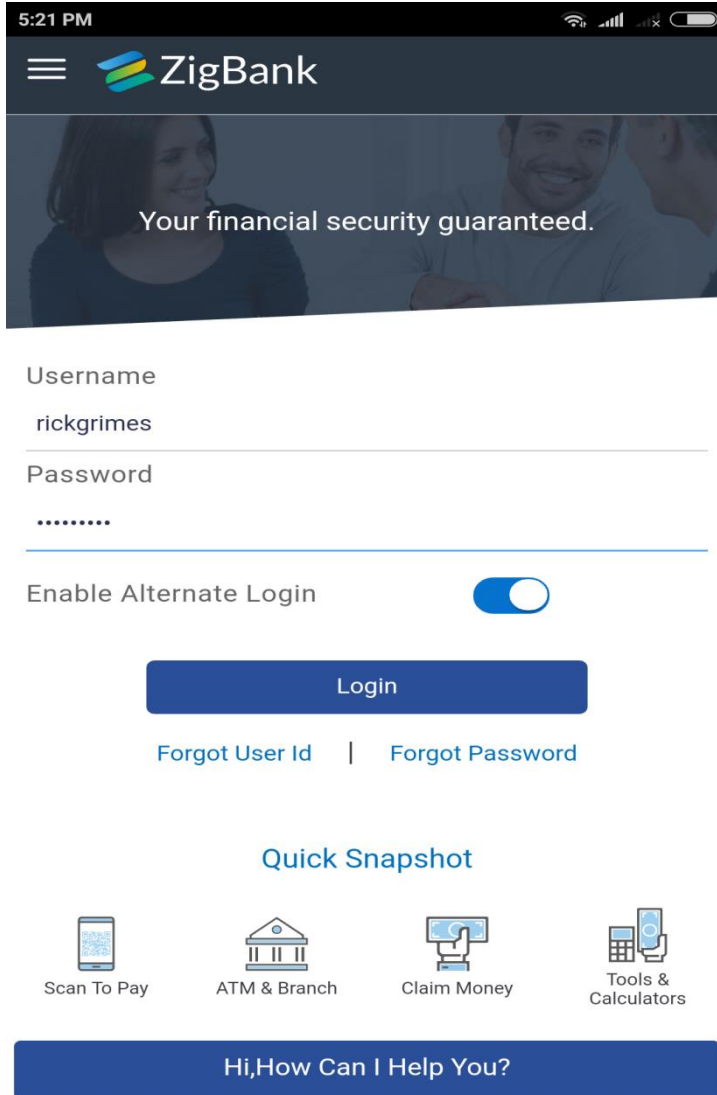
1. Click the **Application Access** button to enable / disable access for the third party application.
 - a. If you select **Enable**,
 - i. Click an account type.
The account check boxes are enabled and you can select/deselect any check box to edit access of these accounts to the third party application
 - ii. Select an account check box. The transactions for which the selected account can be accessed appear.
 - iii. Select/Deselect all or any of the transaction checkboxes to define the transactions through which the selected account can be accessed.
2. Click **Save** to save the changes.
OR
Click **Back** to go back to previous screen.
OR
Click **Cancel** to cancel the operation and navigate back to 'Dashboard.
3. The Third Party Consents – Review screen appears. Verify the details, and click Confirm.
OR
Click Back to go back to the previous screen.
OR
Click **Cancel** to cancel the operation and navigate back to Dashboard.
4. The success message of third party consent setup appears along with the transaction reference number.

5. Click **OK** to complete the transaction and to navigate back to the Dashboard.

8. Device ID Consents

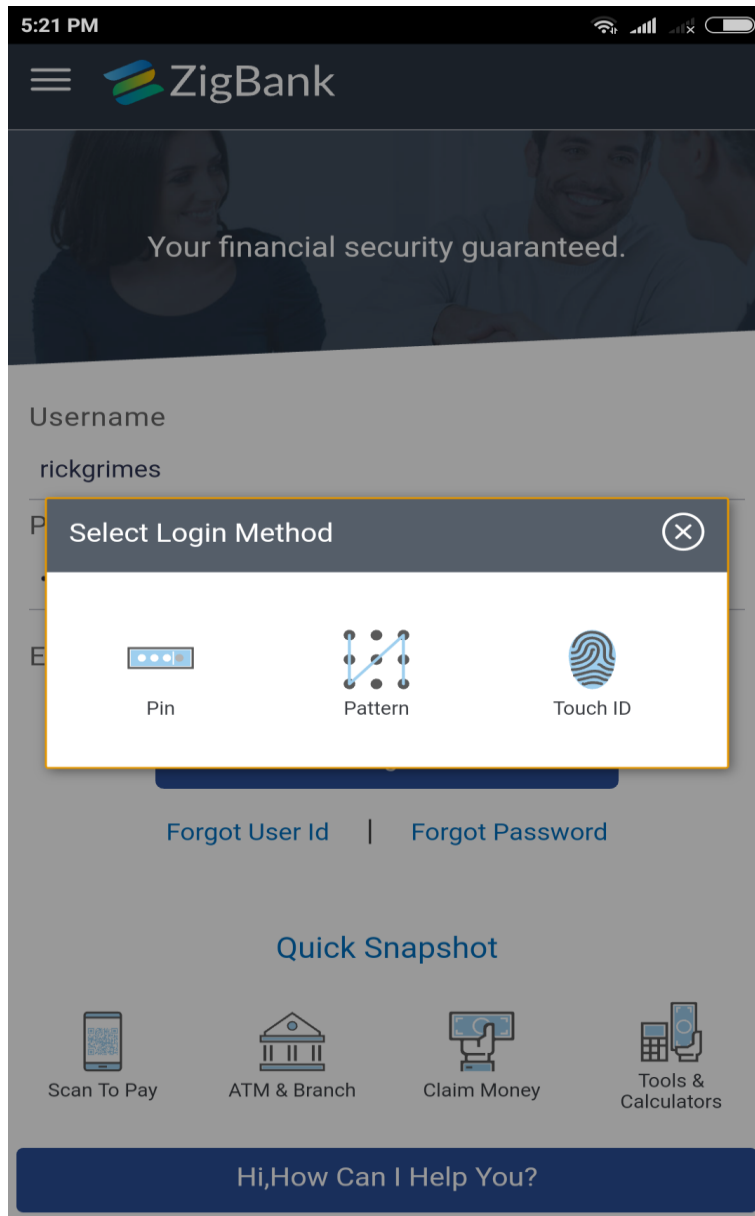
OBAPI framework provides a facility to enable the alternate login via Pin, pattern or touch ID.

1. On the login page, user will get the “Enable Alternate login” functionality. User needs to enable this for alternate login as pin, pattern or touch ID.

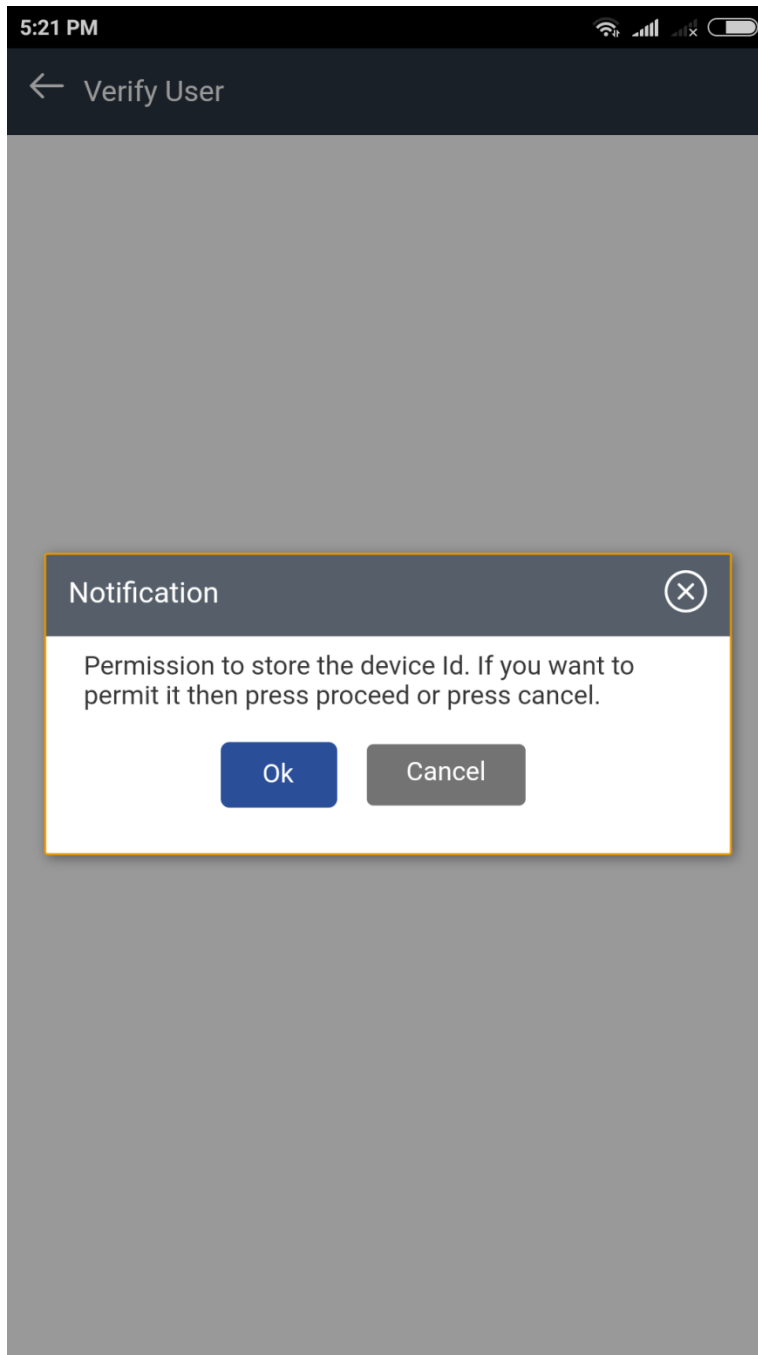


The screenshot shows the ZigBank mobile app interface. At the top, the status bar displays 5:21 PM, signal strength, Wi-Fi, and battery icons. The app header features the ZigBank logo and a hamburger menu icon. Below the header is a banner image with the text "Your financial security guaranteed." The login form includes a "Username" field with the value "rickgrimes" and a "Password" field with masked characters ".....". A toggle switch for "Enable Alternate Login" is currently turned on. Below the form is a blue "Login" button, followed by links for "Forgot User Id" and "Forgot Password". A "Quick Snapshot" section contains four icons: "Scan To Pay", "ATM & Branch", "Claim Money", and "Tools & Calculators". At the bottom is a blue button with the text "Hi,How Can I Help You?".

2. Once user enables the functionality then, "Select Login Method" pop up will come from which user can select the alternate login method.



3. Once user will select the appropriate option, Notification of permission to store the device id message will display before setting up the alternate login method.



Unregister the Device ID

In the Settings page, user can disable the alternate login from all mobile devices.

