

AadhaarAPI Credit Score Gateway [Equifax]

Introduction

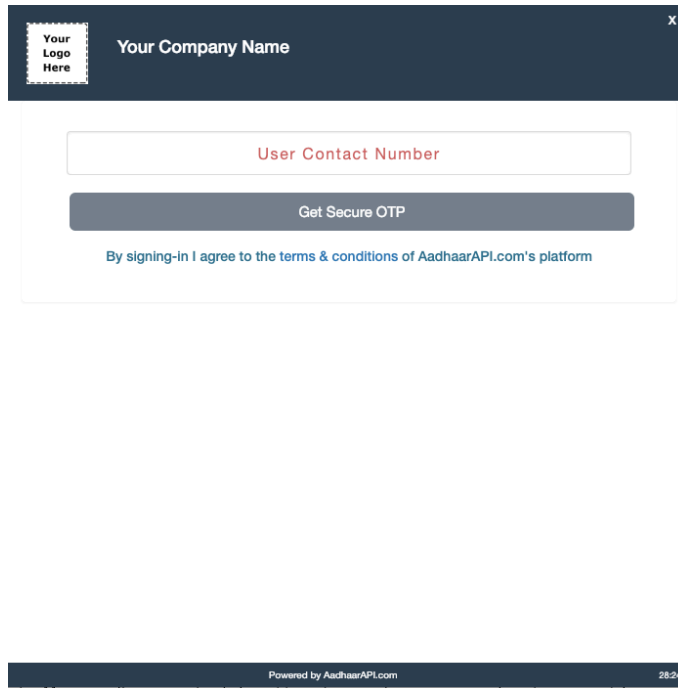
AadhaarAPI provide WEB and Mobile gateway for fetching Credit Score of Users. Using these gateways any organization onboarded with us can get credit report of their customer who have provided consent for fetching same from Bureau.

Process Flow:

1. At your backend server, Initiate the credit score transaction using Rest API [POST] call. Details of these are available in the documents later. You will require API key and Agency Id for accessing this API which can be generated from the Dashboard.
2. This gateway transaction id then needs to be communicated back to the frontend where SDK functions are to be called.
3. After including the SDK files (JS & CSS) at frontend and a small HTML snippet, client has to pass above generated transaction id to an SDK function to create a new gateway object and then open the gateway using another function call.
4. This will open the gateway as shown in above image and the rest of the process till response will be handled by the gateway itself.
5. Once the transaction is successful or failed, appropriate handler function will be called with response JSON, that can be used by the client to process the flow further.
6. Result PDF/XML/JSON will be sent to the returnUrl requested via INIT call.
7. Client will also have a REST API available to pull the status of a gateway transaction from backend and reason of failure.

End User Flow:

1. Customer Login [Phone + OTP]



The form is titled "Your Company Name" and features a placeholder for "Your Logo Here". It contains a text input field for "User Contact Number", a "Get Secure OTP" button, and a link to the terms and conditions. The footer indicates it is "Powered by AadhaarAPI.com" with a timestamp of 28:24.

Your Logo Here

Your Company Name

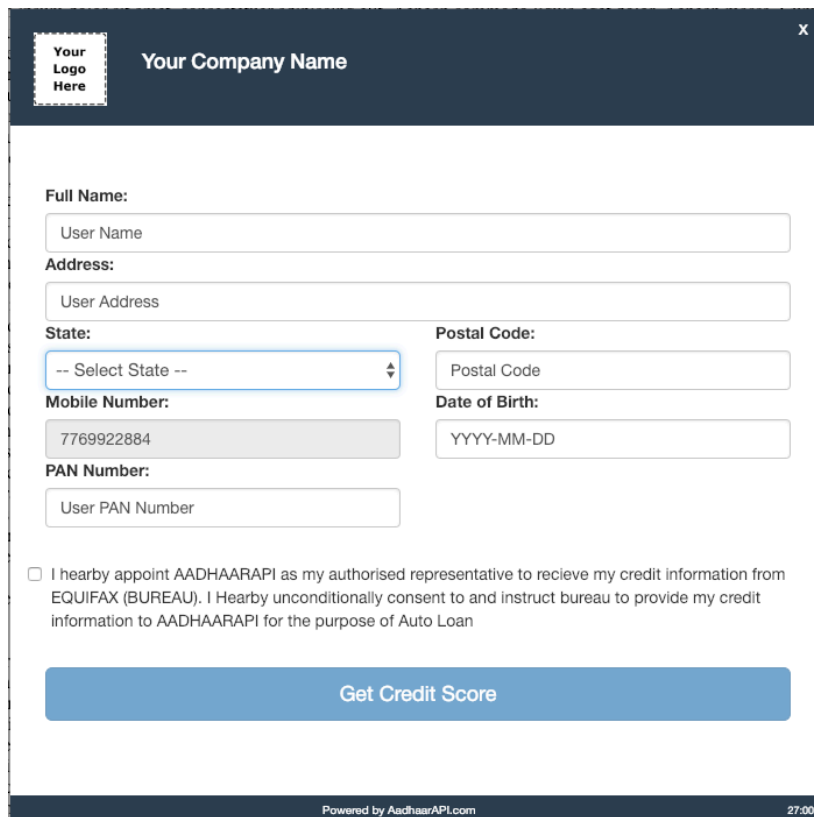
User Contact Number

Get Secure OTP

By signing-in I agree to the terms & conditions of AadhaarAPI.com's platform

Powered by AadhaarAPI.com 28:24

2. User fill's in his Personal Information Which can also be auto-filled using the SDK



The form is titled "Your Company Name" and features a placeholder for "Your Logo Here". It contains fields for "Full Name", "Address", "State", "Postal Code", "Mobile Number", "Date of Birth", and "PAN Number". It also includes a checkbox for consent and a "Get Credit Score" button. The footer indicates it is "Powered by AadhaarAPI.com" with a timestamp of 27:00.

Your Logo Here

Your Company Name

Full Name:

User Name

Address:

User Address

State:

-- Select State --

Postal Code:

Postal Code

Mobile Number:

7769922884

Date of Birth:

YYYY-MM-DD

PAN Number:

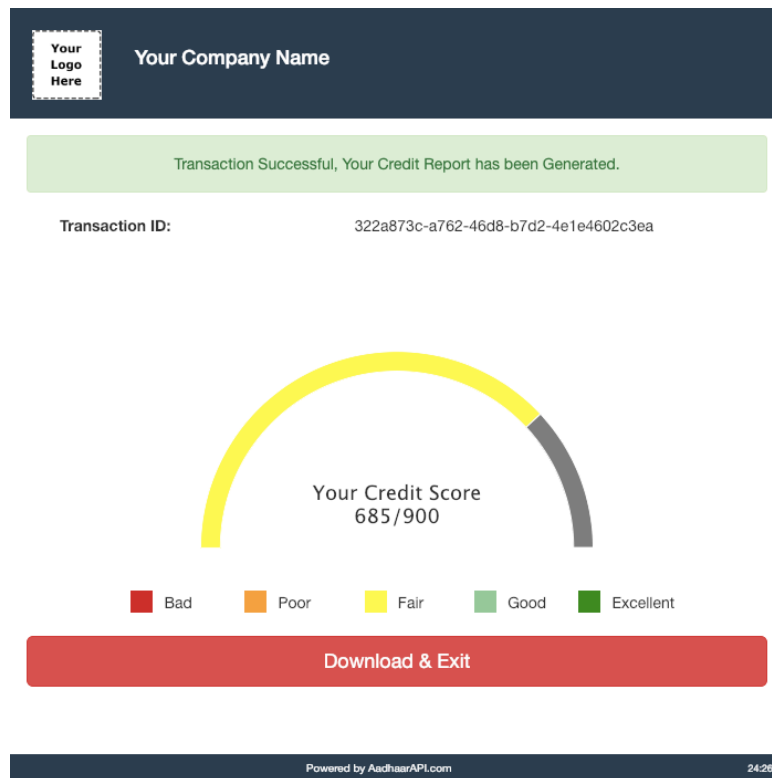
User PAN Number

☐ I hereby appoint AADHAARAPI as my authorised representative to recieve my credit information from EQUIFAX (BUREAU). I Hearby unconditionally consent to and instruct bureau to provide my credit information to AADHAARAPI for the purpose of Auto Loan

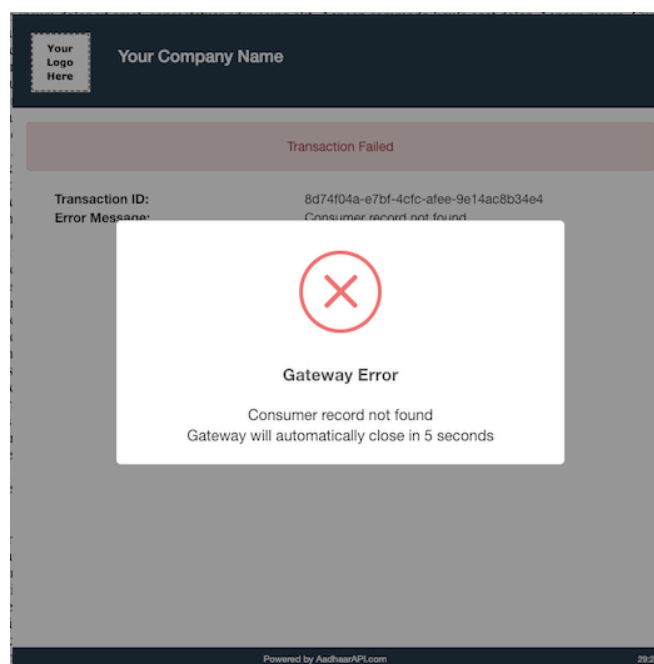
Get Credit Score

Powered by AadhaarAPI.com 27:00

- On successful transaction Customer gets his Credit Score on the screen with option to download the Credit Report. The Credit report is also shared with the agency on their responseUrl.



- On failure during request to Bureau, customer is displayed an error code and error message. Same error details are sent to the responseURL.



Initiating a Gateway Transaction [IP Whitelisted in production]

To initiate a gateway transaction a REST API call has to be made to backend. This call will generate a Gateway Transaction Id which needs to be passed to the frontend web-sdk to launch the gateway.

POST {{base_url}}/gateway/creditscore/init/

Example URL: <https://preprod.quagga.in/gateway/creditscore/init/>

Request Headers: [All Mandatory]

QT_API_KEY: <<your api key value – [available via Dashboard](#)>>
QT_AGENCY_ID: <<your agency id value – [available via Dashboard](#)>>
Content-Type: application/json

Request Body Params: [All Mandatory]

```
{
  "purpose": "<<Purpose code (List of codes is mentioned Below)>>",
  "response_url": "<<POST[REST] URL to which response is to be sent after the transaction is complete>>",
  "format": "<<pdf json or xml >>"
}
```

Parameters	Description/Values	Checks
format	Format in which response would be shared with the agency	Format must be PDF,XML or JSON only
purpose	Purpose of transaction (List of codes mentioned below)	Purpose code must be available in the list.
response_url	POST API URL where the Agency receives the response after the transaction is completed.	A valid POST API URL, else response back to your server will fail.

Response Params:

```
{
  "id": "<<transactionId>>",
  "env": "<< environment code>>",
  "response_url": "<<response url of agency>>"
}
```

The above generated gateway [transactionId](#) has to be made available to frontend to open the gateway.

[Note: A transaction is valid only for 30 mins after generation.](#)

Configuring and launching the gateway

1. Add following CSS file for gateway placeholder before the body tag.

```
<link rel="stylesheet" href="/aadhaar-api-web-sdk.css">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

2. Add following HTML at the end of your page in which you want to open the gateway as a popup.

```
<div id="quaggaModal" class="qmodel">
  <div id="quaggaModelContent" class="qmodel-content"></div>
</div>
```

2. Add following javascript after the above HTML tag

```
<script type="application/javascript" src="aadhaar-api-web-sdk.js"></script>
```

Note: Make sure the files are included in proper order and aadhaar-api-web-sdk.js has correct base URL in it.

4. After above javascript you need to call the following functions to open the gateway.

- a. Setup the gateway UI to match your application.

```
var creditScoreGatewayOptions = {
  company_display_name: '<<Add your company name here>>', //(required)
  front_text_color: 'FFFFFF', //(optional)Add the hex for colour of text of company name
  background_color: '2C3E50', //(optional)Add the hex for background colour to be set for gateway.
  logo_url: 'https://your-square-product-logo-image-url-here.png', //(required)
  full_name: 'User Name', // (optional)
  address_line: 'Address of customer', //(optional)
  state: 'STATE CODE', //(optional, List Of state Codes available below),
  postal: 'User ZIP Code', //(optional)
  dob: 'DOB in YYYY-MM-DD', //(optional)
  pan_number: 'User Pan Number', //(optional)
  customer_phone: 'User Phone Number' // (optional, To be used to login and fetch credit score records)
};
```

Test data [preproduction]

```
full_name: "SATHISHKUMAR M SATHISH KU",
address_line: "223 NACHIANNAN ANGANNAN STREET RAMANATHAPURAM COIMBATORE",
state: "TN",
postal: "641045",
dob: "1986-06-03",
pan_number: "BUXPS2681D",
customer_phone: "User Phone Number"
```

b. Create a new gateway transaction from gateway transaction id.

```
var myAadhaarCSGateway = new CreditScoreGateway (transactionId, creditScoreGatewayOptions);
```

c. Open the created gateway.

```
openCSGateway(myAadhaarCSGateway);
```

- This can be bind to a button click for opening the gateway when needed. For ex:

```
document.getElementById("gatewayBtn").onclick = openGateway;

function openGateway () { openCSGateway(myAadhaarCSGateway) }
```

Once the gateway is open, rest of the flow till response handover is handled by the gateway itself.

d. Handling the different gateway responses. [Keep these functions even if not implemented]

```
function handleCreditScoreSuccess(responseJSON) {

    console.log ('Handling EKYC success at client end');

    console.log(responseJSON);

    //Handle the case when user ekyc is success

}

function handleCreditScoreFailure(errorJSON) {

    console.log ('Handling EKYC failure at client end');

    console.log(errorJSON);

    //Handle the case when user ekyc fails

}

function handleGatewayError(errorJSON) {

    console.log ('Handling Gateway launch failure at client end');
```

```

    console.log(errorJSON);
}

function handleGatewayTermination () {

    console.log ('Handling Gateway Termination at client end');

}

```

Note: Do not run sample files provided directly, instead run them on some local server or hosted machine. The gateway looks for domain name in the URL to communicate back the response securely with the page launching the gateway. This requires a proper URL like <http://localhost:8080/test> or <https://myserver.com/test>. URLs like <file://c:/test/sample.html> will not work.

Transaction Response sent to ResponseURL:

Success:

```

{
  "transaction_id": "<<your transaction ID>>",
  "agency_id": "<<your agency id>>",
  "env": "<<transaction environment>>",
  "CreditScore": "<<credit score>>",
  "detailedResponseFormat": "<<requested format of report>>",
  "detailedResponse": "<<Response in desired format>>",
  "transaction_status": 5
}

```

Failure:

```

{
  "transaction_id": "<<transaction id>>",
  "agency_id": "<<your agency id>>",
  "env": "<<transaction environment>>",
  "error_code": "<<error code>>",
  "error_msg": "<<error message>>",
  "transaction_status": 6
}

```

Pulling Transaction Status at Backend:

In case the POST API call to the response URL fails, there is an option to pull the transaction status from backend using the same Transaction Id.

GET [{{base_url}}/gateway/esign/:esign_transaction_id/fetch/](#)

Response Params:

```
{
  "id": "<<your transaction ID>>",
  "agency_id": "<<your agency id>>",
  "env": "<<enviornment>>",
  "transaction_status": "<<transaction status>>",
  "public_ip": "<<IP address of the request>>",
  "request_medium": "<<request medium>>",
  "transaction_attempts": "<<transaction attempts>>",
  "response_to_agency": "<<If response has been sent successfully to agency URL or not>>",
  "error_code": "<<error code>>",
  "error_msg": "<<error message>>",
  "product_code": "<< The type of report requested from bureu >>",
  "format": "<<response format of detailed report>>",
  "response_url": "<<agency url where response is sent>>",
  "request_timestamp": "<<request timestamp>>",
  "response_timestamp": "<<response timestamp>>"
}
```

Annexure 1 – Transaction Status

INITIATED	1
INITIATION_FAILED	2
OTP_SENT	3
OTP_FAILED	4
SUCCESSFUL	5
FAILED	6
OTP_MISMATCH	7
FP_MISMATCH	8
EXPIRED	9
CONSENT_DENIED	10
TERMINATED_BY_USER	11
OTP_REQUEST_LIMIT_CROSSED	12
OTP_FAILURE_LIMIT_CROSSED	20
OTP_EXPIRED	13
LOGIN_SUCCESS	14
USER_NOTIFIED	18

Annexure 2 – Purpose List

0	Other
1	Auto Loan
2	Housing Loan
3	Property Loan
4	Loan against Shares/Securities
5	Personal Loan
6	Consumer Loan
7	Gold Loan
8	Education Loan
9	Loan to Professional
10	Credit Card
11	Lease
12	Overdraft
13	Two-wheeler Loan
14	Non-Funded Credit Facility
15	Loan Against Bank Deposits
16	Fleet Card
17	Commercial Vehicle Loan
18	Telco - Wireless
19	Telco - Broadband
20	Telco - Landline
31	Secured Credit Card
32	Used Car Loan
33	Construction Equipment Loan
34	Tractor Loan
35	Corporate Credit Card
3A	Auto Lease
51	Business Loan
52	Business Loan-Priority Sector-Small Business
53	Business Loan - Priority Sector- Agriculture
54	Business Loan - Priority Sector- Others
55	Business Non-Funded Credit Facility
56	Business Non-Funded Credit Facility - Priority Sector - Small Business
57	Business Non-Funded Credit Facility - Priority Sector - Agriculture
58	Business Non-Funded Credit Facility - Priority Sector - Other
59	Business Loan Against Bank Deposits
60	Staff Loan
8A	Disclosure

Annexure 3 – Error Codes List

Sr. No	Error Code	Error Message
1	E0001	Customer not active.

2	E0002	Customer ID not found.
3	E0005	Incorrect user or password.
4	E0007	Customer ID entered is invalid.
5	E0010	User ID entered is invalid.
6	E0013	User has exceeded the maximum number of incorrect logon attempts.
7	E0019	User password expired.
8	E0021	User ID does not exist for the given customer.
9	E0022	User ID is not active.
10	E0023	User Account is locked.
11	E0024	User not Found in the database.
12	E0025	User does not have WS access.
13	E0026	Member Number invalid.
14	E0027	Member number has been assigned to a different customer.
15	E0028	Product code does not exist in system.
16	E0029	An Internal Error has occurred.
17	E0032	Please provide any one of the following ID information: Tax ID (PAN Card), Passport ID, Driver License.
18	E0033	Request XML is not well-formed.
		<u>Actual error response message may from below list when input:</u>
		value is not compliant with XSD pattern.
		string value '<Input Value>' is not a valid enumeration value for InquiryPurposeOptions.
		Invalid decimal value: expected at least one digit. (i.e. for TransactionAmount)
		string value '<Input Value>' does not match pattern for type of FullName element in RequestBodyType.
		string value '<Input Value>' does not match pattern for type of FirstName element in RequestBodyType.
		string value '<Input Value>' does not match pattern for type of MiddleName element in RequestBodyType.
		string value '<Input Value>' does not match pattern for type of LastName element in RequestBodyType.
		string value '<Input Value>' is not a valid enumeration value for StateCodeOptions.
		Invalid date value: <Input Value> (i.e. for DOB)
		string value '<Input Value>' is not a valid enumeration value for MaritalStatusOptions.
		string value '<Input Value>' is not a valid enumeration value for GenderOptions.
		string value '<Input Value>' does not match pattern for type of element PANId.
		string value '<Input Value>' does not match pattern for type of element PassportId.
		string length (3) is less than minLength facet (7) for type of element PassportId.
		string value '<Input Value>' does not match pattern for type of element VoterId.
		string length (5) is less than minLength facet (10) for type of element VoterId.
19	E0034	Invalid product Version ID.
20	E0035	Phone(Mobile) is invalid. Please enter only numeric data.
21	E0036	Phone(Mobile) is invalid. Please enter only numeric data,
		cannot be less than 10 numbers.

22	E0037	Phone(Mobile): Mobile phone number cannot start with 1.
23	E0038	Phone(Mobile): Mobile phone number cannot repeat.
24	E0039	Phone(Home): Home phone number is invalid. Please enter
		numeric data only.
25	E0040	Phone(Home): Home phone number cannot be less than 5
		numbers.
26	E0041	Phone(Home): STD Code starting with 00, cannot have phone
		number less than 9 numbers.
27	E0042	Phone(Home): Phone Number cannot repeat.
28	E0043	At least one of the Personal IDs is mandatory.
29	E0044	At least one of the Home Phone Number or Mobile Phone
		Number is mandatory.
30	E0045	Inquiry Purpose is invalid.
31	E0046	System temporarily unavailable.
32	E0047	Inquiry Purpose must be specified.
33	E0048	Address information not present.
34	E0049	Address too long. Address length should not exceed 220
		characters.
35	E0050	First Name is mandatory.
36	E0051	Full Name is mandatory.
37	E0052	Either (AdditionalName1, AddtionalType1) or
		(AdditionalName2, AdditionalType2) must be specified.
38	E0053	Additional name types must be different in case of Father,
		Mother, Husband, Wife.
39	E0054	Invalid Additional Name Type 1.
40	E0055	Inquiry Purpose is invalid.
41	E0056	At least one of the ID Information or Phone numbers is mandatory.
42	E0057	At least one of the ID Information or Phone numbers is mandatory.
43	E0058	Transaction Amount is invalid. It should be numeric.
44	E0059	Date of birth is in wrong format.
45	E0060	First Name is either too long or contains some invalid characters.
46	E0061	Middle Name is either too long or contains some invalid characters.
47	E0062	Last Name is either too long or contains some invalid characters
48	E0063	Full Name is either too long or contains some invalid characters.
49	E0064	Additional Name 1 is either too long or contains some invalid characters.
50	E0065	Additional Name 2 is either too long or contains some invalid characters.
51	E0066	Postal Code (PIN) is in wrong format.
52	E0067	State code is either in wrong format or invalid.
53	E0068	State code must be specified.
54	E0069	PAN card is in wrong format.
55	E0070	Invalid Additional Name Type 2.

56	E0071	Passport ID is in wrong format.
57	E0072	Voter ID is in wrong format.
58	E0073	Gender is wrong.
59	E0074	Member Number is in wrong format.
60	E0075	Security Code is in wrong format.
61	E0076	Company ID, User Name and Password must be specified.
62	E0077	Member Number and Security Code must be specified.
63	E0078	Branch ID too long. Branch ID length should not exceed 30
		characters.
64	E0079	Kendra ID too long. Kendra ID length should not exceed 30
		characters.
65	E0080	Additional Search Field too long. Additional Search Field length
		should not exceed 30 characters.
66	E0081	Postal Code (PIN) must be specified.
67	E0082	Ration Card too long. Ration Card length should not exceed 20 characters.
68	E0083	Additional ID 1 too long. Additional ID 1 length should not exceed 20
		characters.
69	E0084	Additional ID 2 too long. Additional ID 2 length should not exceed 20
		characters.
70	E0085	National ID too long. National ID length should not exceed 20 characters.
71	E0086	Date of birth must be specified.
72	E0087	Customer Reference Number is too long. Customer Reference
		Number length should not exceed 20 characters.
73	E0088	Customer ID / Company ID must be a number.
74	E0089	User ID is invalid. It is either too long or contains invalid characters.
75	E0090	Product code does not exist in system.
76	GSWDOE116	Inquiry parameters:ID - PAN does not match with credit file
77	GSWDOE116	Inquiry parameters:ID - Voter does not match with credit file
78	GSWDOE116	Inquiry parameters:ID - Driver license does not match with credit file
79	GSWDOE116	Inquiry parameters:ID - National ID does not match with credit file
80	GSWDOE116	Inquiry parameters:ID - Passport does not match with credit file
81	GSWDOE116	Inquiry parameters:DOB does not match with credit file
82	GSWDOE116	Inquiry parameters:Name does not match with credit file
83	GSWDOE116	Inquiry parameters:Phone does not match with credit file
84	GSWDOE117	N3 connection not available for given Member Number and Config ID
85	GSWDOE117	Error occurred on N3 request
86	GSWDOE117	Timeout occurred on N3 request
87	99	Unknown Error – This is returned when no report or error info is received from the Bureau.

Moving to Production

1. Contact Business team for access
2. Submit IP address of production testing and live production server(s) for whitelisting
3. Post approval from Business Team generate API key from Dashboard
4. Change base URL in INIT & FETCH API call to <https://prod.aadhaarapi.com>
5. Change QUAGGA_BASE_URL to 'https://prod.aadhaarapi.com' (no trailing /) in SDK js file
6. Test and Go live.