

Design Services Integration Guide

Overview

The Design Services module integrates your Construction CRM Platform with an external AI Design & Rendering Platform for automated architectural design tasks. This guide explains how the integration works and how to use it.

External AI Platform

Webhook URL: `https://gmllorlxfsxmsejhsjpa.supabase.co/functions/v1/n8n-orders-webhook`

Authentication: Uses webhook secret for secure communication

Callback URL: `https://development.abacusai.app/api/webhooks/design-callback`

Configuration

Environment Variables

The webhook secret is stored in the `.env` file:

```
DESIGN_WEBHOOK_SECRET=31851db11bdbfef9a8f5e433769d75a0416d9f922253089b1c08619ad70df2f7
```

Important: Keep this secret secure and never commit it to version control.

How It Works

1. Create a Design Request

Users create design requests through the web interface:

- Navigate to **Design Services** → **New Request**
- Fill in client information (name, email, phone)
- Provide project details:
 - Project Name
 - Project Type (Residential, Commercial, Industrial, etc.)
 - Timeline (Rush, Standard, Flexible)
 - Budget
 - Description
 - Requirements
 - Site Details

2. Submit Tasks to AI Platform

From the design request detail page:

- Click on the **Tasks** tab
- Select one or more task types:
 - **Architectural Design** - Complete architectural design drawings and plans
 - **Structural Engineering** - Structural analysis and engineering

- **MEP Design** - Mechanical, Electrical, and Plumbing design
- **Interior Design** - Interior space planning and design
- **Landscape Design** - Exterior and landscape architecture
- **3D Rendering** - Photorealistic 3D visualization
- **3D Modeling** - Detailed 3D models and BIM
- **Construction Documentation** - Complete construction document set

- Click **Submit Task(s) to AI Platform**

3. Task Processing

When tasks are submitted:

1. The system creates task records in the database with status “Pending”
2. Each task is sent to the external AI platform via webhook
3. The AI platform processes the task and returns:
 - External Task ID
 - Initial status (usually “Queued”)
 - Estimated completion time (if available)
4. Tasks are marked as “Queued” or “Failed” based on the response

4. Receiving Updates

The external AI platform sends updates back via webhook callback:

- **Endpoint:** /api/webhooks/design-callback
- **Method:** POST
- **Authentication:** Uses DESIGN_WEBHOOK_SECRET

Status updates include:

- **Queued** - Task is in queue waiting to be processed
- **Processing** - AI is actively working on the task
- **Completed** - Task finished successfully with results
- **Failed** - Task failed with error message

5. Viewing Results

From the design request detail page:

- **Tasks Tab** - View all tasks, their status, and results
- **AI Analysis Tab** - View AI-generated design analysis:
 - Design Concept
 - Style Recommendations
 - Spatial Layout
 - Material Suggestions
 - Sustainability Features
 - Visualization Prompt

Webhook Payload Format

Outgoing Payload (to External AI Platform)

```
{
  "action": "create_design_task",
  "requestId": "<uuid>",
  "requestNumber": "DR-2026-0001",
  "taskId": "<uuid>",
  "taskType": "Architectural",
  "title": "Architectural Design Task",
  "description": "Architectural Design for PA project",
  "priority": "Normal",
  " projectName": "PA project",
  "projectType": "Residential",
  "requirements": "Design requirements here...",
  "siteDetails": "Site details here...",
  "budget": 1500000,
  "timeline": "Flexible",
  "clientName": "John Doe",
  "clientEmail": "john@example.com",
  "clientPhone": "+1234567890",
  "designConcept": "AI-generated design concept",
  "styleRecommendations": "AI-generated style recommendations",
  "spatialLayout": "AI-generated spatial layout",
  "materialSuggestions": "AI-generated material suggestions",
  "sustainabilityFeatures": "AI-generated sustainability features",
  "visualizationPrompt": "AI-generated visualization prompt",
  "callbackUrl": "https://development.abacusai.app/api/webhooks/design-callback",
  "callbackSecret": "<secret>"
}
```

Incoming Callback (from External AI Platform)

```
{
  "callbackSecret": "<secret>",
  "externalTaskId": "<external_task_id>",
  "taskId": "<our_task_id>",
  "requestId": "<our_request_id>",
  "status": "completed",
  "progress": 100,
  "resultUrl": "https://example.com/result.pdf",
  "resultFiles": [
    "https://example.com/file1.pdf",
    "https://example.com/file2.dwg"
  ],
  "resultData": {
    "custom": "data"
  },
  "startedAt": "2026-01-08T10:00:00Z",
  "completedAt": "2026-01-08T12:30:00Z"
}
```

Status Mapping

External statuses are mapped to internal statuses:

| External Status | Internal Status | Description |
|-------------------------|-----------------|----------------------------|
| queued, pending | Queued | Task is waiting in queue |
| processing, in_progress | Processing | AI is working on task |
| completed, success | Completed | Task finished successfully |
| failed, error | Failed | Task failed with error |
| cancelled, canceled | Cancelled | Task was cancelled |

Design Request Status Flow

1. **Draft** - Initial state when request is created
2. **Submitted** - Request submitted with tasks selected
3. **AIProcessing** - Tasks successfully sent to AI platform
4. **Rendering** - AI is actively processing tasks
5. **UnderReview** - Tasks completed, awaiting review
6. **ClientReview** - Sent to client for feedback
7. **RevisionRequired** - Client requested changes
8. **Approved** - Client approved the designs
9. **InDesign** - Working on revisions/refinements
10. **Completed** - All work finished
11. **Cancelled** - Request cancelled

Troubleshooting

Task Fails with “Unknown action: undefined”

Cause: The external AI platform expects an `action` field in the payload but it was missing or undefined.

Solution: Ensure the payload includes `action: 'create_design_task'` field. This has been fixed in the latest version.

Task Stuck in “Pending” Status

Possible causes:

1. External AI platform is down or not responding
2. Network connectivity issues
3. Invalid webhook URL
4. Authentication failure

Check:

- Review server logs for error messages
- Verify `DESIGN_WEBHOOK_SECRET` is set correctly
- Confirm webhook URL is accessible
- Check task's `errorMessage` field in database

Callback Not Received

Possible causes:

1. Callback URL is incorrect or not accessible
2. Callback secret mismatch
3. External platform not configured to send callbacks

Check:

- Verify callback URL in outgoing payload
- Confirm `DESIGN_WEBHOOK_SECRET` matches on both sides
- Check webhook callback logs: `GET /api/webhooks/design-callback`

No Results After Task Completion

Check:

- View task details in Tasks tab
- Check if `resultUrl` or `resultFiles` fields are populated
- Review `externalResponse` field in database for full callback data

API Endpoints

Design Requests

- `GET /api/design-requests` - List all design requests (with filters)
- `POST /api/design-requests` - Create new design request
- `GET /api/design-requests/[id]` - Get design request details
- `PATCH /api/design-requests/[id]` - Update design request
- `DELETE /api/design-requests/[id]` - Delete design request (admin only)
- `POST /api/design-requests/[id]/submit` - Submit tasks to AI platform

Webhook Callback

- `POST /api/webhooks/design-callback` - Receive updates from AI platform
- `GET /api/webhooks/design-callback` - Verify endpoint is active

Database Schema

DesignRequest Model

```

model DesignRequest {
    id          String          @id @default(cuid())
    requestNumber String         @unique

    // Client Information
    clientName   String
    clientEmail  String
    clientPhone  String?

    // Project Details
    projectName  String
    projectType  DesignProjectType
    projectId    String?

    // Request Details
    description  String
    requirements String
    siteDetails  String?
    budget       Decimal?
    timeline     DesignTimeline

    // Status Tracking
    status        DesignRequestStatus      @default(Draft)
    submittedAt  DateTime?
    completedAt  DateTime?

    // AI Analysis Results
    designConcept String?
    styleRecommendations String?
    spatialLayout  String?
    materialSuggestions String?
    sustainabilityFeatures String?

    // Design Visualization
    visualizationUrl String?
    visualizationPrompt String?

    // Client Feedback
    clientFeedback String?
    revisionNotes  String?

    // External Platform Integration
    externalRequestId String?
    externalStatus   String?
    lastSyncAt     DateTime?

    // Relationships
    tasks          DesignTask[][]
    project        Project?        @relation(fields: [projectId], references: [id])
    createdBy      User           @relation(fields: [createdById], references: [id])
    createdById   String
    createdAt      DateTime        @default(now())
    updatedAt      DateTime        @updatedAt
}

```

DesignTask Model

```

model DesignTask []
  id          String      @id @default(cuid())
  // Task Details
  taskType    DesignTaskType
  title       String
  description String
  priority    String      @default("Normal")
  // Status Tracking
  status      TaskStatus  @default(Pending)
  startedAt   DateTime?
  completedAt DateTime?
  // External Platform Integration
  sentToExternalPlatform Boolean @default(false)
  externalTaskId        String?
  externalStatus         String?
  lastSyncAt            DateTime?
  // Results
  resultUrl     String?
  resultFiles   String[]
  resultData    String?    // JSON
  // Error Handling
  errorMessage  String?
  // Webhook Communication
  webhookPayload String?    // JSON
  webhookResponse String?    // JSON
  externalResponse String?    // JSON
  // Relationships
  designRequest  DesignRequest  @relation(fields: [designRequestId], references: [id], onDelete: Cascade)
  designRequestId String
  createdAt     DateTime      @default(now())
  updatedAt     DateTime      @updatedAt
}

```

Best Practices

- 1. Always provide detailed requirements** - The more information you provide, the better the AI can understand and fulfill your design needs
- 2. Select appropriate timeline** - Use “Rush” for urgent projects, “Standard” for normal projects, and “Flexible” when you have more time
- 3. Monitor task progress** - Check the Tasks tab regularly to see the status of your submitted tasks
- 4. Review AI Analysis** - Before submitting tasks, review the AI Analysis tab to see initial recommendations
- 5. Handle failures gracefully** - If a task fails, check the error message and resubmit if necessary
- 6. Link to projects** - Link design requests to existing projects for better organization and tracking

Security Considerations

1. **Webhook Secret** - Keep the `DESIGN_WEBHOOK_SECRET` secure and rotate it periodically
2. **Callback Verification** - All incoming callbacks are verified using the webhook secret
3. **API Authentication** - All API endpoints require user authentication via NextAuth
4. **Role-Based Access** - Only authorized users can create, view, and manage design requests

Support

For issues or questions:

- Check server logs for error details
- Review task error messages in the UI
- Contact your system administrator