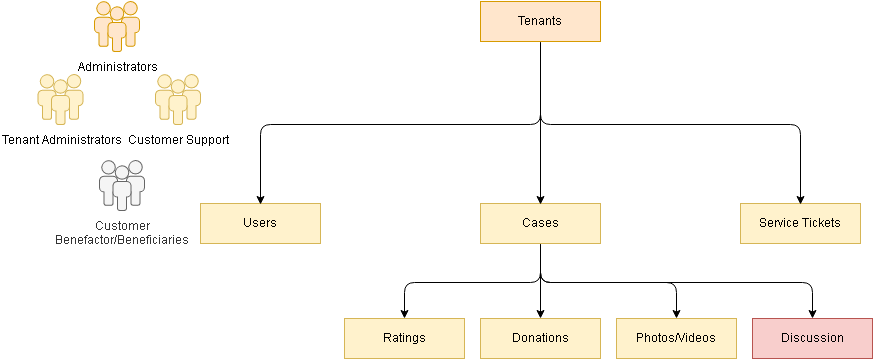
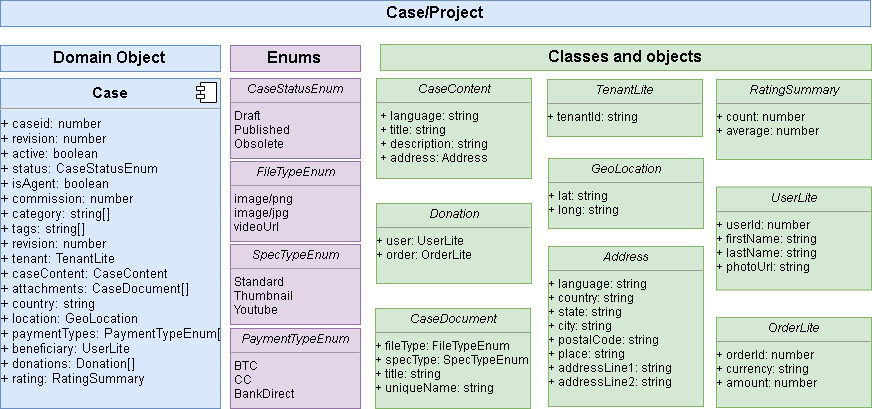
**Multi tenancy data model**

Any data in persistence will be stored isolated for a tenant eg: Users, Cases, Service tickets, documents, etc.



**‘Case’ Data model discovery**



Auth APIs

Login

|  |  |
| --- | --- |
| Title | Login – Issue Token |
| Request Url | ‘api/auth/issue’ |
| Method | POST |
| Request Body: | {"userName": "string", "password": "string"} |
| Response Body: | {"access\_token": "string", "refresh\_token": "string"} |

Renew Login

|  |  |
| --- | --- |
| Title | Renew Login – Re-Issue Token |
| Request Url | ‘api/auth/refresh’ |
| Method | POST |
| Request Header: | "Authorization": " refresh\_token" |
| Response Body: | {"access\_token": "string", "refresh\_token": "string"} |

Register

|  |  |
| --- | --- |
| Title | Renew Login – Re-Issue Token |
| Request Url | ‘api/pub/register’ |
| Method | POST |
| Request Body: | {"userName": "string", "password": "string", "email": "string", "phone": "string", "firstName": "string", "lastName": "string"} |
| Status Codes: | 200, 400 – Bad Request (reason: “user exists”, “invalid parameters”) |

Activate

Forgot Password – send notification to recover password

Set Password – set password if forgot

Change Password

**Domain objects identified**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tenants** | **Users** | **Cases** | **Tickets** |
| tenantId  name  businessName  baseLanguage  baseCurrency  address: { country, state, city, place, addressLine1, pincode }  location: { longitude, latitude }  contacts:[{ctype, firstName, lastName, phone, email}]  -authProviders: [“provider”] – tenant specific config  -payConfig: [“config”] – tenant specific config | tenantId (optional if super admin)  userId (email)  role  email (unique)  phone (unique)  language – default from tenant  basic: {firstName, lastName, profilePhoto , language}  authProviders: [{ provider, userid,token,phash,psalt }]  verification: { activated (default: true)  emailVerified (default: true)  phoneVerified (default: true)}  rating: {count, average} | Tenant: {tenantId, name}  caseId  revision  isAgent  active (latest revision will be active)  content:[{ lang, title, description}]  address: { country, state, city, place, addressLine1, pincode }  location: { longitude, latitude }  status (draft, published, obsolete)  commission  currency  attachments: [ order, isprimary, doc: {fileType, imageType, videoType, title, uniqueName }]  beneficiary: { userId, firstName, lastName }  donations: [{ user: { userId, firstName, lastName }, order: { currency, amount }}]  rating: {count, average}  category: [“category name”]  tags: [“tag name”] | tenantId  ticketType  caseid (optional)  email  phone  title  description  attachments: [ {fileType, imageType, videoType, title, uniqueName }] |
| **Documents (n/a) – file server** | **Case Ratings** | **Case Orders** | **Case Transaction Log** |
| tenantId  docType  refId  refId2  fileType,  imageType,  videoType,  title,  uniqueName | tenantId  caseId  rating  on  comments  user: { userId, firstName, lastName } | tenantId  orderId  caseId  currency  amount  breakdown: [{ name, currency, amount }]  user: { userId, firstName, lastName }  transactions: [{ currency, amount, trnReference, trnHash }] | tenantId  caseId  order: {}  trnInfo: {} |
| CaseReport | Token – claims | **Localization – tenant impact** | Case Discussion |
| tenantId  caseId  on  comments  user: { userId, firstName, lastName } | tenantids  role  userid  email  name - firstName + lastName  language , claims | Tenant specific text – application title, tagline, etc.  **Theme – tenant impact**  Tenant specific styles – logo, background, color, font | N/A – Social media  Specific to tenant – using appid |
|  | **Beneficiary Rating** |  |  |
|  | tenantId  beneficiaryId  rating  on  comments  user: { userId, firstName, lastName } |  |  |

**Logical Diagrams (Components)**

