NESL Asset Data Limited

(A wholly owned subsidiary of National e-Governance Services Limited)

RFP. No: NADL/Account Aggregation/2018/001
Date: 28th June 2018

Request for Proposal
for
Selection of Vendor for Design, Development, Installation, Integration, Configuration, Support & Maintenance of Account Aggregation software

Administrative Office:
NESL Asset Data Limited(NADL)
5th Floor, Spencer Towers,
86, M.G. Road,
Bengaluru – 560001
Phone: - 080 -25580360,022- 22446619
e-mail:- procurement@nadl.co.in

RFP SCHEDULE

RFP. No: NADL/Account Aggregation/2018/001 **Date: 28th June 2018**

Name of the company	NESL Asset Data Limited, Bengaluru
Date of Release of RFP	28th June, 2018
Last Date and Time of receiving pre-bid vendor queries in writing	04th July 2018, 1700 Hrs
Date and Place of Pre-bid Meeting	11th July, 2018, 1100 Hrs National E-Governance Services Ltd 5th Floor, Spencer Towers, 86, M.G. Road, Bengaluru – 560001
Last Date and Place of submission of Bids	25th July, 2018, 1500 Hrs National E-Governance Services Ltd 5th Floor, Spencer Towers, 86, M.G. Road, Bengaluru – 560001
Date, Time and Place of Technical Presentation	27th July, 2018, 1100 Hrs National E-Governance Services Ltd 5th Floor, Spencer Towers, 86, M.G. Road, Bengaluru – 560001
Date, Time and Place of opening of Technical Bids	31st July, 2018, 1530 Hrs National E-Governance Services Ltd 5th Floor, Spencer Towers, 86, M.G. Road, Bengaluru – 560001
Date, Time and Place of opening of Commercial Bids(Tentative)	03rd August, 2018, 1100 Hrs National E-Governance Services Ltd 5th Floor, Spencer Towers, 86, M.G. Road, Bengaluru – 560001
Application Fee	Rs. 2360/- in the form of Demand Draft drawn in favour of NESL Asset Data Limited., payable at Bengaluru.
Contact Information	AVP - IT National E-Governance Services Limited 5th Floor, Spencer Towers, 86, M.G. Road, Bengaluru – 560001 E-Mail : procurement@nadl.co.in

DEFINITIONS

- “Account Aggregator” means a non-banking financial company as notified under in sub-clause (iii) of clause (f) of section 45-I of the Act, that undertakes the business of an account aggregator, for a fee or otherwise, as defined at clause (iv) of sub-section 1 of section 3 of the “Non-Banking Financial Company - Account Aggregator (Reserve Bank) Directions, 2016" issued by Reserve Bank of India.

- "Bank" means - a banking company; or a corresponding new bank; or the State Bank of India; or a subsidiary bank; or such other bank which the Bank may, by notification, specify for the purposes of these directions; and a co-operative bank as defined under clause (cci) of section 5 read with section 56 of the Banking Regulation Act, 1949 (10 of 1949)

- “business of an account aggregator” means the business of providing under a contract, the service of, retrieving or collecting such financial information pertaining to its customer, as may be specified by the Bank from time to time; and consolidating, organizing and presenting such information to the customer or any other financial information user as may be specified by the Bank;

- The Central Registry provides the FIP and AA public key information so ecosystem components can validate the digital signatures of these entities.

- “Company” means a company registered under section 3 of the Companies Act, 1956 or a company registered under sub section (20) of section 2 of the Companies Act, 2013;

- The Corporate Identity Number is a unique 21-digit alpha-numeric number given to all Private Limited Company, One Person Company, Limited Company, Section 8 Company, Nidhi Company and Producer Company registered in India.

- A consent artefact is a machine-readable electronic document that specifies the parameters and scope of data sharing that a user consents to in any data sharing transaction.

- “Financial information user” means an entity registered with and regulated by any financial sector regulator; FIU can request, retrieve and store financial information based on the Customer consent.

- “Depository” means a company which has been granted a certificate of registration under sub-section (1A) of section 12 of the Securities and Exchange Board of India Act, 1992;

- “Financial Information” means information in respect of the following with financial information providers:

    - Bank deposits including fixed deposit accounts, savings deposit accounts, recurring deposit accounts and current deposit accounts,
    - Deposits with NBFCs

- Structured Investment Product (SIP)
- Commercial Paper (CP)
- Certificates of Deposit (CD)
- Government Securities (Tradable)
- Equity Shares
- Bonds
- Debentures
- Mutual Fund Units
- Exchange Traded Funds
- Indian Depository Receipts
- CIS (Collective Investment Schemes) units
- Alternative Investment Funds (AIF) units
- Insurance Policies
- Balances under the National Pension System (NPS)
- Units of Infrastructure Investment Trusts
- Units of Real Estate Investment Trusts
- Any other information as may be specified by the Bank for the purposes of these directions, from time to time;

- "Financial information provider" means bank, banking company, non-banking financial company, asset management company, depository, depository participant, insurance company, insurance repository, pension fund and such other entity as may be identified by the Bank for the purposes of these directions, from time to time; FIP provides the financial information based on Customer consent.

  Examples of FIPs include under different regulators
  FIPs under PFRDA
  - Central Record Keeping Agency (CRA)
    - NSDL
    - Karvy Computer Share

  FIPs under RBI
  - Banks
  - NBFCs

  FIPs under IRDA
  - Insurance Companies
  - Insurance Repositories
    - CDSL Insurance Repository Limited (CDSL IR)
    - Karvy Insurance Repository Limited Audit
    - National Insurance-policy Repository by NSDL Database Management Limited
    - CAMS Insurance Repository Services Limited

  FIPs under SEBI
  - Depository
  - Stock Exchange
  - Mutual funds and their registrars

- “Financial information user” means an entity registered with and regulated by any financial sector regulator;

- “Financial Sector Regulator” for the purpose of these directions, shall mean the Reserve Bank of India, Securities and Exchange Board of India, Insurance Regulatory and Development Authority and Pension Fund Regulatory and Development Authority

- “Customer” for the purpose of these directions means a ‘person’ who has entered into a contractual arrangement with the Account Aggregator to avail services provided by the Account Aggregator

- "Person" means a) an individual, b) a Hindu undivided family, c) a company, d) a firm, e) an association of persons or a body of individuals, whether incorporated or not, and f) every artificial juridical person, not falling within any of the preceding sub-clauses. It defines an entity that may maintain account(s) with one or more FIPs. The customer provides the consent for access to his/her FI
- “User” for the purpose of this document means a “Customer”.

Participants & Stakeholders

The participants in the Account Aggregator Ecosystems are

1. Account Aggregator (AA)
2. Financial Information Provider (FIP)
3. Financial Information User (FIU)
4. Financial Sector Regulator (FSR)
    a. PFRDA
    b. RBI
    c. IRDA
    d. SEBI
5. User
6. Central Registry

TABLE OF CONTENTS

SECTION – I: INVITATION OF RFP

1. Background

National e-Governance Services Limited (NeSL) is India's first Information Utility and is registered with the Insolvency and Bankruptcy Board of India (IBBI) under the aegis of the Insolvency and Bankruptcy Code, 2016 (IBC). The company has been set up by leading banks and public institutions and is incorporated as a union government company**.**

NESL Asset Data Limited (NADL), a Non-Banking Financial Company is a wholly owned subsidiary of NESL. NADL proposes to provide the services of an Account Aggregator duly retrieving, collecting, consolidating, organizing and presenting financial information pertaining to customers, whereby customers would benefit from single view of their financial information

NADL invites proposals from eligible vendors for Design, Development, Installation, Integration, Configuration, Support & Maintenance of Account Aggregation software (hereafter referred as Design and Deployment) as given in **Section – IV: Schedule of Requirements**.

2. Contact Information

Assistant Vice President - IT
National e-Governance Services Limited
5th Floor, Spencer Towers,
86, M.G. Road,
Bengaluru – 560001
E-Mail: procurement@nadl.co.in

3. Pre-Bid Meeting

The pre-bid meeting will be held at address and on dates as given in the RFP schedule above, to sort out/resolve queries raised by the prospective bidder regarding the scope, terms & conditions, etc. The prospective bidders requiring any clarification on the RFP document may send their queries in writing through email. NADL will respond to these queries during the pre-bid meeting. The queries/doubt/clarifications etc. must be sent before the date and time given in RFP schedule above. After the stipulated date and time, no queries would be entertained.

4. How to Apply

The documents as listed below (but not limited to) should be submitted in the four respective **SEALED** envelopes, as given below.

Envelope – 1:

i. Demand Draft of Rs. 2360/- towards processing Fee (including GST).

ii. Demand Draft of Rs. 3,20,000/- (Rs. Three Lakh Twenty Thousand only) towards Earnest Money Deposit or the valid documents, if any exemption from payment of EMD is claimed.

Envelope – 2:

i. Covering letter as per **Annexure – A.**

ii. Letter of Authority as per **Annexure – B.**

iii. A copy of Certificate of Incorporation, Partnership Deed / Memorandum and Articles of Association / any other equivalent document as applicable, showing date & place of incorporation and nature of business / activities.

iv. The copies of the audited Profit and Loss Account or a certificate from a Chartered Accountant, showing the annual turnover and profit for each of the financial years 2016-2017, 2015-2016 and 2014-2015.

v. Copies of PAN and GST registration certificates.

vi. Document/s showing bidder has office/ establishment in/around Bangalore.

vii. Other documents necessary in support of eligibility criteria (Section - II, para 4), product catalogues, brochures, etc.

Envelope – 3:

i. List of clients for whom the bidder has developed and deployed the application software of similar nature, in last five years.

ii. The details of application software developed and deployed by the bidder in last five years, giving details like technology platform used, the scope, volume, spread of software, the size of data, number of transactions, speed / latency, key features, etc.

iii. Copies of at least three supply orders / deployment reports, in support of sub-para 6 of para 4 (Eligibility Criteria) in Section – II.

iv. List of Technical and Administrative personnel on roll of the bidder, giving details of their educational qualifications (with specializations, if any), experience in the specific area as required for this project, etc. **(Annexure – C)**

v. Technical Proposal including (but not limited to) understanding about the project, implementation Methodology, team composition, work schedule, PERT and Activity Schedule, interactions / visits, Data safety/ security measures, Quality Control, modular

structure, escalation hierarchy, technologies /platforms to be used, requirements from NADL/ clients.

vi. The technical proposal should detail the technical architecture and explain the platform scalability for various transaction volumes, flexibility towards modifications, etc. It should also detail the mapping of the proposed technology platform onto the Infrastructure detailed in **Annexure – D**

vii. A statement as per **Annexure - F**, showing bidder's compliance with the technical requirements covering all the parameters (but not limited to) stipulated at para 3: Application Software Requirements, page 32 to 94, Section -IV of this document. The bidder may please note that simply complying with the requirements does not automatically make the bidder technically qualified.

viii. The details required for technical evaluation of bids, pertaining to parameters mentioned onpage 16 to 18, Section - II, at para 8 ii (tables A to D), in tabular form.

ix. Checklist of documents to be submitted, as per **Annexure - E.**

Envelope – 4:

The Price Bid as per format given in **Section - V.**

5. Submitting the Proposal:

All applications must be sent to the contact address as given in the RFP schedule. The applications should be in a sealed envelope. All the four envelopes should be put in a large outer envelope. The outer cover of the envelope should be sealed and super scribed with the following.

NADL: Technology Platform for Development and Deployment of Account Aggregation Software **RFP No. NADL/Account Aggregation/2018/001, dated 28th June, 2018.**

This submission should reach NADL on or before the last date of submission of RFP as given in the RFP schedule. **NADL will not take any liability** for proposals **received late**, for any reasons. If the last date for the receipt of applications mentioned above gets declared a Public Holiday, the last date will be the next working day.

6. Validity of Bids:

The bids submitted against this RFP shall be valid for a period of 90 days from the last date of submission mentioned in the RFP schedule

7. Last Date of submission of RFP Documents

Last date for submission of RFP documents is given in the RFP schedule. The documents can be submitted in person or can be sent through mail/ courier, so as to reach on or before the last date and time stipulated in this document. NADL shall not be responsible for postal / courier delay, if any, or any other reason for non-receipt of document in the specified time and will result in disqualification / rejection of the bid.

8. Opening of RFP Documents

The RFP Documents will be opened, in the presence of the bidders or their authorized representatives, who choose to attend, on the date as given in RFP schedule**,** at the address given at para 2 above. The representatives (maximum two, with an authority letter from the applicant) of interested parties are welcome to attend the opening of the RFP documents.

(END OF SECTION-I)

SECTION II: INSTRUCTIONS TO BIDDERS (ITB)

1. Locations for Deployment of technology platform and for providing services:

At the Datacentre and Disaster Recovery sites of NADL and their client locations within India.

2. Project Timelines:

The successful bidder should complete the process of Design, Development, Installation, Integration and Configuration of Account Aggregation software at NADL within a period of 6 months from the date of placement of order. Subsequent to the above successful deployment, the bidder will be required to provide warranty and maintenance services for the warranty period mentioned in the order.

The order will initially cover the warranty, technical support and maintenance period of 2 years, after successful deployment of the software. However, NADL may extend this period with mutual consent on the same terms and conditions by another 3 years(maximum). The extension shall be for a period of one year at a time.

3. Order Placements:

The Supply Order and payments shall be released by:

NESL Asset Data Limited (NADL),
5th Floor, Spencer Towers,
86, M.G. Road,
Bengaluru – 560001
Phone- 080 -25580360, E-Mail: procurement@nadl.co.in

4. Eligibility Criteria:

1. The bidder must satisfy the eligibility criteria stipulated below. However, fulfilling the eligibility criteria does not automatically mean that their bid is qualified.

2. The bidder should submit the financial instruments, documents and information stipulated at para 4, “How to Apply”, Section – I.

3. The annual sales turnover of bidder in the last two financial years should be at least Rs.20 Crores.

4. The bidder must have at least 5 years of experience in the area of development of software and providing IT services, etc. to clients from financial sector.

5. The bidder should be a profit making company (profit after tax) in at least last financial year

6. Bidder should have executed at least 3 IT projects in the BFSI space, with each project worth Rs. 1 Crore or above. Statutory auditors certificate by the bidder should be relied upon.

7. The bidder must have the qualified manpower having relevant experience and in requisite number, on their rolls, as listed in **Annexure – C.**

8. The bidders must have Office and Service Centre at Bengaluru. The bidder shall enclose the relevant documents in support of this requirement, indicating local address and contact number.

9. Fit and proper criteria as per normal regulatory dispensation, and that there have been no regulatory actions initiated / pending against the bidder by any public authority. Suitable declaration to this effect has to be provided.

10. The bidder shall not be permitted to do/attempt any reverse engineering of the software developed and / IT service, up to a period of two years from the date of completion of the project delivery and acknowledged by NADL of satisfactory receipt. The IPR of the Account Aggregation technology developed shall vest with NADL perpetually. The bidder must submit an undertaking to this effect as per format given in **Annexure - A.**

11. The bidder should have implementation experience in using open source platform, versatile in technology platform such as API, Consent management, data security, etc. for at least one organization in India (Central Government/State Government/PSU/Private Sector Company).

12. The bidder must not be blacklisted/suspended by RBI/ UIDAI/Financial/Educational /Govt. Organizations or debarred from bidding process, as on date of submission of the bids.

Notes:

1. The bidders should provide sufficient documentary evidence to support the eligibility criteria. NADL reserves the right to reject any bid not fulfilling the eligibility criteria.

2. If in the view of bidder, any exemption / relaxation is applicable to them from any of the eligibility requirements, under any Rules / process/ Guidelines/ Directives of Government of India, bidder may submit their claim for the applicable exemption /relaxation, quoting the valid Rule/ process/ Guidelines/ Directives. In this case the bidder must submit necessary and sufficient documents along with the technical bid, in support of his claim. The bid evaluation committee is empowered to take appropriate decision about the claim towards exemption/ relaxation of the bidder.

5. Amendment to Bidding Documents

a. At any time prior to the deadline for submission of bids, NADL may, for any reason, whether on its own initiative or in response to the clarification request by a prospective bidder, modify the bid document.

b. The amendments to the RFP documents, if any, will be notified by release of Corrigendum Notice on https://www.nesl.co.in/tenders/ against this RFP. The amendments/ modifications will be binding on the bidders.

c. NADL at its discretion may extend the deadline for the submission of bids if it thinks necessary to do so or if the bid document undergoes changes during the bidding period, in order to give prospective bidders time to take into consideration the amendments while preparing their bids.

d. The bidder may modify, withdraw or resubmit its bid, before the last date and time of submission of bids.

6. Preparation of Bids

Bidder should avoid, as far as possible, corrections, overwriting, erasures or postscripts in the bid documents. In case however, any corrections, overwriting, erasures or postscripts have to be made in the bids, they should be supported by dated signatures of the same authorized person signing the bid documents. However, bidder shall not be entitled to amend/ add/ delete/ correct the clauses mentioned in the entire tender document.

7. Earnest Money Deposit (EMD)

a. The Earnest Money Deposit (EMD) must be submitted prior to the DUE DATE & TIME of submission of the online technical bid. The EMD is required to be in the form of Demand Draft/ Banker's Cheque in favour of NESL Asset Data Ltd. payable at Bengaluru, India, for an amount of Rs. 3,20,000/- (Rupees Three Lakh Twenty Thousand only).

b. The bidder may claim the exemption from submission of EMD, if eligible. In this case, the bidder must clearly mention the applicable Rule/ Law / Provision under which the exemption is being claimed. The bidder must also submit the necessary and sufficient current and valid documents in support of this claim. The Bid Evaluation Committee of NADL is empowered to decide on grant of exemption on merit, whose decision on the same shall be final and binding on the bidder The bid submitted without EMD or valid exemption documents shall stand rejected. No interest shall be payable on EMD.

c. The EMD will be returned to the bidder(s) whose offer is not accepted, within 30 days from the date of finalization of successful bidder. In case of the bidder whose offer is accepted, the EMD will be returned on submission of Security Deposit(SD), as stipulated at para 5, Section – III of this document. No interest on EMD will be payable to the bidder.

d. NADL reserves the right to forfeit the EMD, if,

    i. The bidder withdraws the bid during the period of bid validity specified in the RFP.

ii. The successful bidder fails to furnish the acceptance in writing, within 15 days of placement of order.

iii. The bidder fails to submit the Security Deposit, as stipulated at para 5, Section – III of this document.

8. Bid Opening & Evaluation of Bids

The technical bids will be evaluated in two steps.

i. The bids will be examined based on eligibility criteria stipulated at Para 4 of Section – II to determine the eligible bidders.

ii. The technical bids of only the eligible bidders shall be further evaluated based on Quality and Cost Based Selection (QCBS) method given below.

The evaluation will be done broadly on 5 parameters (Four Technical parameters & one Financial parameter in the ratio 70:30) with marks and weightage as defined below: -

A: BFSI Institutions as clients:

Sr. No.	Number of Banking, Financial Services and Insurance Institutions as clients	Marks
1	3	5
2	4	6
3	5	7
4	6	9
5	More than 6	10

B: Experience in development of software similar to the requirements of Account Aggregation:

Sr. No.	No of software developed in similar to Account Aggregation domain in last 5 years	Marks
1	1	5
2	2	6
3	3	7
4	4	9

5	More than 4	10

C: Size of Skilled Technical Team available on rolls to build the Account Aggregator Software platform:

Sr. No	Number of Skilled Technical Team	Marks
1	Up to 10	5
2	11 – 15	6
3	16 – 20	7
4	21 – 25	9
5	More than 25	10

D: Technical Presentation:

Sr. No	Evaluation Parameters	Marks
1	The presentation will be mainly evaluated against the following parameters: ● Overall Technical Architecture ● Scalability of the Platform ● Capability of the Platform (in terms of projected transactions/sec, request/response time, etc.) ● Data Security Architecture ● Utilization of Open Source Software ● Commercial Software License Requirements ● Infrastructure Requirements for Platform deployment to address a transaction volume of say, 50 Crores per month (Virtual Machines, Storage, Network Bandwidth, etc.)	50

E: Financial Criterion: The financial score (FS) will be calculated by comparing the price quoted by each bidder with the lowest price quoted.

The illustrative example given at para 9 below may be referred for further clarity.

iii. The minimum qualifying marks for the technical parameters stipulated at A to D above shall be 50. The bidders getting marks less than 50 will be disqualified.

iv. The price bid of the disqualified bidders will not be opened.

v. The bidders whose technical bid is found to meet both the requirements as specified at 8 (i) and 8(ii) above will qualify for opening of their commercial bids. The Technical Score (TS) secured by each qualified bidder shall be informed to the bidders present during the commercial bids opening meeting. The date and venue of the commercial bids opening will be informed separately.

vi. The duly constituted Bid Evaluation Committee (BEC) shall evaluate the bids. The BEC shall be empowered to take appropriate decisions on minor deviations, if any. The bidder's name, bid prices, discounts and such other details considered as appropriate by NADL, will be announced at the time of opening of the commercial bids.

9. Comparison of Bids

i. The Combined Technical and Financial Score (CTFS) with Weightage 70:30 (70 for Technical and 30 for Financial) will be calculated.

ii. The Combined Technical and Financial Score (CTFS) will be taken for comparison of bids and for deciding bidder securing highest score.

iii. An illustrative example for CTFS is given below.

Stage 1: **Technical Marks:**

Bidder details	**Total Technical Marks Obtained for parameters 7 (ii) A to D**
Bidder 1	75
Bidder 2	65
Bidder 3	45
Bidder 4	55
Bidder 5	65

Bidder 3 will be disqualified as the total technical Marks are below 50.

Stage 2: **Conversion of Technical Marks into Technical Score**

Bidder details	Technical Score (TM/MTM)*100	TS
Bidder 1	(75/80)*100 = 93.75	93.75
Bidder 2	(65/80)*100 = 81.25	81.25
Bidder 3	Disqualified	Not calculated
Bidder 4	(55/80)*100 = 68.75	68.75
Bidder 5	(65/80)*100 = 81.25	81.25

TM= Technical Marks; MTM = Maximum Technical Marks; TS = Technical Score

Stage 3: Prices Quoted: The total prices quoted at Section - V, including taxes will be considered for calculating Financial Score.

Bidder details	Price Quoted
Bidder1	100
Bidder2	75
Bidder4	55
Bidder 5	65

Stage 4: Conversion of Financial Bid amount to financial score

Bidder details	Financial Score (LFB/F)*100)	FS
Bidder1	(55/100)*100= 55.0	55.0
Bidder2	(55/75)*100 = 73.33	73.33
Bidder4	(55/55)*100 = 100	100
Bidder 5	(55/65)*100 = 84.61	84.61

LFB= Lowest Financial Bid, F = Financial Bid, FS = Financial Score

Stage 5: Combined Technical and Financial Score (CTFS) with Weightage 70:30

Bidder Details		Weightage of 70 % for Technical & 30 % for Financial Score	CTFS	Rank of the Bidder
Bidder 1		70% of 93.75 + 30 % of 55.0	65.62+ 16.5= 82.12	**2**
Bidder 2		70 % of 81.25 + 30 % of 73.33	56.87 + 21.99 = 78.86	3
Bidder 4		70 % of 68.75 + 30 % of 100	48.12 + 30 = 78.12	4
Bidder 5		70 % of 81.25 +30 % of 84.61	56.87 +25.38 = 82.25	1

10. Placement of Order:

The Works Order will be placed on the bidder securing highest Combined Technical and Financial Score (CTFS), as evaluated from the method described at para 8 and 9 above.

The Order will be placed for the amount as quoted in Part -A of Price Bid, Section - V. The prices quoted at Part - B shall be used for carrying out a Change Order, if any, during the project period (Please refer para 4 of Section - III).

However, NADL reserves the right and has sole discretion to reject the bid securing highest Combined Technical and Financial Score (CTFS).

In case, more than one bidders secure same Combined Technical and Financial Score (CTFS), NADL reserves the right to place order on the bidder having more experience in the area of development of application software of similar nature.

NADL reserves the right to place order / contract on the sole bidder or the sole qualified bidder.

Before the placement of order, the successful bidder is required to sign a Service Agreement (SA), a Mutual Non-Disclosure Agreement (MNDA), Deed of Indemnity with NADL and any such agreements as required by regulators, such as, RBI, UIDAI, CA and CCA. The terms and conditions of these agreements would be mutually decided, before placement of Order.

11. Purchaser's Right to amend / cancel

NADL reserves the right to amend the eligibility criteria, commercial terms & conditions, Scope of Supply, technical specifications, etc.

NADL reserves the right to cancel the entire tender without assigning any reasons thereof.

12. Corrupt or Fraudulent Practices

It is expected that the bidders who wish to bid for this project have highest standards of ethics. NADL will reject bid if it determines that the bidder recommended for award has engaged in corrupt or fraudulent practices while competing for this RFP.

NADL may declare a vendor in-eligible for placement of Order, either indefinitely or for a stated duration, if it at any time it determines that the vendor has engaged in corrupt and fraudulent practices during the placement / execution of Order.

13. Interpretation of the clauses in the RFP Document

In case of any ambiguity/ dispute in the interpretation of any of the clauses in this RFP Document, the interpretation of the clauses by Director, NADL shall be final and binding on all parties.

(END OF SECTION II)

SECTION III: SPECIAL CONDITIONS OF CONTRACT (SCC)

1. Price:

a. The price quoted shall be considered firm and no price escalation will be permitted (except Govt. Statutory Levies), till completion of deliverables / obligations of the successful bidder, as stipulated in the Order.
b. Bidder must quote in INR only and as per price bid format given in **Section – V.**
c. The exact rate and amount of GST currently applicable must be mentioned in the `Price Bid format'. The statutory taxes and duties applicable at the time of completion of activity shall be applicable. NADL will not issue any exemption certificate.
d. The bidder should exercise utmost care to quote the correct percentage of applicable GST. In case due to any error/ oversight, the GST rate quoted by the bidder is different than the actual GST rate as per the tariff, the bidder will not be permitted to rectify the error/oversight. The orders/ contract will be placed with the GST rate quoted by the bidder or actual tariff rate, whichever is LOWER. The difference amount payable, if any, between the quoted GST rate and actual tariff rate shall be borne by the bidder ***by adjustment in the basic price***

2. Software Licenses (if applicable):

The bidder must submit the list and details of software licenses required, if any, for the development of required software. These software licenses will be in the name of NADL.

3. Completeness Responsibility:

Notwithstanding the scope of work, engineering, supply and services stated in bid document, any equipment or material, engineering or technical services which might not be even specifically mentioned under the scope of supply of the bidder and which are not expressly excluded there from but which – in view of the bidder - are necessary for the performance of the equipment in accordance with the specifications are treated to be included in the bid and has to be performed by bidder. The items which are over & above the scope of supply specified in the Schedule of Requirements may be marked as "Additional Items" in Section - V.

4. Change Orders

4.1 The Vendor agrees that the requirements given in the RFP, are broad requirements and are in no way exhaustive and may be modified at the sole discretion of NADL, without any change in time or cost to NADL

4.2 It shall be the responsibility of the Vendor to meet all the requirements of technical criteria contained in this RFP and any upward revisions and / or additions to specifications of the Bid

required to be made shall not constitute a Change Order and shall be carried out without a Change Order and shall be carried out without any time and cost effect to NADL.

4.3 Any upward revision and/or additions consequent to errors, omissions, ambiguities, discrepancies in the specification etc., which the Vendor had not brought to NADL's notice at the time of the Bid, shall not constitute a Change Order and such upward revisions and/or addition shall be carried out by the Vendor without any time and cost effect to NADL.

4.4 The Change Order will be initiated only in case:

4.4.1 NADL directs the Vendor in writing to include any addition to the Scope of work covered under this RFP or delete any part of the Scope of work under this RFP; or

4.4.2. The Vendor requests to delete any part of the work which will not adversely affect the implementation of the Scope of Work under this Agreement and if the deletions proposed are agreed to by NADL and for which cost and time benefits shall be passed on to NADL; or

4.4.3 NADL directs in writing the Vendor to incorporate changes or additions to the technical criteria requirements under this RFP.

4.5 Any changes reasonably required by NADL over and above the minimum requirements given in the Scope of work included in the RFP, before giving its approval to detailed design for complying with technical criteria and changes required to ensure systems compatibility and reliability for safe (as per codes, standards and recommended practices referred in the RFP, Bid) and trouble free operation shall not be construed to be change in the Scope of Work under this RFP. Also, change in codes, APIs, protocols and standards, post submission of the bid, attributable to NADL, regulators, such as RBI and their agencies such as ReBIT shall also not be construed to be change in the Scope of Work under this RFP.

4.6 Any Change Order comprising an alteration which involves change in the cost of the works (which sort of alteration is hereinafter called a "Variation") shall be the subject of an amendment to the Order / Contract by way of an increase or decrease in the Order / Contract Price and adjustment of the implementation schedule, if any.

4.7 If there is a difference of opinion between the Vendor and NADL's Representative whether a particular work or part of the work constitutes a Change Order or not, the matter shall be handled in accordance with the procedures set forth in the next clause 'Procedures for Change Order'.

4.8 Within 14 working days of receiving the comments from NADL on the specification, purchase requisitions and other documents submitted by the Vendor for approval, the Vendor shall respond in writing, which item(s) of the comments is/are potential changes(s) in the Scope of Work covered in this Agreement and shall advise a date by which Change Order (if applicable) will be submitted to NADL.

5. Procedures for Change Order

5.1 During the implementation/maintenance or at any stage during the tenure of the project, if the vendor observes that any new requirement which (other than that required for meeting the technical criteria) is not specific or intended by the Scope of Work has been requested by NADL it shall verbally discuss the matter with the Representative of NADL which may be reduced to writing in note filings or minutes of meeting.

5.2 In case such requirement arises from the side of the Vendor, he would also verbally discuss the matter with the NADL's Representative giving reasons thereof which may be reduced to writing in note filings or minutes of meeting.

5.3 In either of the two cases, the Representatives of the Parties shall discuss on the new requirement for better understanding and shall mutually decide whether such requirement constitutes a Change Order or not.

5.4 If it is mutually agreed that such requirement constitutes a Change Order, then a joint memorandum will be prepared and signed by the Vendor and NADL to confirm a Change Order and basic ideas of necessary agreed arrangement.

5.5 Upon completion of the study referred to above under clause 5.4 above, the results of this study along with all relevant details including the estimated time and cost effect thereof with supporting documents would be submitted to NADL to enable NADL to give a final decision whether the Vendor should proceed with the Change Order or not in the best interest of the works. The estimated cost and time impact indicated by Vendor shall be considered as a ceiling limit and shall be provisionally considered for taking a decision to implement Change Order. The time impact applicable shall be mutually agreed, subsequently, on the basis of the detailed calculations supported by all relevant back up documents. In case the Vendor fails to submit necessary substantiation/calculations and back up documents, the decision of NADL regarding time and cost impact shall be final and binding on the Vendor.

5.6 If NADL accepts the implementation of the Change Order under Clause 5.5 above in writing including the adjustment to the Contract Price, which would be considered as Change Order, the Vendor shall commence to proceed with the relevant work stipulated in the Change Order on signing of the final agreement between the Parties with regard to adjustment of the Contract Price and implementation schedule.

5.7 Examples of Major and Minor changes

Minor changes will not lead to change orders and the vendor shall implement it without any additional cost to NADL. Major changes may lead to change orders. Typical examples of Minor and Major changes are listed below. These examples are only indicative and not exhaustive.

A. Below are few examples of minor changes,
    a. Any modification and addition of financial information types, such as pension funds, unit linked insurance and public provident fund

b. Any modification and addition of the schemas of the financial information types.
c. Any modification due to changes in the specifications of the Central Registry.
d. Any modification due to changes to the security principles (initiated by ReBIT/ NADL or other regulatory bodies) used in the platform for both data-in-transit and data-at-rest.
e. Any modification to meet the changes in the authentication and authorization protocol, initiated due to any regulatory requirement in Aadhaar usage/ Digital signatures etc.
f. Any modification to meet the changes in the API specification details in the ReBIT technical specs.
g. Any modifications due to changes to addition of new dashboards/metrics/charts for business reporting/ monitoring
h. Any modification required to support Joint Holders for various assets.
i. Bug fixes of all types.
j. Performance and Scale improvements as specified by NADL.

B. Below are few examples of major changes,
a. Any addition of asset Types, such as, Real Estate, Healthcare, Telecom, and Personal
b. Any major new feature addition other than specified here or in the RFP
c. Any new language support for Localization.

6. Security Deposit (SD):

Within 15 days of award of placement of Order, the successful bidder must submit the Security Deposit @ 5 % of Order value, in the form of Demand Draft favouring NADL or in the form of Bank Guarantee.

The Security Deposit will be returned to the successful bidder on successful development and deployment of application software and against submission of Performance Security as stipulated at para 9, Section – III. NADL will not pay any interest on the Security Deposit.

7. Warranty:

The supplier shall warrant that the software to be supplied shall be free from all defects and faults, shall be of the highest grade and consistent with the established and generally accepted standards of the type ordered and shall perform in full conformity with the requirements and drawings as per **Section - IV**. The supplier shall be responsible for any defect that may develop, arising from faulty algorithms/design, errors, bugs, inadequate quality to meet requirements and/or otherwise, and shall remedy such defects at his own cost when called upon to do so by the Purchaser who shall state in writing in what respect the software functionality are faulty. The warranty period shall be as stipulated at para 2, Section - II.

If it becomes necessary for the Supplier to modify any defective portion(s) of the software under this clause, the provisions of the Clause 7.3 shall apply to the portion(s) of the software modified.

The warranty shall include 24 X 7 onsite support with 4 hrs response time and 24 hrs (max) resolution time.

8. Inspection and Acceptance Criteria:

8.1 On completion of each module/ functionality of the development and deployment of software, NADL will run suitable test cases to ensure quality and functionality. The deployed technology platform will also be tested/ audited, under the supervision of NADL, as per the directives / guidelines laid down by RBI/UIDAI /CCA/CA, from time to time.
If any part of the software is found not meeting the requirements, the same shall be modified free of all cost to the purchaser.

8.2. If any software or any part thereof, before it is taken over under Clause 8.3 fails to fulfil the requirements of the Order, the Purchaser/ Consignee shall give a notice to the Contractor, setting forth details of such defects or failure and the Contractor shall modify the software to comply with the requirements of the Order forthwith and in any case within a period not exceeding 15 days of the initial report. These replacements /modifications shall be made by the Contractor free of all charges at site. Should it fail to do so within this time, the purchaser reserves the discretion to reject and replace or rectify/modify, at the cost of Contractor, the whole or any portion of the application software as the case may be, which is defective or fails to fulfil the requirements of the Order. The cost of any such replacement/rectification made by the purchaser shall be deducted from the amount payable to the Contractor.

8.3. When the intended functionality of the software called for have been successfully carried out, the consignee or its authorised representative of NADL will issue a Taking over Certificate, normally within two weeks of successful completion of tests, including the security audit of the application.

8.4. Nothing in Clause 8 as above shall in any way release the Contractor from any warranty, penalty or other obligations under this RFP.

The process of inspection and acceptance shall be applicable for the modifications and/ or additions, required - if any, to be incorporated in application software to be developed. The time required for such modifications /additions shall not be counted for calculating penalty.

9. Performance Security:

The bidder shall support the performance and warranty service of the developed and deployed software with a Bank Guarantee.

On successful development and deployment of application software, for claiming the last instalment of 10 % payment, the successful bidder shall submit the Performance Security in the form of a Bank Guarantee of the equivalent amount. This Bank guarantee shall be valid for the warranty period and shall be from a commercial bank and should be negotiable at a bank branch in Bengaluru. The PBG shall be as per the format given at **Annexure - E.**

10. Payments:

A**: Payment for Development and Deployment of software (Sr. No. 1, Part A of Price Bid, Section - V)**

Completion of first milestone – 30% of order / contract amount on acceptance as detailed in para 8 “Inspection and Acceptance Criteria” of Section III. First Milestone requires delivery and acceptance of the following functionalities:

- User Flow through Mobile App
    - Registration
    - Login
    - Consent Request
    - Data Request
    - Report Generation and Delivery
    - Notification
    - Authentication & Authorization
- Single FIP integration, E2E
- First milestone to be completed within 2 months from the award of contract.

Completion of second milestone – 30% of order /contract amount on acceptance as detailed in para 6 “Inspection and Acceptance Criteria” of Section III. Second Milestone requires delivery and acceptance of the following functionalities:

- FIU Flow
    - Registration
    - Login
    - Consent Request
    - Data Request
    - Web Portal & API support
    - Authentication & Authorization
- User Flow
    - Web Portal
- 10 FIP integrations E2E
- 5 FIU integrations E2E
- Second milestone to be completed within 4 months from the award of contract.

Completion of balance functionalities - 40% of order / contract amount on acceptance, as detailed in para 8 “Inspection and Acceptance Criteria” of Section III and against submission of performance bank guarantee of equivalent amount, as per para 9 Section – III of this document. All functionalities to be completed within 6 months from the award of contract.

B: Payment for Warranty, support and maintenance services:(Sr. No. 2 Part A of Price Bid, Section - V)

25 % of the yearly amount towards warranty and support services, as quoted at **Sr. no.2 Part A** of Price Bid, shall be released at the end of every quarter after issuing of Taking Over Certificate for developed software. This payment shall be subject to the fulfilment of warranty and support service requirements by the bidder, as stipulated in the Order. Applicable TDS will be deducted from all payments.

11. Penalties:

a. In case of delay in successful development and / or deployment of required software, NADL reserves the right to levy a penalty @ 0.5 % of the order value per week for first 4 weeks of delay. Thereafter NADL reserves the right to levy a penalty @ 1.0 % of the order value, per week for further delay. The penalty shall be maximum of 10 % of the Order value.
b. NADL reserves the right to cancel the order in case the delay in satisfactory development and /or deployment of application software or any part thereof is more than 10 weeks.
c. The delay in development / deployment arising out of conditions of Force Majeure and for the delay attributed to the NADL will not be considered for the purpose of calculating penalties.
d. If regulatory authorities, such as, RBI, UIDAI and CCA impose any penalty (monetary or otherwise) for non-compliance of their requirements or for breach of any rule, the same will be imposed on Vendor on back-to-back basis. This will be over and above the penalties stipulated at para 11 a above.
e. The penalties, if any will be recovered from the Security Deposit or Performance Bank Guarantee submitted by the bidder.

12. Jurisdiction

The disputes, legal matters, court matters, if any shall be subject to Bengaluru jurisdiction only.

13. Force Majeure:

NADL may consider relaxing the penalty and delivery requirements, as specified in this document, if and to the extent that, the delay in performance or other failure to perform its obligations as stipulated in the Order, is the result of a Force Majeure. Force Majeure is defined as an event of effect that cannot reasonably be anticipated such as acts of God (like earthquakes, floods, fire, storms etc.), acts of states / state agencies, the direct and indirect consequences of wars (declared or undeclared), hostilities, national emergencies, civil commotion and strikes at successful Bidder's premises or any other act beyond control of the bidder.

14. Arbitration:

In case any dispute arises between NADL and successful bidder with respect to this RFP, including its interpretation, implementation or alleged material breach of any of its provisions both the Parties hereto shall endeavour to settle such dispute amicably. If the Parties fail to bring about an

amicable settlement within a period of 30 (thirty) days, dispute shall be referred to the sole arbitrator mutually agreed and appointed by both parties. If the sole arbitrator is not appointed mutually by both the parties, then the District Court Bengaluru shall have exclusive jurisdiction for appointment of sole arbitrator through court. Arbitration proceedings shall be conducted in accordance with the provisions of the Arbitration and Conciliation Act, 1996 and Rules made there under, or any legislative amendment or modification made thereto. The venue of the arbitration shall be Bengaluru. The award given by the arbitrator shall be final and binding on the Parties.

15. Limitation of Liability:

The liability of the vendor / Contractor arising out of breach of any terms/conditions of the RFP / contract/work order and addendums/amendments thereto, misconduct, wilful default will be limited to the total order / contract value. However, liability of the vendor / contractor in case of death/injury/damage caused to the personnel/property of NADL, due to/arising out of/incidental to any act/omission/default/deficiency of bidder/contractor, will be at actuals.

Also, liability of bidder pertaining to claims/ demands by Government agencies, regulatory authorities or third party for losses, penalties, if any, arising in connection with this Contract shall be at actuals.

16. Termination:

Validity of order will remain till fulfilment of all obligations pertaining to development and successful deployment of software including (but not limited to) providing comprehensive warranty, support and maintenance for the period stipulated in the Order.

The successful bidder must acknowledge and agree that the activities of providing satisfactory services for the development and deployment of Account Aggregation software are of paramount importance and matter of immense reputation/pride to nation and NADL. Hence timely performance of all obligations is essence of the Order. Therefore, in case of the delay in providing the stipulated services, and /or defect/under or non- performance pertaining to the services / products supplied by the bidder, NADL will give written notice to the bidder requesting to set the things right within 60 days of notice. If bidder fails to comply with the requirements, NADL shall have the right to cancel the order/s. The successful bidder agrees and accepts that he shall be liable to pay damages claimed by NADL, in the event of cancellation of order, as detailed in the Service Agreement to be signed. The successful bidder may terminate the Service Agreement / Order by at least 30 days' written notice, only in the event of non-payment of undisputed invoices for 90 days from the due date. Except this situation, the successful bidder shall have no right of termination.

NADL reserves the right to terminate the contract / cancel order with or without cause/ reason, by giving 60 days' notice to the successful bidder.

17. Indemnity:

Selected bidder shall save, indemnify and hold harmless NADL from any third party Govt. Claims, losses, penalties, if any, arising in connection with this Contract.

18. Assignment:

Selected bidder/ Party shall not assign, delegate or otherwise deal with any of its rights or obligation under this Contract without prior written permission of NADL.

19. Severability:

If any provision of this Contract is determined to be invalid or unenforceable, it will be deemed to be modified to the minimum extent necessary to be valid and enforceable. If it cannot be so modified, it will be deleted and the deletion will not affect the validity or enforceability of any other provision.

(END OF SECTION - III)

SECTION – IV: SCHEDULE OF REQUIREMENTS

1. Account Aggregator

The Account Aggregator is a Non-Banking Financial Company, expressly setup under the licensing terms of RBI specified in the Master Circular [1], to provide specialized financial information aggregation function for an User (individuals, entities and account holders) from one or more Financial Information Providers (FIPs) where their account(s) exist, based on Consents provided by the User .The master circular [1] defines the role of an account aggregator as an entity involved in 'retrieving, collecting, consolidating, organizing and presenting' financial information pertaining to the customer as a consolidated view of their financial assets. The consolidated view of the assets may be used for various purposes and enable new business models or enhance existing ones.

2. Business Goals & Functional Requirements

2.1 Functional Requirements

Account Aggregators have the following business goals & functional requirements.

SL No.	Business/ Functional Requirements
1	AA shall provide service of retrieving or collecting financial information pertaining to its users, from the FIPs.
2	AA should consolidate, organize and present the financial information from FIPs to its users or any FIU.
3	AA should provide the users with the ability to download the consolidated financial reports in the AA app. They should be able to configure an email or digi-locker where AA can push these reports also.
4	AA shall provide the financial information to an user or FIUs based on the user's explicit consent for sharing own financial data.
5	AA shall provide a mechanism for FIUs to raise a request for consents which can then be further validated and approved by the user.

6	AA shall perform the function of obtaining, submitting and managing the user's consent for the financial assets for which the user wants to share the information.
7	AA should share the consent details given by its user to the corresponding FIP for verification and approval. AA should accept a consent only after FIP has verified it.
8	AA should provide a mechanism for users to retrieve the details of the consents which they have provided for any financial information sharing.
9	An electronic consent artefact shall be capable of being logged, audited and verified.
10	AA should provide a mechanism by which its user can revoke any consent to access the financial information
11	AA should notify users and FIUs as appropriate, whenever any consent is approved, revoked, expired (state change), etc. The notification can be sent over one or more channels like email, SMS, app notifications, or call back API.
12	AA should provide a way for users to sign-up for its services through AA mobile app or web portal.
13	AA should provide a way for FIUs to sign-up to for its services.
14	AA should ensure appropriate mechanism for proper user identification during signup and later when required.
15	AA should ensure appropriate mechanism for user to discover the list of accounts she holds with an FIP.

16	AA should ensure appropriate mechanism for its user to link the financial accounts in FIPs, with AA, so that the financial information for only those assets can be obtained.
17	AA should ensure that the financial information are transferred in a secure way from FIP to AA and from AA to user or FIU and no one other than the intended recipient can get access to it.
18	AA should ensure that all user profile information are stored in a secure way and prevent any unintended access to it.
19	AA should delete all financial information pertaining to its user, once the customer has received those data.
20	AA should never access the authentication credentials of users relating to accounts with various financial information providers.
21	AA should provide a mechanism by which the user can lock the AA account, so that no modification can be made and also no data is shared.
22	At the time of obtaining consent, the Account Aggregator shall inform the user of all necessary attributes to be contained in the consent artefact above and the right of the user to file complaints with relevant authorities in case of non-redressal of grievances.
23	AA should provide detailed audit logs for all data requests, consent related requests, notifications, etc. so that they can be reviewed by the regulator and also used at the time of dispute. These audit logs should contain enough information so that they can also be correlated with the audit logs of FIUs and FIPs. Audit logs should be preserved securely for at-least 5 years.
24	AA should provide reporting on certain business KPIs. These KPIs can be used to provide business insights and can be also used for billing.

25	AA should be in compliance with the Master Direction of RBI and technical specifications from ReBIT published from time to time.

2.2 Types of Financial Information

This section contains examples of the attributes/fields of some of the types of financial information which can be obtained by FIU or User from AA. Detailed xml schemas for some of the financial information types are described in a later section. **These attributes/fields are subject to changes and additional types will also be incorporated. Final schema details will be published by ReBIT in its specification.**

2.2.1 Bank Saving/Current Deposit

1. Account Number
2. Account Holder(s)
3. Account Holder(s) Details
4. Mode of Holding
5. Type of Account
6. Bank and Branch Details
7. Sum of all credit transactions for a time period
8. Sum of all debit transactions for a time period
9. All transactions within a time period.
10. Balance as on date

2.2.2 Bank Term Deposit

1. Account Number
2. Account Holder(s)
3. Account Holder(s) Details
4. Mode of Holding
5. Type of Account
6. Bank and Branch Details
7. Principal Amount
8. Opening Date
9. Maturity Date
10. Tenure
11. Interest Percentage
12. Maturity Amount
13. Interest Payout for a time period
14. Current Value

2.2.3 Insurance

1. Policy Number
2. Policy Term
3. Policy Type
4. Policy Category
5. Coverage Type

6. Policy Holder Details
7. Sum Assured
8. Premium Amount
9. Premium Frequency
10. Commencement Date
11. Policy Status
12. Premium Due Date
13. Premium Amount Date
14. Vested Bonus

2.2.4 Mutual Funds

1. Folio Number,
2. Folio Holder Details
3. Scheme
4. Unit Balance
5. NAV
6. NAV as on
7. Current Value
8. Cost of Investment
9. Dividend Earned
10. Amount
11. Price
12. Balance Units
13. Transaction Details (Purchase, Sip)

2.2.5 Stocks

1. Account Id
2. Account Holders
3. Cash Position
4. Transaction Type (Common stock, Preferred stock, Additional paid-in Capital, Contributed Surplus, retained earnings)
5. Transactions (BSE Symbol, NSE Symbol, Company Name, Exchange, ISIN, Rate, Total Charge, Trade Value)

2.2.6 Bonds

1. Account Id
2. Account Holders
3. Compounding Frequency
4. Credit Rating
5. Current Value
6. Description
7. Face Value
8. Interest Computation
9. Interest On Maturity
10. Interest Pay-out
11. Interest Periodic Pay-out Amount
12. Interest Rising

13. Issue Date
14. Maturity Date
15. Principal Amount
16. Quote
17. Quote Date
18. Symbol
19. Taxable
20. Tenure Days
21. Tenure Months
22. Tenure Years
23. Transaction Types (opening, interest, TDS, instalment, closing)

3. Non Functional Requirements

3.1 Scalability

The architecture should be scalable and upgradable to higher levels of usage. Typically, it should be capable of scaling to the following requirement:

- 500 FIPs
- 1000 FIUs
- 300 millions of registered users
- 100 consent requests/sec
- 1000 data requests/sec

3.2 Availability

The service provided by AA should be highly available for Users, FIPs and FIUs. The target availability metric should be 99.99% (52.56 minutes downtime per year).

3.3 Operability

The services should be easily deployable, upgradable, maintainable. They should have high degree of resilience to failure and need no restarts or reboot. It should be easy to monitor key operational metrics, including health status for the services. Data movement and migration should be minimized. It should use open-source software with support licenses. In case support license is not available, the software selected should be a widely used one with large and vibrant developer community. All software selected should be verified by NADL.

3.4 Latency & Throughput

Latency of an APIs request is the time taken to get the response of an APIs request. Throughput is measured by the number of such API requests per unit time. Latency defines how usable the service is while throughput affects the scalability of the system. The AA public API latencies should be < 80 milliseconds.

3.5 Security

Because of the sensitive nature of the information that AA deals with, security and privacy needs to be extremely important.

3.5.1 Information Security

- Financial Information of a user needs to be completely secured within AA. None of the services and components in AA should be able to decrypt and read the financial data, while it is in transit from FIP to FIU or User through AA. This financial information data should never be stored.
- Any user profile information like Aadhaar/PAN/mobile number and credentials like certificates, keys etc. which are used by AA services and components should be encrypted and stored in secure data vaults.

3.5.2 Network Security

- All API calls between AA and FIUs should happen over SSL layer (HTTPs)
- All API calls between AA and FIPs should happen over SSL layer (HTTPs)
- All internal networks should be protected using firewall and internet facing services should be placed in DMZ.
- Appropriate security should be in place to prevent DDoS attacks

3.5.3 Data Privacy

Financial Information provided through AA should not contain Personally Identifiable Information (PII) information in raw format. Here are a basic set of data masking and identity protection rules that need to be followed to ensure that privacy is maintained:

3.5.3.1 Tokenization

Account numbers, card numbers, phone numbers, personal identifiers (PAN, Aadhaar, etc.) should be tokenized using Virtual IDs issued to the FIU, AA, and FIP (as defined in the User Identifier section of the Consent Artefact XML).

3.5.3.2 Data Masking

For instance, only the last four digits of a credit card number involved in a transaction may be revealed - XXXX XXXX XXXX 1564. Data Masking may be static, on-the-fly, or dynamic.

3.5.4 Identity Verification and Authentication

- When a user signs up with the AA platform, KYC checks are done to verify the identity of the user.
- When user links his or her FIP accounts to receive data, the user's identity should be verified by FIP to ensure that the user is the correct owner of the account.
- When a user gives consent or revoke consents etc., user should be authenticated using OTP.

3.5.5 API security

- All API calls to AA services should be authenticated and authorized.
- All API calls to FIPs should be authenticated and authorized.

- All API calls should use secure https protocol.
- Payloads should be digitally signed by the requester and validated by the receiver for non-repudiation.
- Input parameters should be validated for correct format, syntax and types

3.5.6 Applicable National and International Data Protection Laws & Regulations

Any existing or proposed National Laws, and International Laws, such as GDPR should be evaluated and if applicable, the platform needs to support all compliance requirements.

3.6 Extensibility

The system should be designed and architected with future growth into consideration. It should be easy to extend the system to incorporate new features and use cases as well, to handle higher concurrent load. The system should also adapt to evolving specification of various financial information types.

3.7 Usability

A complex user-interface will generally prevent adoption. Therefore, the UI should be very simple and intuitive. User Experience needs to be minimalist and abstract all the complexities. Any functional flow should be achieved with minimum number of intuitive steps. Please see Reference section for some of the high level guidelines for web and mobile applications.

3.8 Accessibility

Since the AA app can be used by varied type and number of users, the user experience has to be designed following the right accessibility guidelines. Please see the Reference section for some of the guidelines and best practices present in the W3C standards.

3.8.1 Additional guidelines for India provided by Govt. Of India, please see the Reference section for India specific Accessibility guidelines.

3.9 Testability

All services and components should be individually tested using unit, integration, end to end, reliability & performance, security test cases. Unit test cases should be written for all services. Integration test cases should be written to test the APIs of the services in an integration environment along with other services. In end to end test cases the entire system should be tested for all user flows. Security or Penetration tests need to be performed for all services and infrastructure to check any vulnerabilities. Performance test cases should also be run in end to end environment and key latency and throughput metrics should be tracked for each version. Appropriate tools should be used to automate all test runs. Regressions need to be performed with unit, integration, performance test cases for each new build. The builds should be certified once all tests run successfully and metrics obtained satisfy the minimum acceptance criterion.

The following minimum acceptance criterion for test metrics are:

- Unit Tests: 80% coverage
- Integration Tests: 90% coverage
- End to End Tests: 90% coverage
- Security Tests: High and Medium category vulnerabilities need to be addressed.

NADL will employ third party agencies to perform Vulnerability and Penetration Test before go-live and at periodic intervals as determined by NADL. The vendor is expected to fix the high/medium priority vulnerabilities before deploying to production.

3.10 Logging & Auditability

To enable quick debugging, all components and services of AA should log appropriately. Verbosity of logs should be configurable to any of DEBUG/INFO/WARN/ERROR/FATAL level. Log rotation techniques should be used while creating and writing to the log files. All logs should be uploaded to an archive storage and preserved for certain number of days. Additional processing can be done on these logs for anomaly detection. All logs should be searchable. Alerting should also be done by looking at error and exception logs. For monitoring purposes using open source tools such as MonIT, logs should be properly written along with specific syntax, error code etc.

All API requests to AA and their responses should be logged by AA. All notifications between AA, FIPs, FIUs and Users in consent flow, data flow, signup flows should also be logged. The log should contain

1) API request details like URI, http headers etc.
2) Transaction ids
3) Meta information from the payloads

Audit details and interoperability would provide additional reassurance that this data can't be tampered with as it travels between systems. Audit logs need to be encrypted and preserved and should be accessible for verification by the regulators.

3.11 Business Continuity

AA should be deployed in at-least two data centers, one of them should be the primary data center (DC) and another will be for disaster recovery (DR). When the primary data center goes down or is not reachable, it should be easy and quick to route all incoming traffic to DR. Once DC is back, the traffic should be restored back to DC again. The latency of serving a request from DR should not have significant difference than in DC. The data between DC and DR should be continuously synced, preferably through streaming mechanism.

3.11.1 RPO (Recovery Point Objective)

It is the point in time to which systems and data must be recovered after an outage. It defines the amount of data loss that a business can endure. The RPO should be no more than 15 minutes.

3.11.2 RTO (Recovery Time Objective)

It is the time within which systems and applications must be recovered after an outage. It defines the amount of downtime that a business can endure and survive. The RTO should be no more than one hour.

3.12 Monitoring & Alerting

All services and components should be monitored for health. Each service should define health API which can be polled by monitoring agents periodically to determine if the service is down. If the service is down, then Alerts should be generated. Several operational metrics should be tracked in real time (< 1 sec) and displayed in some dashboard. Alerts can be triggered when anomalies are detected in those metrics.

Some of the representative operational metrics which need to be tracked are,

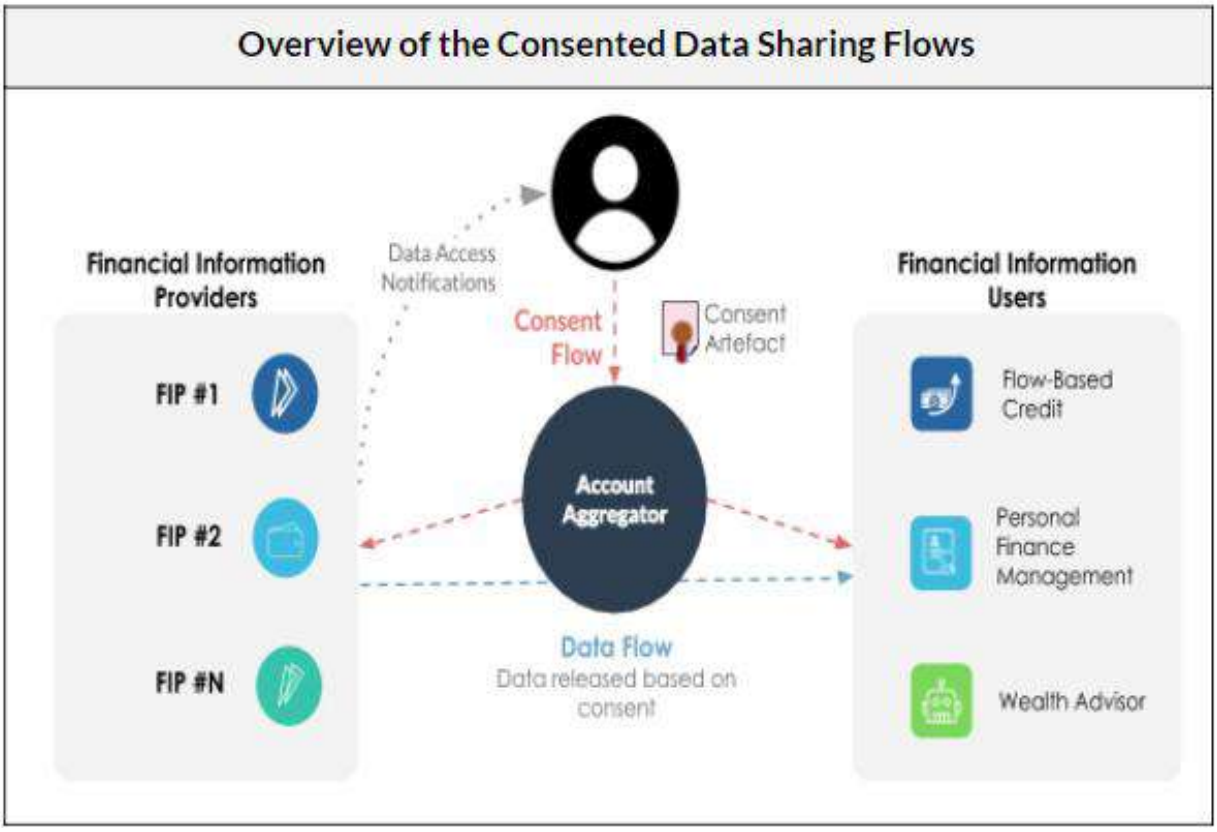
1) System stats like CPU, I/O, memory usage for each VM
2) Number of http API requests to each service
3) Number of success and failures of https requests
4) Average latency of API requests and responses
5) Number of database access for read and write
6) Average query latency for databases
7) Ability to detect long running SQL queries
8) Number of instances of services running
9) Last heartbeat time from a service

4. Consent Driven Framework for AA

Account Aggregator shall provide services to a User or FIU, based on the User's explicit consent about data sharing. The Consent is electronically sought from the User and used to verify any data request by FIU or User herself. The Consent Framework is central to the implementation of the AA. No data about the user will be shared by AA without a valid consent from her.

Consent from user is required when,

1) FIU wants to access User's financial information for certain purpose.
2) User wants to obtain financial information for one or more of his/her FIP accounts.



4.1 Request for Consent

When FIU wants to access User's financial information, it should raise a request for consent, asking for details of types of financial data (e.g. saving bank, mutual funds, stocks, insurance etc) that the FIU wants to access. The User can then provide the details of FIPs and accounts and approve the consent request. The request for consent will have an expiry time beyond which a new request has to be generated. The request for consent should unambiguously indicate the purposes for which the data is being collected.

4.2 Consent Artefact

When a user approves a consent request a set of consent artefacts, one per FIP, is created. A consent artefact is a machine-readable electronic document that specifies all the parameters, scope, purpose, frequency etc. of data share that a user consents to. The consent artefact is also shared with FIP for verification. Once a consent is approved by User as well as verified by FIP, financial information can be shared based on the details in the consent artefact. A consent can be paused, revoked, restarted. A consent can also expire. Data should not be shared on an expired consent.

4.2.1 Consent Artefact for FIU or User

Once the user gives consent to access the financial data from an FIP, a consent artefact is created which is signed by user and stored by AA. Please refer to an example electronic

consent artefact as specified in the ReBIT Technical standard. The actual structure of the artefact may vary from this.

4.2.2 Consent Artefact for FIP

AA uses the User's consent artefact to create another artefact containing information required by FIP; digitally signs it and shares with FIP. FIPs need to validate the consent artefact and details within it. The Verification steps consist of:

1) Verify the credentials of the AA
2) Verify that the artefact is valid and contains all mandatory information.
3) Verify that it has been issued by authorized user who has account with the FIP.

The consent artefact shared with FIP should not contain any details of FIU.

AA should accept any consent once FIP also validates and approves it. At the time of sharing the financial information, FIPs can refer to the appropriate consent artefact details to check the validity of data share. FIP should validate this in real-time. Please refer to an example electronic consent artefact as specified ReBIT Technical standard. The actual structure of the artefact may vary from this.

5. User Application & Flows

Users can access the Account Aggregator service through both mobile app and web portal. Users can download and install the AA mobile app from app stores. AA will support apps for both Android and iOS.

5.1 On boarding & Signup

To sign-up the user needs to provide the mobile number and select a VUA (Virtual User Address) which will identify the user on AA platform. User will also need to choose a password. The VUA and password can be used for subsequent logins. During the sign-up process the user needs to do her KYC. The KYC process is described in later section.

Additionally, a 6-digit PIN can be setup for the mobile App which acts as an additional security measure to prevent the App being used by malicious person.

5.1.1 Virtual User Address

Each User who has an account with the AA is identified by the unique "Virtual User Address" of the form "account@AA_identifier" form. The term "account" should only contain a-z, A-Z, 0-9,. (dot), - (hyphen). This VUA will also be used by the FIU to route the request and connect the User's account maintained by AA. This construct is conceptually similar to the UPI's Virtual Payment Address (VPA)

5.2 KYC

Any user enrolled on the AA platform needs to have the KYC done by AA. This can be done in any of the following way,

1. The user has the option of providing the Aadhaar number, using which e-KYC can be done. Aadhaar based e-KYC can be done within the user's registration flow.
2. The user has the option to upload any scanned copy of the required documents as specified by NADL, the KYC will be done offline and the User will be notified.
3. The user also has the option to visit KYC centres as specified by NADL and provide physical copies of the documents for KYC

Once KYC is successfully done the user's account will be activated.

5.3 Login

To login to the AA App or the web portal, the user uses the VUA and password which was setup during signup. For 2 factor authentication OTP can be used. Password restriction policy should adhere to the password security policy given in the NESL information security policy document, which will be shared along with the service agreement with the successful bidder.

5.4 Discovery & Linkage

Once the user signs in, she needs to discover and link the FIP accounts with AA. There are 2 flows for it.

5.4.1. Discovery Flow

The user selects an FIP from the available list of FIPs. AA uses the mobile number, name, customer relationship number, registered address etc to retrieve the list of accounts the user has with the FIP. These details also need to be registered against the users accounts with FIP. Account numbers should be masked by FIP.

5.4.2. Linkage

Once the accounts with FIPs are listed, user can select individual account and link with AA. During this linking process the user's identity should be verified by FIP. FIP can use registered mobile OTP and registered email for this verification. This step is needed to ensure that the right person is linking to the right account. Once validated, FIP notifies AA to link the account with users' profile. Consent can be given and financial information can be shared only for linked accounts.

5.4.3 De-Linkage

A User can de-link one or more of the account at any point of time. At the time of de-linking, user's identity must be verified by FIP. Once de-linked, all consents which refers to that account for any data access should be revoked automatically and user and FIP should be notified. Alternatively, delinking of account should be allowed when there is no active consent of that account.

5.5 Consent Approval and Consent Artefact Creation

There are multiple ways consents can be created for users to approve

5.5.1 FIU Initiated Flow

There can be several ways by which a consent request can be initiated from FIU. Some of these are,

1) User raising the request by signing into FIU Application / Portal
2) FIU Business Process Workflow raises the consent request through AA APIs
3) FIU raises the request by signing into AA portal

Both (1) and (2) above require FIU to integrate with the open APIs provided by AA. The consent requests should contain details about user identification on the AA Platform, Asset Types, Frequency Total Time, Purpose etc for which the financial information is required. In (1), the user may also provide the FIP accounts for the required Asset types.
Once the consent request is raised, AA platform receives this request. User gets a notification about the consent request upon which she can login to AA application. The user needs to provide all necessary information which is asked for. If the accounts are not linked then the User needs to link them with AA. Once linked, the user can approve the request, digitally signs the consent artefacts. AA stores these artefacts. The user can also reject the consent request from the AA applications.

5.5.2 User Initiated Flow

In this flow, the User can create a consent by specifying the list of FIPs, their accounts and the type of data required on the AA application itself. These accounts should have been already linked with the User's profile. User digitally signs the consent artefact, which is then stored by AA.

5.6 Data Retrieval

Once the consent is approved by the user, financial information of these user can now be shared. There are two flows.

5.6.1 FIU initiated flow

In this flow the data is received by the FIU. The flow is described in the FIU flow section 6.

5.6.2 User initiated flow

In this flow the data is received by the User.

5.6.2.1 One-time

The consent has been given by User for only one time sharing. User initiates the request. AA requests the corresponding FIP for the data as specified in the consent artefact. Once FIP returns the data, AA notifies the User. User logins to the app to retrieve the data. Since AA will store this information only upto certain period, User should get the data within that period. Beyond this period, User needs to initiate this request again, provided the consent artefact is still valid.

5.6.2.2 Periodic

The consent has been given by user for periodic data sharing. The consent artefact should also specify the start time, frequency and total period for which data needs to be shared. User initiates this data share using AA API. After that AA manages the data share for the duration of the consent. At every interval, AA triggers the data request by initiating a call to FIP. Once FIP returns the data, AA notifies the User or FIU. User can login to the app or portal and retrieve the data. FIU calls AA API to fetch the data. Since AA will store User's financial data in a transient storage only up to certain period, User should get the data within that period. Beyond this period, User needs to initiate this request again, using the same consent artefact but providing the period for the data. The consent artefact needs to be valid at the time of this request.

5.7 Financial Report Generation

In the user initiated flow, AA sends notification to User about availability of the financial information from an FIP. The user can login to AA app on his/her device and fetch the data from the AA server. The app will store the data locally on user's device. When all required financial information are available from all the FIPs, user can request for a consolidated report. The app can use the locally stored information and generate the reports. The app should use a viewer to display the report on the device screen.

5.8 Report Delivery

The user can register the email id or digilocker account with AA. When a consolidated report is generated, the AA app can email the report or push it to digilocker account as per User's choice.

5.9 Profiles

In this flow the user can edit or update various profile information

1) Email
2) Digilocker account details to forward the consolidated reports

These information need to be stored in a secure way. Any other profile information required can be added or modified in this flow.

5.10 Consent Management

This flow provides user with a dashboard either through the web portal or the mobile app, where she is able to view list of all consents either initiated by FIU or by the user herself. For each consent she should be able to view the consent artefacts, current status of consents, data delivery status,

etc. It provides an analytical view of all the consents. In order to approve, revoke, pause any consent artefact, the user needs to additionally authenticate herself using OTP based mechanism.

5.11 Analytics

The user is provided with a transaction dashboard showing details about the transactions like,

1) Completed & Pending transactions for user-initiated flows
2) Completed & Pending transactions with FIUs for FIU initiated flows
3) Consent statistics

5.12 Billing & Payment

The user is provided with a payment dashboard showing details about the payments,

1) Itemized monthly bills for all chargeable transactions.
2) Paid or Unpaid bill details

5.13 Notification

User receives notification on various events from FIU, AA, FIP through SMS, in-app notifications. Examples of such notifications are,

- Consent requests by FIU
- Data requested by FIU
- Data received by FIU
- Data denied to FIU
- Data available for User

5.14 Right to be Forgotten

Users has the “Right to be Forgotten”. When user selects it, the following can be done

- All outstanding data requests for the user’s financial information have to be stopped
- All consent requests of the users should be revoked
- Any other data or metadata pertaining to the user in the system has to be removed.
- The user should be de-registered from the platform.

Implementation of the above should be as per the applicable data privacy and security laws and regulations.

6. FIU Application & Flows

FIU can access the Account Aggregator services programmatically through AA APIs and also through AA Web portal.

6.1 On Boarding & Signup

FIUs can sign up in the AA web portal by creating its unique login id and password. It also needs to provide the necessary information as required by AA. AA may verify this information. The

verification process can include offline steps. Once verified the FIU account becomes active on the AA platform. FIU will also have a unique FIU Id provided by NADL.

6.2 Login

FIU can login to the portal using its login id and password which was setup during the registration process. For 2 factor authentication OTP can be used to any registered mobile. Password restriction policy should adhere to the password security policy given in the NESL information security policy document, which will be shared along with the service agreement with the successful bidder.

6.3 Access Key Management

FIU should be provided with a dashboard where it can generate and manage its Secret Access Key which is required by FIU to digitally sign the API requests it makes to AA for consent and data.

6.4 Consent Creation Request

6.4.1 User Initiated through FIU Application

User can login to FIU application or website and provides the required FIPs and account details as needed by FIU. User also provides the VUA and/or mobile number. The user should also provide a time duration for which it wants to store the user's data. FIU application calls AA APIs to send the user details and the consent request with all FIP and account details. User gets a notification about the consent request upon which she can login to AA app. If the accounts specified in the consent request are not linked then the User needs to link them with AA. Once linked, the user can approve the request, digitally signs the consent artefacts. AA stores these artefacts. This flow requires FIU application to use AA APIs.

6.4.2 FIU initiated through FIU Application

FIU can create a request for consent in which it provides the details of what type of financial information are needed and VUA and/or mobile number of the user. For example, it can ask for specific Asset Types, Account Types, Transaction Types information. FIU should also provide a time duration for which it wants to store the user's data. FIU application calls AA APIs to send consent request. User gets a notification about the consent request upon which she can login through AA app or web portal. Once the user logs in she can fill in the details from the available list of FIPs and accounts which have been already linked. The User can then approve the request for consent, digitally signs the consent artefacts. AA stores these artefacts. This flow requires FIU application to use AA APIs. Adequate controls , alerts, trails etc shall be provided to User.

6.4.3 FIU initiated through AA portal

AA would be providing a web portal for FIU. FIU can login to that portal and create a request for consent in which it provides the details of what type of financial information are needed and VUA and/or mobile number of the user. For example, it can ask for specific Asset Types, Account Types, Transaction Types information. FIU should also provide a time duration for which it wants to store the user's data. The application calls AA APIs to send consent request.

User gets a notification about the consent request upon which she can login through AA app or web portal. Once the user logs in she can fill in the details from the available list of FIPs and accounts which have been already linked. The User can then approve the request for consent, digitally signs the consent artefacts. AA stores these artefacts.

In all of the above, the User can also reject the consent request from the AA application if she doesn't want to provide all details requested by the FIU. Consent Request also has an expiry time. If user doesn't take any action on the consent request, it expires. FIU needs to create a new request once the previous request has expired.

6.5 Data Retrieval

FIU should use the AA open APIs to build both these flows. Please refer to the ReBIT technical standard for details of the APIs.

6.5.1 One-time

The consent has been given by user for only one time sharing. For one-time data share the FIU initiates the request. AA requests the corresponding FIP for the data as specified in the consent artefact. Once FIP returns the data, AA notifies the FIU. FIU calls AA API to fetch the data. Since AA will store this information only up to certain period, FIU should fetch the data within that period. Beyond this period, FIU needs to initiate this request again, provided the consent artefact is still valid.

6.5.2 Periodic

The consent has been given by user for periodic data sharing. The consent artefact should also specify the start time, frequency and total period for which data needs to be shared. FIU initiates this data share using an API call. After that AA manages the data share for the entire period. At every interval, AA triggers the data request by initiating a call to FIP. Once FIP returns the data, AA notifies the FIU. FIU calls AA API to fetch the data. Since AA will store this information up to 72 hours, FIU should fetch the data within that period. Beyond this period, User needs to initiate this request again, using the same consent artefact but providing the period for the data. The consent artefact needs to be valid at the time of this request.

6.6 Consent Management

In the FIU web portal provided by AA, FIU should be able to manage the lifecycle of all consents. It should be able to create a consent request to a user, view all consents which are requested to the users, consent artefacts, current status of consents, data delivery status, etc.

6.7 Profile Management

In this flow FIU should be able to manage its profile details like name, contact address, emails etc.

6.8 Analytics

The FIU is provided with a analytics dashboard showing details about the transactions made,

1) Completed & Pending transactions for FIU-initiated flow
2) Consent statistics

6.9 Billing & Payment

The FIU is provided with a payment dashboard showing details about the payments,

1. Itemized monthly bills for all chargeable transactions.
2. Paid or Unpaid bill details

6.10 Notification

FIU receives notification on various events from AA through API call-backs. Examples of such notifications are,

- Consent requests approved by User
- Consent requests rejected by User
- Data available for FIU

7. FIP Flows

The following are the primary flows for FIP. FIPs need to implement and expose APIs for the following use cases.

7.1 Account Discovery & Linkage

During the discovery flow, FIP should be able to list all accounts held by an user, using the registered mobile number, name, address, customer relationship number or a combination of them. In the Linkage flow, when a user links one or more of his/her FIP accounts with AA, the user's identify must be verified by FIP. Once FIP verifies the user, the account can be linked to the user's profile in AA.

7.2 User Authentication

In the Linkage flow, when a user links one or more of the FIP accounts with AA, the users identify must be verified by FIP. The FIP can use an OTP based mechanism to user's registered mobile to authenticate the user.

7.3 Consent Verification and Management

When a user gives consent to data requests, AA forwards the details of the consent to FIPs. FIP should be able to verify if the consent artefact is valid. It should store the consent artefact for later use.

7.4 Data Request

For all active consent artefacts, AA sends a request for data to FIP. FIP must accept and validate the request against the consent artefact and return the required data in real time. The returned data should be in machine readable XML format encrypted. The response should be digitally signed.

8. Account Aggregator Admin Flow

AA should provide a web portal for AA admin. There would be three primary sections,

8.1 FIU Section

This dashboard should list details of all FIUs which have registered with the AA platform. The following are few information which will be available in the dashboard. More information can be added as required.

1) FIU profile details

8.2 FIP Section

This dashboard should list details of all FIPs with which AA has partnered with to retrieve user's financial information. These FIPs can be of various types and under various regulatory bodies. The following are few information which will be available in the dashboard. More information can be added as required.

1) FIP Profile details

8.3 User Section

Following are some of the information which will be available in this dashboard. More information can be added as required.

1) Total number of registered users
2) Roles and Permissions
    a) Different Roles configured in the system
    b) Permissions associated with the Roles
    c) Details of users who have Admin Role and Business User Role

9. Business Reporting

AA Management team will require visibility into the business functions of AA. Business team should therefore define a set of KPIs and charts which are relevant for AA business. One or more dashboards should be created to show these KPIs and trend charts. These metrics should be as accurate as possible, collected and processed in near-real time. Some of the business KPIs can be,

- FIU metrics
    - Number of FIU signups
    - Number of active FIUs
    - List of FIUs along with status
    - Total number of successful, failed data requests from FIU.
    - Total number of consent requests from FIU
- FIP metrics
    - List of FIPs
    - Total number of successful, failed data requests served by an FIP
- User metrics

- Number of User signups
- Number of active Users
- Number of users whose KYC are pending
- Total number of successful, failed data requests from Users.
- Total number of consents created and approved by Users
- Geographic and Demographic distribution of various metrics.

Dashboard should also provide trend charts for individual metrics and drill down across various dimensions like FIU, FIP, etc. Business Reporting should be available only to selected users.

10. Billing and Invoice

Based on the pricing model, the system should be able to calculate and generate detailed invoices for customers of AA. The invoice should be accurate and timely. It should be visible to the customer when she logs in to the application. Additionally, the system can automatically email the invoices to appropriate email address of the customer. Billing system should be capable of supporting both prepaid and post-paid models.

11. Design Guidelines

11.1 Open API

All interactions between AA, FIP, FIU and User (App) should be through APIs. APIs provide programmatic interfaces for sharing and accessing information. AA as a service provider should make public a set of well-defined API contracts which can be used by FIU and FIPs to communicate with AA. The set of APIs should be very minimal, properly defined and very well documented.

11.2 Open source frameworks

For implementation of AA platform, the recommendation is to use open source frameworks along with support license (e.g. Linux Redhat). Most Open source frameworks has a large developer community and therefore fast innovations which can be leveraged in the platform. The licenses of the open source frameworks need to be verified. Apache and MIT license and Mozilla Public License software should be used. NADL should be consulted before selecting the open source framework.

11.3 Micro services Architecture

The system should be built following the design principles of micro service architecture. Each such loosely coupled service should be independently developed, tested and deployed. They should define a clear and minimal set of contract APIs to communicate with other services. The services should be stateless allowing it to scale easily. State should be stored in separate storages. Each service should also manage the storage and lifecycle of the data it owns.

Micro services should be designed with API-first approach, i.e. they should define clean API contracts which can be used by other micro services to communicate. APIs should be versioned appropriately so that changes can be incorporated with newer versions.

11.4 Minimalist and Evolutionary Architecture

The design should be simple and minimalistic. It should not present adoption barriers for the ecosystem. The design of the systems should be evolutionarily - their capabilities should be built incrementally while allowing for rapid adoption.

11.5 Horizontal Scalability

The AA service should be built to support horizontal scalability. As API request volumes increase, it should be possible to simply add additional servers to handle the load. This can be achieved easily if the services are developed as stateless services. Data store services which can be horizontally scalable should be preferred.

11.6 Security & Privacy

Security of information and privacy of customer's data are extremely important due to the nature of the application.

11.6.1 Data Security

User's financial information can never be decrypted in AA. Strong cryptographic techniques should be used for this, so that it is never possible for AA systems to decrypt and read the data. Data-at-rest should be encrypted and stored in Vaults. The keys and certificates used for encryptions and decryptions should be managed separately within HSM (Hardware Security Module)

11.6.2 API Security

These are the some of the basic security practices that must be followed for securing APIs:

1. IPSec
2. IP Whitelisting
3. Digitally Signed API Requests and Responses
4. Validations
    a. Input Validation
    b. URL Validation
    c. Validate incoming content-types
    d. Validate response types
5. Output Encoding
    a. Security Headers
    b. JSON Encoding
    c. XML encoding
6. HTTPS-only

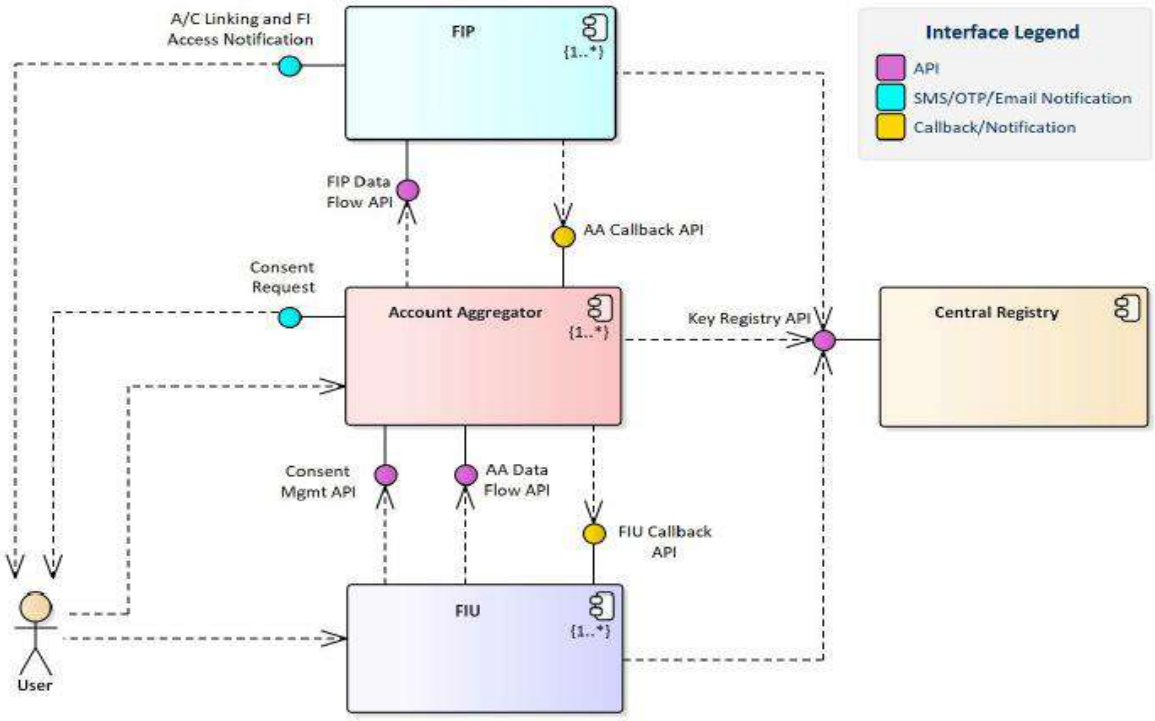
AA platform should also comply to the NESL information security policy, which will be shared along with the service agreement with the successful bidder. All APIs specified by AA needs to follow the security guidelines as mentioned above. All API requests and responses need to be digitally signed by the requester for non-repudiation.

A good guideline for managing security in REST APIs is provided by the OWASP (Open Web Application Security Project) community:
https://www.owasp.org/index.php/REST_Security_Cheat_Sheet

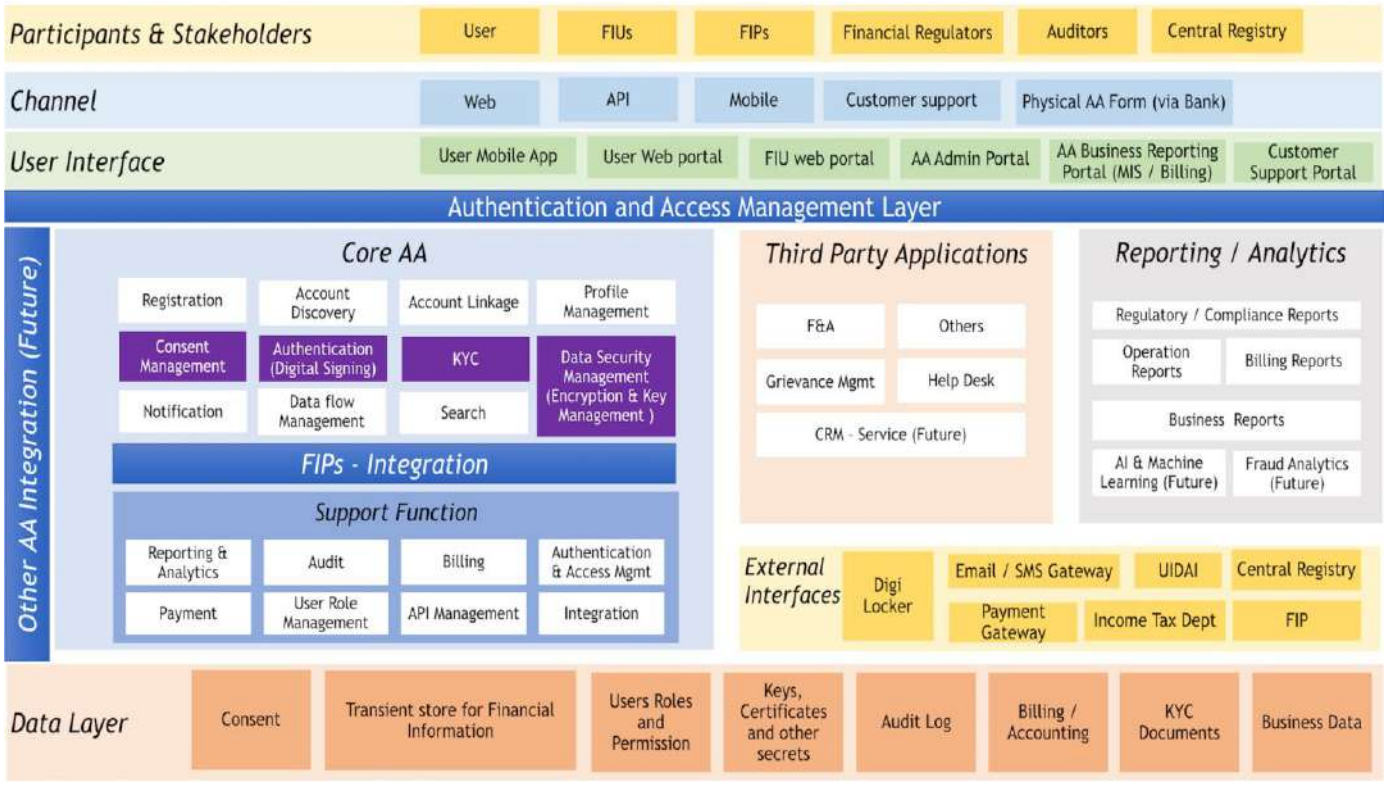
12. Architecture of Account Aggregator Server

12.1 Architecture

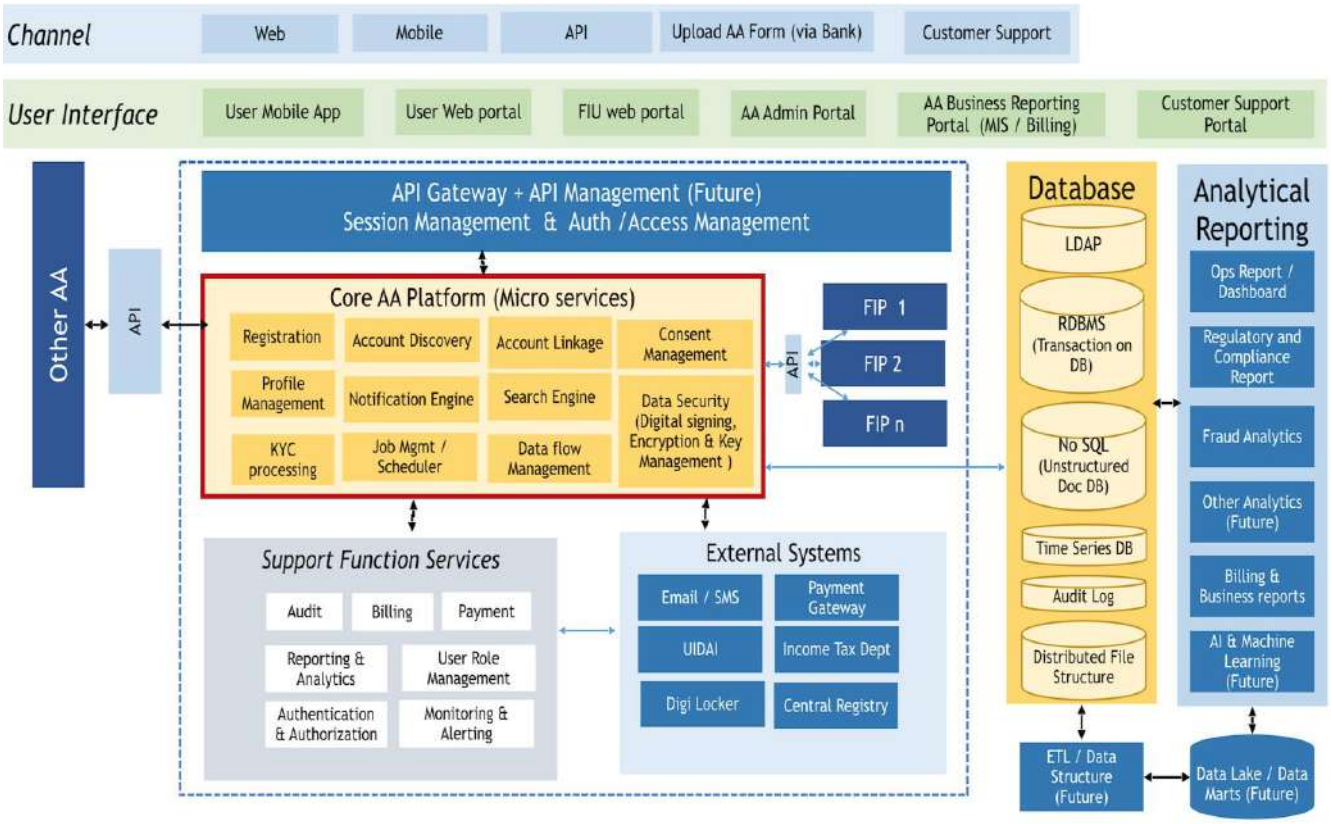
12.1.1 Data Flow Architecture



12.1.2 Functional Architecture



12.1.3 Technical Architecture



12.2 Components and Services

The following components and services should define the Account Aggregator system.

12.2.1 User Registration & Account creation

To use the services provided by AA, user needs to register on the AA platform and create an account. This service manages the user registration and account creation on the AA platform. The user will create a VUA and password to login to the platform. This service will also initiate the KYC process of the customer.

12.2.2 FIU Registration & Account creation

Any FIU who wants to receive user's data needs to get consent from the user. For this FIU needs to be registered to AA platform. The registration process will involve FIU providing required information for verifications. This may also require some offline verification steps. Once verification is complete FIU's account will be active and FIU will have a unique FIU Id.

12.2.3 Financial Information Account Discovery, Linking and Unlinking

Before any data request can be done for a FIP account, the user has to link her FIP accounts with AA. This is accomplished using Discovery and Linking flow. In the Discovery flow, user's information like Aadhaar/PAN/mobile/Name/Customer Number are used to discover all the accounts the user holds with the FIP. The user can then select a set of accounts out of that list to link with her AA account. During the Linking flow the user's identity needs to be verified by FIP to ensure that the correct user is linking with the correct account. This can be done using OTP to her mobile number. At any point later, the user can unlink an already linked account. This should again be verified by AA using OTP.

12.2.4 User Profile Management

This service manages the lifecycle of User, their profiles on the AA platform. It also manages all the financial information accounts which the user has linked with AA. Given the sensitivity of financial information, AA must put in place appropriate user management and user protection mechanisms. Users must have provisions to lock and unlock their AA accounts for data access.

12.2.5 FIU Profile Management

This service manages the lifecycle of FIU, their profiles on the AA platform. This should provide mechanism for adding, updating, deleting, storing and retrieving the FIU profile details.

12.2.6 FIU Key Management

This service manages the lifecycle of the Access keys for the FIU. It should provide a mechanism to securely generate the keys which FIU needs to use while calling the APIs. There should be a mechanism to rotate the keys.

12.2.7 Identity & Authentication

12.2.7.1 User Identity & Authentication

This service manages the authentication and validation of the identity of the user at signup, sign in, account linking and unlinking time. At the time of login to the AA platform from the app the user is authenticated using a 2-factor authentication with VUA, Password and OTP. Token based authentication protocol OAuth2 should be used, with JSON Web Token Standard. Also, each API request and response payloads should be digitally signed by the requester for non-repudiation.

12.2.7.2 FIU Identity & Authentication

12.2.7.2.1 Web Portal

At the time of login to the AA platform from the web portal the FIU is authenticated using a 2-factor authentication with VUA, Password and OTP. Token based authentication protocol OAuth2 should be used, with JSON Web Token Standard.

12.2.7.2.2 API

While programmatically calling AA APIs for data and consent, FIU should sign each API call using the secret access key. This is used to validate the FIU.

12.2.8 Authorization

The service manages the roles and permissions for all types of users on the platform. Different types of user roles are

1) Users
2) Administrator
3) Business User

Each role has specific set of permissions. A user can have a combination of one or more roles. A user having a “Business” role will be able to see all Business Reports. By default, all users registering with the AA platform will have “Users” role. A user with “Administrator” role will be seeded to the system. The Administrator then can assign any other role as required to any user.

12.2.9 Consent Management

The lifecycle of consent artefacts is managed by Consent Management Service. When user approves and digitally sign the artefact, it is stored by AA. Digitally signed Consent Artefacts are also shared with FIPs for subsequent verification. If FIUs have requested for the data share, then the corresponding approved and verified artefacts, will also be shared with FIUs. When a consent is revoked or it expires; data should no more be shared based on the same consent. Appropriate notifications should be sent to FIU, User, FIPs about this. All digitally signed consent artefacts, should be also archived so that they can be accessed at later stage if needed

12.2.10 Data Flow Management

The data flow service manages the entire financial information flow between FIP ->AA and AA -> User and AA->FIU. It is an asynchronous flow. When FIU or User sends a request, it validates and forwards the requests to FIP. Sometime later, FIP notifies AA that the data is available. AA pulls the data and stores it in its transient store and notifies FIU or User about the availability of data. User can also view the status of the requests in the app. FIU or User then pulls data from AA. The data flowing through AA is encrypted and transient in nature. AA will need to provide a strict time bound by which the data needs to be fetched by user or FIU. After that AA has to ensure that the data is deleted from its transient store. The data will be stored in the transient store up to a maximum of 72 hours. The data flow will generally be of two types:

- One time
- Periodic

For one-time data access, after the consent is created and approved, FIU initiates the request for data. For periodic data access, after the consent is created and approved, AA manages and initiates the data requests to FIPs for the entire period.

12.2.11 Data Security & Privacy Management

12.2.11.1 Data-in-transit

The financial information that is shared by FIP to User or FIU, needs to be fully encrypted as it passes through AA. None of the AA components and services should be able to decrypt and read any of the financial information. This data should be encrypted using cryptographic algorithms by FIP and can only be decrypted by the original requester, User or FIU. The details of the algorithm used to encrypt the data from FIP to FIU and User is described here.

Data shared as part of the data flow will be secured using an encryption mechanism that ensures perfect forward secrecy. This means that even if any of the key materials stored at FIPs or FIUs (either long-term private keys or session keys) are compromised at a given point in time, data that was exchanged in the past (i.e. before that point in time) would not be possible to decipher. This is a strong guarantee of secrecy which is necessary to ensure for financial data.

We describe the mechanism here; corresponding APIs are in the appendix. The mechanism uses Diffie-Hellman Key Exchange (DHE). DHE is used in many Internet protocols (like SSH and TLS) for establishing shared secret keys between remote parties.

12.2.11.1.1 Encryption for FIU initiated Flow

1) When making the request for data, FIU picks a set of Diffie-Hellman (DH) parameters, generates a DH key pair (dhsk(U), dhpk(U)) (which is a short-term public-private key pair) and generates a 32-byte random value, rand(U). It sends these values to AA, along with the data request via a digitally-signed API call.
2) AA ensures that the data request is in keeping with the terms of the artefact and, if so, it forwards the request to the FIP, again via a digitally-signed API call.
3) FIP checks that the consent artefact is valid (as above), that the data being requested is in keeping with the terms of the artefact and if so, it generates a fresh DH public-private key pair in the same group as specified by the FIU ((dhsk(P), dhpk(P)) and also a 32-byte random value rand(P). Using dhpk(U) and dhsk(P), it computes a DH shared key dhk(U,P)

and using (dhk(U,P), rand(U), rand(P)) as key material, it computes a 256-bit session key sk(U,P) which is used to encrypt the data sent from FIP to FIU. To ensure integrity of the encrypted data, FIP also signs the encrypted data using its long-term private key before sending it to AA.

The DHE mechanism ensures that the shared key dh(U,P) can also be computed at the other end by the FIU using the values dhpk(P) (FIP's DH public key) and dhsk(U) (FIU's DH private key). For this reason, the FIP must accompany the encrypted data with dhpk(P) and rand(P) when sending it. All values must be digitally signed using FIP's long-term private key so that the FIU can verify the validity of the same.
At the end of the data flow, both FIU and FIP must delete all short-term key material that was generated in the process. This includes all DH key pairs, random nonces and the session key. This step is necessary for ensuring forward secrecy.

12.2.11.1.2 Encryption for User Initiated Flow

The exact same flow would work except that all actions performed by the FIU would instead be performed by the AA app on the User's phone. The following guidelines are to be followed in implementing the AA app, if the AA decides to offer the User initiated flow functionality:

1) DHE key pairs must be generated locally on the User's phone and the private keys from the DHE key pairs must never be communicated to the AA server.
2) The decryption of the User's data must also take place on the User's phone and the decrypted data (User's financial information) must never be communicated to the AA server.
3) As stated above, the private keys must be deleted from the User's phone at the end of the data flow.

The AA app should be implemented in a manner such that it can be easily verified (by an auditor) that the above three guidelines are followed.

12.2.11.1.3 Encryption for Periodic Data Access
For encrypting data in the case of periodic data access, FIUs and FIPs set up shared keys for a period of usage via a single exchange. A period is a series of multiple data accesses which is defined as follows: if the frequency of data access is WEEKLY or less frequent, the period should be set as 3 months; otherwise, the period should be set as frequency, multiplied by 12.

1. At the beginning of each period, FIU generates n key pairs (dhsk(U)[1], dhpk(U)[1]), ... , (dhsk(U)[n], dhpk(U)[n]) where n is the number of data accesses in that period. It generates a random nonce rand(U) and sends the selected DH parameters, the value n, the n DH public keys dhpk(U)[1..n], the start-date and end-date of the period, the value rand(U) and the consent artefact to AA via a digitally-signed API call.
2. AA forwards the same, after validation, to FIP via a digitally-signed API call
3. FIP stores the DH key pairs and rand(U) (and other fields).

For the “i”th instance of data access in a period (i ranging from 1 to n), FIP generates a fresh DH key pair (dhsk(P)[i], dhpk(P)[i]) and a random nonce rand(P)[i] and computes a DH shared key dhk(U,P)[i] and a session key sk(U,P)[i] using these values and the “i”th public key shared by FIU (i.e. dhpk(U)[i]) earlier. The data encryption and transmission then happens as usual, after which the DH keys dhsk(U)[i], dhsk(P)[i], and the session key sk(U,P) are deleted by the respective entities.

12.2.11.1.4 Choice of Diffie-Hellman Parameter

DHE will be performed over Elliptic Curve Cryptography (ECC) groups. We recommend the use of Curve25519, which is used in DHE implementations in a lot of protocols like SSH and WhatsApp.

12.2.11.2 Data-at-Rest

Both AA and User should digitally sign and encrypt the consent artefacts and store it. If any Aadhaar/PAN/Account (or any other confidential) information is used as part of user profile, it should be encrypted and stored in separate data vault. These should also be tokenized and the reference keys should be used elsewhere in all other services where this information are needed. The encryption key should be in HSM, which should be used for all encryption and decryption. This data vault should be in a secured network zone and access to it should be controlled only through a single micro service with proper access control mechanism. It should not be computationally feasible to recover this confidential information from the reference keys.

12.2.12 Key Secrets Management

All certificates, keys, passwords, etc. needs to be managed by keeping it in Vault. Any user specific profile information needs to be stored also in Vault, encrypted. The encryption key should be in HSM which is responsible for encryption and decryption of this data.

12.2.13 Notification Management

The service manages all types of notifications in linking, consent and data flow, to and from AA. It provides call back end points to receive notifications from FIP. Also it can trigger notifications as required to User and FIU using appropriate channels like SMS, Email, URL or In-app Notifications. Each notification will have its own payloads based on the standard defined in the Reference section (15). The payload size and information content may vary depending on the notification channel used. Please refer to ReBIT technology for details of all types of notifications.

12.2.14 Audit Management

All API requests should be logged. All events in the consent flow, data flow and notification flows must be logged. The audit logs should clearly indicate, at-least the followings

- Metadata from all consent approval, pause, revocation API requests
- Metadata from all data API requests.

Audit Management service manages the audit data by providing a tamper-resistant way to log these events. It persists the audit data for certain period of time and provides a mechanism to retrieve when necessary. A distributed ledger, e.g. blockchain, can be used to store these transaction records immutably for audit purpose.

12.2.15 KYC Processing

When a user registers with AA, KYC needs be done. This service handles the KYC in two ways.

12.2.15.1 Online e-KYC

The service calls the Aaadhar based e-KYC APIs in real time.

12.2.15.2 Offline KYC

The service uploads and stores the required documents submitted by the user. It also triggers a business process workflow which manually validates the required documents and performs the KYC of the customers. Once KYC is done, user's account is activated. The KYC details are stored securely.

12.2.16 Reporting & Analytics

ETL processes should be designed to consume the transaction metadata and compute metrics in real time for various reports shown to AA or FIU or Users. These aggregated metrics should be stored and shown in dashboards.

12.2.17 Billing and Invoice Generation

This service should be able to process all the transaction metadata and based on the business model, calculate the detailed billing information for the customers. It should also generate invoice for each customer. The service can use existing billing solution (e.g Tally) for this purpose.

12.2.18 Payment

This service is responsible for providing a payment interface to both Users and FIUs. It integrates with payment gateway to process the payments from the customers.

12.2.19 Measurement, Monitoring & Alerting

All components and services within AA should be monitored in real time (< 1sec). System level metrics and Application level metrics as appropriate, should be collected which can be plotted on monitoring dashboard to look at the historical trends. AA should also monitor the availability of services from FIP, using status APIs hosted by FIPs. An Alerting system needs to be built to alert on various operational and application metrics when they go beyond configurable thresholds.

12.2.20 Integration with 3rd party services

The platform requires integration with several services such as

- Email Service

- SMS Gateway
- Payment Gateway
- Aadhaar based eKYC
- Aadhaar based eSign
- PAN validation service from Income Tax department of Govt. of India

Tested Integration code for the above services except eKYC are available with NESL and can be reused by the vendor.

13. Central Registry

Central Registry provides details of the public endpoints, certificates issued by registered Certificate Issuing Authority for FIPs and AAs at a central place. FIPs and AA can use the certificates listed there to verify the digital signatures in the API responses for non-repudiation. The CDAC DSC validation API can be used to verify the certificate and the chain root. Central Registry may also provide the profile of the participant, i.e. what role that entity plays, i.e. AA, FIP etc. Since, Central Registry is still an evolving concept from ReBIT, the above details are subject to some modification.

14. AA to FIP Connection

All FIPs publish their Https endpoints which are listed in the central registry. To connect to an FIP, AA needs to discover the URL of FIP from central registry. When calling the FIP APIs, AA should provide the required credentials so that AA can be authenticated by the FIP. All request and response payloads should also be digitally signed.

15. Financial Information Types

Schemas for some of the financial information types are shown below. These are subject to changes and addition of new types.

Deposits

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:aa="http://api.rebit.org.in/FISchema/deposit" elementFormDefault="qualified"
 targetNamespace="http://api.rebit.org.in/FISchema/deposit"
    xmlns:vc="http://www.w3.org/2007/XMLSchema-versioning" vc:minVersion="1.1">
    <!-- -->
    <xs:simpleType name="SummaryType">
            <xs:restriction base="xs:string">
                    <xs:enumeration value="deposit"/>
            </xs:restriction>
    </xs:simpleType>
```

```
    <xs:simpleType name="AccountType">
        <xs:restriction base="xs:string">
            <xs:enumeration value="savings"/>
            <xs:enumeration value="current"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="HoldersType">
        <xs:restriction base="xs:string">
            <xs:enumeration value="single"/>
            <xs:enumeration value="joint"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="Date">
        <xs:restriction base="xs:date"></xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="SummaryIfsc">
        <xs:restriction base="xs:string">
            <xs:pattern     value="[a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][0][0-9][0-9][0-9][0-9][0-
9][0-9]"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="SummaryFacility">
        <xs:restriction base="xs:string">
            <xs:enumeration value="OD"/>
            <xs:enumeration value="CC"/>
            <xs:enumeration value="DLOD"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="HolderAadhar">
        <xs:restriction base="xs:string">
            <xs:pattern value="[0-9]{12}"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="HolderEmail">
        <xs:restriction base="xs:string">
            <xs:pattern value="[^@]+@[^\.]+\..+"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="HolderPan">
        <xs:restriction base="xs:string">
            <xs:pattern     value="[a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][0-9][0-9][0-9][0-
9][a-zA-Z]"/>
        </xs:restriction>
    </xs:simpleType>
    <!--  -->
    <xs:element name="Account">
```

```
            <xs:complexType>
                  <xs:annotation>
                        <xs:documentation></xs:documentation>
                  </xs:annotation>
                  <xs:sequence>
                        <xs:element ref="aa:Summary" minOccurs="0"/>
                        <xs:element ref="aa:Transactions" minOccurs="0"/>
                  </xs:sequence>
                  <xs:attribute name="id" use="required" type="xs:NCName"/>
                  <xs:attribute name="type" use="required" type="aa:SummaryType"/>
            </xs:complexType>
      </xs:element>
      <xs:element name="Summary">
            <xs:complexType>
                  <xs:sequence>
                        <xs:element ref="aa:Holders"/>
                  </xs:sequence>
                  <xs:attribute name="branch" use="required"/>
                  <xs:attribute name="currentBalance" use="required"/>
                  <xs:attribute name="currentODLimit" use="required"/>
                  <xs:attribute name="facility" use="required" type="aa:SummaryFacility"/>
                  <xs:attribute name="ifscCode" use="required" type="aa:SummaryIfsc"/>
                  <xs:attribute name="micrCode" use="required"/>
                  <xs:attribute name="openingDate" use="required" type="aa:Date"/>
                  <xs:attribute name="type" use="required" type="aa:AccountType"/>
            </xs:complexType>
      </xs:element>
      <xs:element name="Holders">
            <xs:complexType>
                  <xs:sequence>
                        <xs:element ref="aa:Holder"/>
                  </xs:sequence>
                  <xs:attribute name="type" use="required" type="aa:HoldersType"/>
            </xs:complexType>
      </xs:element>
      <xs:element name="Holder">
            <xs:complexType>
                  <xs:attribute name="aadhaar" use="required" type="aa:HolderAadhar"/>
                  <xs:attribute name="address" use="required"/>
                  <xs:attribute name="email" use="required" type="aa:HolderEmail"/>
                  <xs:attribute name="landline" use="required"/>
                  <xs:attribute name="mobile" use="required"/>
                  <xs:attribute name="name" use="required"/>
                  <xs:attribute name="order" use="required"/>
                  <xs:attribute name="pan" use="required" type="aa:HolderPan"/>
            </xs:complexType>
```

```
    </xs:element>
    <xs:element name="Transactions">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="aa:Transaction"/>
            </xs:sequence>
            <xs:attribute name="endDate" use="required" type="aa:Date"/>
            <xs:attribute name="startDate" use="required" type="aa:Date"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="Transaction">
        <xs:complexType>
            <xs:attribute name="amount" use="required" type="xs:float"/>
            <xs:attribute name="balance" use="required" type="xs:float"/>
            <xs:attribute name="date" use="required" type="aa:Date"/>
            <xs:attribute name="id" use="required"/>
            <xs:attribute name="narration" use="required"/>
            <xs:attribute name="reference" use="required"/>
        </xs:complexType>
    </xs:element>
</xs:schema>
```

Term Deposit

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:aa="http://api.rebit.org.in/FISchema/term_deposit" elementFormDefault="qualified"
 targetNamespace="http://api.rebit.org.in/FISchema/term_deposit"
    xmlns:vc="http://www.w3.org/2007/XMLSchema-versioning" vc:minVersion="1.1">
    <xs:simpleType name="AccountType">
        <xs:restriction base="xs:string">
            <xs:enumeration value="savings"/>
            <xs:enumeration value="current"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="HolderAadhar">
        <xs:restriction base="xs:string">
            <xs:pattern value="[0-9]{12}"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="HolderEmail">
        <xs:restriction base="xs:string">
```

```
                    <xs:pattern value="[^@]+@[^\.]+\..+"/>
                </xs:restriction>
        </xs:simpleType>
        <xs:simpleType name="HolderPan">
                <xs:restriction base="xs:string">
                    <xs:pattern    value="[a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][0-9][0-9][0-9][0-
9][a-zA-Z]"/>
                </xs:restriction>
        </xs:simpleType>
        <!--  -->
        <xs:element name="Account">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element ref="aa:Summary"/>
                        <xs:element ref="aa:Transactions"/>
                    </xs:sequence>
                    <xs:attribute name="id" use="required" type="xs:NCName"/>
                    <xs:attribute name="type" use="required" type="xs:NCName"/>
                </xs:complexType>
        </xs:element>
        <xs:element name="Summary">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element ref="aa:Holders"/>
                    </xs:sequence>
                    <xs:attribute name="accountType" use="required" type="aa:AccountType"/>
                    <xs:attribute name="compoundingFrequency" use="required"/>
                    <xs:attribute name="currentValue" use="required"/>
                    <xs:attribute name="description" use="required"/>
                    <xs:attribute name="interestComputation" use="required"/>
                    <xs:attribute name="interestOnMaturity" use="required"/>
                    <xs:attribute name="interestPayout" use="required"/>
                    <xs:attribute name="interestPeriodicPayoutAmount" use="required"/>
                    <xs:attribute name="interestRate" use="required"/>
                    <xs:attribute name="maturityDate" use="required"/>
                    <xs:attribute name="openingDate" use="required"/>
                    <xs:attribute name="principalAmount" use="required"/>
                    <xs:attribute name="recurringAmount" use="required"/>
                    <xs:attribute name="recurringDepositDay" use="required"/>
                    <xs:attribute name="tenureDays" use="required"/>
                    <xs:attribute name="tenureMonths" use="required"/>
                    <xs:attribute name="tenureYears" use="required"/>
                </xs:complexType>
        </xs:element>
        <xs:element name="Holders">
                <xs:complexType>
```

```
                    <xs:sequence>
                        <xs:element ref="aa:Holder"/>
                    </xs:sequence>
                    <xs:attribute name="type" use="required"/>
                </xs:complexType>
            </xs:element>
            <xs:element name="Holder">
                <xs:complexType>
                    <xs:attribute name="aadhaar" use="required" type="aa:HolderAadhar"/>
                    <xs:attribute name="address" use="required"/>
                    <xs:attribute name="email" use="required" type="aa:HolderEmail"/>
                    <xs:attribute name="landline" use="required"/>
                    <xs:attribute name="mobile" use="required" type="xs:string"/>
                    <xs:attribute name="name" use="required" type="xs:string"/>
                    <xs:attribute name="pan" use="required" type="aa:HolderPan"/>
                    <xs:attribute name="rank" use="required"/>
                </xs:complexType>
            </xs:element>
            <xs:element name="Transactions">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element ref="aa:Transaction"/>
                    </xs:sequence>
                    <xs:attribute name="endDate" use="required" type="xs:date"/>
                    <xs:attribute name="startDate" use="required" type="xs:date"/>
                </xs:complexType>
            </xs:element>
            <xs:element name="Transaction">
                <xs:complexType>
                    <xs:attribute name="amount" use="required" type="xs:string"/>
                    <xs:attribute name="date" use="required" type="xs:date"/>
                    <xs:attribute name="id" use="required"/>
                    <xs:attribute name="narration" use="required"/>
                    <xs:attribute name="type" use="required"/>
                    <xs:attribute name="valueDate" use="required" type="xs:date"/>
                </xs:complexType>
            </xs:element>
        </xs:schema>
```

Mutual Fund

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
```

```
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:aa="http://api.rebit.org.in/FISchema/mutual_funds"
elementFormDefault="qualified"
 targetNamespace="http://api.rebit.org.in/FISchema/mutual_funds"
    xmlns:vc="http://www.w3.org/2007/XMLSchema-versioning" vc:minVersion="1.1">
    <!--  -->
    <xs:simpleType name="HoldingEmail">
        <xs:restriction base="xs:string">
            <xs:pattern value="[^@]+@[^\.]+\..+"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="HoldingPan">
        <xs:restriction base="xs:string">
            <xs:pattern    value="[a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][0-9][0-9][0-9][0-
9][a-zA-Z]"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="HoldingType">
        <xs:restriction base="xs:string">
            <xs:enumeration value="sole"/>
            <xs:enumeration value="joint"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="HoldingNominee">
        <xs:restriction base="xs:string">
            <xs:enumeration value="notRegistered"/>
            <xs:enumeration value="name"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="TransactionType">
        <xs:restriction base="xs:string">
            <xs:enumeration value="changeOfFolio"/>
            <xs:enumeration value="purchase"/>
            <xs:enumeration value="SIPPurchase"/>
            <xs:enumeration value="switchIn"/>
            <xs:enumeration value="systematicTransferIn"/>
            <xs:enumeration value="switchOut"/>
            <xs:enumeration value="redeem"/>
            <xs:enumeration value="systematicTransferout"/>
            <xs:enumeration value="systematicWithdrawal"/>
            <xs:enumeration value="dividendReinvestment"/>
            <xs:enumeration value="dividendPayout"/>
            <xs:enumeration value="reversal"/>
            <xs:enumeration value="transferIn"/>
            <xs:enumeration value="dematIn"/>
            <xs:enumeration value="transferOut"/>
```

```
                    <xs:enumeration value="dematOut"/>
                    <xs:enumeration value="others"/>
                </xs:restriction>
        </xs:simpleType>
        <!--  -->
        <xs:element name="Account">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element ref="aa:Holdings"/>
                        <xs:element ref="aa:Transactions"/>
                    </xs:sequence>
                    <xs:attribute name="id" use="required" type="xs:string"/>
                    <xs:attribute name="type" use="required" type="xs:string"
fixed="mutualfunds"/>
                </xs:complexType>
        </xs:element>
        <xs:element name="Holdings">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element ref="aa:Holding"/>
                    </xs:sequence>
                </xs:complexType>
        </xs:element>
        <xs:element name="Holding">
                <xs:complexType>
                    <xs:attribute name="amc" use="required" type="xs:string"/>
                    <xs:attribute name="amfiCode" use="required" type="xs:string"/>
                    <xs:attribute name="email" use="required" type="aa:HoldingEmail"/>
                    <xs:attribute name="folioNo" use="required" type="xs:string"/>
                    <xs:attribute name="holdingType" use="required" type="aa:HoldingType"/>
                    <xs:attribute name="holder1Name" use="required" type="xs:string"/>
                    <xs:attribute name="holder2Name" use="required" type="xs:string"/>
                    <xs:attribute name="isin" use="required" type="xs:string"/>
                    <xs:attribute name="nav" use="required" type="xs:string"/>
                    <xs:attribute name="nominee" use="required" type="aa:HoldingNominee"/>
                    <xs:attribute name="pan" use="required" type="aa:HoldingPan"/>
                    <xs:attribute name="registrar" use="required" type="xs:string"/>
                    <xs:attribute name="scheme" use="required" type="xs:string"/>
                    <xs:attribute name="ucc" use="required" type="xs:string"/>
                    <xs:attribute name="units" use="required" type="xs:string"/>
                    <xs:attribute name="value" use="required" type="xs:string"/>
                </xs:complexType>
        </xs:element>
        <xs:element name="Transactions">
                <xs:complexType>
                    <xs:sequence>
```

```
                        <xs:element ref="aa:Transaction"/>
                    </xs:sequence>
                    <xs:attribute name="endDate" use="required" type="xs:date"/>
                    <xs:attribute name="startDate" use="required" type="xs:date"/>
                </xs:complexType>
            </xs:element>
            <xs:element name="Transaction">
                <xs:complexType>
                    <xs:attribute name="amc" use="required" type="xs:string"/>
                    <xs:attribute name="amfiCode" use="required" type="xs:string"/>
                    <xs:attribute name="amount" use="required" type="xs:string"/>
                    <xs:attribute name="cost" use="required" type="xs:string"/>
                    <xs:attribute name="date" use="required" type="xs:date"/>
                    <xs:attribute name="folioNo" use="required" type="xs:string"/>
                    <xs:attribute name="id" use="required" type="xs:string"/>
                    <xs:attribute name="isin" use="required" type="xs:string"/>
                    <xs:attribute name="narration" use="required" type="xs:string"/>
                    <xs:attribute name="nav" use="required" type="xs:string"/>
                    <xs:attribute name="scheme" use="required" type="xs:string"/>
                    <xs:attribute name="stt" use="required" type="xs:string"/>
                    <xs:attribute name="type" use="required" type="aa:TransactionType"/>
                    <xs:attribute name="ucc" use="required" type="xs:string"/>
                    <xs:attribute name="units" use="required" type="xs:string"/>
                </xs:complexType>
            </xs:element>
        </xs:schema>
```

Insurance

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:aa="http://api.rebit.org.in/FISchema/insurance_policies"
elementFormDefault="qualified"
 targetNamespace="http://api.rebit.org.in/FISchema/insurance_policies"
    xmlns:vc="http://www.w3.org/2007/XMLSchema-versioning" vc:minVersion="1.1">
    <!--  -->
    <xs:simpleType name="SummaryTypes">
        <xs:restriction base="xs:string">
            <xs:enumeration value="life"/>
            <xs:enumeration value="medical"/>
            <xs:enumeration value="vehicle"/>
```

```
                <xs:enumeration value="home"/>
                <xs:enumeration value="others"/>
            </xs:restriction>
        </xs:simpleType>
        <xs:simpleType name="SummaryPolicyType">
            <xs:restriction base="xs:string">
                <xs:enumeration value="term"/>
                <xs:enumeration value="endowment"/>
                <xs:enumeration value="money back"/>
                <xs:enumeration value="pension plan"/>
                <xs:enumeration value="whole life"/>
                <xs:enumeration value="children's plan"/>
                <xs:enumeration value="loan cover term"/>
                <xs:enumeration value="comprehensive"/>
                <xs:enumeration value="third party"/>
                <xs:enumeration value="others"/>
            </xs:restriction>
        </xs:simpleType>
        <xs:simpleType name="SummaryCoverType">
            <xs:restriction base="xs:string">
                <xs:enumeration value="term rop"/>
                <xs:enumeration value="critical illness"/>
                <xs:enumeration value="health"/>
                <xs:enumeration value="self"/>
                <xs:enumeration value="family"/>
                <xs:enumeration value="Building"/>
                <xs:enumeration value="Contents"/>
                <xs:enumeration value="others"/>
            </xs:restriction>
        </xs:simpleType>
        <xs:simpleType name="SummarypremiumFrequency">
            <xs:restriction base="xs:string">
                <xs:enumeration value="monthly"/>
                <xs:enumeration value="quarterly"/>
                <xs:enumeration value="halfYearly"/>
                <xs:enumeration value="yearly"/>
                <xs:enumeration value="oneTime"/>
            </xs:restriction>
        </xs:simpleType>
        <xs:simpleType name="HoldersType">
            <xs:restriction base="xs:string">
                <xs:enumeration value="single"/>
                <xs:enumeration value="joint"/>
            </xs:restriction>
        </xs:simpleType>
        <xs:simpleType name="HoldersRank">
```

```
            <xs:restriction base="xs:string">
                <xs:enumeration value="1"/>
                <xs:enumeration value="2"/>
            </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="HolderAadhar">
            <xs:restriction base="xs:string">
                <xs:pattern value="[0-9]{12}"/>
            </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="HolderEmail">
            <xs:restriction base="xs:string">
                <xs:pattern value="[^@]+@[^\.]+\..+"/>
            </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="HolderPan">
            <xs:restriction base="xs:string">
                <xs:pattern    value="[a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][0-9][0-9][0-9][0-
9][a-zA-Z]"/>
            </xs:restriction>
    </xs:simpleType>
    <!--  -->
    <xs:element name="Account">
            <xs:complexType>
                <xs:sequence>
                    <xs:element ref="aa:Summary"/>
                    <xs:element ref="aa:Transactions"/>
                </xs:sequence>
                <xs:attribute name="id" use="required" type="xs:string"/>
                <xs:attribute        name="type"        use="required"        type="xs:string"
fixed="insurance"/>
            </xs:complexType>
    </xs:element>
    <xs:element name="Summary">
            <xs:complexType>
                <xs:sequence>
                    <xs:element ref="aa:Holders"/>
                    <xs:element ref="aa:Riders"/>
                    <xs:element ref="aa:MoneyBacks"/>
                    <xs:element ref="aa:Covers"/>
                </xs:sequence>
                <xs:attribute name="coverAmount" use="required" type="xs:string"/>
                <xs:attribute                name="coverType"                use="required"
type="aa:SummaryCoverType"/>
                <xs:attribute name="maturityBenefit" use="required" type="xs:string"/>
                <xs:attribute name="nextPremiumDueDate"  type="xs:date"/>
```

```
                <xs:attribute name="policyDescription"  type="xs:string"/>
                <xs:attribute name="policyExpiryDate" type="xs:date"/>
                <xs:attribute name="policyName" use="required" type="xs:string"/>
                <xs:attribute name="policyStartDate" use="required" type="xs:date"/>
                <xs:attribute name="policyType" use="required"
type="aa:SummaryPolicyType"/>
                <xs:attribute name="premiumAmount" use="required" type="xs:string"/>
                <xs:attribute name="premiumFrequency" use="required"
type="aa:SummarypremiumFrequency"/>
                <xs:attribute name="premiumPaymentMonths" use="required"
type="xs:string"/>
                <xs:attribute name="premiumPaymentYears" use="required"
type="xs:string"/>
                <xs:attribute name="sumAssured" use="required" type="xs:string"/>
                <xs:attribute name="tenureMonths" use="required" type="xs:string"/>
                <xs:attribute name="tenureYears" use="required" type="xs:string"/>
                <xs:attribute name="type" use="required" type="aa:SummaryTypes"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="Holders">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="aa:Holder"/>
            </xs:sequence>
            <xs:attribute name="type" use="required" type="aa:HoldersType"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="Holder">
        <xs:complexType>
            <xs:attribute name="aadhaar" use="required" type="aa:HolderAadhar"/>
            <xs:attribute name="address" use="required" type="xs:string"/>
            <xs:attribute name="email" use="required" type="aa:HolderEmail"/>
            <xs:attribute name="landline" use="required" type="xs:string"/>
            <xs:attribute name="mobile" use="required" type="xs:integer"/>
            <xs:attribute name="name" use="required" type="xs:string"/>
            <xs:attribute name="pan" use="required" type="aa:HolderPan"/>
            <xs:attribute name="rank" use="required" type="aa:HoldersRank"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="Riders">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="aa:Rider"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
```

```
<xs:element name="Rider">
	<xs:complexType>
		<xs:attribute name="policyEndDate" use="required" type="xs:date"/>
		<xs:attribute name="policyStartDate" use="required" type="xs:date"/>
		<xs:attribute name="premiumAmount" use="required" type="xs:string"/>
		<xs:attribute name="riderType" use="required" type="xs:string"/>
		<xs:attribute name="sumAssured" use="required" type="xs:string"/>
		<xs:attribute name="tenureMonths" use="required" type="xs:string"/>
		<xs:attribute name="tenureYears" use="required" type="xs:string"/>
	</xs:complexType>
</xs:element>
<xs:element name="MoneyBacks">
	<xs:complexType>
		<xs:sequence>
			<xs:element ref="aa:MoneyBack"/>
		</xs:sequence>
	</xs:complexType>
</xs:element>
<xs:element name="MoneyBack">
	<xs:complexType>
		<xs:attribute name="amount" use="required" type="xs:string"/>
		<xs:attribute name="date" use="required" type="xs:date"/>
		<xs:attribute name="description" use="required" type="xs:string"/>
	</xs:complexType>
</xs:element>
<xs:element name="Covers">
	<xs:complexType>
		<xs:sequence>
			<xs:element ref="aa:Cover"/>
		</xs:sequence>
	</xs:complexType>
</xs:element>
<xs:element name="Cover">
	<xs:complexType>
		<xs:attribute name="amount" use="required" type="xs:string"/>
		<xs:attribute name="description" use="required" type="xs:string"/>
	</xs:complexType>
</xs:element>
<xs:element name="Transactions">
	<xs:complexType>
		<xs:sequence>
			<xs:element ref="aa:Transaction"/>
		</xs:sequence>
		<xs:attribute name="endDate" use="required" type="xs:date"/>
		<xs:attribute name="startDate" use="required" type="xs:date"/>
	</xs:complexType>
```

```
    </xs:element>
    <xs:element name="Transaction">
        <xs:complexType>
            <xs:attribute name="amount" use="required" type="xs:string"/>
            <xs:attribute name="date" use="required" type="xs:date"/>
            <xs:attribute name="id" use="required" type="xs:string"/>
            <xs:attribute name="narration" use="required" type="xs:string"/>
            <xs:attribute name="type" use="required" type="xs:string"/>
        </xs:complexType>
    </xs:element>
</xs:schema>
```

Bonds

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:aa="http://api.rebit.org.in/FISchema/bonds" elementFormDefault="qualified"
 targetNamespace="http://api.rebit.org.in/FISchema/bonds"
    xmlns:vc="http://www.w3.org/2007/XMLSchema-versioning"          vc:minVersion="1.0"
vc:maxVersion="1.1">
    <xs:simpleType name="SummaryTaxable">
        <xs:restriction base="xs:string">
            <xs:enumeration value="yes"/>
            <xs:enumeration value="no"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="HoldersType">
        <xs:restriction base="xs:string">
            <xs:enumeration value="single"/>
            <xs:enumeration value="joint"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="HolderAadhar">
        <xs:restriction base="xs:string">
            <xs:pattern value="[0-9]{12}"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="HolderEmail">
        <xs:restriction base="xs:string">
            <xs:pattern value="[^@]+@[^\.]+\..+"/>
```

```
            </xs:restriction>
        </xs:simpleType>
        <xs:simpleType name="HolderPan">
            <xs:restriction base="xs:string">
                <xs:pattern    value="[a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][0-9][0-9][0-9][0-
9][a-zA-Z]"/>
            </xs:restriction>
        </xs:simpleType>
        <xs:simpleType name="HoldersRank">
            <xs:restriction base="xs:string">
                <xs:enumeration value="1"/>
                <xs:enumeration value="2"/>
            </xs:restriction>
        </xs:simpleType>
        <xs:simpleType name="SummaryinterestComputation">
            <xs:restriction base="xs:string">
                <xs:enumeration value="simple"/>
                <xs:enumeration value="compound"/>
            </xs:restriction>
        </xs:simpleType>
        <xs:simpleType name="SummaryinterestCompoFrequency ">
            <xs:restriction base="xs:string">
                <xs:enumeration value="monthly"/>
                <xs:enumeration value="quarterly"/>
                <xs:enumeration value="halfYearly"/>
                <xs:enumeration value="yearly"/>
            </xs:restriction>
        </xs:simpleType>
        <xs:simpleType name="SummaryInterestPayout">
            <xs:restriction base="xs:string">
                <xs:enumeration value="monthly"/>
                <xs:enumeration value="quarterly"/>
                <xs:enumeration value="halfYearly"/>
                <xs:enumeration value="yearly"/>
                <xs:enumeration value="onMaturity"/>
            </xs:restriction>
        </xs:simpleType>
        <xs:simpleType name="TransactionTypes">
            <xs:restriction base="xs:string">
                <xs:enumeration value="opening"/>
                <xs:enumeration value="interest"/>
                <xs:enumeration value="tds"/>
                <xs:enumeration value="installment"/>
                <xs:enumeration value="closing"/>
                <xs:enumeration value="others"/>
            </xs:restriction>
```

```
    </xs:simpleType>
    <!--  -->
    <xs:element name="Account">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="aa:Summary"/>
                <xs:element ref="aa:Transactions"/>
            </xs:sequence>
            <xs:attribute name="id" use="required" type="xs:string"/>
            <xs:attribute name="type"  type="xs:string" fixed="bonds"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="Summary">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="aa:Holders"/>
            </xs:sequence>
            <xs:attribute name="FIP" use="required" type="xs:string"/>
            <xs:attribute name="compoundingFrequency" use="required"
type="aa:SummaryinterestCompoFrequency"/>
            <xs:attribute name="coupon" use="required" type="xs:string"/>
            <xs:attribute name="creditRating" use="required" type="xs:string"/>
            <xs:attribute name="currentValue" use="required" type="xs:string"/>
            <xs:attribute name="description" use="required" type="xs:string"/>
            <xs:attribute name="faceValue" use="required" type="xs:string"/>
            <xs:attribute name="interestComputation" use="required"
type="aa:SummaryinterestComputation"/>
            <xs:attribute name="interestOnMaturity" use="required" type="xs:string"/>
            <xs:attribute name="interestPayout" use="required"
type="aa:SummaryInterestPayout"/>
            <xs:attribute name="interestPeriodicPayoutAmount" use="required"
type="xs:string"/>
            <xs:attribute name="interestRate" use="required" type="xs:string"/>
            <xs:attribute name="isin" use="required" type="xs:string"/>
            <xs:attribute name="issueDate" use="required" type="xs:date"/>
            <xs:attribute name="maturityDate" use="required" type="xs:date"/>
            <xs:attribute name="principalAmount" use="required" type="xs:string"/>
            <xs:attribute name="quote" use="required" type="xs:string"/>
            <xs:attribute name="quoteDate" use="required" type="xs:date"/>
            <xs:attribute name="symbol" use="required" type="xs:string"/>
            <xs:attribute name="taxable" use="required" type="aa:SummaryTaxable"/>
            <xs:attribute name="tenureDays" use="required" type="xs:string"/>
            <xs:attribute name="tenureMonths" use="required" type="xs:string"/>
            <xs:attribute name="tenureYears" use="required" type="xs:string"/>
        </xs:complexType>
    </xs:element>
```

```
    <xs:element name="Holders">
        <xs:complexType mixed="true">
            <xs:sequence>
                <xs:element minOccurs="0" maxOccurs="unbounded"
ref="aa:Holder"/>
            </xs:sequence>
            <xs:attribute name="type" use="required" type="aa:HoldersType"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="Holder">
        <xs:complexType>
            <xs:attribute name="aadhaar" use="required" type="aa:HolderAadhar"/>
            <xs:attribute name="address" use="required" type="xs:string"/>
            <xs:attribute name="email" use="required" type="aa:HolderEmail"/>
            <xs:attribute name="landline" type="xs:string"/>
            <xs:attribute name="mobile" use="required" type="xs:integer"/>
            <xs:attribute name="name" use="required" type="xs:string"/>
            <xs:attribute name="pan" use="required" type="aa:HolderPan"/>
            <xs:attribute name="rank" use="required" type="aa:HoldersRank"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="Transactions">
        <xs:complexType mixed="true">
            <xs:sequence>
                <xs:element minOccurs="0" maxOccurs="unbounded"
ref="aa:Transaction"/>
            </xs:sequence>
            <xs:attribute name="endDate" use="required" type="xs:date"/>
            <xs:attribute name="startDate" use="required" type="xs:date"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="Transaction">
        <xs:complexType>
            <xs:attribute name="amount" use="required" type="xs:string"/>
            <xs:attribute name="date" use="required" type="xs:date"/>
            <xs:attribute name="id" use="required" type="xs:string"/>
            <xs:attribute name="narration" type="xs:string"/>
            <xs:attribute name="type" type="aa:TransactionTypes"/>
            <xs:attribute name="valueDate" use="required" type="xs:date"/>
        </xs:complexType>
    </xs:element>
</xs:schema>
```

Equities

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:aa="http://api.rebit.org.in/FISchema/equities" elementFormDefault="qualified"
 targetNamespace="http://api.rebit.org.in/FISchema/equities"
    xmlns:vc="http://www.w3.org/2007/XMLSchema-versioning"          vc:minVersion="1.0"
vc:maxVersion="1.1">
    <!--  -->
    <xs:simpleType name="HoldingEmail">
        <xs:restriction base="xs:string">
            <xs:pattern value="[^@]+@[^\.]+\..+"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="HoldingPan">
        <xs:restriction base="xs:string">
            <xs:pattern    value="[a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][0-9][0-9][0-9][0-
9][a-zA-Z]"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="HolderAadhar">
        <xs:restriction base="xs:string">
            <xs:pattern value="[0-9]{12}"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="TransactionsType">
        <xs:restriction base="xs:string">
            <xs:enumeration value="common stock"/>
            <xs:enumeration value="prefered stock"/>
            <xs:enumeration value="additional paid-in capital"/>
            <xs:enumeration value="contributed surplus"/>
            <xs:enumeration value="retained earning"/>
            <xs:enumeration value=" others"/>
        </xs:restriction>
    </xs:simpleType>
    <!--  -->
    <xs:element name="Account">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="aa:Summary"/>
                <xs:element ref="aa:Transactions"/>
            </xs:sequence>
            <xs:attribute name="id" use="required" type="xs:string"/>
            <xs:attribute name="type" use="required" type="xs:string"/>
        </xs:complexType>
```

```
    </xs:element>
    <xs:element name="Summary">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="aa:Holders"/>
            </xs:sequence>
            <xs:attribute name="cashPosition" use="required" type="xs:string"/>
            <xs:attribute name="openingDate" use="required" type="xs:date"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="Holders">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="aa:Holder"/>
            </xs:sequence>
            <xs:attribute name="type" use="required" type="xs:string"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="Holder">
        <xs:complexType>
            <xs:attribute name="aadhaar" use="required" type="aa:HolderAadhar"/>
            <xs:attribute name="address" use="required" type="xs:string"/>
            <xs:attribute name="email" use="required" type="xs:string"/>
            <xs:attribute name="landline" use="required" type="xs:string"/>
            <xs:attribute name="mobile" use="required" type="xs:string"/>
            <xs:attribute name="name" use="required" type="xs:string"/>
            <xs:attribute name="pan" use="required" type="aa:HoldingPan"/>
            <xs:attribute name="rank" use="required" type="xs:string"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="Transactions">
        <xs:complexType>
            <xs:sequence>
                <xs:element ref="aa:Transaction"/>
            </xs:sequence>
            <xs:attribute name="endDate" use="required" type="xs:date"/>
            <xs:attribute name="startDate" use="required" type="xs:date"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="Transaction">
        <xs:complexType>
            <xs:attribute name="bseSymbol" use="required" type="xs:string"/>
            <xs:attribute name="companyName" use="required" type="xs:string"/>
            <xs:attribute name="date" use="required" type="xs:date"/>
            <xs:attribute name="exchange" use="required" type="xs:string"/>
            <xs:attribute name="id" use="required" type="xs:string"/>
```

```
                    <xs:attribute name="isin" use="required" type="xs:string"/>
                    <xs:attribute name="narration" use="required" type="xs:string"/>
                    <xs:attribute name="nseSymbol" use="required" type="xs:string"/>
                    <xs:attribute name="otherTaxes" use="required" type="xs:string"/>
                    <xs:attribute name="rate" use="required" type="xs:string"/>
                    <xs:attribute name="stt" use="required" type="xs:string"/>
                    <xs:attribute name="totalCharge" use="required" type="xs:string"/>
                    <xs:attribute name="tradeValue" use="required" type="xs:string"/>
                    <xs:attribute name="type" use="required" type="aa:TransactionsType"/>
                    <xs:attribute name="units" use="required" type="xs:string"/>
            </xs:complexType>
    </xs:element>
</xs:schema>
```

National Pension Scheme

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
 xmlns:aa="http://api.rebit.org.in/FISchema/nps" elementFormDefault="qualified"
 targetNamespace="http://api.rebit.org.in/FISchema/nps"
 xmlns:vc="http://www.w3.org/2007/XMLSchema-versioning" vc:minVersion="1.1">
 <!--  -->
 <xs:simpleType name="SummaryStatus">
    <xs:restriction base="xs:string">
    <xs:enumeration value="active"/>
    <xs:enumeration value="deactivated"/>
    </xs:restriction>
 </xs:simpleType>
 <xs:simpleType name="SummaryTier1Status">
    <xs:restriction base="xs:string">
    <xs:enumeration value="active"/>
    <xs:enumeration value="deactivated"/>
    <xs:enumeration value="frozen"/>
    </xs:restriction>

 </xs:simpleType>
 <xs:simpleType name="SummaryTier1SchemePreferenceType">
    <xs:restriction base="xs:string">
    <xs:enumeration value="auto"/>
    <xs:enumeration value="active"/>
    </xs:restriction>
 </xs:simpleType>
```

```
<xs:simpleType name="SummaryTier2Status">
   <xs:restriction base="xs:string">
   <xs:enumeration value="active"/>
   <xs:enumeration value="frozen"/>
   <xs:enumeration value="deactivated"/>
   <xs:enumeration value="na"/>
   </xs:restriction>

</xs:simpleType>
<xs:simpleType name="SummaryTier2SchemePreferenceType">
   <xs:restriction base="xs:string">
   <xs:enumeration value="auto"/>
   <xs:enumeration value="active"/>
   <xs:enumeration value="na"/>
   </xs:restriction>
</xs:simpleType>


<xs:simpleType name="HolderAadhar">
   <xs:restriction base="xs:string">
   <xs:pattern value="[0-9]{12}"/>
   </xs:restriction>
</xs:simpleType>

<xs:simpleType name="HolderEmail">
   <xs:restriction base="xs:string">
   <xs:pattern value="[^@]+@[^\.]+\..+"/>
   </xs:restriction>
</xs:simpleType>
<xs:simpleType name="HolderPan">
   <xs:restriction base="xs:string">
   <xs:pattern value="[a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][a-zA-Z][0-9][0-9][0-9][0-9][a-zA-Z]"/>
   </xs:restriction>
</xs:simpleType>

<!--  -->
<xs:element name="Account">
   <xs:complexType>
   <xs:sequence>
   <xs:element ref="aa:Summary"/>
   <xs:element ref="aa:Holdings"/>
   <xs:element ref="aa:Transactions"/>
   </xs:sequence>
   <xs:attribute name="id" use="required" type="xs:string"/>
   <xs:attribute name="type" use="required" type="xs:string"/>
   </xs:complexType>
```

```
  </xs:element>
  <xs:element name="Summary">
     <xs:complexType>
     <xs:sequence>
     <xs:element ref="aa:Holders"/>
     <xs:element ref="aa:SchemeChoices"/>
     </xs:sequence>
     <xs:attribute name="currentValue" use="required"/>
     <xs:attribute name="debtAssetValue" use="required"/>
     <xs:attribute name="equityAssetValue" use="required"/>
     <xs:attribute name="openingDate" use="required"/>
     <xs:attribute name="otherAssetValue" use="required"/>
     <xs:attribute name="status" use="required" type="aa:SummaryStatus"/>
     <xs:attribute name="tier1FreeUnits" use="required"/>
     <xs:attribute name="tier1InvestmentCost" use="required"/>
     <xs:attribute name="tier1InvestmentValue" use="required"/>
     <xs:attribute name="tier1NAVDate" use="required"/>
     <xs:attribute name="tier1SchemePreferenceType" use="required"
type="aa:SummaryTier1SchemePreferenceType"/>
     <xs:attribute name="tier1Status" use="required" type="aa:SummaryTier1Status"/>
     <xs:attribute name="tier2FreeUnits" use="required"/>
     <xs:attribute name="tier2InvestmentCost" use="required"/>
     <xs:attribute name="tier2InvestmentValue" use="required"/>
     <xs:attribute name="tier2NAVDate" use="required"/>
     <xs:attribute name="tier2SchemePreferenceType" use="required"
type="aa:SummaryTier2SchemePreferenceType"/>
     <xs:attribute name="tier2Status" use="required" type="aa:SummaryTier2Status"/>
     </xs:complexType>
  </xs:element>
  <xs:element name="Holders">
     <xs:complexType>
     <xs:sequence>
     <xs:element ref="aa:Holder"/>
     </xs:sequence>
     </xs:complexType>
  </xs:element>
  <xs:element name="Holder">
     <xs:complexType>
     <xs:attribute name="aadhaar" use="required" type="aa:HolderAadhar"/>
     <xs:attribute name="address" use="required" type="xs:string"/>
     <xs:attribute name="email" use="required" type="aa:HolderEmail"/>
     <xs:attribute name="landline" use="required" type="xs:string"/>
     <xs:attribute name="mobile" use="required" type="xs:string"/>
     <xs:attribute name="name" use="required" type="xs:string"/>
     <xs:attribute name="pan" use="required" type="aa:HolderPan"/>
     </xs:complexType>
```

```
</xs:element>
<xs:element name="SchemeChoices">
   <xs:complexType>
   <xs:sequence>
   <xs:element ref="aa:SchemeChoice"/>
   </xs:sequence>
   </xs:complexType>
</xs:element>
<xs:element name="SchemeChoice">
   <xs:complexType>
   <xs:attribute name="allocationPercent" use="required" type="xs:string"/>
   <xs:attribute name="pfmId" use="required" type="xs:string"/>
   <xs:attribute name="pfmName" use="required" type="xs:string"/>
   <xs:attribute name="schemeId" use="required" type="xs:string"/>
   <xs:attribute name="schemeName" use="required" type="xs:string"/>
   </xs:complexType>
</xs:element>
<xs:element name="Holdings">
   <xs:complexType>
   <xs:sequence>
   <xs:element ref="aa:Tier1Holdings"/>
   <xs:element ref="aa:Tier2Holdings"/>
   </xs:sequence>
   </xs:complexType>
</xs:element>
<xs:element name="Tier1Holdings">
   <xs:complexType>
   <xs:sequence>
   <xs:element ref="aa:Tier1Holding"/>
   </xs:sequence>
   </xs:complexType>
</xs:element>
<xs:element name="Tier1Holding">
   <xs:complexType>
   <xs:attribute name="amount" use="required" type="xs:string"/>
   <xs:attribute name="amountInTransition" use="required" type="xs:string"/>
   <xs:attribute name="blockedUnits" use="required" type="xs:string"/>
   <xs:attribute name="freeUnits" use="required" type="xs:string"/>
   <xs:attribute name="nav" use="required" type="xs:string"/>
   <xs:attribute name="schemeName" use="required" type="xs:string"/>
   <xs:attribute name="totalUnits" use="required" type="xs:string"/>
   <xs:attribute name="totalValueOfScheme" use="required" type="xs:string"/>
   </xs:complexType>
</xs:element>
<xs:element name="Tier2Holdings">
   <xs:complexType>
```

```
    <xs:sequence>
    <xs:element ref="aa:Tier2Holding"/>
    </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Tier2Holding">
    <xs:complexType>
    <xs:attribute name="amount" use="required" type="xs:string"/>
    <xs:attribute name="amountInTransition" use="required" type="xs:string"/>
    <xs:attribute name="blockedUnits" use="required"/>
    <xs:attribute name="freeUnits" use="required" type="xs:string"/>
    <xs:attribute name="nav" use="required" type="xs:string"/>
    <xs:attribute name="schemeName" use="required" type="xs:string"/>
    <xs:attribute name="totalUnits" use="required" type="xs:string"/>
    <xs:attribute name="totalValueOfScheme" use="required" type="xs:string"/>
    </xs:complexType>
  </xs:element>
  <xs:element name="Transactions">
    <xs:complexType>
    <xs:sequence>
    <xs:element ref="aa:Tier1SchemeTransactions"/>
    <xs:element ref="aa:Tier2SchemeTransactions"/>
    <xs:element ref="aa:Tier1InvestmentTransactions"/>
    <xs:element ref="aa:Tier2InvestmentTransactions"/>
    </xs:sequence>
    <xs:attribute name="endDate" use="required" type="xs:date"/>
    <xs:attribute name="startDate" use="required" type="xs:date"/>
    </xs:complexType>
  </xs:element>
  <xs:element name="Tier1SchemeTransactions">
    <xs:complexType>
    <xs:sequence>
    <xs:element ref="aa:Tier1SchemeTransaction"/>
    </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Tier1SchemeTransaction">
    <xs:complexType>
    <xs:attribute name="allocationPercent" use="required" type="xs:string"/>
    <xs:attribute name="amount" use="required" type="xs:string"/>
    <xs:attribute name="cumulativeUnits" use="required" type="xs:string"/>
    <xs:attribute name="date" use="required" type="xs:date"/>
    <xs:attribute name="id" use="required" type="xs:string"/>
    <xs:attribute name="narration" use="required" type="xs:string"/>
    <xs:attribute name="nav" use="required" type="xs:string"/>
    <xs:attribute name="schemeName" use="required" type="xs:string"/>
```

```
    <xs:attribute name="type" use="required" type="xs:string"/>
    <xs:attribute name="units" use="required" type="xs:string"/>
    </xs:complexType>
  </xs:element>
  <xs:element name="Tier2SchemeTransactions">
    <xs:complexType>
    <xs:sequence>
    <xs:element ref="aa:Tier2SchemeTransaction"/>
    </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Tier2SchemeTransaction">
    <xs:complexType>
    <xs:attribute name="allocationPercent" use="required" type="xs:string"/>
    <xs:attribute name="amount" use="required" type="xs:string"/>
    <xs:attribute name="cumulativeUnits" use="required" type="xs:string"/>
    <xs:attribute name="date" use="required" type="xs:string"/>
    <xs:attribute name="id" use="required" type="xs:string"/>
    <xs:attribute name="narration" use="required" type="xs:string"/>
    <xs:attribute name="nav" use="required" type="xs:string"/>
    <xs:attribute name="schemeName" use="required" type="xs:string"/>
    <xs:attribute name="type" use="required" type="xs:string"/>
    <xs:attribute name="units" use="required" type="xs:string"/>
    </xs:complexType>
  </xs:element>
  <xs:element name="Tier1InvestmentTransactions">
    <xs:complexType>
    <xs:sequence>
    <xs:element ref="aa:Tier1InvestmentTransaction"/>
    </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Tier1InvestmentTransaction">
    <xs:complexType>
    <xs:attribute name="date" use="required" type="xs:string"/>
    <xs:attribute name="employerContribution" use="required" type="xs:string"/>
    <xs:attribute name="id" use="required" type="xs:string"/>
    <xs:attribute name="narration" use="required" type="xs:string"/>
    <xs:attribute name="subscriberContribution" use="required" type="xs:string"/>
    <xs:attribute name="totalContribution" use="required" type="xs:string"/>
    <xs:attribute name="type" use="required" type="xs:string"/>
    </xs:complexType>
  </xs:element>
  <xs:element name="Tier2InvestmentTransactions">
    <xs:complexType>
    <xs:sequence>
```

```
      <xs:element ref="aa:Tier2InvestmentTransaction"/>
      </xs:sequence>
      </xs:complexType>
   </xs:element>
   <xs:element name="Tier2InvestmentTransaction">
      <xs:complexType>
      <xs:attribute name="date" use="required" type="xs:date"/>
      <xs:attribute name="employerContribution" use="required" type="xs:string"/>
      <xs:attribute name="id" use="required" type="xs:string"/>
      <xs:attribute name="narration" use="required" type="xs:string"/>
      <xs:attribute name="subscriberContribution" use="required" type="xs:string"/>
      <xs:attribute name="totalContribution" use="required" type="xs:string"/>
      <xs:attribute name="type" use="required" type="xs:string"/>
      </xs:complexType>
   </xs:element>
 </xs:schema>
```

16. Data Backup and Archival

Application data stores should be deployed with replication enabled. This would ensure higher availability of services accessing any datastore. Data from all application data stores should be backed up periodically and archived to ensure that in case of data loss, it can be recovered from the archive. The archive storage should be highly durable, reliable and available. In case the data has Personally Identifiable Information, it needs to be securely stored.

17. Fraud Monitoring

The transaction patterns between AA, FIU and User should be monitored for any fraudulent activity. Any unusual access patterns, too frequent access, over consenting, consent fatigue, and other such anomalies must be identified and monitored.

18. Localization

The mobile app and the web portal for the user have to support Localization for various Indian Languages. Initially the implementation would be for:

1. English
2. Hindi

Later, support for other Indian regional languages will be added.

19. Digital Signatures

19.1 FIU

FIU can login to the web portal provided by AA and generate a private-public key. AA stores the public key at the server end. When FIU is calling the AA APIs programmatically it can digitally sign the requests using the private key.

19.2 User Application

When a user registers in the mobile app, a public-private key pair is generated. The private key is stored locally in the system, encrypted by user's password. The public key is stored in the AA backend. Similarly, when the user logs into the web portal, a pair of public-private keys are generated (if none exists on the device). The private key is stored locally in the system, encrypted using user's password. All API requests and responses between User App and AA must be digitally signed using this private key. The consent artefact which is approved by the user must also be digitally signed using this private key.

19.3 AA to FIP and FIP to AA

All API calls from AA to FIP should be digitally signed using the private key of the AA. FIP will use the public key of the AA to validate. The public key of AA is available through Central Registry.

All notification API calls from FIP to AA should be digitally signed using the private key of the FIP. AA will use the public key of the FIP to validate. The public key of FIPs are available through Central Registry.

20. Development Practices

Agile development practices should be followed through iterative, test driven development methods. The source code should be committed to a version control system, preferably Git or SVN. Continuous Integration through Jenkins should be set up which will automatically build the services or modules and run unit tests on them. Multiple Environments should be setup for various levels of tests. These can be,

1) Development Environment
2) QA Environment
3) E2E Integration Test Environment
4) Performance Test Environment
5) UAT (User Acceptance Tests) Environment
6) Production Environment
7) Production Hotfix Environment

A new build should be deployed into these environments and should be subjected to automated and manual tests as required. Automated test scripts should be written for end to end testing and regression testing. Once a build is pushed to production the source code should be tagged with release dates and version numbers. Test coverages numbers should be measured and tracked for

each services for every build. Each of the micro service should have a build and release version and they should have an API (/v1/build) which will return that information. Every functionality in the platform should be available as independent service. For e.g. consent framework, should be available as a micro service. NADL should be able to support current version and the previous version of any API (in case other participant in the ecosystem uses NADL's API). For all external API calls, there should be an emulator, ability to turn off and on, the external API call. i.e. if esign is used in the platform, there should be ability to turn off esign and place a dummy signature instead of making the actual call. This will be helpful in testing phase and eliminate availability of external services. AA platform should adhere to the Master Directive published by the RBI, throughout the life cycle of the service. The link to the master directive by RBI is provided in the Reference section.

21. Development Documentation

The detailed architecture, source code, deployment, APIs and any other relevant details of the Account Aggregator Platform needs to be fully documented. These documentations for all versions should to be shared with NADL. Also appropriate training of NADL team should be undertaken covering all aspects of development

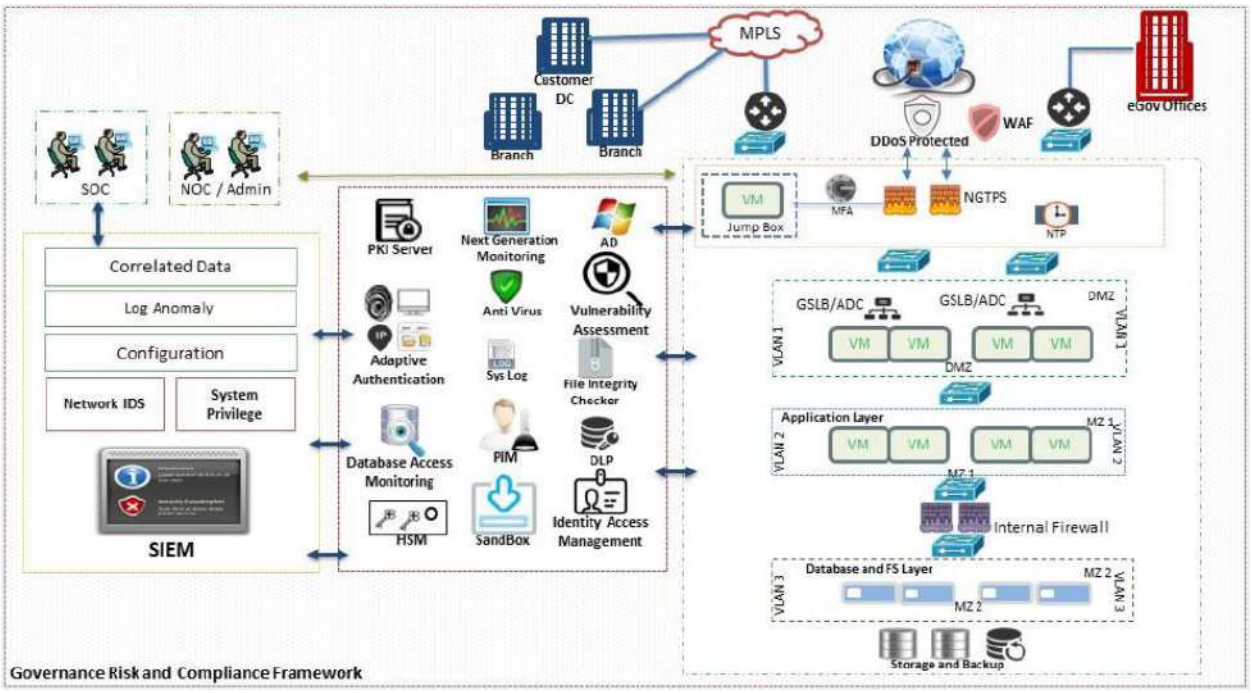
22. Issue (Bug) tracking and priorities

A bug tracking tool (e.g. JIRA) should be used for managing all bugs in the systems. The following guidelines should be considered related to bugs

- The bugs should be marked with priority levels P0, P1, P2 … with P0 having the highest priority.
- Bugs for each services should be tracked separately
- Any security related issues will always have P0 and should be addressed immediately. The security issue should be resolved in the best way possible, no later than an hour from the time it is reported. There can be an immediate risk mitigation along with a longer term appropriate fix.
- No P0 bugs should remain unattended within 12 hours and resolved within 24 hours. There can be an immediate work around or short term resolution followed by a longer term fix.
- No P1 bugs should remain unattended within 48 hours and resolved within a week. There can be an immediate work around or short term resolution followed by a longer term fix.
- A release to production can only happen if there is no P0 and P1 bugs.

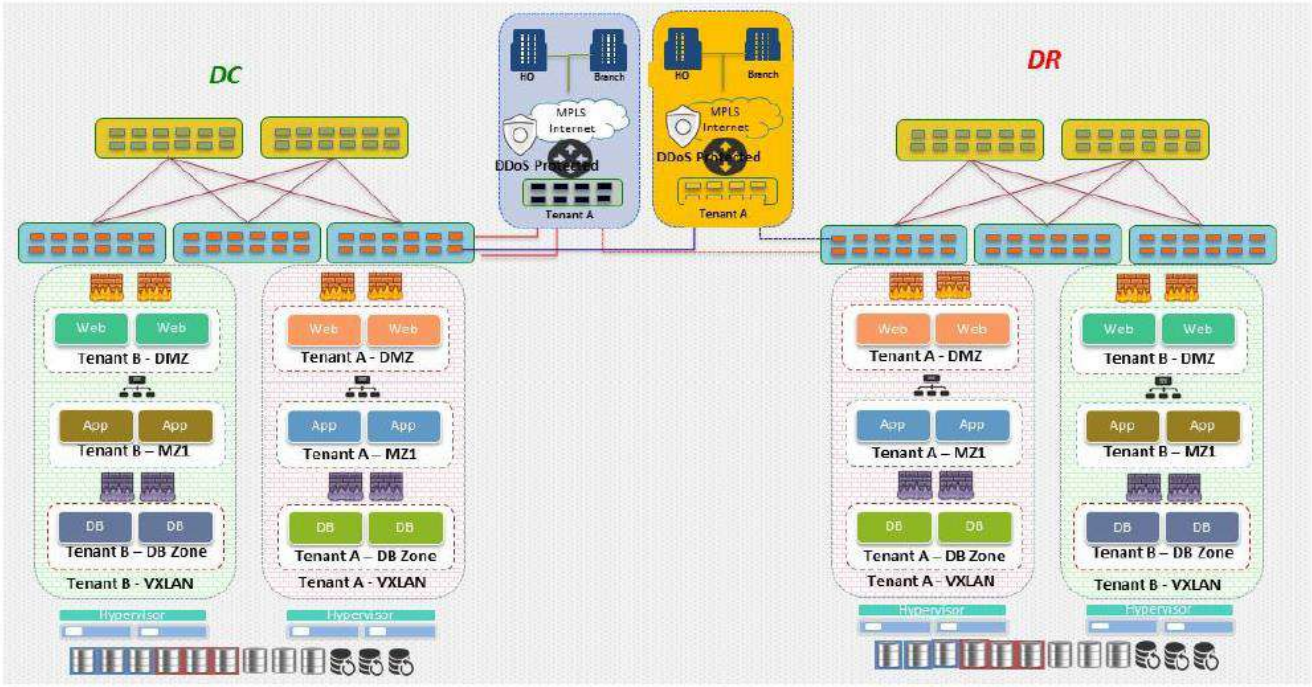
23. Infrastructure

23.1 Data Centers



Disclaimer: This is a reference diagram and not the actual one

23.2 DC & DR - Business Continuity (This a reference diagram and not the actual one)



Disclaimer: This is a reference diagram and not the actual one

23.3 Virtual Machines

Virtual Machine(VM) with below combination of CPU, RAM, Storage and OS. Hardware requirement needs to be mapped to the below configuration:

CPU / RAM (Per VM)	**Storage Space (Per VM)**	Operating System
1 core 12GB	**400 GB**	RHEL 7.3
2 core 8GB	**400 GB**	RHEL 7.3
2 core 24GB	**400 GB**	RHEL 7.3
2 core 16GB	**400 GB**	RHEL 7.3
4 core 48GB	**400 GB**	RHEL 7.3
4 core 64GB	**400 GB**	RHEL 7.3
8 core 128GB	**400 GB**	RHEL 7.3
12 core 192GB	**400 GB**	RHEL 7.3
16 core 256GB	**400 GB**	RHEL 7.3
16 core 256GB	**2 TB on DAS**	RHEL 7.3
24 core 384GB	**400 GB**	RHEL 7.3
32 core 512GB	**400 GB**	RHEL 7.3
32 core 512GB	**2 TB on DAS**	RHEL 7.3

24. Technical Stack

Some of the common technical stacks required,

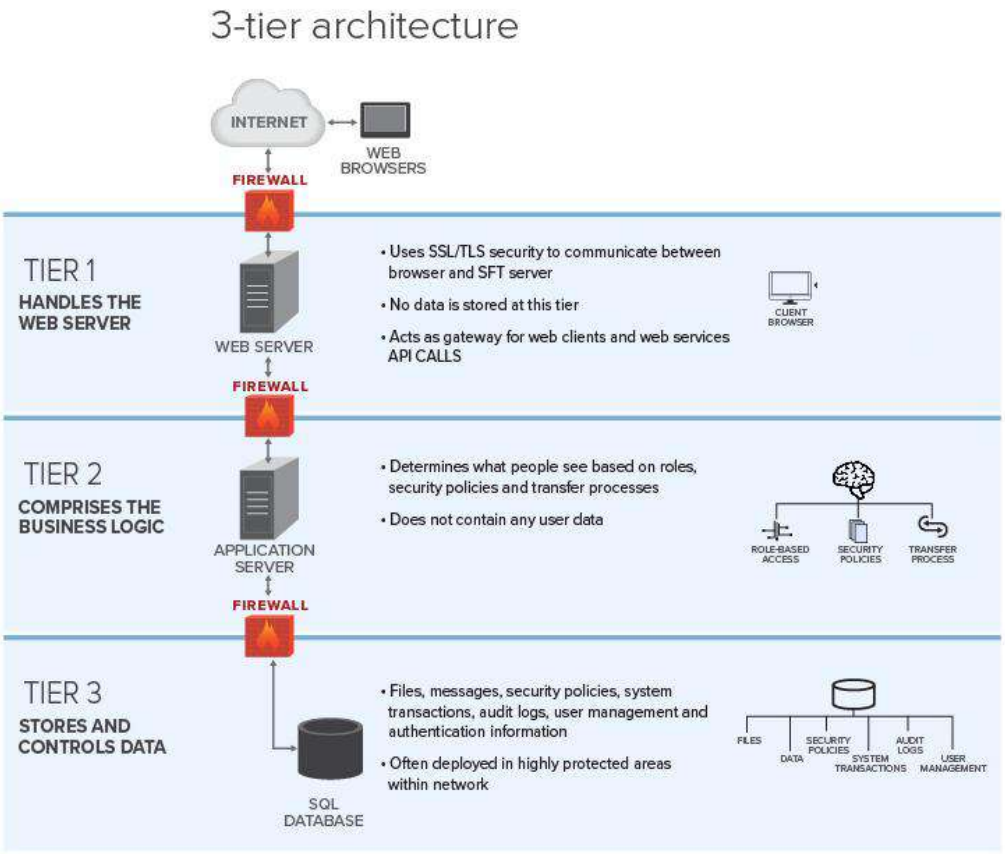
- OS Platform: Linux
- Database: MySQL
- Programming Language: Java (>= 1.8), Python (>=3.6)
- Messaging: Kafka
- Authentication, Authorization, & Security: OpenLDAP, Cryptography Techniques for Encryption, Decryption, Key Exchange Protocols
- Web-server: Nginx, Apache
- Distributed Coordination: Zookeeper
- Framework: Spring Boot
- Monitoring: Prometheus, Grafana

- Mobile App: Android, IOS
- API Gateway: Kong
- Unit Testing: Junit
- Devops: Chef/Ansible, Jenkins

Above is an indicative technical stack. NADL will have the final say in the selection of stack.

25. Deployment Architecture

The following represents high level multi-data center deployment.



26. Developer Experience

In order to ensure easy and seamless consumption of these APIs by developers for the purpose of development (integration), it's important that the developer experience for the APIs exposed by FIPs and AAs be given prime importance[1].

The APIs should be developer friendly and at minimum the following considerations must be met:

- Developer Portal: It must be publicly accessible and must contain:
    - API Sandbox
    - API Documentation and Reference

1

- Quick start Guides
- Open Source Libraries and SDKs

FIPs and AAs may further engage with developers via hackathons and other feedback channels like a Developer Forum or StackOverflow.

27. Grievance Redressal

There must be a mechanism in place by AAs to redress grievances of the users. This improves user satisfaction and builds user trust. The user may record their grievances / provide their feedback in writing, verbally, or digitally. SLAs must be put in place to ensure prompt response and necessary escalations in case of delays. Grievance Redressal can be further enabled through the use of detailed status and error messages in response to each request.

28. References

1. A good guideline for managing security in REST APIs is provided by the OWASP (Open Web Application Security Project) community: https://www.owasp.org/index.php/REST_Security_Cheat_Sheet
2. RBI Master Circular: https://www.rbi.org.in/Scripts/NotificationUser.aspx?Id=10598&Mode=0
3. Digital Locker Technology Framework: http://dla.gov.in/sites/default/files/pdf/DigitalLockerTechnologyFramework%20v1.1.pdf
4. Electronic Consent Framework, Technology Specifications: http://dla.gov.in/sites/default/files/pdf/MeitY-Consent-Tech-Framework%20v1.1.pdf
5. e-Sign specifications: http://www.cca.gov.in/cca/?q=eSign.html
6. Secure coding guidelines : https://wiki.mozilla.org/WebAppSec/Secure_Coding_Guidelines#Password_Storage
7. Income Tax Department, Online PAN verification Options, https://www.incometaxindia.gov.in/Pages/tax-services/online-pan-verification.aspx (accessed 10 March, 2018)
8. Department of Science & Technology, Government of India. “National Data Sharing and Accessibility Standards”. https://data.gov.in/sites/default/files/NDSAP.pdf (accessed 2 February,2018)
9. Institute for Development and Research in Banking Technology (IDRBT). “Certificate Authority”. http://www.idrbt.ac.in/idrbtca.html (accessed 2 February,2018)
10. Usability Guidelines for web applications https://www.designprinciplesftw.com/collections/10-usability-heuristics-for-user-interface-design

11. Usability Guidelines for mobile applications
    https://developer.apple.com/carplay/human-interface-guidelines/overview/introduction/
    https://developer.android.com/design/get-started/principles.html
12. Accessibility Guidelines for web applications
    https://www.w3.org/WAI/intro/wcag
    https://www.w3.org/standards/webdesign/accessibility
13. Accessibility Guidelines for mobile applications
    https://www.w3.org/WAI/mobile/
    https://www.w3.org/TR/mobile-accessibility-mapping/
14. Additional guidelines for India provided by Govt. Of India
    Indian Accessibility initiative & guidelines(Rights of Persons with Disabilities Act, 2016 (RPD)
    https://w3c.github.io/wai-policies-prototype/policies/india/
    Guidelines for Indian Government websites: http://web.guidelines.gov.in/
    Some features to consider on Accessibility in Indian web:
    http://goidirectory.nic.in/accessibilityfeatures.php
    http://digitalindia.gov.in/content/accessibility-statement
15. Open API specifications from ReBIT http://api.rebit.org.in/
16. Financial Information Schema https://api.rebit.org.in/schema
17. Open standards framework published by Meity
    http://egovstandards.gov.in/sites/default/files/Framework%20for%20Adoption%20of%20Open%20Source%20Software%20in%20e-Governance%20Systems.pdf
18. https://tools.ietf.org/html/rfc7519

(END OF SECTION – IV)

Section – V: Price Bid Format

Part A: Lump sum charges for Development and Deployment of Account Aggregator Software

Sr. No	Description	Unit	Quantity	Unit Price Rs.	GST %	GST Rs.	Amount Rs.
1	Lump sum charges towards development and deployment of Application Software for Account Aggregation, as per features and specifications stipulated in Section – IV of the RFP document	Lump sum	One				
2	Warranty, support and maintenance services as stipulated in RFP document, after deployment	Year	Five				
3	Commercial software required, if any - for the development and deployment of software for Account Aggregation	Number					

Part B: Manpower Rates for implementing Change Order:

Manpower required and estimated person-months for implementing a Major Change in the specifications is provided in the table below as a reference. Bidders are requested to provide the details in columns E, F & G. The total of Column H (Part B) will be added to Part A for evaluating the Financial Bid.

Sr. No.	Area of Expertise / Skill set (A)	Educational Qualification (B)	Relevant Work Exp. (years) (C)	Number (D)	Monthly Charges(Per Person) Rs. (E)	Applicable GST (F)	Estimated Man-months (G)	Amount Rs. (Per person month charge x est. man months) H = (E+ F)*G
1	User Experience Design (UX)	Graduate in Comp Sc/Electronic/Electrical	10+	1			1	
2	UI Developer	Graduate in Comp Sc/Electronic/Electrical	5-10	2			3	
3	Java Developer (Junior))	Graduate in Comp Sc/Electronic/Electrical	0 - 4	4			6	
4	Java Developer (Senior)	Graduate in Comp	5 - 10	3			6	

		Sc/Electronic/Electrical						
5	Principal Architect	Graduate, Post Graduate (desirable) in Comp Sc/Electronic/Electrical	12+	1			4	
6	DevOps Lead	Graduate in Comp Sc/Electronic/Electrical	10+	1			6	
7	Information Security Expert	Graduate in Comp Sc/Electronic/Electrical	10+	1			2	
8	QA with Automation Expertise(Junior)	Graduate in Comp Sc/Electronic/Electrical	2 - 4	2			4	
9	QA with Automation Expertise(Senior)	Graduate in Comp Sc/Electronic/Electrical	5 - 8	1			4	
10	MySQL DBA (Junior)	Graduate in Comp Sc/Electronic/Electrical	2 - 4	1			2	
11	MySQL DBA (Senior)	Graduate in Comp Sc/Electronic/Electrical	5 - 10	1			2	
12	Data Engineer (Junior)	Graduate in Comp Sc/Electronic/Electrical	4 - 6	1			4	
13	Data Engineer (Senior)	Graduate in Comp Sc/Electronic/Electrical	7 - 12	1			4	
14	Project Manager	PMP Certified	12+	1			3	
						Total of Part A and Part B Rs.		

Notes:

- **The financial bids will be evaluated by combining the prices quoted for both the Part A and Part B. However, the order will be placed for Part - A only.**
- **After placement of order, during the project period, if any change order is requested/required by NADL, the amount payable if any, vide the change order, will be computed on the basis of manpower costs quoted in Part - B. The estimated man-months**

for different categories of manpower given in Table is only for reference purposes. The actual utilization will be as per NADL project requirements.

- **The manpower cost at part B shall remain firm over the period stipulated in order.**
- **Above rates may be used by NADL or its parent company NeSL or group company of NeSL for any similar software development requirements which may not be part of this RFP.**

(End of Section – V)

ANNEXURE – A- Covering Letter

Date:
To:
Director,
NESL Asset Data Limited (NADL)
5th Floor, Spencer Towers,
86, M.G. Road, Bengaluru – 560001
Phone: - 080 -25580360, 022- 22446619
e-mail:- procurement@nadl.co.in

Subject: Submission of the Bid for Development and Deployment of Application Software

Dear Sir,

We, the undersigned, are pleased to offer to provide services towards Development and Deployment of Application Software for Account Aggregation, to NADL, Mumbai, in response to your RFP. No: **NADL/Account Aggregation/2018/001,** dated: 28th June 2018

We are hereby submitting our bid for same, comprising Envelopes 1 to 4.

We hereby declare that all the information and statements made in this bid are true and we accept that any misinterpretation contained in it, may lead to our disqualification.

We agree to abide by all the terms and conditions of the RFP document. We would hold the terms of our proposal valid for 90 days as stipulated in the RFP document.

We also undertake that we are not blacklisted or debarred from bidding process, by any Educational / R&D / Govt. Organization, as on date of submission of the bids and that there have been no regulatory actions initiated / pending against us as on the date of release of RFP.

We also undertake that, we shall not use the Account Aggregation technology developed under this project for any reverse engineering purposes, for a period of at least two years from the date of completion of project deliverables. We agree that the IPR of the Account Aggregation technology developed will vest with NADL perpetually.

We understand you are not bound to accept any bid you receive.

The undersigned is authorised to sign this bid document. The authority letter to this effect is enclosed.

Yours sincerely,

Authorized Signatory:
Name and Title of Signatory:
e-mail:
Mobile No:

ANNEXURE - B – Letter of Authority

(To be submitted in Original on Letterhead)

Date:
To:
PROJECT MANAGER
NESL Asset Data Limited (NADL)
5th Floor, Spencer Towers,
86, M.G. Road,
Bengaluru – 560001
Phone: - 080 -25580360, 022- 22446619
e-mail:- procurement@nadl.co.in

Subject: Authority Letter
Reference RFP. No: NADL/Account Aggregation/2018/001, dated: 28th June 2018

Dear Sir,

We, M/s _____ (Name of the bidder) having registered office at _____ (address of the bidder) herewith submit our bid against the said RFP document.

Mr./Ms. _____ (Name and designation of the signatory), whose signature is appended below, is authorized to sign and submit the bid documents on our behalf against RFP.

Specimen Signature:

The undersigned is authorised to issue such authorisation on behalf of us.

For M/s _____ (Name of the bidder)

Signature and company seal
Name
Designation
Email
Mobile No.

Annexure C: Details of Technical Manpower on Roll

Sr. No	Position /Designation	Educational Qualification	Experience in years	Numbers on Roll
1				
2				
3				

Virtual Machine(VM) with below combination of CPU, RAM, Storage and OS:

CPU / RAM (Per VM)	Storage Space (Per VM)	Operating System
1 core 12GB	400 GB	RHEL 7.3
2 core 8GB	400 GB	RHEL 7.3
2 core 24GB	400 GB	RHEL 7.3
2 core 16GB	400 GB	RHEL 7.3
4 core 48GB	400 GB	RHEL 7.3
4 core 64GB	400 GB	RHEL 7.3
8 core 128GB	400 GB	RHEL 7.3
12 core 192GB	400 GB	RHEL 7.3
16 core 256GB	400 GB	RHEL 7.3
16 core 256GB	2 TB on DAS	RHEL 7.3
24 core 384GB	400 GB	RHEL 7.3
32 core 512GB	400 GB	RHEL 7.3
32 core 512GB	2 TB on DAS	RHEL 7.3

Vendor needs to align the hardware sizing to the above configurations.

Databases

- MYSQL

Other Applications (Openstack)

- JBOSS - RHEL Subscription
- Nginx, Apache, Apache Zookeeper
- Open LDAP
- Kafka, Quartz

Software: The Licenses for RHEL 7.x is available with NADL. For any other software such as API gateway, etc., required licenses, need to be arranged by the vendor.

Annexure – E:

Irrevocable Performance Bank Guarantee (Draft copy only)
(On non-judicial paper of appropriate value)

To,
Director,
NESL Asset Data Limited (NADL)
5th Floor, Spencer Towers,
86, M.G. Road,
Bengaluru – 560001
Phone: - 080 -25580360, 022- 22446619
e-mail:- procurement@nadl.co.in

BANKS GUARANTEE NO:

DATE:

Dear Sir(S)

This has reference to the contract / Order No. _____ Dated _____ placed by NESL Asset Data Limited(NADL) on M/s_____ (Name & Address of vendor) for development, deployment, technical support and warranty of application software for accounts aggregation at NADL's site.

The conditions of this order provide that the vendor shall,

1. Arrange to develop the application software at NADL, as per details given in said contract /order, and
2. Arrange to install / deploy the said application software at NADL's site, to the entire satisfaction of NADL and
3. Arrange to provide on-site technical and other services to NADL as per scope stipulated in said order / contract
4. Arrange for the comprehensive warranty service support towards the software developed and deployed by supplier on site as per the warranty clause in said contract / order.

M/s (Name of Vendor) has accepted the said order/contract with the terms and conditions stipulated therein and have agreed to issue the performance bank guarantee (PBG) on their part, towards
promises and assurance of their contractual obligations vide the Supply Order No/.Contract _____

M/s. _____ (name of vendor) holds an account with us and has approached us and at their request and in consideration of the promises, we hereby furnish such guarantees as mentioned hereinafter.

NADL shall be at liberty without reference to the Bank and without affecting the full liability of the
Bank hereunder to take any other undertaking of security in respect of the suppliers obligations and / or liabilities under or in connection with the said contract or to vary the terms vis-a – vis the supplier or the said contract or to grant time and or indulgence to the supplier or to reduce or to increase or otherwise vary the prices or the total contract value or to forebear from enforcement of
all or any of the obligations of the supplier under the said contract and/or the remedies of NADL under any security now, or hereafter held by NADL and no such dealing(s) with the supplier or release or forbearance whatsoever shall have the effect of releasing the bank from its full liability of
NADL hereunder or of prejudicing right of NADL against the bank.

This undertaking guarantee shall be a continuing undertaking guarantee and shall remain valid and
irrevocable for all claims of NADL and liabilities of the supplier arising up to and until _____ (date)

This undertaking guarantee shall be in addition to any other undertaking or guarantee or security whatsoever the that NADL may now or at any time have in relation to its claims or the supplier's obligations/liabilities under and / or in connection with the said contract and NADL shall have the full authority to take recourse to or enforce this undertaking guarantee in preference to the other undertaking or security (ies) at its sole discretion and no failure on the part of NADL in enforcing or
requiring enforcement of any other undertaking or security shall have the effect of releasing the bank from its full liability hereunder.

We _____ (Name of Bank) hereby agree and irrevocably undertake and promise that if in your (NADL's) opinion any default is made by M/s _____ (Name of supplier) in performing any of the terms and /or conditions of the agreement or if in your opinion they commit any breach of the contract or there is any demand by you against M/s _____ (Name of supplier), then on notice to us by you, we shall on demand and without demur and without
reference to M/s _____ (Name of supplier), pay you, in any manner in which you may direct, the amount of Rs. _____/- (Rupees _____ Only) or such portion thereof as may be demanded by you not exceeding the said sum and as you may from time
to time require. Our liability to pay is not dependent or conditional on your proceeding against M/s
_____ (Name of supplier) and we shall be liable & obligated to pay the aforesaid amount as
and when demanded by you merely on an intimation being given by you and even before any legal
proceedings, if any, are taken against M/s _____ (Name of supplier)
The Bank hereby waives all rights at any time inconsistent with the terms of this undertaking guarantee and the obligations of the bank in terms hereof shall not be anywise affected or

suspended by reason of any dispute or disputes having been raised by the supplier (whether or not
pending before any arbitrator, Tribunal or Court) or any denial of liability by the supplier or any order or any order or communication whatsoever by the supplier stopping or preventing or purporting to stop or prevent payment by the Bank to NADL hereunder.

The amount stated in any notice of demand addressed by NADL to the Bank as claimed by NADL from the supplier or as suffered or incurred by NADL on the account of any losses or damages or costs, charges and/or expenses shall as between the Bank and NADL be conclusive of the amount so claimed or liable to be paid to NADL or suffered or incurred by NADL, as the case may be and payable by the Bank to NADL in terms hereof.
You shall have full liberty without reference to us and without affecting this guarantee, postpone for any time or from time to time the exercise of any of the powers and rights conferred on you under the contact with the said M/s _____ (Name of supplier) and to enforce or to
forbear from endorsing any power or rights or by reason of time being given to the said M/s _____ (name of supplier) which under law relating to the sureties would but for the provisions have the effect of releasing us.

You will have full liberty without reference to us and without affecting this guarantee, postpone for
any time or from time to time the exercise of any of the powers and rights conferred on you under the contract with the said M/s _____ (Name of supplier) and to enforce or to forbear from endorsing any power or rights or by reason of time being given to the said M/s _____ (Name of supplier) which under law relating to the sureties would but for the provisions have the effect of releasing us.

Your right to recover the said sum of Rs. _____/- (Rupees _____ only) from us in manner aforesaid will not be affected/ or suspended by reason of the fact that any dispute or disputes have been raised by the said M/s _____ (Name of Vendor) and/ or that any dispute or disputes are pending before any officer, tribunal or court or Arbitrator.

The guarantee herein contained shall not be determined or affected by the liquidation or winding up, dissolution or change of constitution or insolvency of the said M/s _____ (Name of Vendor)
but shall in all respects and for all purposes be binding and operative until payment of all dues to NADL in respect of such liability or liabilities.
Our liability under this guarantee is restricted to Rs._____/- (Rupees _____Only). Our guarantee shall remain in force until unless a suit action to enforce a claim under guarantee is filed against us within six months from (which is date of expiry of guarantee) all your rights under the said guarantee shall be forfeited and we shall be relieved and discharged from all liabilities there under.

We have power to issue this guarantee in your favour under Memorandum and Articles of Association of our Bank and the undersigned has full power to do under the power of Attorney dated…

Notwithstanding anything contained herein:
A. Our liability under this guarantee shall not exceed Rs_____ (in words)
B. This bank guarantee shall be valid up to _____ and unless a suit for action to enforce a claim under guarantee is filed against us within six months from the date of expiry of guarantee, all your rights under the said guarantee shall be forfeited and we shall be relieved and discharged from all liabilities there after i.e. after six months from the date of expiry of this Bank guarantee
C. We are liable to pay the guaranteed amount or any parts thereof under this bank guarantee only and only if you serve upon us a written claim or demand or before _____
D. The Bank guarantee will expire on (Min 24 months from the date of successful deployment of software in the order) _____ unless the same is renewed with the mutual consent of the parties herein.

Granted by the Bank

Yours faithfully,

For (Name of Bank)
SEAL OF THE BANK
Authorised Signatory

Annexure F - Technical Feature Compliance Statement

This is an indicative statement of features to be implemented by the bidder. Bidder's implementation should at the minimum comply with all the technical requirements stipulated at Para 3: Application Software Requirements, Section -IV of this document.

Major Features	**Complied (Yes / No)**
User Flow	
Registration	
Login	
KYC	
Consent Management	
Data Flow Management	
Notification	
Analytics	
Billing & Payment	
FIU Flow	
Registration	
Login	
Consent Management	
Data Flow Management	
Analytics	
Notification	
Billing & Payment	
FIU dashboard	
General Platform Features	
FIP Integrations	
Admin Dashboards	
Business Reporting Flow	

Security of Data-At-Rest	
Security of Data-In-Flight	
API Security	
Network Security	
Digital Signature for Non-Repudiability	
Identity & Authentication	
User Roles and Authorization	
Audit Trials & Logging	
Real-time Monitoring	
Compliance with Applicable National & International Data Protection Laws and Regulations	
Scale and Performance as required	

Annexure – G: Documents Checklist

SL. No	Documents to be Submitted	Submitted (Yes / No)
	Envelope – 1	
1	Demand Draft for Rs. 2360/- towards processing fee	
2	Demand Draft for Rs. 3,20,000/- or exemption document for EMD	
	Envelope – 2	
3	Covering Letter as per **Annexure - A**.	
4	Authority Letter as per **Annexure – B**	
5	A copy of Certificate of Incorporation, Partnership Deed / Memorandum and Articles of Association / any other equivalent document showing date and place of incorporation, as applicable.	
6	The copies of the audited Profit and Loss Account or a certificate from a Chartered Accountant, showing the annual turnover and profit for each of the financial years 2016-2017, 2015-2016 and 2014-2015.	
7	Copies of PAN and GST registration certificates.	
8	The documents establishing that the bidder has office/ establishment in/around Bengaluru.	
9	Other documents necessary in support of eligibility criteria (Section - II, para 4), product catalogues, brochures, etc.	
10	Check –list as per **Annexure - G**	
	Envelope – 3	
11	List of clients for whom the bidder has developed and deployed the application software of similar nature, in last five years.	
12	The details of application software developed and deployed by the bidder in last five years, giving details like technology platform used, the scope, volume, spread of software, the size of data, number of transactions, speed / latency, key features, etc.	

13	Copies of at least three supply orders / deployment reports, in support of sub-para 6 of para 4 (Eligibility Criteria) in Section – II.	
14	List of Technical and Administrative personnel on roll of the bidder, giving details of their educational qualifications (with specializations, if any), experience in the specific area as required for this project, etc. **(Annexure – C)**	
15	Technical Proposal including (but not limited to) understanding about the project, implementation Methodology, team composition, work schedule, PERT and Activity Schedule, interactions / visits, Data safety/ security measures, Quality Control, modular structure, escalation hierarchy, technologies /platforms to be used, requirements from NADL/ clients.	
16	The technical proposal giving the details of technical architecture and explain the platform scalability for various transaction volumes, flexibility towards modifications, etc. It should also detail the mapping of the proposed technology platform onto the Infrastructure detailed in **Annexure – D** : Application Software Requirements, Section -IV of this document. The bidder may please note that simply complying with the requirements does not automatically make the bidder technically qualified.	
17	A statement as per Annexure - F, showing bidder's compliance with the technical requirements covering all the parameters (but not limited to) stipulated at para 3: Application Software Requirements, Section -IV of this document. The bidder may please note that simply complying with the requirements does not automatically make the bidder technically qualified.	
18	The details required for technical evaluation of bids, pertaining to parameters mentioned in Section - II, at para 8 ii (tables A to D), in tabular form.	
19	Bidder should ensure to provide documents / declaration pertaining to the eligibility criteria mentioned in of this document.	
	Envelope – 4	
20	Price Bid as per format given in **Section - V**	

(END OF DOCUMENT)