🚀 Deployment Guide – Camunda 8 SaaS

This guide will help you deploy and test the **Loan Approval Process** using:

- ✅ Camunda Web Modeler (SaaS)
- ✅ Zeebe Java Client
- ✅ Angular App for User Task Form (External Task UI)

🧩 Prerequisites

1. ✅ Camunda SaaS Account
2. ✅ Java 17+ installed
3. ✅ Angular CLI + Node.js
4. ✅ Camunda Zeebe Java Client library
5. ✅ An API client (Postman or REST client for testing)

🔹 Step 1: Import and Deploy the BPMN

1. Go to **Camunda Web Modeler** → Create new project
2. Import loan-approval.bpmn from the ZIP
3. Validate the model:
    - Send Task: loanApplicationRequest (message name)
    - Receive Task: loanApplicationResponse (message name)
    - User Task: Form key → embedded:app:forms/loan-review-form.html
4. Click **Deploy**

🔹 Step 2: Get Camunda Cluster Details

Go to **Camunda Console → API → Zeebe**:

- **Cluster ID**
- **Client ID / Secret**
- **Region (e.g., bru-2)**

You'll need these for connecting via Java and REST APIs.

◆ Step 3: Configure Java Client

1. Use the provided LoanClient.java
2. Update these values in your code:

```
.gatewayAddress("your-cluster-id.bru-2.zeebe.camunda.io:443")

.clientId("your-client-id")

.clientSecret("your-client-secret")
```

3. Add Zeebe dependencies in your Maven pom.xml or Gradle.

Maven Example:

```
<dependency>
  <groupId>io.camunda</groupId>
  <artifactId>zeebe-client-java</artifactId>
  <version>8.5.0</version>
</dependency>
```

◆ Step 4: Start the Process via Java

Run the Java class LoanClient.java to:

- Start process instance
- Send loanApplicationResponse message to the receive task

Make sure the message has a **correlation key** matching a process variable (loanId).

◆ Step 5: Set Up Angular App

1. Install Angular:

```
npm install -g @angular/cli
```

2. Navigate to angular-app directory in the project

```
cd angular-app

npm install
```

3. Update task.service.ts with your Tasklist API endpoint:

```
private apiUrl = 'https://bru-2.tasklist.camunda.io/<your-cluster-id>';
```

4. Run Angular app:

```
ng serve
```

5. Access it at http://localhost:4200

🔹 Step 6: Complete the Task via Angular

- Angular app will fetch the external task assigned
- Form will show applicant info and allow approve/reject
- On submit, it will complete the task and forward flow to gateway

---

🔹 Step 7: Monitor via Camunda Console

- Go to **Operate** panel
- Track:
  - Process instance progress
  - Variables and decision paths
  - Task state

✅ Testing Checklist

Task	**Status**
BPMN deployed in SaaS	✅
Java client starts process	✅
Message reaches receive task	✅
User task shown in Angular form	✅
Task completed and flow finishes	✅
