



Print and Signage Business Management

Product overview and user guide

Version 1.0 | 28 October 2026

Manage enquiries, print jobs, artwork, production, GST documents, inventory, attendance and payroll in one role based workspace. A single sign in gives each user the menus and records appropriate to their account.

This guide explains how to set up an installation, run everyday work, manage access and connect an existing website. The application opens at the login screen. The business can keep its PHP, Next.js or WordPress website and use the API to display services and send enquiries into the CRM.

What the application provides

Area	Main capabilities
Sales and customers	Contact leads, quote requests, customer history, quotations and invoices
Jobs and delivery	Job stages, designer assignment, artwork versions, approvals, production and delivery challans
People and costs	Staff roles, attendance, overtime, editable payslips, employee self service and expenses
External websites	Published services, quote form schema, secure enquiry capture and usage reports

Find the right section

Pages	Topic
2 to 3	Getting started and the complete default access matrix
4 to 5	Sales, documents, design, production and stock
6	Attendance, payslip corrections and employee access
7 to 14	Website APIs, service cards, quote forms, CRM capture and monitoring
15	Settings, troubleshooting and client handover

Use the actual installation URL supplied by the administrator. Examples in this guide use a fictional host and contain no working credentials.



Getting started and account access

Administrator setup

1. Open the application folder. A new installation starts its guided installer; an installed application opens sign in. Complete readiness checks, database configuration, administrator details and installation locking. Select demo data only when needed.
2. Set the business name, contact information, address, GSTIN, state, bank details, document numbering, terms, logo and authorised signature in Settings. Check the saved logo on an invoice and a payslip.
3. Configure SMTP host, port, TLS or SSL, username, password and sender address. Check the email queue and test delivery before inviting real users.
4. Create staff in Staff, assign the appropriate role, set payroll details and use Send password link. Search by name, email, phone, employee ID or designation; filter by role/status and choose a page size.
5. Review Settings then Access control. Default role capabilities are listed on page 3. Restrict a role where needed; the editor does not grant unrestricted access to every module.

Staff sign in

All staff use the same /login page. Their role determines the dashboard items. Each user can manage their own password from their profile; Forgot password sends a reset link. Accounts can be activated or deactivated by an authorised administrator. Staff cannot deactivate their own account through the status toggle.

Customer invitations and email confirmation

When saving a customer with an email address, enable the customer invitation option. The queued email invites the customer to confirm their email and choose a password. The invitation expires after 24 hours and is single use. The customer then signs in at the same login page.

Staff profiles include unique employee codes, validated mobile numbers and optional bank account/IFSC details. Admin can deactivate staff or delete them from the directory while retaining linked attendance, payroll and job history.

Customer profiles provide invitation/resend, password link and account enable/disable controls. Updating a linked email address requires verification again. Creating an API enquiry does not automatically activate or invite a customer account. Public self registration is not enabled.

What customers see

The customer workspace contains their own jobs and status, issued quotations/invoices/challans, quotation request history and a service quote request form. Draft business documents and other customers' records are unavailable. Internal job notes are excluded. Customer accounts have no staff management, payroll or API token access.



Roles and default access

Owner or Admin: full business management, staff and settings access, all API integrations, and exclusive authority to reopen paid payroll and attendance. Admin access cannot be removed in the permission editor.

The matrix shows the defaults for the other staff roles. **M** means manage, **V** means view, **O** means own or assigned records, and **-** means no module access. Saved permissions may restrict these defaults.

Module	Reception	Designer	Production	Accountant	Developer
Overview	V	V	V	V	-
Jobs	M	O	V	-	-
Customers	M	-	-	V	-
Design Studio	-	M*	-	-	-
Production	-	-	M	-	-
Quote Requests	M	-	-	-	-
Quotations	M	-	-	M	-
Invoices	-	-	-	M	-
Delivery Challans	M	-	V	V	-
Services	V	-	-	-	-
Products	V	-	-	V	-
Inventory	-	-	M	V	-
Attendance	O	O	O	V	-
Salary management	-	-	-	M	-
My Payslips	O	O	O	O	-
Expenses and Analytics	-	-	-	M / V	-
Staff and Settings	-	-	-	-	-
Contact Leads	M	-	-	-	-
API and Integrations	-	-	-	-	M*

Assignment limits: designers work on assigned jobs. Developers can access only integrations assigned to their account; they cannot create integrations, change scopes or browse business modules. Admin creates integrations and controls assignment.

Employee payslips: My Payslips exposes only the signed in employee's paid slips, including print/PDF. Draft and reopened slips are hidden. Accountants with salary management access can additionally manage business payroll. All signed in users retain their own password/profile controls.

Customers: own customer workspace only, as described on page 2. They do not inherit any staff role.



Sales enquiries and business documents

From a website enquiry to the sales team

1. Open Quote Requests for service quotation enquiries, or Contact Leads for general contact submissions. API submissions appear here rather than in a public website interface.
2. For a quote request, review customer details, selected service, quantity, project requirements and any configured specification fields. Move the request through new, reviewed, quoted or closed as the work progresses.
3. For a contact lead, review the message and source, update new/in progress/closed status, and convert it to a customer when appropriate. Check existing customer matches to avoid duplicate records.
4. Create the formal quotation with the correct customer, line items, quantities, discounts, GST and terms. A quote request itself is an enquiry, not an issued quotation or a confirmed order.

Quotations and invoices

Create and review documents before issuing them. GST is calculated using the configured business and customer states, with local CGST/SGST or interstate IGST as applicable. Confirm the state, GSTIN, HSN, quantities and rates before sharing.

An accepted quotation can be converted to a job or invoice using its available actions. The application retains the source relationship. Review the copied invoice before issuing it. Record customer payments in the invoice workflow so paid amounts, balances and the revenue graph reflect actual receipts.

Open a document to use Print / Save PDF or the WhatsApp sharing action where available. Supply a valid customer phone number and complete the send in WhatsApp. This action is a user initiated sharing flow, not an unattended WhatsApp campaign or bulk messaging service.

Delivery challans and customer history

Create delivery challans using the relevant job/customer details and review quantities, delivery address and vehicle information. Keep dispatch paperwork consistent with the actual material sent. The customer record brings together linked work and business documents for follow up.

Revenue overview

The dashboard revenue graph is based on recorded customer payments over its displayed period. Creating an invoice without recording a receipt does not add collected revenue. Outstanding amounts come from unpaid invoice balances. Always compare the selected dates and stored payments when investigating an unexpected total.



Design production stock and expenses

Job tracking

Create a job with its customer, title, product or service context, quantity, dimensions/material information, deadline, priority and assigned designer. Update the stage as work moves through design, approval, printing, finishing, readiness, dispatch and delivery. Job history records status changes and notes.

Design Studio and artwork approval

1. The assigned designer opens the job in Design Studio and uploads a draft artwork version or supplies a supported artwork link.
2. Submit the draft for review. Submission updates the job to pending approval; uploading a file alone is not the approval step.
3. The owner/admin or reception records approval or rejection for submitted artwork. Designers do not approve their own submitted artwork through the approval action.
4. If changes are requested, upload a new version and submit it again. Keep each version tied to the job instead of replacing the historical review trail.

Production and delivery

Production staff use the Production workspace to monitor stages and move work forward using their permitted actions. Check the latest approved artwork, specifications and quantities before printing. Update the job when finishing is complete, when it is ready, and when it is dispatched or delivered.

Products and inventory

Products hold the operational rate card. Services are the separately managed catalogue published through the website API. Do not assume that creating a product automatically publishes a service card.

Use product active/inactive toggles to control available products. Inventory tracks materials, categories, SKUs, size/unit, minimum stock, cost and supplier information. Use its filters to locate items and record stock movements to keep quantities and low stock checks meaningful. Review balances before promising production capacity.

Expenses and reporting

Record an expense with its date, amount, payment method, category/tag, optional linked job, notes and receipt. Use consistent categories for meaningful reporting. Analytics and expense reports support follow up on costs and business performance; their usefulness depends on keeping jobs, receipts, expenses and payroll records current.

Access follows the role matrix. For example, the default production role can manage inventory, whereas the accountant can view it. Only an authorised manager should change operational stock records.



Attendance payroll and My Payslips

Record and correct attendance

Choose the month and employee in Attendance. Record status, clock in/out, break minutes and notes. Overtime is calculated after breaks above the configured shift hours. An end time earlier than the start is treated as an overnight shift.

Own attendance access is limited to marking the signed in user present today. A user cannot overwrite an existing attendance record or approve their own leave/corrections. An attendance manager can use Edit on an existing record. Paid payroll locks attendance for that employee and month.

Generate and edit a payslip

1. Check the staff salary, overtime rate, PF/ESI eligibility and Settings payroll rules. Complete attendance first: missing paid days can reduce salary.
2. Open Salary, select the month, choose Generate payslip and add any allowance or other deductions. Open View slip to review the breakdown.
3. For a draft, use Edit payslip to change basic pay, paid days, loss of pay, overtime hours/pay, allowance, PF, ESI, other deductions and notes. Enter a correction reason. Net pay is calculated on save; this does not change the staff salary profile.
4. Alternatively, use Recalculate from attendance. Previous correction details stay in history. After review, record the payment date and mark the salary paid.

Reopen a paid month as administrator

Open the paid slip and expand Reopen payslip & attendance, enter a reason and reopen. Attendance also links to this action for a locked month. Only Owner/Admin can reopen; the accountant cannot. The slip returns to draft and its current paid date clears, while the previous payment date, amounts and correction reason remain in history.

Correct the attendance, then regenerate the payslip or review and save all amounts. The application blocks marking a stale slip paid until this review is complete. Reopening does not reverse a bank payment. Compare the old payment with the corrected amount before deciding how to settle the difference.

Employee access

Reception, Designer, Production and Accountant accounts have My Payslips. Select a month, open a paid slip and use Print / Save PDF. Employees cannot access another person's slip or draft/reopened slips. Payroll managers retain their wider management permissions. A reopened slip becomes available in self service again after it is marked paid.



How external websites connect

The API lets a business keep its existing website while using this application as the operational CRM. The website can show published services, build a quote form from the configured fields and send quote/contact enquiries to the team. No public website template is bundled with the application.

The connection flow

1. An admin publishes services in Services and creates an integration with the minimum required scopes.
2. The website server requests the service catalogue from the CRM API and renders its own service cards. Each service response includes a quote form definition.
3. The visitor fills in a form on the website. The browser submits it to that website's own server handler.
4. The website server validates/spam checks the submission and sends JSON to the CRM with its bearer token and an idempotency key.
5. The CRM stores the enquiry in Quote Requests or Contact Leads. The sales team follows the normal sales workflow.

API endpoint reference

Method and path	Required scope	Use
GET /api/v1/ping	Any active token	Check connection and integration name
GET /api/v1/services	services:read	Published service cards and form definitions
GET /api/v1/quote-form?service_id=1	services:read	Form definition for one service
GET /api/v1/products	products:read	Active operational rate card
POST /api/v1/quote-requests	quotes:write	Capture a quotation enquiry
POST /api/v1/contact-leads	contacts:write	Capture a general contact enquiry

Paths are relative to the installation folder. For example, an installation at <https://crm.example.com/print-app> uses <https://crm.example.com/print-app/api/v1/services>. Older `api.php?route=v1/...` endpoints remain supported.

Important boundaries

This is a server to server integration. Keep tokens out of HTML, public JavaScript and browser storage. There is no browser CORS API access. A developer must provide the website presentation and server handler. The current API does not expose payroll, internal jobs, customer documents or an order tracking endpoint.



Build service cards and quote forms

Load the catalogue

```
curl "https://crm.example.com/print-app/api/v1/services" \
-H "Authorization: Bearer YOUR_TOKEN"
```

Read the response data array. Service records provide ID, slug, name, description, icon, optional image URL, features, starting price, currency, price unit/label and quote_form. Services must be published; a service linked to an inactive product is not returned. Use service_id or slug to select one service. Catalogue pagination supports limit 1 to 100 and offset up to 1000000.

The quote_form object provides action, method, content type, fields, defaults, honeypot and idempotency header information. Honour its required fields and validation rather than hardcoding one form for every service. Optional specifications may include width, height, material, finishing, deadline and delivery address, depending on service configuration.

Send a quotation enquiry

```
curl "https://crm.example.com/print-app/api/v1/quote-requests" \
-H "Authorization: Bearer YOUR_TOKEN" \
-H "Content-Type: application/json" \
-H "Idempotency-Key: quote-20261028-0001" \
--data '{
  "service_id": 1,
  "name": "Priya Rao",
  "email": "priya@example.com",
  "phone": "+919000000001",
  "company": "Example Retail",
  "quantity": 20,
  "message": "Please quote for our outdoor banner project.",
  "source_url": "https://www.example.com/services",
  "external_reference": "website-enquiry-0001",
  "website": ""
}'
```

Replace service_id 1 with a currently published service returned by the catalogue. Add only specification fields enabled for that service. Quantity must be positive with at most four decimal places. Deadline uses YYYY-MM-DD. The website honeypot must remain empty; do not use it for the visitor's real company website.

What happens in the CRM

A valid quote submission creates a Quote Request and links or creates a customer by email. Service/contact specifications are retained with the enquiry. If multiple customers share the same email, the request can fail with a conflict for the team to resolve. The response contains a reference and request_id. Display a success message only after a successful response; it is not an automatic price quote or order confirmation.



PHP example for service cards

These examples run on your external website, not inside the CRM installation. They require PHP 8 with cURL and sessions. Configure PRINT_API_BASE as your installation URL ending in /api/v1, and PRINT_API_TOKEN as its private bearer token. Use services:read and quotes:write scopes. Never commit the token to public source control.

Create api_client.php on the website server

```
<?php
function crm(string $path, ?array $body = null,
    ?string $key = null): array {
    $base = rtrim(getenv('PRINT_API_BASE') ?: '', '/');
    $token = getenv('PRINT_API_TOKEN') ?: '';
    if (!$base || !$token) throw new RuntimeException('API not set');
    $ch = curl_init($base . '/' . $path);
    $headers = ['Authorization: Bearer ' . $token];
    if ($body !== null) {
        $headers[] = 'Content-Type: application/json';
        $headers[] = 'Idempotency-Key: ' . $key;
        curl_setopt($ch, CURLOPT_POST, true);
        curl_setopt($ch, CURLOPT_POSTFIELDS, json_encode($body));
    }
    curl_setopt_array($ch, [CURLOPT_RETURNTRANSFER => true,
        CURLOPT_TIMEOUT => 15, CURLOPT_HTTPHEADER => $headers]);
    $raw = curl_exec($ch);
    $status = curl_getinfo($ch, CURLINFO_HTTP_CODE);
    curl_close($ch);
    $json = is_string($raw) ? json_decode($raw, true) : null;
    if ($status < 200 || $status >= 300 || !is_array($json)) {
        error_log('CRM status ' . $status . ' request ' .
            ($json['request_id'] ?? 'unavailable'));
        throw new RuntimeException('Enquiry service unavailable');
    }
    return $json;
}

function esc($v): string {
    return htmlspecialchars((string)$v, ENT_QUOTES, 'UTF-8');
}
```

Render services.php with a Get Quote link

```
<?php require 'api_client.php';
$catalogue = crm('services?limit=50');
foreach ($catalogue['data'] as $service): ?>
<article class="service-card">
    <h2><?= esc($service['name']) ?></h2>
    <p><?= esc($service['description']) ?></p>
    <p><?= esc($service['price_label']) ?></p>
    <a href="quote.php?service_id=<?= (int)$service['id'] ?>">
        Get Quote
    </a>
</article>
<?php endforeach; ?>
```

Style service-card to match your website. Render features or images as needed, escaping text and validating image URLs. Request subsequent pages if the catalogue contains more than 50 services. Starting prices are indicative service labels, not a final calculated quotation.



PHP example for the quote option

Get Quote opens the selected service form. Fetch the current schema from the CRM so your website follows each service's enabled fields. The following quote.php example generates the fields and sends the visitor to your local submit-quote.php handler. The bearer token is never included in the HTML.

Create quote.php

```
<?php
session_start();
require 'api_client.php';
$id = filter_input(INPUT_GET, 'service_id', FILTER_VALIDATE_INT);
if (!$id || $id < 1) { http_response_code(400); exit('Bad service'); }
$schema = crm('quote-form?service_id=' . $id)['data'];
$_SESSION['csrf'] ??= bin2hex(random_bytes(32));
$key = 'quote-' . bin2hex(random_bytes(16));
$_SESSION['quote_forms'][$key] = ['service_id' => $id,
                                'created' => time()];

?>
<h1>Request a quote for <?= esc($schema['service_name']) ?></h1>
<form method="post" action="submit-quote.php">
  <input type="hidden" name="csrf" value="<?= esc($_SESSION['csrf']) ?>">
  <input type="hidden" name="submission" value="<?= esc($key) ?>">
  <?php foreach ($schema['fields'] as $field):
    $name = $field['name'];
    $type = $field['type'];
    $value = $schema['defaults'][$name] ?? '';
    $required = !empty($field['required']) ? 'required' : '';
  ?>
  <label><?= esc($field['label']) ?>
    <?php if ($type === 'textarea'): ?>
      <textarea name="<?= esc($name) ?>" <?= $required ?>></textarea>
    <?php else:
      $type = in_array($type, ['text', 'email', 'tel', 'number', 'date'])
        ? $type : 'text'; ?>
      <input type="<?= esc($type) ?>" name="<?= esc($name) ?>"
        value="<?= esc($value) ?>" <?= $required ?>
      <?php foreach (['min', 'max', 'step', 'maxlength'] as $attr):
        $source = $attr === 'maxlength' ? 'max_length' : $attr;
        if (isset($field[$source])):
          ?> <?= $attr ?>="<?= esc($field[$source]) ?>"<?php endif;
        endforeach; ?>
      <?php endif; ?>
    </label>
  <?php endforeach; ?>
  <div hidden aria-hidden="true">
    <input name="website" tabindex="-1" autocomplete="off">
  </div>
  <button type="submit">Send quote request</button>
</form>
```

Use secure session cookies on HTTPS and add your website's anti-spam/rate-limit checks. Expire old quote_forms session entries and limit outstanding forms in production. Browser validation improves usability; repeat validation on the server. The next page shows the core submission checks.



PHP example for sending enquiries

submit-quote.php reads the service ID from the server session, not a visitor supplied hidden service value. It sends only fields defined by the CRM schema and converts numeric inputs to JSON numbers. Reuse the same stored key and payload when retrying after a connection failure.

```
<?php
session_start();
require 'api_client.php';
if ($_SERVER['REQUEST_METHOD'] !== 'POST') {
    http_response_code(405); exit;
}
$csrf = is_string($_POST['csrf'] ?? null) ? $_POST['csrf'] : '';
if (!hash_equals($_SESSION['csrf'] ?? '', $csrf) || !$csrf) {
    http_response_code(419); exit('Reload the quote form');
}
$key = is_string($_POST['submission'] ?? null)
    ? $_POST['submission'] : '';
$pending = $_SESSION['quote_forms'][$key] ?? null;
if (!$pending || time() - $pending['created'] > 3600) {
    http_response_code(422); exit('Form expired; start again');
}
if (($_POST['website'] ?? '') !== '') {
    http_response_code(422); exit('Submission rejected');
}
try {
    $schema = crm('quote-form?service_id=' .
        (int)$pending['service_id'])['data'];
    $body = ['service_id' => (int)$pending['service_id'],
        'website' => ''];
    foreach ($schema['fields'] as $field) {
        $name = $field['name']; $v = $_POST[$name] ?? '';
        if (!is_string($v)) throw new RuntimeException('Bad value');
        $v = trim($v);
        if (!empty($field['required']) && $v === '')
            throw new RuntimeException('Missing required field');
        if ($v === '') continue;
        if ($field['type'] === 'number') {
            if (!is_numeric($v) || !is_finite((float)$v))
                throw new RuntimeException('Bad number');
            $v = (float)$v;
        }
        $body[$name] = $v;
    }
    if (isset($pending['payload'])) $body = $pending['payload'];
    $_SESSION['quote_forms'][$key]['payload'] = $body;
    $result = crm('quote-requests', $body, $key);
    unset($_SESSION['quote_forms'][$key]);
    echo 'Request received. Reference: ' .
        esc($result['data']['reference']);
} catch (Throwable $e) {
    http_response_code(422);
    echo 'Could not submit. Check details or retry this submission.';
}
```

This is a minimal integration example. Add field length/range/email validation, website rate limiting, accessible error feedback, abuse controls and a POST/redirect/GET success page before launch. Preserve the pending payload for an uncertain retry; do not edit it under the same key. Clear it and start a new form only when the earlier outcome is known.



Where website enquiries appear in the CRM

Example from service card to sales follow up

A visitor selects Banner Printing on your website, clicks Get Quote and submits their contact details, quantity and requirements. Your website server calls POST /api/v1/quote-requests. A successful CRM response contains a reference, status, kind and request_id. Show the reference to the visitor; retain request_id for technical diagnosis.

```
{
  "data": {
    "reference": "API-<generated reference>",
    "status": "new",
    "kind": "quote"
  },
  "request_id": "<generated request id>"
}
```

The example above is illustrative; use the actual values returned by the API. Enquiry references are not job tracking numbers.

Website information	CRM destination or behavior
Name, email, phone and company	Customer match by email. If no match exists, a customer record is created. Existing customer details are not silently overwritten.
Selected service and configured form fields	Quote Request retains service ID, service snapshot and submitted specifications/contact snapshot.
Quantity and project message	Quote Request quantity and description, with initial status new.
General contact form with subject	POST contact-leads creates a Contact Leads record instead of a quotation request.
Token, endpoint and response status	API usage metadata for the authorised admin or assigned developer; enquiry bodies are not copied into usage logs.

What reception or admin does next

1. Sign in and open Quote Requests. Locate the customer, service and requirements. Move the request from new to reviewed as you assess it.
2. Prepare a formal quotation with final rates, quantities, tax and terms. Update the enquiry to quoted or closed as appropriate. The API does not automatically issue a priced quotation or create a paid order.
3. When the quotation is accepted, use its available conversion actions to create the job or invoice and continue production and billing.
4. For a Contact Lead, review the message, update its follow-up status and convert to a customer when suitable.

Website API capture does not require the visitor to hold a CRM login and does not automatically send a customer invitation. Invite the customer separately when portal access is wanted. Multiple matching customer emails can return a conflict; resolve the records instead of blindly retrying.



Contact capture validation and retries

Send a contact lead

```
curl "https://crm.example.com/print-app/api/v1/contact-leads" \
-H "Authorization: Bearer YOUR_TOKEN" \
-H "Content-Type: application/json" \
-H "Idempotency-Key: contact-20261028-0001" \
--data '{
  "name": "Arun Shah",
  "email": "arun@example.com",
  "phone": "+919000000002",
  "subject": "Shop signage enquiry",
  "message": "Please contact me about signage for a new shop.",
  "source_url": "https://www.example.com/contact",
  "website": ""
}'
```

Contact submissions appear in Contact Leads. The CRM requires name, valid email, subject and a message of at least 10 characters. The service quote form may impose additional requirements such as phone. JSON bodies must be objects, use Content-Type application/json and stay within 64 KiB. Unsupported fields are rejected.

Retry without creating duplicates

Create one Idempotency-Key per logical submission. It must contain 8 to 100 letters, digits, dots, colons, underscores or hyphens. Reuse the same token, key, endpoint and unchanged JSON payload after a network failure. An identical retry returns the saved result with Idempotency-Replayed: true. Reusing a key for different content or another endpoint produces a conflict. A newly rotated token is not a substitute for checking an uncertain earlier submission.

Handle errors in the website server

HTTP status	Action
401	Check full bearer token, expiry, revocation, owner status and header forwarding.
403	Check token scope and allowed source server IPs.
409	Check reused idempotency key or duplicate customer email records.
413 or 415	Reduce JSON payload size or correct Content-Type.
422	Show a useful validation message; correct required fields or key format.
426	Use HTTPS outside permitted localhost testing.
429	Wait for Retry-After before retrying; reduce request volume.
503	Check CRM availability and database; retry safely using the same key.

Log the response request_id for diagnosis, not the bearer token or full personal enquiry body. Apply spam protection and request throttling to the website's own form handler as well as the CRM checks.



Manage API credentials and usage

Create an integration as administrator

1. Open API & Integrations then Bearer tokens. Use a separate named integration for each connected website.
2. Enter the website URL and assign an active admin or developer owner. Select only the needed scopes: services:read and quotes:write for service enquiries, contacts:write for contact forms, and products:read only when needed.
3. Set the request limit, optional expiry and optional allowed server IPs. The supported range is 1 to 600 requests per minute; IP restrictions accept exact IPv4/IPv6 addresses, not domain names or CIDR ranges.
4. Save, copy the full bearer token, and store it in the website server's environment or protected configuration. Confirm access with ping, then test a service read and one enquiry.
5. Open Usage report and confirm the token is recognised and the request succeeded. Check that the enquiry appears in the correct CRM module.

Who can see the token

Admin can see all integrations and their tokens. Developer accounts see only assigned integrations, technical documentation and technical usage reports. Reception, Designer, Production, Accountant and Customer accounts cannot view these tokens. Only admins can create integrations or edit assignment, scopes and restrictions.

Assigned developers and admins can copy tokens and rotate or revoke integrations within their authorised access. Rotation immediately stops the old token. Reassigning an integration owner rotates the token too. Revoked tokens cannot be rotated back into service; create a new integration. Deactivating an owner account revokes its integrations, so reactivation alone does not restore those tokens.

Usage reports and interpretation

Filter by integration, period and errors. Reports show request counts/errors and recent requests with UTC time, request ID, endpoint/method, response status, observed source IP, reported origin host, client user agent and duration. The latest requests table is limited to 200 records; usage retention is 30 days.

For server to server calls, the source IP is the website server's outbound address. A reported origin host is client supplied and does not prove website ownership. Usage logs do not store bearer tokens, request bodies or customer enquiry payloads. The website URL on an integration is a label/context field; use server IP restrictions where a stable outbound IP is available.

Technology choices

PHP can call the API with cURL; Next.js can call it from a server route; WordPress can use a server side handler with wp_remote_get/wp_remote_post. In each case the public form talks to that website's server, which keeps the token private. No framework specific website UI is required by the CRM.



Settings support and client handover

Email and background delivery

Configure SMTP using the correct host, port and TLS/SSL option. A blank password field retains the saved password. Invites, password links and configured notifications enter the email queue. The application attempts background processing during normal requests; no cPanel cron is required for normal use. Optional scheduled processing can improve delivery when there is no user traffic. Review queued/failed messages and provider settings if mail does not arrive.

Common questions

Issue	Check
An employee cannot see a payslip	Use My Payslips, select the correct month and confirm the slip belongs to that user and is paid. Draft/reopened slips are hidden.
Attendance is locked	Admin opens the paid payslip and uses Reopen payslip & attendance with a reason. Correct attendance, then review/regenerate salary.
Payroll cannot be marked paid	Attendance changed or the period was reopened. Review and save the payslip, or recalculate from attendance first.
A service is missing on the website	Publish it, check linked product status, services:read scope, API response and any website cache.
API shows 401 in Postman	Select Bearer Token and paste the full current token, not a request ID. Use a supported endpoint and check server header forwarding.
Clean URLs give 404	Upload .htaccess and enable Apache mod_rewrite with permitted overrides. Check the installation folder and configured application URL.
An invoice total differs from revenue	Invoice creation is not a receipt. Verify recorded payments and graph dates.

Before handing the system to the business

1. Replace demonstration branding and data as appropriate; verify business details, numbering, tax settings, logo and payroll rules.
2. Test one staff login and one customer invitation end to end. Verify SMTP delivery, customer isolation, document print/PDF and employee paid payslip visibility.
3. For a connected website, test services, quote schema, quote/contact capture, CRM records, request IDs and usage reports. Rotate temporary test credentials and install the final token privately.
4. Back up the database, installed configuration and uploads. Keep installation locking in place. When updating, preserve existing configuration and storage; the v1.0 update does not require a database replacement.

Maintain one accountable admin and a named integration owner for each website. Use the activity log and correction history when investigating business changes. Access checks apply to backend actions and records, not only to which menu items are displayed.