

UltraTech Castle

Castle Microservices

API Documentation

1. Regulatory APIs

Api collection for Regulatory details:

a. Regulatory Land Records

Description	API to fetch the landRecord offices. If pincode is not provided in the request, then using the JWT token, it fetches the pincode as per the plot address and using that pincode, fetches the land records data for regulatory purposes
Method	GET
URL	/regulatoryInfo/landrecord
Header	Authorization: [JWT Token]

Request :

```
{
    "pincode": Number
}
```

Response :

```
{
    "status": "success",
    "requestId": null,
    "result": {
        "offices": [{
            "name": String,
            "workingHours": String,
            "address": String,
            "latitude": Float,
            "longitude": Float
        }],
        "documents": [{
            "name": String,
            "description": String,
            "procedureToGetDocument": [{
                "stepNumber": String,
                "stepTitle": String,
                "stepDescription": String
            }],
            "sourceLink": String,
            "lastUpdated": DateTime
        }],
    }
}
```

RequestParameter :

Parameter	Type	Description	isMandatory
pincode	number	Pincode for which landrecords to be fetched	No

b. Construction Details

Description	Fetches the construction details according to the pincode provide OR the pincode if the Plot
Method	GET
URL	/regulatoryInfo/constructionDetails
Headers	Authorization: [JWT Token]

Request:

```
{
    "pincode": Number
}
```

Response :

```
{
    "offices": [{
        "name": String,
        "address": String,
        "latitude": Float,
        "longitude": Float,
        "workingHours": String,
        "otherOffice": Boolean
    }],
    "documents": [{
        "name": String,
        "description": String,
        "procedureToGetDocument": [{
            "stepNumber": String,
            "stepTitle": String,
            "stepDescription": String
        }],
        "sourceLink": String,
        "lastUpdated": Date
    }],
    "processForVerification": [{
        "stepNumber": String,
        "stepTitle": String,
        "stepDescription": String
    }],
    "construction": {
        "setbackFront": Number,
        "setbackRear": Number,
        "setbackSide": Number,
        "far": Number,
        "gcp": Number
    },
    "areaTypes": String Array
}
```

Request Parameter :

Parameter	Type	Description	isMandatory
pincode	number	Pincode for which construction details are to be fetched	No

c. Post Regulatory Info Feedback

Description	API to post the feedback and concerns for the regulatory info
Method	POST
URL	/regulatoryInfo/feedback
Headers	Authorization: [JWT Token]

Request:

```
{
  "feedback": String,
  "concerns": String Array
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "message": "Feedback stored"
  }
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
feedback	String	Feedback Text entered by the user	Yes
concerns	String Array	Area of concerns provided as feedback like Land record office details, Documents information, etc	Yes

2. Design Library APIs

APIs for Get Layout, Get 3d and VR service request

a. Design Notification Service

Description	API used by KTR for pushing notification in the application
Method	POST
URL	/userOnboardingService/designNotification
Headers	Authorization: [JWT Token]

Request:

```
{
  "requestType": String,
  "userId": Number,
  "userServiceId": Number,
  "partnerName": String,
  "partnerNumber": String,
  "dateTime": String
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": "Notification Sent"
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
requestType	String	Type of service request (projectEstimate, VR, 3D, etc)	Yes
userId	Number	Id of the user who had raised the service request	Yes
userServiceId	Number	Id of the service request	Yes
partnerName	String	Name of the partner who has serviced the request of user	Yes
partnerNumber	String	Contact number of the partner	Yes
dateTime	String	Date and time at which the service request was done	Yes

b. Approved plan

Description	API used to fetch the approved plan for a service request
Method	GET
URL	/designLibrary/planApproved
Headers	Authorization: [JWT Token]

Request:

```
{
  "appld": Number,
  "requestId": Number
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "serviceRequestId": Number,
    "planDetails": [{
      "planId": Number,
      "planName": String,
      "planImageUrl": String,
      "planPdfUrl": String,
      "planDescription": String,
      "planStatus": String,
    }]
  }
}
```

```

    "isApproved": Boolean
  }
}

```

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes
requestId	Number	Id of the service request against which the approved plan is sought.	Yes

Response Parameters

Parameter	Type	Description
serviceRequestId	Number	Id for the service request
planDetails	JSON Array	Array of plan details.
planId	Number	Id of the plan
planName	String	Name of the plan
planImageUrl	String	URL of the image for the plan
planPdfUrl	String	URL of the pdf for the plan
planDescription	String	Description of the plan
planStatus	String	Current Status of the plan
isApproved	Boolean	Whether the plan is approved or not

c. Update Design request

Description	API used to update the existing design service request
Method	POST
URL	/designLibrary/postUpdateDesignLayoutPlan
Headers	Authorization: [JWT Token]

Request:

```

{
  "appld": Number,
  "status": String,
  "serviceRequestId": Number,
  "stage": Number,
  "uploadedImages": String Array
}

```

Response :

```

{
  "status": "success",
  "requestId": null,
}

```

```
    "result": "Design Request saved"
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes
status	String	Status to be updated (SUCCESSFUL or CANCELLED)	Yes
serviceRequestId	Number	Id for the service request	Yes
stage	String	To specify the stage for which the design is being updated	Yes
uploadedImages	String Array	Array of urls for each image associated with the request	Yes

d. Post Design Change Request

Description	API used to request a change a design service request which is already provided
Method	POST
URL	/designLibrary/requestAttempts
Headers	Authorization: [JWT Token]

Request:

```
{
  "appld": Number,
  "reason": String,
  "serviceRequestId": Number,
  "stage": String
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": "Requested for change"
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes
reason	String	Reason provided by the user for requesting a change	Yes
serviceRequestId	Number	Id for the service request	Yes
stage	String	To specify the stage for which the design is being updated	Yes

e. Get Design Change Request

Description	API used to fetch a design change request raised by a user
Method	GET
URL	/designLibrary/requestAttempts
Headers	Authorization: [JWT Token]

Request:

```
{
  "serviceRequestId": Number
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "attemptLeft": Number,
    "requests": [{
      "requestStatus": String,
```



```

    "requestDate": DateTime
  }}
}

```

Request Parameters :

Parameter	Type	Description	isMandatory
serviceRequestId	Number	Id for the service request	Yes

Response Parameters

Parameter	Type	Description
attemptLeft	Number	Number of attempts left for the design request for the user
requests	JSON Array	Array of request details
requestStatus	String	Current Status of the design change request
requestDate	DateTime	Date and Time of the design change request

f. Layout Design Request

Description	API used to raise a design layout request by the user
Method	POST
URL	/designLibrary/postSketchTo2DDesign
Headers	Authorization: [JWT Token]

Request:

```

{
  "appld": Number,
  "familyDetails": {
    "adults": Number,
    "infants": Number,
    "kids": Number,
    "memberCount": Number,
    "seniorCitizens": Number
  },
  "houseDesignRequestStage": String,
  "houseDesignRequestType": String,
  "houseDetails": {
    "budget": String,
    "floorCount": String,
    "rooms": String
  },
  "planDescription": String,
  "planName": String,
  "plotDetails": {
    "plotAddress": String,
    "plotArea": String
  },
  "sketches": {
    String <FileName>: String <base64String>
  },
  "siteImages": {
    String <FileName>: String <base64String>
  }
}

```

```
}
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "message": "Request is Saved",
    "serviceRequestId": Number,
    "designRequestId": Number
  }
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes
familyDetails	Json	Details of the family	No
adults	Number	Number of adults in the family	No
infants	Number	Number of infants in the family	No
kids	Number	Number of kids in the family	No
memberCount	Number	Total number of people in the family	No
seniorCitizens	Number	Total number of senior citizens in the family	No
houseDesignRequest Stage	String	Stage for which design is requested. Only 2D is considered valid in this call	Yes
houseDesignRequest Type	String	To identify whether it is a custom design layout as input or the one provided by Ultratech (CUSTOM or UTECH)	Yes
houseDetails	Json	Details of the house for which request is raised	No
budget	String	Budget of the house (eg: 1 Lakh)	No
floorCount	String	No of Floors (Eg: Ground Floor)	No
rooms	String	No of rooms (Eg: 1 BHK)	No
planDescription	String	Description of the plan	Yes
planName	String	Name of the plan	Yes
plotDetails	Json	Details of the plot	Yes
plotAddress	String	Address of the plot	Yes
plotArea	String	Area of the plot	Yes
sketches	Json	Json object containing filename as key and base64 string denoting the image as value	No
sitelmages	Json	Json object containing filename as key and base64 string denoting the image as value	No

Response Parameters :

Parameter	Type	Description
serviceRequestId	Number	Id for the service request

designRequestId	Number	Id for the design request for the above service request id
-----------------	--------	--

g. 3D Design Request

Description	API used to raise a design 3D request by the user
Method	POST
URL	/designLibrary/post2DTo3DDesign
Headers	Authorization: [JWT Token]

Request:

```
{
  "appld": Number,
  "familyDetails": {
    "adults": Number,
    "infants": Number,
    "kids": Number,
    "memberCount": Number,
    "seniorCitizens": Number
  },
  "houseDesignRequestStage": String,
  "houseDesignRequestType": String,
  "houseDetails": {
    "budget": String,
    "floorCount": String,
    "rooms": String
  },
  "planDescription": String,
  "planName": String,
  "plotDetails": {
    "plotAddress": String,
    "plotArea": String
  },
  "design2dImages": {
    String <FileName>: String <base64String>
  }
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "message": "Request is Saved",
    "serviceRequestId": Number,
    "designRequestId": Number
  }
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes
familyDetails	Json	Details of the family	No
adults	Number	Number of adults in the family	No

infants	Number	Number of infants in the family	No
kids	Number	Number of kids in the family	No
memberCount	Number	Total number of people in the family	No
seniorCitizens	Number	Total number of senior citizens in the family	No
houseDesignRequestStage	String	Stage for which design is requested. Only 3D is considered valid in this call	Yes
houseDesignRequestType	String	To identify whether it is a custom design layout as input or the one provided by Ultratech (CUSTOM or UTECH)	Yes
houseDetails	Json	Details of the house for which request is raised	No
budget	String	Budget of the house (eg: 1 Lakh)	No
floorCount	String	No of Floors (Eg: Ground Floor)	No
rooms	String	No of rooms (Eg: 1 BHK)	No
planDescription	String	Description of the plan	Yes
planName	String	Name of the plan	Yes
plotDetails	Json	Details of the plot	Yes
plotAddress	String	Address of the plot	Yes
plotArea	String	Area of the plot	Yes
design2DImages	Json	Json object containing filename as key and base64 string denoting the image as value	No

Response Parameters :

Parameter	Type	Description
serviceRequestId	Number	Id for the service request
designRequestId	Number	Id for the design request for the above service request id

h. Schedule VR Visit Request

Description	API to schedule the VR visit for a design
Method	POST
URL	/designLibrary/postScheduleVRVisit
Headers	Authorization: [JWT Token]

Request:

```
{
  "appld": Number,
  "familyDetails": {
    "adults": Number,
    "infants": Number,
    "kids": Number,
    "memberCount": Number,
    "seniorCitizens": Number
  },
  "houseDesignRequestStage": String,
```

```

    "houseDesignRequestType": String,
    "houseDetails": {
        "budget": String,
        "floorCount": String,
        "rooms": String
    },
    "residentialAddress": String,
    "vrServiceVisitDate": String,
    "vrServiceVisitTime": String,
    "plotDetails": {
        "plotAddress": String,
        "plotArea": String
    }
}

```

Response :

```

{
    "status": "success",
    "requestId": null,
    "result": {
        "message": "Request is Saved",
        "serviceRequestId": Number,
        "designRequestId": Number
    }
}

```

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes
familyDetails	Json	Details of the family	No
adults	Number	Number of adults in the family	No
infants	Number	Number of infants in the family	No
kids	Number	Number of kids in the family	No
memberCount	Number	Total number of people in the family	No
seniorCitizens	Number	Total number of senior citizens in the family	No
houseDesignRequest Stage	String	Stage for which design is requested. Only VR is considered valid in this call	Yes
houseDesignRequest Type	String	To identify whether it is a custom design layout as input or the one provided by Ultratech (CUSTOM or UTECH)	Yes
houseDetails	Json	Details of the house for which request is raised	No
budget	String	Budget of the house (eg: 1 Lakh)	No
floorCount	String	No of Floors (Eg: Ground Floor)	No
rooms	String	No of rooms (Eg: 1 BHK)	No
plotDetails	Json	Details of the plot	No
plotAddress	String	Address of the plot	No
plotArea	String	Area of the plot	No
residentialAddress	String	Residential address provided by the user	Yes
vrServiceVisitDate	Date	Date for scheduling VR Visit in YYYY-MM-DD format	Yes

vrServiceVisitTime	String	Time interval in hours for scheduling the VR Visit (eg: 9am-7pm)	Yes
--------------------	--------	--	-----

Response Parameters :

Parameter	Type	Description
serviceRequestId	Number	Id for the service request
designRequestId	Number	Id for the design request for the above service request id

i. Schedule 3d Visit Request

Description	API to schedule the 3D visit for a design
Method	POST
URL	/designLibrary/postSchedule3DService
Headers	Authorization: [JWT Token]

Request:

```
{
  "appld": Number,
  "familyDetails": {
    "adults": Number,
    "infants": Number,
    "kids": Number,
    "memberCount": Number,
    "seniorCitizens": Number
  },
  "houseDesignRequestStage": String,
  "houseDesignRequestType": String,
  "houseDetails": {
    "budget": String,
    "floorCount": String,
    "rooms": String
  },
  "residentialAddress": String,
  "designResourcePath": String,
  "plotDetails": {
    "plotAddress": String,
    "plotArea": String
  }
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "message": "Request is Saved",
    "serviceRequestId": Number,
    "designRequestId": Number
  }
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
-----------	------	-------------	-------------

appld	Number	Id of the application from where the API request is raised	Yes
familyDetails	Json	Details of the family	No
adults	Number	Number of adults in the family	No
infants	Number	Number of infants in the family	No
kids	Number	Number of kids in the family	No
memberCount	Number	Total number of people in the family	No
seniorCitizens	Number	Total number of senior citizens in the family	No
houseDesignRequest Stage	String	Stage for which design is requested. Only VR is considered valid in this call	Yes
houseDesignRequest Type	String	To identify whether it is a custom design layout as input or the one provided by Ultratech (CUSTOM or UTECH)	Yes
houseDetails	Json	Details of the house for which request is raised	No
budget	String	Budget of the house (eg: 1 Lakh)	No
floorCount	String	No of Floors (Eg: Ground Floor)	No
rooms	String	No of rooms (Eg: 1 BHK)	No
plotDetails	Json	Details of the plot	No
plotAddress	String	Address of the plot	No
plotArea	String	Area of the plot	No
residentialAddress	String	Residential address provided by the user	No
designResourcePath	String	URL of the design image for which request is raised	Yes

Response Parameters :

Parameter	Type	Description
serviceRequestId	Number	Id for the service request
designRequestId	Number	Id for the design request for the above service request id

j. Design Callback Request

Description	API for requesting a callback for against a design request
Method	GET
URL	/support/requestCallback
Headers	Authorization: [JWT Token]

Request:

```
{
  "appld": Number,
  "reason": String,
  "serviceRequestId": Number
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "data": "Requested for callback"
  }
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes
reason	String	Reason for requesting the design callback request	Yes
serviceRequestId	Number	Id for the service request	Yes

3. Search APIs

Collection of search API used For IHB and partner app

a. Search History

Description	Fetches the search history of a user according to the search category mentioned in params. Otherwise fetches the search history of all categories
Method	GET
URL	/searchService/searchHistory
Headers	Authorization: [JWT Token]

Request:

```
{
  "type": String Array,
  "limit": Number,
  "offset": Number
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "totalRows": Number,
    "startIndex": Number,
    "data": [{
      "value": String,
      "type": String
    }]
  }
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
type	String	Type / category names for which the history is to be fetched	No

offset	Number	Starting index of the results	Yes
limit	Number	Total number of results to be returned from offset	Yes

Response Parameters :

Parameter	Type	Description
totalRows	Number	Total number of records in the result set
startIndex	Number	Starting index number of the record in current api call
data	JSON Array	Result set
type	String	Type / category name
value	String	Value of the search in history

b. Search API

Description	API used for searching documents and content on SOLR
Method	GET
URL	/searchService/search
Headers	Authorization: [JWT Token]

Request:

```
{
  "appld": Number,
  "limit": Number,
  "offset": Number,
  "query": [{
    "type": String,
    "value": String Array
  }]
}
```

Example:

```
{
  "appld": 1,
  "limit": 20,
  "offset": 0,
  "query": [{
    "type": "category",
    "value": ["material"]
  }, {
    "type": "location",
    "value": ["18.5552408,73.7715515"]
  }, {
    "type": "typeOfProjects",
    "value": ["residential"]
  }]
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
```

```

        "totalRows": Number,
        "startIndex": Number,
        "data": [{
            "filter": String,
            "data": [{
                "resourcePath": String,
                "imagePath": String,
                "name": String
            }],
        }],
    }
}

```

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes
offset	Number	Starting index of the results	Yes
limit	Number	Total number of results to be returned from offset	Yes
query	Json Array	Collective query where each object in the array is forms a request condition	Yes
type	String	Category over which search is to be done	Yes
value	String Array	Values for which search is to be done	Yes

Response Parameters :

Parameter	Type	Description
totalRows	Number	Total number of records in the result set
startIndex	Number	Starting index number of the record in current api call
data	JSON Array	Result set
filter	String	Filter ran on Solr
resourcePath	String	URL of the resource
imagePath	String	URL of the image of resource
name	String	Name of the resource

c. Experts search API

Description	API to fetch the expert details from AEM
Method	GET
URL	/expertDetails
Headers	Authorization: [JWT Token]

Request:

```

{
    "resourcePath": String
}

```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "avatar": String,
    "name": String,
    "experience": Number,
    "rating": Number,
    "website": String,
    "address": String,
    "businessStartTime": Number,
    "businessEndTime": Number,
    "education": String,
    "expertise": String Array,
    "projectType": String Array,
    "projects": String Array,
    "reviews": String Array,
    "phone": String,
    "isFavourite": Boolean
  }
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
resourcePath	String	URL of the resource path of expert in AEM	Yes

Response Parameters :

Parameter	Type	Description
avatar	Number	Image avatar URL of the expert
name	Number	Name of the expert
experience	JSON Array	Experience of the expert
rating	Number	Rating of the expert
website	String	Website of the expert
address	String	Address of the expert
businessStartTime	Number	Business Start time in 24hr format
businessEndTime	Number	Business End time in 24hr format
education	String	Education qualification of the expert
expertise	String Array	List of expertise
projectType	String Array	Type of projects worked on by the expert
projects	String Array	Projects worked on by the expert
reviews	String Array	Reviews of expert
phone	String	Phone number of expert
isFavourite	Boolean	True, if the expert fetched is marked as favourite. Else False

d. Search Suggestion

Description	API used to fetch the suggestions of the term according to the context in the parameter
-------------	---

Method	GET
URL	/searchService/suggest
Headers	Authorization: [JWT Token]

Request:

```
{
  "query": {
    "suggest": String,
    "context": String Array
  }
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": [{
    "term": String,
    "payload": String
  }]
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
query	Json	Query object for search filter	Yes
suggest	String	The keys/part of text to search on	Yes
context	String Array	List of contexts to search on. Like “Experts”, “Materials”. If not provided, search is done irrespective of context	No

Response Parameters

Parameter	Type	Description
term	String	Result string of the search
payload	String	Context over which the term was found

e. Design Elements Search

Description	API to search on design elements
Method	GET
URL	/search/designElements
Headers	Authorization: [JWT Token]

Request:

```
{
  "query": [{
    "category": String,
```

```

        "limit": Number,
        "offset": Number
    }}
}
Response :
{
    "status": "success",
    "requestId": null,
    "result": [{
        "filter": String,
        "data": [{
            "resourcePath": String,
            "imagePath": String,
            "name": String
        }],
        "totalRows": Number
    }]
}

```

Request Parameters :

Parameter	Type	Description	isMandatory
query	Json	List of filter objects; all of those will be ORed	Yes
category	String	The design element category to search on.(Grills, Doors, etc)	Yes
offset	Number	Starting index of the results	Yes
limit	Number	Total number of results to be returned from offset	Yes

Response Parameters

Parameter	Type	Description
filter	String	Category of the result object in the results array
data	Json Array	List of results for the filter
resourcePath	String	URL for the resource path of the design Element from AEM
imagePath	String	URL for the image path of the design Element from AEM
name	String	Name of the design element
totalRows	Number	Number of results for the category or filter

f. Store Details

Description	API for fetching the store details stored in AEM.
Method	GET
URL	/searchService/storeDetails
Headers	Authorization: [JWT Token]

Request:

```

{
    "resourcePath": String
}

```

Response :

```

{
  "status": "success",
  "requestId": null,
  "result": {
    "carousellImages": String Array,
    "name": String,
    "category": String,
    "rating": Number,
    "latitude": Float,
    "longitude": Float,
    "address": String,
    "firmType": String,
    "businessStartTime": String,
    "businessEndTime": String,
    "emailId": String,
    "valueAddedServices": String,
    "products": String Array,
    "reviews": [{
      "avatar": String,
      "reviewer": String,
      "rating": Number,
      "review": String
    }],
    "phone": String,
    "isFavourite": Boolean
  }
}

```

Request Parameters :

Parameter	Type	Description	isMandatory
resourcePath	String	URL for the resource path of the store	Yes

Response Parameters

Parameter	Type	Description
carousellImages	String Array	List of images returned for the store
name	String	Name of the store
category	String	Store's category
rating	Number	Store's rating
latitude	Float	Store's location latitude
longitude	Float	Store's location longitude
address	String	Store's address
firmType	String	Store's firm type
businessStartTime	Number	Store's start time in 24 hr format
businessEndTime	Number	Store's end time in 24 hr format
emailId	String	Store's emailId
valueAddedServices	String Array	Value added services provided by the Store
products	String Array	List of products in the store
reviews	Json Array	List of Reviews object
avatar	String	Reviewer's avatar

reviewer	String	Reviewer's name
rating	Number	Reviewer's rating for the store
review	String	Review text
phone	String	Store's phone number

4. IHB User Profile APIs

Collection of APIs For IHB profile

a. Family Members API

Description	API to fetch the user's family members details
Method	GET
URL	/userService/getfamilymembers
Headers	Authorization: [JWT Token]

Request:

None. User is identified using the JWT token and the family details are returned for that user

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "memberCount": Number,
    "infant": Number,
    "adults": Number,
    "seniorCitizen": Number,
    "children": Number
  }
}
```

Response Parameters :

Parameter	Type	Description
adults	Number	Number of adults in the family
infants	Number	Number of infants in the family
children	Number	Number of kids in the family
memberCount	Number	Total number of people in the family
seniorCitizens	Number	Total number of senior citizens in the family

b. Profile Picture Upload API

Description	API to upload the user's profile picture
Method	POST
URL	/userService/profilePic/upload
Headers	Authorization: [JWT Token]

Request:

```
{
    "fileName": String,
    "base64String": String
}
```

Response :

```
{
    "status": "success",
    "requestId": null,
    "result": "Profile Updated"
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
fileName	String	Type / category names for which the history is to be fetched	Yes
base64String	String	base64 string denoting the image data	Yes

c. User language update API

Description	API to Update the language for a IHB and partner.
Method	POST
URL	/userService/updateLanguage
Headers	Authorization: [JWT Token]

Request:

```
{
    "languageCode": String
}
```

Response :

```
{
    "status": "success",
    "requestId": null,
    "result": "language successfully updated."
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
languageCode	String	Language code to be updated for user (en,hi,gu,mr)	Yes

d. User house details update API

Description	API to Update the house details for a IHB
Method	POST
URL	/userService/updateHouseDetails
Headers	Authorization: [JWT Token]

Request:

```
{
  "housePurpose": String,
  "fundSource": String,
  "houseBudget": Float,
  "floorCount": String,
  "roomsCount": Number,
  "hasGarden": Boolean,
  "hasBasement": Boolean,
  "hasParking": Boolean,
  "parkingType": String,
  "twoWheelerCount": Number,
  "fourWheelerCount": Number
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": "Profile Updated."
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
housePurpose	String	Purpose of the house	No
fundSource	String	Source of fund for the house	No
floorCount	Number	Number of floors	No
roomCount	Number	Number of rooms	No
hasBasement	Boolean	If the basement is present or not	No
hasParking	Boolean	If parking is present or not	No
parkingType	String	Type of Parking in the house	No
twoWheelerCount	Number	Number of two wheelers vehicles	No
FourWheelerCount	Number	Number of four wheelers vehicles	No

e. User family details update API

Description	API to Update the family details for a IHB
Method	POST
URL	/userService/updatefamilymembers
Headers	Authorization: [JWT Token]

Request:

```
{
  "memberCount": 6,
  "infant": true,
  "adults": false,
  "seniorCitizen": true
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": "Profile Updated."
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
memberCount	Number	Total number of members in family	Yes
infant	Boolean	are infants in the family	No
adults	Boolean	are adults in the family	No
seniorCitizen	Boolean	are senior citizens in the family	No

f. User plot details update API

Description	API to Update the plot details for a IHB
Method	POST
URL	/userProfileService/updatePlotDetails
Headers	Authorization: [JWT Token]

Request:

```
{
  "plotNumber": String,
  "plotArea": String,
  "plotLatitude": Float,
  "plotLongitude": Float,
  "streetAddress": String,
  "city": String,
  "pin": String,
  "landmark": String
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": "Updated Plot Address."
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
plotNumber	String	Plot number	No
plotArea	String	Area of the plot with measurement unit (Sq ft, sq mt)	No
plotLatitude	Float	Latitude of the plot	No
plotLongitude	Float	Longitude of the plot	No
streetAddress	String	Address of the plot	No

city	String	Plot's city	No
pin	String	Pincode of the area of the plot	No
landmark	String	Landmark of the plot	No

g. User profile fetch API

Description	API to fetch profile details for a IHB
Method	GET
URL	/userService/getuserprofile
Headers	Authorization: [JWT Token]

Request:

None. User is identified using the JWT token and the profile details are returned for that user.

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "firstName": String,
    "lastName": String,
    "email": String,
    "dob": Date,
    "gender": String,
    "profilePicUrl": String,
    "profession": String,
    "minIncome": Number,
    "maxIncome": Number
  }
}
```

Response Parameters :

Parameter	Type	Description
firstName	String	First name of the user
lastName	String	Last name of the user
email	String	Email of the user
dob	Date	Date of birth of the user
gender	String	Gender of the user
profilePicUrl	String	Profile picture's url of the user
profession	String	Profession of the user
minIncome	Number	Minimum Income of the user
maxIncome	Number	Maximum Income of the user

h. User's address fetch API

Description	API to fetch address details for a IHB
Method	GET
URL	/userService/getuseraddress
Headers	Authorization: [JWT Token]

Request:

None. User is identified using the JWT token and the address details are returned for that user.

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "userId": Number,
    "flatNumber": String,
    "streetAddress": String,
    "area": String,
    "city": String,
    "pin": Number,
    "landmark": String,
    "state": String,
    "country": String,
    "latitude": Float,
    "longitude": Float
  }
}
```

Response Parameters :

Parameter	Type	Description
userId	Number	Id of the user
flatNumber	String	Flat Number of the user
streetAddress	String	Street address of the user
area	String	Area of the user
city	String	City of the user
pin	Number	Pincode of the user's location
landmark	String	Landmark of the user's location
state	String	User's state
country	String	User's country
latitude	Float	Latitude of the plot
longitude	Float	Longitude of the plot

i. User's profile completion percentage fetch API

Description	API to fetch the profile completion percentage of the user
Method	GET
URL	/userService/completionPercent
Headers	Authorization: [JWT Token]

Request:

None. User is identified using the JWT token and the completion percentages are returned for that user.

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "userProfile": Float,
    "family": Float,
    "houseDetails": Float,
    "totalPercent": Float
  }
}
```

Response Parameters :

Parameter	Type	Description
userProfile	Float	Profile Completion percentage
family	Float	Family Details Completion percentage
houseDetails	Float	House Details Completion percentage
totalPercent	Float	Total Profile Completion percentage

5. Material Provider APIs

Collection of APIs For Material Provider

a. Fetching Store Details

Description	API to fetch the store’s details
Method	GET
URL	/partnerProfile/dealer/store/details
Headers	Authorization: [JWT Token]

Request:

```
{
    "resourcePath": String
}
```

Response :

```
{
    "status": "success",
    "requestId": null,
    "result": {
        "storeName": String,
        "openHours": [{
            "day": String,
            "time": {
                "start": String,
                "end": String
            }
        }],
        "firmType": String,
        "gstNumber": String,
        "utclNumber": String,
        "establishmentYear": String,
        "godown": {
            "size": String,
            "unit": String,
            "structure": String
        },
        "shopSize": {
            "size": String,
            "unit": String
        },
        "storeImageBitmaps": {
            "frontViewImageBitmaps": [{
                "base64String": String
            }],
            "godownViewImageBitmaps": [{
                "base64String": String
            }]
        },
        "noTemporarySalesPerson": Number,
        "noPermanentSalesPerson": Number
    }
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
resourcePath	String	URL of the store identifying the store in AEM	Yes

Response Parameters :

Parameter	Type	Description
storeName	String	Name of the store
openHours	Json Array	List of hours when the store is open
day	String	Week days when the store remains open
time	Json	Time object for start and end time
time.start	String	Start time in HH:MM format
time.end	String	End time in HH:MM format
firmType	String	Type of firm
gstNumber	String	GST number of store
utclNumber	String	Ultratech Number of store
establishmentYear	String	Year in which store was established
godown	Json	Godown details
godown.size	String	Size of the godown
godown.unit	String	Unit of measurement of the size of godown
godown.structure	String	Structure of the godown
shopSize	Json	Shop Size Details
shopSize.size	String	Shop's size
shopSize.unit	String	Unit of measurement of shop's size
storeImageBitmaps	Json	Details of Images of store
frontViewImageBitmaps	Json Array	Array of objects having base64String as key and image in base 64 format representing store's front view images
godownViewImageBitmaps	Json Array	Array of objects having base64String as key and image in base 64 format representing godown images
noTemporarySalesPerson	Number	Total number of temporary sales persons
noPermanentSalesPerson	Number	Total number of permanent sales persons

b. Update Store Details

Description	API to update the store's details
Method	POST
URL	/partnerProfile/dealer/store/details
Headers	Authorization: [JWT Token]

Request:

```
{
  "storeName": String,
  "openHours": [{
    "day": String,
    "time": {
      "start": String,
      "end": String
    }
  }],
  "firmType": String,
  "gstNumber": String,
  "utclNumber": String,
  "establishmentYear": String,
  "godown": {
    "size": String,
    "unit": String,
    "structure": String
  },
  "shopSize": {
    "size": String,
    "unit": String
  },
  "storeImageBitmaps": {
    "frontViewImageBitmaps": [{
      "base64String": String
    }],
    "godownViewImageBitmaps": [{
      "base64String": String
    }]
  },
  "noTemporarySalesPerson": Number,
  "noPermanentSalesPerson": Number
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": "Store details saved successfully."
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
storeName	String	Name of the store	Yes
openHours	Json Array	List of hours when the store is open	No
day	String	Week days when the store remains open	No
time	Json	Time object for start and end time	No
time.start	String	Start time in HH:MM format	No
time.end	String	End time in HH:MM format	No
firmType	String	Type of firm	No
gstNumber	String	GST number of store	Yes
utclNumber	String	Ultratech Number of store	No
establishmentYear	String	Year in which store was established	Yes

godown	Json	Godown details	No
godown.size	String	Size of the godown	No
godown.unit	String	Unit of measurement of the size of godown	No
godown.structure	String	Structure of the godown	No
shopSize	Json	Shop Size Details	No
shopSize.size	String	Shop's size	No
shopSize.unit	String	Unit of measurement of shop's size	No
storeImageBitmaps	Json	Details of Images of store	Yes
frontViewImageBitmaps	Json Array	Array of objects having base64String as key and image in base 64 format representing store's front view images	Yes
godownViewImageBitmaps	Json Array	Array of objects having base64String as key and image in base 64 format representing godown images	No
noTemporarySalesPerson	Number	Total number of temporary sales persons	No
noPermanentSalesPerson	Number	Total number of permanent sales persons	No

c. Fetching Products in a store

Description	API used for fetching the product and services provided by material provider
Method	GET
URL	/partnerProfile/dealer/store/products
Headers	Authorization: [JWT Token]

Request:

None. Partner's user id is fetched from the token and it is used to fetch the products for the store

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "products": [{
      "name": String,
      "origin": String,
      "types": [{
        "name": String,
        "origin": String
      }]
    }]
  }
}
```

Response Parameters :

Parameter	Type	Description
-----------	------	-------------

products	Json Array	List of products
name	String	Name of the products like Cement, Steel etc
origin	String	Source of the product
types	Json Array	Variants of the product
types[].name	String	Name of the variant
types[].origin	String	Origin of the variant

d. Adding Products in a store

Description	API used for updating the products and services of the user
Method	POST
URL	/partnerProfile/dealer/store/products
Headers	Authorization: [JWT Token]

Request:

```
{
  "products": [{
    "name": String,
    "origin": String,
    "types": [{
      "name": String,
      "origin": String
    }]
  }],
  "appld": Number
}
```

Response :

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes
products	Json Array	List of products	Yes
name	String	Name of the products like Cement, Steel etc	Yes
origin	String	Source of the product	Yes
types	Json Array	Variants of the product	Yes
types[].name	String	Name of the variant	Yes
types[].origin	String	Origin of the variant	Yes

e. Updating address of a store

Description	API used for updating the address of the user's store
Method	POST
URL	/partnerProfile/dealer/store/address
Headers	Authorization: [JWT Token]

Request:

```
{  
  "type": String,  
  "entityNumber": Number,  
  "streetAddress": String,  
  "city": String,  
  "pin": String,  
  "landmark": String,  
  "taluka": String,  
  "district": String,  
  "state": String,  
  "country": String,  
  "isRuralArea": Boolean,  
  "latitude": String,  
  "longitude": String  
}
```

Response :

```
{  
  "status": "success",  
  "requestId": null,  
  "result": {  
    "message": "Submitted store address successfully"  
  }  
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
type	String	Type of address. "store" is allowed in this call only	Yes
entityNumber	Number	Id of the address	No
streetAddress	String	Street address	No
city	String	City	Yes
pin	String	Pincode	No
landmark	String	Landmark for the store's location	No
taluka	String	Taluka in which the store is present	No
district	String	Store location's district	No
state	String	Store location's state	No
country	String	Store location's country	No
isRuralArea	Boolean	If the store is in rural area	No
latitude	String	Store location's latitude	No
longitude	String	Store location's longitude	No

f. Fetching address of a store

Description	API used for fetching the address of the user's store
Method	GET
URL	/partnerProfile/dealer/store/address
Headers	Authorization: [JWT Token]

Request:

None. JWT token is used to identify the partner user and store's address is fetched for that user

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "type": String,
    "entityNumber": Number,
    "streetAddress": String,
    "city": String,
    "pin": String,
    "landmark": String,
    "taluka": String,
    "district": String,
    "state": String,
    "country": String,
    "isRuralArea": Boolean,
    "latitude": String,
    "longitude": String
  }
}
```

Response Parameters :

Parameter	Type	Description
type	String	Type of address. "store" is allowed in this call only
entityNumber	Number	Id of the address
streetAddress	String	Street address
city	String	City
pin	String	Pincode
landmark	String	Landmark for the store's location
taluka	String	Taluka in which the store is present
district	String	Store location's district
state	String	Store location's state
country	String	Store location's country
isRuralArea	Boolean	If the store is in rural area
latitude	String	Store location's latitude
longitude	String	Store location's longitude

g. Updating services offered by a store

Description	API used for updating the services offered by the user's store
Method	POST
URL	/partnerProfile/dealer/store/offeredServices
Headers	Authorization: [JWT Token]

Request:

```
{
  "appld": Number,
  "servicesOffered": [{
    "name": String,
    "origin": String,
    "isAlreadyAdded": Boolean,
    "isSelected": Boolean
  }]
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "message": "Services offered saved successfully."
  }
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes
serviceOffered	Json Array	List of services offered by the store	Yes
isAlreadyAdded	Boolean	Is the service already added and has to be updated	Yes
isSelected	Boolean	is the service selected	Yes
name	String	Name of the service	Yes
origin	String	Service added by Utec or Custom	Yes

6. Onboarding APIs

Collection of APIs For User Onboarding

a. OTP Generation

Description	API to generate OTP
Method	POST
URL	/userOnboardingService/generateOTP
Headers	None.

Request:

```
{
  "appld": Number,
  "mobileNumber": String
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "message": "Otp sent succesfully",
    "isExistingUser": true
  }
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes
mobileNumber	String	Mobile number of the user to generate otp	Yes

b. OTP Verification

Description	API to verify OTP with the one generated and the one entered by the user. Responds accessToken and refreshToken in response header if OTP is valid. Max retries: 3
Method	POST
URL	/userOnboardingService/verifyOtp
Headers	None

Request:

```
{
  "appld": Number,
  "mobileNumber": String,
  "language": String,
```

```
"role": String,  
"otp": String  
}
```

Response :

```
{  
  "status": "success",  
  "requestId": null,  
  "result": {  
    "otpVerified": Boolean,  
    "firstTimeUser": Boolean  
  }  
}
```

Response Headers :

```
{  
  "Authorization": String,  
  "refreshToken": String  
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
appId	Number	Id of the application from where the API request is raised	Yes
mobileNumber	String	Mobile number of the user to generate otp	Yes
language	String	Language selected by the user in languageCode(en, hi, etc)	Yes
role	String	Whether IHB or Partner	Yes
otp	String	OTP received by the user	Yes

Response Parameters :

Parameter	Type	Description
otpVerified	Boolean	Whether the otp is correct or not.
firstTimeUser	Boolean	Whether the user has logged in first time or not

Response Headers :

Parameter	Type	Description
Authorization	String	Authorization header contains access token. This is used as Authorization header in subsequent API calls
refreshToken	String	Refresh token is used to refresh the Authorization token in case it expires

c. Renew Access Token

Description	API used for renewing access token using the refresh token
Method	POST
URL	/renewToken

Headers	refreshToken: [Refresh Token]
----------------	-------------------------------

Request:

None. Refresh token is validated and the access token is regenerated and provided in the response header

Response :

```
{
    "status": "success",
    "requestId": null
}
```

Response Headers :

```
{
    "Authorization": String,
    "refreshToken": String
}
```

Response Headers :

Parameter	Type	Description
Authorization	String	Authorization header contains access token. This is used as Authorization header in subsequent API calls
refreshToken	String	Refresh token is used to refresh the Authorization token in case it expires

7. User Favourites and Setting APIs

a. Notification Preference Fetch API

Description	API to fetch the notification preference of the user
Method	GET
URL	/userService/getPreferences
Headers	Authorization: [JWT Token]

Request:

None. User is identified using the JWT Token and then the user's preferences is fetched.

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "customAlerts": Boolean,
    "pushNotificationsEnabled": Boolean,
    "smsEnabled": Boolean,
    "emailEnabled": Boolean,
    "phoneCallEnabled": Boolean
  }
}
```

Response Parameters :

Parameter	Type	Description
customAlerts	Boolean	Whether custom alerts are enabled or not
pushNotificationsEnabled	Boolean	Whether push notifications are enabled or not
smsEnabled	Boolean	Whether SMSes are enabled or not
emailEnabled	Boolean	Whether emails are enabled or not
phoneCallEnabled	Boolean	Whether phone call notifications are enabled or not

b. Notification Preference Update API

Description	API to update the notification preference of the user
Method	POST
URL	/userService/addPreferences
Headers	Authorization: [JWT Token]

Request:

```
{
  "customAlerts": Boolean,
  "pushNotificationsEnabled": Boolean,
```

```

    "smsEnabled": Boolean,
    "emailEnabled": Boolean,
    "phoneCallEnabled": Boolean
}

```

Response :

```

{
    "status": "success",
    "requestId": null,
    "result": "user alert preferences successfully updated."
}

```

Request Parameters :

Parameter	Type	Description	isMandatory
customAlerts	Boolean	Whether custom alerts are to be enabled or not	Yes
pushNotificationsEnabled	Boolean	Whether push notifications are to be enabled or not	Yes
smsEnabled	Boolean	Whether SMSes are to be enabled or not	Yes
emailEnabled	Boolean	Whether emails to be are enabled or not	Yes
phoneCallEnabled	Boolean	Whether phone call notifications to be are enabled or not	Yes

c. Remove from favourite API

Description	API to remove a resource from favourites
Method	POST
URL	/userProfileService/removeFavourite
Headers	Authorization: [JWT Token]

Request:

```

{
    "favouriteType": String,
    "favouriteResourcePath": String
}

```

Response :

```

{
    "status": "success",
    "requestId": null,
    "result": "user favourite successfully removed."
}

```

Request Parameters :

Parameter	Type	Description	isMandatory
favouriteType	String	Identifying the type of resource (Grills, doors, articles)	Yes
favouriteResourcePath	String	Resource path's URL	Yes

d. Fetch favourites API

Description	API to fetch favourite resources for a user
Method	GET
URL	/userService/getFavourite
Headers	Authorization: [JWT Token]

Request:

```
{
  "favouriteType": String,
  "limit": Number,
  "offset": Number,
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": [{
    "id": Number,
    "type": String,
    "resourcePath": String,
    "previewImage": String,
    "title": String,
    "subtitle": String,
    "description": String,
    "latitude": String,
    "longitude": String,
    "tags": String Array,
    "rating": Number,
    "price": String,
    "area": String,
    "businessStartTime": String,
    "businessEndTime": String,
    "storeAddress": String,
    "mobNumber": String
  }]
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
favouriteType	String	Identifying the type of resource (Grills, doors, articles)	Yes
offset	Number	Starting index of the results	Yes
limit	Number	Total number of results to be returned from offset	Yes

Response Parameters :

Parameter	Type	Description
id	String	id of the favourite resource

type	String	Type of the resource(Architect, Store, Grill, etc)
resourcePath	String	URL for the resource path of the design Element from AEM
previewImage	String	URL for the image path of the design Element from AEM
title	String	Title of the resource
subtitle	String	Subtitle, if any, of the resource
description	String	Description of the resource
latitude	String	Latitude of the location of the resource
longitude	String	Longitude of the location of the resource
tags	String Array	List of speciality tags associated with the resource
rating	String	Rating of the resource
price	String	Price of the resource, if applicable
area	String	Area of the resource, if applicable
businessStartTime	String	Business Start time of the resource, if applicable
businessEndTime	String	Business End time of the resource, if applicable
storeAddress	String	Store's address of the resource, if applicable
mobNumber	String	Mobile Number of the resource, if applicable

e. Fetch favourite Count API

Description	API to fetch favourite count per resource type for a user
Method	GET
URL	/userProfileService/getFavouriteCount
Headers	Authorization: [JWT Token]

Request:

None. JWT Token is used to identify the user.

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": [{
    "favouriteType": String,
    "count": Number
  }]
}
```

Response Parameters :

Parameter	Type	Description
favouriteType	String	Type of the resource(Architect, Store, Grill, etc)

count	Number	Number of counts on the resource type
-------	--------	---------------------------------------

f. Add a resource as favourite for a user

Description	API to add a resource as favourite for a user
Method	POST
URL	/userService/addFavourite
Headers	Authorization: [JWT Token]

Request:

```
{
  "favouriteType": String,
  "favouriteResourcePath": String
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": "user favourite added."
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
favouriteType	String	Identifying the type of resource (Grills, doors, articles)	Yes
favouriteResourcePath	String	Resource path's URL	Yes

8. Budgeting APIs

a. Estimate API

Description	API to fetch the budget estimate for the user
Method	GET
URL	/estimatorService/getEstimate
Headers	Authorization: [JWT Token]

Request:

None. User is identified using the JWT Token and then the user’s budget estimate is fetched.

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "minPrice": Number,
    "maxPrice": Number,
    "stdEstimateType": String,
    "carpetArea": Number,
    "floorCount": String,
    "material": {
      "cement": {
        "packageType":String
      },
      "steel": {
        "packageType": String
      },
      "structure": {
        "packageType": String
      }
    },
    "finishing": {
      "flooring": {
        "packageType": String
      },
      "windowsDoors": {
        "packageType": String
      },
      "electrical": {
        "packageType": String
      },
      "painting": {
        "packageType": String
      },
      "bathroomFixtures": {
        "packageType": String
      },
      "plumbing": {
        "packageType": String
      }
    }
  }
}
```

```

    },
    "kitchen": {
        "isModular": Boolean
    },
    "parking": {
        "isActive": Boolean,
        "twoWheelerCount": Number,
        "fourWheelerCount": Number
    }
}
}

```

Response Parameters :

Parameter	Type	Description
minPrice	Number	Minimum Price in the estimate
maxPrice	Number	Maximum Price in the estimate
stdEstimateType	String	Standard estimate type
carpetArea	Number	Carpet Area in the estimate
floorCount	String	Number of floors
material	Json	Details about the materials
cement	Json	Json containing the package type for cement
steel	Json	Json containing the package type for steel used
structure	Json	Json containing the package type for structures
finishing	Json	Details about the finishing
flooring	Json	Json containing the package type for flooring used
windowsDoors	Json	Json containing the package type for windows and doors
electrical	Json	Json containing the package type for electrical fixtures
painting	Json	Json containing the package type for painting
bathroomFixtures	Json	Json containing the package type for bathroom fixtures
plumbing	Json	Json containing the package type for plumbing
kitchen	Json	Details about the kitchen
isModular	Boolean	Is the kitchen modular or not
parking	Json	Details about the parking
isActive	Boolean	Is parking considered or not
twoWheelerCount	Number	Total number of 2 wheeler vehicles
fourWheelerCount	Number	Total number of 4 wheeler vehicles

b. Budget Details API

Description	API to fetch the rate of commodities from the AEM
Method	GET

URL	/estimatorService/getBudgetDetails
Headers	Authorization: [JWT Token]

Request:
None.

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "data": {
      "material": {
        "cement": {
          "economical": {
            "metrics": String,
            "quantityMultiplier": Float,
            "quantity": Number,
            "price": Number
          },
          "medium": {
            "metrics": String,
            "quantityMultiplier": Float,
            "quantity": Number,
            "price": Number
          },
          "premium": {
            "metrics": String,
            "quantityMultiplier": Float,
            "quantity": Number,
            "price": Number
          }
        },
        "steel": {
          "economical": {
            "metrics": String,
            "quantityMultiplier": Float,
            "quantity": Number,
            "price": Number
          },
          "medium": {
            "metrics": String,
            "quantityMultiplier": Float,
            "quantity": Number,
            "price": Number
          },
          "premium": {
            "metrics": String,
            "quantityMultiplier": Float,
            "quantity": Number,
            "price": Number
          }
        }
      },
      "structure": {
        "economical": {
          "metrics": String,
          "quantityMultiplier": Float,
          "quantity": Number,
          "price": Number
        },
        "medium": {
```

```

        "metrics": String,
        "quantityMultiplier": Float,
        "quantity": Number,
        "price": Number
    },
    "premium": {
        "metrics": String,
        "quantityMultiplier": Float,
        "quantity": Number,
        "price": Number
    }
},
"finishing": {
    "flooring": {
        "economical": {
            "metrics": String,
            "quantityMultiplier": Float,
            "quantity": Number,
            "price": Number
        },
        "medium": {
            "metrics": String,
            "quantityMultiplier": Float,
            "quantity": Number,
            "price": Number
        },
        "premium": {
            "metrics": String,
            "quantityMultiplier": Float,
            "quantity": Number,
            "price": Number
        }
    },
    "windowsDoors": {
        "economical": {
            "metrics": String,
            "quantityMultiplier": Float,
            "quantity": Number,
            "price": Number
        },
        "medium": {
            "metrics": String,
            "quantityMultiplier": Float,
            "quantity": Number,
            "price": Number
        },
        "premium": {
            "metrics": String,
            "quantityMultiplier": Float,
            "quantity": Number,
            "price": Number
        }
    }
},
"electrical": {
    "economical": {
        "metrics": String,
        "quantityMultiplier": Float,
        "quantity": Number,
        "price": Number
    },
    "medium": {
        "metrics": String,
        "quantityMultiplier": Float,
        "quantity": Number,

```

```
        "price": Number
    },
    "premium": {
        "metrics": String,
        "quantityMultiplier": Float,
        "quantity": Number,
        "price": Number
    }
},
"painting": {
    "economical": {
        "metrics": String,
        "quantityMultiplier": Float,
        "quantity": Number,
        "price": Number
    },
    "medium": {
        "metrics": String,
        "quantityMultiplier": Float,
        "quantity": Number,
        "price": Number
    },
    "premium": {
        "metrics": String,
        "quantityMultiplier": Float,
        "quantity": Number,
        "price": Number
    }
},
"bathroomFixtures": {
    "economical": {
        "metrics": String,
        "quantityMultiplier": Float,
        "quantity": Number,
        "price": Number
    },
    "medium": {
        "metrics": String,
        "quantityMultiplier": Float,
        "quantity": Number,
        "price": Number
    },
    "premium": {
        "metrics": String,
        "quantityMultiplier": Float,
        "quantity": Number,
        "price": Number
    }
},
"plumbing": {
    "economical": {
        "metrics": String,
        "quantityMultiplier": Float,
        "quantity": Number,
        "price": Number
    },
    "medium": {
        "metrics": String,
        "quantityMultiplier": Float,
        "quantity": Number,
        "price": Number
    },
    "premium": {
        "metrics": String,
```

```

        "quantityMultiplier": Float,
        "quantity": Number,
        "price": Number
    }
}
},
"kitchen": {
    "modular": {
        "economical": {
            "metrics": String,
            "quantityMultiplier": Float,
            "quantity": Number,
            "price": Number
        },
        "medium": {
            "metrics": String,
            "quantityMultiplier": Float,
            "quantity": Number,
            "price": Number
        },
        "premium": {
            "metrics": String,
            "quantityMultiplier": Float,
            "quantity": Number,
            "price": Number
        }
    }
},
"parking": {
    "twoWheeler": {
        "metrics": String,
        "quantityMultiplier": Float,
        "quantity": Number,
        "price": Number
    },
    "fourWheeler": {
        "metrics": String,
        "quantityMultiplier": Float,
        "quantity": Number,
        "price": Number
    }
},
"others": {
    "services": {
        "economical": {
            "metrics": String,
            "quantityMultiplier": Float,
            "quantity": Number,
            "price": Number
        },
        "medium": {
            "metrics": String,
            "quantityMultiplier": Float,
            "quantity": Number,
            "price": Number
        },
        "premium": {
            "metrics": String,
            "quantityMultiplier": Float,
            "quantity": Number,
            "price": Number
        }
    },
    "labour": {

```

```

    "economical": {
      "metrics": String,
      "quantityMultiplier": Float,
      "quantity": Number,
      "price": Number
    },
    "medium": {
      "metrics": String,
      "quantityMultiplier": Float,
      "quantity": Number,
      "price": Number
    },
    "premium": {
      "metrics": String,
      "quantityMultiplier": Float,
      "quantity": Number,
      "price": Number
    }
  },
  "aggregates": {
    "metrics": String,
    "quantityMultiplier": Float,
    "quantity": Number,
    "price": Number
  },
  "medium": {
    "metrics": String,
    "quantityMultiplier": Float,
    "quantity": Number,
    "price": Number
  },
  "premium": {
    "metrics": String,
    "quantityMultiplier": Float,
    "quantity": Number,
    "price": Number
  }
},
"sand": {
  "economical": {
    "metrics": String,
    "quantityMultiplier": Float,
    "quantity": Number,
    "price": Number
  },
  "medium": {
    "metrics": String,
    "quantityMultiplier": Float,
    "quantity": Number,
    "price": Number
  },
  "premium": {
    "metrics": String,
    "quantityMultiplier": Float,
    "quantity": Number,
    "price": Number
  }
}
}
```

Response Parameters :

Parameter	Type	Description
material	Json	Budget details about the materials
cement	Json	Budget details about cement
steel	Json	Budget details about the steel
structure	Json	Budget details about the structure
finishing	Json	Budget details about the finishing
flooring	Json	Budget details about the flooring
windowsDoors	Json	Budget details about the windowsDoors
electrical	Json	Budget details about the electrical
painting	Json	Budget details about the painting
bathroomFixtures	Json	Budget details about the bathroomFixtures
plumbing	Json	Budget details about the plumbing
kitchen	Json	Budget details about the kitchen
modular	Json	Budget details about the kitchen modular
parking	Json	Budget details about the parking
twoWheeler	Json	Budget details about the 2 wheeler vehicle parking
fourWheeler	Json	Budget details about the 4 wheeler vehicle parking
others	Json	Budget details about the other aspects
services	Json	Budget details about the services
labour	Json	Budget details about the labour
aggregates	Json	Budget details about the aggregates
sand	Json	Budget details about the sand

c. Save Budget API

Description	API to save the budget estimate for the user
Method	GET
URL	/estimatorService/saveEstimate
Headers	Authorization: [JWT Token]

Request:

```
{
  "minPrice": Number,
  "maxPrice": Number,
  "stdEstimateType": String,
  "carpetArea": Number,
  "floorCount": String,
  "material": {
    "cement": {
      "packageType":String
```

```

    },
    "steel": {
        "packageType": String
    },
    "structure": {
        "packageType": String
    }
},
"finishing": {
    "flooring": {
        "packageType": String
    },
    "windowsDoors": {
        "packageType": String
    },
    "electrical": {
        "packageType": String
    },
    "painting": {
        "packageType": String
    },
    "bathroomFixtures": {
        "packageType": String
    },
    "plumbing": {
        "packageType": String
    }
},
"kitchen": {
    "isModular": Boolean
},
"parking": {
    "isActive": Boolean,
    "twoWheelerCount": Number,
    "fourWheelerCount": Number
}
}

```

Response :

```

{
    "status": "success",
    "requestId": null,
    "result": {
        "message": "Estimate saved successfully"
    }
}

```

Request Parameters :

Parameter	Type	Description
minPrice	Number	Minimum Price in the estimate
maxPrice	Number	Maximum Price in the estimate
stdEstimateType	String	Standard estimate type
carpetArea	Number	Carpet Area in the estimate
floorCount	String	Number of floors
material	Json	Details about the materials
cement	Json	Json containing the package type for cement

steel	Json	Json containing the package type for steel used
structure	Json	Json containing the package type for structures
finishing	Json	Details about the finishing
flooring	Json	Json containing the package type for flooring used
windowsDoors	Json	Json containing the package type for windows and doors
electrical	Json	Json containing the package type for electrical fixtures
painting	Json	Json containing the package type for painting
bathroomFixtures	Json	Json containing the package type for bathroom fixtures
plumbing	Json	Json containing the package type for plumbing
kitchen	Json	Details about the kitchen
isModular	Boolean	Is the kitchen modular or not
parking	Json	Details about the parking
isActive	Boolean	Is parking considered or not
twoWheelerCount	Number	Total number of 2 wheeler vehicles
fourWheelerCount	Number	Total number of 4 wheeler vehicles

9. Advisory APIs

a. Service Requests Fetch API

Description	Api for fetching the consultancy services of a user sorted by schedule (past then latest)
Method	GET
URL	/advisoryServices/postUserServices
Headers	Authorization: [JWT Token]

Request:

```
{
  "limit": Number,
  "offset": Number,
  "query": [{
    "type": String
    "value": String Array
  }]
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "data": {
      "pastServices": [{
        "servicesRequested": [{
          "constructionStage": String,
          "serviceName": String,
          "requestStatus": String,
          "timeSlot": String,
          "serviceDate": DateTime,
          "creationDate": DateTime,
          "lastUpdated": DateTime,
          "serviceCompletedDate": DateTime,
          "allotedAdvisor": String
        }],
        "requestId": Number,
        "crmTicketId": Number,
        "userAttachmentUrls": String Array,
        "utechAttachmentUrls": String Array,
        "requestDescription": String Array
      }],
      "upcomingServices": [{
        "servicesRequested": [{
          "constructionStage": String,
          "serviceName": String,
          "requestStatus": String,
          "timeSlot": String,
          "serviceDate": DateTime,
          "creationDate": DateTime,
          "lastUpdated": DateTime,
          "serviceCompletedDate": DateTime,
```

```

        "allotedAdvisor": String
    },
    "requestId": Number,
    "crmTicketId": Number,
    "userAttachmentUrls": String Array,
    "utechAttachmentUrls": String Array,
    "requestDescription": String Array
  },
  "latestService": [{
    "servicesRequested": [{
      "constructionStage": String,
      "serviceName": String,
      "requestStatus": String,
      "timeSlot": String,
      "serviceDate": DateTime,
      "creationDate": DateTime,
      "lastUpdated": DateTime,
      "serviceCompletedDate": DateTime,
      "allotedAdvisor": String
    }],
    "requestId": Number,
    "crmTicketId": Number,
    "userAttachmentUrls": String Array,
    "utechAttachmentUrls": String Array,
    "requestDescription": String
  }]
}

```

Request Parameters :

Parameter	Type	Description	isMandatory
offset	Number	Starting index of the results	Yes
limit	Number	Total number of results to be returned from offset	Yes
query	Json Array	List of Query filters	Yes
type	Number	Type of filter eg: schedule, serviceType, serviceName, requestStatus	Yes
value	String Array	Values of the filter type	Yes

Response Parameters :

Parameter	Type	Description
pastServices	Json Array	List of all past services
upcomingServices	Json Array	List of all upcoming services
latestServices	Json Array	List of all latest services
servicesRequested	Json Array	List of services requested
constructionStage	String	Construction stage
serviceName	String	Name of service
requestStatus	String	Status of the service request
timeSlot	String	Time slot of the service request
serviceDate	DateTime	Service request date time

creationDate	DateTime	Service creation date time
lastUpdated	DateTime	Service last update date time
serviceCompletedDate	DateTime	Service completed date tome
allotedAdvisor	String	Allocated advisor for the service
requestId	Number	Request id for the service
crmTicketId	Number	CRM Id for the service
userAttachmentUrls	String Array	List of urls for the files attached in the service by user
utechAttachmentUrls	String Array	List of urls for the files attached in the service by Utratech
requestDescription	String	Description of the service request

b. Service Requests Post API

Description	Api for saving advisory service request of a user
Method	POST
URL	/advisoryServices/postUserServices/
Headers	Authorization: [JWT Token]

Request:

```
{
  "appld": Number,
  "packages": [{
    "constructionStage": String,
    "date": Date,
    "service": String,
    "timeSlot": String
  }],
  "plotLocation": String,
  "requestDescription": String
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "message": "service Request saved successfully",
    "serviceRequestId": Number
  }
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes
packages	Json Array	List of service packages	Yes
constructionStage	String	Stage of construction	Yes
date	Date	Date of service request	Yes

service	String	Name of the service	Yes
timeSlot	String	Time slot range of service request	Yes
plotLocation	String	Address of the plot	Yes
requestDescription	String	Description about the service request	Yes

Response Parameters :

Parameter	Type	Description
serviceRequestId	Number	Id of the service request that is raised

c. Service Requests Feedback API

Description	Api for saving feedback of a advisory service request raised by the user
Method	POST
URL	/advisoryServices/feedback/
Headers	Authorization: [JWT Token]

Request:

```
{
  "requestId": Number,
  "feedback": String,
  "rating": Number
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "message": "Feedback Posted"
  }
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
requestId	Number	Id of the service request against which feedback is being given	Yes
feedback	String	Feedback text provided by the user	Yes
rating	Number	Rating given by the user	Yes

d. Service Requests Description Fetch API

Description	Api for fetching the meta information like video, description of consultancy service from AEM
Method	GET
URL	/advisoryServices/servicesDescription/
Headers	Authorization: [JWT Token]

Request:

```
{
    "serviceName": String
}
```

Response :

```
{
    "status": "success",
    "requestId": null,
    "result": {
        "data": {
            "videoThumbnail": String,
            "videoLink": String,
            "content": String
        }
    }
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
serviceName	String	Name of the service (eg: Executive Visit)	Yes

Response Parameters :

Parameter	Type	Description
videoThumbnail	String	URL for the thumbnail of the video
videoLink	String	URL for the link of the video
content	String	Text Content description for the service request

e. Service Requests Feedback Fetch API

Description	Api for fetching the feedback of the service request
Method	GET
URL	/advisoryServices/getfeedback/
Headers	Authorization: [JWT Token]

Request:

```
{
    "requestId": Number
}
```

Response :

```
{
    "status": "success",
    "requestId": null,
    "result": {
        "requestId": Number,
        "feedback": String,
        "rating": Number
    }
}
```

```
}  
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
requestId	Number Array	List of Id of the service request against which feedback is sought	Yes

Response Parameters :

Parameter	Type	Description
requestId	Number	Id of the service request against which feedback was given
feedback	String	Feedback text provided by the user
rating	Number	Rating given by the user

10. AEM Proxy APIs

APIs for accessing DAM Content and AEM Models

a. AEM Response Proxy API

Description	AEM proxy API for fetching the data from AEM based on resource path of the resource
Method	GET
URL	/aemResponseProxy/
Headers	Authorization: [JWT Token]

Request:

```
{
  "resourcePath": String,
  "type ": String
}
```

Response :

Response Object depends upon the type of the resources being fetched. It returns the json schema of the identified resource.

Request Parameters :

Parameter	Type	Description	isMandatory
resourcePath	String	URLidentifying the resource in AEM	Yes
type	String	Type of the resource (eg: designElement)	Yes

11. User Onboarding APIs

a. Logout request API

Description	API for logging the user out of the application
Method	POST
URL	/userOnboardingService/logOut/
Headers	Authorization: [JWT Token]

Request:

```
{
  "appld": Number,
  "refreshToken": String
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "message": "Logged out successfully."
  }
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes
refreshToken	String	Refresh token for the user	Yes

b. FCM Registration API

Description	API for registering the user in Firebase Cloud Messaging
Method	POST
URL	/userOnboardingService/registration/
Headers	Authorization: [JWT Token]

Request:

```
{
  "appld": Number,
  "deviceId": String,
  "isDeleted": Boolean,
  "pushPlatform": String,
  "registrationToken": String
}
```

Response :


```
{
  "status": "success",
  "requestId": null,
  "result": "Fcm Token Saved."
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes
deviceId	String	Device Id	Yes
isDeleted	Boolean	Whether to delete the registration	Yes
pushPlatform	String	Platform to send the push notifications. "fcm" is required	Yes
registrationToken	String	Registration Token	Yes

c. FAQs Fetch API

Description	API for fetching the frequently asked questions from AEM
Method	GET
URL	/userOnboardingService/faqs/
Headers	Authorization: [JWT Token]

Request:

None.

Response :

```
{
  "status": "success",
  "exception": null,
  "result": {
    "activities": [{
      "faqs": [{
        "answer": String,
        "question": String
      }],
      "name": String
    }],
    "general": [{
      "answer": String,
      "question": String
    }]
  }
}
```

Response Parameters :

Parameter	Type	Description
activities	Json Array	List of faqs for each type under activities category
general	Json Array	List of faqs for each type under general category
faqs	Json Array	List of Questions and Answers Object

name	String	Type of the faqs
------	--------	------------------

12. Detailed Estimate APIs

a. Add Estimate API

Description	API for adding Single detailed estimate of user
Method	POST
URL	/detailedEstimation/addEstimate/
Headers	Authorization: [JWT Token]

Request:

```
{
  "appld": 1,
  "images": [{
    "base64String": String,
    "fileName": String
  }],
  "plotArea": String,
  "title": String
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "data": "We will notify you with the detailed estimate in 2 - 3 days"
  }
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes
images	Json Array	List of image objects	Yes
base64String	String	Image in base64 encoding	Yes
fileName	String	Name of the image file	Yes
plotArea	String	Size of the plot	Yes
title	String	Title of the estimate	Yes

b. Fetch all detailed estimates API

Description	API for fetching all detailed estimate of user
Method	GET

URL	/detailedEstimation/getAllEstimate/
Headers	Authorization: [JWT Token]

Request:

```
{
  "appld": 1
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "data": [{
      "id": Number,
      "serviceRequestId": Number,
      "title": String,
      "isFinalized": Boolean,
      "status": String,
      "plotArea": String,
      "date": DateTime,
      "lastUpdated": DateTime,
      "totalEstimate": Number,
      "comment": String,
      "attachmentUrl": String Array
    }]
  }
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes

Response Parameters :

Parameter	Type	Description
id	Number	Id of the detailed estimate request
serviceRequestId	Number	id of the service request within which detailed estimate request is present
title	String	Title of the request
isFinalized	Boolean	Whether the estimate is finalized
status	String	Status of the estimate request (Eg: In Progress)
plotArea	String	Area of the plot in request
date	DateTime	Date of request
lastUpdated	DateTime	Date of last update of request
totalEstimate	Number	Total Estimate for the request
comment	String	Comments, if any
attachmentUrl	String Array	List of urls for files attached with the request

c. Fetch user designs API

Description	API for fetching all the user designs for the detailed estimate requests of user
Method	GET
URL	/detailedEstimation/getUserDesigns/
Headers	Authorization: [JWT Token]

Request:

```
{
  "appld": 1
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "data": {
      "images": String Array,
      "designId": Number
    }
  }
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes

Response Parameters :

Parameter	Type	Description
images	String Array	List of design images for the service request
designId	Number	id of the service request design

d. Fetch Detailed Estimate API

Description	API for fetching a single detailed estimate requests of user
Method	GET
URL	/detailedEstimation/getEstimate/
Headers	Authorization: [JWT Token]

Request:

```
{
    "estimate": 1
}
```

Response :

```
{
    "status": "success",
    "requestId": null,
    "result": {
        "data": {
            "estimateName": String,
            "estimateAmount": Number,
            "estimateDate": Date,
            "phase": [{
                "title": String,
                "amount": Number,
                "detail": [{
                    "title": String,
                    "amount": Number,
                    "detail": [{
                        "title": String,
                        "subtitle": String,
                        "amount": Number
                    }
                ]
            }
        ]
    }
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
estimateId	Number	Id of the detailed estimate service request	Yes

Response Parameters :

Parameter	Type	Description
estimateName	String	Name of the detailed estimate request
estimateAmount	Number	Amount of the detailed estimate request
estimateDate	Date	Date of the detailed estimate request
phase	Json Array	List of detailed estimate components
title	String	Title of the component
amount	Number	Amount of the component
detail	Json Array	Details of the component if any

13. Financial Service APIs

a. Loan Request API

Description	API for raising loan request for a user
Method	POST
URL	/financialServices/loanRequest/
Headers	Authorization: [JWT Token]

Request:

```
{
  "appld": Number,
  "amount": Number,
  "applicants": [{
    "applicantType": String,
    "currentEmployer": String,
    "dob": Date,
    "emailId": String,
    "existingEmi": Number,
    "maritalStatus": String,
    "mobileNumber": String,
    "modeOfIncome": String,
    "name": String,
    "panNumber": String,
    "residentialAddress": String,
    "totalIncome": Number
  }],
  "objective": String,
  "propertyAddress": String
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "message": "Loan request saved succesfully"
  }
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes
amount	Number	Amount for which loan request is raised	Yes
applicants	Json Array	Applicants list	Yes
applicantType	String	Type of the applicant (Primary or Secondary)	Yes
currentEmployer	String	Name of the current employer	Yes

dob	String	Date of Birth	Yes
emailId	String	Email ID of the applicant	Yes
existingEmi	Number	Current EMI	Yes
maritalStatus	String	Married or unmarried	Yes
mobileNumber	String	MobileNumber	Yes
modeOfIncome	String	Whether salaried or businessman	Yes
name	String	Name of the applicant	Yes
panNumber	String	PAN Number of the applicant	Yes
residentialAddress	String	Residential Address of the applicant	Yes
totalIncome	Number	Total Income of the applicant	Yes
objective	String	Objective of the loan (Eg: House building)	Yes
propertyAddress	String	Address of the property	Yes

b. Loan Documents API

Description	API for fetching documents required for a loan from AEM
Method	GET
URL	/financialServices/loanDocuments/
Headers	Authorization: [JWT Token]

Request:

None

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "data": [{
      "title": String,
      "description": String
    }]
  }
}
```

Response Parameters :

Parameter	Type	Description
data	Json Array	List of documents object
title	String	Title of the document specifying the reason (Eg: Identity Proof)
description	String	Comma separated list of documents

c. Loan Request Fetch API

Description	API for fetching loan request for a user
Method	GET
URL	/financialServices/getLoanRequest/
Headers	Authorization: [JWT Token]

Request:

None. User is identified using JWT token and loan request is returned for that user

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": [{
    "appld": Number,
    "amount": Number,
    "applicants": [{
      "applicantType": String,
      "currentEmployer": String,
      "dob": Date,
      "emailId": String,
      "existingEmi": Number,
      "maritalStatus": String,
      "mobileNumber": String,
      "modeOfIncome": String,
      "name": String,
      "panNumber": String,
      "residentialAddress": String,
      "totalIncome": Number
    }],
    "objective": String,
    "propertyAddress": String
  }]
}
```

Response Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes
amount	Number	Amount for which loan request is raised	Yes
applicants	Json Array	Applicants list	Yes
applicantType	String	Type of the applicant (Primary or Secondary)	Yes
currentEmployer	String	Name of the current employer	Yes
dob	String	Date of Birth	Yes
emailId	String	Email ID of the applicant	Yes
existingEmi	Number	Current EMI	Yes
maritalStatus	String	Married or unmarried	Yes

mobileNumber	String	MobileNumber	Yes
modeOfIncome	String	Whether salaried or businessman	Yes
name	String	Name of the applicant	Yes
panNumber	String	PAN Number of the applicant	Yes
residentialAddress	String	Residential Address of the applicant	Yes
totalIncome	Number	Total Income of the applicant	Yes
objective	String	Objective of the loan (Eg: House building)	Yes
propertyAddress	String	Address of the property	Yes

14. Learn Module APIs

a. Add learn Preferences API

Description	API for adding learn topic preferences for a user
Method	POST
URL	/learnService/addPreferences/
Headers	Authorization: [JWT Token]

Request:

```
{
  "appld": Number,
  "value": String Array
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "message": "User preferences saved successfully."
  }
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes
value	String Array	List of preferences / learn module categories	Yes

b. Update learn Preferences API

Description	API for updating learn topic preferences for a user
Method	POST
URL	/learnService/updatePreferences/
Headers	Authorization: [JWT Token]

Request:

```
{
  "appld": Number,
  "value": String Array
}
```

Response :

```
{
  "status": "success",
```

```

    "requestId": null,
    "result": {
        "message": "User preferences updated successfully."
    }
}

```

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes
value	String Array	List of preferences / learn module categories	Yes

c. Learn Categories fetch API

Description	API for fetching learn topics from AEM
Method	GET
URL	/learnService/getLearnCategories/
Headers	Authorization: [JWT Token]

Request:

```

{
    "lang": String
}

```

Response :

```

{
    "status": "success",
    "requestId": null,
    "result": {
        "learnTopicNames": [{
            "value": String,
            "displayName": String,
            "backgroundImage": String,
            "isSelected": Boolean
        }]
    }
}

```

Request Parameters :

Parameter	Type	Description	isMandatory
lang	String	Language in which the categories are to be returned (Eg: en)	Yes

Response Parameters :

Parameter	Type	Description
learnTopicNames	Json Array	List of object containing the learn categories data
value	String	Backend Value of the category name
displayName	String	Display value of category

backgroundImage	String	Image link for the category
isSelected	Boolean	Is selected by the user

d. Learn Content Data Fetch API

Description	API for fetching learn contents from AEM
Method	GET
URL	/learnService/getTopicCards/
Headers	Authorization: [JWT Token]

Request:

```
{
  "appld": Number,
  "limit": Number,
  "offset": Number,
  "query": [{
    "type": String,
    "value": String Array
  }]
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "totalRows": Number,
    "startIndex": Number,
    "cardDetails": [{
      "type": String,
      "metadata": {
        "pdfCard": String,
        "video": String,
        "article": {
          "description": String,
          "title": String,
          "resourcePath": String,
          "solutionTag": String Array,
          "authorType": String,
          "articleUrl": String,
          "imageUrl": String
        }
      }
    }]
  }
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the request is raised	Yes
query	Json	List of filter objects; all of those will be ORed	Yes

type	String	Type of content to be fetched	Yes
value	String Array	List of values for the filter type	Yes
offset	Number	Starting index of the results	Yes
limit	Number	Total number of results to be returned from offset	Yes

Response Parameters :

Parameter	Type	Description
totalRows	Number	Total number of records in the result set
startIndex	Number	Starting index number of the record in current api call
cardDetails	Json Array	Details of the content
type	String	Type of the content
metadata	Json	Data of the content
pdfCard	String	PDF File url
video	String	Video url link
article	Json	Article data
description	String	Description of the content
title	String	Title of the content
resourcePath	String	URL of the content in AEM
solutionTag	String Array	Tags associated with the content
authorType	String	Type of author of the content
articleUrl	String	URL of the article
imageUrl	String	URL of the image of content

15. Notification APIs

a. Update Notification API

Description	API request for marking notification read before a given timestamp
Method	POST
URL	/support/updateNotification/
Headers	Authorization: [JWT Token]

Request:

```
{
  "appld": Number,
  "timeStamp": DateTime
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": "Status updated."
}
```

Request Parameters :

Parameter	Type	Description	isMandatory
appld	Number	Id of the application from where the API request is raised	Yes
timeStamp	DateTime	Timestamp before which notification is marked read	Yes

b. Fetch Notifications API

Description	API request for fetching notifications for a user
Method	GET
URL	/support/notification/
Headers	Authorization: [JWT Token]

Request:

None. User is identified using the JWT Token and then the user's notifications is fetched.

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "recent": {
      "notification": [{
        "notificationId": Number,
        "title": String,
```

```

        "notificationText": String,
        "notificationObject": {
            "serviceRequestId": Number
        },
        "status": String,
        "date": DateTime
    }}
},
"earlier": {
    "notification": [{
        "notificationId": Number,
        "title": String,
        "notificationText": String,
        "notificationObject": {
            "serviceRequestId": Number
        },
        "status": String,
        "date": DateTime
    }]
}
}
}

```

Response Parameters :

Parameter	Type	Description
recent	Json	Details about the recent notifications
earlier	Json	Details about the earlier notifications
notification	Json Array	List of notification objects
notificationId	Number	Id of the notification
title	String	Title of notification
notificationText	String	Text content of the notification
notificationObject	Json	Additional details about the particular notification
status	String	Status of notification (eg: unread)
date	DateTime	Date and time of notification

16. Progress Tracker APIs

a. Fetch house building stages API

Description	API for fetching the House building stages from AEM
Method	GET
URL	/progressTrackerService/houseBuildingStages/
Headers	Authorization: [JWT Token]

Request:

None.

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "homeBuildingStages": [{
      "stageId": String,
      "stageName": String,
      "description": String,
      "thumbnail": String
    }]
  }
}
```

Response Parameters :

Parameter	Type	Description
homeBuildingStages	Json Array	List of various stages in house building
stageId	String	Id of the stage in backend
stageName	String	Name of the stage
description	String	Description of the stage
thumbnail	String	Thumbnail image of the stage

b. Fetch expense history API

Description	API for fetching the expense history of the user based on the order in which it was added
Method	GET
URL	/progressTrackerService/expenseHistory/
Headers	Authorization: [JWT Token]

Request:

```
{
    "sortOrder": String
}
```

Response :

```
{
    "status": "success",
    "requestId": null,
    "result": {
        "totalResults": Number,
        "expenses": [{
            "id": Number,
            "categoryName": String,
            "subCategoryName": String,
            "amount": Number,
            "addedOn": DateTime,
            "type": String,
            "documentsAttached": Number
        }]
    }
}
```

Request Parameters:

Parameter	Type	Description	isMandatory
sortOrder	String	Order of addedOn field in which the results are to be returned (ASC or DESC)	Yes

Response Parameters :

Parameter	Type	Description
totalResults	Number	Total number of results returned
expenses	Json Array	List of expenses
id	Number	Id of the expense
categoryName	String	Name of category for the expense
subCategoryName	String	Name of sub category for the expense
amount	Number	Total amount of expense
addedOn	DateTime	Date and time at which the expense was added
type	String	Type of Expense
documentsAttached	Number	Number of documents attached with the expense

c. Fetch expense pdf API

Description	API for fetching the pdf file for the expense
Method	GET
URL	/progressTrackerService/generateExpensePdf/

Headers	Authorization: [JWT Token]
----------------	----------------------------

Request:

```
{
  "sortOrder": String
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "downloadLink": String
  }
}
```

Request Parameters:

Parameter	Type	Description	isMandatory
sortOrder	String	Order of addedOn field in which the results are to be returned (ASC or DESC)	Yes

Response Parameters :

Parameter	Type	Description
downloadLink	String	Url of the pdf generated

d. Fetch detailed estimate API

Description	API for generating PDF of detailed estimate added by the user
Method	GET
URL	/progressTrackerService/getDetailedEstimatePdf/
Headers	Authorization: [JWT Token]

Request:

```
{
  "detailedEstimateId": String
}
```

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": {
    "downloadLink": String
  }
}
```

Request Parameters:

Parameter	Type	Description	isMandatory
-----------	------	-------------	-------------

detailedEstimateId	Number	Id of the detailed estimate against which pdf is requested	Yes
--------------------	--------	--	-----

Response Parameters :

Parameter	Type	Description
downloadLink	String	Url of the pdf generated

e. Fetch progress tracker categories API

Description	API's for fetching the Progress tracker categories of each particular stage
Method	GET
URL	/progressTrackerService/category/
Headers	Authorization: [JWT Token]

Request:

None.

Response :

```
{
  "status": "success",
  "requestId": null,
  "result": [{
    "id": Number,
    "categoryName": String,
    "subCategory": [{
      "id": Number,
      "subCategoryName": String,
      "isDeleted": Number,
      "types": [{
        "id": Number,
        "typeTitle": String,
        "isTypeDeleted": Number
      }]
    }]
  }]
}
```

Response Parameters :

Parameter	Type	Description
result	Json Array	List of categories object
id	Number	Id of the category
categoryName	String	Name of the category
subCategory	JsonArray	List of sub categories
subCategory[].id	Number	Id of the sub category
subCategoryName	String	Name of the sub category
isDeleted	Boolean	If the subcategory is removed or not
types	Json Array	List of types within the sub category
types[].id	Number	Id of the type

typeTitle	String	Title of the type
isTypeDeleted	Boolean	Whether the type is removed or not