

KCC-MISS Loan & Claim Details entry in Kisan Rin Portal (KRP)

DATA-ENTRY THROUGH MANUAL-MODE AND INTEGRATION (APIs) ON KISAN RIN
PORTAL WITH CBS / MIDDLEWARE OF BANKS
VERSION 16.3 DATED 5th JANUARY 2024

Version	Description	Release Date	Created by	Reviewed & Approved by
1	Functional requirement specification of integration of KCC-ISS portal with CBS/Middleware of banks	16.08.22	Aurionpro Solutions Limited (ASL)	ASL
2	Integration document – KCC-ISS portal with CBS/Middleware of banks	25.08.22	ASL	ASL
3	Integration document – KCC-ISS portal with CBS/Middleware of banks – Change incorporated based on technical review meeting	30.08.22	ASL	ASL
4	Integration document – KCC-ISS portal with CBS/Middleware of banks – Change incorporated based on technical review meeting	05.09.22	ASL	DA&FW, MoAFW, GoI
5	KCC/ISS portal with CBS/Middleware of banks – Draft release to banks	06.09.22	ASL	DA&FW, MoAFW, GoI
6	KCC/ISS portal with CBS/Middleware of banks – Change incorporated based on technical review meeting	16.09.22	ASL	DA&FW, MoAFW, GoI
7	KCC/ISS portal with CBS/Middleware of banks – Final draft for circulation	23.09.22	ASL	DA&FW, MoAFW, GoI
8	KCC/ISS portal with CBS/Middleware of banks – Updated with CBS APIs & their URLs	16.10.22	ASL	DA&FW, MoAFW, GoI
9	KCC/ISS portal with CBS/Middleware of banks – Updated with Bank master API, payloads & their URLs	27.10.22	ASL	DA&FW, MoAFW, GoI
10	KCC/ISS portal with CBS/Middleware of banks – Updated based on interaction with banks on 16.11.22	29.10.22	ASL	DA&FW, MoAFW, GoI
11	KCC/ISS portal with CBS/Middleware of banks – Updated as per actual terms & conditions used in policy circulars	25.11.22	ASL	DA&FW, MoAFW, GoI
12	KCC/ISS portal with CBS/Middleware of banks – Specific updates based on UAT carried out by banks & inputs received	16.12.22	ASL	DA&FW, MoAFW, GoI
13	KCC/ISS portal with CBS/Middleware of banks – Specific updates based on feedback given by banks	21.12.22	ASL	DA&FW, MoAFW, GoI
14	KCC/ISS portal with CBS/Middleware of banks – Specific updates based latest code-optimizations	12.01.23	ASL	DA&FW, MoAFW, GoI
15	KCC/ISS portal with CBS/Middleware of banks – Specific updates based latest code-optimizations	20.02.2023	ASL	DA&FW, MoAFW, GoI

16	KCC/ISS portal with CBS/Middleware of banks – capturing Claim-details via APIs	01.11.2023	ASL	DA&FW, MoAFW, GoI
----	--	------------	-----	----------------------

Prepared by

Department of Agriculture and Farmers Welfare,
Ministry of Agriculture and Farmers Welfare, New Delhi, India

Document Version History

S.No	Version	Date	Type of change	Details	Description
1	12 16-Dec-2022	16-Dec-2022	Modification	Change in names of tabs	<ul style="list-style-type: none"> ▪ ‘Crop’ in activities (multi-select) → Agri & Veg crops ▪ ‘Plantation’ in activities (multi-select) → Horti crops
2			Modification	‘Loan sanctioned on’ field in each activity in ‘Activities (multi-select)’ section	<p>✓ Loan sanctioned on in each activity in activities (multi-select) tab shall be a derived date but editable field upto current date. The derived date is the year on year incremental date of KCC loan sanctioned date entered in financial details tab.</p> <p>Example: For current FY 2022-23, If KCC loan sanctioned date in 3rd tab is 12.05.2018, then default editable value of loan sanctioned date in all activities in 4th tab is 12.05.2022.</p>
3			Modification	Financial details section	<ul style="list-style-type: none"> ▪ loanSanctionedDate → kccLoanSanctionedDate: Date of sanctioning of KCC loan / renewal of KCC loan account has to be captured. ▪ loanSanctionedAmount → kccLimitSanctionedAmount: Total sanctioned limit under KCC account of farmer has to be captured.
4			Addition	Financial details section	<ul style="list-style-type: none"> ▪ kccDrawingLimitForFY: Allowed withdrawal limit under sanctioned KCC account for the financial year selected at the start of application has to be captured.
5			Addition	Animal husbandry tab in ‘Activities (multi-select)’ section	<ul style="list-style-type: none"> ▪ liveStockCode: There are two livestock types (Diary, Poultry & small ruminant). Based on livestock type selected, the livestock name (refer DB schema for name of livestock) has to be captured in the form of code.
6			Deletion and Addition	Horti crops (erstwhile Plantation) tab in ‘Activities (multi-	<ul style="list-style-type: none"> ▪ No of trees and Age of trees are deleted. ▪ plantationCode: The plantation code based on name of plantation (refer DB schema for name of plantation) has to be captured.

S.No	Version	Date	Type of change	Details	Description
				'select)' section	
7			Addition	Inland fisheries tab in 'Activities (multi-select)' section	<ul style="list-style-type: none"> ▪ inlandType: The inland code based on type of inland (refer DB schema for type of inland such as pond, tank, open water bodies etc) has to be captured. ▪ totalUnits: Against the inland type selected, the total units have to be captured. ▪ totalArea: Against the inland type selected, the total area in hectare has to be captured.
8			Addition	Marine fisheries tab in 'Activities (multi-select)' section	<ul style="list-style-type: none"> ▪ marineType: The marine code based on type of marine (refer DB schema for type of marine such as Registered fishing vessel / boat, Fishing in estuary and sea etc) has to be captured. ▪ totalUnits: Against the marine type selected, the total units have to be captured.
9		21-Dec-2022	Modification	Change in field description	<ul style="list-style-type: none"> ▪ Change of description for field "kccDrawingLimitforFY" - Table 7 in schema
10	13		Deletion	Change in field applicability	<ul style="list-style-type: none"> ▪ Removed HORTI AND VEG CROPS under field "landArea" in table 9 in schema ▪ Removed cropType in table 9 in schema ▪ Removed cropType in payload as the dropdown is removed from UI in AGRI CROPS in multi-select (activity type)
11			Modification	Payload wherever applicable	<ul style="list-style-type: none"> ▪ Payload updated with crop codes & plantation codes inserted in production DB
12			Modification	Renaming of tabs	<ul style="list-style-type: none"> ▪ Renaming of tabs AGRI AND VEG CROPS to AGRI CROPS, PLANTATION to HORTI AND VEG CROPS
13			Modification	User manual	<ul style="list-style-type: none"> ▪ Updated user manual (for manual entry of loan application) in Annexure III

S.No	Version	Date	Type of change	Details	Description
14	14	12-Jan-2023	Modification	Updated section 5.1.2.7	<ul style="list-style-type: none"> Updated API response for crop master. Category code & category name included.
15			Modification	Updated Section 5.1.3	<ul style="list-style-type: none"> Updated length of Recipient unique ID from 17 to 19. Bank-branch code updated to 6 digits Only recordStatus = 1 and 2 are allowed (status 3 and 4 are not applicable henceforth)
16			Modification	Updated annexure II (schema)	<ul style="list-style-type: none"> Updated field length of branchCode to 6 in table 5 Updated field length of cropCode, plantationCode to 9 in table 9
17			Modification	User manual	<ul style="list-style-type: none"> Updated user manual (entry screens)
18			Modification	ANNEXURE II (DB SCHEMA)	<ul style="list-style-type: none"> Bank code will be of 3 digit Village is optional for hybrid mode isPrimary is withdrawn beneficiaryName - data validation updated landVillage - length is fixed to 6 surveyNumber - minimum length 1 liveStockCode – codes specified for different live-stocks unitCount - max length updated to 5000
19			Modification	Updates in sections: 4.2. Hybrid entry mode 5.1.10. Configuration of payload attributes	<ul style="list-style-type: none"> Facility to update API-fields configuration under hybrid-mode (conditional)
20	15	20-Feb-2023	Modification	5.1.3 Submit Batch	<ul style="list-style-type: none"> Recipient unique ID description updated Clause added for cases where incorrect branch code is sent isPrimary node removed from request-body

S.No	Version	Date	Type of change	Details	Description
					<ul style="list-style-type: none"> ▪ added sample json for recordStatus=2 request & response
21			Modification	5.1.3.C Add Loan Detail /Activities	<ul style="list-style-type: none"> ▪ section merged with Submitbatch API ▪ preuniqueId & recordStatus 3 added to request-body of the API ▪ API-response updated
22			Modification	5.1.4 Delete application(s) by application number	<ul style="list-style-type: none"> ▪ End-point named to deletebyapplicationnumbers ▪ financialYear node added to request ▪ Request method name is POST
23			Modification	5.1.6 Get data by batch ack id	<ul style="list-style-type: none"> ▪ added recipientUniqueId to API-response
24			Modification	5.1.7 Get data by Recipient's unique ID(s)	<ul style="list-style-type: none"> ▪ applicationNumbers node removed from request ▪ batchAckId and recipientUniqueIds added to response ▪ isPrimary removed from response
25			Modification	5.1.8 Get data by application number(s)	<ul style="list-style-type: none"> ▪ databyapplicationnumbers API added ▪ recipientUniqueId added to response
26			Modification	5.1.9 Get data by Sender's unique ID(s)	<ul style="list-style-type: none"> ▪ batchAckId added to request ▪ isPrimary removed from response
27			Modification	ANNEXURE II (DB SCHEMA) – 3. Basic Details	<ul style="list-style-type: none"> ▪ 2 columns for Mandatory-validation flag for Full-automation & Hybrid modes added ▪ beneficiaryName, beneficiaryPassbookName &

S.No	Version	Date	Type of change	Details	Description
					relativeName max length increased to 100
28			Modification	ANNEXURE II (DB SCHEMA) - 6. Joint Account Holders	<ul style="list-style-type: none"> ▪ Name max length increased to 100
29	16	1-Nov-2023	Addition	Section 5.1.10 to 5.1.16	<p>New APIs have been added for capturing claim details:</p> <ul style="list-style-type: none"> ▪ API to Submit claim data added ▪ API to Get Claim data by Batch Acknowledgement ID added ▪ API to get status of Claim-Batch ▪ API to get Claim data by Recipient Unique ID has been added ▪ API to get Claim data by Sender's Unique ID added ▪ API to get Claim data by Claim-number has been added ▪ API to Delete by Claim Number added
30			Addition	Annexure II (DB Schema)	<ul style="list-style-type: none"> ▪ DB Schema for IS-PRI claims have been added
31	16.1	1-Dec-2023	Update	Section 5.1.11	<ul style="list-style-type: none"> ▪ Submit partial/complete claim data API
32	16.2	28-Dec-2023	Update	Section 6.5	<ul style="list-style-type: none"> ▪ claimType added to schema-table
33	16.3	05-Jan-2024	Update	Section 5.1.11.1	<ul style="list-style-type: none"> ▪ Update farmer-mobile/joint Account Holders

TABLE OF CONTENTS

DOCUMENT VERSION HISTORY	3
1. INTRODUCTION	10
1.1. PURPOSE OF THE DOCUMENT	10
1.2. RELATED DOCUMENTS	10
2. SCOPE OF WORK	11
3. USER-ROLE AND MANAGEMENT	11
3.1. CREATION OF USER.....	11
3.2. BANK-BRANCH CREATION	12
4. MODES OF SHARING OF DATA.....	13
4.1. END-TO-END ELECTRONIC MODE	13
4.2. HYBRID ENTRY MODE.....	13
4.3. MANUAL ENTRY MODE.....	14
4.4. HANDLING UNAVAILABILITY OF SELECTIVE DATA IN API	14
4.5. HIGH LEVEL INFORMATION FLOW DIAGRAM	15
4.5.1.1. Setting up configuration for CBS integration on KCC / ISS portal	15
4.5.1.2. Application processing: Manual mode	15
4.5.1.3. Application processing: End to End electronic mode & Hybrid Modes	15
5. SCHEMA, SAMPLE PAYLOAD AND API RESPONSE	16
5.1. ENDPOINTS (DESCRIPTION LEVEL) AND PURPOSE.....	16
5.1.1. Authentication	16
5.1.2. APIs to get master data	16
5.1.2.1. Get all states.....	17
5.1.2.2. Get districts by state code	19
5.1.2.3. Get locations by district code.....	20
5.1.2.4. Get all banks	21
5.1.2.5. Get bank branches by bank code	22
5.1.2.6. Get general master configuration.....	23
5.1.2.7. Get crop master	24
5.1.3. Submit batch (array of applications).....	25
A. For recordStatus=1	27
B. For recordStatus=2	29
C. For recordStatus=3 (Add Loan Detail /Activities).....	31
5.1.4. Delete application(s) by application number	33
5.1.5. Get status by batch ack id	34
5.1.6. Get data by batch ack id.....	35
5.1.7. Get data by Recipient's unique ID(s).....	36
5.1.8. Get data by application number(s).....	39
5.1.9. Get data by Sender's unique ID(s).....	42
5.1.10. Get data by multiple batch ack id	45
5.1.11. Submit Claim Batch.....	47
5.1.11.1. Update farmer-mobile/joint Account Holders	49
5.1.12. Get Claim data by Batch Acknowledgement ID.....	50
5.1.13 Claim Batch Status	51
5.1.14 Claim data by Recipient Unique ID	51
5.1.15 Claim data by Sender's Unique ID	53
5.1.16 Claim data by Claim-number.....	54
5.1.17 Delete by Claim Number.....	55

5.1.18 Get Claim data by Multiple Batch Acknowledgement ID	56
5.1.19 Configuration of payload attributes	58
5.2. SCHEMA	58
6. APPENDIX	58
ANNEXURE-I.....	59
6.1. SAMPLE ENCRYPTION AND DECRYPTION CODE IN C#.....	59
6.2. SAMPLE ENCRYPTION AND DECRYPTION CODE IN JAVASCRIPT	63
6.3. ONLINE UTILITY TO ENCRYPT/DECRYPT THE REQUEST	64
ANNEXURE II (DB SCHEMA).....	70
6.4. FOR KCC LOAN-APPLICATIONS	70
6.5. FOR IS-PRI CLAIMS	80
ANNEXURE III.....	82
USER-MANUAL FOR ENTRY OF APPLICATIONS UNDER THE MANUAL-MODE.....	82

1. Introduction

1.1. Purpose of the document

The purpose of this document is to present the integration requirements of Kisan Credit Card (KCC) & interest subvention scheme (ISS) portal with core banking solution (CBS)/Middleware of banks. This document aims to provide schema design, sample payload, API request & responses based on communication between KCC-ISS portal and CBS/Middleware of the Banks. This will enable banks to evaluate and analyse the feasibility of integrating CBS/Middleware of the bank with KCC-ISS portal. The objective of this initiative of the DA&FW is to electronically transfer the data of KCC borrowers and applicable interest subvention maintained by the banks to the KCC-ISS portal being developed by DA&FW, GoI.

1.2. Related Documents

- Master circulars by RBI for implementation of Kisan Credit Card (KCC) scheme for the years 2017, 2018, 2019 and continuation notification 21-22



RBI circular on KCC
03-07-2017



RBI circular on KCC
04-07-2018



RBI circular on KCC
04-02-2019



4.RBI continuation
notification 21-22.pdf

- Circular by DA&FW for continuation of modified ISS for short term loans for agriculture and allied activities availed through KCC during the financial year 2021-22 and 2022-23



Guidelines for
modified ISS for 2021

- Letter sent to RBI and NABARD for development of KCC-ISS portal (dated 27th June, 2022)



Letter to RBI
NABARD for preparin

- Minutes of the meeting with Banks on Technical requirements for integration of KCC-ISS portal with CBS (dated 20th October, 2022)



CBS Minutes of the
meeting held on 16-0

2. Scope of work

As per clause 2 (viii) of the letter issued by RBI vide its letter no. RBI/2022-23/35 FIDD.CO.FSD.BC.No.3/05.02.001/2022-23 dated 28th April 2022 to all the banks specifying the need to capture and submit the data of beneficiaries under the Modified Interest Subvention Scheme (MISS) on KCC-ISS Portal. The same is reproduced below for reference:

All the above mentioned short term loans processed in 2021-22 are required to be brought on the ISS portal / DBT platform. Banks are advised to capture and submit category-wise data of beneficiaries under the Scheme and report the same on the ISS portal, individual farmer-wise, to settle the audited claims arising from 2021-22 onwards.

Further, on 27th June 2022, DA&FW has released a letter to RBI and NABARD on launching of KCC-ISS portal. The data fields to be captured through the portal are mentioned at para 3 of the above referred letter.

Further, it has been decided to electronically collect the data of farmers/borrowers of KCC including information about loans sanctioned and eligible interest subvention claims from all banks from FY 2021-22 onwards for all the farmers who have availed KCC loans.

3. User-role and management

3.1. Creation of user

Following roles (and login-credentials) of PMFBY-portal will work on KCC-ISS portal:

- i. Bank HQ Admin
- ii. Bank Branch Admin
- iii. Bank Branch-user

In case a new branch Admin/user is to be created then same shall be created by Bank HQ admin/Branch Admin on PMFBY, respectively.

If, any bank does not have HQ Admin credentials on PMFBY then same should be requested through letter signed by authorized-signatory, on bank's letterhead. The letter should be addressed to CEO, PMFBY (Department of Agriculture & Farmer Welfare) stating following details:

Name	
Designation	
Official e-mail	
Mobile number	

The letter (soft-copy) can be shared through bank's official e-mail account to: anil.kumar004@gov.in, kartikey.upadhyay@aurionpro.com and copy to: ajay.k.singh@nic.in

To create Branch Admin from Bank HQ, on PMFBY:

To create Branch-admin on PMFBY portal, Bank HQ Admin has to login and go to User Console page

Home Downloads Report **User Console** KCC

Season/Year: Andaman Nicobar Islands/Rabi/PMFBY/2022

User Console

Create User

Manage User Pending Approvals Branch Creation IFSC Creation Branch Edit

Official Information

Stakeholder: BANK

Choose Bank Type: COOPERATIVE

User Category: Branch Admin

State: Select State

District: Select District

Search Bank By IFSC: YES NO

Bank Name*: Select

Branch Name*: Select

Personal Information

Title*: Select

Name*: Enter Name

Email: Enter Email

Mobile No*: Enter Mobile No.

Employee ID: Enter Name

Office Landline No: Code Phone No.

Create

Banks will also be able to create users from KCC-ISS portal through its User-module (upcoming module)

3.2. Bank-branch creation

To create Branch-user on PMFBY portal, Branch Admin has to login and go to User Console page:

Home Application Payment Downloads Report **User Console** KCC AH&D KCC

Season/Year: PUNJAB/Kharif/PMFBY/2022 Print Page

User Console Commercial Bank Branch Manager

[Create User](#) [Manage User](#) [Pending Approvals](#) [PACS](#)

Official Information

Stakeholder	Choose Bank Type	User Category
BANK	COMMERCIAL	Branch User
State	District	Search Bank By IFSC
Select State	Select District	<input type="radio"/> YES <input checked="" type="radio"/> NO
Bank Name*	Branch Name*	
Select	Select	

Personal Information

Title*	Name*	Email	Mobile No*
Select	Enter Name	Enter Email	Enter Mobile No.
Employee ID	Office Landline No.		
Enter Name	Code	Phone No.	

[Discard](#) [Create](#)

4. Modes of sharing of data

The test-environment URL for KCC-ISS portal is <http://fasalrin.gov.in>

There will be three modes of data-transfer between ISS/KCC portal & financial institution(s):

4.1. End-to-end electronic mode

Complete data of farmers/borrowers of KCC including information about loans sanctioned and eligible interest subvention claims of the banks as per pre-defined schema/fields shall be transferred electronically through API services by integrating the CBS/Middleware of the respective banks with the KCC-ISS Portal.

4.2. Hybrid entry mode

Partial data of farmers/borrowers of KCC including information about loans sanctioned and eligible interest subvention claims of the banks as per pre-defined schema/fields shall be transferred electronically through API services by integrating the CBS/Middleware of the

respective banks with the KCC-ISS Portal. For rest of the fields which are not available in the CBS/Middleware of the respective banks, the requisite fields will be entered manually on KCC-ISS portal by the respective branches of the bank, using the specific login credentials. For those branches/users who do not have credentials on NCIP, new credentials of Bank Branch Admin can be created by HO admin on NCIP or KCC-ISS portal.

All the schema/fields are mandatory for KCC-ISS Portal. In this mode, bank HQ admin will flag the fields as ‘Yes’ which are available in their CBS/Middleware and those fields would be uploaded through CBS (APIs); remaining fields have to be marked as ‘No’ and they would have to be entered manually by bank-branch users. System will process the batches as per the fields defined in the configuration settings.

The field data uploaded through the CBS (APIs) will be displayed to branch-user in read-only mode and cannot be edited. The applications for which the partial-data is filled up through electronic mode shall not be considered as complete and only after manual-entry of remaining fields by the branch user can the application be submitted.

***Some fields will be compulsory for creation of beneficiary-application in KCC-ISS portal; these fields have to be sent by API under Hybrid-mode.**

Henceforth, Bank may update field-configurations (hybrid mode) as per their requirement, subject to the condition that there are no batches under processing.

4.3. Manual entry mode

Complete data of farmers/borrowers of KCC including information about loans sanctioned and eligible interest subvention claims of the banks as per pre-defined template shall be directly entered on the KCC-ISS portal by the respective bank branch users of the bank using the specific login credentials created for them on National Crop Insurance Portal (NCIP). For those branches/users who do not have credentials on NCIP, new credentials can be created by the bank HO admin in a same was as is applicable for NCIP.

Each bank HQ shall have to declare specifically about their preferred mode of data sharing. Once the preference for a specific mode is declared, the same shall not be allowed to be changed at later stage during the particular financial year.

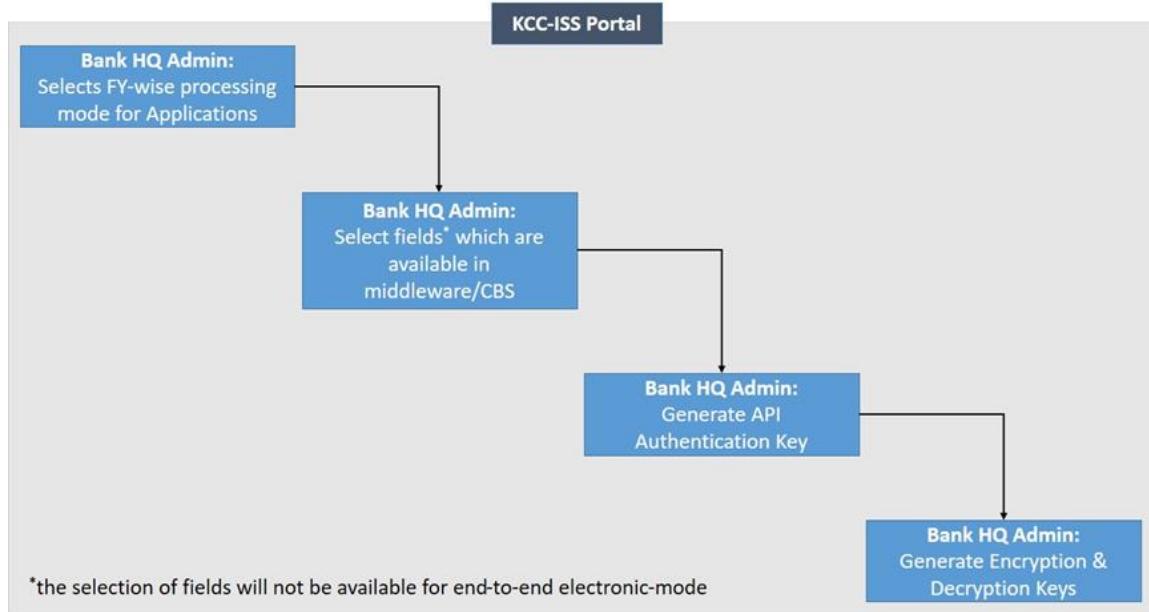
4.4. Handling unavailability of selective data in API

In case of Hybrid OR End-to-end automation modes, the minimum mandatory fields defined by Ministry would necessarily have to be submitted by banks (through API).

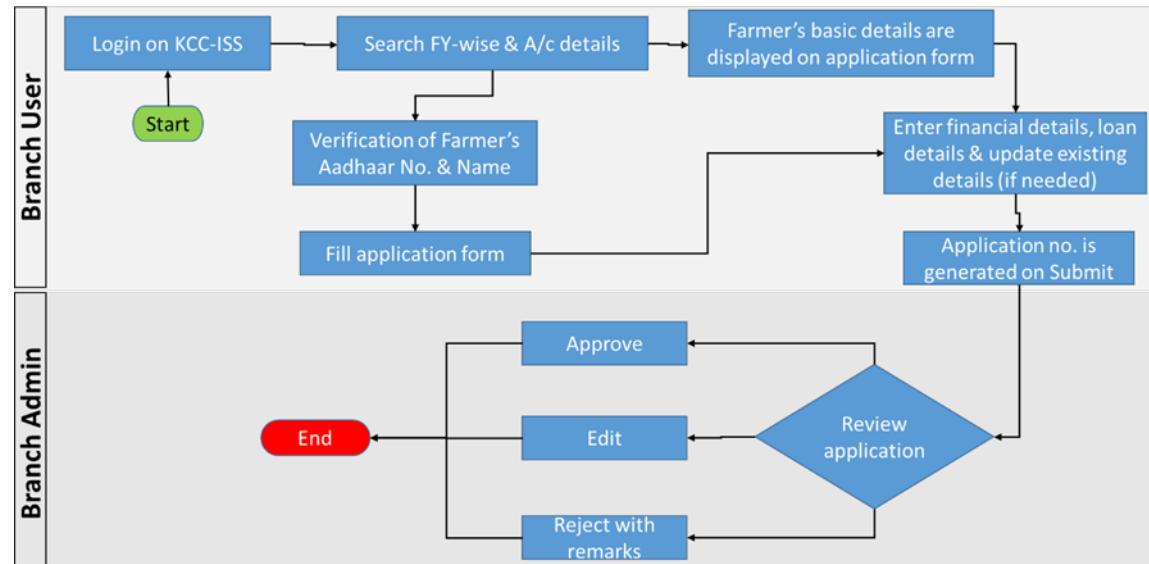
In cases, where details are available for some accounts and unavailable for remaining accounts, Bank would be required to send “\$” value through API but branch user would be required to fill such fields manually through their login and only then would the application be submitted.

4.5. High level information flow diagram

4.5.1.1. Setting up configuration for CBS integration on KCC / ISS portal

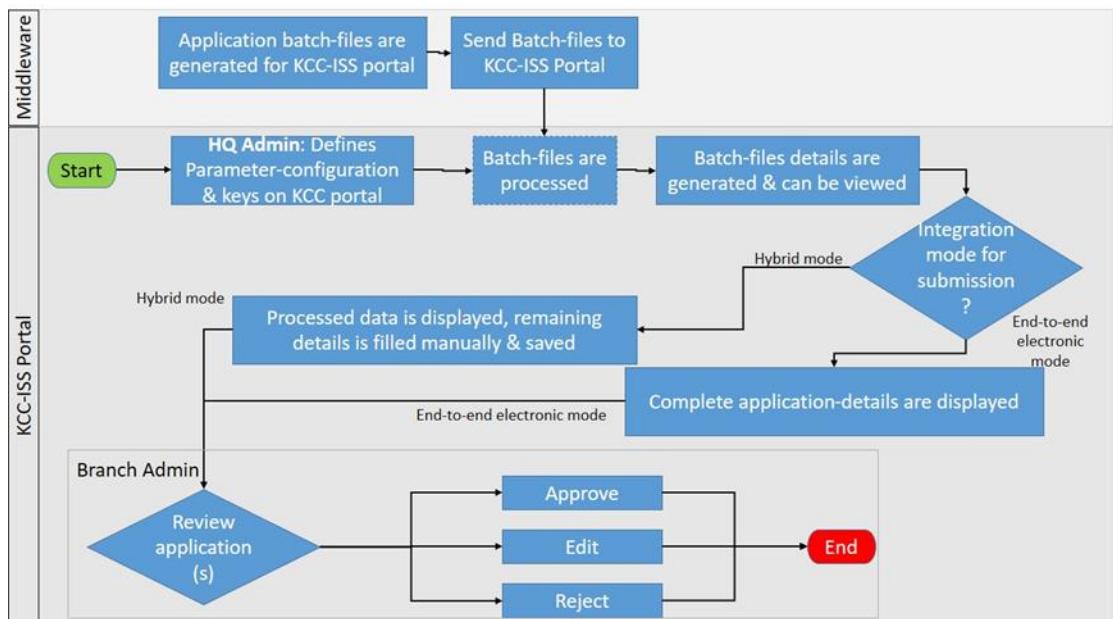


4.5.1.2. Application processing: Manual mode



In case branch manager rejects application with remarks, the application will be visible to branch-user for viewing and editing as per remarks. The branch user shall resubmit the application to branch-admin and same workflow as explained in diagram is applicable.

4.5.1.3. Application processing: End to End electronic mode & Hybrid Modes



In case branch manager rejects application with remarks, the application is visible in the login of branch user for viewing and editing as per remarks. The branch user shall resubmit the application and same workflow as explained in diagram is applicable.

5. Schema, sample payload and API response

There are two systems involved in communication. For ease of understanding and for the purpose of this document, we define

- Recipient – KCC-ISS portal of DA&FW
- Sender – CBS/Middleware of the Banks

Further auth key will be generated for encryption / decryption of pay load.

5.1. Endpoints (description level) and purpose

5.1.1. Authentication

Both the systems will communicate only through specified IPs. Further sender will generate a token/ auth key from KCC-ISS portal using their HQ Admin credentials which shall be required for authentication of transfer before transferring of each payload & however, the sender shall have a flexibility to change the token/auth key at any point in time, accordingly the payloads shall also have the changed token/auth key as input attribute for successful transfer of payloads.

The encryption algorithm (auth keys) is given in Annexure I of this document. The sender shall generate Decryption Key & Decryption IV key and shall have a flexibility to change the same at any point in time.

5.1.2. APIs to get master data

Using the APIs mentioned below, sender can extract the master data required for preparing payload data:

5.1.2.1. Get all states

Using this API, sender can know the codes and name of all states.

Endpoint: <https://fasalrin.gov.in/v1/issintegration/states>

Note: Payload data against each API will be sent in the format shown but it is received by the recipient in encrypted form. For this, the user sending payload has to encrypt the data (content inside the data node) and then send to ISS portal.

Sample Request-body is mentioned below for reference purpose:

```
{  
  "authCode": "8482a4e9-c3f7-4bb-8f3a-a8a3a7d1ed12",  
  "data": "af10cf6eefcb0ba04f4c78bfc8b30f695c79edcc252c0a381ffa6a1d9fe4ef8c"  
}
```

Sample Response-body is mentioned below for reference purpose:

```
{  
  "status": true,  
  "data": "9e42faf5c4e2e042f5a95eb7e9574d820cc8236b7b0a421d4db471213abd726bbac  
13eb1fb737fd8abb93ee03e4e5d0e2261604fcfc8b145a18374e26e116cd25e902b935c1a054c  
58769c2e2ac4c0edf8541a9ef6213bd4e6383853c799ce1fba1d609ff651237ca11db0093be8e  
a1126d78b8b075e9e9c6a92f877ba277875873127c380999e683a4a4397ed014927bfce7bb8  
b01713f384116b7474967306763b91a305275b5bf55eb18ae9fa784ae2e4c3780f844a853d5  
312eb33c002a6ee1d55e0e68db63f237132a83f6b608250725d480cdd06845e9bfa2d9dd6e4  
33e3dcf9c8eb181f3f860acaea78290a9605cac40aa88796e1fef5a7ea04414be7f4b4fc0dd6c4  
f3a908675bf7d3ebfb362f4e78c1bc0c3971ca047dfa73e5edb3e74681f5aaaf37b22a4790f21  
d1919da4f874c3003d8181e0c641805fd9959851924ab8fb6a9d36932b24614d6f8211bb534  
be75d0a2e86da4a3a1217d9f52a855fa923a001e2b13e9d23539af48cdc170c342cf3391bb9c  
bcd6d8db176d12470026532fc87febc5beb0c2b3749dc21830e383aac4aade3b8a59f3d50909  
b864dd79884f1c74bf77bdd53da9a839f04b447266b62b76379656fde2f2cbfa79891ea3148  
9c3fb7afeb3846b08e47aa894d60e42851416fe33370ec2a624b10ffa36225f276b262e4375b  
42d05e0ef84e22f8468e28f64d3c39007cd9fe7c6a72674b59275230fb1ea4cbb5de07686314  
428a05ccc9fd98f4c5637198be9d2c13b5fc9582c57384c692c9a5dc8a421f9aa8b3ad8dcbbe  
d6cca0a1cc1f10c40152a4a1e4502054f4b06eaa62d1ebe3d899a4ad8cb01fab51a795f18678c  
de8a482e082de2ca4a5774f41529b7c136c16fa852439e2c99b05e92711a77f5851dc725107c  
38625df9022e3bac2030891e47fc9e648ddd198750df03444b4da6106048139a7741057c25e  
a7dafcb99487c4b131074521ba394499704df7670e0f04ddfb1bbc58cf16cf18a7033ccc9148  
8249da303049d7f5b0b947ad9b62f7f624db2b3b48f23308b5043ec05440634566835f7eaea2  
9853767c621424e75af4e65826389fceffee5ae31e0f287a326f92ba1c93977d2e1ee13924488  
37fd97283e0c90e9cb55e767103f4973603a47925915bbe1dbcbe4e1b556194cc8a9b0392d86  
745cb02c64e378172db1d24140aa1450ed77f40c6c183f9dd72eb88a50ff3d56114790f65fa6  
0e716816d6724062913906688b93dfcecc295c9af8d472ac7bde08c8e8f0e517cde7bd83f190  
9efdc847b18be806ffed74244008ae70ed1b72d6a888fca72122920751c0d77c033988c80542  
cf22232e13baebe6f7c013b64e4b6acd8ce156bf9451ac010afb99b0e84f6cbad103aa447c30  
b830b3e881b5661267bca1d2c3e37b30221ed3c63d3b2ec4d37d4ff117309e0069448b50f35  
c241361dfb3d68aad6def825390a336835a7eb1282e70dd69ad4119a624e3627126f2ca7524  
50db3367b498c6e1fa56deb5925dd4f98de8304aa9859bc02526b7f449574587d3eb3f5aaec  
962ef641090ed762880ee71c1ad23a39df6943807efe233d6e6fab5cab4d8bcc67399f3b76
```

```

5cbd2bef41961eb0f14ebff6424508f75a70957682c5d7fa3a3da95674bcc52ffb13ac0138db9
e14cb81fb0faa63401c901564048de23db7a693b0663f72c69957832a3ab90e77bddb338fe77
16dfaccb07a594a180758c97eca060683741ba79b7c0776d6629c8e4accd40be1b9f348ce0c2
05a80a35e1df531c5b580020acdccf29cc4464b1961a217f7cd5fc7e9a1b9d16656d9356b383
30f05a986cd0ddc59a63f5b0846260d636a3712a2089f82bf715372fc83e3d8417aaf7c536c2
99139b750c481c0a1cdca9059cea2be9990d900032becb3353d193d3a61353d3295f61f2a15
1cbab0a903fa5dbb74769da4ef771c866700d6b0006f4ccfe121fb04d6cd9007c3a772857e2d
dbb7cb31844e8ec712449d9f70b158fbc281dc65cf520f4b879e3b67a7710aad02f8610a992
128e88ba36b1db6636737ed0f1bb950f0cc79f13176f2c865d18cde1b3dc2227d904d2c2350
36034e2f458c9f8a6091adb374c0d3af49a408c77e28acf9828829a2f07edf69f18b7cfe36c49
3704ccf6bba1d26ebb25d9f31658b7e8503a23a374e9948abae0be0713b4177f49b4b1b10d9
be07e9e973f3de29eba60c6d95a82281991b6d27a8dc8f737b93735365aaef44c2dd74a2aaa
ec503099e7a358235338ed6921c99b5be93d51b599fdc5b4fe45402f9d2804b993b5cc7ca0ad
78fba4591ed23d1aa2eabf509065ceb1d99c9bdc50471cba94a0db0fd26d53c5b5c171d25682
794e13fa47aa272c5eef65b961a4e7b9b7757874449ff9d765f3214064b014c932cc7939c313
9e0f586c45d80d49c48f233e5c4399568b24001240db57b1ebd3c043db2238f28152f5281daf
96595539992ce6cf3ad65d8d8e681249a3da0d307f1a5e4670c538750e1405beac4c8f1dc30b
91aac3d5360396b9a13072f",
    "error": ""
}

```

Request (HTTP POST):

```
{
    "authCode":"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",
    "data": {
        "stateCode":"00"
    }
}
```

Encrypted with auth keys

Response:

```
{
    "status":true,
    "data":{
        "states":[
            {
                "stateCode":"06",
                "stateName":"HARYANA"
            },
            {
                "stateCode":"35",
                "stateName":"ANDAMAN AND NICOBAR ISLANDS"
            }
        ]
    }
}
```

```
},
{
    "stateCode":"28",
    "stateName":"ANDHRA PRADESH"
},
{
    "stateCode":"12",
    "stateName":"ARUNACHAL PRADESH"
},
{
    "stateCode":"18",
    "stateName":"ASSAM"
},
...
]
},
"error":""
```

Note: Payload data against each API will be sent in the format shown above but it is received by the recipient in encrypted form. For this, the user sending payload has to encrypt the data (content inside the data node) and then send to ISS portal.

5.1.2.2. Get districts by state code

Using this API, sender can know the codes and name of all districts by sending a state code.

Endpoint: <https://fasalrin.gov.in/v1/issintegration/districts>

Request (HTTP POST):

```
{
    "authCode":"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",
    "data":{
        "stateCode":"09"
    }
}
```

Response:

```
{
    "status":true,
    "data":{
        "districts":[
            {
                "districtCode":"195",

```

```
        "districtName":"Ghazipur"
    },
    {
        "districtCode":"193",
        "districtName":"Ballia"
    },
    ...
]
},
"error": ""
}
```

Note: Payload data against each API will be sent in the format shown above but it is received by the recipient in encrypted form. For this, the user sending payload has to encrypt the data (content inside the data node) and then send to ISS portal.

5.1.2.3. Get locations by district code

Using this API, sender can know the codes and name of all locations by passing a district code (Taluka to village).

Endpoint: <https://fasalrin.gov.in/v1/issintegration/locations>

Request (HTTP POST):

```
{
    "authCode":"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",
    "data":{
        "districtCode":"389"
    }
}
```

Response:

```
{
    "status":true,
    "data":{
        "stateCode":"21",
        "stateName":"ODISHA",
        "districtCode":"389",
        "districtName":"Gajapati",
        "locations":[
            {
                "Level4":"PoliceStation",
                "Level4Code":"03100",
                "Level4Name":"Adva",
                "Level7":"Village",

```

```

    "Level7Code":"413460",
    "Level7Name":"Sapalaguda"
},
{
    "Level4":"PoliceStation",
    "Level4Code":"03100",
    "Level4Name":"Adva",
    "Level7":"Village",
    "Level7Code":"413461",
    "Level7Name":"Movel"
},
...
]
},
"error": ""
}

```

Note: Payload data against each API will be sent in the format shown above but it is received by the recipient in encrypted form. For this, the user sending payload has to encrypt the data (content inside the data node) and then send to ISS portal.

5.1.2.4. Get all banks

Using this API, sender can know the bank codes that has to be inserted in payload of other APIs further (mainly bank branches).

Endpoint: <https://fasalrin.gov.in/v1/issintegration/banks>

Request (HTTP POST):

```
{
    "authCode":"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",
    "data": {
        "bankCode":"00"
    }
}
```

Note: With “00” bank code, list of all banks will be obtained in response.

Response:

```
{
    "status":true,
    "data":{
        "banks":[
            {
                "bankCode":"01",
                "bankName":"State Bank of India"
            },
        ]
    }
}
```

```
{
  "bankCode":"XX",
  "bankName":"Ab Bank Limited"
},
{
  "bankCode":"XX",
  "bankName":"Abu Dhabi Commercial Bank Ltd"
},
{
  "bankCode":"XX",
  "bankName":"Agra District Co-operative Bank Agra"
},
{
  "bankCode":"XX",
  "bankName":"Ahmednagar Dccb"
},
{
  "bankCode":"XX",
  "bankName":"Ahmednagar District Central Co-op Bank Ltd"
},
{
  "bankCode":"XX",
  "bankName":"Akola DCCB"
},
...
]
},
"error": ""
}
```

Note: For representation purpose, number "XX" is mentioned. The actual number / bank code can be seen when API is called.

Payload data against each API will be sent in the format shown above but it is received by the recipient in encrypted form. For this, the user sending payload has to encrypt the data (content inside the data node) and then send to ISS portal.

5.1.2.5. Get bank branches by bank code

Using this API, sender can know the codes, name and IFSC of all branches by passing a bank code.

Endpoint: <https://fasalrin.gov.in/v1/issintegration/branches>

Request (HTTP POST):

```
{
  "authCode":"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",
  "data":{
    "bankCode":"XX"
  }
}
```

```

}

Response:
{
  "status":true,
  "data":{
    "bankCode":"XX",
    "bankName":"Kerala State Co-operative Bank",
    "branches":[
      {
        "branchCode":"6504",
        "branchName":"ERNAKULAM BRANCH",
        "ifscCode":"IBKL046KS01"
      },
      {
        "branchCode":"6508",
        "branchName":"KUNNAMKULAM EVE",
        "ifscCode":"THRS0000041"
      },
      ...
    ]
  },
  "error": ""
}

```

Note: The output shown above is sample. The request and response parameters remain same. Payload data against each API will be sent in the format shown above but it is received by the recipient in encrypted form. For this, the user sending payload has to encrypt the data (content inside the data node) and then send to ISS portal.

5.1.2.6. Get general master configuration

Using this API, sender can know the general information required to be filled in payload (Eg: social category, gender etc)

Endpoint: <https://fasalrin.gov.in/v1/issintegration/masterConfigs>

Request (HTTP POST):

```
{
  "authCode":"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",
  "data":{
    "masterCode":"00"
  }
}
```

Response:

```
{
  "status":true,
  "data":[

```

```
{
  "type":"CASTE",
  "options":[
    {
      "value":"GEN",
      "code":1
    },
    {
      "value":"OBC",
      "code":2
    },
    {
      "value":"ST",
      "code":3
    },
    {
      "value":"SC",
      "code":4
    }
  ]
},
....,
],
"error": ""
}
```

Note: Payload data against each API will be sent in the format shown above but it is received by the recipient in encrypted form. For this, the user sending payload has to encrypt the data (content inside the data node) and then send to ISS portal.

5.1.2.7. Get crop master

Using this API, Sender can know the crop related information crop code, crop name and crop category.

Endpoint: <https://fasalrin.gov.in/v1/issintegration/crops>

Request (HTTP POST):

```
{
  "authCode":"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",
  "data":{
    "cropCode":"00"
  }
}
```

Response:

```
{
  "status":true,
  "data":{
```

```
"crops": [
  {
    "cropCode": "021000100",
    "cropName": "Acid Lime",
    "categoryCode": "02",
    "categoryName": "Horti & Veg Crops"
  },
  {
    "cropCode": "010900200",
    "cropName": "African Sarson",
    "categoryCode": "01",
    "categoryName": "Agri Crops"
  },
  {
    "cropCode": "021000300",
    "cropName": "Almond",
    "categoryCode": "02",
    "categoryName": "Horti & Veg Crops"
  },
  {
    "cropCode": "021400400",
    "cropName": "Aloe Vera",
    "categoryCode": "02",
    "categoryName": "Horti & Veg Crops"
  },
  ...
],
},
"error": ""
}
```

Note: Payload data against each API will be sent in the format shown above but it is received by the recipient in encrypted form. For this, the user sending payload has to encrypt the data (content inside the data node) and then send to ISS portal.

5.1.3. Submit batch (array of applications)

Sender will submit a set of applications which are received in the form of array. Bank will generate the unique batch id. Batch id must be unique across the age i.e. it should not be duplicate at any point of time.

Batch ID shall be in below format:

Batch ID (14 Digits): <6 character DDMMYY>+<3 character Bank code as per NCIP>+< 5 digit running number>

If Today's date is 29 Aug 2022 the Batch id can be **29082202300001**

A set of 1000 or less applications will comprise a batch, each application number must have sender's unique ID across the age and within the batch. For each successfully accepted batch, the applications of the payload will be validated and for each successfully accepted application, a recipient's unique ID will also be generated.

Sender's Unique ID (19 digits):[Batch ID 14 characters] + [5 digit running ID]

Recipient unique ID (19 digit):

[last 2-digits YY of YYYY]
+ [2-digit state code as per KCC-ISS Portal]
+ [6-digit Bank-Branch code as per KCC-ISS Portal]
+ [9-digit running number]

Note: if branch code is incorrect then '99999999' will be used in recipient ID [YY+99999999+ 9-digit running number]

Depending on load and system performance, the number of applications per batch may be increased in future and accordingly communication will be sent to the concerned.

The batch will be accepted on the basis of correctness of a specified JSON. After successful acceptance of the batch, a unique batch acknowledgment id (Batch ack id) will be returned. Format for Batch ack id is as below

Batch Ack Id (14 digits): DDMMYY (6 digit) + 8 digit running number

The payload is divided into two parts:

Part 1 (Mandatory): Farmer Details (Basic Details, Residential Details, Account Details)
Part 2: Loan Detail (Loan detail, Activities)

Attribute “**RecordStatus**” signifies the completeness of the record. Banker has the option to share the records in parts.

- A. recordStatus =1 complete Farmer detail (Personal details tab, Account details tab) & Loan Detail (Financial details, Activity multi-select tab)
- B. recordStatus =2 only complete Farmer details
- C. recordStatus = 3 (Loan detail can be updated later refer - Add Loan Detail /Activities)

Endpoint: <https://fasalrin.gov.in/v1/issintegration/submitbatch>

A. For recordStatus=1

Request (HTTP POST):

```
{
  "authCode": "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",
  "data": {
    "batchId": "29082202301220",
    "financialYear": "2022-2023",
    "applications": [
      {
        "uniqueId": "2908222301220911121",
        "recordStatus": 1,
        "basicDetails": {
          "beneficiaryName": "TestName",
          "aadhaarNumber": "XXXXXXXXXXXX",
          "beneficiaryPassbookName": "TestName",
          "mobile": "XXXXXXXXXX",
          "dob": "1990-08-02",
          "gender": 2,
          "socialCategory": 2,
          "farmerCategory": 1,
          "farmerType": 2,
          "primaryOccupation": 1,
          "relativeType": 2,
          "relativeName": "Test relative"
        },
        "residentialDetails": {
          "residentialVillage": "058136",
          "residentialAddress": "Test Address",
          "residentialPincode": "567876"
        },
        "accountDetails": {
          "accountNumber": "12354689092",
          "ifsc": "PUNB0173900",
          "branchCode": "0402",
          "accountHolder": 2,
          "jointAccountHolders": [
            {
              "name": "Test name",
              "aadhaarNumber": "XXXXXXXXXXXX"
            },
            {
              "name": "Test name joint account",
              "aadhaarNumber": "XXXXXXXXXXXX"
            }
          ]
        },
        "loanDetails": {
      
```

```
"kccLoanSanctionedDate":"2022-06-17",
"kccLimitSanctionedAmount":5521,
"kccDrawingLimitForFY":5000
},
"activities":[
{
    "activityType":1,
    "loanSanctionedDate":"2022-06-17",
    "loanSanctionedAmount":6666,
    "activityRows":[
        {
            "landVillage":"413731",
            "cropCode":"011507800",
            "surveyNumber":"123",
            "khataNumber":"12",
            "landArea":5,
            "landType":2,
            "season":1
        }
    ]
},
{
    "activityType":2,
    "loanSanctionedDate":"2022-06-17",
    "loanSanctionedAmount":6666,
    "activityRows":[
        {
            "landVillage":"413731",
            "plantationCode":"021400400",
            "surveyNumber":"123",
            "khataNumber":"12",
            "plantationArea":1
        }
    ]
},
{
    "activityType":3,
    "loanSanctionedDate":"2022-06-17",
    "loanSanctionedAmount":6666,
    "activityRows":[
        {
            "landVillage":"413731",
            "liveStockType":1,
            "liveStockCode":2,
            "unitCount":23
        }
    ]
},
{
    "activityType":4,
    "loanSanctionedDate":"2022-06-17",
    "loanSanctionedAmount":6666
}
```

```
"loanSanctionedAmount":6666,
"activityRows":[
  {
    "landVillage":"413731",
    "inlandType":1,
    "totalUnits":12,
    "totalArea":2
  }
]
},
{
  "activityType":5,
  "loanSanctionedDate":"2022-06-17",
  "loanSanctionedAmount":6666,
  "activityRows":[
    {
      "landVillage":"413731",
      "marineType":3,
      "totalUnits":2
    }
  ]
},
.....,
]
},
}
```

Response:

```
{  
    "status": true,  
    "data": {  
        "status": 1,  
        "batchAckId": "13102200000014",  
        "errors": []  
    },  
    "error": ""  
}
```

B. For recordStatus=2

Request (HTTP POST)

```
{  
    "authCode": "Xd9ad29e-3ede-XXXXXX-fc19e6afdcXX",  
    "data": {
```

```

    "batchId": "17022305701234",
    "financialYear": "2022-2023",
    "applications": [
        {
            "uniqueId": "1702230570123400006",
            "recordStatus": 2,
            "basicDetails": {
                "beneficiaryName": "record status two",
                "aadhaarNumber": "273229631511",
                "beneficiaryPassbookName": "record status two",
                "mobile": "9432847402",
                "dob": "1990-08-02",
                "gender": 2,
                "socialCategory": 1,
                "farmerCategory": 1,
                "farmerType": 2,
                "primaryOccupation": 1,
                "relativeType": 2,
                "relativeName": "Test relative"
            },
            "accountDetails": {
                "accountNumber": "786286745500",
                "ifsc": "SBIN0000353",
                "branchCode": "040",
                "accountHolder": 2,
                "jointAccountHolders": [
                    {
                        "name": "Test",
                        "aadhaarNumber": "563229631511"
                    }
                ]
            }
        }
    ]
}

```

Response:

```

{
    "status": true,
    "data": {
        "status": 1,
        "batchAckId": "17022300031018",
        "errors": ""
    },
    "error": ""
}

```

Note: Payload data against each API will be sent in the format shown above but it is received by the recipient in encrypted form. For this, the user sending payload has to encrypt the data (content inside the data node) and then send to ISS portal.

C. For recordStatus=3 (Add Loan Detail /Activities)

The sender can add loan details at later stage i.e. sender can send the farmer detail earlier and loan detail at later stage but farmer details must be uploaded first. In case, loan details uploaded first without farmer details the request will be rejected.

Endpoint: <https://fasalrin.gov.in/v1/issintegration/addloandetails>

Request (HTTP POST):

```
{  
    "authCode": "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",  
    "data": {  
        "batchId": "29082202301220",  
        "financialYear": "2022-2023",  
        "applications": [  
            {  
                "preuniqueId": "2908222301220944421",  
                "recordStatus": 3,  
                "uniqueId": "2908222301220911121",  
                "farmerId": "290822123456789",  
                "loanDetails": {  
                    "kccLoanSanctionedDate": "2022-06-17",  
                    "kccLimitSanctionedAmount": 5521,  
                    "kccDrawingLimitForFY": 5000  
                },  
                "activities": [  
                    {  
                        "activityType": 1,  
                        "loanSanctionedDate": "2022-06-17",  
                        "loanSanctionedAmount": 6666,  
                        "activityRows": [  
                            {  
                                "landVillage": "413731",  
                                "cropCode": "011507800",  
                                "surveyNumber": "123",  
                                "khataNumber": "12",  
                                "landArea": 5,  
                                "landType": 2,  
                                "season": 1  
                            }  
                        ]  
                    }  
                ]  
            }  
        ]  
    }  
}
```

```
        }
    ],
},
{
    "activityType":2,
    "loanSanctionedDate":"2022-06-17",
    "loanSanctionedAmount":6666,
    "activityRows":[
        {
            "landVillage":"413731",
            "plantationCode":"021400400",
            "surveyNumber":"123",
            "khataNumber":"12",
            "plantationArea":1
        }
    ]
},
{
    "activityType":3,
    "loanSanctionedDate":"2022-06-17",
    "loanSanctionedAmount":6666,
    "activityRows":[
        {
            "landVillage":"413731",
            "liveStockType":1,
            "liveStockCode":2,
            "unitCount":23
        }
    ]
},
{
    "activityType":4,
    "loanSanctionedDate":"2022-06-17",
    "loanSanctionedAmount":6666,
    "activityRows":[
        {
            "landVillage":"413731",
            "inlandType":1,
            "totalUnits":12,
            "totalArea":2
        }
    ]
},
{
```

```
        "activityType":5,  
        "loanSanctionedDate":"2022-06-17",  
        "loanSanctionedAmount":6666,  
        "activityRows": [  
            {  
                "landVillage":"413731",  
                "marineType":3,  
                "totalUnits":2  
            }  
        ]  
    }  
}
```

Response(success):

```
{  
    "status": true,  
    "data": {  
        "status": 1,  
        "batchAckId": "13102200000014",  
        "errors": []  
    },  
    "error": ""  
}
```

Note: Payload data against each API will be sent in the format shown above but it is received by the recipient in encrypted form. For this, the user sending payload has to encrypt the data (content inside the data node) and then send to ISS portal.

preuniqueId: is the uniqueId value sent in submitbatch API, for recordStatus 2.

5.1.4.Delete application(s) by application number

Sender can send application(s) (maximum limit = 1000) by sending array of application numbers to delete the earlier sent successful applications due to any error in data. Application can be deleted only if the IS claim is not registered against the applications else error will be returned. In case Bank wants to edit an application uploaded through CBS/Middleware, the application needs to be deleted first and re-upload the application with corrected values.

Endpoint: <https://fasalrin.gov.in/v1/issintegration/deletebyapplicationnumbers>

Request (HTTP POST):

```
{  
    "authCode": "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",  
}
```

```
"data":{  
    "batchId":"20012323732737",  
    "financialYear":"2022-2023",  
    "applicationNumbers": [  
        "2210234560000000183",  
        "2210234560000000184"  
    ]  
}  
}
```

Response:

```
{  
    "status": true,  
    "data": {  
        "status": 1,  
        "batchAckId": "13102200000014",  
        "errors": []  
    },  
    "error": ""  
}
```

Note: Payload data against each API will be sent in the format shown above but it is received by the recipient in encrypted form. For this, the user sending payload has to encrypt the data (content inside the data node) and then send to ISS portal.

5.1.5. Get status by batch ack id

The output / status of batch processing shall be conveyed as a number along with its interpretation. Sender has option to view the status by entering batch ack ID. There are 4 statuses identified:

- 0 – Discarded
- 1 – Pending for processing
- 2 – Processing
- 3 – Processed (result may be either success or failure against each application. The status of each application can be viewed using 5.1.6‘Get data by batch ack id’).

Endpoint: <https://fasalrin.gov.in/v1/issintegration/batchstatus>

Request (HTTP POST):

```
{  
    "authCode": "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",  
    "data": {  
        "batchAckId": "13102200000014"  
    }  
}
```

}

Response:

```
{  
  "status": true,  
  "data": {  
    "batchAckId": "13102200000014",  
    "batchId": "29082202301220",  
    "status": 3,  
    "errors": []  
  },  
  "error": ""  
}
```

Note: Payload data against each API will be sent in the format shown above but it is received by the recipient in encrypted form. For this, the user sending payload has to encrypt the data(content inside the data node) and then send to ISS portal.

5.1.6.Get data by batch ack id

Sender can view the data of all applications processed in a particular batch ack ID. Status of each application along with data accepted by KCC-ISS portal is displayed.

- 1 – Success (For success applications, recipient's & sender's unique ids, Batch id and Batch ack id shall be returned)
- 0 – Failure (here, reason for failure is displayed with batch id, batch ack id, sender's unique id)

Endpoint: <https://fasalrin.gov.in/v1/issintegration/databybatchackid>

Request (HTTP POST):

```
{  
  "authCode": "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",  
  "data": {  
    "batchAckId": "13102200000014"  
  }  
}
```

Response:

```
{  
  "status": true,  
  "data": {  
    "batchAckId": "13102200000014",  
    "batchId": "29082202301220",  
    "status": 2,  
    "applications": [  
      {  
        "applicationId": "13102200000014",  
        "batchAckId": "13102200000014",  
        "batchId": "29082202301220",  
        "status": 2,  
        "message": "Application accepted" // This message is displayed if the application is accepted  
      }  
    ]  
  }  
}
```

```
{  
    "uniqueId": "2908222301220911121",  
    "recipientUniqueId": "2210234560000000184",  
    "applicationNumber": "221023456000000018",  
    "farmerId": "290822123456789",  
    "applicationStatus": 1,  
    "errors": []  
},  
....  
]  
,  
"error": ""  
}
```

NOTE:

1. **Farmer id:** Unique id will be generated for each farmer based on Aadhaar number against the accepted record. For one Aadhaar one unique farmer id will be generated.
2. Payload data against each API will be sent in the format shown above but it is received by the recipient in encrypted form. For this, the user sending payload has to encrypt the data (content inside the data node) and then send to ISS portal.

5.1.7.Get data by Recipient's unique ID(s)

Sender can know the status of application and view application data (maximum limit = 1000) by sending array of recipient's unique IDs.

Endpoint: <https://fasalrin.gov.in/v1/issintegration/databyrecipientuniqueids>

Request (HTTP POST):

```
{  
    "authCode": "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",  
    "data": {  
        "batchAckId": "13102200000014",  
        "recipientUniqueIds": [  
            "2210234560000000183",  
            "2210234560000000184"  
        ]  
    }  
}
```

Response:

```
{
```

```

"status":true,
"data":[
{
  "uniqueId":"290822230122091112",
  "recipientUniqueId": "2210234560000000183",
  "applicationNumber":"221023456000000018",
  "farmerId": "290822123456789",
  "status":1,
  "basicDetails":{
    "beneficiaryName":"TestName",
    "aadhaarNumber":"XXXXXXXXXXXXXX",
    "beneficiaryPassbookName":"TestName",
    "mobile":"XXXXXXXXXXX",
    "dob":"1990-08-02",
    "gender":2,
    "socialCategory":2,
    "farmerCategory":1,
    "farmerType":2,
    "primaryOccupation":1,
    "relativeType":2,
    "relativeName":"Test relative"
  },
  "residentialDetails":{
    "residentialVillage":"058136",
    "residentialAddress":"Test Address",
    "residentialPincode":"567876"
  },
  "accountDetails":{
    "accountNumber":"12354689092",
    "ifsc":"PUNB0173900",
    "branchCode":"0402",
    "accountHolder":2,
    "jointAccountHolders":[
      {
        "name":"Test name",
        "aadhaarNumber":"XXXXXXXXXXXX"
      },
      {
        "name":"Test name joint account",
        "aadhaarNumber":"XXXXXXXXXXXX"
      }
    ],
    "loanDetails":{
      "kccLoanSanctionedDate":"2022-06-17",
      "kccLimitSanctionedAmount":5521,
      "kccDrawingLimitForFY":5000
    },
    "activities":[

```

```
{
  "activityType":1,
  "loanSanctionedDate":"2022-06-17",
  "loanSanctionedAmount":6666,
  "activityRows":[
    {
      "landVillage":"413731",
      "cropCode":"011507800",
      "surveyNumber":"123",
      "khataNumber":"12",
      "landArea":5,
      "landType":2,
      "season":1
    }
  ]
},
{
  "activityType":2,
  "loanSanctionedDate":"2022-06-17",
  "loanSanctionedAmount":6666,
  "activityRows":[
    {
      "landVillage":"413731",
      "plantationCode":"021400400",
      "surveyNumber":"123",
      "khataNumber":"12",
      "plantationArea":1
    }
  ]
},
{
  "activityType":3,
  "loanSanctionedDate":"2022-06-17",
  "loanSanctionedAmount":6666,
  "activityRows":[
    {
      "landVillage":"413731",
      "liveStockType":1,
      "liveStockCode":2,
      "unitCount":23
    }
  ]
},
{
  "activityType":4,
  "loanSanctionedDate":"2022-06-17",
  "loanSanctionedAmount":6666,
  "activityRows":[
    {
      "landVillage":"413731",
      "inlandType":1,
    }
  ]
}
```

```

        "totalUnits":12,
        "totalArea":2
    }
],
{
    "activityType":5,
    "loanSanctionedDate":"2022-06-17",
    "loanSanctionedAmount":6666,
    "activityRows":[
        {
            "landVillage":"413731",
            "marineType":3,
            "totalUnits":2
        }
    ]
},
],
"error": ""
}

```

Note: Payload data against each API will be sent in the format shown above but it is received by the recipient in encrypted form. For this, the user sending payload has to encrypt the data (content inside the data node) and then send to ISS portal.

5.1.8. Get data by application number(s)

Sender can know the status of application and view application data (maximum limit = 1000) by sending array of recipient's unique IDs.

Endpoint: <https://fasalrin.gov.in/v1/issintegration/databyapplicationnumbers>

Request (HTTP POST):

```

{
    "authCode":"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",
    "data":{
        "batchAckId": "13102200000014",
        "applicationNumbers":[
            "2210234560000000181",
            "2210234560000000182"
        ]
    }
}

```

}

Response:

```
{  
    "status":true,  
    "data": [  
        {  
            "uniqueId": "290822230122091112",  
            "recipientUniqueId": "2210234560000000184",  
            "applicationNumber": "221023456000000018",  
            "farmerId": "290822123456789",  
            "status": 1,  
            "basicDetails": {  
                "beneficiaryName": "TestName",  
                "aadhaarNumber": "XXXXXXXXXXXXXX",  
                "beneficiaryPassbookName": "TestName",  
                "mobile": "XXXXXXXXXXXX",  
                "dob": "1990-08-02",  
                "gender": 2,  
                "socialCategory": 2,  
                "farmerCategory": 1,  
                "farmerType": 2,  
                "primaryOccupation": 1,  
                "relativeType": 2,  
                "relativeName": "Test relative"  
            },  
            "residentialDetails": {  
                "residentialVillage": "058136",  
                "residentialAddress": "Test Address",  
                "residentialPincode": "567876"  
            },  
            "accountDetails": {  
                "accountNumber": "12354689092",  
                "ifsc": "PUNB0173900",  
                "branchCode": "0402",  
                "accountHolder": 2,  
                "jointAccountHolders": [  
                    {  
                        "name": "Test name",  
                        "aadhaarNumber": "XXXXXXXXXXXXXX"  
                    },  
                    {  
                        "name": "Test name joint account",  
                        "aadhaarNumber": "XXXXXXXXXXXXXX"  
                    }  
                ]  
            },  
            "loanDetails": {  
                "loanAmount": 1000000,  
                "loanTenure": 12, "loanRate": 10,  
                "loanType": "Kisan Rin",  
                "loanStatus": "Approved",  
                "loanApprovalDate": "2023-08-01",  
                "loanDisbursementDate": "2023-08-15",  
                "loanRepaymentDate": "2024-07-31",  
                "loanInterestRate": 10,  
                "loanEMI": 83333.33  
            }  
        }  
    ]  
}
```

```
"kccLoanSanctionedDate":"2022-06-17",
"kccLimitSanctionedAmount":5521,
"kccDrawingLimitForFY":5000
},
"activities":[
{
    "activityType":1,
    "loanSanctionedDate":"2022-06-17",
    "loanSanctionedAmount":6666,
    "activityRows":[
        {
            "landVillage":"413731",
            "cropCode":"011507800",
            "surveyNumber":"123",
            "khataNumber":"12",
            "landArea":5,
            "landType":2,
            "season":1
        }
    ]
},
{
    "activityType":2,
    "loanSanctionedDate":"2022-06-17",
    "loanSanctionedAmount":6666,
    "activityRows":[
        {
            "landVillage":"413731",
            "plantationCode":"021400400",
            "surveyNumber":"123",
            "khataNumber":"12",
            "plantationArea":1
        }
    ]
},
{
    "activityType":3,
    "loanSanctionedDate":"2022-06-17",
    "loanSanctionedAmount":6666,
    "activityRows":[
        {
            "landVillage":"413731",
            "liveStockType":1,
            "liveStockCode":2,
            "unitCount":23
        }
    ]
},
{
    "activityType":4,
    "loanSanctionedDate":"2022-06-17",
```

```

"loanSanctionedAmount":6666,
"activityRows":[
  {
    "landVillage":"413731",
    "inlandType":1,
    "totalUnits":12,
    "totalArea":2
  }
],
{
  "activityType":5,
  "loanSanctionedDate":"2022-06-17",
  "loanSanctionedAmount":6666,
  "activityRows":[
    {
      "landVillage":"413731",
      "marineType":3,
      "totalUnits":2
    }
  ]
},
],
"error": ""
}

```

5.1.9. Get data by Sender's unique ID(s)

Sender can know the status of application and view application data (maximum limit = 1000) by sending array of recipient's unique Ids.

Endpoint: <https://fasalrin.gov.in/v1/issintegration/databysenderuniqueids>

Request (HTTP POST):

```
{
  "authCode":"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",
  "data":{
    "batchAckId": "13102200000014",
    "uniqueIds":[
      "290822230122091112"
    ]
  }
}
```

Response:

```
{
  "status":true,
```

```

"data": [
  {
    "uniqueId": "290822230122091112",
    "applicationNumber": "221023456000000018",
    "farmerId": "290822123456789",
    "status": 1,
    "basicDetails": {
      "beneficiaryName": "TestName",
      "aadhaarNumber": "XXXXXXXXXXXXXX",
      "beneficiaryPassbookName": "TestName",
      "mobile": "XXXXXXXXXXX",
      "dob": "1990-08-02",
      "gender": 2,
      "socialCategory": 2,
      "farmerCategory": 1,
      "farmerType": 2,
      "primaryOccupation": 1,
      "relativeType": 2,
      "relativeName": "Test relative"
    },
    "residentialDetails": {
      "residentialVillage": "058136",
      "residentialAddress": "Test Address",
      "residentialPincode": "567876"
    },
    "accountDetails": {
      "accountNumber": "12354689092",
      "ifsc": "PUNB0173900",
      "branchCode": "0402",
      "accountHolder": 2,
      "jointAccountHolders": [
        {
          "name": "Test name",
          "aadhaarNumber": "XXXXXXXXXXXXXX"
        },
        {
          "name": "Test name joint account",
          "aadhaarNumber": "XXXXXXXXXXXXXX"
        }
      ]
    },
    "loanDetails": {
      "kccLoanSanctionedDate": "2022-06-17",
      "kccLimitSanctionedAmount": 5521,
      "kccDrawingLimitForFY": 5000
    },
    "activities": [
      {
        "activityType": 1,
        "loanSanctionedDate": "2022-06-17",
      }
    ]
  }
]

```

```

"loanSanctionedAmount":6666,
"activityRows":[
  {
    "landVillage":"413731",
    "cropCode":"011507800",
    "surveyNumber":"123",
    "khataNumber":"12",
    "landArea":5,
    "landType":2,
    "season":1
  }
]
},
{
  "activityType":2,
  "loanSanctionedDate":"2022-06-17",
  "loanSanctionedAmount":6666,
  "activityRows":[
    {
      "landVillage":"413731",
      "plantationCode":"021400400",
      "surveyNumber":"123",
      "khataNumber":"12",
      "plantationArea":1
    }
  ]
},
{
  "activityType":3,
  "loanSanctionedDate":"2022-06-17",
  "loanSanctionedAmount":6666,
  "activityRows":[
    {
      "landVillage":"413731",
      "liveStockType":1,
      "liveStockCode":2,
      "unitCount":23
    }
  ]
},
{
  "activityType":4,
  "loanSanctionedDate":"2022-06-17",
  "loanSanctionedAmount":6666,
  "activityRows":[
    {
      "landVillage":"413731",
      "inlandType":1,
      "totalUnits":12,
      "totalArea":2
    }
  ]
}

```

```

        ],
      },
      {
        "activityType":5,
        "loanSanctionedDate":"2022-06-17",
        "loanSanctionedAmount":6666,
        "activityRows":[
          {
            "landVillage":"413731",
            "marineType":3,
            "totalUnits":2
          }
        ]
      }
    ],
    "error": ""
  }

```

Note: Payload data against each API will be sent in the format shown above but it is received by the recipient in encrypted form. For this, the user sending payload has to encrypt the data (content inside the data node) and then send to ISS portal.

5.1.10. Get data by multiple batch ack id

Sender can view the data of all applications processed with multiple batch ack IDs. Status of each application along with data accepted by KCC-ISS portal is displayed.

1 – Success (For success applications, recipient's & sender's unique ids, Batch id and Batch ack id shall be returned)

0 – Failure (here, reason for failure is displayed with batch id, batch ack id, sender's unique id)

recordType has 3 values:-

- 1 - All records
- 2 - Only successful records
- 3 - Only failed records

Note that maximum allowed size of batchAckIds Array is 20.

Endpoint: <https://fasalrin.gov.in/v1/issintegration/databymultiplebatchackid>

Request (HTTP POST):

```

{
  "authCode":"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",
  "data":{
    "financialYear":"2022-2023",

```

```
"batchAckIds":["15072300017931","22052300017923"],  
"recordType":1  
}  
}
```

Response:

```
{  
    "status":true,  
    "data": [  
  
        {  
            "batch_ack_id": "15072300017931",  
            "batch_status": 3,  
            "records": [  
                {  
                    "uniqueId": "1507230061236961181",  
                    "recordStatus": "1",  
                    "application_status": 1,  
                    "farmerId": "23002780507",  
                    "recipient_unique_id": "2199999999002850649"  
                },  
                {  
                    "uniqueId": "1507230061236961182",  
                    "recordStatus": "1",  
                    "application_status": 0,  
                    "errors": "Resident Village Code is not in Master; Land Village Code is not in  
Master; ",  
                    "recipient_unique_id": "2199999999002850650"  
                }  
            ],  
            {  
                "batch_ack_id": "22052300017923",  
                "batch_status": 3,  
                "records": [  
                    {  
                        "uniqueId": "1507230061236961181",  
                        "recordStatus": "1",  
                        "application_status": 1,  
                        "farmerId": "23002780507",  
                        "recipient_unique_id": "2199999999002850649"  
                    },  
                    {  
                        "uniqueId": "1507230061236961182",  
                        "recordStatus": "1",  
                        "application_status": 0,  
                        "errors": "Resident Village Code is not in Master; Land Village Code is not in  
Master; ",  
                        "recipient_unique_id": "2199999999002850650"  
                    }  
                ]  
            }  
        }  
    ]  
}
```

```
        "recordStatus": "1",
        "application_status": 0,
        "errors": "Resident Village Code is not in Master; Land Village Code is not in
Master; ",
        "recipient_unique_id": "2199999999002850650"
    }
]
}
],
{
    "error": ""
}
```

5.1.11. Submit Claim Batch

With this API, banks can submit their claim-details in the form of batch(s).

Note:

1. Kindly pass 1 for COMPLETE and 2 for Partial-claim for submissionType node
(kindly refer schema at section 6.2 for more details)
2. Farmer-mobile and aadhar of Joint-account holder must be saved on system, before
submitting claims (refer 5.1.11.1)

Endpoint: <https://fasalrin.gov.in/v1/issintegration/submitclaimbatch>

Method: HTTPS POST

Request:

```
{
    "authCode": "ea48d593-a6de-xxxx-xxxx-xxxxxxxxxxxx",
    "data": {
        "batchId": "03082300190087",
        "financialYear": "2022-2023",
        "applications": [
            {
                "uniqueId": "0308230019008765431",
                "loanApplicationNumber": "2227008559000623231",
                "claimType": "IS",
                "submissionType": 1,
                "firstLoanDisbursalDate": "2022-09-01",
                "loanRepaymentDate": "2023-04-21",
                "maxWithdrawalAmount": 900000,
                "applicableISAmount": 1200
            },
            {

```

```
"uniqueId":"0308230019008765433",
"loanApplicationNumber":"2227008559000600001",
"claimType":"PRI",
"submissionType":1,
"isEligibleForPRI":1,
"firstLoanDisbursalDate":"2022-09-01",
"loanRepaymentDate":"2023-04-21",
"maxWithdrawalAmount":800000,
"applicablePRIAMount":1200
},
{
"uniqueId":"0308230019008765432",
"loanApplicationNumber":"2227008559000623231",
"claimType":"PRI",
"isEligibleForPRI":0
},
{
"uniqueId":"0308230019008765481",
"loanApplicationNumber":"2127008559000623231",
"claimType":"IS",
"submissionType":2,
"isRollover": 1,
"referenceClaimNumber": "21164910000000663",
"referenceClaimFY": "2021-2022",
"loanRepaymentDate":"2022-08-21",
"maxWithdrawalAmount":900000,
"applicableISAMount":1200
},
{
"uniqueId":"0308230019008765483",
"loanApplicationNumber":"2227008559000600221",
"claimType":"PRI",
"submissionType":2,
"isRollover": 1,
"referenceClaimNumber": "22162510000006163",
"referenceClaimFY": "2022-2023",
"isEligibleForPRI":1,
"loanRepaymentDate":"2022-03-21",
"maxWithdrawalAmount":800000,
"applicablePRIAMount":1200
}
]
}
```

}

Response:

```
{  
    "status": true,  
    "data": {  
        "status": 1,  
        "batchAckId": "03052300000021",  
        "errors": ""  
    },  
    "error": ""  
}
```

5.1.11.1. Update farmer-mobile/joint Account Holders

Through this API, banks can submit the farmer's mobile and aadhar details of Joint Account holder, through the API (**mandatory prior to claim-submission**).

Mandatory: uniqueId, farmerId, branchCode, accountNumber.

Endpoint: <https://fasalrin.gov.in/v1/issintegration/submitupdatebatch>

Method: HTTPS POST

Request:

```
{  
    "authCode": "362434a9-07d7-46a6-b3d5-aba90f24ffcf",  
    "data": {  
        "financialYear": "2021-2022",  
        "batchId": "13122306454123",  
        "applications": [  
            {  
                "uniqueId": "1312230645412376123",  
                "farmerId": "23002810362",  
                "mobile": "8712345423",  
                "branchCode": "165617",  
                "accountNumber": "UPDATEINFO101",  
                "jointAccountHolders": [  
                    {  
                        "name": "GENDRAM RAWAT",  
                        "aadhaarNumber": "871234541232"  
                    },  
                    {  
                        "name": "Devram Anaji Gaikwad",  
                        "aadhaarNumber": "341234542312"  
                    }  
                ]  
            }  
        ]  
    }  
}
```

```
        }
    ]
}
]
}
}
```

Sample response:

```
{
  "status": true,
  "data": {
    "status": 1,
    "batchAckId": "05012400000273",
    "errors": ""
  },
  "error": ""
}
```

5.1.12. Get Claim data by Batch Acknowledgement ID

Endpoint: <https://fasalrin.gov.in/v1/issintegration/claimdatabybatchackid>

Method: HTTPS POST

Request:

```
{
  "authCode": "86d3e73b-xxxx-xxxx-xxxx-xxxxxxxxxx",
  "data": {
    "batchAckId": "19042300000012"
  }
}
```

Response:

```
{
  "status":true,
  "data":{
    "batchAckId":"03082300000039",
    "batchId":"03082300190087",
    "status":3,
    "applications":[
      {
        "uniqueId":"0308230019008765431",
        "claimType":"IS",
        "submissionType":2,
        "loanApplicationNumber":"2227008559000623231",
        "applicationStatus":0,
        "errors":"Max withdrawal amount is more than total Sanctioned Amount;Applicable IS is more than max allowed IS;"
      }
    ]
  }
}
```

```
"recipientUniqueID":"22008559000000016"
},
{
  "uniqueId":"0308230019008765432",
  "claimType":"PRI",
  "submissionType":1,
  "loanApplicationNumber":"2227008559000623231",
  "applicationStatus":1,
  "errors":"",
  "recipientUniqueID":"22008559000000014",
  "claimNumber":"22008559000000709"
}
]
},
"error":""
```

5.1.13 Claim Batch Status

Endpoint: <https://fasalrin.gov.in/v1/issintegration/claimbatchstatus>

Method: HTTPS POST

Request:

```
{
  "authCode": "86d3e73b-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
  "data": {"batchAckId": "19042300000012"}
}
```

Response:

```
{
  "status": true,
  "data": {
    "batchAckId": "19042300000012",
    "batchId": "19042300110001",
    "status": 3
  },
  "error": ""
}
```

5.1.14 Claim data by Recipient Unique ID

Endpoint: <https://fasalrin.gov.in/v1/issintegration/claimdatabyrecipientuniqueids>

Method: HTTPS POST

Request:

```
{
  "authCode": "86d3e73b-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
```

```

"data": {
    "batchAckId": "19042300000012",
    "recipientUniqueIds": [
        "22008559000000014",
        "22008559000000016",
        "22008559000000017"
    ]
}
}

```

Response:

```

{
    "status":true,
    "data":[
        {
            "uniqueId": "1904230011000157001",
            "claimType": "PRI",
            "isEligibleForPRI": 0,
            "loanApplicationNumber": "2227008559000623231",
            "recipientUniqueId": "22008559000000014",
            "claimNumber": "22008559000000709",
            "error": ""
        },
        {
            "uniqueId": "1904230011000157002",
            "claimType": "IS",
            "submissionType": 1,
            "loanApplicationNumber": "2227008559000623231",
            "firstLoanDisbursalDate": "2022-09-01",
            "loanRepaymentDate": "2023-04-21",
            "maxWithdrawalAmount": 900000,
            "applicableISAmount": 1200,
            "recipientUniqueId": "22008559000000016",
            "claimNumber": null,
            "error": "Max withdrawal amount is more than total Sanctioned Amount;Applicable IS is more than max allowed IS;"
        },
        {
            "uniqueId": "1904230011000157003",
            "claimType": "IS",
            "submissionType": 2,
            "isRollover": 1,
            "referenceClaimFY": "2022-2023",
            "referenceClaimNumber": "22164910000000667",
            "loanApplicationNumber": "2227008559000623231",
            "loanRepaymentDate": "2022-10-21",
            "maxWithdrawalAmount": 900000,
            "applicableISAmount": 1200,
            "recipientUniqueId": "22008559000000017",
            "claimNumber": "22164910000080213",
            "error": null
        }
    ]
}

```

```
        "error":""  
    }  
],  
"error":""  
}
```

5.1.15 Claim data by Sender's Unique ID

Endpoint: <https://fasalrin.gov.in/v1/issintegration/claimdatabysenderuniqueids>

Method: HTTPS POST

Request:

```
{  
    "authCode": "86d3e73b-xxxx-xxxx-xxxx-xxxxxxxxxxxx",  
    "data": {  
        "batchAckId": "19042300000012",  
        "uniqueIds": [  
            "1904230011000157001",  
            "1904230011000157002",  
            "1904230011000157003"  
        ]  
    }  
}
```

Response:

```
{  
    "status":true,  
    "data": [  
        {  
            "uniqueId": "1904230011000157001",  
            "claimType": "PRI",  
            "isEligibleForPRI": 0,  
            "loanApplicationNumber": "2227008559000623231",  
            "recipientUniqueId": "22008559000000014",  
            "claimNumber": "22008559000000709",  
            "error": ""  
        },  
        {  
            "uniqueId": "1904230011000157002",  
            "claimType": "IS",  
            "submissionType": 1,  
            "loanApplicationNumber": "2227008559000623231",  
            "firstLoanDisbursalDate": "2022-09-01",  
            "loanRepaymentDate": "2023-04-21",  
            "maxWithdrawalAmount": 900000,  
            "applicableISAmount": 1200,  
            "recipientUniqueId": "22008559000000016",  
            "claimNumber": null,  
            "error": ""  
        }  
    ]  
}
```

```

    "error":"Max withdrawal amount is more than total Sanctioned Amount;Applicable IS is
    more than max allowed IS;"}
    {
      "uniqueId":"1904230011000157003",
      "claimType":"IS",
      "submissionType":2,
      "isRollover":1,
      "referenceClaimFY":"2022-2023",
      "referenceClaimNumber":"22164910000000667",
      "loanApplicationNumber":"2227008559000623231",
      "loanRepaymentDate":"2022-10-21",
      "maxWithdrawalAmount":900000,
      "applicableISAmount":1200,
      "recipientUniqueId":"22008559000000017",
      "claimNumber":"22164910000080213",
      "error":""
    }
  ],
  "error": ""
}

```

5.1.16 Claim data by Claim-number

Endpoint: <https://fasalrin.gov.in/v1/issintegration/claimdatabyclaimnumbers>

Method: HTTPS POST

Response:

```
{
  "authCode": "86d3e73b-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
  "data": {
    "batchAckId": "1904230000012",
    "claimNumbers": [
      "2216561700000107","2216561700000108","2216561700000109"
    ]
  }
}
```

Response:

```
{
  "status":true,
  "data":[
    {
      "uniqueId":"0308230019008765432",
      "claimType":"PRI",
      "isEligibleForPRI":0,
      "loanApplicationNumber":"2227008559000623231",
      "recipientUniqueId":"22008559000000014",
```

```

    "claimNumber":"22165617000000107",
    "error":"",
},
{
    "uniqueId":"0308230019008765431",
    "claimType":"IS",
    "submissionType":1,
    "loanApplicationNumber":"2227008559000623231",
    "firstLoanDisbursalDate":"2022-09-01",
    "loanRepaymentDate":"2023-04-21",
    "maxWithdrawalAmount":900000,
    "applicableISAmount":1200,
    "recipientUniqueId":"22008559000000016",
    "claimNumber":"22165617000000108",
    "error":"",
},
{
    "uniqueId":"0308230019008765465",
    "claimType":"PRI",
    "submissionType":2,
    "isEligibleForPRI":1,
    "isRollover":1,
    "loanApplicationNumber":"2227008559000623231",
    "loanRepaymentDate":"2023-04-21",
    "maxWithdrawalAmount":900000,
    "referenceClaimNumber":"21165617000020301",
    "referenceClaimFY":"2021-2022",
    "applicablePRIAmount":1200,
    "recipientUniqueId":"22008559000000017",
    "claimNumber":"22165617000000109",
    "error":"",
}
],
"error":"",
}

```

5.1.17 Delete by Claim Number

Endpoint: <https://fasalrin.gov.in/v1/issintegration/deletebyclaimnumbers>

Method: HTTPS POST

Request:

```
{
    "authCode": "86d3e73b-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
    "data": {
        "batchId": "21042300110678",
        "financialYear": "2022-2023",
        "claimNumbers": [

```

```
"22165617000000253",
"22165617000000890"
]
}
}
```

Response:

```
{
  "status": true,
  "data": {
    "status": 1,
    "batchAckId": "21042300000032",
    "errors": ""
  },
  "error": ""
}
```

5.1.18 Get Claim data by Multiple Batch Acknowledgement ID

Endpoint: <https://fasalrin.gov.in/v1/issintegration/claimdatabymultiplebatchackid>

Sender can view the data of all applications processed with multiple batch ack IDs. Status of each application along with data accepted by KCC-ISS portal is displayed.

1 – Success (For success applications, recipient's & sender's unique ids, Batch id and Batch ack id shall be returned)

0 – Failure (here, reason for failure is displayed with batch id, batch ack id, sender's unique id)

recordType has 3 values :-

1 - All records

2 - Only successful records

3- Only failed records

Note that maximum allowed size of batchAckIds Array is 20.

Method: HTTPS POST

Request:

```
{
  "authCode": "2713bb57-1aca-4671-9fdd-24045d36fd8f",
  "data": {
    "financialYear": "2022-2023",
  }
}
```

```
        "batchAckIds":["07092300000076","06092300000046"] ,  
        "recordType":1  
    }  
}
```

Response:

```
{  
    "status": true,  
    "data": [  
        {  
            "batchAckID": "06092300000046",  
            "batchStatus": 3,  
            "records": [  
                {  
                    "uniqueId": "0609230029300212345",  
                    "claimType": "PRI",  
                    "loanApplicationNumber": "2103164910002277212",  
                    "submissionType": "2",  
                    "applicationStatus": 0,  
                    "errors": "referenceClaimNo is not approved;",  
                    "recipientUniqueID": "22164910000000019"  
                },  
                {  
                    "uniqueId": "0609230029300212346",  
                    "claimType": "PRI",  
                    "loanApplicationNumber": "2103164910002277214",  
                    "submissionType": "2",  
                    "applicationStatus": 1,  
                    "errors": "",  
                    "claimNumber": "22164910000000763",  
                    "recipientUniqueID": "22164910000000021"  
                }  
            ]  
        },  
        {  
            "batchAckID": "07092300000076",  
            "batchStatus": 3,  
            "records": [  
                {  
                    "uniqueId": "0609230029300212345",  
                    "claimType": "PRI",  
                    "loanApplicationNumber": "2103164910002277212",  
                    "submissionType": "2",  
                    "applicationStatus": 0,  
                    "errors": "referenceClaimNo is not approved;"  
                }  
            ]  
        }  
    ]  
}
```

```
        "recipientUniqueID": "22164910000000019"
    }
]
}
],
"error": ""
}
```

5.1.19 Configuration of payload attributes

In case user selects full automation mode for generating applications in KCC / ISS portal, the mandatory fields are shown in ‘Required’ column in DB schema. In case user selects hybrid mode, the user has option to choose which fields are to be sent through CBS integration first and which fields shall be updated manually later. For more detail please refer Annexure II (schema).

Bank may delete and create new field-configurations (hybrid mode) as per their requirement, subject to the condition that there are no batches under processing.

5.2. Schema

The **schema** of KCC-ISS portal is mentioned in **Annexure-II**. The schema and payload are given for understanding the DB design of KCC-ISS portal, analyzing how to create payload and send to KCC-ISS portal. Further discussion can be done jointly for arriving at a solution to integrate portal with CBS/Middleware.

6. Appendix

This document is prepared based on discussions held with technical team and management of KCC-ISS portal nominated by DA&FW.

ANNEXURE-I

Encryption of payload data (content inside data node)

Content inside data node of payload must be encrypted by AES with following algorithm. For reference, the AES algorithm tutorial and testing link is given below:

<https://www.devglan.com/online-tools/aes-encryption-decryption>

<https://levelup.gitconnected.com/implementing-aes-encryption-in-node-js-and-c-from-scratch-6ee7b47ae6d4>

The algorithm required two keys as mentioned below:

Encryption and decryptionKey (AES 256 CBC)
= <ED9466BBE4B0C4EC9E6431693XXXXXXXX>

Text format = Hex

Encryption and decryption IV = 9493508806CXXXXXX

authToken = the authentication key generated by KCC-ISS portal

Encryption and decryption Key (AES 256 CBC), Encryption and decryption IV can be generated and changed at any point of time by the Bank HO admin through its KCC-ISS portal login.

6.1. Sample encryption and decryption code in C#

```
using System;
using System.IO;
using System.Security.Cryptography;
using System.Text;

namespace Aes_Example
{
    class AesExample
    {
        public static void Main()
        {
            string original = "your string here";
            string decryptText = "decrypt text to check";
            const string Key = "Secret Key";
            const string IV = "IV String";

            using (AesCryptoServiceProvider myAes = new AesCryptoServiceProvider())
            {
                myAes.IV = UTF8Encoding.UTF8.GetBytes(IV);
                myAes.Key = UTF8Encoding.UTF8.GetBytes(Key);
            }
        }
    }
}
```

```

byte[] encrypted = Encrypt(original, myAes.Key, myAes.IV);
string hex = BitConverter.ToString(encrypted).Replace("-", "");

byte[] decryptbyte = StringToByteArray(decryptText);
string roundtrip = Decrypt(decryptbyte, myAes.Key, myAes.IV);

}

public static byte[] StringToByteArray(String hex)
{
    int NumberChars = hex.Length;
    byte[] bytes = new byte[NumberChars / 2];
    for (int i = 0; i < NumberChars; i += 2)
        bytes[i / 2] = Convert.ToByte(hex.Substring(i, 2), 16);
    return bytes;
}

static byte[] Encrypt(string plainText, byte[] Key, byte[] IV)
{
    if (plainText == null || plainText.Length <= 0)
        throw new ArgumentNullException("plainText");
    if (Key == null || Key.Length <= 0)
        throw new ArgumentNullException("Key");
    if (IV == null || IV.Length <= 0)
        throw new ArgumentNullException("IV");
    byte[] encrypted;

    using (AesCryptoServiceProvider aesAlg = new AesCryptoServiceProvider())
    {
        aesAlg.Key = Key;
        aesAlg.IV = IV;

        ICryptoTransform encryptor = aesAlg.CreateEncryptor(aesAlg.Key, aesAlg.IV);

        using (MemoryStream msEncrypt = new MemoryStream())
        {
            using (CryptoStream csEncrypt = new CryptoStream(msEncrypt, encryptor,
                CryptoStreamMode.Write))
            {
                using (StreamWriter swEncrypt = new StreamWriter(csEncrypt))
                {
                    swEncrypt.Write(plainText);
                }
                encrypted = msEncrypt.ToArray();
            }
        }
    }
}

```

```

        }

    }

    return encrypted;
}

static string Decrypt(byte[] cipherText, byte[] Key, byte[] IV)
{
    if (cipherText == null || cipherText.Length <= 0)
        throw new ArgumentNullException("cipherText");
    if (Key == null || Key.Length <= 0)
        throw new ArgumentNullException("Key");
    if (IV == null || IV.Length <= 0)
        throw new ArgumentNullException("IV");

    string plaintext = null;

    using (AesCryptoServiceProvider aesAlg = new AesCryptoServiceProvider())
    {
        aesAlg.Key = Key;
        aesAlg.IV = IV;
        ICryptoTransform decryptor = aesAlg.CreateDecryptor(aesAlg.Key, aesAlg.IV);
        using (MemoryStream msDecrypt = new MemoryStream(cipherText))
        {
            using (CryptoStream csDecrypt = new CryptoStream(msDecrypt, decryptor,
                CryptoStreamMode.Read))
            {
                using (StreamReader srDecrypt = new StreamReader(csDecrypt))
                {
                    plaintext = srDecrypt.ReadToEnd();
                }
            }
        }
    }

    return plaintext;
}
}

```

Sample encryption and decryption code in Java

```
import javax.crypto.Cipher;  
import javax.crypto.SecretKey;
```

```
import javax.crypto.SecretKeyFactory;
import javax.crypto.spec.IvParameterSpec;
import javax.crypto.spec.PBEKeySpec;
import javax.crypto.spec.SecretKeySpec;
import java.nio.charset.StandardCharsets;
import java.security.spec.KeySpec;
import java.util.Base64;

public class Main {
    private static final String SECRET_KEY = "Secret Key";
    private static final String ivString = "IV string";
    private static final byte[] keyParse = SECRET_KEY.getBytes();

    public static String byteArrayToHexString(byte[] bytes)
    {
        StringBuffer buffer = new StringBuffer();
        for(int i=0; i<bytes.length; i++)
        {
            if(((int)bytes[i] & 0xff) < 0x10)
                buffer.append("0");
            buffer.append(Long.toString((int) bytes[i] & 0xff, 16));
        }
        return buffer.toString();
    }

    public static String encrypt(String strToEncrypt) {
        try {
            byte[] iv = ivString.getBytes();
            IvParameterSpec ivspec = new IvParameterSpec(iv);
            SecretKeyFactory factory = SecretKeyFactory.getInstance("PBKDF2WithHmacSHA256");
            Cipher cipher = Cipher.getInstance("AES/CBC/PKCS5Padding");
            SecretKeySpec key = new SecretKeySpec(keyParse, "AES");
            cipher.init(Cipher.ENCRYPT_MODE, key, ivspec);
            return
byteArrayToHexString(cipher.doFinal(strToEncrypt.getBytes(StandardCharsets.UTF_8)));
        } catch (Exception e) {
            System.out.println("Error while encrypting: " + e.toString());
        }
        return null;
    }

    public static String decrypt(String strToDecrypt) {
        try {
```

```
byte[] iv = ivString.getBytes();
IvParameterSpec ivspec = new IvParameterSpec(iv);

Cipher cipher = Cipher.getInstance("AES/CBC/PKCS5PADDING");
SecretKeySpec key = new SecretKeySpec(keyParse, "AES");
cipher.init(Cipher.DECRYPT_MODE, key, ivspec);

byte[] val = new byte[strToDecrypt.length() / 2];
for (int i = 0; i < val.length; i++) {
    int index = i * 2;
    int j = Integer.parseInt(strToDecrypt.substring(index, index + 2), 16);
    val[i] = (byte) j;
}

return new String(cipher.doFinal(val));
} catch (Exception e) {
    System.out.println("Error while decrypting: " + e.toString());
}
return null;
}

}
```

6.2. Sample encryption and decryption code in Javascript

```
var crypto = require('crypto');

const encryptionType = 'aes-256-cbc';
const encryptionEncoding = 'hex';
const bufferEncryption = 'utf-8';
module.exports = {
    encrypt: (text, eskey, esIv )=> {
        const key = Buffer.from(eskey, bufferEncryption);
        const iv = Buffer.from(esIv, bufferEncryption);
        const cipher = crypto.createCipheriv(encryptionType, key, iv);
        let encrypted = cipher.update(text, bufferEncryption, encryptionEncoding);
        encrypted += cipher.final(encryptionEncoding);
        return encrypted;
    },
    decrypt: (encrypted, eskey, esIv) => {
        const buff = Buffer.from(encrypted, encryptionEncoding);
        const key = Buffer.from(eskey, bufferEncryption);
        const iv = Buffer.from(esIv, bufferEncryption);
        const decipher = crypto.createDecipheriv(encryptionType, key, iv);
```

```

const deciphered = decipher.update(buff) + decipher.final();
return deciphered;
}
}

```

6.3. Online utility to encrypt/decrypt the request

This utility is only for reference purpose for the bank teams to match the encrypted and decrypted text.

- Get the IV Key from ISS Encrypt & Decrypt keys

Index	IV Key	Secret Key	Created On	Valid Till
1	ILieMwfBFZPVSBcz	A3CfAllVmjqHBgalc1Y3cqm3pbBJrEsB	14/11/2022	13/05/2023
2	U8KT8zK95oNcXgvx	JPjlbIKNdpie3hbsEzww0ZuoGGZ4epc	13/11/2022	14/11/2022
3	dOB78jXoqXy2XQdo	WuQYHrw4D7gZUYnnC2XSYNpCF1Cn6UKO	13/11/2022	13/11/2022
4	aiNlikJuJUGPFKfa	YmtSmk6sdkSSAxjQzMNmXjZDuCkqi8S	01/11/2022	13/11/2022

- Get the Secret key from ISS Encrypt & Decrypt keys

Index	IV Key	Secret Key	Created On	Valid Till
1	ILieMwfBFZPVSBcz	A3CfAllVmjqHBgalc1Y3cqm3pbBJrEsB	14/11/2022	13/05/2023
2	U8KT8zK95oNcXgvx	JPjlbIKNdpie3hbsEzww0ZuoGGZ4epc	13/11/2022	14/11/2022
3	dOB78jXoqXy2XQdo	WuQYHrw4D7gZUYnnC2XSYNpCF1Cn6UKO	13/11/2022	13/11/2022
4	aiNlikJuJUGPFKfa	YmtSmk6sdkSSAxjQzMNmXjZDuCkqi8S	01/11/2022	13/11/2022

- Enter the below details under “AES Online Encryption”
 - Enter the “**data**” node” from the Main Request under “Enter text to be Encrypted” field
 - Select “**CBC**” from “Select Cipher Mode of Encryption” field
 - Select “**256**” from Key Size in Bits” field
 - Enter the **IV Key** under “Enter IV (Optional)” field
 - Enter the **Secret key** under “Enter secret key” field

AES Online Encryption

Enter text to be Encrypted

```
{
  "batchId": "1281553592233",
  "financialYear": "2022-2023",
  "applications": [
    {
      "applicationId": "12345678901234567890123456789012"
    }
  ]
}
```

Select Cipher Mode of Encryption

Key Size in Bits

Enter IV (Optional)

Enter Secret Key

AES Online Decryption

Enter text to be Decrypted

Input Text Format: Base64 Hex

Select Cipher Mode of Decryption

Enter IV Used During Encryption(Optional)

Key Size in Bits

Enter Secret Key used for Encryption

- f) Select the “Output Text Format” as “Hex” and Click on “Encrypt” button and verify that the Request Is Encrypted successfully and displayed under “AES Encrypted Output”

Key Size in Bits

Enter IV (Optional)

Enter Secret Key

Output Text Format: Base64 Hex

Encrypt

AES Encrypted Output:

Result goes here

Enter IV Used During Encryption(Optional)

Key Size in Bits

Enter Secret Key used for Encryption

Decrypt

AES Decrypted Output (Base64):

Base64 encoded result goes here

Decode to Plain Text

Enter IV (Optional)

Enter Secret Key

Output Text Format: Base64 Hex

Encrypt

AES Encrypted Output:

```
17864DD2186A24629D2733454AFCAE035630
7A886D3EA981C0C2C5A7C5A31C3867C14DA
85DC81570148C198D86329695557C3D09852
39C6AE6CA63FF9A6DD7DE7F795A6C860002
8E81F2ED9ACEE8C68CA907F9448EBA50D62A
```

INITIALIZATION VECTOR

Key Size in Bits

256

Enter Secret Key used for Encryption

Decrypt

AES Decrypted Output (Base64):

Decode to Plain Text

Get the Encrypted Output and Put in the Body for “Submit Batch request” in POST MAN Tool

The screenshot shows the Postman interface with a POST request to 'submitbatch'. The 'Body' tab is selected, displaying the following JSON payload:

```
{
  "authCode": "1a2b3a95b-d5dc-42b0-9e71-0e966a37811d",
  "data": "17864DD2186A24629D2733454AFCAE035630
7A886D3EA981C0C2C5A7C5A31C3867C14DA
85DC81570148C198D86329695557C3D09852
39C6AE6CA63FF9A6DD7DE7F795A6C860002
8E81F2ED9ACEE8C68CA907F9448EBA50D62A"
}
```

The response status is 200 OK.

Submit the Request and verify that response is received in encrypted form from POST MAN Tool

```

POST districts
POST masterConfigs
POST locations
POST branches
POST crops
POST submitbatch
POST databybatchid
POST databyrecipientuniqueids
POST addloandetails
POST Bank
POST DeleteRecipientUniquid
POST BatchStatus
POST DatabySendersUniqueID
POST EncryptRequest
POST decryptRequest
POST SubmittingBatch

Params Authorization Headers (10) Body Pre-request Script Tests Settings Cookies Beautify
none form-data x-www-form-urlencoded raw binary GraphQL JSON
1 "authCode": "1a20a95b-d5dc-42b0-9e71-0e966a37811d",
2 ...
3 ...
1 "status": true,
2 "data":
3 ...
4 ...
5 ...

Body Cookies Headers (11) Test Results
Pretty Raw Preview Visualize JSON
Status: 200 OK Time: 56 ms Size: 628 B Save Response

```

2. Decrypting the Text

1. Enter the below details under “AES Online Decryption” section

- Get the Text from POST MAN and put under “Enter the Text to Decrypt” field
- Select “Hex” radio button under “Input Text format”
- Select “CBC” from “Select Cipher Mode of Decryption” Dropdown
- Enter the IV Key under “Enter IV User During Encryption(Optional) field
- Enter the Secret key under “Enter secret Key user for Encryption) field

AES Online Encryption

Enter text to be Encrypted

```
{
  "batchId": "1281553592233",
  "financialYear": "2022-2023",
  "applications": [
    {
      ...
    }
  ]
}
```

Select Cipher Mode of Encryption

CBC

Key Size in Bits

256

Enter IV (Optional)

ILieMwfBFZPVSBcz

Enter Secret Key

A3CfAllVmjqjHBgalc1Y3cqm3pbBJrEsB

Output Text Format: Base64 Hex

Encrypt

AES Online Decryption

Enter text to be Decrypted

```
514cf7bb333eea9a6d696369d3d03c514456bb
8fe0e6eced1bd287b18699dadfcbe69cd34260
5c5bd90496c148acf8f07cbb842bf058660f721
21e79a14e15a0
```

Input Text Format: Base64 Hex

Select Cipher Mode of Decryption

CBC

Enter IV Used During Encryption(Optional)

ILieMwfBFZPVSBcz

Key Size in Bits

256

Enter Secret Key used for Encryption

A3CfAllVmjqjHBgalc1Y3cqm3pbBJrEsB

Decrypt

f) Click on Decrypt button and verify the Text is Decrypted successfully

A3CfAllVmjqHBgalc1Y3cqm3pbBJrEsB

Output Text Format: Base64 Hex

Encrypt

AES Encrypted Output:

```
17864DD2186A24629D2733454AFCAE035630
7A886D3EA981C0C2C5A7C5A31C3867C14DA
85DC8I570148C198D86329695557C3D09852
39C6AE6CA63FF9A6DD7DE7F795A6C860002
8E81F2ED9ACEE8C68CA907F9448EBA50D62A
```

Enter Secret Key used for Encryption

Decrypt

AES Decrypted Output (Base64):

Decode to Plain Text

Plain text will appear here

ILieMwfBFZPVSBCz

Enter Secret Key

Output Text Format: Base64 Hex

Encrypt

AES Encrypted Output:

```
17864DD2186A24629D2733454AFCAE035630
7A886D3EA981C0C2C5A7C5A31C3867C14DA
85DC8I570148C198D86329695557C3D09852
39C6AE6CA63FF9A6DD7DE7F795A6C860002
8E81F2ED9ACEE8C68CA907F9448EBA50D62A
```

Key Size in Bits

256

Enter Secret Key used for Encryption

Decrypt

AES Decrypted Output (Base64):

eyJzdGF0dXMiOjEslmJhdGNoQWNrSWQiOiJNDExMjlwMDAwMDE3OSIsImVycm9yIjI6Ij9

Decode to Plain Text

Plain text will appear here

g) Click no “Decode to Plain Text” button and verify the Text is Decoded into plain text successfully

Output Text Format: Base64 Hex

Encrypt

AES Encrypted Output:

```
17864DD2186A24629D2733454AFCAE035630  
7A886D3EA981C0C2C5A7C5A31C3867C14DA  
85DC81570148C198D86329695557C3D09852  
39C6AE6CA63FF9A6DD7DE7F795A6C860002  
8E81F2ED9ACEE8C68CA907F9448EBA50D62A
```

Decrypt

AES Decrypted Output (Base64):

```
eyJzdGF0dXMiOjEslmJhdGNoQWNrSWQiOiIxND  
ExMjlwMDAwMDE3OSIsImVycm9yI6IiJ9
```

Decode to Plain Text

```
{"status":1,"batchAckId":"14112200000179","errors":[]}
```

ANNEXURE II (DB SCHEMA)**6.4. For KCC Loan-applications****1. Batch Properties**

Field Name	Type	Required?	Min Value (num) / Length (string)	Max Value (num) / Length (string)	Description
batchId	String	Yes	14	14	Unique identifier of a batch of applications
financialYear	String	Yes	9	9	Financial year of the loan applications

2. Application

Field Name	Type	Required?	Min Value/Length	Max Value/Length	Description	Value Options
uniqueId	String	Yes	19	19	Unique identifier of an application	
recordStatus	Number	Yes	1	2	Identifier for what data a loan application will contain (Refer section 5.1.3 – Submit batch array of applications)	1 2
basicDetails	Object	No			Basic details of the beneficiary	
residentialIDdetails	Object	No			Residential details of the beneficiary	
accountDetails	Object	No			Account details of the beneficiary	
loanDetails	Object	No			Loan details of the beneficiary	
activities	Array of objects	No			Activities of the beneficiary	

3. Basic Details

Field Name	Type	Min Value (num) / Length (string)	Max Value (num) / Length (string)	Description	Value Options (Codes will be shared through an API)	Required in Full-automation?	Required in Hybrid mode?
beneficiaryName	String	2	100	Name of the beneficiary	Allowed: . () space and alphabets are allowed. This name and Aadhaar will be checked from UIDAI	Yes	Yes
aadhaarNumber	String	12	12	Aadhaar number of the beneficiary		Yes	Yes
beneficiaryPassbookName	String	2	100	Name of the beneficiary as per passbook		Yes	No
mobile	String	10	10	Mobile number of beneficiary		Yes	No
dob	String	10	10	DOB of the beneficiary (format - YYYY-MM-DD)	Allowed from 18 to 125 years	Yes	No
gender	Number	1	3	Gender code of the beneficiary	MALE (1) FEMALE (2) OTHER (3)	Yes	No
socialCategory	Number	1	4	Social category code of the beneficiary	SC (1) ST (2) GEN (3) OBC (4)	Yes	No

Field Name	Type	Min Value (num) / Length (string)	Max Value (num) / Length (string)	Description	Value Options (Codes will be shared through an API)	Required in Full-automation?	Required in Hybrid mode?
farmerCategory	Number	1	3	Farmer category code of the beneficiary	OWNER (1) SHARECROPPER (2) TENANT (3)	Yes	No
farmerType	Number	1	3	Farmer type code of the beneficiary	SMALL (1) OTHER (2) MARGINAL (3)	Yes	No
primaryOccupation	Number	1	3	Primary occupation code of the beneficiary	FARMER (1) FISHRIES (2) ANIMAL HUSBANDARY (3)	Yes	No
relativeType	Number	1	4	Relative type code of the beneficiary	SON OF (1) DAUGHTER OF (2) CARE OF (3) WIFE OF (4)	Yes	No
relativeName	String	2	100	Relative name of the beneficiary		Yes	No

4. Residential Details

Field Name	Type	Min Value (num) / Length (string)	Max Value (num) / Length (string)	Description	Value Options (Codes will be shared through an API)	Required in Full-automation?	Required in Hybrid mode?
residentialVillage	String	6	6	Code of the village		No	No
residentialAddress	String	10	255	Street address of the beneficiary		No	No
residentialPincode	String	6	6	PINCODE of the address of the beneficiary		Yes	

5. Account Details

Field Name	Type	Min Value (num) / Length (string)	Max Value (num) / Length (string)	Description	Value Options (Codes will be shared through an API)	Required in Full-automation?	Required in Hybrid mode?
accountNumber	String	5	20	Account number of the beneficiary		Yes	Yes
branchCode	String	6	6	Branch code as per KCC-ISS portal		Yes	Yes

ifsc	String	11	11	IFSC code of the branch		Yes	Yes
accountHolder	Number	1	2	Account holder type of the beneficiary	SINGL E (1) JOINT (2)	Yes	No
jointAccountHolders	Array of objects			This contains joint account holders' details		No	No

6. Joint Account Holders

Only applicable when Account-Details tab accountHolder field has value “2”

Field Name	Type	Min Value (num) / Length (string)	Max Value (num) / Length (string)	Description	Value Options (Codes will be shared through an API)	Required in Full-automation ?	Required in Hybrid mode?
name	String	2	100	Name of the joint account holder		Yes	Yes
aadhaarNumber	String	12	12	Joint account holder Aadhaar number		Yes	Yes

7. Loan Details

Field Name	Type	Min Value (num) / Length (string)	Max Value (num) / Length (string)	Description	Value Options (Codes will be shared through an API)	Required in Full-automation ?	Required in Hybrid mode?

kccLoanSanctioned Date	String	10	10	Loan sanctioned or loan renewed date (format YYYY-MM-DD)		Yes	Yes
kccLoanSanctioned Amount	Number	100	999999999	Loan sanctioned / renewed amount (INR)		Yes	Yes
kccDrawingLimitfor FY	Number	100	999999999	Maximum withdrawal limit against KCC for the selected financial year		Yes	Yes

8. Activities

Field Name	Type	Min Value (num) / Length (string)	Max Value (num) / Length (string)	Description	Value Options (Codes will be shared through an API)	Required in Full-automation?	Required in Hybrid mode?
activityType	Number	1	5	Activity type code	AGRI CROPS (1) HORTI AND VEG CROPS (2) ANIMAL HUSBANDRY (3) INLAND FISHERIES (4) MARINE FISHERIES (5)	Yes	Yes

loanSanctionedDate	String	10	10	Loan sanctioned or loan renewed date (format - YYYY-MM-DD)		Yes	Yes
loanSanctionedAmount	Number	100	999999999	Loan sanctioned / renewed amount for the selected FY (INR)		Yes	Yes
activityRows	Array of objects			This contains rows of activities			

9. Activity Details

Field Name	Type	Required?	Min Value (num) / Length (string)	Max Value (num) / Length (string)	Activity Type (Applicability wrt field name)	Description	Value Options (Codes will be shared through an API)	Required in Full-automation?	Required in Hybrid mode?
landVillage	String	Yes	6	6	AGRI CROPS HORTI AND VEG CROPS ANIMAL HUSB ANDRY INLAND FISHERIES MARINE	Land village code		Yes	No

Field Name	Type	Required?	Min Value (num) / Length (string)	Max Value (num) / Length (string)	Activity Type (Applicability wrt field name)	Description	Value Options (Codes will be shared through an API)	Required in Full-automation?	Required in Hybrid mode?
					FISHERIES				
cropCode	String	Yes	9	9	AGRI CROPS	Crop code based on name		Yes	No
surveyNumber	String	Yes	1	30	AGRI CROPS HORTI AND VEG CROPS	Survey number		Yes	No
khataNumber	String	Yes	1	30	AGRI CROPS HORTI AND VEG CROPS	Khata number		Yes	No
landArea	Number	Yes	0.0001	50	AGRI CROPS	Land area in hectare		Yes	No
landType	Number	Yes	1	2	AGRI CROPS	Land type code	IRRIGATED (1) NON-IRRIGATED (2)	Yes	No
season	Number	Yes	1	3	AGRI CROPS	Season type code	KHARIF (1) RABI (2) SUMMER/ZAID/OTHERS (3)	Yes	No
plantationCode	String	Yes	9	9	HORTI AND VEG CROPS	Plantation code based on name		Yes	No

Field Name	Type	Required?	Min Value (num) / Length (string)	Max Value (num) / Length (string)	Activity Type (Applicability wrt field name)	Description	Value Options (Codes will be shared through an API)	Required in Full-automation?	Required in Hybrid mode?
plantationArea	Number	Yes	0.001	50	HORTI AND VEG CROPS	Plantation Area in hectare		Yes	No
liveStockType	Number	Yes	1	2	ANIMAL HUSBANDRY	Live stock type code	DIARY (1) POULTRY & SMALL RUMINANT (2)	Yes	No
liveStockCode	Number	Yes	1	10	ANIMAL HUSBANDRY	Live stock code based on name	Value is dependent on selected value in liveStockType (Masters shared) In case of Dairy: COW (1) BUFFALO (2) GOAT (3) JENNY (4) CAMEL (5) In case of POULTRY & SMALL RUMINANT: SHEEP (1) GOAT (2) PIG (3) POULTRY (4) BIRDS (5) RABBIT (6)	Yes	No
unitCount	Number	Yes	1	5000	ANIMAL HUSBANDRY	Number of units		Yes	No

Field Name	Type	Required?	Min Value (num) / Length (string)	Max Value (num) / Length (string)	Activity Type (Applicability wrt field name)	Description	Value Options (Codes will be shared through an API)	Required in Full-automation?	Required in Hybrid mode?
inlandType	Number	Yes	1	6	INLAND FISHERIES	Inland type code	POND (1) TANK (2) OPEN WATER BODIES (3) RACEWAY (4) HATCHERY (5) REARING UNIT (6)	Yes	No
totalUnits	Number	Yes	1	100 00	INLAND FISHERIES MARINE FISHERIES	Number of units		Yes	No
totalArea	Number	Yes	0.0 001	50	INLAND FISHERIES	Land owned in hectare		Yes	No
marineType	Number	Yes	1	3	MARINE FISHERIES	Marine type code	REGISTERED FISHING VESSEL / BOAT (1) FISHING IN ESTUARY AND SEA (2) FISH FARMING / MARICULTURE ACTIVITIES (ESTUARY AND SEA) (3)	Yes	No

6.5. For IS-PRI Claims

Batch Properties

Field Name	Type	Required?	Min Value (num) / Length (string)	Max Value (num) / Length (string)	Description
batchId	String	Yes	14	14	Unique identifier of a batch of applications
financialYear	String	Yes	9	9	Financial year of the loan applications

Applications

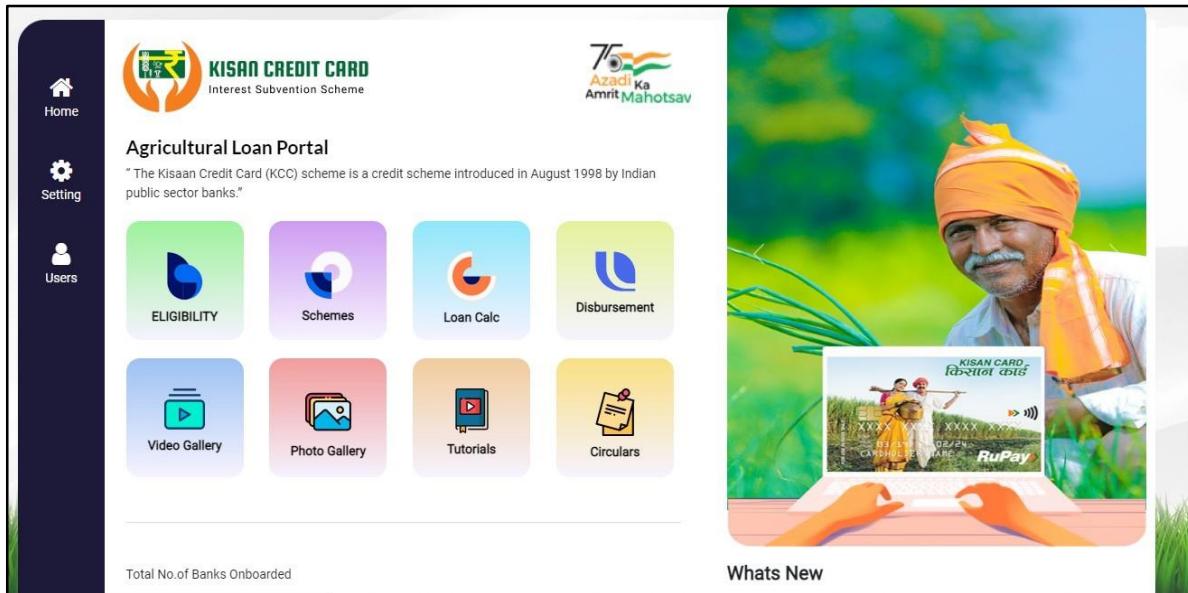
Field Name	Type	Required?	Min Value/Length	Max Value/Length	Description	Value Options
uniqueId	String	Yes	19	19	Unique identifier of an application	
loanApplicationNumber	String	Yes	19	19	Application Number	
claimType	String	Yes	Not applicable	Not applicable	IS for Interest-subsidy or PRI for Prompt Repayment Incentive, as applicable	IS or PRI
eligibleForPRI	Number	Yes (not required in case of IS claim)	0	1	Represents whether application is eligible for PRI or not.	0 - Not eligible 1 - Eligible
firstLoanDisbursementDate	String	Yes (not required in case if claim is rollovered)	10	10	Date of first loan disbursal. It should be greater than earliest loan sanctioned date of all activities. Date should be in 'YYYY-MM-DD' format.	
loanRepaymentDate	String	Yes	10	10	Date of loan Repayment. It should be less than 1 year from firstLoanDisbursementDate. Date should be in 'YYYY-MM-DD' format.	
maxWithd	Num	Yes	0	<= total loan	Should be <= total loan sanctioned amount of all activities combined	

Field Name	Type	Required?	Min Value/Length	Max Value/Length	Description	Value Options
rawal Amou nt	mb er			sanctione d (all activities)		
applicableISAm ount	N u m b er	Yes (not required in case of PRI application)	0	<= A	Applicable Interest subvention amount Formula for calculation: $A = (LCD - FDD) \times 2\% \div 364 \times (\text{minimum of either ELIS OR MWA})$	
applicablePRIAm ount	N u m b er	Yes	0	<= B	Applicable prompt repayment incentive $B = (LCD - FDD) \times 3\% \div 364 \times (\text{minimum of either ELPRI OR MWA})$	
isRoll over	N u m b er	No	0	1	Denotes whether the claim application is a rollover of any previous claim.	
submi ssion Type	N u m b er	YES (not required in case of isEligibleForPRI=0)	1	2	Denotes whether the claim is Complete or partial. 1 implies COMPLETE and 2 implies Partial	
refere nceCl aimF Y	St r i n g	NO	9	9	Denotes the financial year of the reference claim Number	
refere nceCl aimN umbe r	St r i n g	NO	17	17	Denotes the claim Number on which the claim application will be rolled-over upon	

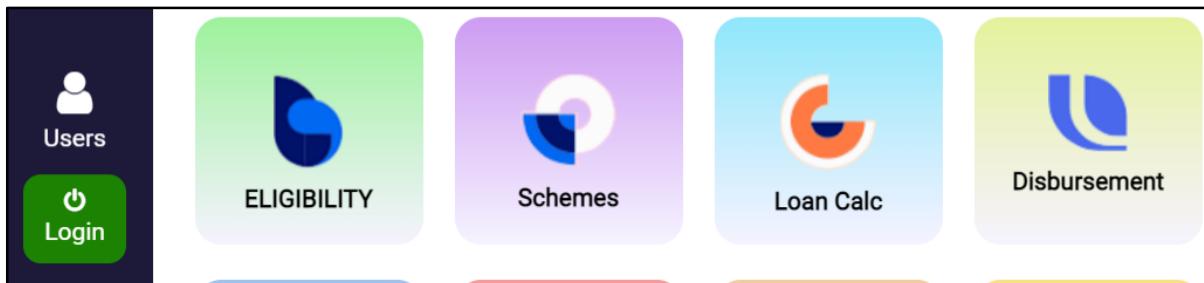
Annexure III

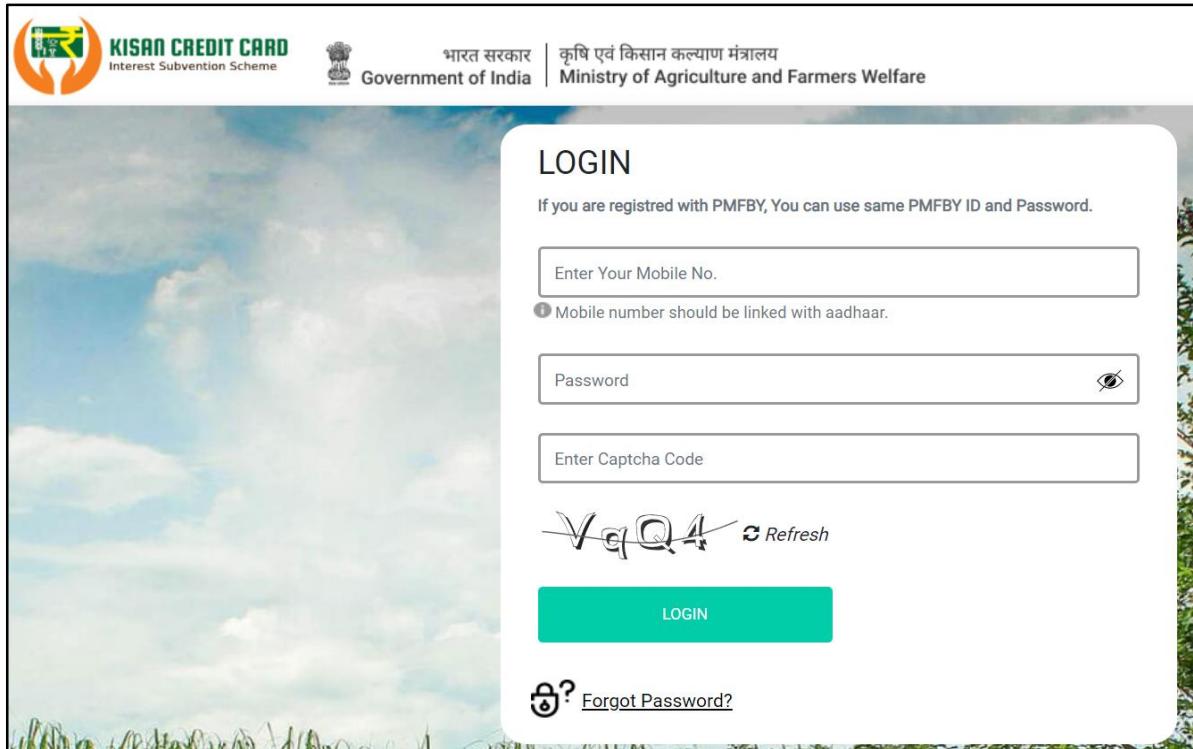
User-manual for entry of applications under the Manual-mode

- Enter the URL in the browser. User shall be navigated to home page as below:
URL: <https://fasalrin.gov.in>



- Click on “Users” menu form left menu bar and select “Login” sub-menu. User shall be navigated to “Login Page”. Enter valid mobile number, password and captcha and click on login button to login to the application with Dashboard as landing page. Incase users are registered in PMFBY portal, they can use the same login and password.





The image shows the dashboard for a 'Branch User' on the Kisan Credit Card platform. On the left is a vertical menu bar with options: 'Welcome Bank Branch User' (selected), 'Dashboard', 'Loan Application', 'User Profile', and 'Logout'. The main area displays a welcome message 'Welcome Punjab National Bank (NAGWAN Branch)' and a financial year selector set to '2022-2023'. Below this are four colored boxes showing application statistics: a purple box for 'Approved' applications (14), a pink box for 'Pending for Approval' (0), an orange box for 'Draft' applications (11), and a red box for 'Rejected' applications (0). Each box has a 'View Details' button.

Upon login the landing page has menu bar on left side along with customized dashboard. On the right-side top, the role can be seen (Branch user in this case). The dashboards are customized as per role and information specific to bank.

In order to view the Approved/ Pending for approval / Draft / Rejected applications, click on corresponding “View details” button.

- To submit loan application, Select loan applications>submit loan application menu. User shall be navigated to “**Loan account**” Page as below:

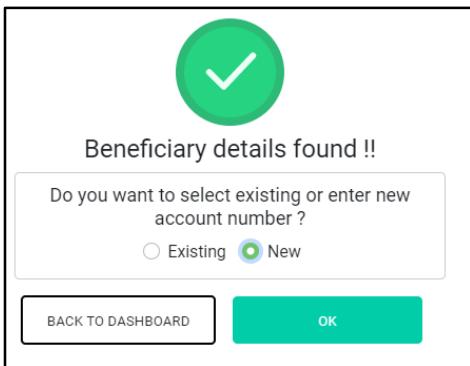
The screenshot shows the Kisan Credit Card Application Form for the Current Financial Year (FY - 2022-2023). On the left sidebar, there are links for Welcome BranchUser, Dashboard, Loan Application (selected), Submit Loan Application, User Profile, and Logout. The main form has tabs for Applicant Details, Account Details, Financial Details, Activity (Multi-Select), and Term Loan Details. The Applicant Details tab is active. A modal window titled 'Fetch record by Aadhaar number' is open. It contains fields for 'Financial Year *' (set to 2022-2023) and 'Aadhaar Number *' (partially redacted). Below these fields are 'BACK' and 'FETCH RECORD' buttons.

Select the financial year and enter **Aadhaar number** and click on “**FETCH RECORD**” button. Beneficiary details are found if the aadhar number has been used before to create partial information in any of the tabs. Hence there may be cases where existing loan account numbers are retrieved and displayed. There may be cases where only applicant details are retrieved and displayed.

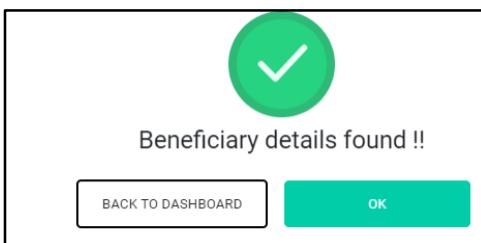
→ In case existing loan account numbers in database are fetched and showcased, user has option to choose existing loan account number from dropdown and proceed. As the user has selected existing account number, kindly note that the account details are not allowed to modify in account details tab. Otherwise this is a contradiction to the option selection.

The screenshot shows a confirmation dialog box with a green checkmark icon. The text inside says "Beneficiary details found !!". Below it is a question: "Do you want to select existing or enter new account number ?" with radio buttons for "Existing" (selected) and "New". A dropdown menu labeled "Account numbers *" shows a list of options, with "91234567802345" highlighted in blue.

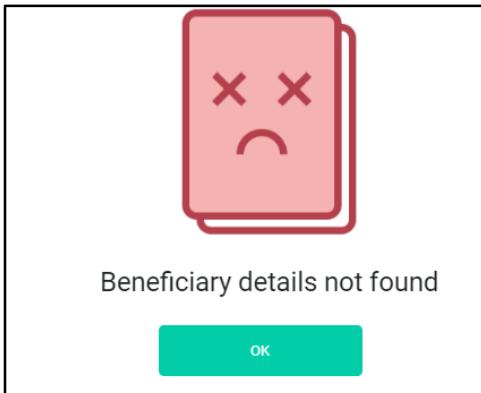
→ In case beneficiary details are found but user wants to select new account number, user can do so. The applicant details are captured and user can save and continue for updating remaining tabs.



→ In case beneficiary details are found but **not** loan account number the beneficiary details are retrieved and showcased.



→ If valid aadhar details are entered but no beneficiary details are present, system displays as shown below. User has to enter the name as per aadhaar and proceed.



→ In case the aadhar number is incorrect, system displays message to correct aadhar number.

Fetch record by Aadhaar number

Financial Year *	2022-2023
Aadhaar Number *	<input type="text" value="XXXXXXXXXXXX"/> Please enter valid Aadhaar Number.
<input type="button" value="BACK"/> <input type="button" value="FETCH RECORD"/>	

- As relevant in above cases, enter required details under “Personal details” and “Residential Address” and click on “Save and Continue” button to **Save and Continue** to account details page. The mandatory fields are marked in *****.

Application Form

Applicant Details	Account Details	Financial Details	Activity (Multi-Select)	Term Loan Details
Personal Details <div style="display: flex; justify-content: space-between;"> <div> <p>Name (As per Aadhaar) *</p> <input type="text" value="Srini"/> </div> <div> <p>Aadhaar Number *</p> <input type="text" value="989797897987"/> ✓ </div> <div> <p>Name (As per passbook) *</p> <input type="text"/> </div> </div> <div style="display: flex; justify-content: space-between;"> <div> <p>Date Of Birth *</p> <input type="date"/> </div> <div> <p>Age *</p> <input type="text" value="0"/> </div> <div> <p>Gender *</p> <input type="text" value="Select"/> </div> <div> <p>Mobile Number *</p> <input type="text"/> </div> </div> <div style="display: flex; justify-content: space-between;"> <div> <p>Social Category *</p> <input type="text" value="Select"/> </div> <div> <p>Farmer Category *</p> <input type="text" value="Select"/> </div> <div> <p>Farmer Type *</p> <input type="text" value="Select"/> </div> </div> <div style="display: flex; justify-content: space-between;"> <div> <p>Primary Occupation *</p> <input type="text" value="Select"/> </div> <div> <p>Relative Type *</p> <input type="text" value="Select"/> </div> <div> <p>Relative Name *</p> <input type="text"/> </div> </div>				
Residential Address <div style="display: flex; justify-content: space-between;"> <div> <p>State *</p> <input type="text" value="MAHARASHTRA"/> </div> <div> <p>District *</p> <input type="text" value="Mumbai"/> </div> <div> <p>Block / Subdistrict *</p> <input type="text" value="Select"/> </div> <div> <p>Village *</p> <input type="text" value="Select"/> </div> </div> <div style="display: flex; justify-content: space-between;"> <div> <p>Street Address</p> <input type="text" value="Enter Street Address"/> </div> <div> <p>Pincode *</p> <input type="text"/> </div> </div>				
<input type="button" value="SAVE & CONTINUE"/>				

- Enter all the valid details under ‘Account Details’ page and click on **Save and Continue** button to navigate to Financial details page

Applicant Details	Account Details	Financial Details	Activity (Multi-Select)	Term Loan Details
<div style="display: flex; align-items: center;"> <div style="flex: 1;"> <p>Functional Account Details</p> <div style="display: flex; justify-content: space-between;"> <div> <p>State *</p> <input type="text" value="MAHARASHTRA"/> </div> <div> <p>District *</p> <input type="text" value="Mumbai"/> </div> <div> <p>Bank *</p> <input type="text" value="State Bank Of India"/> </div> <div> <p>Branch Name *</p> <input type="text" value="BOMBAY-DADAR-SBIN0000353"/> </div> </div> <div style="margin-top: 10px;"> <p>Account Number *</p> <input type="text" value="ENTER ACCOUNT NO"/> </div> <div style="margin-top: 10px;"> <p>Confirm Account Number *</p> <input type="text" value="CONFIRM ACCOUNT NO"/> </div> <div style="margin-top: 10px;"> <p>Account Holder *</p> <input type="text" value="Select"/> </div> </div> <div style="margin-top: 20px; border: 1px solid #ccc; padding: 5px; width: fit-content;"> Beneficiary details saved successfully </div> </div>				
<input type="button" value="BACK"/> <input type="button" value="SAVE & CONTINUE"/>				

Note: If user selects Account holder as “Joint” then “Number of Account holders” Dropdown shall display. Then user needs to select the number of account holders and enter the details.

Application Form

Account Details

Functional Account Details															
State *	District *	Bank *	Branch Name *												
MAHARASHTRA	Mumbai	State Bank Of India	BOMBAY-DADAR-SBIN0000353												
Account Number *	Confirm Account Number *	Account Holder *	No. of Account Holder *												
.....	ABAB11	JOINT	2												
<table border="1"> <thead> <tr> <th>S.No</th> <th>Name</th> <th>Aadhaar No.</th> <th>Aadhaar Verification</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Srinivas</td> <td>789997897897</td> <td>VERIFIED</td> </tr> <tr> <td>2</td> <td>Enter Secondary Account Name</td> <td>Enter Secondary Account Aadhaar No</td> <td>VERIFY AADHAAR</td> </tr> </tbody> </table>				S.No	Name	Aadhaar No.	Aadhaar Verification	1	Srinivas	789997897897	VERIFIED	2	Enter Secondary Account Name	Enter Secondary Account Aadhaar No	VERIFY AADHAAR
S.No	Name	Aadhaar No.	Aadhaar Verification												
1	Srinivas	789997897897	VERIFIED												
2	Enter Secondary Account Name	Enter Secondary Account Aadhaar No	VERIFY AADHAAR												
BACK		SAVE & CONTINUE													

→ Enter all the valid details under “**Financial Details**” Page and click on “**Save and Continue**” button to save the Financial details and navigate to Activity details page.

भारत सरकार | कृषि एवं किसान कल्याण मंत्रालय
Government of India | Ministry of Agriculture and Farmers Welfare

Skip to Main Content | Account details saved successfully

Application Form

Financial Details

KCC loan sanctioned / KCC renewed on *	KCC limit sanctioned (INR) *	KCC drawing limit for current FY (INR) *
<input type="text"/>	<input type="text"/> Enter Amount (INR)	<input type="text"/> Enter Amount (INR)
KCC limit tenure (months) *	Next KCC renewal date *	
60	<input type="text"/>	
BACK	SAVE & CONTINUE	

भारत सरकार | कृषि एवं किसान कल्याण मंत्रालय
Government of India | Ministry of Agriculture and Farmers Welfare

User Category : Branch User
Skip to Main Content | Change Language : English ▾

Application Form

Financial Details

KCC loan sanctioned / KCC renewed on *	KCC limit sanctioned (INR) *	KCC drawing limit for current FY (INR) *
01/04/2022	1000	1000
KCC limit tenure (months) *	Next KCC renewal date *	
60	<input type="text"/> 31/03/2027	
BACK	SAVE & CONTINUE	

After “**Save and Continue**” user shall be navigated to Activity (multi-select) details page.

Application Form

Applicant Details Account Details Financial Details **Activity (Multi-Select)** Term Loan Details

Activity Details

Choose Activity Type

AGRI CROPS **HORTI & VEG CROPS** **ANIMAL HUSBANDRY** **FISHERIES**

BACK

- Select the required Sub activity tab (single or multiple out of 4 activity types) and enter all the required details under the Sub activity details and click on **Save and Continue** button. Here, user can select one or multiple activity types and enter corresponding data. The loan sanctioned for the financial year can be entered against each activity type. A default sanction date shall appear. The user has flexibility to edit the date.

Application Form (Current FY - 2022-2023)

Applicant Details Account Details Financial Details **Activity (Multi-Select)** Term Loan Details

Activity Details

Choose Activity Type

Agri Crops **Horti & Veg Crops** **Animal Husbandry** **Fisheries**

As per financial details tab.
KCC drawing limit for current FY
₹ 50,000
₹ 50,000 Bal.

Agri Crops



Loan Sanctioned (INR) *

Enter Loan Sanctioned amount (INR)

Loan Rollover Date *

12/04/2022



Land Location Hierarchy *
(State, District, Subdistrict, Village)

Crop *

Survey Number *

Khata Number *

Land Area *

Land Type *

Season *

(State, District, Subdistrict, V)

Select



Survey No.

Khata Number

Land Area

Select

Select



[Find Location?](#)

ADD

Click on Find location to enter the location details. Enter the land location details and click on Proceed button, then user shall be navigated to sub activity details page.

Choose Your Land Location

State * District * Block / Subdistrict * Village *

MAHARASHTRA Mumbai Greater Mumbai (M Cc) Greater Mumbai (M Cc)

PROCEED

Application Form (Current FY - 2022-2023)

Applicant Details Account Details Financial Details **Activity (Multi-Select)** Term Loan Details

Activity Details

Choose Activity Type

Agri Crops Horti & Veg Crops Animal Husbandry Fisheries

As per financial details tab.
KCC drawing limit for current FY
₹ 50,000
₹ 49,200 Bal.

Agri Crops

Loan Sanctioned (INR) *	800	Loan Rollover Date *	01/04/2022			
Land Location Hierarchy * (State, District, Subdistrict, Village)	Crop *	Survey Number *	Khata Number *	Land Area * (hectare)	Land Type *	Season *
PUNJAB, Amritsar, Ajnali	African Sarson	10	10	1	NON-IRRIGATI	KHARIF

[Find Location?](#)

ADD

Click on **Add** button to enter multiple crop details.

Activity Details

Choose Activity Type

Agri Crops **Horti & Veg Crops** **Animal Husbandry** **Fisheries**

As per financial details tab.
KCC drawing limit for current FY
₹ 50,000
₹ 49,200 Bal.

Agri Crops

Loan Sanctioned (INR) *	800	Loan Rollover Date *	01/04/2022
Land Location Hierarchy *	(State, District, Subdistrict, Village)	Crop *	African Sarson
		Survey Number *	10
		Khata Number *	10
		Land Area *	1
		(hectare)	NON-IRRIGA
		Land Type *	KHARIF
		Season *	

[Find Location?](#)

PUNJAB, Amritsar, Ajnali	Castor (Rehr, Rendi, Arandi) - RF	20	102	0.5	IRRIGATED	RABI	
--------------------------	-----------------------------------	----	-----	-----	-----------	------	--

[Find Location?](#)

ADD

BACK **ACTIVITY SUMMARY** **SAVE & CONTINUE**

→ Click on “Activity summary” button to check the activity details, mainly financial values.

S.No.	Activity Type	Loan Sanctioned (INR)
1	Agri Crops	₹ 800
	Total	₹ 800

[Find Location?](#)

In similar manner, select other sub activities as needed and follow the same steps as above to add Horti & veg crops / Animal husbandry / Inland fisheries / Marine fisheries activity details.

→ Click on “Save and Continue” button. The screen automatically navigates to “Term Loan Details” tab which is only for display purpose. The user can view the details of loans sanctioned against the 4 activity types and balance amount wrt KCC drawing limit for current FY that is applicable for term loan.

Application Form

Applicant Details	Account Details	Financial Details	Activity (Multi-Select)	Term Loan Details								
S.No.	Description			Loan Sanctioned (INR)								
1	Agri Crops			₹ 800								
Loans sanctioned for current FY				₹ 800								
<table border="1"> <thead> <tr> <th>Description</th> <th>Amount (INR)</th> </tr> </thead> <tbody> <tr> <td>KCC drawing limit for current FY</td> <td>₹ 1,000</td> </tr> <tr> <td>Loans sanctioned for current FY</td> <td>₹ 800</td> </tr> <tr> <td>Term Loan For Current FY (KCC Drawing Limit For Current FY - Loans Sanctioned For Current FY)</td> <td>₹ 200</td> </tr> </tbody> </table>					Description	Amount (INR)	KCC drawing limit for current FY	₹ 1,000	Loans sanctioned for current FY	₹ 800	Term Loan For Current FY (KCC Drawing Limit For Current FY - Loans Sanctioned For Current FY)	₹ 200
Description	Amount (INR)											
KCC drawing limit for current FY	₹ 1,000											
Loans sanctioned for current FY	₹ 800											
Term Loan For Current FY (KCC Drawing Limit For Current FY - Loans Sanctioned For Current FY)	₹ 200											
PREVIEW												

Note: The above images are shown for representative purpose to ensure all necessary features are showcased. Users are requested to perform a transaction and ensure continuous flow of information. In the above screenshot, INR 200 is for term loan.

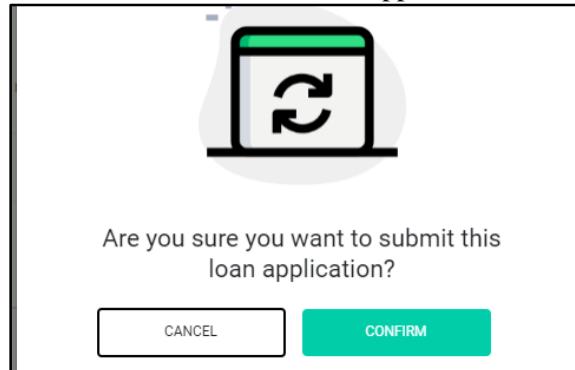
View Application Form

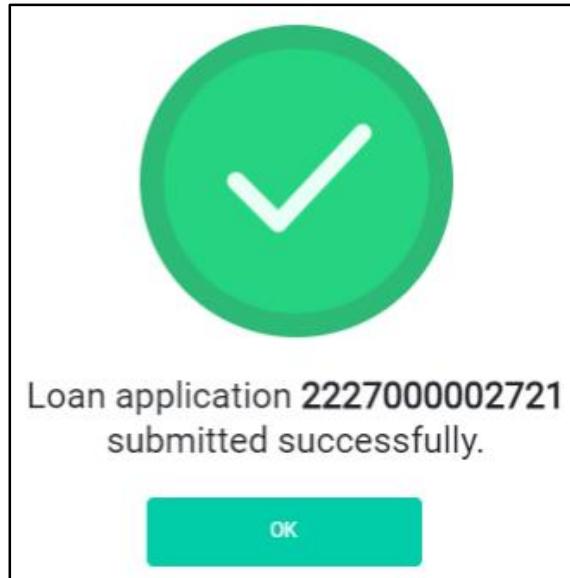
Personal Details		
Name Srinivas	Aadhar No. 989898098908	Farmer ID 22000003612
D.O.B 01/12/1982	Gender MALE	Mobile 9978978978
Farmer Category OWNER	Farmer Type SMALL	Primary Occupation FARMER
Relative Type SON OF	Relative Name Sri	
Residential Address		
State MAHARASHTRA	District Mumbai	Block / Subdistrict Greater Mumbai (M Co
Village Greater Mumbai (M Corp.)	Street Address	Pincode 400709

KCC ISS Portal (<https://fasalrin.gov.in/>)

Functional Account Details		
Bank Name State Bank Of India	Bank Branch BOMBAY-DADAR	Account Number 109109109
Account Holder SINGLE		
Financial Details		
KCC loan sanctioned / KCC renewed on 01/04/2022	KCC limit sanctioned (INR) ₹ 1,000	KCC drawing limit for current FY (INR) ₹ 1,000
KCC limit tenure (months) 60	Next KCC renewal date 31/03/2027	
Activity Details		
Agri Crops		
Loan Sanctioned (INR) ₹ 800	Loan Sanctioned On 01/04/2022	
BACK	SUBMIT	

→ Click on “Submit” button to submit the application. Click on **Confirm** button from the popup window. On successful submission the Application number is generated.





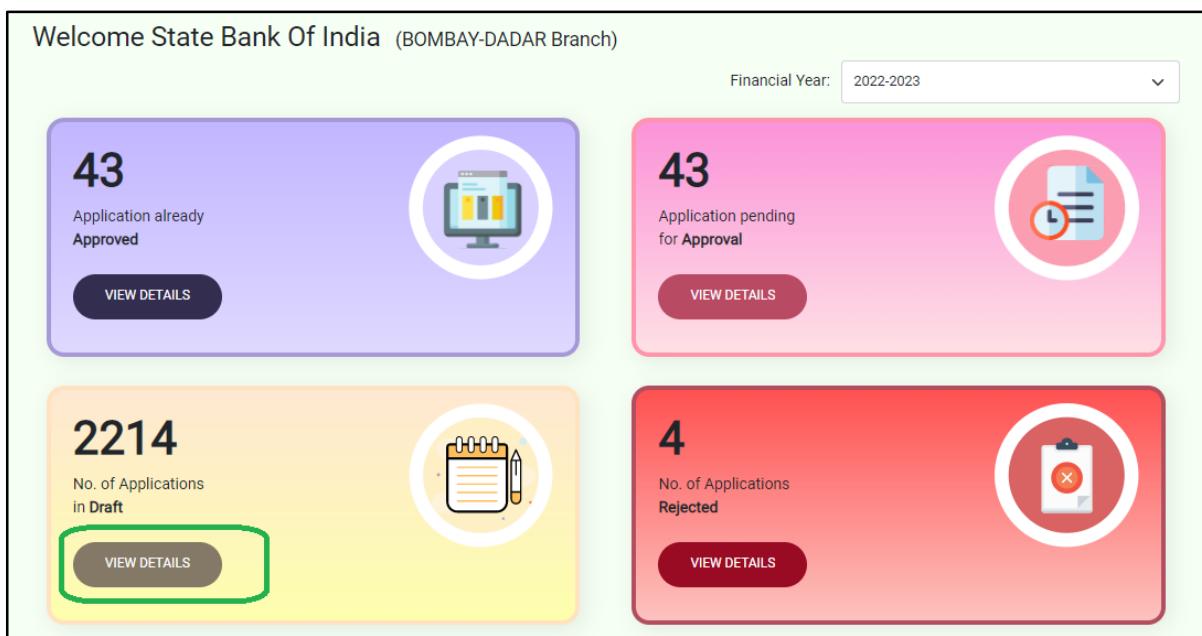
- In order to view the Submitted application navigate to “**Dashboard**” and Select “**View details**” under “**Applications pending for Approval**”

Welcome State Bank Of India (BOMBAY-DADAR Branch)

Financial Year: 2022-2023

Category	Count	Status
Approved	43	Application already Approved
Pending Approval	43	Application pending for Approval
Draft	2214	No. of Applications in Draft
Rejected	4	No. of Applications Rejected

- In order to edit and update the draft application, click on view details under “No of Applications in Draft” from dashboard.



- The application in draft will have 3 options
- Delete.
 - Preview if user wants to preview the draft application
 - Edit and update the details in case of any modifications. Click on any of the tabs to update application details

Loan Application List

Financial Year		Application Status					
Total Count: 2214	2022-2023	DRAFT	Search Here				
S.No.	Name	Mobile	Application No.	Farmer ID	Account No	Status	Action
1	Srinivas Status Four	9682390005	2227000002720	22000003558	ABCDEFGHIZZ1	Draft	EDIT DELETE PREVIEW
2	Dollar Test One	9987897878	2227000002719	22000003559	HJKLMON1	Draft	EDIT DELETE PREVIEW
3	Srinivas Status Four	9682390005	2227000002718	22000003558	ABCDEFGHIZZ1	Draft	EDIT DELETE PREVIEW
4	Srinivas Status Three	9682390004	2227000002708	22000003557	ABC4725HI1	Draft	EDIT DELETE PREVIEW
5	Srinivas Status Three	9682390004	2227000002707	22000003556	ABC4E2GHI1	Draft	EDIT DELETE PREVIEW
6	Srinivas Status Three	9682390004	2227000002706	22000003555	ABCDEFGHI1	Draft	EDIT DELETE PREVIEW

The screenshot shows the 'Application Form' section of the KCC ISS Portal. At the top, there is a header with the Government of India logo, the Ministry of Agriculture and Farmers Welfare, and links for User Category (Branch User), Skip to Main Content, and Change Language (English). Below the header, the title 'Application Form' is displayed. A navigation bar at the top of the form includes tabs for 'Applicant Details' (which is selected and highlighted in blue), 'Account Details', 'Financial Details', 'Activity (Multi-Select)', and 'Term Loan Details'. The 'Applicant Details' tab is expanded, showing a 'Personal Details' section with fields for Name (As per Aadhaar) containing 'Srinivas Status Four', Aadhaar Number (563229631199), Name (As per passbook) containing 'Srinivas Status Four', Date of Birth (02/08/1990), Age (32), Gender (FEMALE), Mobile Number (9682390005), Social Category (SC), Farmer Category (OWNER), Farmer Type (SMALL), Primary Occupation (FARMER), Relative Type (SON OF), and Relative Name (Test relative). There is also a 'Residential Address' section which is currently collapsed.

In order to update the Application details, click on **Applicant details** tabs and update.

In order to update the account details (ONLY IF **confirm account number** is not done previously), click on Account details tab and update the details. Once confirm account number is entered, the account number is not editable. This has been done for security reasons.

In order to update the financial details, click on Financial tab and update the details. In order to update the Activity details, click on Activity tabs and update the details

The screenshot shows the 'Application Form' section of the KCC ISS Portal, specifically the 'Account Details' tab. At the top, there is a header with the Government of India logo, the Ministry of Agriculture and Farmers Welfare, and links for User Category (Branch User), Skip to Main Content, and Change Language (English). Below the header, the title 'Application Form' is displayed. A navigation bar at the top of the form includes tabs for 'Applicant Details' (selected and highlighted in blue), 'Account Details' (selected and highlighted in blue), 'Financial Details', 'Activity (Multi-Select)', and 'Term Loan Details'. The 'Account Details' tab is expanded, showing a 'Functional Account Details' section with fields for State (MAHARASHTRA), District (Mumbai), Bank (State Bank Of India), Branch Name (BOMBAY-DADAR-SBIN000353), Account Number (XXXXXX...), Confirm Account Number (ABCDEFIGHIZZ1), Account Holder (JOINT), and No. Of Account Holder (2). Below this, there is a table titled 'Aadhaar Verification' showing two entries:

S.No	Name	Aadhaar No.	Aadhaar Verification
1	Srinivas Status Four	563229631199	VERIFIED
2	Test name joint account	034523613403	VERIFIED

At the bottom of the form, there are 'BACK' and 'UPDATE & CONTINUE' buttons.



भारत सरकार | कृषि एवं किसान कल्याण मंत्रालय
Ministry of Agriculture and Farmers Welfare

User Category : Branch User
[Skip to Main Content](#) | Change Language : English ▾

Application Form

Applicant Details
Account Details
Financial Details
Activity (Multi-Select)
Term Loan Details

Financial Details

KCC loan sanctioned / KCC renewed on *	KCC limit sanctioned (INR) *	KCC drawing limit for current FY (INR) *
17/06/2022	1000	1000
KCC limit tenure (months) *	Next KCC renewal date *	
60	16/06/2027	

BACK
UPDATE & CONTINUE

Choose Activity Type

As per financial details tab.

KCC drawing limit for current FY	
₹ 50,000	
₹ 49,200 Bal.	

Agri Crops
Horti & Veg Crops
Animal Husbandry
Fisheries

Agri Crops

Loan Sanctioned (INR) *	Loan Rollover Date *					
800	01/04/2022					
Land Location Hierarchy * <small>(State, District, Subdistrict, Village)</small>						
PUNJAB, Amritsar, Ajnali	Crop * <small>African Sarson</small>	Survey Number * <small>10</small>	Khata Number * <small>10</small>	Land Area * <small>(hectare)</small>	Land Type * <small>NON-IRRIGA</small>	Season * <small>KHARIF</small>
PUNJAB, Amritsar, Ajnali Castor (Rehri, Rendi, Arandi) - RF 20 102 0.5 IRRIGATED RABI ✖						

[Find Location?](#)

[Find Location?](#)

ADD

Horti & Veg Crops

Loan Sanctioned (INR) *	Loan Rollover Date *
-------------------------	----------------------

→ To submit the updated details update the necessary details and click on “**Update and Continue**” button

ADD
(1)

Fisheries (Marine Fisheries & Aquaculture)

Loan Sanctioned (INR) *	Loan Sanctioned On *	
<input type="text" value="1000.6"/>	<input type="text" value="17/06/2022"/> (1)	
Land Location Hierarchy * (State, District, Subdistrict, Village)	Marine fisheries and allied activities *	Total Nos *
ODISHA,Gajapati,Mohana,Gutuma	FISH FARMING/MARICULTURE ACTIVITIES (ESTUARY & SEA)	<input type="text" value="2"/>

[Find Location?](#)

ADD
UPDATE & CONTINUE

BACK
ACTIVITY SUMMARY

Functional Account Details

Bank Name State Bank Of India	Bank Branch BOMBAY-DADAR	Account Number ABCDEFGHIZZ1
Account Holder JOINT	Number of Account Holder 2	

Financial Details

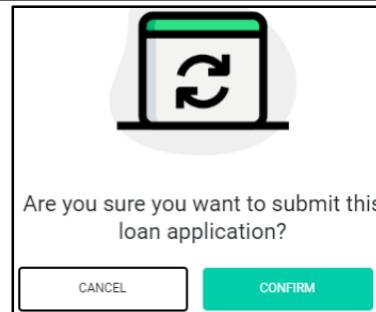
KCC loan sanctioned / KCC renewed on 17/06/2022	KCC limit sanctioned (INR) ₹ 5,521	KCC drawing limit for current FY (INR) ₹ 1,000
KCC limit tenure (months) 60	Next KCC renewal date 16/06/2027	

Activity Details

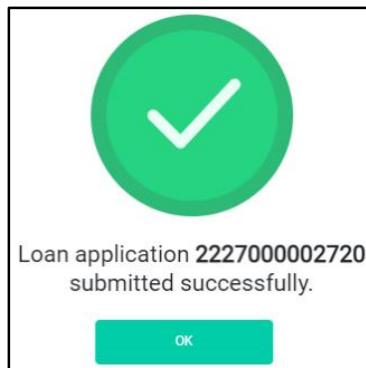
Agri & Veg Crops

Loan Sanctioned (INR) ₹ 1,000	Loan Sanctioned On 17/06/2022
---	---

BACK
SUBMIT

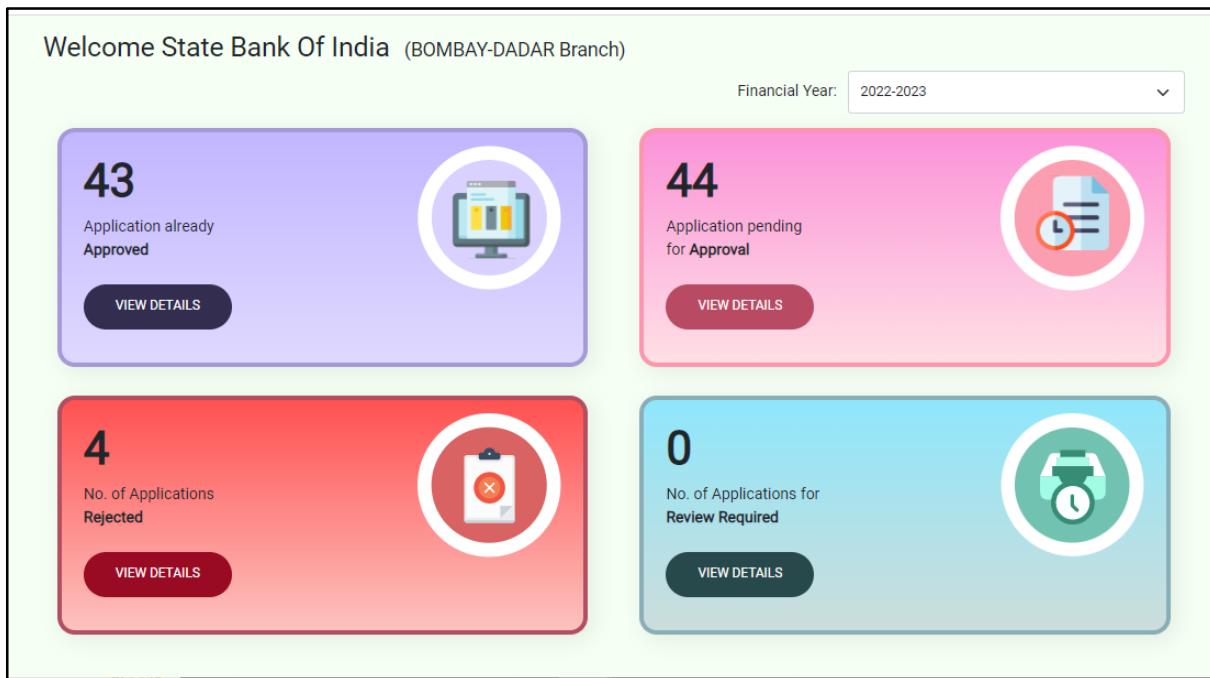


→ Click on Confirm button to submit the Updated Draft Application



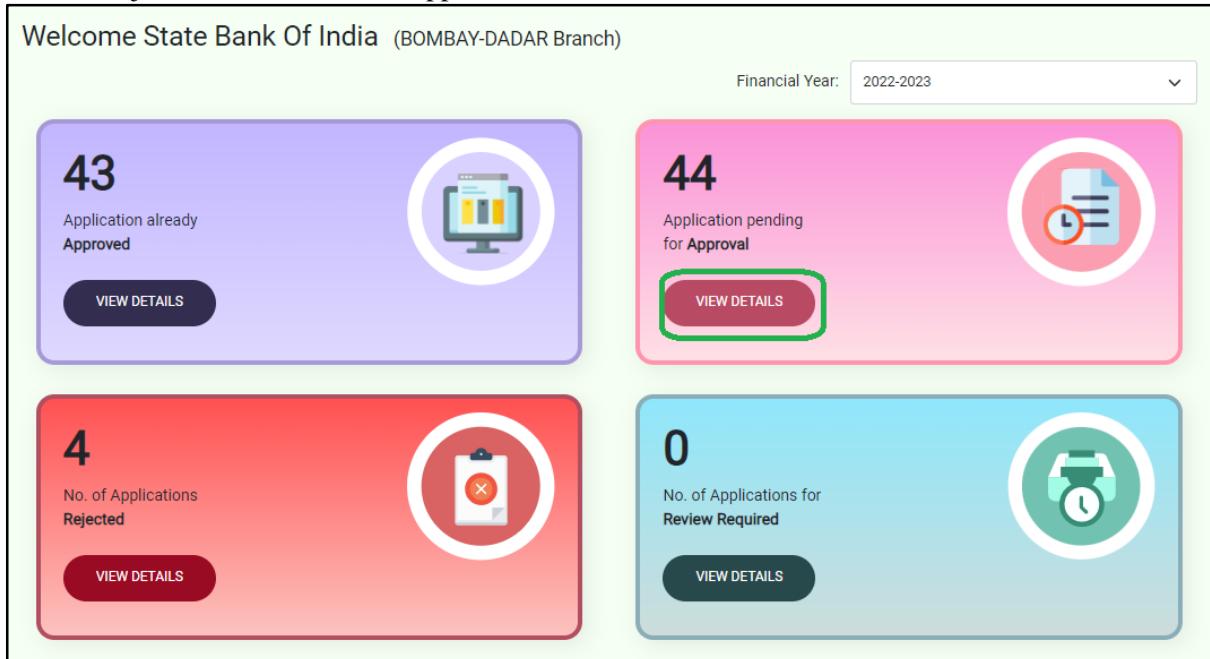
Steps to “Approve”/“Reject”/“Edit”/“Delete” the Application submitted for Approval (Branch head)

→ Login into Application with “**Branch Head**” then user shall be navigated to “Dashboard”



In order to view the Approved / Rejected / Pending for submission approval click on corresponding “View Details” button.

→ Click on “View Details” button under “Applications pending for approval” to Approve / Reject / Edit or Delete the application



User shall be navigated to “Application pending for approval” listing page

भारत सरकार
कृषि एवं किसान कल्याण मंत्रालय

Ministry of Agriculture and Farmers Welfare

User Category : Branch Head

Skip to Main Content
| Change Language : English ▾

Loan Application List

Total Count: 44
Financial Year
Application Status

2022-2023
SUBMITTED
Search Here

S.No.	Name	Mobile	Application No.	Farmer ID	Account No	Status	Action
1	Srinivas	9897897897	2227000002721	22000003562	ABAB11	Submitted	REVIEW
2	Srinivas Status Four	9682390005	2227000002720	22000003558	ABCDEFGHIZZ1	Submitted	REVIEW
3	Srinivas Status Four	9682390005	2227000002709	22000003558	ABCDEFGHIZZ1	Submitted	REVIEW
4	SriniJ	9656756756	2227000002699	22000003547	ABC1	Submitted	REVIEW
5	qwqw	9876541234	2227000002675		12312312388	Submitted	REVIEW
6	SJ	9987897897	2227000002661	22000003516	AABB	Submitted	REVIEW

Apply filters required year and application status if need be. Click on “Review” button.

भारत सरकार
कृषि एवं किसान कल्याण मंत्रालय

Ministry of Agriculture and Farmers Welfare

User Category : Branch Head

Skip to Main Content
| Change Language : English ▾

Loan Application List

Total Count: 44
Financial Year
Application Status

2022-2023
SUBMITTED
Search Here

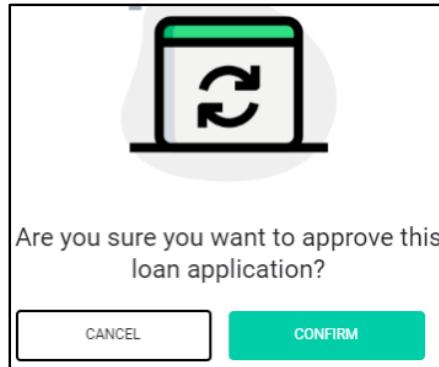
S.No.	Name	Mobile	Application No.	Farmer ID	Account No	Status	Action
1	Srinivas	9897897897	2227000002721	22000003562	ABAB11	Submitted	REVIEW
2	Srinivas Status Four	9682390005	2227000002720	22000003558	ABCDEFGHIZZ1	Submitted	REVIEW
3	Srinivas Status Four	9682390005	2227000002709	22000003558	ABCDEFGHIZZ1	Submitted	REVIEW
4	SriniJ	9656756756	2227000002699	22000003547	ABC1	Submitted	REVIEW
5	qwqw	9876541234	2227000002675		12312312388	Submitted	REVIEW
6	SJ	9987897897	2227000002661	22000003516	AABB	Submitted	REVIEW

→ User shall be navigated to application form where user can approve, edit, reject or delete the form

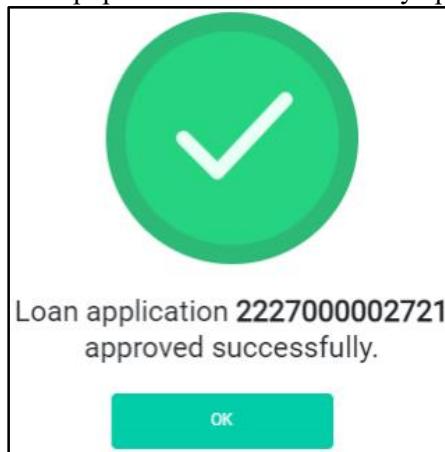
Functional Account Details				
Bank Name State Bank Of India	Bank Branch BOMBAY-DADAR	Account Number ABAB11		
Account Holder JOINT	Number of Account Holder 2			
Financial Details				
KCC loan sanctioned / KCC renewed on 01/04/2022	KCC limit sanctioned (INR) ₹ 1,000	KCC drawing limit for current FY (INR) ₹ 900		
KCC limit tenure (months) 60	Next KCC renewal date 31/03/2027			
Activity Details				
Agri & Veg Crops <table border="1"> <tr> <td>Loan Sanctioned (INR) ₹ 1,000</td> <td>Loan Sanctioned On 01/04/2022</td> </tr> </table>			Loan Sanctioned (INR) ₹ 1,000	Loan Sanctioned On 01/04/2022
Loan Sanctioned (INR) ₹ 1,000	Loan Sanctioned On 01/04/2022			
BACK	APPROVE	REJECT	EDIT	DELETE

4. Click on “**Approve**” button and Click on “**Confirm**” button from the Popup displayed to approve loan application

Functional Account Details				
Bank Name State Bank Of India	Bank Branch BOMBAY-DADAR	Account Number ABAB11		
Account Holder JOINT	Number of Account Holder 2			
Financial Details				
KCC loan sanctioned / KCC renewed on 01/04/2022	KCC limit sanctioned (INR) ₹ 1,000	KCC drawing limit for current FY (INR) ₹ 900		
KCC limit tenure (months) 60	Next KCC renewal date 31/03/2027			
Activity Details				
Agri & Veg Crops <table border="1"> <tr> <td>Loan Sanctioned (INR) ₹ 1,000</td> <td>Loan Sanctioned On 01/04/2022</td> </tr> </table>			Loan Sanctioned (INR) ₹ 1,000	Loan Sanctioned On 01/04/2022
Loan Sanctioned (INR) ₹ 1,000	Loan Sanctioned On 01/04/2022			
BACK	APPROVE	REJECT	EDIT	DELETE



Click on “Confirm” button from Popup window to successfully approve the application



→ Click on “Edit” button to edit the loan application

Functional Account Details				
Bank Name State Bank Of India	Bank Branch BOMBAY-DADAR	Account Number ABCDEFGHIZZ1		
Account Holder JOINT	Number of Account Holder 2			
Financial Details				
KCC loan sanctioned / KCC renewed on 17/06/2022	KCC limit sanctioned (INR) ₹ 5,521	KCC drawing limit for current FY (INR) ₹ 1,000		
KCC limit tenure (months) 60	Next KCC renewal date 16/06/2027			
Activity Details				
Agri & Veg Crops <table border="1"> <tr> <td>Loan Sanctioned (INR) ₹ 1,000</td> <td>Loan Sanctioned On 17/06/2022</td> </tr> </table>			Loan Sanctioned (INR) ₹ 1,000	Loan Sanctioned On 17/06/2022
Loan Sanctioned (INR) ₹ 1,000	Loan Sanctioned On 17/06/2022			
BACK	APPROVE	REJECT		
	EDIT	DELETE		

This website belongs to Department of Agriculture and Farmers Welfare, Ministry of Agriculture and Farmers Welfare, Government of India.

Update required changes, click on update and continue button.

Activity Details

Choose Activity Type

AGRI CROPS **HORTI & VEG CROPS** **ANIMAL HUSBANDRY** **FISHERIES**

Agri Crops

Loan Sanctioned (INR) *	800	Loan Sanctioned On *	01/04/2022
Land Location Hierarchy *	MAHARASHTRA,Mumba	Crop *	African Sarson
Survey Number *	10	Khata Number *	10
Land Area *	10	Land Type *	IRRIGATED
(hectare)	KHARIF	Season *	RABI

[Find Location?](#)

MAHARASHTRA,Mumba	Bajra - Napier Hybrid	20	20	20	NON-IRRIGA	RABI
-------------------	-----------------------	----	----	----	------------	------

[Find Location?](#)

ADD

BACK **ACTIVITY SUMMARY** **UPDATE & CONTINUE**

Click on “Preview” button

Application Form

Term Loan Details

S.No.	Description	Loan Sanctioned (INR)
1	Agri Crops	₹ 800
	Loans sanctioned for current FY	₹ 800

Description	Amount (INR)
KCC drawing limit for current FY	₹ 1,000
Loans sanctioned for current FY	₹ 800
Term Loan For Current FY (KCC Drawing Limit For Current FY - Loans Sanctioned For Current FY)	₹ 200

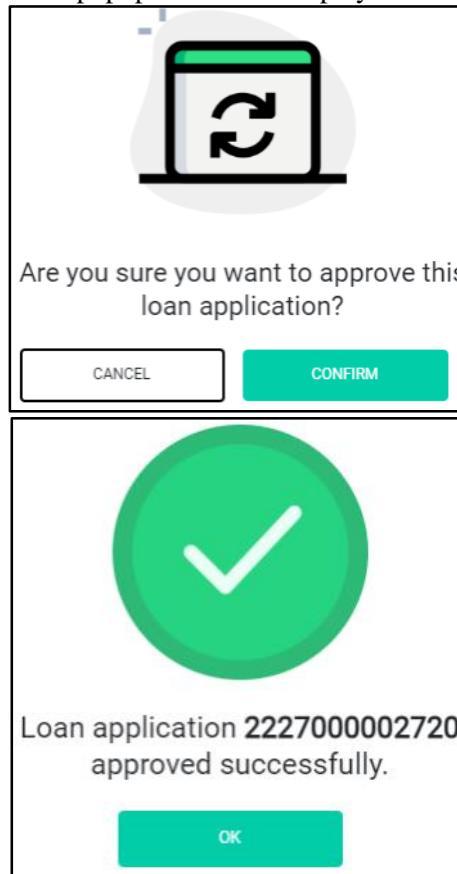
PREVIEW

Note: In case KCC drawing limit for current FY and sum of loans sanctioned against 4 activity types for the current FY matches, the term loan is not applicable. Hence, the term loan = 0. In the above case showcased, term loan is INR 200.

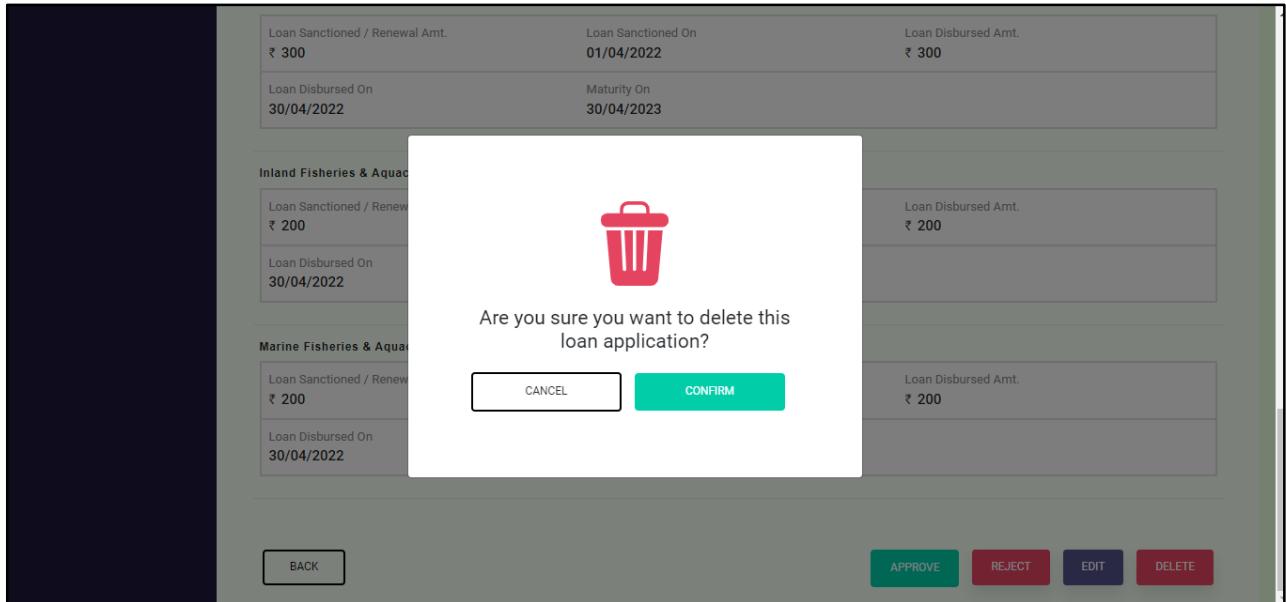
Click on “Approve” button to approve the changes and click on “Confirm” button from the popup displayed to approve loan application.

Functional Account Details				
Bank Name State Bank Of India	Bank Branch BOMBAY-DADAR	Account Number ABCDEFGHIZZ1		
Account Holder JOINT	Number of Account Holder 2			
Financial Details				
KCC loan sanctioned / KCC renewed on 17/06/2022	KCC limit sanctioned (INR) ₹5,521	KCC drawing limit for current FY (INR) ₹1,000		
KCC limit tenure (months) 60	Next KCC renewal date 16/06/2027			
Activity Details				
Agri & Veg Crops <table border="1"> <tr> <td>Loan Sanctioned (INR) ₹ 1,000</td> <td>Loan Sanctioned On 17/06/2022</td> </tr> </table>			Loan Sanctioned (INR) ₹ 1,000	Loan Sanctioned On 17/06/2022
Loan Sanctioned (INR) ₹ 1,000	Loan Sanctioned On 17/06/2022			
BACK	APPROVE			

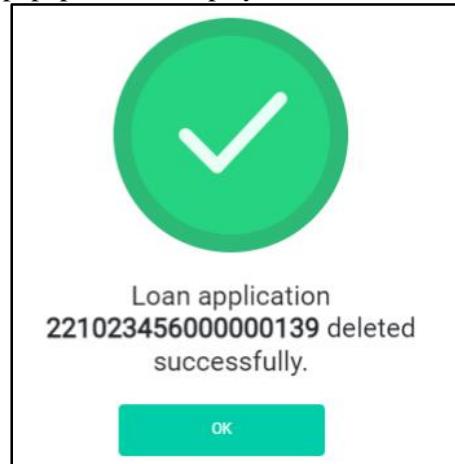
On Clicking on confirm button the popup window is displayed with confirmation.



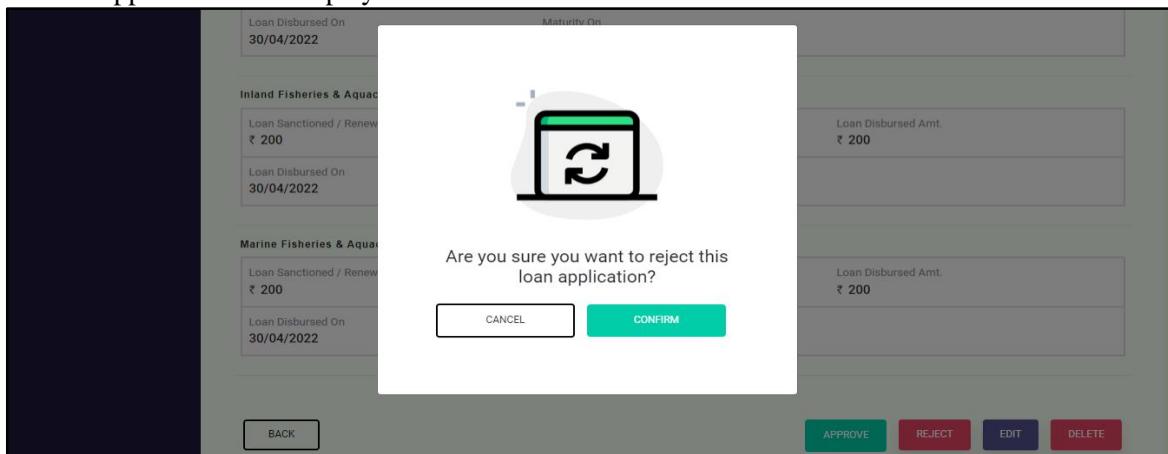
→ Click on “Delete” button to Delete the loan application



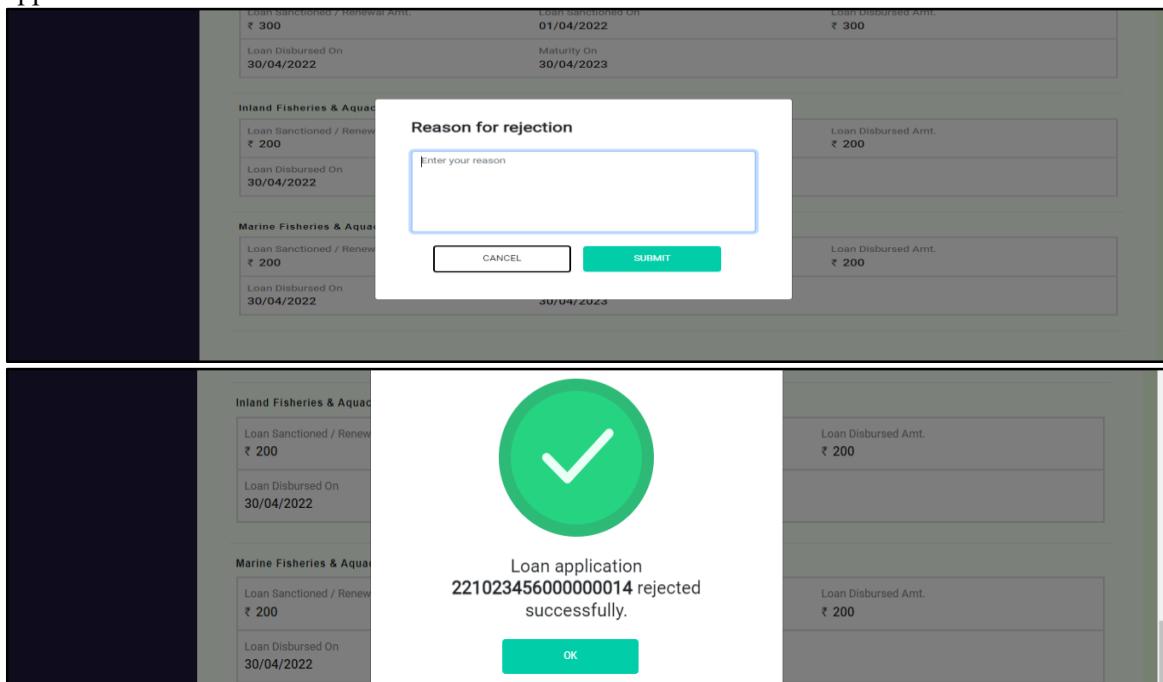
On click on Confirm button a popup shall be displayed with confirmation message for deletion



→ To reject the application, click on “Reject” button, Confirmation popup for rejecting application will display.



Click on “Confirm” button, Reason for rejection popup will display. Enter reason for rejecting the application and click on “Submit”



→ Naviage to dashboard and click on “View details” under No of Application rejected to View the Rejected application

Category	Count	Action
Approved	45	View Details
Pending Approval	42	View Details
Rejected	4	View Details
Review Required	0	View Details

Loan Application List							
Financial Year		Application Status					
Total Count: 4		2022-2023		REJECTED		Search Here	
S.No.	Name	Mobile	Application No.	Farmer ID	Account No	Status	Action
1	Sri	9897676876	2227000002648	22000003502	CD01	Rejected	PREVIEW
2	Ashish	8764838976	2227000001571	22000002437	5432109876	Rejected	PREVIEW
3	tejas reject form	9820826699	2227000001526	22000002387	702142025512121212	Rejected	PREVIEW
4	hhjj	8297009772	2227000000455	22000001310	717171717171717171	Rejected	PREVIEW

Click on “Preview” button to View the details

Loan Application List							
Financial Year		Application Status					
Total Count: 4		2022-2023		REJECTED		Search Here	
S.No.	Name	Mobile	Application No.	Farmer ID	Account No	Status	Action
1	Sri	9897676876	2227000002648	22000003502	CD01	Rejected	PREVIEW
2	Ashish	8764838976	2227000001571	22000002437	5432109876	Rejected	PREVIEW
3	tejas reject form	9820826699	2227000001526	22000002387	702142025512121212	Rejected	PREVIEW
4	hhjj	8297009772	2227000000455	22000001310	717171717171717171	Rejected	PREVIEW

View Application Form

Personal Details

Name Sri	Aadhar No. 797987897897	Farmer ID 22000003502
D.O.B 30/06/1980	Gender MALE	Mobile 9897676876
Farmer Category OWNER	Farmer Type SMALL	Primary Occupation FARMER
Relative Type SON OF	Relative Name Sri	

Residential Address

State MAHARASHTRA	District Mumbai	Block / Subdistrict Greater Mumbai (M Co
Village Greater Mumbai (M Corp.)	Street Address r-170,Navimumbai	Pincode 400080

Steps to resubmit the rejected application

→ Login with **Bank Branch User** and navigate to Dashboard

Welcome State Bank Of India (BOMBAY-DADAR Branch)

Financial Year: 2022-2023

45
Application already Approved
VIEW DETAILS

42
Application pending for Approval
VIEW DETAILS

2213
No. of Applications in Draft
VIEW DETAILS

4
No. of Applications Rejected
VIEW DETAILS

→ Click on “View Details” under “No of Application rejected”

Welcome State Bank Of India (BOMBAY-DADAR Branch)

Financial Year: 2022-2023

45
Application already Approved
VIEW DETAILS

42
Application pending for Approval
VIEW DETAILS

2213
No. of Applications in Draft
VIEW DETAILS

4
No. of Applications Rejected
VIEW DETAILS

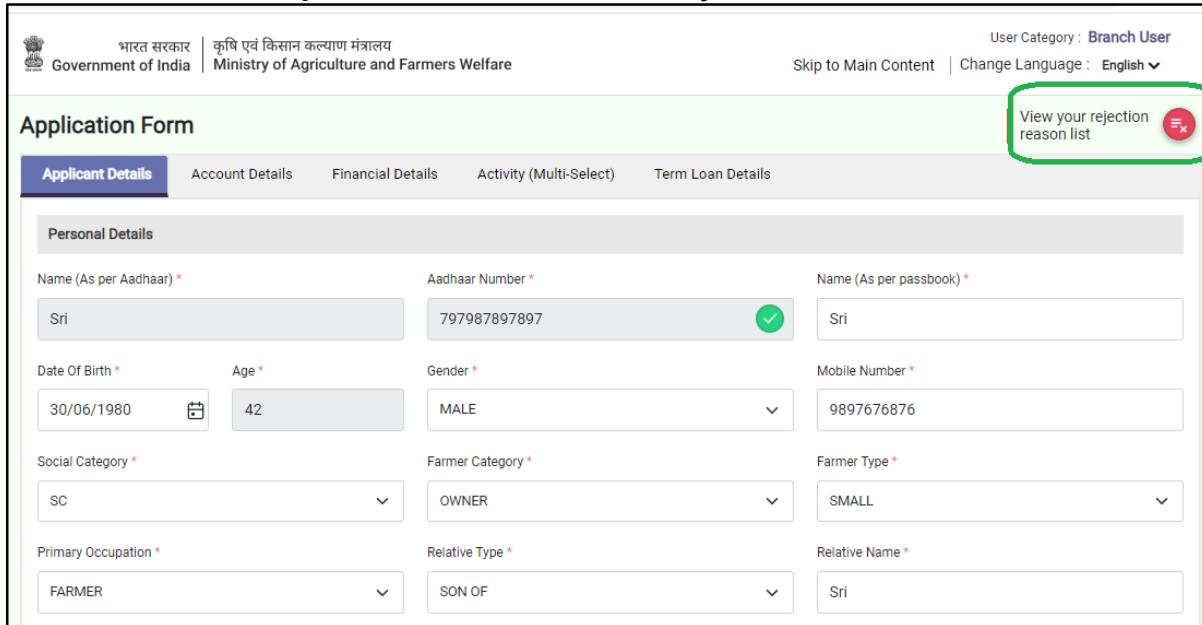
Loan Application List							
Financial Year		Application Status					
Total Count: 4		2022-2023		REJECTED		Search Here	
S.No.	Name	Mobile	Application No.	Farmer ID	Account No	Status	Action
1	Sri	9897676876	2227000002648	22000003502	CD01	Rejected	EDIT DELETE PREVIEW
2	Ashish	8764838976	2227000001571	22000002437	5432109876	Rejected	EDIT DELETE PREVIEW
3	tejas reject form	9820826699	2227000001526	22000002387	70214202551212121212	Rejected	EDIT DELETE PREVIEW
4	hhjj	8297009772	2227000000455	22000001310	71717171717171717171	Rejected	EDIT DELETE PREVIEW

In order to **Delete** the Rejected Application user shall click on “**Delete**” button and click on Confirm button.

In order to **Preview** the Rejected application user shall click on “**Preview**” button

In order to resubmit the rejected application, user shall click “**Edit**” button and use

Click on “**View Your Rejected reason list**” to view the Rejected reason



The screenshot shows the 'Application Form' page. At the top, it displays the Government of India logo, the Ministry of Agriculture and Farmers Welfare, and user category information ('User Category: Branch User'). Below this, there are tabs for 'Applicant Details', 'Account Details', 'Financial Details', 'Activity (Multi-Select)', and 'Term Loan Details'. The 'Applicant Details' tab is active. The 'Personal Details' section contains fields for Name (As per Aadhaar), Aadhaar Number, Name (As per passbook), Date of Birth, Age, Gender, Mobile Number, Social Category, Farmer Category, Farmer Type, Primary Occupation, Relative Type, and Relative Name. A red oval highlights the 'View your rejection reason list' button, which is located in the top right corner of the form area.

Rejection comments

Remark:
Rejecting
Date:- 15/12/2022 , Time:- 3:35:53 PM

CLOSE

Click on **Close** button from the popup the user shall be redirected to Application details page

भारत सरकार | कृषि एवं किसान कल्याण मंत्रालय
Government of India | Ministry of Agriculture and Farmers Welfare

User Category : Branch User
Skip to Main Content | Change Language : English ▾

Application Form

Applicant Details Account Details Financial Details Activity (Multi-Select) Term Loan Details

Personal Details

Name (As per Aadhaar) *	Aadhaar Number *	Name (As per passbook) *	
Sri	797987897897	Sri	
Date Of Birth *	Age *	Gender *	Mobile Number *
30/06/1980	42	MALE	9897676876
Social Category *	Farmer Category *	Farmer Type *	
SC	OWNER	SMALL	
Primary Occupation *	Relative Type *	Relative Name *	
FARMER	SON OF	Sri	

In order to Update the Application details click on Application tab and make the necessary changes and click on “Update and Continue” button

Application Form

User Category : Branch User | Skip to Main Content | Change Language : English

Applicant Details

Personal Details

Name (As per Aadhaar) *	Aadhaar Number *	Name (As per passbook) *
Sri	797987897897	Sri
Date Of Birth *	Age *	Gender *
30/06/1980	42	MALE
Social Category *	Farmer Category *	Farmer Type *
SC	OWNER	SMALL
Primary Occupation *	Relative Type *	Relative Name *
FARMER	SON OF	Sri

The aadhar verification shown above is only sample. In real-time, the aadhar number + name is validated.

Residential Address

State *	District *	Block / Subdistrict *	Village *
MAHARASHTRA	Mumbai	Greater Mumbai (M Corp.)	Greater Mumbai (M Corp.)
Street Address	Pincode *		
r-170,Navimumbai	400080		

UPDATE & CONTINUE

In order to update the account details, click on “Accounts tab” and update the details and click on “Update and Continue” Button

User Category : Branch User
Skip to Main Content | Change Language : English

View your rejection reason list

Application Form

Applicant Details **Account Details** Financial Details Activity (Multi-Select) Term Loan Details

Functional Account Details

State *	District *	Bank *	Branch Name *
MAHARASHTRA	Mumbai	State Bank Of India	BOMBAY-DADAR-SBIN000353

Account Number *	Confirm Account Number *	Account Holder *	No. Of Account Holder *
....	CD01	JOINT	2

S.No	Name	Aadhaar No.	Aadhaar Verification
1	Sri	797987897897	VERIFIED
2	Sri	778978978978	VERIFIED

BACK **UPDATE & CONTINUE**

In order to Update the financial details, click on “Financial details” tab and update the details and click on Update and Continue” button

User Category : Branch User
Skip to Main Content | Change Language : English

View your rejection reason list

Application Form

Applicant Details Account Details **Financial Details** Activity (Multi-Select) Term Loan Details

Financial Details

KCC loan sanctioned / KCC renewed on *	KCC limit sanctioned (INR) *	KCC drawing limit for current FY (INR) *
01/04/2022	1000.25	Enter Amount (INR)

KCC limit tenure (months) *	Next KCC renewal date *
1	30/04/2022

BACK **UPDATE & CONTINUE**

In order to update the activity details, click on Activity details tab and update the details and click on Update and Preview button.

KCC ISS Portal (<https://fasalrin.gov.in/>)

Choose Activity Type

Agri Crops **Horti & Veg Crops** **Animal Husbandry** **Fisheries**

As per financial details tab.
KCC drawing limit for current FY
₹ 50,000
₹ 49,200 Bal.

Agri Crops

Loan Sanctioned (INR) *	800	Loan Rollover Date *	01/04/2022
Land Location Hierarchy *	PUNJAB, Amritsar, Ajnali	Crop *	African Sarson
(State, District, Subdistrict, Village)		Survey Number *	10
		Khata Number *	10
		Land Area *	1
		(hectare)	
		Land Type *	NON-IRRIGA
			KHARIF

Find Location?

PUNJAB, Amritsar, Ajnali	Castor (Rehri, Rendi, Arandi) - RF	20	102	0.5	IRRIGATED	RABI	
--------------------------	------------------------------------	----	-----	-----	-----------	------	--

Find Location?

ADD

BACK **ACTIVITY SUMMARY** **SAVE & CONTINUE**

Click on preview button from Term details page

Applicant Details	Account Details	Financial Details	Activity (Multi-Select)	Term Loan Details																	
<table border="1"> <thead> <tr> <th>S.No.</th> <th>Description</th> <th>Loan Sanctioned (INR)</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Agri Crops</td> <td>₹ 800</td> </tr> <tr> <td colspan="2">Loans sanctioned for current FY</td> <td>₹ 800</td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th>Description</th> <th>Amount (INR)</th> </tr> </thead> <tbody> <tr> <td>KCC drawing limit for current FY</td> <td>₹ 1,000</td> </tr> <tr> <td>Loans sanctioned for current FY</td> <td>₹ 800</td> </tr> <tr> <td>Term Loan For Current FY (KCC Drawing Limit For Current FY - Loans Sanctioned For Current FY)</td> <td>₹ 200</td> </tr> </tbody> </table>					S.No.	Description	Loan Sanctioned (INR)	1	Agri Crops	₹ 800	Loans sanctioned for current FY		₹ 800	Description	Amount (INR)	KCC drawing limit for current FY	₹ 1,000	Loans sanctioned for current FY	₹ 800	Term Loan For Current FY (KCC Drawing Limit For Current FY - Loans Sanctioned For Current FY)	₹ 200
S.No.	Description	Loan Sanctioned (INR)																			
1	Agri Crops	₹ 800																			
Loans sanctioned for current FY		₹ 800																			
Description	Amount (INR)																				
KCC drawing limit for current FY	₹ 1,000																				
Loans sanctioned for current FY	₹ 800																				
Term Loan For Current FY (KCC Drawing Limit For Current FY - Loans Sanctioned For Current FY)	₹ 200																				

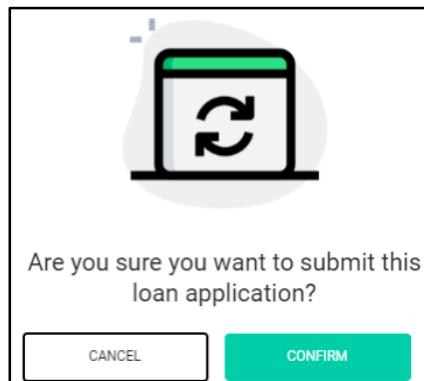
PREVIEW

(Image for representation purpose)

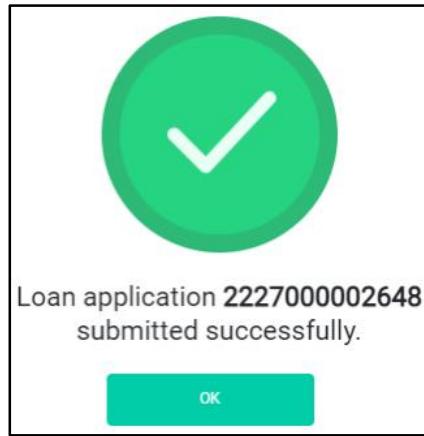
To Submit the updated application Click on “Update and Continue” button and click on Submit button

Functional Account Details				
Bank Name State Bank Of India	Bank Branch BOMBAY-DADAR	Account Number 109109109		
Account Holder SINGLE				
Financial Details				
KCC loan sanctioned / KCC renewed on 01/04/2022	KCC limit sanctioned (INR) ₹ 1,000	KCC drawing limit for current FY (INR) ₹ 1,000		
KCC limit tenure (months) 60	Next KCC renewal date 31/03/2027			
Activity Details				
Agri Crops <table border="1"> <tr> <td>Loan Sanctioned (INR) ₹ 800</td> <td>Loan Sanctioned On 01/04/2022</td> </tr> </table>			Loan Sanctioned (INR) ₹ 800	Loan Sanctioned On 01/04/2022
Loan Sanctioned (INR) ₹ 800	Loan Sanctioned On 01/04/2022			
BACK	SUBMIT			

Click on Confirm button to Resubmit the “Rejected Application” with updated details as per the Rejected comments

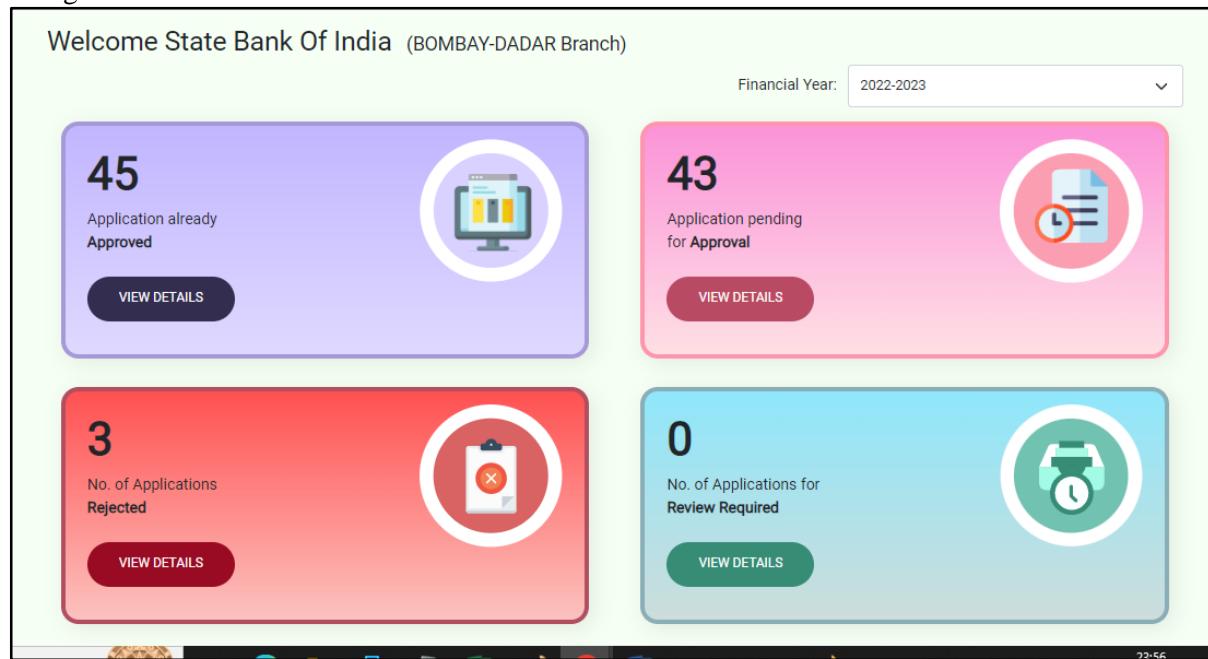


Confirmation popup window shall be displayed after successfully Resubmission of Rejected Application

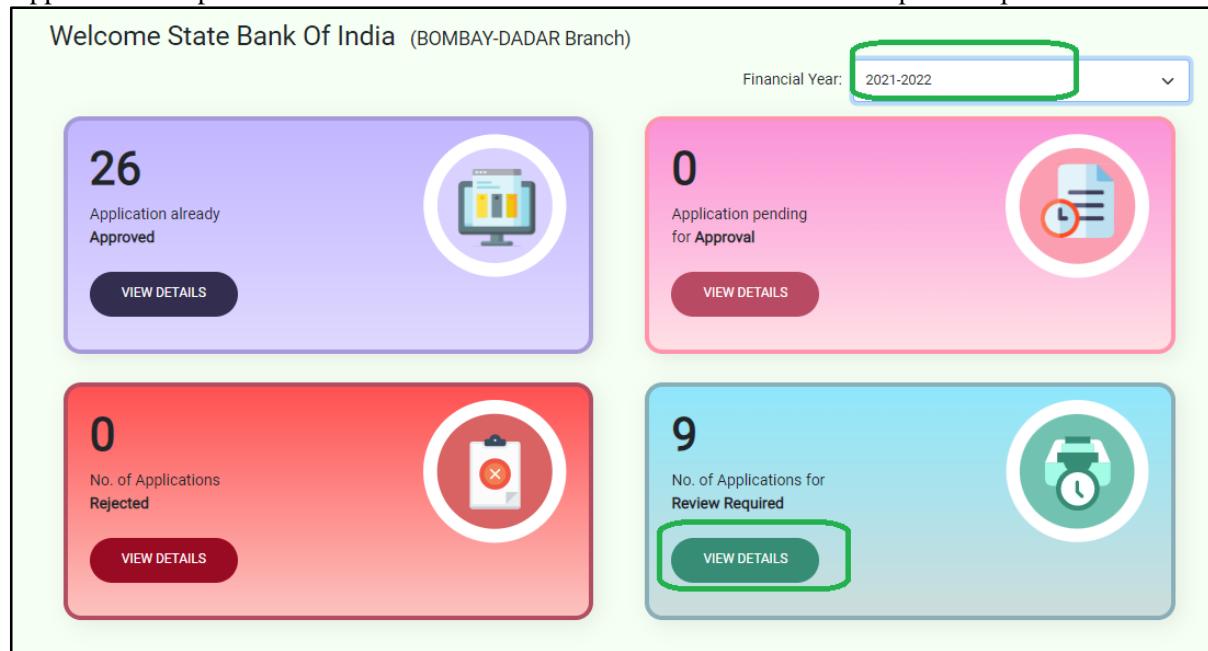


Steps to “Edit and Approve” the Application with review required status

1.Login as Branch Head User



2.Select the Financial Year from the Drpdown for which Application “Review Required” Application are present and click on View Details button from “Review Required” option



Loan Application List							
Financial Year		Application Status					
Total Count: 9		2021-2022	REVIEW REQUIRED	Search Here			
S.No.	Name	Mobile	Application No.	Farmer ID	Account No	Status	Action
1	Dollar Test One		2127000000124	22000003560	CCAABBHIJK011		
2	Fields Test	9676876876	2127000000102	22000003522	CD0111		
3	Dollar test two	9675675898	2127000000099	22000003485	764486743333		
4	Dollar Testing Full Auto two	9999999999	2127000000098	22000002454	764486745566		
5	Dollar Testing Full Automation	8374753757	2127000000097	22000002452	764486745555		
6	Dollar sign next full automation	9473843434	2127000000092	22000002419	764486745357		
7	Rakesh B Testing	8747474747	2127000000089	22000002395	764486745645		

Click on Edit button and enter all the mandatory details from all the Tabs

Application Form

Applicant Details	Account Details	Financial Details	Activity (Multi-Select)	Term Loan Details																								
Personal Details <table border="1"> <tr> <td>Name (As per Aadhaar) *</td> <td>Aadhaar Number *</td> <td>Name (As per passbook) *</td> </tr> <tr> <td>Dollar Test One</td> <td>563229631588</td> <td></td> </tr> <tr> <td>Date Of Birth *</td> <td>Age *</td> <td>Gender *</td> </tr> <tr> <td><input type="date"/></td> <td>0</td> <td>Select</td> </tr> <tr> <td>Social Category *</td> <td>Farmer Category *</td> <td>Farmer Type *</td> </tr> <tr> <td>Select</td> <td>Select</td> <td>Select</td> </tr> <tr> <td>Primary Occupation *</td> <td>Relative Type *</td> <td>Relative Name *</td> </tr> <tr> <td>Select</td> <td>Select</td> <td><input type="text"/> Enter Relative Name</td> </tr> </table>					Name (As per Aadhaar) *	Aadhaar Number *	Name (As per passbook) *	Dollar Test One	563229631588		Date Of Birth *	Age *	Gender *	<input type="date"/>	0	Select	Social Category *	Farmer Category *	Farmer Type *	Select	Select	Select	Primary Occupation *	Relative Type *	Relative Name *	Select	Select	<input type="text"/> Enter Relative Name
Name (As per Aadhaar) *	Aadhaar Number *	Name (As per passbook) *																										
Dollar Test One	563229631588																											
Date Of Birth *	Age *	Gender *																										
<input type="date"/>	0	Select																										
Social Category *	Farmer Category *	Farmer Type *																										
Select	Select	Select																										
Primary Occupation *	Relative Type *	Relative Name *																										
Select	Select	<input type="text"/> Enter Relative Name																										

Date Of Birth *

Age *

Gender *

Mobile Number *

Social Category *

Farmer Category *

Farmer Type *

Primary Occupation *

Relative Type *

Relative Name *

Residential Address

State *

District *

Block / Subdistrict *

Village *

Street Address

Pincode *

UPDATE & CONTINUE

Application Form

Applicant Details
Account Details
Financial Details
Activity (Multi-Select)
Term Loan Details

Beficiary details updated successfully

Functional Account Details

State *

District *

Bank *

Branch Name *

Account Number *

Confirm Account Number *

Account Holder *

No. of Account Holder *

S.No	Name	Aadhaar No.	Aadhaar Verification
1	Dollar Test One	563229631588	VERIFIED
2	Test joint account	234523613403	VERIFIED

BACK
UPDATE & CONTINUE

Update the required details and click on **Update & continue** from Account details tab

Application Form

Applicant Details **Account Details** Financial Details Activity (Multi-Select) Term Loan Details

Functional Account Details

State *	District *	Bank *	Branch Name *
MAHARASHTRA	Mumbai	State Bank Of India	BOMBAY-DADAR-SBIN0000353

Account Number *	Confirm Account Number *	Account Holder *	No. Of Account Holder *
.....	CCAABBHIJK011	JOINT	2

S.No	Name	Aadhaar No.	Aadhaar Verification
1	Dollar Test One	563229631588	VERIFIED
2	Test joint account	234523613403	VERIFIED

BACK
UPDATE & CONTINUE

Application Form

Applicant Details Account Details **Financial Details** Activity (Multi-Select) Term Loan Details

Financial Details

KCC loan sanctioned / KCC renewed on *

KCC limit sanctioned (INR) *

KCC drawing limit for current FY (INR) *

KCC limit tenure (months) *

Next KCC renewal date *

BACK
UPDATE & CONTINUE

Account details updated successfully

Update the desired details from financial details and click on **update and continue** button

Applicant Details Account Details Financial Details **Activity (Multi-Select)** Term Loan Details

Activity Details

Choose Activity Type

AGRI CROPS **HORTI & VEG CROPS** **ANIMAL HUSBANDRY** **FISHERIES**

Agri Crops

Loan Sanctioned (INR) * Loan Sanctioned On *

100 11/11/2021

Land Location Hierarchy *
(State, District, Subdistrict, Village) Crop * Survey Number * Khata Number * Land Area *
(hectare) Land Type * Season *

ODISHA,Gajapati,Mohana Select Survey No. Khata Number Land Area Select Select

ADD

Update the required details from Activity details

Fisheries (Marine Fisheries & Aquaculture)

Loan Sanctioned (INR) * Loan Sanctioned On *

200 11/11/2021

Land Location Hierarchy *
(State, District, Subdistrict, Village) Marine fisheries and allied activities * Total Nos *

ODISHA,Gajapati,Mohana,Gutuma REGISTERED FISHING VESSEL/BOAT 10

ADD

BACK **ACTIVITY SUMMARY** **UPDATE & CONTINUE**

Click on Preview button from Term Loan Details

Application Form

Applicant Details	Account Details	Financial Details	Activity (Multi-Select)	Term Loan Details								
S.No.	Description			Loan Sanctioned (INR)								
1	Agri & Veg Crops			₹ 3000								
2	Horti Crops			₹ 3700								
3	Animal Husbandry			₹ 1500								
4	Inland Fisheries & Aquaculture			₹ 1500								
Loans sanctioned for current FY				₹ 9,700								
<table border="1"> <thead> <tr> <th>Description</th> <th>Amount (INR)</th> </tr> </thead> <tbody> <tr> <td>KCC drawing limit for current FY</td> <td>₹ 10,000</td> </tr> <tr> <td>Loans sanctioned for current FY</td> <td>₹ 9,700</td> </tr> <tr> <td>Term Loan For Current FY (KCC Drawing Limit For Current FY - Loans Sanctioned For Current FY)</td> <td>₹ 300</td> </tr> </tbody> </table>					Description	Amount (INR)	KCC drawing limit for current FY	₹ 10,000	Loans sanctioned for current FY	₹ 9,700	Term Loan For Current FY (KCC Drawing Limit For Current FY - Loans Sanctioned For Current FY)	₹ 300
Description	Amount (INR)											
KCC drawing limit for current FY	₹ 10,000											
Loans sanctioned for current FY	₹ 9,700											
Term Loan For Current FY (KCC Drawing Limit For Current FY - Loans Sanctioned For Current FY)	₹ 300											

PREVIEW

(Image for representation purpose)

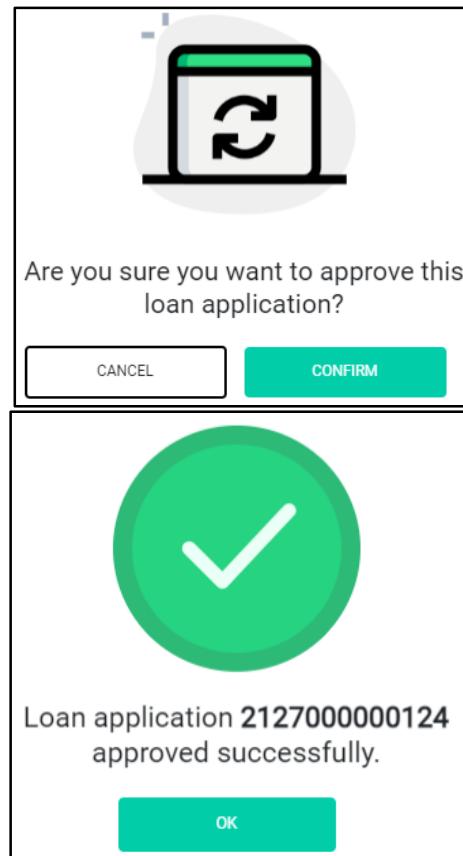
Click on Approve button

Horti Crops	
Loan Sanctioned (INR) ₹ 200	Loan Sanctioned On 11/11/2021
Animal Husbandry	
Loan Sanctioned (INR) ₹ 300	Loan Sanctioned On 11/11/2021
Inland Fisheries & Aquaculture	
Loan Sanctioned (INR) ₹ 200	Loan Sanctioned On 11/11/2021
Marine Fisheries & Aquaculture	
Loan Sanctioned (INR) ₹ 200	Loan Sanctioned On 11/11/2021

BACK

APPROVE

Click on Confirm button from the Popup displayed and check that the Application is approved successfully.



~~~ END OF DOCUMENT ~~~