****

|  |
| --- |
| **Listing Garage** |
| Solution Design Document |

****

**Table of contents**

[Chapter 1 – Journey Overview and Prioritization 4](#_Toc47104522)

[1.1 Introduction: 4](#_Toc47104523)

[1.1.1 IPO & New Listing 4](#_Toc47104524)

[1.1.2 Major Roles: 4](#_Toc47104525)

[1.2 NSE AS IS Equity IPO Listing Process: 5](#_Toc47104526)

[1.2.1 NSE Equity IPO details process flow Stages: 5](#_Toc47104527)

[1.2.2 Final Listing Process Summary (Workflow from T-2 day to T+6 Day) 6](#_Toc47104528)

[1.3 Reimagined Journey: 8](#_Toc47104529)

[Chapter 2 - Equity Listing - Preferred Reality Technology 9](#_Toc47104530)

[2.1 Design Constraint & Dependencies & Assumptions 9](#_Toc47104531)

[2.1.1 Design Constraints 9](#_Toc47104532)

[2.1.2 Dependencies 9](#_Toc47104533)

[2.1.3 Assumptions 9](#_Toc47104534)

[2.2 Identified Risk and Mitigation 9](#_Toc47104535)

[2.3 Key Non-functional Requirements 9](#_Toc47104536)

[2.4 Technology components 10](#_Toc47104537)

[2.4.1 Overall Technology component and Design 10](#_Toc47104538)

[2.4.2 UI Component 10](#_Toc47104539)

[2.4.3 API Gateway 10](#_Toc47104540)

[2.4.4 Micro Services 12](#_Toc47104541)

[2.4.5 Message Bus 14](#_Toc47104542)

[2.4.6 Event Bus 14](#_Toc47104543)

[2.4.7 Database 14](#_Toc47104544)

[2.4.8 BPM and Workflow Approval 14](#_Toc47104545)

[2.4.9 Document Management Solution 15](#_Toc47104546)

[2.4.10 Common Business Services 15](#_Toc47104547)

[2.5 Micro services 17](#_Toc47104548)

[2.5.1 List of Micro Services 18](#_Toc47104549)

[2.6 Data Architecture 20](#_Toc47104550)

[2.6.1 Data Model 20](#_Toc47104551)

[2.6.2 Data Migration 21](#_Toc47104552)

[2.7 Object Model 21](#_Toc47104553)

[2.8 Data Validations 21](#_Toc47104554)

[2.9 Business Rules 21](#_Toc47104555)

[2.10 Sequence Diagrams for Key Business Process Flows of the Journeys 21](#_Toc47104556)

[2.11 Logging 21](#_Toc47104557)

[2.12 Error and Exception Handling 21](#_Toc47104558)

[2.13 Auditing 22](#_Toc47104559)

[2.14 Caching 22](#_Toc47104560)

[2.15 Integration Interfaces 22](#_Toc47104561)

[2.15.1 Integration Flows 22](#_Toc47104562)

[2.15.2 Interface List 23](#_Toc47104563)

[2.15.3 Integration with ENIT 23](#_Toc47104564)

[2.15.4 Third Party Integration 23](#_Toc47104565)

[2.16 APIs Exposed For API Channel Interface 24](#_Toc47104566)

[2.17 Security Design 24](#_Toc47104567)

[2.17.1 Transport Security 24](#_Toc47104568)

[2.17.2 Data Security 24](#_Toc47104569)

[2.17.3 Authentication 25](#_Toc47104570)

[2.17.4 Authorization 25](#_Toc47104571)

# – Journey Overview and Prioritization

## **Introduction:**

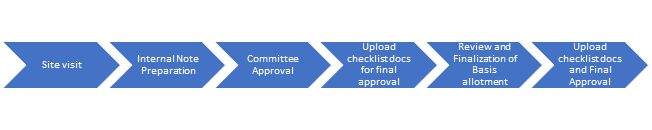
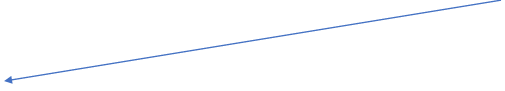
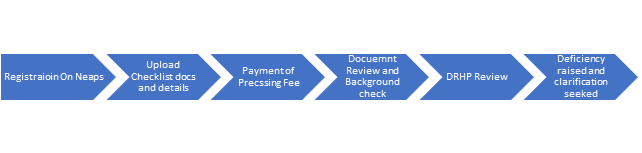
IPO & New Listing

* Initial Public Offer (IPO) is a process through which an unlisted Company can be listed on the stock exchange by offering its securities to the public in the primary market. The object of an IPO may be relating to expansion of existing activities of the Company or setting up of new projects or any other object as may be specified by the Company in its offer document or just to get its existing equity shares listed by diluting the stake of existing equity shareholders through offer for sale.
* New Listing is a process through which a company, which is already listed on other stock exchange/s, approaches the Exchange for listing of its equity shares. The companies fulfilling the eligibility criteria prescribed by the Exchange; from time to time; are listed on the Exchange.

### Major Roles:

* **Merchant Banker**: A merchant bank is a company that conducts underwriting, loan services, financial advising, and fundraising services for large corporations and high net worth individuals. Unlike retail or commercial banks, merchant banks do not provide services to the general public. They do not provide regular banking services like checking accounts and do not take deposits.
* **Issuer**: An issuer is a legal entity that develops, registers and sells securities to [finance](https://www.investopedia.com/terms/f/financing.asp) its operations. Issuers may be corporations, [investment trusts](https://www.investopedia.com/articles/basics/09/unit-investment-trust.asp), or domestic or foreign governments. Issuers are legally responsible for the obligations of the issue and for reporting financial conditions, material developments and any other operational activities as required by the regulations of their jurisdictions.
* **Security Exchange (E.g. NSE)**: A stock exchange, securities exchange is a facility where stockbrokers and traders can buy and sell securities, such as shares of stock and bonds and other financial instruments.
* RTA: Registrar and Transfer Agents principal functions are to issue and cancel certificates to reflect changes in ownership of the securities of an entity and to act as an intermediary for the company.
* **Depositories**: Depositories are institutions which hold your securities (Shares, bonds, debentures, Mutual Fund Units) in electronic form which is also known as dematerialization of shares or DEMAT account. So, Depositories are mainly responsible and accountable for safe-keeping of your securities and keep a record of all your trades. There are two main Depositories in India. 1- CDSL, 2-NSDL

## **NSE AS IS Equity IPO Listing Process:**



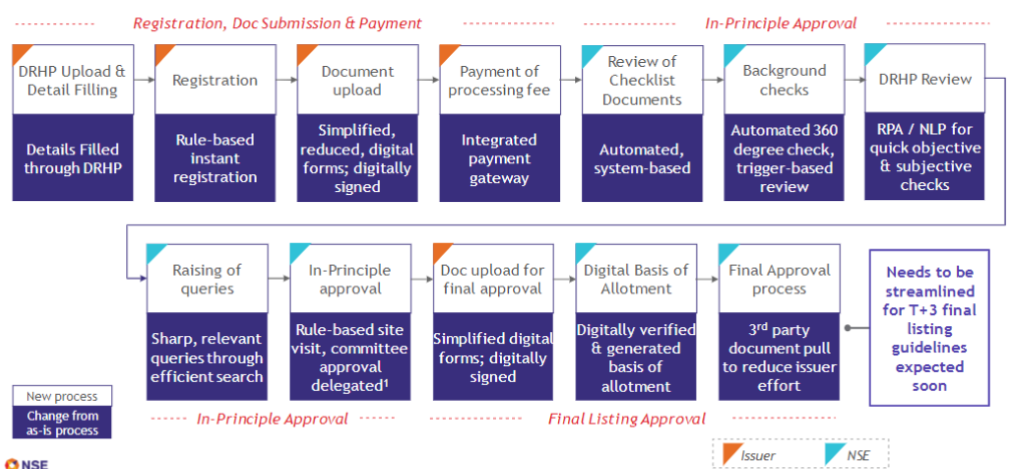
### NSE Equity IPO details process flow Stages:

1. Registration Stage:
   1. User (Issuer) Selects on New Registration on NEAPS Website
   2. User fill in registration form
   3. User uploads DRHP file
   4. Exchange officer validate the registration fields and documents
   5. Issuer receives login credentials
2. Checklist docs and Fee Payment:
   1. Issuer logs in using username and password
   2. Issuer uploads 10+ documents and confirms upload
   3. Issuer enters details of Promotors, Directors, Capital Structure Details, and Fee payment details
   4. Exchange allocates details to specific officer
3. In-Principle Approval Stage 1:
   1. Exchange officer verifies adherence eligibility criteria on basis of DRHP and Checklist documents
   2. Exchange officer performs detail analysis of DRHP by reading through each section
   3. If required clarification or extra documents are sought from Issuer/MB
   4. Investigation team searches debarred list for promotors and directors
4. In-Principle Approval Stage 2:
   1. Site Visit performed and Visiting officer prepare site visit report and sends email to exchange
   2. Exchange officer uploads the report on NEAPS
   3. Committee approval stage: Exchange officer prepares approval note and place before subcommittee of RoC (Regulatory Oversight Committee)
   4. Granting of In-Principle approval to Issuer and approval note sent to Issuer via e-mail
5. Book Building Process:
   1. RHP after SEBI approval re-submitted to exchange on NEAPS
   2. Issuer make 1% of Issue size as security payment to NSE through virtual account and F&A team of NSE validate the payment
   3. Issuer submits of Issue and Bidding parameters over email along with DRHP
   4. Bidding Opens
   5. Mock bidding set market participants (Banks, Brokers, Sponsors Bank and RTA)
   6. Bidding process starts:
      1. Bankers uploads bids on system and sent it through API
      2. UPI bids sent to sponsor bank through API
      3. Investor PAN and DP details sent to depositories for Validation
      4. Bid-Book Shared with RTA at EOD through SFTP
      5. Reconciliation file sent to Sponsor bank at EOD
   7. Bidding process Ends
6. Final Listing approval process:
7. Payment Stage1:
   1. PAN no and Company name sent to F&A team through NEAPS
   2. F&A team sends e-mail to IDBI bank with details of PAN no, Company name and Purpose
   3. IDBI responds back with Virtual account number
   4. F&A team raises request with IT team to send virtual, account no in NEAPS
   5. Email sent to Issuer with details of Virtual account number
   6. Issuer pays the fee send details to NSE
   7. NSE checks and verifies the payment

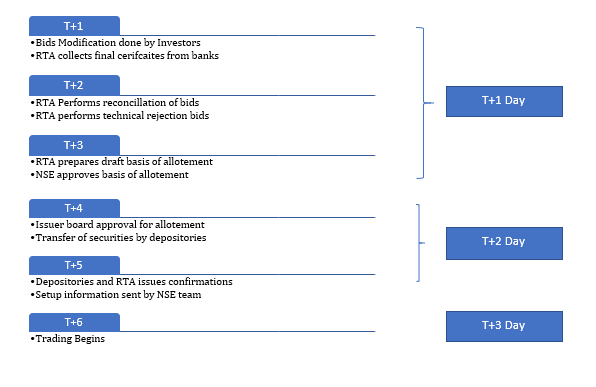
### Final Listing Process Summary (Workflow from T-2 day to T+6 Day)



## **Reimagined Journey:**



* Steps taken for Reducing final listing timeline from T+6 to T+3 Days



# - Equity Listing - Preferred Reality Technology

## **Design Constraint & Dependencies & Assumptions**

### Design Constraints

### Dependencies

### Assumptions

## **Identified Risk and Mitigation**

## **Key Non-functional Requirements**

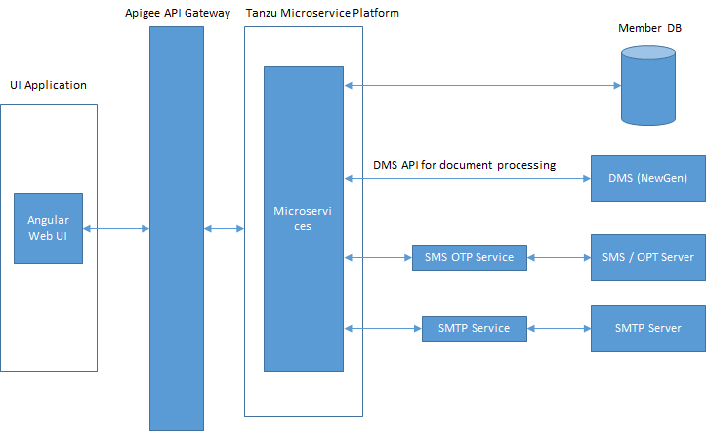
Refer to NSE Project Parivartan Digital Transformation Roadmap Document for Overall Key Non Functional Requirements defined. Few key Non Functional Requirement deviations listed below

No deviated Non Functional Requirement identified for Member Compliance. Member Compliance to follow NSE Project Parivartan Digital Transformation Roadmap Document for NFR and Preferred NFR.

|  |  |  |
| --- | --- | --- |
| Area | NFR | Specifications |
| **Availability** | Uptime requirements of new system | 99.99% availability |
| **Reliability** | The ability of a system to perform its required functions under stated conditions for a specific period of time. | Application Design should consider maximum of resilient features like Auto Scaling when failure of instance, Circuit breaker etc.. |
| **Recovery** | Recovery Point Objectives (RPO) | 30 min |
| Recovery Time Objectives (RTO) | 30 min |
| Backup frequencies – how often is the transaction data, config data, code backed-up | Data – 15 min, config, code – 30min |
| **Maintainability** | Conformance to Enterprise Architecture standards | Architects to validate Architecture standards being adapted |
| Conformance to Technical design standards | Architects to validate Architecture standards being adapted |
| Conformance to coding standards | Application to use Code Quality Tools to validate code quality standards to include coding standards rules, security rules etc.  Code quality should be above 95% |
| Conformance to best practices. | Application Technical Leads to verify and validate all best practice being followed with peer and external review check list. |
| **Usability** | Internationalization / localization requirements – languages, spellings, keyboards, etc | English language support only for now. No Internationalization and Localization support requirement |

## **Technology components**

### Overall Technology component and Design

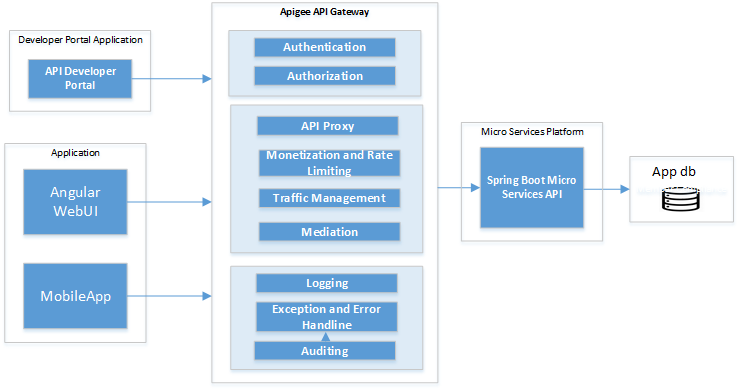


### UI Component

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Area** | **Decision** | **Rationale** |
| 1 | Front end – web | **Angular** | Angular is better from a security perspective. It has better support for 12 factor-coding standards. |
| 2 | Front end –mobile | **Reactnative** | Performance and user experience of React Native for mobile app development is better. |

### API Gateway

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Area** | **Decision** | **Rationale** |
| 3 | API Gatewway | **Apigee** | Apigee is better from the code flexibility, OOB policy availability, etc. |



**Apigee Gateway Key Creatures**

**Authentication –** Apigee API Gateway to has be enable with OAuth 2.0 Authentication out of the box feature and has to connect to LDAP.

Refer to NSE Project Parivartan Digital Transformation Roadmap Document for Authentication Process flow and sequence Flow

**Authorization –** Apigee API Gateway has to be enable with OAuth 2.0 Authorization out of the box feature.

Refer to NSE Project Parivartan Digital Transformation Roadmap Document for Authorization Process Flow and Sequence Flow

**API Proxy -** Provided features like API Reverse Proxy, Soap Service, No Target and Hosted Target. Use API Reverse option to create new proxy targeting to Sprig Boot Micro Services

|  |  |  |  |
| --- | --- | --- | --- |
| *S.No* | *Proxy Details* | *value* | *Comments* |
| *1* | *Proxy Name* | *Name of Gateway API* | *Same as Micro Service API Name* |
| *2* | *Base Path* | *This is resource End point URL where Angular UI can use for service invocation. This URL needs to be prefix with base URL of Apigee.* |  |
| *3* | *Description* | *Description of Gateway Service* | *Possible description of Spring Boot Micro Service* |
| *4* | *Target URL* | *Target URL of Spring Boot Micro Services* |  |
| *5* | *API Key, OAuth 2.0, Passthrough – No Authentication* | *Choose OAuth2.0 Option* |  |

**Monetization and Rate Limiting –** APIGEE API Gateway provides Monetization Feature to enable various rate plans for API Developers to subscribe and API Consumption rates will be calculated based upon Number of Request Vs Rate Plan

There is no requirement for API Monetization and consumption of APIs by external users hence no monetization applicable.

**Traffic Management –** APIGEE API Gateway provides traffic management feature to throttle APIs based upon number of concurrent/requests per second/minute/hour.

This throttling will be based on load for individual micro services and will be determined later.

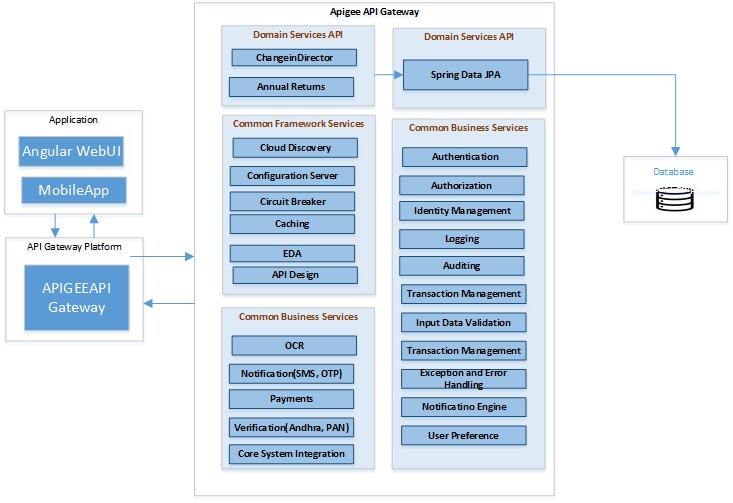
**Mediation –** APIGEE API Gateway provides feature to mediate request by applying various mediation policies like XML to JSON, JSON to XML etc.. All incoming and outgoing request will be designing with JSON format for both incoming and outgoing messages.

Individual application to apply this policy based upon the need for mediation.

**API Developer Portal –** APIGEE API Gateway provides Developer Portal for Developers to create account and self subscribe for APIs they want to consume. Developers will also be able to do basic monitoring of consumption/usage of their APIs with some key performance metrics for monitoring purpose. Based upon monitoring and observation, developers can throttle the requests to optimize using Traffic Management

### Micro Services

|  |  |  |
| --- | --- | --- |
| **S.No** | **Area** | **Comments** |
| 1 |  | Refer attached document for list of common components with various sections defined below • Common Framework Components • Common Business Services Cross Cutting Concerns |



#### Application Domain Services

Application Domain services are specific to Garages like Member Compliance to achieve any certain business functionality pertaining to Member Compliance. Spring Boot will be used to develop Micro Services API and Eureka for Cloud Service Discovery.

For Eg. ChangeInDirector Micro Service having end points like /DirectorList, /DirectorContactDetails etc.. Specific to Application business functionality

Refer to NSE Project Parivartan Digital Transformation Roadmap Document for developing Micro services, standards and Guidelines etc..

#### Common Business Services

Common Business Services are Micro Services common across Application Domain Services. Common Business Services also use Spring Boot Micro Services and Eureka for Cloud Service Discovery

For Eg. Notification Service to send SMS, OTP, Email etc. Any Application Domain Service to call this Notification Common Micro Service to send SMS notification desired for their application functionality

Refer to NSE Project Parivartan Digital Transformation Roadmap Document for developing Micro services, standards and Guidelines etc..

#### Common Framework Services

Common Framework Services are integral components of Spring Boot Micro Services and this will be developed as framework components for Spring Boot Micro Services to function. These Framework components will assed as maven dependency in Spring Boot Micro Services.

For Eg. Spring Cloud Discovery – Eureka is integral component of Spring Boot Micro Service and used for discovering multiple micro services

Refer NSE Project Parivartan Digital Transformation Roadmap Document for components in detail for each Framework components.

#### Micro Services Cross Cutting Concerns

Cross Cutting Concerns are utility components aspect over Spring Boot Micro Services. These services will be developed as either utility jar or independent Micro service.

For Eg. Logging and Exception Handling Framework to be developed as utility Jar whereas Auditing will be developed as independent common Micro Service where other Application specific Domain Services can invoke this Auditing Micro Service EndPoint for Audit purpose.

### Message Bus

### Event Bus

### Database

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Area** | **Decision** | **Rationale** |
| 5 | Application Database – Structured | **Oracle** |  |

### BPM and Workflow Approval

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Area** | **Decision** | **Rationale** |
| 5 | Business Process Management % Work Flow Approval | **IBM Servicenow** |  |

This section to be elaborated in next version

### Document Management Solution

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Area** | **Decision** | **Rationale** |
| 5 | Document Management System | **NextGen** |  |

This section to be elaborated in next version

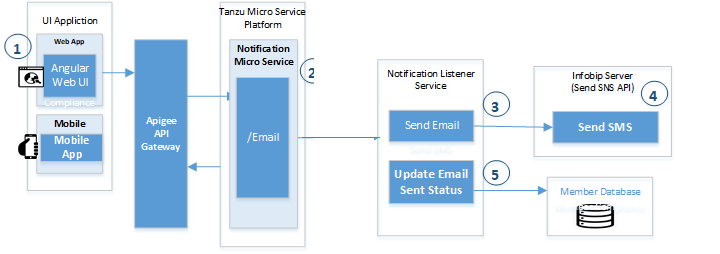
### Common Business Services

#### OCR/NLP Service

Product and Requirement yet to be discussed and finalized

#### Notification - Send OTP & SMS

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Area** | **Decision** | **Rationale** |
| 5 | Send SMS and OTP | **InfoBip Provided APIs** | Send SMS by calling InfoBip APIs  <https://dev.infobip.com/#programmable-communications/sms> |



#### Payment Services - BillDesk

Received documentation. Discussion/brainstorming in progress. This section to be detailed in next version.

To Draft Process flow in next version

**Browser Based Integration**

|  |  |  |
| --- | --- | --- |
| **Attrubite** | **Vaue** | **Comments** |
| Code Snipped to be included in Angular App | <script src="https://pgi.billdesk.com/payments-checkout-widget/src/app.bundle.js"></script> |  |
| JavaScript Code to initialize and invoke BillDesk | bdPayment.initialize ({  "msg":" ABCD|123456|NA|100.00|XYZ|NA|NA|INR|DIRECT|R|abcd|NA|NA|F|john@doe1.com|9820 198201|NA|NA|NA|NA|NA|NA|AB6VN3245B66FE9511DB2A854AAA32ADC563E789CF213CA1 9E274F18F330G547",  "options": {  "enableChildWindowPosting": true,  "enablePaymentRetry": true,  "retry\_attempt\_count": 2,  "txtPayCategory": "NETBANKING"  },  "callbackUrl": "https://www.merchant-domain.com/payment\_response.jsp"  }); |  |
| Transaction Status to be recessed in Merchant end | Successful transaction  Update <record> set STATUS = ‘SUCCESS’ where ORIGINALSTATUS=’PENDING’ and ORDERNUMBER=’123456789’ and TRANSACTIONAMOUNT=’100.00’  Failure transaction  Update <record> set STATUS = ‘FAILURE’ where ORIGINALSTATUS=’PENDING’ and ORDERNUMBER=’123456789’ and TRANSACTIONAMOUNT=’100.00’ |  |

**Server to Server Response Receiving**

Internal Discussion in progress. To be draft in next version

##### API List

Payment Transaction is sent through Initialize function

Online Query API:

<https://www.billdesk.com/pgidsk/PGIQueryController>

Refund API

<https://www.billdesk.com/pgidsk/PGIRefundController>

To be continued..

#### Background Verification & PAN Verification

Discussion in progress from requirement and perspective. This section to be detailed in next version.

## **Micro services**

|  |  |  |
| --- | --- | --- |
| **Microservice** | **Description** | **Dependencies** |
| Fetch List for the Regional Officers | Develop API to retrieve the details of Regional Officers and email is | Internal |
| Update the RO details for Site | API to update the Regional Office details for the Issuer Site. NSE Officer can change it. | Internal |
| Notification bar to enable Regional Office to visit the Site | Develop API to show notifications to regional office to start the Site visit review. | Internal |
|  |  |  |
| E-mail alerts | Develop API to send alerts on email for Site visit initiation process. | Internal |
|  |  |  |
| Add new Site for the Issuer | NSE Officer can add the New site for the Issuer with valid site details. | Internal |
| NSE DMS | Develop API Integration with Document Management System (DMS) | Internal |
| Save NSE Officer comment | Develop API to Save comments for the Site related queries. | Internal |
| Save RO Officer Site Visit Form | Develop API Integration with Document Management System (DMS) to same site visit details along with Attachment | Internal |
| Save RO Officer Site Visit Form observations | API to implement Regional officer response. | Internal |
| Fetch pin code details | API to fetch state and RO details after pin code is entered by RO Officer. | Internal |

### List of Micro Services

#### Application Domain Services

##### Site Visit

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***User Story/Function*** | ***Service Name*** | | ***Endpoint Description*** | ***Resource Endpoint*** |
| *Ro Selection* | *SiteVisitService* | *Fetch Regional Office Details along with Regional Office Id, Name and email* | | */api/v1/listing/site/ro* |
|
|
|
| *Update RO Details* | *SiteVisitService* | *Update the RO for existing site* | | */api/v1/listing/site/ro/* |
| *Fetch RO details from pincode* | *SiteVisitService* | *Fetch Regional Office Details from pin code* | | */api/v1/listing/site/state/{pincode}* |
| *Get Site Details* | *SiteVisitService* | *Fetch all site details for the selected Issuer ID* | | */api/v1/listing/site/details/{applicationid}* |
|
|
| *Create the new Site for the Issuer* | *SiteVisitService* | *Create the new sire for the selected Issuer* | | */api/v1/listing/site/{applicationId}* |
|
|
| *When Site visitor Officer visit the site and update the question and answer for site.* | *SiteVisitService* | *Update Question and answer for Site Visit* | | */api/v1/listing/site/draft/sitevisitform/{applicationId}* |
| *When Site visitor Officer visit the site and update the question and answer for site.* | *SiteVisitService* | *Add new question and answer for site visit* | | */api/v1/listing/site/submit/sitevisitform/{applicationId}* |
| *Add comment for undesired answer* | *SiteVisitService* | *Add the comment for the undesired answer for the site* | | */api/v1/listing/site/question/comment* |
| *Fetch state data from pin code* | *StateMasterService* | *Fetch the state details from pin code details* | | */api/v1/pincode/{pincode}* |

Please find attached request response details.



##### Issuer Registration

|  |  |  |  |
| --- | --- | --- | --- |
| ***User Story/Function*** | ***Service Name*** | ***Service Description*** | ***Resource Endpoint*** |
|  |  |  |  |

##### DRHP upload and review

|  |  |  |  |
| --- | --- | --- | --- |
| ***User Story/Function*** | ***Service Name*** | ***Service Description*** | ***Resource Endpoint*** |
|  |  |  |  |

##### Common Business Services

Below common services are only on assumption basis and these needs to be discussed and revised once we reach actual use case/story

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *S.No* | *Common User Story/Function* | *Service Name* | *Service Description* | *Resource Endpoint* | *API Spec* |
| *2* | *Send OTP to subscribed mobile number* | *Notification Service* | *Geenrate OTP using Java program logic, then send SMS along with OTP using InfoBip Server to designated mobile number* | */SMSwithOTP* |  |
| *3* | *Send SMS to registered Mobile umber* | *Notification Service* | *Send SMS using Vodafone Infobip serve to designated mobile number* | */SMS* | <https://qggrr2.api.infobip.com/sms/2/text/advanced>  <https://qggrr2.api.infobip.com/sms/1/text/query> |
| *Send Bulk SMS using Vodafone infobip Server to designated multiple mobile number* | */BuldSMS* |  |
| *4* | *Send Email to recipients* | *Notification Service* | *This platform service will receive message from Kafka and send single of bulk email to email recipients using InfoBip Emal featue* | */Email*  */BulkEmail* |  |
| *5* | *Read OCR from PDF/images* | *OCR Service* | *This service to be provided by Fin Tech as third party service and possible to have multiple resource endpoints for multiple set of documents* | *For Eg. (Sample only)*  */PANNumber*  */Signture*  */AadharNumber*  */ConsentForm*  */Signature* |  |
| *6* | *Verify PAN Card* | *Verification Service* | *This service end point to invoke API from Fin Tech to verify PAN Card Number* | */PAN/{ID}* |  |
| *7* | *Back Ground Check Verification* | *This service end point is to verify back ground of given persion by invoking Fin Tech API(To be confirmed)* | */BackgroundCheck/{id}* |  |
| *8* | *Verify Aadhar Card* | *This service end point is to verify Aadhar Card for a given Aadhar number by invoking Fin Tech API(To be confirmed)* | */AadharNumber/{ID}* |  |
| *8* | *Payment Vateway Service to do Payment using BillDesk* | *Payment Service* | *This service Endpoint is to invoke BillDesk API to do payment due* | */Payment/{ID}* | ***Prod or Dev BillDesk URL?***  *To Check the status of trnsaction*  <https://www.billdesk.com/pgidsk/PGIQueryController> |
|  |  |  |  |  |  |

## **Data Architecture**

*<Cover the following points – What are the different data domain models for these microservices? Which databases will be implemented? What are the key data entities and relationships for microservices?>*

### Data Model

Attached is data model details.

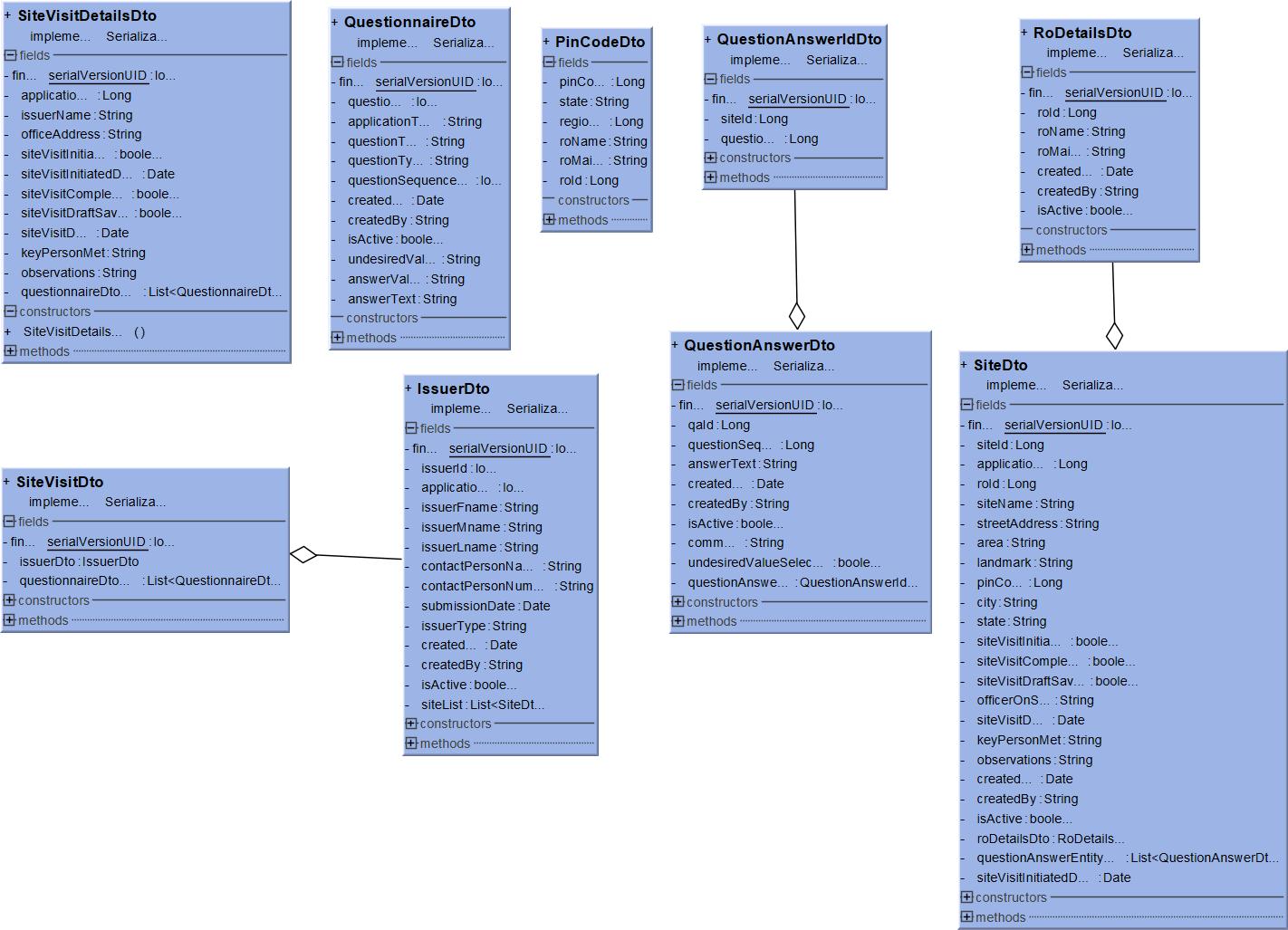


### Data Migration

*<Cover the following points – Database names, Schema and Tables of the Existing Applications which need to be migrated to the Target Application Database Schema/Objects. Specify the table/column mappings details etc. Also specify the file format in which existing application team to prepare the extract. >*

## **Object Model**

Attached is Object model details.



## **Data Validations**

*<Cover the following points – Refer user story/Epic Functional Flow output of Garage and specify data validations design>*

## **Business Rules**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr. No.** | **Parameter checked** | **Condition** | **Value** | **Source of Parameter** |
| 1 | Issue Size | Greater than | 250 cr | In-Principle Approval Application data |
| 2 | Company type | Equals | Government Company | In-Principle Approval Application data |
| 3 | Promoter Status | Equals | Listed Company | In-Principle Approval Application data |
| 4 | Promoter Status | Equals | Promoter of Listed Company | In-Principle Approval Application data |

Drools is a Business Rules Management System (BRMS) solution. It provides a core Business Rules Engine (BRE), a web authoring and rules management application (Drools Workbench). Business rules will be configured in the drools and at runtime API call will be used to check the business validation.

It is in progress. Will be updating how API will fetch the drool business condition

## **Sequence Diagrams for Key Business Process Flows of the Journeys**

*<Cover the following points – For the key process flows of this journey, how are the technical components (microservices/objects/other software components) across different architectural layers interacting with each other? >*

## **Logging**

Refer Common Cross Cutting Concerns Design for Logging

## **Error and Exception Handling**

Refer Common Cross Cutting Concerns Design for Exception and Error Handling

## **Auditing**

Refer Common Cross Cutting Concerns Design for Auditing

## **Caching**

Refer Common Cross Cutting Concerns Design for Caching

## **Integration Interfaces**

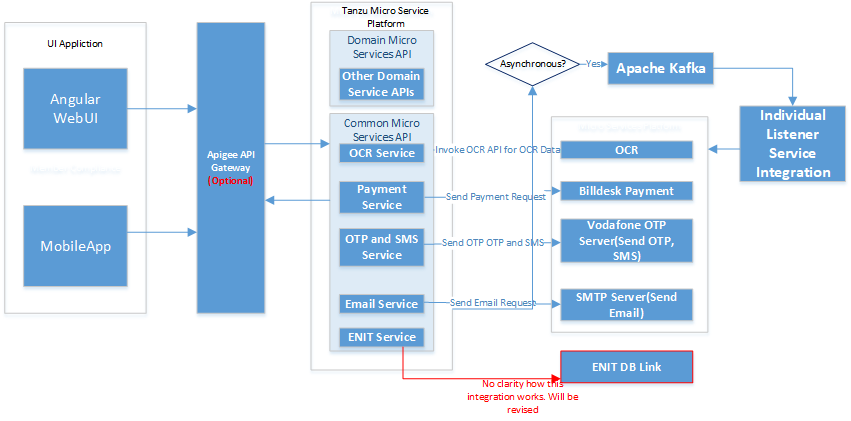
*<Cover the following points –*

*For this journey, how are the different microservices/objects interacting with each other and NSE Core Systems (Trading, Clearing, Surveillance, Enterprise Systems/Apps, C2N apps, DWH)?*

*Which are the existing interfaces which will have to be retired?*

*Which are the new interfaces which will have to be newly built?>*

### Integration Flows

****

*<Cover the following points – Plot all the components (microservices/objects/other software components) across different architectural layers. For each step in the reimagined journey process, plot the step-by-step numbered interaction between these components. For each interaction, also mention the type of connection, data format, etc.>*

### Interface List

There is only one interface listed for Member compliance and having no clarity. This will be discussed and updated in next version.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Sr. No.*** | ***System*** | ***Market Segment*** | ***Incoming/ Outgoing*** | ***Push / Pull*** | ***Data Entity Exchanged*** | ***Interface Protocol*** | ***Message Format*** | ***Sync/ Async*** | ***Batch/ Real-time*** | ***Business event/ Time of invocation*** | ***Peak Rate*** | ***If multiple records, preak count*** | ***Data Size Exchanged*** | ***Description (Including other important details from interface)*** |
| 1 | ENIT |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

### Integration with ENIT

Need to discuss and get more clarity

### Third Party Integration

#### Payment Gateway Integration – BillDesk

#### Integration – PAN Verification

Future Step – Discussion pending from requirement point of view

#### Integration – OCR

Future Step – Discussion pending from requirement point of view

#### Integration – (OTP, SMS)

API based Integration to send SMS and find API List below to Send SMS. OTP to be generated through java program logic and SMS to be sent along with OTP.. Refer to above section for detail process flow and sequence diagram

##### API to Send SMS

<https://dev.infobip.com/#programmable-communications/sms>

##### Integration – Send Email Service

SMTP Server Details to be furnished

## **APIs Exposed For API Channel Interface**

*<Cover the following points – Refer user story/Epic Functional Flow output of Garage and specify APIs exposed and its detailed end points signatures for consumption of stakeholders e.g. Trading Members, Clearing Members etc>*

Yet to determine

## **Security Design**

**Discussion in progress and to be finalized. Update this section later.**

### Transport Security

### Data Security

**How to secure data with encryption, vormetric, tool usage, data masking etc..**

### Authentication

#### Authentication for Web

#### Authentication for API Gateway

#### Authentication for Micro Services

### Authorization

#### Authorization for Web

#### Authorization for API Gateway

#### Authorization for Micro Services