

Full End-to-End Mortgage Flow Scenario

Purpose: Complete API-driven flow from tenant bootstrap through mortgage completion.

Use Case: Guide Postman Flows design and E2E testing.

Last Updated: 2026-01-12

Test Location: <services/mortgage-service/tests/e2e/full-mortgage-flow/>

Summary

This scenario walks through the complete lifecycle of a 10/90 mortgage application, starting from tenant creation through to the first mortgage payment. All operations are performed via REST APIs—nothing is seeded manually or via direct database access.

Key Characteristics:

- 10% downpayment is **ONE_TIME** (single payment, not instalments)
- 90% mortgage at 9.5% p.a. over 20 years
- All configuration via REST API
- Complete KYC with document approval
- Full payment processing

Actors

Actor	Role	Organization	Description
System Admin	Bootstrap Operator	-	Uses bootstrap secret to initialize tenant
Adaeze	Mortgage Operations Officer	QShelter	Reviews documents (Stage 1: Internal), configures payment methods
Nkechi	Loan Officer	Access Bank	Reviews documents (Stage 2: Bank), uploads preapproval letter
Emeka	Developer Representative	Lekki Gardens	Reviews property-related documents
Chidi	Customer	-	First-time homebuyer, age 40

Two-Stage Document Approval

Documents requiring bank approval go through a two-stage process:

1. **Stage 1 (INTERNAL):** Adaeze (QShelter's Mortgage Operations Officer) reviews documents first
2. **Stage 2 (BANK):** Nkechi (Access Bank's Loan Officer) reviews after internal approval

Only after BOTH stages approve can the documentation phase complete.

Services Involved

Service	Port (Local)	Responsibilities

user-service	3002	Auth, users, roles, permissions, tenant bootstrap
property-service	3003	Properties, variants, units
mortgage-service	Lambda	Payment plans, methods, applications, phases
notification-service	3004	Event handling, notifications
payment-service	3005	Wallet, payment processing

Prerequisites

- LocalStack running with SNS/SQS configured
- All services deployed locally
- Bootstrap secret configured: `BOOTSTRAP_SECRET=local-bootstrap-secret`

Phase 1: Tenant Bootstrap

Step 1.1: Bootstrap Tenant with Roles and Admin

The bootstrap endpoint creates the tenant, default roles (admin, user, mortgage_ops, finance, legal), their permissions, and the first admin user in a single idempotent operation.

```
POST {{userServiceUrl}}/admin/bootstrap-tenant
x-bootstrap-secret: {{bootstrapSecret}}
Content-Type: application/json

{
  "tenant": {
    "name": "QShelter Real Estate",
    "subdomain": "qshelter"
  },
  "admin": {
    "email": "adaeze@qshelter.com",
    "password": "SecureAdmin123!",
    "firstName": "Adaeze",
    "lastName": "Okonkwo"
  }
}
```

Expected Response (201 Created):

```
{
  "success": true,
  "data": {
    "tenant": {
      "id": "{{tenantId}}",
      "name": "QShelter Real Estate",
      "subdomain": "qshelter",
      "isNew": true
    },
    "roles": [
      "admin",
      "user",
      "mortgage_ops",
      "finance",
      "legal"
    ],
    "users": [
      {
        "id": "{{adminId}}",
        "email": "adaeze@qshelter.com",
        "password": "SecureAdmin123!",
        "firstName": "Adaeze",
        "lastName": "Okonkwo",
        "isDefault": true
      }
    ]
  }
}
```

```

"roles": [
    { "id": "...", "name": "admin", "isNew": true, "permissionsCount": 1 },
    { "id": "...", "name": "user", "isNew": true, "permissionsCount": 6 },
    {
        "id": "...",
        "name": "mortgage_ops",
        "isNew": true,
        "permissionsCount": 7
    },
    { "id": "...", "name": "finance", "isNew": true, "permissionsCount": 6 },
    { "id": "...", "name": "legal", "isNew": true, "permissionsCount": 5 }
],
"admin": {
    "id": "{{adaezeId}}",
    "email": "adaeze@qshelter.com",
    "isNew": true
}
}
}

```

Store Variables:

- tenantId → from response
- adaezeId → from response

Step 1.2: Admin Logs In

```

POST {{userServiceUrl}}/auth/login
Content-Type: application/json

{
    "email": "adaeze@qshelter.com",
    "password": "SecureAdmin123!"
}

```

Expected Response (200 OK):

```

{
    "success": true,
    "data": {
        "accessToken": "{{adaezeAccessToken}}",
        "refreshToken": "...",
        "expiresIn": 900
    }
}

```

Store Variables:

- adaezeAccessToken → for Authorization header

Step 1.3: Create QShelter Platform Organization

```

POST {{userServiceUrl}}/organizations
Authorization: Bearer {{adaezeAccessToken}}
x-tenant-id: {{tenantId}}
Content-Type: application/json

{
  "name": "QShelter",
  "type": "PLATFORM",
  "email": "support@qshelter.com",
  "isPlatformOrg": true
}

```

Expected Response (201 Created):

```
{
  "success": true,
  "data": {
    "id": "{{qshelterOrgId}}",
    "name": "QShelter",
    "type": "PLATFORM",
    "isPlatformOrg": true
  }
}
```

Store Variables:

- qshelterOrgId → for platform organization reference

Step 1.4: Add Adaeze as Mortgage Operations Officer

```

POST {{userServiceUrl}}/organizations/{{qshelterOrgId}}/members
Authorization: Bearer {{adaezeAccessToken}}
x-tenant-id: {{tenantId}}
Content-Type: application/json

{
  "userId": "{{adaezeId}}",
  "role": "MANAGER",
  "title": "Mortgage Operations Officer"
}

```

Expected Response (201 Created):

```
{
  "success": true,
  "data": {
    "id": "...",
    "userId": "{{adaezeId}}",
    "organizationId": "{{qshelterOrgId}}",
    "role": "MANAGER",
    "title": "Mortgage Operations Officer"
  }
}
```

```
        "title": "Mortgage Operations Officer"
    }
}
```

Step 1.5: Create Access Bank Organization

```
POST {{userServiceUrl}}/organizations
Authorization: Bearer {{adaezeAccessToken}}
x-tenant-id: {{tenantId}}
Content-Type: application/json

{
  "name": "Access Bank PLC",
  "type": "BANK",
  "email": "mortgages@accessbank.com"
}
```

Expected Response (201 Created):

```
{
  "success": true,
  "data": {
    "id": "{{accessBankOrgId}}",
    "name": "Access Bank PLC",
    "type": "BANK"
  }
}
```

Store Variables:

- accessBankOrgId → for bank organization reference

Step 1.6: Register Nkechi (Bank Loan Officer)

```
POST {{userServiceUrl}}/auth/signup
Content-Type: application/json

{
  "email": "nkechi@accessbank.com",
  "password": "BankOfficer123!",
  "firstName": "Nkechi",
  "lastName": "Okonkwo"
}
```

Expected Response (201 Created):

```
{
  "success": true,
  "data": {
```

```

    "accessToken": "{{nkechiAccessToken}}",
    "refreshToken": "...",
    "user": {
        "id": "{{nkechiId}}",
        "email": "nkechi@accessbank.com",
        "firstName": "Nkechi",
        "lastName": "Okonkwo"
    }
}
}

```

Store Variables:

- nkechiId → from response
- nkechiAccessToken → for Authorization header

Step 1.7: Add Nkechi to Access Bank

```

POST {{userServiceUrl}}/organizations/{{accessBankOrgId}}/members
Authorization: Bearer {{adaezeAccessToken}}
x-tenant-id: {{tenantId}}
Content-Type: application/json

{
    "userId": "{{nkechiId}}",
    "role": "MANAGER",
    "title": "Loan Officer"
}

```

Expected Response (201 Created):

```

{
    "success": true,
    "data": {
        "id": "...",
        "userId": "{{nkechiId}}",
        "organizationId": "{{accessBankOrgId}}",
        "role": "MANAGER",
        "title": "Loan Officer"
    }
}

```

Phase 2: Customer Registration

Step 2.1: Chidi Signs Up

```

POST {{userServiceUrl}}/auth/signup
Content-Type: application/json

```

```
{  
  "email": "chidi@gmail.com",  
  "password": "CustomerPass123!",  
  "firstName": "Chidi",  
  "lastName": "Nnamdi"  
}
```

Expected Response (201 Created):

```
{  
  "success": true,  
  "data": {  
    "accessToken": "{{chidiAccessToken}}",  
    "refreshToken": "...",  
    "expiresIn": 900,  
    "user": {  
      "id": "{{chidiId}}",  
      "email": "chidi@gmail.com",  
      "firstName": "Chidi",  
      "lastName": "Nnamdi"  
    }  
  }  
}
```

Store Variables:

- chidiId → from response
- chidiAccessToken → for Authorization header

Phase 3: Property Setup

Step 3.1: Create Property (Admin)

```
POST {{propertyServiceUrl}}/properties  
Authorization: Bearer {{adaezeAccessToken}}  
x-tenant-id: {{tenantId}}  
Content-Type: application/json  
  
{  
  "title": "Lekki Gardens Estate",  
  "description": "Premium residential estate in Lekki Phase 1",  
  "category": "SALE",  
  "propertyType": "APARTMENT",  
  "country": "Nigeria",  
  "currency": "NGN",  
  "city": "Lagos",  
  "district": "Lekki"  
}
```

Expected Response (201 Created):

```
{  
  "success": true,  
  "data": {  
    "id": "{{propertyId}}",  
    "title": "Lekki Gardens Estate",  
    "status": "DRAFT"  
  }  
}
```

Step 3.2: Create Property Variant (Admin)

```
POST {{propertyServiceUrl}}/property/properties/{{propertyId}}/variants  
Authorization: Bearer {{adaezeAccessToken}}  
x-tenant-id: {{tenantId}}  
Content-Type: application/json  
  
{  
  "name": "3-Bedroom Flat",  
  "nBedrooms": 3,  
  "nBathrooms": 3,  
  "nParkingSpots": 1,  
  "area": 150,  
  "price": 85000000,  
  "totalUnits": 20,  
  "availableUnits": 15  
}
```

Expected Response (201 Created):

```
{  
  "success": true,  
  "data": {  
    "id": "{{variantId}}",  
    "name": "3-Bedroom Flat",  
    "price": 85000000,  
    "status": "AVAILABLE"  
  }  
}
```

Step 3.3: Create Property Unit (Admin)

```
POST  
{{propertyServiceUrl}}/property/properties/{{propertyId}}/variants/{{variantId}}/units  
Authorization: Bearer {{adaezeAccessToken}}  
x-tenant-id: {{tenantId}}  
Content-Type: application/json  
  
{
```

```
"unitNumber": "14B",
"floorNumber": 14,
"blockName": "Block B"
}
```

Expected Response (201 Created):

```
{
  "success": true,
  "data": {
    "id": "{{unitId}}",
    "unitNumber": "14B",
    "floorNumber": 14,
    "status": "AVAILABLE"
  }
}
```

Step 3.4: Publish Property (Admin)

```
PATCH {{propertyServiceUrl}}/property/properties/{{propertyId}}/publish
Authorization: Bearer {{adaezeAccessToken}}
x-tenant-id: {{tenantId}}
```

Expected Response (200 OK):

```
{
  "success": true,
  "data": {
    "id": "{{propertyId}}",
    "status": "PUBLISHED",
    "publishedAt": "2025-01-XX..."
  }
}
```

Phase 4: Payment Configuration

Step 4.1: Create One-Off Downpayment Plan (Admin)

```
POST {{mortgageServiceUrl}}/payment-plans
Authorization: Bearer {{adaezeAccessToken}}
x-tenant-id: {{tenantId}}
x-idempotency-key: adaeze-create-downpayment-plan-{{$timestamp}}
Content-Type: application/json

{
  "name": "10% One-Off Downpayment",
  "description": "Single upfront payment for property reservation",
  "frequency": "ONE_TIME",
```

```
"interestRate": 0,  
"gracePeriodDays": 30  
}
```

Expected Response (201 Created):

```
{  
  "success": true,  
  "data": {  
    "id": "{{downpaymentPlanId}}",  
    "name": "10% One-Off Downpayment",  
    "frequency": "ONE_TIME"  
  }  
}
```

Step 4.2: Create Flexible Mortgage Plan (Admin)

```
POST {{mortgageServiceUrl}}/payment-plans  
Authorization: Bearer {{adaezeAccessToken}}  
x-tenant-id: {{tenantId}}  
x-idempotency-key: adaeze-create-mortgage-plan-{{$timestamp}}  
Content-Type: application/json  
  
{  
  "name": "Flexible Mortgage at 9.5%",  
  "description": "Monthly payments at 9.5% annual interest, term selected by applicant",  
  "frequency": "MONTHLY",  
  "allowFlexibleTerm": true,  
  "minTermMonths": 60,  
  "maxTermMonths": 360,  
  "termStepMonths": 12,  
  "maxAgeAtMaturity": 65,  
  "interestRate": 9.5,  
  "gracePeriodDays": 15  
}
```

Expected Response (201 Created):

```
{  
  "success": true,  
  "data": {  
    "id": "{{mortgagePlanId}}",  
    "name": "Flexible Mortgage at 9.5%",  
    "allowFlexibleTerm": true,  
    "minTermMonths": 60,  
    "maxTermMonths": 360  
  }  
}
```

Step 4.3: Create Payment Method with 4 Phases (Admin)

```
POST {{mortgageServiceUrl}}/payment-methods
Authorization: Bearer {{adaezeAccessToken}}
x-tenant-id: {{tenantId}}
x-idempotency-key: adaeze-create-payment-method-{{$timestamp}}
Content-Type: application/json

{
  "name": "10/90 Lekki Mortgage",
  "description": "Underwriting → Downpayment → Final Documentation → Mortgage",
  "requiresManualApproval": true,
  "phases": [
    {
      "name": "Underwriting & Documentation",
      "phaseCategory": "DOCUMENTATION",
      "phaseType": "KYC",
      "order": 1,
      "requiredDocumentTypes": ["ID_CARD", "BANK_STATEMENT", "EMPLOYMENT_LETTER"],
      "reviewRequirements": [
        { "party": "INTERNAL", "required": true },
        { "party": "BANK", "required": true }
      ],
      "reviewOrder": "SEQUENTIAL",
      "stepDefinitions": [
        { "name": "Upload Valid ID", "stepType": "UPLOAD", "order": 1 },
        { "name": "Upload Bank Statements", "stepType": "UPLOAD", "order": 2 },
        { "name": "Upload Employment Letter", "stepType": "UPLOAD", "order": 3 },
        { "name": "Internal Review", "stepType": "APPROVAL", "order": 4,
          "reviewParty": "INTERNAL" },
        { "name": "Bank Review", "stepType": "APPROVAL", "order": 5, "reviewParty": "BANK" }
      ]
    },
    {
      "name": "10% Downpayment",
      "phaseCategory": "PAYMENT",
      "phaseType": "DOWNPAYMENT",
      "order": 2,
      "percentOfPrice": 10,
      "paymentPlanId": "{{downpaymentPlanId}}"
    }
  ]
}
```

```
{
  "name": "Final Documentation",
  "phaseCategory": "DOCUMENTATION",
  "phaseType": "VERIFICATION",
  "order": 3,
  "stepDefinitions": [
    {
      "name": "Mortgage Operations Officer Uploads Final Offer",
      "stepType": "UPLOAD",
      "order": 1,
      "metadata": {
        "documentType": "FINAL_OFFER",
        "uploadedBy": "INTERNAL"
      }
    },
    { "name": "Customer Signs Final Offer", "stepType": "SIGNATURE", "order": 2
  }
],
},
{
  "name": "20-Year Mortgage",
  "phaseCategory": "PAYMENT",
  "phaseType": "MORTGAGE",
  "order": 4,
  "percentOfPrice": 90,
  "interestRate": 9.5,
  "paymentPlanId": "{{mortgagePlanId}}"
}
]
```

Expected Response (201 Created):

```
{
  "success": true,
  "data": {
    "id": "{{paymentMethodId}}",
    "name": "10/90 Lekki Mortgage",
    "phases": [
      {
        "id": "...",
        "name": "Underwriting & Documentation",
        "phaseType": "KYC"
      },
      { "id": "...", "name": "10% Downpayment", "phaseType": "DOWNPAYMENT" },
      {
        "id": "...",
        "name": "Final Documentation",
        "phaseType": "VERIFICATION"
      },
      { "id": "...", "name": "20-Year Mortgage", "phaseType": "MORTGAGE" }
    ]
  }
}
```

```
        ]
    }
}
```

Step 4.4: Link Payment Method to Property (Admin)

```
POST {{mortgageServiceUrl}}/payment-methods/{{paymentMethodId}}/properties
Authorization: Bearer {{adaezeAccessToken}}
x-tenant-id: {{tenantId}}
x-idempotency-key: adaeze-link-payment-method-{{$timestamp}}
Content-Type: application/json

{
  "propertyId": "{{propertyId}}",
  "isDefault": true
}
```

Expected Response (201 Created):

```
{
  "success": true,
  "data": {
    "paymentMethodId": "{{paymentMethodId}}",
    "propertyId": "{{propertyId}}",
    "isDefault": true
  }
}
```

Step 4.5: Configure Document Requirement Rules (Admin)

```
POST {{mortgageServiceUrl}}/payment-methods/{{paymentMethodId}}/document-rules
Authorization: Bearer {{adaezeAccessToken}}
x-tenant-id: {{tenantId}}
Content-Type: application/json

{
  "rules": [
    {
      "context": "APPLICATION_PHASE",
      "phaseType": "KYC",
      "documentType": "ID_CARD",
      "isRequired": true,
      "description": "Valid government-issued ID (NIN, Passport, or Driver License)",
      "maxSizeBytes": 5242880,
      "allowedMimeTypes": ["image/jpeg", "image/png", "application/pdf"]
    },
    {
      "context": "APPLICATION_PHASE",

```

```

    "phaseType": "KYC",
    "documentType": "BANK_STATEMENT",
    "isRequired": true,
    "description": "Last 6 months bank statements",
    "maxSizeBytes": 10485760,
    "allowedMimeTypes": ["application/pdf"],
    "expiryDays": 90
  },
  {
    "context": "APPLICATION_PHASE",
    "phaseType": "KYC",
    "documentType": "EMPLOYMENT_LETTER",
    "isRequired": true,
    "description": "Employment confirmation letter",
    "maxSizeBytes": 5242880,
    "allowedMimeTypes": ["application/pdf"]
  }
]
}

```

Phase 5: Customer Application

Step 5.1: Chidi Creates Application

```

POST {{mortgageServiceUrl}}/applications
Authorization: Bearer {{chidiAccessToken}}
x-tenant-id: {{tenantId}}
x-idempotency-key: chidi-create-application-{{$timestamp}}
Content-Type: application/json

{
  "propertyUnitId": "{{unitId}}",
  "paymentMethodId": "{{paymentMethodId}}",
  "title": "Purchase Agreement – Lekki Gardens Unit 14B",
  "applicationType": "MORTGAGE",
  "totalAmount": 85000000,
  "monthlyIncome": 2500000,
  "monthlyExpenses": 800000,
  "applicantAge": 40,
  "selectedMortgageTermMonths": 240
}

```

Expected Response (201 Created):

```
{
  "success": true,
  "data": {
    "id": "{{applicationId}}",
    "applicationNumber": "APP-2025-XXXX",
    "status": "PENDING"
  }
}
```

```

    "status": "DRAFT",
    "phases": [
      {
        "id": "{{kycPhaseId}}", "phaseType": "KYC", "status": "PENDING" },
      {
        "id": "{{downpaymentPhaseId}}",
        "phaseType": "DOWNPAYMENT",
        "status": "PENDING"
      },
      {
        "id": "{{verificationPhaseId}}",
        "phaseType": "VERIFICATION",
        "status": "PENDING"
      },
      {
        "id": "{{mortgagePhaseId}}",
        "phaseType": "MORTGAGE",
        "status": "PENDING"
      }
    ]
  }
}

```

Store Variables:

- applicationId
- kycPhaseId
- downpaymentPhaseId
- verificationPhaseId
- mortgagePhaseId

Step 5.2: Verify Phase Amounts

```

GET {{mortgageServiceUrl}}/applications/{{applicationId}}/phases
Authorization: Bearer {{chidiAccessToken}}
x-tenant-id: {{tenantId}}

```

Expected Response (200 OK):

```

{
  "success": true,
  "data": [
    { "phaseType": "KYC", "totalAmount": 0 },
    { "phaseType": "DOWNPAYMENT", "totalAmount": 8500000 },
    { "phaseType": "VERIFICATION", "totalAmount": 0 },
    { "phaseType": "MORTGAGE", "totalAmount": 76500000, "interestRate": 9.5 }
  ]
}

```

Step 5.3: Submit Application

```
POST {{mortgageServiceUrl}}/applications/{{applicationId}}/transition
Authorization: Bearer {{chidiAccessToken}}
x-tenant-id: {{tenantId}}
x-idempotency-key: chidi-submit-application-{{$timestamp}}
Content-Type: application/json

{
  "action": "SUBMIT",
  "note": "Submitting for processing"
}
```

Expected Response (200 OK):

```
{
  "success": true,
  "data": {
    "id": "{{applicationId}}",
    "status": "PENDING"
  }
}
```

Step 5.4: Activate KYC Phase

```
POST
{{mortgageServiceUrl}}/applications/{{applicationId}}/phases/{{kycPhaseId}}/activate
Authorization: Bearer {{chidiAccessToken}}
x-tenant-id: {{tenantId}}
x-idempotency-key: chidi-activate-kyc-{{$timestamp}}
```

Expected Response (200 OK):

```
{
  "success": true,
  "data": {
    "id": "{{kycPhaseId}}",
    "status": "IN_PROGRESS"
  }
}
```

Phase 6: KYC Document Upload & Approval

Step 6.1: Upload Documents (Customer)

Upload ID Card:

```
POST
{{mortgageServiceUrl}}/applications/{{applicationId}}/phases/{{kycPhaseId}}/documents
Authorization: Bearer {{chidiAccessToken}}
```

```

x-tenant-id: {{tenantId}}
x-idempotency-key: chidi-doc-id-card-{{$timestamp}}
Content-Type: application/json

{
  "documentType": "ID_CARD",
  "url": "https://s3.amazonaws.com/qshelter/chidi/id.pdf",
  "fileName": "id.pdf"
}

```

Upload Bank Statement:

```

POST
{{mortgageServiceUrl}}/applications/{{applicationId}}/phases/{{kycPhaseId}}/documents
Authorization: Bearer {{chidiAccessToken}}
x-tenant-id: {{tenantId}}
x-idempotency-key: chidi-doc-bank-statement-{{$timestamp}}
Content-Type: application/json

{
  "documentType": "BANK_STATEMENT",
  "url": "https://s3.amazonaws.com/qshelter/chidi/bank.pdf",
  "fileName": "bank.pdf"
}

```

Upload Employment Letter:

```

POST
{{mortgageServiceUrl}}/applications/{{applicationId}}/phases/{{kycPhaseId}}/documents
Authorization: Bearer {{chidiAccessToken}}
x-tenant-id: {{tenantId}}
x-idempotency-key: chidi-doc-employment-{{$timestamp}}
Content-Type: application/json

{
  "documentType": "EMPLOYMENT_LETTER",
  "url": "https://s3.amazonaws.com/qshelter/chidi/employment.pdf",
  "fileName": "employment.pdf"
}

```

Step 6.2: Complete Upload Steps (Customer)

```

POST
{{mortgageServiceUrl}}/applications/{{applicationId}}/phases/{{kycPhaseId}}/steps/comp
Authorization: Bearer {{chidiAccessToken}}
x-tenant-id: {{tenantId}}
x-idempotency-key: chidi-step-upload-id-{{$timestamp}}
Content-Type: application/json

```

```
{ "stepName": "Upload Valid ID" }
```

```
POST  
{{mortgageServiceUrl}}/applications/{{applicationId}}/phases/{{kycPhaseId}}/steps/completed  
Authorization: Bearer {{chidiAccessToken}}  
x-tenant-id: {{tenantId}}  
x-idempotency-key: chidi-step-upload-bank-{{$timestamp}}  
Content-Type: application/json  
  
{ "stepName": "Upload Bank Statements" }
```

```
POST  
{{mortgageServiceUrl}}/applications/{{applicationId}}/phases/{{kycPhaseId}}/steps/completed  
Authorization: Bearer {{chidiAccessToken}}  
x-tenant-id: {{tenantId}}  
x-idempotency-key: chidi-step-upload-employment-{{$timestamp}}  
Content-Type: application/json  
  
{ "stepName": "Upload Employment Letter" }
```

Step 6.3: Mortgage Operations Officer Retrieves Documents for Review (Stage 1: Internal)

Adaeze (QShelter's Mortgage Operations Officer) reviews documents first before they go to the bank.

```
GET  
{{mortgageServiceUrl}}/applications/{{applicationId}}/phases/{{kycPhaseId}}/documents  
Authorization: Bearer {{adaezeAccessToken}}  
x-tenant-id: {{tenantId}}
```

Expected Response (200 OK):

```
{  
  "success": true,  
  "data": [  
    { "id": "{{docId1}}", "documentType": "ID_CARD", "status": "PENDING" },  
    {  
      "id": "{{docId2}}",  
      "documentType": "BANK_STATEMENT",  
      "status": "PENDING"  
    },  
    {  
      "id": "{{docId3}}",  
      "documentType": "EMPLOYMENT_LETTER",  
      "status": "PENDING"  
    }]
```

```
]  
}
```

Step 6.4: Mortgage Operations Officer Approves Each Document (Stage 1: Internal)

Adaeze performs the internal (Stage 1) review with `reviewParty: "INTERNAL"`:

```
POST  
{{mortgageServiceUrl}}/applications/{{applicationId}}/documents/{{docId1}}/review  
Authorization: Bearer {{adaezeAccessToken}}  
x-tenant-id: {{tenantId}}  
x-idempotency-key: adaeze-approve-doc-1-{{$timestamp}}  
Content-Type: application/json  
  
{  
  "status": "APPROVED",  
  "reviewParty": "INTERNAL",  
  "note": "ID verified successfully - internal review"  
}
```

(Repeat for docId2 and docId3)

Step 6.5: Bank Loan Officer Reviews Documents (Stage 2: Bank)

After internal approval, Nkechi (Access Bank's Loan Officer) performs the bank (Stage 2) review:

```
POST  
{{mortgageServiceUrl}}/applications/{{applicationId}}/documents/{{docId1}}/review  
Authorization: Bearer {{nkechiAccessToken}}  
x-tenant-id: {{tenantId}}  
x-idempotency-key: nkechi-approve-doc-1-{{$timestamp}}  
Content-Type: application/json  
  
{  
  "status": "APPROVED",  
  "reviewParty": "BANK",  
  "note": "ID verified - bank approval"  
}
```

(Repeat for docId2 and docId3)

Step 6.6: Bank Uploads Preapproval Letter

After approving documents, Nkechi uploads the bank's preapproval letter:

```
POST  
{{mortgageServiceUrl}}/applications/{{applicationId}}/phases/{{kycPhaseId}}/documents  
Authorization: Bearer {{nkechiAccessToken}}  
x-tenant-id: {{tenantId}}  
x-idempotency-key: nkechi-upload-preapproval-{{$timestamp}}  
Content-Type: application/json
```

```
{  
  "documentType": "PREAPPROVAL_LETTER",  
  "url": "https://s3.amazonaws.com/qshelter/applications/chidi-preapproval.pdf",  
  "fileName": "chidi-preapproval.pdf"  
}
```

Step 6.7: KYC Phase Completes

After both INTERNAL (Adaeze) and BANK (Nkechi) reviews are approved, the KYC documentation phase completes.

Phase 7: Downpayment (One-Time Payment)

Step 7.1: Generate Downpayment Installment

```
POST {{mortgageServiceUrl}}/applications/{{applicationId}}/phases/{{downpaymentPhaseId}}/installments  
Authorization: Bearer {{chidiAccessToken}}  
x-tenant-id: {{tenantId}}  
x-idempotency-key: chidi-generate-downpayment-{{$timestamp}}  
Content-Type: application/json  
  
{  
  "startDate": "{$isoTimestamp}"  
}
```

Expected Response (200 OK):

```
{  
  "success": true,  
  "data": {  
    "installments": [  
      {  
        "id": "{{downpaymentInstallmentId}}",  
        "amount": 8500000,  
        "dueDate": "...",  
        "status": "PENDING"  
      }  
    ]  
  }  
}
```

Step 7.2: Record Payment

```
POST {{mortgageServiceUrl}}/applications/{{applicationId}}/payments  
Authorization: Bearer {{chidiAccessToken}}  
x-tenant-id: {{tenantId}}  
x-idempotency-key: chidi-pay-downpayment-{{$timestamp}}
```

```
Content-Type: application/json

{
  "phaseId": "{{downpaymentPhaseId}}",
  "installmentId": "{{downpaymentInstallmentId}}",
  "amount": 8500000,
  "paymentMethod": "BANK_TRANSFER",
  "externalReference": "TRF-CHIDI-DOWNPAYMENT-001"
}
```

Expected Response (201 Created):

```
{
  "success": true,
  "data": {
    "id": "{{paymentId}}",
    "reference": "{{paymentReference}}",
    "amount": 8500000,
    "status": "PENDING"
  }
}
```

Step 7.3: Process Payment Confirmation (Webhook Simulation)

```
POST {{mortgageServiceUrl}}/applications/payments/process
Authorization: Bearer {{chidiAccessToken}}
x-tenant-id: {{tenantId}}
x-idempotency-key: process-downpayment-{{$timestamp}}
Content-Type: application/json

{
  "reference": "{{paymentReference}}",
  "status": "COMPLETED",
  "gatewayTransactionId": "GW-TRX-123456"
}
```

Expected: Downpayment phase COMPLETED, Verification phase auto-activates.

Phase 8: Final Documentation

Step 8.1: Mortgage Operations Officer Uploads Final Offer Letter

```
POST
{{mortgageServiceUrl}}/applications/{{applicationId}}/phases/{{verificationPhaseId}}/documents
Authorization: Bearer {{adaezeAccessToken}}
x-tenant-id: {{tenantId}}
x-idempotency-key: adaeze-upload-final-offer-{{$timestamp}}
Content-Type: application/json
```

```
{  
  "documentType": "FINAL_OFFER",  
  "url": "https://s3.amazonaws.com/qshelter/applications/chidi-final-offer.pdf",  
  "fileName": "chidi-final-offer.pdf"  
}
```

Step 8.2: Mortgage Operations Officer Completes Upload Step

```
POST  
{{mortgageServiceUrl}}/applications/{{applicationId}}/phases/{{verificationPhaseId}}/steps  
Authorization: Bearer {{adaezeAccessToken}}  
x-tenant-id: {{tenantId}}  
x-idempotency-key: adaeze-step-final-offer-{{$timestamp}}  
Content-Type: application/json  
  
{ "stepName": "Mortgage Operations Officer Uploads Final Offer" }
```

Step 8.3: Customer Signs Final Offer

```
POST  
{{mortgageServiceUrl}}/applications/{{applicationId}}/phases/{{verificationPhaseId}}/steps  
Authorization: Bearer {{chidiAccessToken}}  
x-tenant-id: {{tenantId}}  
x-idempotency-key: chidi-sign-final-offer-{{$timestamp}}  
Content-Type: application/json  
  
{ "stepName": "Customer Signs Final Offer" }
```

Expected: Verification phase COMPLETED, Mortgage phase auto-activates.

Phase 9: Mortgage Activation

Step 9.1: Generate Mortgage Installments

```
POST  
{{mortgageServiceUrl}}/applications/{{applicationId}}/phases/{{mortgagePhaseId}}/installments  
Authorization: Bearer {{chidiAccessToken}}  
x-tenant-id: {{tenantId}}  
x-idempotency-key: chidi-generate-mortgage-{{$timestamp}}  
Content-Type: application/json  
  
{  
  "startDate": "{$isoTimestamp}"  
}
```

Expected Response (200 OK):

```
{
  "success": true,
  "data": {
    "installments": [
      // 240 monthly installments
      {
        "id": "...",
        "amount": "...",
        "dueDate": "...",
        "installmentNumber": 1
      },
      { "id": "...", "amount": "...", "dueDate": "...", "installmentNumber": 2 }
      // ... (240 total)
    ],
    "summary": {
      "totalInstallments": 240,
      "monthlyPayment": "...",
      "totalAmount": 76500000,
      "totalInterest": "..."
    }
  }
}
```

Step 9.2: Sign and Activate Application

```
POST {{mortgageServiceUrl}}/applications/{{applicationId}}/sign
Authorization: Bearer {{chidiAccessToken}}
x-tenant-id: {{tenantId}}
x-idempotency-key: chidi-sign-application-{{$timestamp}}
```

Expected Response (200 OK):

```
{
  "success": true,
  "data": {
    "id": "{{applicationId}}",
    "status": "ACTIVE",
    "signedAt": "2025-01-XX..."
  }
}
```

Phase 10: First Mortgage Payment (Optional)

Step 10.1: Get Pending Installment

```
GET
{{mortgageServiceUrl}}/applications/{{applicationId}}/phases/{{mortgagePhaseId}}/instal
status=PENDING&limit=1
```

```
Authorization: Bearer {{chidiAccessToken}}
x-tenant-id: {{tenantId}}
```

Step 10.2: Record First Monthly Payment

```
POST {{mortgageServiceUrl}}/applications/{{applicationId}}/payments
Authorization: Bearer {{chidiAccessToken}}
x-tenant-id: {{tenantId}}
x-idempotency-key: chidi-mortgage-payment-1-{{$timestamp}}
Content-Type: application/json

{
  "phaseId": "{{mortgagePhaseId}}",
  "installmentId": "{{firstInstallmentId}}",
  "amount": "{{monthlyPaymentAmount}}",
  "paymentMethod": "BANK_TRANSFER",
  "externalReference": "TRF-CHIDI-MORTGAGE-001"
}
```

Missing Endpoints Summary

The following endpoints need to be implemented:

Property Service

Method	Endpoint	Description
POST	/properties/:id/variants	Create property variant
GET	/properties/:id/variants	List property variants
PUT	/properties/:id/variants/:variantId	Update variant
DELETE	/properties/:id/variants/:variantId	Delete variant
POST	/properties/:id/variants/:variantId/units	Create property unit
GET	/properties/:id/variants/:variantId/units	List units in variant
PUT	/units/:unitId	Update unit
DELETE	/units/:unitId	Delete unit
PATCH	/properties/:id/publish	Publish property
PATCH	/properties/:id/unpublish	Unpublish property

Mortgage Service

Method	Endpoint	Description
POST	/payment-methods/:id/document-rules	Bulk create document requirement rules

GET	/payment-methods/:id/document-rules	List document requirement rules
-----	-------------------------------------	---------------------------------

Postman Flow Design

The Postman Flow should be organized in these folders:

- QShelter E2E Flow
 - 1. Bootstrap
 - Bootstrap Tenant
 - Admin Login
 - 2. Customer Registration
 - Customer Signup
 - 3. Property Setup
 - Create Property
 - Create Variant
 - Create Unit
 - Publish Property
 - 4. Payment Configuration
 - Create Downpayment Plan
 - Create Mortgage Plan
 - Create Payment Method
 - Link to Property
 - Configure Document Rules
 - 5. Application Creation
 - Create Application
 - Verify Phase Amounts
 - Submit Application
 - Activate KYC Phase
 - 6. KYC Process
 - Upload Documents (x3)
 - Complete Upload Steps (x3)
 - Admin Review Documents (x3)
 - Complete Review Step
 - Sign Provisional Offer
 - 7. Downpayment
 - Generate Installment
 - Record Payment
 - Process Payment
 - 8. Final Documentation
 - Admin Upload Final Offer
 - Complete Upload Step
 - Sign Final Offer
 - 9. Mortgage Activation
 - Generate Installments
 - Sign Application
 - 10. Monthly Payment (Optional)
 - Record First Payment

Environment Variables

```
{  
    "userServiceUrl": "http://localhost:3002",  
    "propertyServiceUrl": "http://localhost:3003",  
    "mortgageServiceUrl": "http://localhost:3001",  
    "notificationServiceUrl": "http://localhost:3004",  
    "paymentServiceUrl": "http://localhost:3005",  
    "bootstrapSecret": "local-bootstrap-secret",  
    "tenantId": "",  
    "adaezeId": "",  
    "adaezeAccessToken": "",  
    "chidiId": "",  
    "chidiAccessToken": "",  
    "propertyId": "",  
    "variantId": "",  
    "unitId": "",  
    "downpaymentPlanId": "",  
    "mortgagePlanId": "",  
    "paymentMethodId": "",  
    "applicationId": "",  
    "kycPhaseId": "",  
    "downpaymentPhaseId": "",  
    "verificationPhaseId": "",  
    "mortgagePhaseId": ""  
}
```