ID AUTOMATION DOCUMENTATION V1.4

Project Title:	ID Automation (Permanent ID)
Version:	V1.4
Last Updated:	02-16-26
Release Date:	02-18-26
Project Lead:	[Name]
Development Team:	Miguel L. Manuel, Kzyrell A. Dela Paz
Github Repository	[Link]

1. Overview

The Employee ID Registration System is a web-based, end-to-end ID card automation platform that streamlines the complete lifecycle of employee ID processing. It replaces the traditional, manual approach to ID creation with a structured digital workflow where employees submit their information through a self-service portal, the system standardizes and enhances photos using AI, and HR manages review, approval, and print-ready outputs through a dedicated dashboard. The process ends with automatic routing of completed ID files to the correct Point of Contact (POC) branch for printing and distribution.

The current process being improved is the manual coordination of ID requests, which typically involves back-and-forth communication, inconsistent photo quality, manual validation of employee details, and time-consuming compilation of print materials. These steps often result in long turnaround times, repeated rework, and routing mistakes when IDs are sent to the wrong branch or printing contact. The system addresses these pain points by introducing a centralized workflow that reduces cycle time and improves visibility, including status tracking from submission up to completion.

At a high level, the proposed solution is a single platform composed of two main experiences. First, the Employee Portal allows authenticated employees to log in via Lark

SSO, upload or generate a professional headshot using BytePlus Seedream, apply automatic background removal through Cloudinary AI, provide their details with a live ID preview, and submit a digital signature via a built-in canvas tool. Second, the HR Dashboard provides a controlled environment for People Support to review submissions, validate details, preview ID cards, generate print-ready PDFs, perform bulk approvals, and trigger delivery steps.

Once an ID is finalized, the system supports automated POC routing by calculating the nearest POC branch using branch coordinates and haversine distance logic. HR can then send the completed PDF to the assigned POC through Lark messaging with the required file attachments. Throughout the process, the system also supports enterprise-level visibility through integration with Lark Bitable for real-time syncing and status tracking.

This system is intended for two main user groups: employees who need an easy and consistent way to request their IDs, and HR (People Support) who are responsible for validating, approving, generating, and distributing ID cards at scale. It is specifically designed to reduce manual work, prevent routing errors, standardize ID photo outputs, and enable faster processing from request to print-ready delivery.

2. Purpose and Goals

The Employee ID Registration System exists to standardize and automate the employee ID issuance process so that ID requests can be submitted, validated, and completed through one controlled workflow. The system was built to reduce the manual effort involved in collecting employee details, checking photo compliance, producing print-ready ID outputs, and coordinating distribution to the correct branch or point of contact. By combining employee self-service submission, AI-assisted photo processing, and an HR-managed review and generation dashboard, the solution improves turnaround time, reduces errors, and ensures that each ID request is processed consistently and securely.

More specifically, the project addresses recurring operational pain points such as incomplete or inconsistent employee data, repeated rework due to non-compliant photos, reliance on manual compilation of ID files, and routing mistakes when output files are sent to the wrong recipient. The system ensures that ID data and required assets (photo and signature) are captured in a structured format, reviewed under HR control, and converted into export-ready outputs that can be distributed using the organization’ s existing internal platform.

The primary goals of this project are:

To reduce manual ID processing by digitizing employee submission, validation, and HR review into a single end-to-end workflow.

To standardize and improve ID photo quality through AI headshot generation and background removal to produce consistent, professional outputs.

To increase HR review speed and control using a dedicated dashboard with searchable records, preview galleries, and bulk actions for faster decision-making.

To minimize routing and delivery errors by automatically determining the correct POC and enabling centralized sending/distribution of print-ready ID files to POCs (including bulk sending).

To improve visibility and accountability by maintaining real-time status progression (e.g., Reviewing → Rendered → Approved → Sent to POC → Completed) and ensuring every request is traceable from submission to turnover.

3. Success Metrics

The success of the Employee ID Registration System is evaluated based on how effectively it improves reliability, efficiency, data accuracy, and overall user experience compared to the previous process. These success metrics cover both technical performance such as system uptime, processing speed, and output correctness, and operational impact such as reduced manual work, fewer reprocessing cases, and smoother HR review. By tracking measurable indicators across employee submission, image processing, and HR dashboard workflows, the project ensures the system delivers consistent value and supports long-term operational needs.

Metric	Description	Measurement Method	Target / Expected Outcome
Service Uptime (Core App)	Availability of the web app (landing, employee form, HR dashboard) and backend API routes.	Monitor uptime via hosting analytics and synthetic checks (HTTP 200 health checks per route).	⩾ 99.5% monthly uptime for core pages and APIs.
API Error Rate (5xx)	Frequency of server-side failures during normal use (submission, generation, HR actions).	Track server logs and count 5xx responses vs total requests.	⩽ 1% of total requests return 5xx.
Submission Reliability	Whether employee submissions consistently save records and required assets (photo/signature)	Compare form submission count vs successfully created employee records in DB + image URLs stored.	⩾ 98% submissions complete end-to-end without manual dev intervention.

	without breaking the workflow.		
Session/Auth Reliability (HR Access)	HR login/session stability (no unexpected logouts, valid cookie/session handling).	Count successful HR logins vs auth failures; review session-expiry logs.	⩾ 99% successful HR logins for valid users; sessions expire only as configured.
Recovery Expectation (Deployment Rollback / Restart)	Ability to restore service after deployment errors or config issues.	Measure time to rollback/redeploy from Git + re-run environment config verification.	RTO ⩽ 30 minutes for typical deployment misconfigurations.
Employee Form Page Load	Time for the employee form UI to become usable on standard connections/devices.	Browser performance timings (LCP/DOMContentLoaded) on desktop + mobile.	⩽ 3 seconds on average connections for initial load.
Submission Processing Time (Non-AI)	Time from “Submit” to confirmation when AI generation is not invoked (basic upload + save).	Timestamp at request start and response completion in backend logs.	⩽ 5 seconds average (excluding external API calls).
AI Headshot Generation Time (BytePlus)	End-to-end time to receive an AI-enhanced headshot result from the AI service.	Backend logs around AI request start/end; average over sample submissions.	⩽ 60 seconds average, ⩽ 120 seconds p95 (network-dependent).
Background Removal Time (Remove.bg / integrated removal)	Time to produce a background-removed photo when enabled.	Log timing for background removal request/response.	⩽ 20 seconds average, ⩽ 45 seconds p95 (service-dependent).
HR Dashboard Load (Employee List)	Speed of loading the HR dashboard table/list of employees.	Measure API response time for employees list endpoint + frontend render time.	⩽ 2 seconds average for typical dataset sizes.

HR Actions Response Time (Approve/Reject/Delete/ Update)	Speed of HR operations (status updates, deletions, PDF upload actions).	Backend logs for request duration per HR action endpoint.	⩽ 3 seconds average per action (excluding PDF upload time).
Batch Export Readiness (PDF/ZIP)	Ability to generate/export requested ID outputs reliably (single/batch).	Count export attempts vs successful files produced and accessible.	⩾ 95% success rate for batch exports under normal load.
PDF Upload Success (Cloudinary)	Reliability of storing generated ID PDFs in Cloudinary and retrieving the resulting URL	Count successful Cloudinary PDF uploads vs upload attempts; validate URL accessibility.	⩾ 98% successful PDF uploads with valid accessible URLs.
Data Accuracy (Employee Details)	Correctness of stored employee data fields (names, ID number, department, position, branch/location, etc.).	HR spot-check sampling vs submitted form values; validation rules triggered by form constraints.	⩽ 1% of records require HR correction due to system-side issues (excluding user input mistakes).
ID Number Uniqueness	Ensures no duplicate ID numbers exist in the database.	Enforce uniqueness checks; periodic DB query for duplicates.	0 duplicates allowed (hard requirement).
Barcode Correctness	Ensures generated barcode corresponds to the expected identifier format and scans successfully.	Scan-test sample IDs; verify barcode value matches record.	⩾ 99% scan success rate on printed and digital IDs.
Image Output Quality (Accept/Reject Rate)	Measures how often AI headshots or processed photos are accepted without needing reprocessing.	Track how many times HR requests re-generation or manual replacement.	⩾ 85% acceptance on first output; ⩽ 15% require reruns.
Photo Compliance (Face/Expression Rules)	Adherence to photo rules (e.g., closed mouth if required, centered face, professional look).	HR review outcomes + automated checks if implemented; track reasons for rejection.	⩾ 90% of submissions meet requirements with minimal HR back-and-forth.

Integration Reliability (Lark Base / Google Sheets, if enabled)	Success rate of syncing employee records and links to external tracking systems.	Count successful sync operations vs failures; error logs with retry counts.	⩾ 95% sync success; failures must be logged and recoverable.
Adoption (Employee Usage)	Whether employees actually use the system instead of manual submission channels.	Compare number of hires/employees needing IDs vs submissions made in the system.	⩾ 90% of ID requests go through the system after rollout.
Completion Rate (Form Completion)	Percentage of employees who start and successfully finish submission.	Track page visits → successful submissions funnel.	⩾ 85% completion rate (after instructions stabilized).
HR Throughput Improvement	Measures speed/volume HR can process compared to manual workflow.	Compare average processing time per ID (baseline vs system).	Reduce HR handling time by ⩾ 50% per ID request.
User Satisfaction (HR)	HR perception of usability, clarity, and efficiency of dashboard and outputs.	Short HR survey after dry run + after rollout; qualitative notes.	⩾ 4.2 / 5 average satisfaction score.
User Satisfaction (Employees)	Employee experience on clarity of form + ease of submission and photo guidance.	Quick feedback form after submission; support ticket counts.	⩾ 4.0 / 5 average satisfaction; downward trend in "how-to" questions over time.

4. Project Timeline

This section outlines the four-week delivery timeline of the Employee ID Registration System, covering activities from planning and requirements alignment up to system completion and the final dry run. The timeline follows a Scrum-style execution where work is broken into weekly sprint cycles. Each sprint produces usable outputs and ends with review, feedback incorporation, and readiness checks for the next iteration.

Sprint/Phase:	**Timeframe:**
	Week 1 (Jan 14–16)

Sprint 1 – Discovery, Research, Planning, and Process Proposal

Objectives:

- Establish a clear understanding of the current Employee ID request process and its pain points
- Define the proposed automated workflow (employee submission to HR review)
- Produce proposal artifacts required for HR review and initial approval
- Begin early exploration of AI-based image standardization feasibility

Detailed Activities:

- Conducted initial research and benchmarking of ID request workflows and comparable systems
- Reviewed requirements, constraints, and scope to align the proposal with real HR operations
- Drafted the proposed Employee ID request workflow and self-service submission approach
- Created visual documentation artifacts:
  - Process flow diagram
  - Employee user journey outline
- Wrote initial user stories for both Employee and HR perspectives
- Defined draft HR review and approval criteria as a baseline for validation rules
- Organized and tracked proposal tasks using Jira for visibility and sprint discipline
- Prepared proposal outputs for HR presentation and consolidated feedback points
- Updated direction from temporary ID to permanent ID after HR clarified constraints
- Began UI/UX direction planning for the permanent ID workflow
- Started R&D on AI image enhancement and prompt consistency for uniform headshots

Key Deliverables:

1. Proposed end-to-end workflow (employee submission → HR review)
2. Process flow diagram and user journey outline
3. Initial user stories (employee-side and HR-side)
4. Draft HR review/approval criteria
5. Presentation-ready proposal pack for HR review
6. Early findings on AI image enhancement feasibility and prompt direction

Review & Outcomes:

- HR engagement completed and workflow proposal validated for refinement
- Scope adjustment confirmed: transition from temporary ID to permanent ID workflow
- Development direction established for implementing the permanent ID solution with AI photo standardization

Sprint/Phase:

Timeframe:

Sprint 2 – System Build, Integrations, and Platform Consolidation	Week 2 (Jan 19–23)

Objectives:

- Build the working Employee Portal and begin HR Portal implementation
- Integrate core processing services: submission handling, AI headshot generation, and background removal
- Establish deployment readiness (local + cloud deployment behavior)
- Explore and validate integration options for data storage and syncing

Detailed Activities:

- Completed the baseline front-end structure for the Employee ID Registration flow and ensured responsiveness (desktop/mobile)
- Implemented the submission workflow with initial database handling and basic HR-side viewing structure
- Tested deployment on Vercel and documented platform constraints, especially around file storage and Drive behavior
- Integrated Cloudinary as a practical solution for image storage in a serverless environment
- Connected the AI headshot generation workflow (BytePlus) and began tuning for acceptable results and stability
- Explored background removal workflow and validated viability for achieving consistent ID photo quality
- Worked on landing page + HR page integration so employee and HR experiences remain on one platform
- Investigated data handling alternatives and constraints:
  - Continued testing for Google Sheets and Drive
  - Explored Lark Base/Bitable feasibility and permission requirements
- Began ID template preparation and initiated database-related adjustments/migration tasks
- Implemented HR dashboard locally and identified gaps/bugs when deployed to Vercel
- Consolidated known remaining items required before dry run

Key Deliverables:

1. Employee Portal (working submission experience)
2. Initial HR Portal structure and dashboard foundation
3. Cloudinary integration for image storage
4. BytePlus AI headshot generation integrated into the workflow

Review & Outcomes:

- System reached “functional end-to-end” level for core flows under local testing
- Deployment constraints clarified (serverless storage limitations and cloud sync considerations)
- Integration direction validated: Cloudinary as primary image storage for deployment
- HR portal implementation confirmed as a priority for sprint continuation

5. Background removal workflow explored/validated
6. Deployed test environment on Vercel with known limitations documented
7. Landing page + HR page integration under one platform
8. Initial ID template work and database adjustments started

Sprint/Phase:

Sprint 3 – HR Demo Iterations, Reviewer Workflow, and Stabilization

Timeframe:

Week 3 (Jan 26–30)

Objectives:

- Stabilize HR-side reviewer workflow and session handling
- Apply feedback-based improvements for demo readiness
- Fix HR dashboard issues to support review and management operations
- Ensure system behavior is consistent enough for dry run preparation

Detailed Activities:

- Conducted project walkthrough discussions to validate the end-to-end flow and identify gaps
- Shared repository structure for organized collaboration and change tracking
- Collected and documented feedback items for incorporation into the next HR-facing demo
- Performed demo-focused improvements and refinements based on observed usability gaps
- Implemented HR reviewer-side functionality to support actual review behaviors
- Strengthened HR session management to improve access control and stability
- Applied HR dashboard fixes to reduce display and workflow issues

Key Deliverables:

1. Improved demo-ready build based on HR-facing review requirements
2. HR reviewer-side workflow implemented
3. HR session handling strengthened and aligned with protected access behavior
4. HR dashboard fixes applied for improved stability and usability

Review & Outcomes:

- Demo improvements completed and HR workflow readiness increased
- HR portal became more stable and operationally aligned for review actions
- System moved into “dry run readiness” phase with reduced critical blockers

Sprint/Phase:

Timeframe:

Week 4 (Feb 2-6)

Sprint 4 – Access Readiness, Operational Dependencies, and Dry Run Preparation

Objectives:

- Finalize access requirements needed for HR and system operation
- Resolve operational dependencies tied to storage and platform access
- Prepare the system for demo dry run conditions with minimal friction for HR

Detailed Activities:

- Processed HR dashboard access requirements and ensured appropriate access paths are clear
- Validated system access dependencies, including drive/storage access expectations and operational constraints
- Ensured readiness steps are complete so HR can execute dry run activities without access blockers
- Completed final access-oriented adjustments required for smooth dry run execution

Key Deliverables:

1. HR dashboard access requirements completed and documented
2. Drive/system access readiness confirmed for operational usage
3. Dry run readiness supported by resolved access dependencies

Review & Outcomes:

- Access blockers reduced/removed to support smooth dry run execution
- System positioned for final demo and documentation alignment

Sprint/Phase:

Sprint 5 – Deployment Readiness, Final QA, and Turnover Preparation

Timeframe:

Week 5 (Feb 9–13)

Objectives:

- Finalize remaining system corrections identified during dry run feedback.
- Validate outputs and access controls under real HR usage conditions.
- Ensure documentation reflects updated cost assumptions and operational scaling.
- Prepare the system for deployment readiness and turnover, with confirmed branch-level rules and notification behavior.

Detailed Activities:

1. PDF Output Validation

- Verified the Field Officer ID output and removed the unintended expiration field, ensuring expiration is only applied where applicable.

2. **POC Mapping Finalization**

- Refined branch mappings to be more specific for clarity and correct routing:

- Bulacan → Malolos
- Laguna → Calamba
- Pampanga → San Fernando

- Confirmed POC routing logic across all 17 branches.

3. **Notification Improvement for Tracy**

- Updated the notification process to include an attachment so recipients can view submitted details directly, instead of receiving a plain notification only.

4. **Development Cost and Billing Clarification**

- Updated the cost section to include all tools/platform**s** as potential company expenses.
- Confirmed whether payment is company-charged and documented the assumption accordingly.
- Added a clarification point in documentation on how the company will be informed about cost breakdown if deployed (e.g., monitoring platform usage dashboards, monthly billing summaries, and internal reporting).

5. **Scaling and Volume Confirmation**

- Updated project assumptions to 3,000–3,500 employees, including new hires, for capacity and cost projections.

6. **Screenshot Restriction / Watermark Review**

- Checked the screenshot restriction behavior on desktop.
- Fixed the issue where the watermark scaled too large and only showed "CONFIDENTIAL – ID Preview," ensuring watermark remains visible but does not obstruct usability.

7. **HR-led Dry Run Execution**

- Conducted the next dry run where HR users operated the system themselves, validating usability, flow correctness, and access restrictions.

Key Deliverables:

1. Updated PDF generation rules with corrected expiration behavior.
2. Finalized branch-level POC mapping (including Bulacan/Malolos, Laguna/Calamba, Pampanga/San Fernando).
3. Enhanced notification mechanism with attachments for complete visibility of details.
4. Revised cost documentation reflecting full platform/tooling breakdown and updated headcount assumptions (3,000–3,500).
5. Corrected desktop watermark behavior for screenshot restriction.

Review & Outcomes:

- All remaining issues from the prior dry run were addressed and validated.
- HR successfully completed a hands-on dry run using the system end-to-end.
- Documentation was updated to reflect final operational assumptions, cost exposure, and deployment readiness requirements.
- The system was confirmed ready for deployment preparation and turnover before the agreed target window (before Feb 13).

6. Completed HR-led dry run results with logged findings.

5. Revision History

This section records all updates made to this document over time. It helps track what changed per version, when the change was made, and who made it, ensuring the documentation stays aligned with the latest system implementation and stakeholder requirements.

Grid

	Changes or Comments Details	Version	Date Implemented	Develop
1	Initial draft created: document outline,..	V1.0	2026/01/15	Lyca G
2	Added core sections (Purpose & Goals,...	V1.1	2026/01/25	Lyca G
3	Updated workflow details and HR das..	V1.2	2026/02/02	Lyca G
4	Expanded documentation for testing/..	V1.3	2026/02/05	Lyca G
5	Revised technical sections to align wit..	V1.4	2026/02/09	Lyca G

5 records

6. Process Flow and Workflow Description

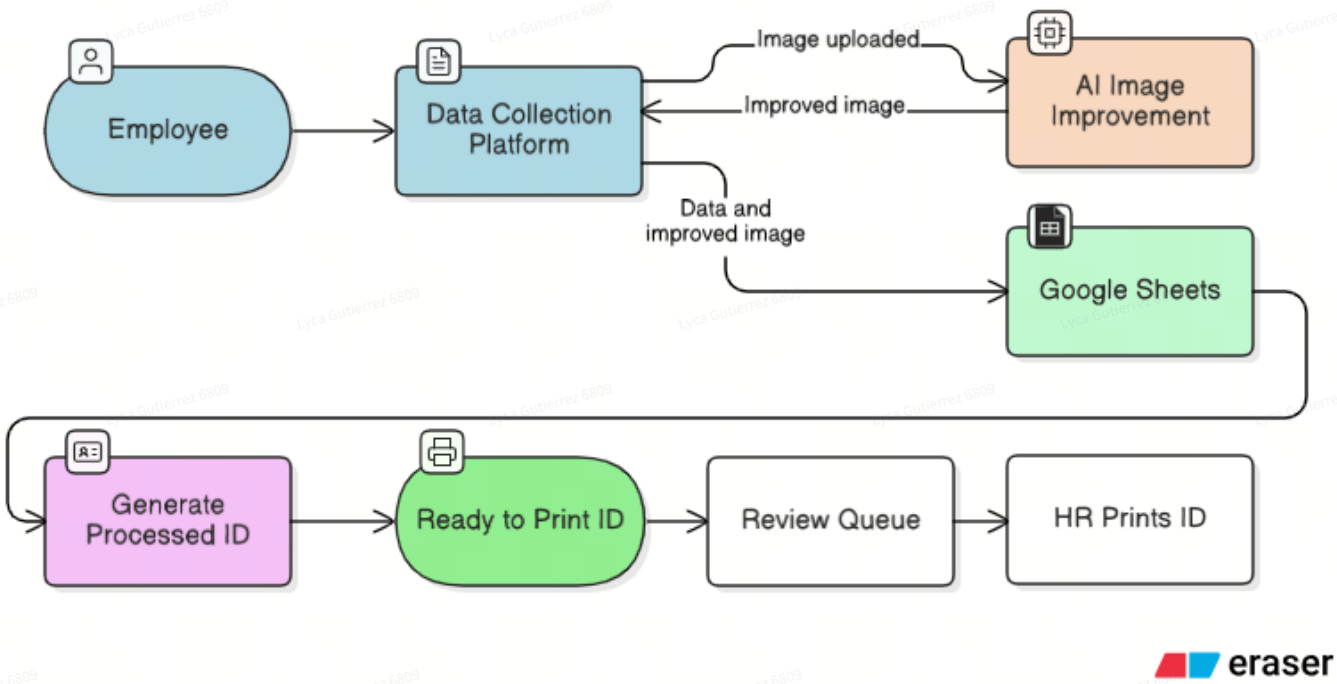


Figure 1. *Version 1: High-level*

The workflow starts when an employee submits their ID request through the registration platform, providing required personal details and uploading a photo and signature. The system then processes the photo through automated enhancement and background removal to meet ID quality standards. Once the employee data and processed image are ready, the system generates a digital, print-ready ID based on the approved template and routes it to HR for review. After HR validates and approves the ID, the final output is forwarded to the designated Point of Contact (POC) for printing and distribution

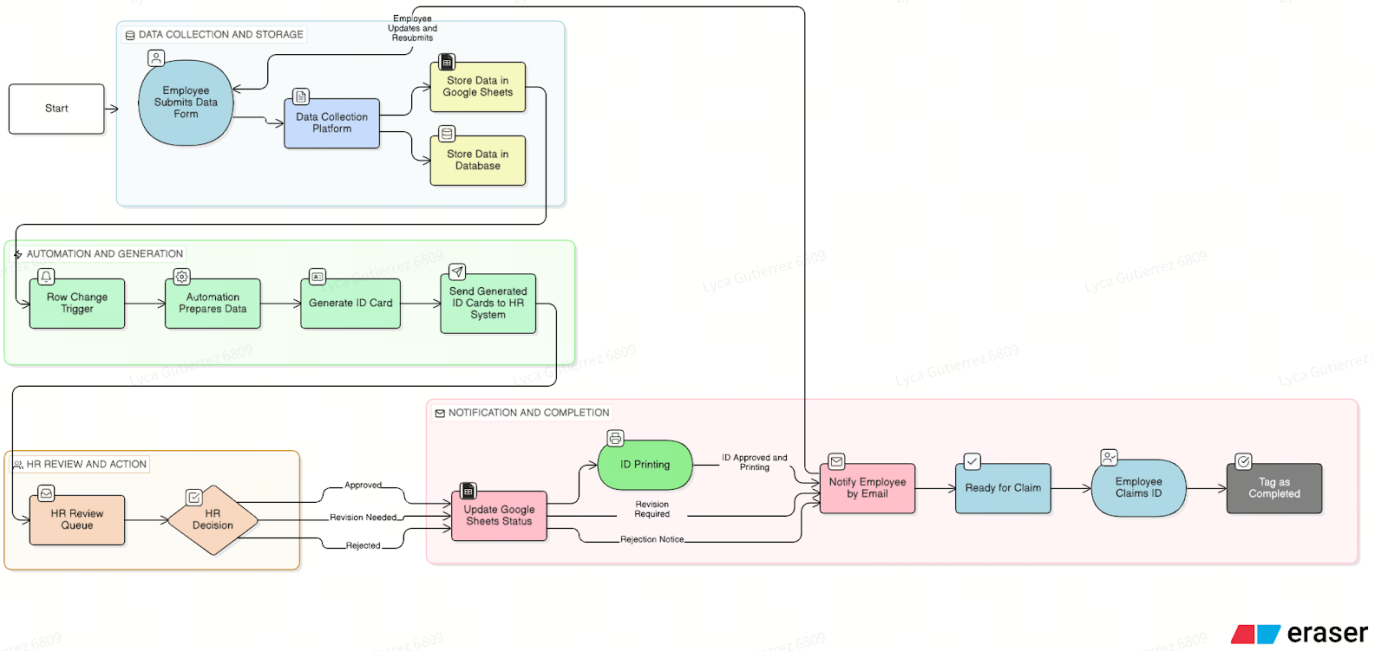


Figure 2. *Version 2: Low-level*

The process begins when an employee accesses the ID Registration platform and submits the required information for ID issuance, including personal details, role and assignment information, and uploaded assets such as a photo and signature. Upon submission, the system performs validation checks to confirm completeness and required formatting. Once validated, the submission is saved in the internal database as the primary record, while a synchronized copy is also logged in the tracking sheet or external repository used for monitoring and reporting. If the employee updates or resubmits details, the system refreshes the stored record and updates the tracking layer to keep both aligned.

After a new or updated entry is recorded, the system prepares the data for downstream processing by normalizing key fields, verifying required assets, and organizing the submission into a consistent structure. The uploaded photo is then routed through the image processing pipeline. This pipeline may include AI-based headshot enhancement to

improve lighting and framing, followed by background removal to ensure the photo meets ID quality standards. The processed image is returned to the system and stored alongside the employee's submission for use in ID rendering.

Using the validated employee data and the processed image, the system generates a digital employee ID based on the approved ID template and marks the output as ready for review. The request then enters the HR review queue, where HR personnel verify correctness, compliance, and overall output quality. During this stage, HR can approve the ID, return it for revision if corrections are needed, or reject the request, with the corresponding status reflected across the system for tracking.

Once HR approves the ID, the workflow moves into distribution and turnover. The approved, print-ready ID output is routed to the designated Point of Contact (POC) based on the employee's branch or assignment, ensuring the correct recipient receives the file for printing and release. The system records this handoff as part of the status progression, and the request is completed once the ID has been successfully turned over through the POC channel and confirmed as ready for issuance.

7. Deployment and Installation Instructions

This section provides detailed instructions for installing, configuring, and deploying the Employee ID Automation System in both local and production environments.

7.1 Deployment Environments Supported

The system supports the following environments:

- **Local Development Environment**
  - Used for development, testing, and debugging
  - Full file system access available
- **Production Environment (Serverless Deployment)**
  - Used for live employee and HR operations
  - Optimized for scalability and security
  - Relies on cloud services for storage and integrations
  - Deployed as a serverless application on Vercel

7.2 Prerequisites

Before installation, ensure the following requirements are met:

7.2.1 System Requirements

- Operating System: Windows, macOS, or Linux
- Python version: Python 3.9 or later
- Internet access for API integrations
- Modern web browser (Chrome, Edge, Firefox)

7.2.2 Required Accounts and Services

Required:

- **Cloudinary account** (image hosting + background removal via Cloudinary AI)
- **BytePlus account/API key** (Seedream headshot generation)
- **Lark Developer account** (OAuth/SSO + app credentials; required for employee login and profile autofill)

Optional (feature-based):

- Google Sheets access via **Google Service Account** (for optional Sheets syncing)
- Supabase project (for persistent database storage; otherwise SQLite fallback is used)
- remove.bg API key (optional fallback background removal; Cloudinary is the primary method)

7.3 Local Installation Procedure

Step 1: Clone the Project Repository

```
git clone <repository-url>
cd hr-id-automation-main
```

Step 2: Create and Activate a Virtual Environment

```
python -m venv venv
```

Activation:

- Windows

```
venv\Scripts\activate
```

- macOS / Linux

```
source venv/bin/activate
```

Step 3: Create and Activate a Virtual Environment

```
pip install -r requirements.txt
```

This installs all backend, frontend, and integration dependencies required by the system.

7.4 Environment Variable Configuration

Step 4: Create Environment Configuration File

a. Copy the configuration file:

```
cp .env.example .env
```

b. Open the .env file and configure the required values:

- Cloudinary credentials
- BytePlus API key
- Lark App credentials (OAuth/SSO configuration)
- Optional integration credentials (Supabase, Google Sheets, remove.bg)

Important: *The .env file must never be committed to version control.*

7.5 Database Initialization

Step 5: Automatic Database Setup

The system initializes database connectivity on startup:

- If **Supabase credentials** are provided, the system connects to Supabase for persistent storage.
- If Supabase is not configured, the system uses **SQLite fallback** and creates required tables automatically.
- Minor schema updates for SQLite fallback are applied automatically during initialization.

No manual database creation is required for local development.

7.6 Running the Application Locally

Step 6: Start the Development Server

```
uvicorn app.main:app --reload
```

Step 7: Start the Development Server

- Landing Page:
  link
- Employee ID Submission Form:
  link
- HR Dashboard:
  Link

7.7 Production Deployment Procedure

Step 8: Configure Production Environment Variables

- Set all environment variables directly in the hosting platform’ s environment settings (Vercel Project Settings).
- Do **not** rely on a .env file in production.

Step 9: Deploy Application

- Deploy the application as a serverless Python service.
- Static assets are served through the platform’ s static file routing.
- Backend routes are handled by the FastAPI application entry point configured for the serverless environment.

7.8 File Storage and Image Handling

Due to serverless file system limitations:

- Uploaded images are **not stored permanently on the server**
- All employee photos and generated ID images are:
  - Uploaded to cloud-based storage
  - Referenced using public URLs
- Temporary runtime files may be stored in ephemeral directories (e.g., /tmp) during execution.

7.9 Post-Deployment Verification

After deployment, perform the following checks:

a. Access the landing page successfully
b. Complete an employee submission flow:
   - Confirm employee authentication via Lark SSO
   - Confirm AI headshot generation executes successfully (BytePlus)
   - Confirm background removal is applied (Cloudinary AI; optional remove.bg if enabled)
   - Confirm data is stored correctly (Supabase if configured; otherwise fallback behavior applies)
c. Log in to the HR dashboard
d. Verify the submitted request appears in the HR management interface
e. Verify status progression and export/PDF generation functions are working

7.10 Logging and Monitoring

- Application logs are generated during startup and runtime
- Logs should be reviewed to confirm:
  - Successful initialization
  - Successful API integrations (Cloudinary, BytePlus, Lark, optional integrations)
  - Error handling behavior
- Logs are used for troubleshooting and system monitoring through the deployment platform runtime logs.

7.11 Deployment Notes and Constraints

- Data persistence depends on external services:
  - Cloudinary provides persistent image storage
  - Supabase provides persistent database storage when enabled
- Serverless environments may reset temporary storage (/tmp)
- For long-term scalability, a managed database service is recommended
- All integrations are designed to fail gracefully when optional credentials are missing (feature disables instead of blocking the core flow)

7.12 Access Requirements

This subsection defines who is allowed to deploy, configure, and manage the system.

7.12.1 Deployment Access

The following access is required to deploy the system:

Repository access with permission to pull the latest source code

Environment configuration access for:

- API keys
- Authentication credentials (Lark app settings)
- Optional Integration settings

- Deployment platform access to:
  - Configure environment variables
  - Trigger deployments
  - Monitor build and runtime logs

Only authorized personnel should be granted deployment access to prevent misconfiguration or security risks.

7.12.2 System Access Levels

The system enforces role-based access control:

Employee Access

- Can submit ID registration requests
- Cannot view other employee data
- Cannot access administrative tools

HR Access

- Can access the HR Dashboard
- Can review, approve, and manage ID requests
- Can generate and export ID cards
- Access is restricted via authenticated login

Administrative Access

- Manages deployment and configuration
- Controls environment variables and integrations
- Performs maintenance and troubleshooting

7.13 Deployment Responsibility

This subsection defines ownership and accountability for system deployment.

7.13.1 Responsible Party

Deployment and configuration of the system are handled by the assigned development team

The team is responsible for:

Initial deployment

Environment setup

Integration configuration

Deployment verification

7.13.2 Post-Deployment Ownership

After deployment:

HR Team

- Owns operational usage of the system
- Manages employee submissions and approvals
- Provides functional feedback

Development Team

- Provides technical support
- Handles system updates and bug fixes
- Implements approved enhancements

7.13.3 Change Management

Any deployment changes must:

Be tested in a controlled environment

Be approved before production rollout

Emergency fixes may be deployed with post-implementation review

8. Server Requirements

This section defines the infrastructure, runtime, and service requirements necessary to operate the Employee ID Automation system in both local and deployed environments. The

system is designed to function within a lightweight serverless architecture while maintaining compatibility with standard development machines.

8.1 Supported Environments

The system supports the following execution environments:

- **Local Development Environment**
- **Cloud-Based Serverless Environment (Vercel)**

The application dynamically adjusts behavior depending on environment configuration (e.g., whether Supabase credentials are present, serverless temporary storage usage, and integration toggles).

8.2 Application Runtime Requirements

Requirement	Specification
Programming Language	Python 3.9 or higher
Application Framework	FastAPI
Web Server	ASGI-compatible server (e.g., Uvicorn)
Package Manager	pip
Environment Configuration	.env file or platform-based environment variables
Template Engine	Jinja2 (server-rendered HTML templates)

8.3 Compute Requirements

Resource	Minimum Requirement	Notes
CPU	1 vCPU	Suitable for concurrent form submissions and HR access
Memory	512 MB (local), 1024 MB (cloud recommended)	Higher memory improves stability for image processing and PDF rendering workflows
Execution Time	Platform-limited (serverless max duration)	Requests must complete within the deployment platform’ s function execution limit; external API latency

		(AI/background removal) can affect runtime
Concurrent Requests	Moderate	Designed for internal organizational usage

8.4 Storage Requirements

8.4.1 File Storage

Type	Requirement
Local Development	Writable file system
Cloud Deployment	Ephemeral storage only

Notes:

- Persistent server-side file storage is not required.
- Uploaded and processed images are stored via external cloud services (e.g., Cloudinary).
- Temporary files may be written to /tmp during request execution in serverless environments.

8.4.2 Database Storage

Component	Requirement
Database Engine (Primary when configured)	Supabase (PostgreSQL)
Database Engine (Fallback)	SQLite
Local Storage (SQLite fallback)	Local file-based database (database.db)
Cloud Storage (SQLite fallback)	Temporary filesystem (/tmp/database.db)

Important Consideration:

- When running on SQLite fallback in serverless mode, database state may not persist across restarts.
- When Supabase credentials are configured, the system uses Supabase for persistent storage.

8.5 Network and Connectivity Requirements

The server must have outbound internet access to communicate with external services.

Service	Purpose
Cloud Image Hosting (Cloudinary)	Image upload, retrieval, and image processing delivery
AI Image Generation API (BytePlus)	Professional headshot generation
Background Removal	Cloudinary AI (default) and optional remove.bg fallback
Data Sync Services (Optional)	Google Sheets appending and reporting
Authentication Provider (Lark)	Employee authentication (OAuth 2.0 + PKCE) and profile autofill
Barcode Service	Barcode generation for ID validation elements

Inbound access is limited to standard HTTP/HTTPS requests.

8.6 Environment Variable Requirements

The following environment variables must be accessible to the server at runtime:

Category	Examples
Lark Configuration (Required)	LARK_APP_ID, LARK_APP_SECRET, LARK_REDIRECT_URI, Bitable identifiers
Image Storage (Required)	Cloudinary credentials
AI Services	BytePlus API key/model; optional remove.bg key
Database (Optional)	SUPABASE_URL, SUPABASE_KEY (SQLite fallback if missing)
Optional Integrations	Google service account JSON, spreadsheet ID, worksheet name
Security	JWT_SECRET (recommended for HR token signing)
Routing/Test Flags (Optional)	POC test mode and test recipient settings

Missing optional variables do not prevent system execution but disable related features.

8.7 Security Requirements

Area	Requirement
Credential Storage	Environment variables only
HR Authentication	JWT-based authentication stored in cookies (stateless for serverless compatibility)
Employee Authentication	Lark OAuth 2.0 with PKCE
Data Exposure	No hardcoded secrets in source code
Network	HTTPS recommended for deployed environments

Sensitive credentials must never be committed to version control.

8.8 Logging and Monitoring Requirements

Aspect	Requirement
Logging Level	Informational and error-level logs
Log Output	Console-based
Monitoring	Deployment platform runtime logs

Logs are used for:

Deployment verification

Runtime error detection

Integration troubleshooting

8.9 Scalability Considerations

- The system is designed for **internal organizational scale**
- Horizontal scaling is handled by the deployment platform (serverless).
- External services handle storage and processing scalability
- No manual load balancing is required

8.10 Limitations

Limitation	Description
Persistent Storage (Fallback Mode)	SQLite on serverless uses ephemeral /tmp storage and may not persist across restarts
Large File Uploads	Limited by external service constraints
Long-Running Tasks	Bound by serverless execution limits and external API latency

These limitations are addressed through external integrations and architectural design.

9. Tech Stacks

This section lists all technologies, platforms, and services used in the development, deployment, and operation of the Employee ID Automation system. The stack covers backend processing, frontend interfaces, AI services, data storage, authentication, and hosting infrastructure. All tools and frameworks listed below are currently implemented in the system or are officially part of its deployment and integration setup.

9.1 Backend Technologies

Component	Technology	Description
Backend Framework	**FastAPI (Python)**	Core backend framework used to build API endpoints, handle HTTP requests, manage routing, and enforce request validation. Chosen for its performance, async support, and clean API design.
Application Server	**Uvicorn (ASGI)**	ASGI server used to run the FastAPI application locally and in deployment environments.
Authentication HR)	**Session-based Authentication (Custom)**	HR authentication implemented using session cookies and bcrypt password hashing, defined in auth.py.
Authentication (Employee)	**Lark SSO (OAuth 2.0 + PKCE)**	Employee authentication uses Lark OAuth with PKCE for secure sign-in, enabling user identity verification and profile-based autofill.

API Design	**RESTful APIs**	Backend exposes REST endpoints for employee submission, AI processing, HR dashboard access, and data retrieval.
Environment Configuration	**python-dotenv**	Loads environment variables securely from .env files during local development and deployment.
File Upload Handling	**python-multipart**	Enables handling of multipart form submissions for image uploads and signature capture.
Async File I/O	**aiofiles**	Supports async-friendly file operations where applicable during local or serverless execution.
HTTP Utilities	**urllib / HTTP client libs**	Used for calling external services (e.g., optional remove.bg calls) and fetching/processing remote assets as part of the workflow.

9.2 Frontend Technologies

Component	Technology	Description
Markup	**HTML (Jinja2 Templates)**	Server-rendered HTML templates used for the landing page, employee form, HR login, HR dashboard, and gallery views.
Styling	**CSS (Custom Stylesheets)**	Custom CSS files used for layout, responsiveness, animations, and UI consistency across employee and HR views.
Client-side Logic	**Vanilla JavaScript**	Handles form interactions, live ID preview updates, AI image generation calls, background removal triggers, and UI feedback.
PDF Rendering (Client-side)	**html2canvas**	Captures the ID card preview/layout in the browser as an image for accurate export rendering.
PDF Export (Client-side)	**jsPDF**	Generates print-ready PDFs in the browser based on captured ID layouts, supporting bulk export workflows.
UI Design Approach	**Responsive Web Design**	Interfaces are optimized for desktop and mobile usage as required by HR and employee accessibility needs.

9.3 Database and Storage

Component	Technology	Description
Primary Database (Credential-based)	**Supabase (PostgreSQL)**	Cloud-based PostgreSQL database used when Supabase credentials are configured, enabling persistent storage and production readiness.
Fallback Database	**SQLite**	Used as a fallback database when Supabase credentials are not set; also used during local development for storing employee records, statuses, and metadata.
Local Storage (Dev)	**File System (Local)**	Used during local development to store uploaded photos and signatures.
Cloud Storage	**Cloudinary**	Used in deployment to store employee images and AI-generated photos due to serverless filesystem limitations.
Serverless Temporary Storage	**/tmp Directory (Vercel)**	Temporary storage used during execution in Vercel’ s serverless environment.

9.4 AI and Image Processing Services

Component	Technology	Description
AI Headshot Generation	**BytePlus Seedream 4.5**	Generates professional, standardized employee headshots from uploaded images.
Background Removal (Primary)l	**Cloudinary AI Background Removal**	Performs background removal and image standardization as the default method for producing clean ID photos.
Background Removal (Optional)	**Remove.bg API**	Optional fallback background removal method used only when REMOVEBG_API_KEY is provided.
Image Processing	**Pillow (PIL)**	Handles image resizing, format handling, and processing before upload or rendering.

9.5 Identification and Validation Services

Component	Technology	Description
Barcode Generation	barcodeAPI.org	Generates barcodes used for employee ID validation and scannable identification elements.

9.6 External Integrations

Integration	Technology	Purpose
Image Hosting	**Cloudinary API**	Stores original, AI-generated, and processed images securely and provides public URLs.
Data Synchronization (Optional)	**Google Sheets (gspread + Service Account)**	Allows employee submission data and image URLs to be appended to Google Sheets for reporting and tracking when configured.
Service Account Credential	**GOOGLE_SERVICE_ACCOUNT_JSON**	JSON credential used to authorize Google Sheets (and related scopes where applicable) via a service account.
Collaboration Platform	**Lark (Implemented)**	Provides employee authentication (SSO), profile autofill, status/data tracking via Bitable, and messaging support for routing outputs to POCs.
Tracking / Database Sync	**Lark Bitable (Implemented)**	Enables real-time syncing of employee records and status progression for enterprise visibility and tracking.
Notifications / Routing	**Lark Messaging (Implemented)**	Supports sending or notifying the designated POC with finalized outputs/links as part of distribution workflow.

9.7 Deployment and Infrastructure

Component	Technology	Description
Hosting Platform	**Vercel (Serverless)**	Hosts the FastAPI backend and frontend as serverless functions.
Deployment Method	**Git-based Deployment**	Code is deployed directly from the GitHub repository.

Configuration Management	**Environment Variables**	Sensitive credentials and configuration values are stored as environment variables.
Routing	**Vercel Routing Rules**	Handles static file serving and API request routing.

9.8 Development and Productivity Tools

Tool	Purpose
GitHub	Source code version control and collaboration
Lark Docs	Task tracking, sprint planning, and progress monitoring
Figma / Figma Make	UI/UX planning and layout prototyping
AI Tools	ChatGPT, Claude Opus 4.5 (used for ideation, prompt refinement, and development support)

9.9 Security and Access Control

Area	Implementation
Credential Management	Environment variables only (no hardcoded secrets)
Password Security	bcrypt hashing
HR Access Control	Restricted to authenticated HR users
Employee Authentication	Lark OAuth 2.0 with PKCE
Session Handling	Time-bound session cookies
File Access	Cloud-hosted images with controlled URLs

Note: Some technologies listed above operate under free-tier usage during development and testing. Paid plans are required only when usage thresholds are exceeded in production.

10. Database Structure

The Employee ID Registration System uses a relational database to store employee submissions, generated assets, and HR processing status. The implementation supports Supabase (PostgreSQL) for persistent storage when configured, with a SQLite fallback to ensure the system can still run in local development and in environments where Supabase is not available.

The database is initialized during application startup and is designed to support the full lifecycle of an employee ID request—from submission, AI image processing, HR review, up to ID rendering and export.

10.1 Database Engine

Database Type: Supabase (PostgreSQL) with SQLite fallback

Initialization Method: init_db() in database.py

Environment Handling:

- **If Supabase credentials are set (SUPABASE_URL, SUPABASE_KEY):**
  The system connects using the Supabase client and **verifies** the required tables exist (table creation is handled through Supabase setup scripts / dashboard).
- **If Supabase credentials are not set or Supabase initialization fails:**
  The system uses **SQLite** as a fallback database.

SQLite Storage Location:

- **Local Development:** database.db stored in the project root
  Cloud / Serverless Deployment (Vercel fallback mode): /tmp/database.db (ephemeral storage)

This design allows the system to run consistently across environments while acknowledging the temporary nature of storage in serverless deployments when operating in SQLite fallback mode.

10.2 Database Tables Overview

The system uses two primary tables:

- **employees** — Stores employee submissions, image references, processing states, and generated ID outputs.
- **security_events** — Stores security-related logs (e.g., screenshot/recording attempt events) captured by the system.

10.3 employees Table Structure

The employees table is created automatically in SQLite fallback mode if it does not exist. Below is the schema and explanation of each field.

Table: employees

Column Name	Data Type	Description
id	INTEGER (Primary Key, Auto Increment)	Unique internal identifier for each submission
employee_name	TEXT (Not Null)	Full name of the employee (display/reference)
first_name	TEXT	First name (stored separately when provided)
middle_initial	TEXT	Middle initial (stored separately when provided)
last_name	TEXT	Last name (stored separately when provided)
suffix	TEXT	Optional suffix (e.g., Jr., III)
id_nickname	TEXT	Optional nickname displayed on the ID
id_number	TEXT (Not Null)	Official employee ID number
position	TEXT (Not Null)	Employee job position
location_branch	TEXT	Employee branch / location (when applicable)
department	TEXT	Employee department
email	TEXT	Employee email address
personal_number	TEXT	Employee contact number
photo_path	TEXT (Not Null)	Local file path of the uploaded or generated photo (used in fallback/local flows)
photo_url	TEXT	Public URL of the original photo (e.g., Cloudinary)
new_photo	INTEGER (Default: 1)	

		Flag indicating whether a new photo was generated
new_photo_url	TEXT	URL of AI-enhanced or AI-generated photo
nobg_photo_url	TEXT	URL of background-removed photo
signature_path	TEXT	Local file path of the signature image
signature_url	TEXT	Public URL of the signature image
status	TEXT (Default: 'Reviewing')	Current processing status
date_last_modified	TEXT	Timestamp of the last update
id_generated	INTEGER (Default: 0)	Flag indicating if an ID output has been generated
render_url	TEXT	URL or path to the final rendered ID output
emergency_name	TEXT	Emergency contact name
emergency_contact	TEXT	Emergency contact number
emergency_address	TEXT	Emergency contact address
field_officer_type	TEXT	Field officer classification/type (when applicable)
field_clearance	TEXT	Field clearance value/flag (when applicable)
fo_division	TEXT	Field officer division (when applicable)
fo_department	TEXT	Field officer department (when applicable)
fo_campaign	TEXT	Field officer campaign (when applicable)
resolved_printer_branch	TEXT	System-resolved printer/POC branch used for routing

10.4 security_events Table Structure

The security_events table is created automatically in SQLite fallback mode to record client security-related events.

Table: security_events

Column Name	Data Type	Description
id	INTEGER (Primary Key, Auto Increment)	Unique identifier for the event
event_type	TEXT (Not Null)	Type of security event (e.g., screenshot attempt)
details	TEXT	Additional event details
user_id	INTEGER	Related user/employee record id (if applicable)
username	TEXT	Username/email if available
url	TEXT	Page/route where the event occurred
user_agent	TEXT	Browser user agent string
screen_resolution	TEXT	Client screen resolution
timestamp_server	TEXT (Not Null)	Server-side timestamp
timestamp_client	TEXT	Client-side timestamp (if captured)
created_at	TEXT (Not Null)	Event creation timestamp

10.5 Status and Lifecycle Tracking

The database schema supports progressive state changes throughout the ID automation process.

- **Initial State:**
  - status = 'Reviewing'
  - id_generated = 0
- **During Processing:**

- AI-generated and processed image URLs are stored (e.g., new_photo_url, nobg_photo_url)
- Signature assets are saved and linked (signature_url and/or signature_path)
- date_last_modified is updated on changes

- **Post-HR Approval:**
  - status is updated through workflow states (e.g., Approved, Sent to POC, Completed, Removed)
  - id_generated is set to 1 when an output is produced
  - render_url is populated with the final output reference (URL/path)

10.6 Migration and Schema Evolution

Basic schema migration logic is implemented for SQLite fallback databases to ensure backward compatibility:

- New columns are added dynamically using ALTER TABLE statements if they do not yet exist (e.g., new_photo_url, nobg_photo_url, emergency and field-officer fields, and routing resolution fields).
- This prevents database breakage when updates are introduced without requiring manual intervention.

10.7 Design Considerations

a. Dual-mode database setup (Supabase + SQLite fallback) supports both persistent production storage and lightweight local/serverless fallback execution.

b. URL-based asset storage (Cloudinary and related services) improves compatibility with cloud deployments and reduces reliance on local file persistence.

c. Single-table employee data design simplifies HR review and export workflows while still allowing schema expansion as requirements grow.

11. Troubleshooting

This section outlines common issues that may occur during development, deployment, or operation of the Employee ID Registration System, along with their possible causes and recommended resolutions. The goal is to provide a clear reference for developers and administrators to quickly diagnose and address system problems without disrupting operations.

Common Issues	Possible Causes	Troubleshooting Steps

Application Does Not Start or Crashes on Launch	• Missing or incorrectly configured environment variables • Required Python dependencies not installed • Incorrect Python version	1. Verify that all required environment variables are set in the .env file or deployment environment (e.g., Cloudinary credentials, AI service keys). 2. Confirm that dependencies are installed by running: `pip install -r requirements.txt` 3. Ensure the Python version meets the minimum requirement (Python 3.9+). 4. Check terminal or server logs for error messages indicating missing modules or configuration issues.
Landing Page or Static Assets Not Loading Properly	• Static files not mounted correctly • Incorrect file paths in HTML templates • Deployment platform limitations	1. Verify that the /static directory is correctly mounted in the FastAPI application. 2. Confirm that referenced CSS and JavaScript files exist in app/static/. 3. Clear browser cache and reload the page. 4. For deployed environments, ensure that static files are included in the deployment build and are not excluded by configuration rules.
Employee Form Submission Fails	• Invalid or missing form inputs • Image upload failure • Backend validation errors	1. Confirm that all required form fields are filled out correctly. 2. Check file upload size and format compatibility. 3. Inspect backend logs for validation or parsing errors. 4. Verify that multipart form handling is enabled and properly configured.
AI Headshot Generation Not Working	• Missing or invalid AI service API key • External AI service latency or downtime • Unsupported image input format	1. Confirm that the AI service API key is correctly set in the environment variables. 2. Test the system using a valid image file that meets expected input requirements.

		3. Monitor logs for timeout or request errors related to the AI service. 4. If the external service is unavailable, verify that the system gracefully falls back to original image handling.
Background Removal Fails or Is Unavailable	• Background removal service not installed or disabled • Resource limitations in the deployment environment • Service dependency download failure	1. Check whether the background removal dependency is installed and available. 2. Review logs to confirm whether the service reports itself as unavailable. 3. Test locally to determine whether the issue is environment-specific. 4. If deployed in a constrained serverless environment, disable background removal temporarily and proceed with standard image handling.
Images Do Not Appear in the HR Dashboard	• Image storage service credentials missing or invalid • Incorrect image URL saved in the database • Deployment environment file system restrictions	1. Verify that the image storage service credentials are correct and active. 2. Check database records to ensure valid image URLs are stored. 3. Confirm that images are being served from an external storage service rather than a restricted local filesystem. 4. Test image URLs directly in a browser to confirm accessibility.
HR Login or Dashboard Access Issues	• Invalid login credentials • Expired or missing session cookies • Misconfigured access control logic	1. Verify that HR credentials are correctly configured in the environment variables. 2. Clear browser cookies and attempt to log in again. 3. Check session handling logic to confirm sessions are being created and validated correctly. 4. Review access logs to confirm whether authentication requests are being rejected or redirected.

Data Not Persisting After Deployment	• Use of temporary storage in serverless environments • Application restart or redeployment • Expected behavior of non-persistent databases	1. Confirm whether the application is running in a serverless environment with ephemeral storage. 2. Understand that data stored in temporary directories may reset on redeployments. 3. For long-term persistence, plan migration to a managed external database solution. 4. Document any expected data reset behavior to stakeholders.
Deployment-Specific Issues	• Environment variable mismatch between local and deployed environments • Platform-specific limitations (e.g., file size, execution time) • Build or routing misconfiguration	1. Compare local .env values with deployed environment settings. 2. Review deployment logs for build-time or runtime errors. 3. Confirm routing and entry-point configurations. 4. Redeploy the application after applying fixes to ensure changes take effect.
Logging and Escalation		1. Collect relevant logs, error messages, and reproduction steps. 2. Identify whether the issue is frontend, backend, or third-party service related. 3. Escalate the issue to the development team with complete technical details. 4. Document the issue and resolution for future reference.

12. List of Authorize Users

This section defines the user roles authorized to access and operate specific parts of the Employee ID Registration System. Access is role-based and enforced through authentication, session handling, and permission checks to ensure that sensitive employee data and administrative functions are protected.

12.1 Authorized User Roles

User Role	Access Scope	Primary Responsibilities	Authentication Method
Employee	Employee Portal (ID Application Form)	• Submit personal and employment details • Upload photo and signature • Review live ID preview before submission	Company account authentication (e.g., Lark login or equivalent)
HR Personnel	HR Dashboard	• Review employee ID applications • Validate submitted information • Approve or reject requests • Generate and manage ID outputs	Restricted authentication (session-based or company-authenticated login)
System Administrator	Application Configuration & Deployment	• Manage environment variables • Configure integrations (image storage, AI services, data sync) • Deploy and maintain the system	Server / platform-level access

12.2 Access Control Principles

- Access is **limited strictly to assigned roles** and cannot be shared across user types.
- Employees **cannot** access HR dashboard functionalities.
- HR users **cannot** modify system configuration or deployment settings.
- Administrative access is limited to personnel responsible for system maintenance and infrastructure.

12.3 Authentication and Authorization Rules

- All HR access requires successful authentication before dashboard access is granted.

- User sessions are validated on each protected request to prevent unauthorized access.
- Unauthorized users attempting to access restricted routes are redirected or denied access.
- Authentication credentials and access rules are configurable via environment variables or platform-level settings.

12.4 Security Considerations

- Credentials and tokens are never hardcoded and are managed through environment configuration.
- Session expiration and validation mechanisms prevent prolonged unauthorized access.
- Access rights are reviewed and updated as organizational roles change.

13. User Documentation

Manuals or guides that help end-users navigate and use the system.

14. Development Cost

This section presents a detailed breakdown of the estimated development and operational costs required to build, deploy, and maintain the Employee ID Automation system. Although the system is internally developed, it relies on multiple third-party platforms that introduce **usage-based and subscription costs**, particularly as the number of employees increases.

Note: *All estimates below are computed using an* ***expected scale of 3,000 employees****, which represents a realistic operational volume for company-wide ID issuance.*

14.1 Cost Scope and Assumptions

Item	Description
Development Type	Internal (no outsourcing)
Billing Model	Mixed (Free tier, subscription, usage-based)
Employee Volume	3,000 employees
Cost Coverage	AI processing, image storage, database, hosting
Currency	PHP (with USD reference where applicable)

Note: Development labor costs are excluded as this is an internal initiative. This section focuses on tooling and infrastructure costs only.

14.2 AI and Image Processing Costs

14.2.1 Background Removal (Primary: Cloudinary AI)

Background removal is applied to ensure clean, uniform ID photos. In the current system, background removal is performed primarily via Cloudinary AI, and is accounted for under Cloudinary usage/credits.

Cost Type: Included in Cloudinary monthly credits / usage

Risk: High usage if reprocessing is triggered frequently

Mitigation: Limit retries, cache processed outputs, and reuse transformed URLs

14.2.2 Background Removal (Optional Fallback: remove.bg)

Remove.bg may be used only as a fallback when enabled via REMOVEBG_API_KEY. Pricing is credit-based and varies by the plan/credits purchased at the time of procurement.

Pricing Logic:

- 1 credit = 4 images
- 1 credit = $1
- 3 credits = $3

Cost Computation:

Item	Value
Total Employees	3,000
Images per Credit	4
Total Credits Required	750 credits
Cost per Credit	$1
Total Cost (USD)	$750
Conversion to PHP	₱60/USD
Total Cost (PHP)	**₱45,000**

Cost Type: Usage-based, optional

Risk: Retry-heavy usage increases cost

Mitigation: Use Cloudinary AI as default; restrict remove.bg to exception handling only

Note: Exchange rates are estimated and subject to change.

14.2.3 AI Headshot Generation (BytePlus – Seedream 4.5)

BytePlus Seedream is used for AI-powered headshot enhancement, ensuring consistent lighting, framing, and professional appearance.

Pricing Logic:

- 1 image = $0.04 (₱2.40)
- Maximum of 5 retries per employee (to handle errors or unsatisfactory outputs)

Cost Computation:

Item	Value
Total Employees	3,000
Cost per Image	₱2.40
Max Attempts per Employee	5
Cost per Employee	₱12
Total Cost (PHP)	**₱36,000**

Cost Type: One-time processing

Risk: Retry-heavy usage increases cost

Mitigation: Prompt optimization and retry limits

14.3 Barcode Generation Costs

14.3.1 barcodeAPI.org

Barcodes are used for ID verification and scanning, with each barcode consuming one API token.

Plan	Cost	Daily Token Limit

Free Tier	₱0	10,000 tokens/day
Pro Tier	$15/month	25,000 tokens/day

- 1 token = 1 barcode
- Expected usage fits entirely within Free Tier

Estimated Cost: ₱0
Upgrade Trigger: Only if daily generation exceeds 10,000 barcodes

14.4 Image Storage Costs

14.4.1 Cloudinary (Image Hosting, Delivery, & Transformations)

Cloudinary stores:

- Employee photos
- AI-enhanced images
- Background-removed assets
- Rendered ID visuals

Plan Comparison (official pricing in USD; convert to PHP using company FX rate):

Plan	Cost	Users	Monthly Credits
Free	₱0	3 users / 1 account	25
Plus	₱5,340	3 users / 2 accounts	225

- 1 credit = 1GB
- Free tier reached its limit during development
- **Plus plan is required** for sustained operations

Cost Type: Recurring (monthly)
Risk: High-resolution assets increase credit usage
Mitigation: Asset optimization and compression

14.5 Database and Storage Costs

14.5.1 Supabase (PostgreSQL + Storage)

Supabase is used for persistent employee data and metadata when enabled.

Storage Pricing

Plan	Included Storage	Monthly Cost
Free	1 GB	₱0
Pro	100 GB	$0.021 per GB/month
Over-usage	—	$0.00002919 per GB-Hour

- Free tier exceeded as data volume increased
- Upgrade required for stable production use

Cost Type: Usage-based
Risk: Image-heavy workflows increase storage
Mitigation: Offload images to Cloudinary, store URLs only

14.6 Hosting and Deployment Costs

14.6.1 Vercel (Application Hosting)

The system is deployed using Vercel Serverless Functions. Pricing and limits depend on plan and function configuration (duration/memory).

Current Status

- Running on **Hobby (Free) Plan**
- No charges incurred yet

Upgrade Pricing (If Limits Are Exceeded)

Resource	Cost
Function Duration	$0.18 per GB-Hour
Function Invocations	$0.60 per 1M invocations

Risk: Exceeding plan limits can require upgrade or incur usage charges
Mitigation: Optimize image workflow, reduce function calls, cache transformed assets, minimize reprocessing

14.7 Consolidated Cost Summary

One-Time / Volume-Based Costs

Item	Cost (PHP)
AI Headshot Generation	₱36,000
Background Removal	₱45,000
Total One-Time AI Cost	**₱81,000**

Recurring Monthly Costs (Minimum)

Item	Cost (PHP)
Cloudinary Plus Plan	₱5,340
Supabase Storage	Usage-based
Vercel Hosting	₱0 (current)

15. Stakeholders

This part of the document identifies the key individuals and groups involved in the development, usage, review, and decision-making processes of the Employee ID Registration System. Stakeholders play distinct roles in ensuring that the system meets operational requirements, aligns with organizational policies, and delivers its intended value.

15.1 Primary Stakeholders

Stakeholder	Role	Responsibility
Human Resources (HR) Team	System Owner / Primary User	Reviews employee submissions, validates information, approves ID generation, and manages employee records
Employees	End Users	Submit personal information, photos, and signatures for ID registration
Project Lead	Product Oversight	Oversees system direction, prioritizes features, and ensures alignment with business requirements

Development Team	System Development	Designs, builds, tests, and maintains the Employee ID Registration System

15.2 Secondary Stakeholders

Stakeholder	Role	Responsibility
IT / Technical Support	Infrastructure Support	Assists with deployment, system access, troubleshooting, and environment configuration
Management / Decision Makers	Review and Approval	Reviews project progress, evaluates effectiveness, and approves rollout or enhancements
Data Protection / Compliance	Advisory	Ensures proper handling of employee data and adherence to internal data privacy standards

15.3 Stakeholders Benefits

Stakeholder	Benefit	Description / Impact
Employees	Simplified ID Application	Employees can submit ID details, photos, and signatures through a single self-service form without manual paperwork or email exchanges.
	Faster ID Processing	Automated validation and streamlined workflows reduce waiting time for ID issuance.
	Professional ID Output	AI-powered image enhancement and background removal ensure consistent and professional-looking ID photos.
Human Resources (HR) Team	Centralized Management	

		All ID requests are collected and managed in one system instead of scattered emails or files.
	Reduced Manual Work	Automation minimizes repetitive tasks such as manual data entry, photo editing, and follow-ups.
	Improved Data Accuracy	Form validation and controlled inputs reduce errors and incomplete submissions.
	Faster Review and Approval	A dedicated HR dashboard enables quicker review, approval, and tracking of ID requests.
Management	Operational Efficiency	The system reduces administrative overhead and improves turnaround time for employee onboarding support tasks.
	Process Visibility	Clear workflows and status tracking provide better oversight of ID issuance activities.
Organization	Standardized ID Process	Establishes a consistent and repeatable process for employee ID registration across the organization.
	Scalability	The system can support increasing employee volume without a proportional increase in manual workload.
	Foundation for Future Enhancements	The automated workflow supports future integrations, reporting, and system improvements.

16. User Feedback

The feedback summarized below was gathered during system demonstrations, walkthroughs, and review discussions with intended users. These insights played a key role in validating requirements, refining workflows, and ensuring the system aligns with actual operational needs.

16.1 Sources of Feedback

Feedback for the system is collected from the following primary user groups:

- **End Users (Employees)**
  Employees who use the system to submit ID information, upload photos, generate AI headshots, and preview their IDs.
- **HR Users**
  HR personnel who access the HR Dashboard to review submissions, manage employee records, approve IDs, and export outputs.
- **Supervisors / Stakeholders**
  Team leads and stakeholders who evaluate the system during demos and review overall process alignment, efficiency, and scalability.

16.2 Feedback Collection Methods

User input was collected through the following channels:

- Live demonstrations and walkthrough sessions
- Clarification meetings and Q&A discussions
- Observations during testing and pilot usage
- Direct comments and suggestions raised during reviews
- Reported issues encountered during local and deployed usage

Feedback is documented and reviewed after each major milestone, demo, or iteration.

16.3 Key Feedback and Corresponding Actions

Feedback Area	User Input / Observation	System Response / Action Taken
Background Removal Cost Optimization	Supervisor suggested exploring Canva Premium background removal as a potentially	Option was evaluated conceptually; however, the supervisor later confirmed that the current background removal implementation was sufficient, and no

	lower-cost alternative for HR-side processing.	change was required. Existing solution retained, with Canva noted as an optional future alternative.
Position Input on New ID Template	HR clarified that for the new ID template, positions should no longer be manually typed by employees.	Position handling was aligned to predefined options and HR-controlled logic rather than free-text manual input.
Personal Details – Name Suffix	HR requested support for name suffixes (e.g., Jr., Sr., III) as part of employee personal details.	Suffix field was added to personal details handling and included in ID data mapping.
Signature Layout Adjustment	HR requested that the position label be moved closer to the signature area, specifically near the arm portion of the ID layout.	ID layout positioning was adjusted to relocate the position text near the signature area for better visual alignment.
Manual Name Entry Errors	HR observed that auto-linking names from Lark caused frequent errors; employees would still manually input names, sometimes incorrectly (e.g., Already indicated “MI”, still end-users keep entering full middle name instead of middle initial).	Manual name input was retained, and clear input instructions were added to the Personal Details section to guide proper formatting.
Temporary ID Expiration Rules	HR clarified that Temporary IDs should not have expiration dates, except for interns and freelance employees.	Expiration logic was updated to apply only to interns and freelancers; other temporary IDs remain without expiration.
Position Classification – “Others”	HR clarified that the “Others” option	Logic and documentation were updated to include Agents and Solutions under the “Others” classification.

	should include Agents and Solutions roles.	
Position List Management	Due to the large number of positions, HR suggested simplifying options by grouping them as Agents or Solutions rather than listing all roles.	Position selection logic was simplified while awaiting a finalized HR-provided position list.
Form Options Confirmation	HR confirmed that the current form options (Field Officer, Freelancer, Intern, Others) were acceptable as-is.	Existing form options were retained without changes.
Screenshot Restriction (Employee Side)	HR and supervisors suggested that end users should not be allowed to take screenshots while filling out the ID application form, to prevent data leakage and misuse of personal information.	Screenshot restriction was identified as a security and privacy enhancement. The requirement was documented and evaluated for implementation feasibility, with enforcement planned at the application and browser-behavior level where possible.
HR Dashboard Access	Since only HR has access to the HR Dashboard, HR requested the ability to take screenshots freely for reporting and issue escalation.	Screenshot restrictions were not enforced for HR users to allow easier documentation and reporting.
Point of Contact (POC) Notifications	HR clarified that specific individuals must receive POC notifications.	POC notification requirements were noted for targeted email or message routing.
Drive Access Without Login	HR asked whether files stored in Drive could be accessed without logging into the system.	Confirmed and implemented via publicly accessible Drive links with controlled permissions.
Missing Location Field	HR pointed out that the employee form lacked a	A mandatory Location / Branch field was added to the employee submission

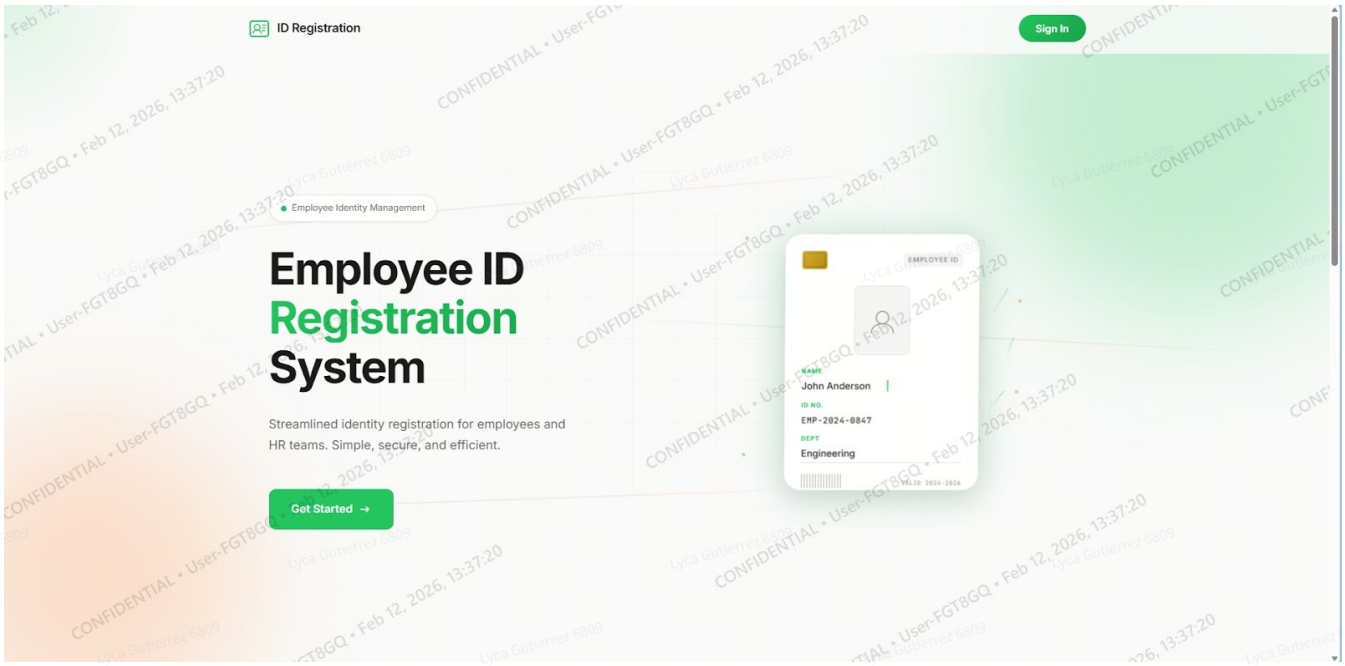
	location/branch field, which is required.	form.
Company Selection – SPMA vs SPMCorp	HR clarified that SPMA should not be selectable by employees in the form.	Company assignment logic was removed from employee input and handled internally.
Company Access Rules	HR clarified that only SPMA employees should be able to access the system, except for specific field roles.	Access rules were refined to restrict system access accordingly, with exceptions handled by role.
Multiple ID Templates	HR confirmed that SPMA and SPMCorp require different ID templates.	Template separation was documented and planned for role- and company-based rendering.
Field Role Access	HR clarified that field positions may belong to either SPMA or SPMCorp, depending on assignment.	Company assignment was moved to internal logic rather than employee input to avoid confusion.
Company Assignment Accuracy	HR noted that some employees do not know which company they belong to.	Company assignment responsibility was moved to the system/HR side, based on internal directories.
Facial Feature Validation	A test was conducted using an image with a facial mole, which successfully passed AI processing.	Test confirmed AI reliability for facial feature retention; no changes required.
Facial Expression Requirement	HR specified that showing teeth is not allowed and that mouths should be closed in ID photos.	Instructions were added and AI prompt constraints were adjusted to enforce closed-mouth expressions.
Scanned / Low-Quality Images	HR asked whether scanned ID photos or wallet-sized images could be accepted,	Image handling logic and AI processing were tested against scanned and low-quality inputs, with acceptable results noted.

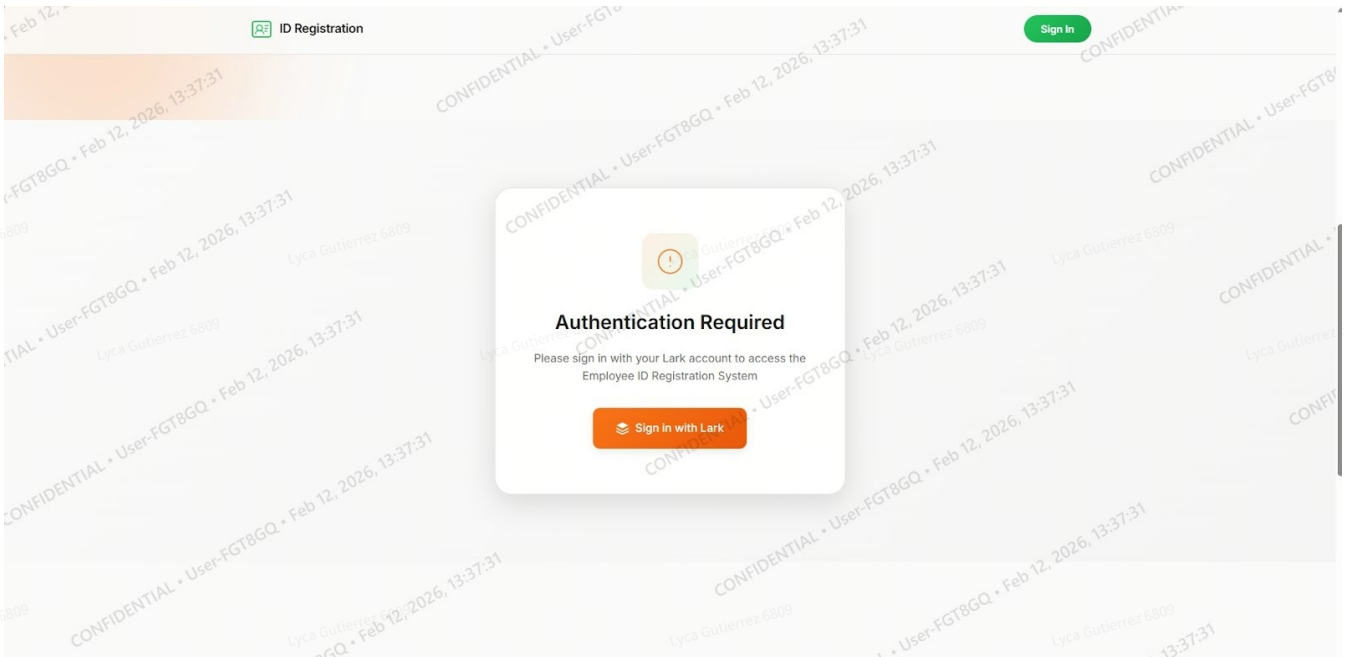
	especially for field employees.	
Mobile Camera Constraints	HR highlighted that field employees may submit blurred or low-quality images via mobile phones.	AI enhancement and validation logic were tuned to accommodate mobile-captured images where possible.
Accessories (Eyeglasses)	Optional suggestion to ask employees to remove eyeglasses for cleaner photos, possibly via HR announcement.	Not enforced programmatically; added as optional instruction to be communicated by HR.
PDF Download Concerns	HR questioned the need to download ID PDFs locally, citing storage limitations on HR laptops. HR further suggested that the "Download PDF" button be removed entirely, as downloading is not necessary for their workflow.	The "Download PDF" option was deprioritized in favor of server-side generation and controlled distribution. IDs are now intended to be generated and sent directly to designated POCs via email, reducing local storage dependency.
Barcode Content	HR requested the removal of numeric values displayed alongside barcodes.	Barcode rendering was updated to display the barcode only, without visible numeric text.
Outfit Guidelines (CEO Feedback)	CEO suggested modernizing attire guidelines: casual but presentable, fashionable styles, and avoiding outdated looks.	AI prompt guidelines were updated to reflect modern, professional attire standards.
Multiple Outfit Variations	CEO suggested providing multiple outfit options rather than a single fixed look.	Multiple AI styling variations were explored and noted for enhancement flexibility.

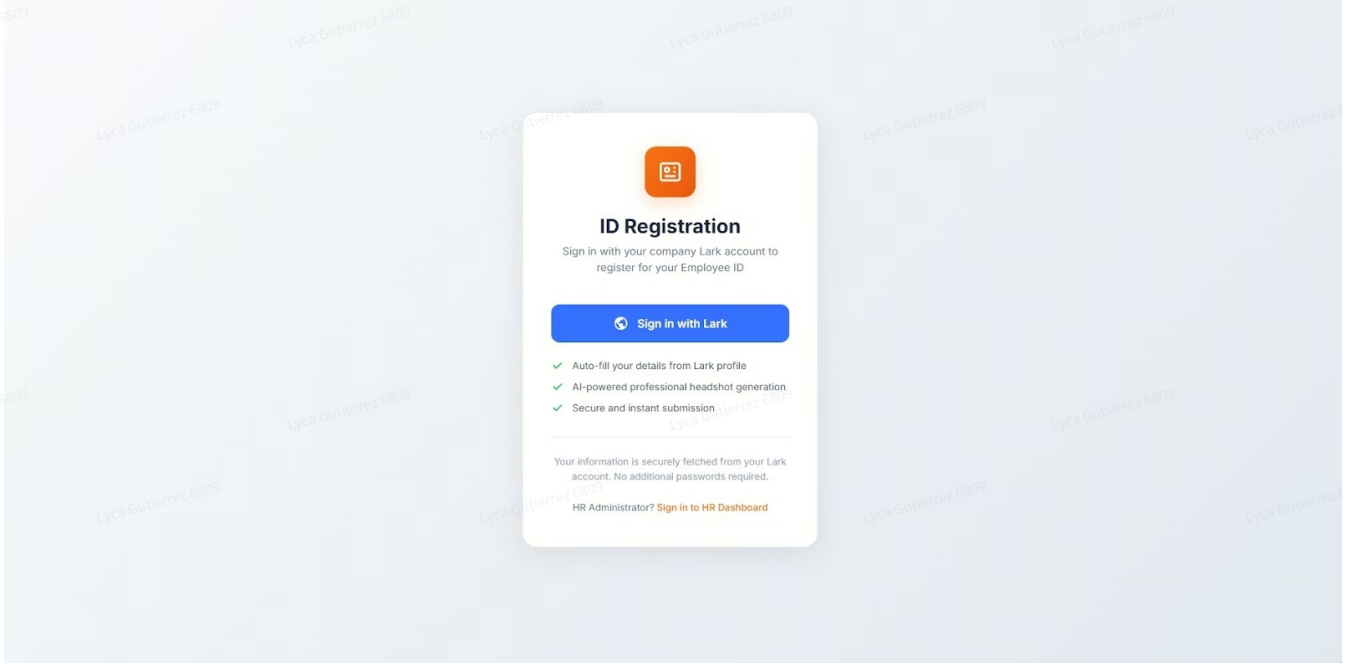
17. UI/UX Mock ups

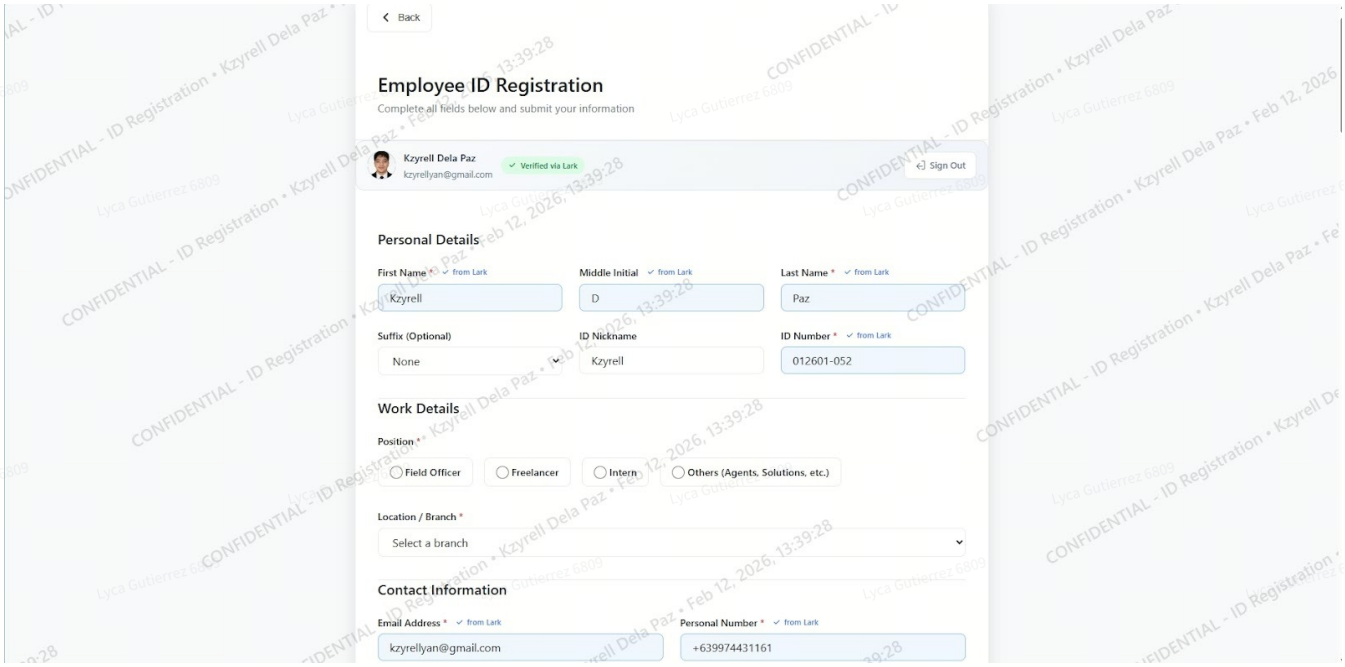
17.1 End User (Employee)

The Employee Portal is a self-service flow where employees log in, fill out their personal and ID details, upload or generate a professional headshot, and preview their ID in real time before submitting. The UI is designed to be guided and error-resistant, with clear field instructions, required field validation, and an easy-to-use signature canvas. The goal is to make submissions fast, consistent, and complete—minimizing back-and-forth with HR and ensuring the employee’ s ID data and photo meet standard requirements on the first try.





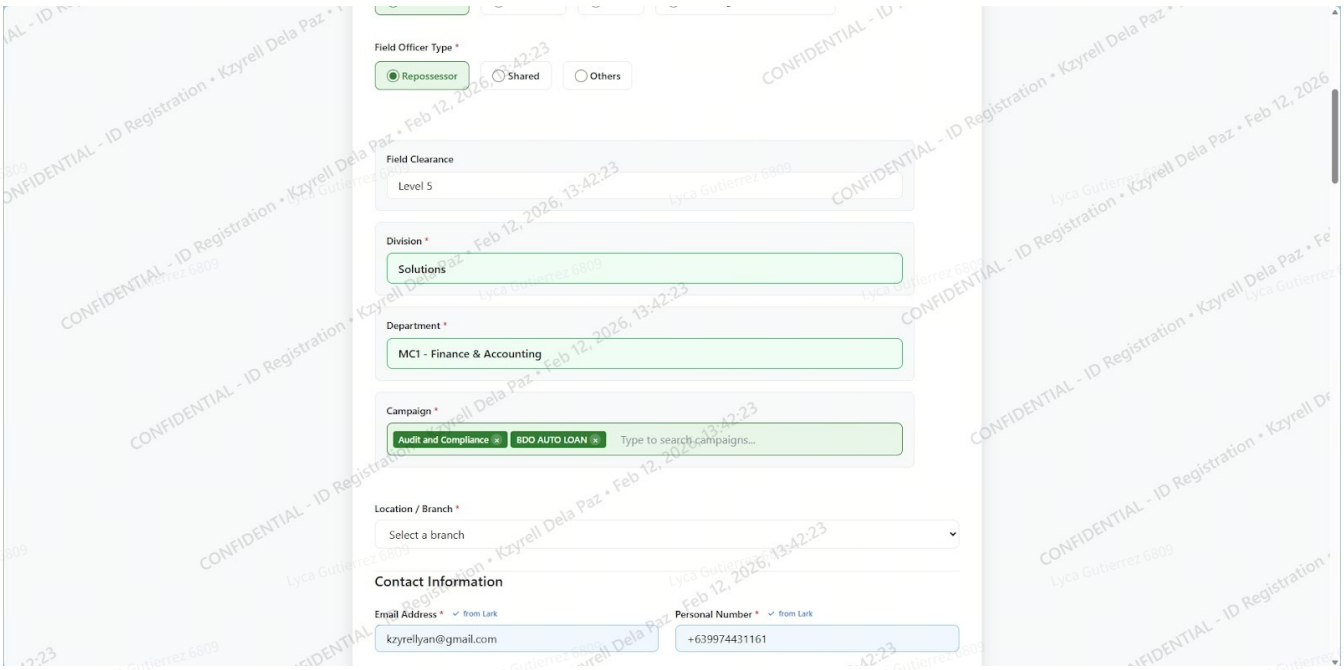

ID Registration
Sign in with your company Lark account to register for your Employee ID
Sign in with Lark
Auto-fill your details from Lark profile
AI-powered professional headshot generation
Secure and instant submission
Your information is securely fetched from your Lark account. No additional passwords required.
HR Administrator? Sign in to HR Dashboard


Back
Employee ID Registration
Complete all fields below and submit your information
Kzyrell Dela Paz
kzyrellyan@gmail.com
Verified via Lark
Sign Out
Personal Details
First Name * from Lark
Kzyrell
Middle Initial from Lark
D
Last Name * from Lark
Paz
Suffix (Optional)
None
ID Nickname
Kzyrell
ID Number * from Lark
012601-052
Work Details
Position *
Field Officer
Freelancer
Intern
Others (Agents, Solutions, etc.)
Location / Branch *
Select a branch
Contact Information
Email Address * from Lark
kzyrellyan@gmail.com
Personal Number * from Lark
+639974431161


Click to upload or drag and drop
PNG, JPG up to 5MB
Original Photo
AI Enhanced
Outfit:
Navy Polo Shirt
Regenerate
Male Attire (Smart Casual)
Navy Polo Shirt
Light Blue Shirt + Cardigan
White Polo Shirt
Soft Gray Collared Shirt
Female Attire (Smart Casual)
Sage Green Blouse
Cream Collared Blouse
Dusty Rose + Cardigan
Soft Blue Collared Shirt
Digital Signature
Sign Below *


ID Card Preview
Live preview of your ID card. Fill in the form to see changes. (Note: Your Image background will be removed)
Front
Back
SPM S.P. MADRID
Kzyrell
TEST
Kzyrell D. Paz
012601-052
www.spmadrid.com


Front
Back
SPM S.P. MADRID
17th Floor Chatham House Valero, Cor V.A. Rufino
St. Lgry, Makati City Metro Manila
Kzyrell's Contact
In case of emergency:
Emergency Contact Name
09XXXXXXXXX
Contact Address
www.okpo.com/spm/kzyrellpd


Field Officer Type *
Repossessor
Shared
Others
Field Clearance
Level 5
Division *
Solutions
Department *
MC1 - Finance & Accounting
Campaign *
Audit and Compliance
BDO AUTO LOAN
Type to search campaigns...
Location / Branch *
Select a branch
Contact Information
Email Address * from Lark
kzyrellyan@gmail.com
Personal Number * from Lark
+639974431161

ID Card Preview

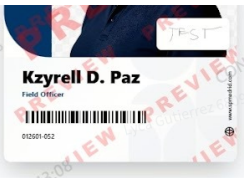
Live preview of your ID card. Fill in the form to see changes. (Note: Your Image background will be removed)

SPMC ID CARD

Front | Back

SPMA ID CARD

Front | Back



SPMA ID CARD

Front | Back

Review Your Information

Please verify all details are correct before submitting.

PERSONAL DETAILS

FIRST NAME	M.I.	LAST NAME
Kzyrell	D	Paz

Review Your Information

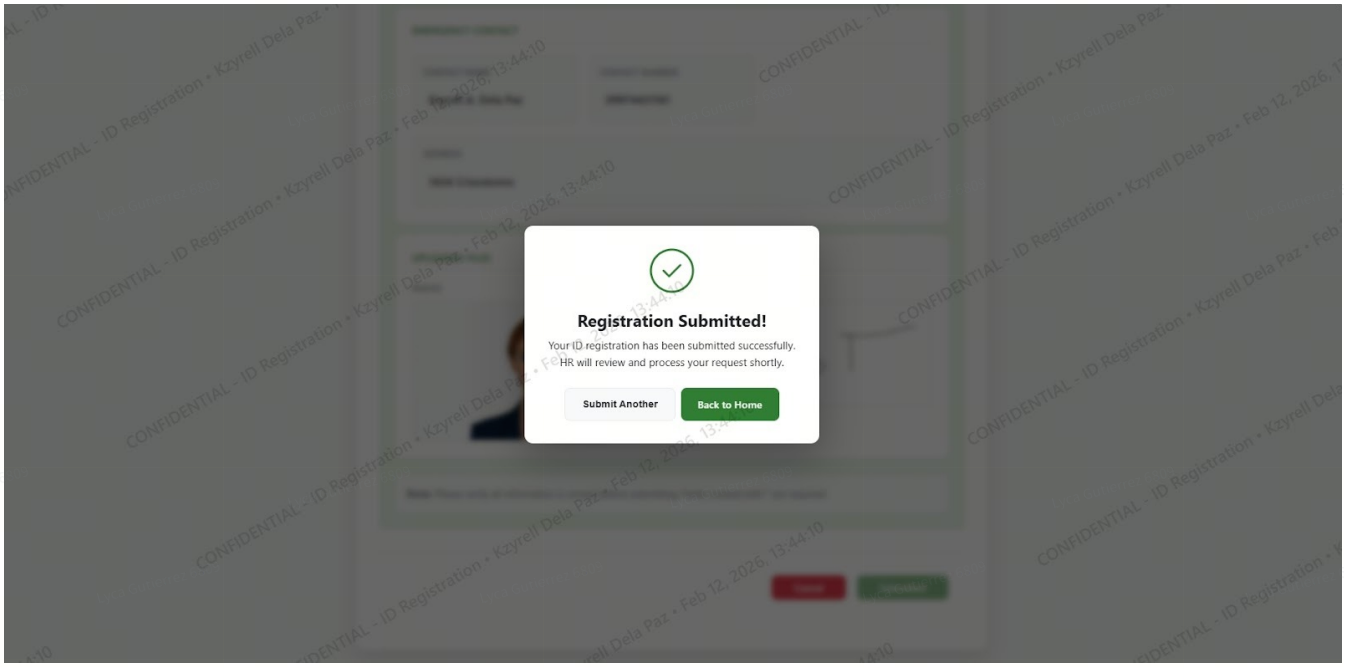
Please verify all details are correct before submitting.

PERSONAL DETAILS

FIRST NAME	M.I.	LAST NAME
Kzyrell	D	Paz
SUFFIX	ID NICKNAME	ID NUMBER
-	Kzyrell	012601-052

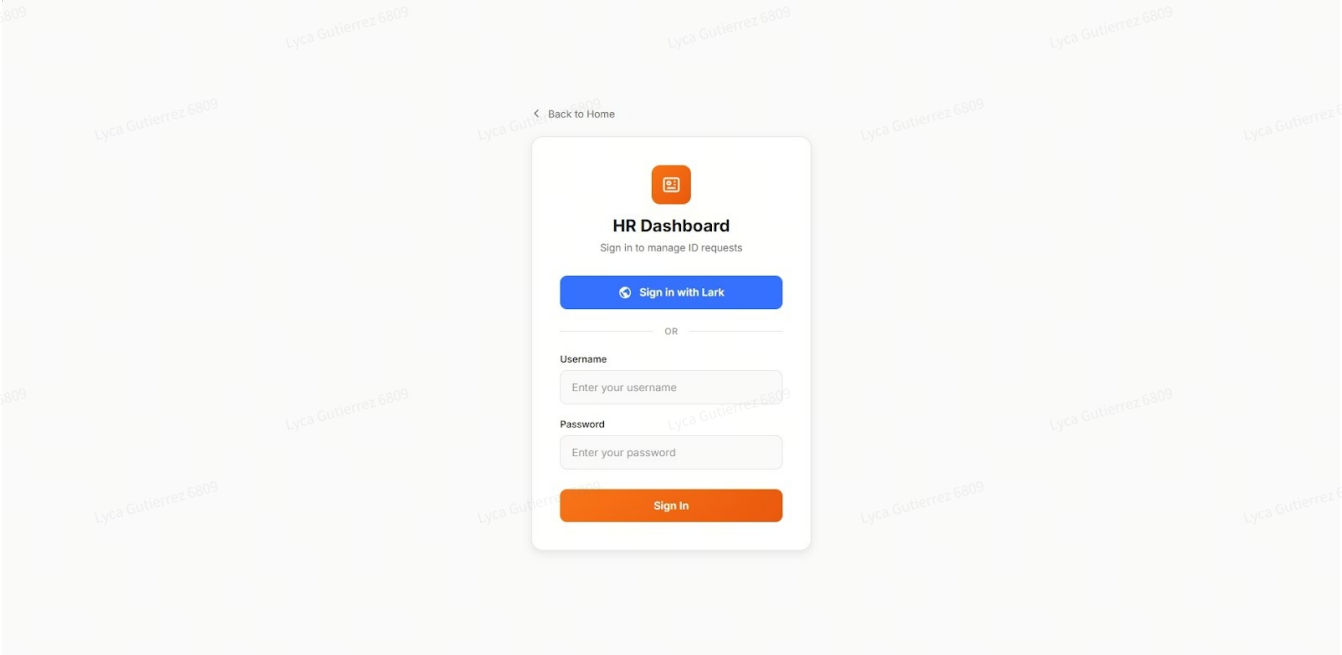
WORK DETAILS

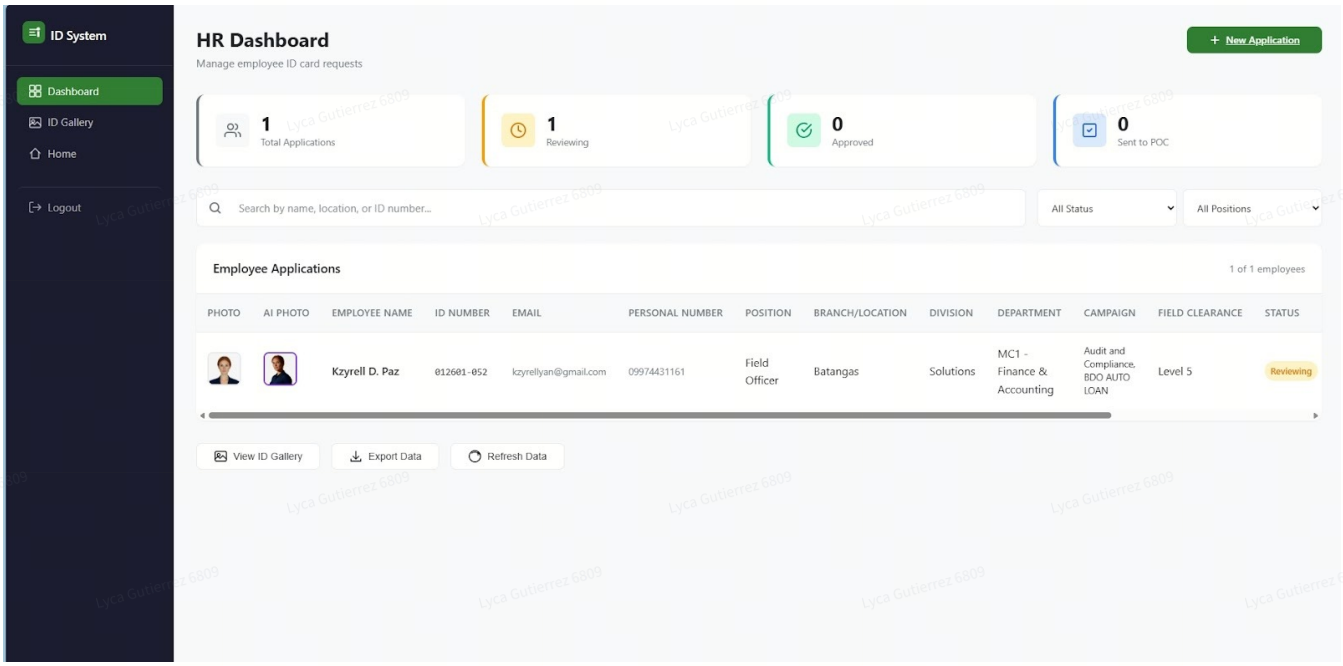
POSITION	LOCATION / BRANCH
Field Officer	-
FIELD OFFICER TYPE	FIELD CLEARANCE	DIVISION
Reprocessor	Level 5	Solutions
DEPARTMENT	CAMPAIGN
MC1 - Finance & Accounting	Audit and Compliance,BDO AUTO LOAN

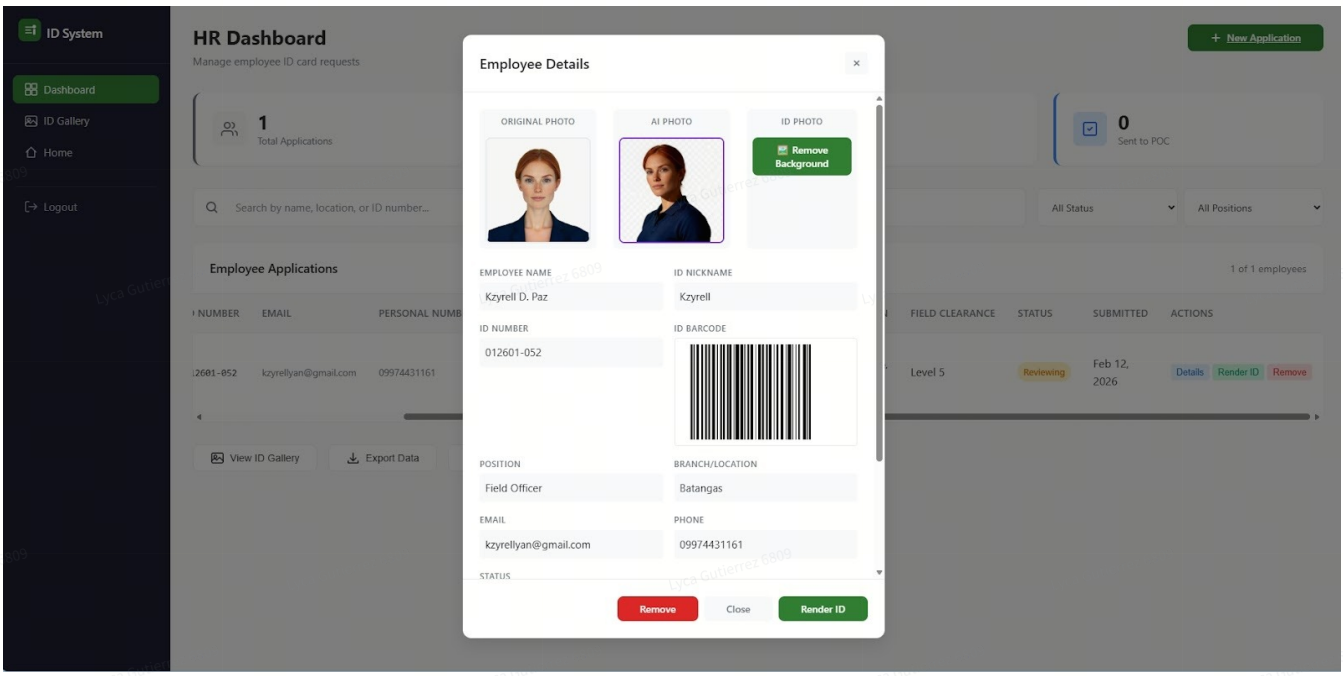


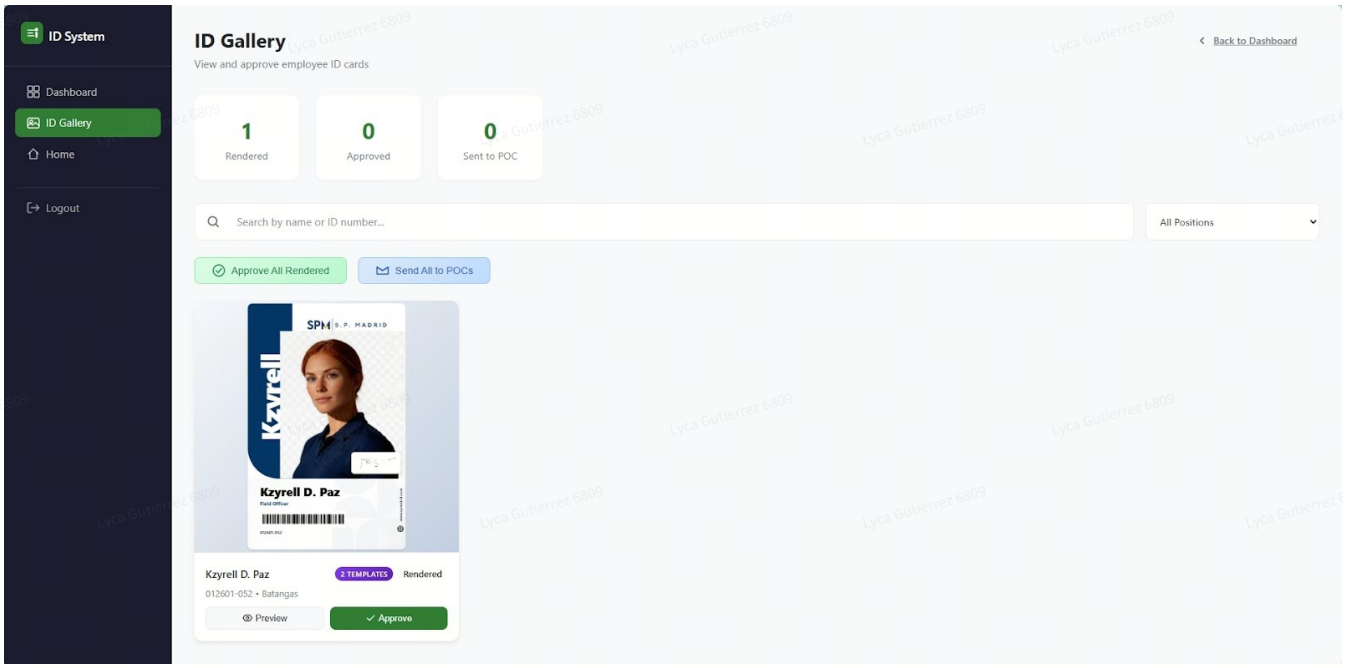
17.2 HR User

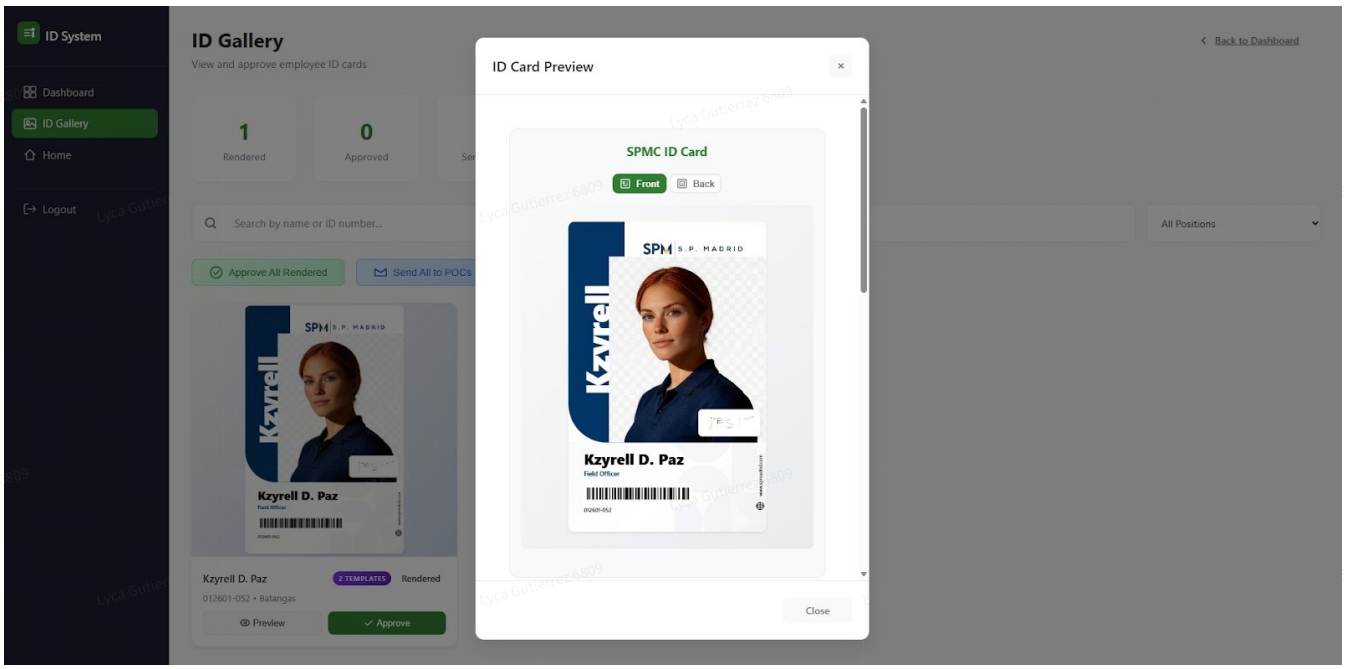
The HR Dashboard is an internal admin workspace for People Support to review and manage employee submissions at scale. It provides searchable records, status tracking, and preview views so HR can quickly validate details, check photo/signature compliance, and move requests through the workflow (e.g., Reviewing → Approved → Sent to POC → Completed). The UI supports efficient processing through bulk actions and export tools, allowing HR to generate print-ready outputs, reduce rework, and ensure each completed ID is routed to the correct POC for printing and distribution.

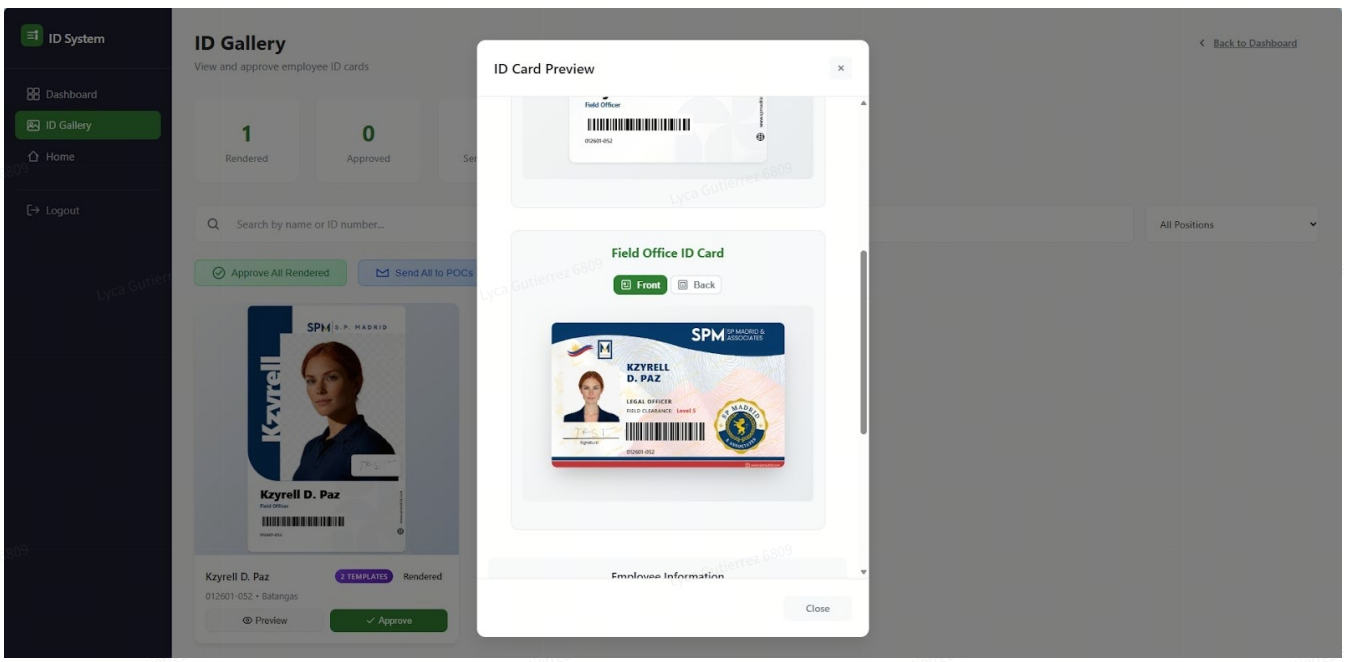

Back to Home
HR Dashboard
Sign in to manage ID requests
Sign in with Lark
OR
Username
Enter your username
Password
Enter your password
Sign In


ID System
Dashboard
ID Gallery
Home
Logout
HR Dashboard
Manage employee ID card requests
+ New Application
1 Total Applications
1 Reviewing
0 Approved
0 Sent to POC
Search by name, location, or ID number...
All Status
All Positions
Employee Applications
1 of 1 employees
PHOTO
AI PHOTO
EMPLOYEE NAME
ID NUMBER
EMAIL
PERSONAL NUMBER
POSITION
BRANCH/LOCATION
DIVISION
DEPARTMENT
CAMPAIGN
FIELD CLEARANCE
STATUS
Kzyrell D. Paz
012601-052
kzyrellyan@gmail.com
09974431161
Field Officer
Batangas
Solutions
MC1 - Finance & Accounting
Audit and Compliance, BDO AUTO LOAN
Level 5
Reviewing
View ID Gallery
Export Data
Refresh Data


ID System
HR Dashboard
Manage employee ID card requests
+ New Application
Dashboard
ID Gallery
Home
Logout
1 Total Applications
0 Sent to POC
Search by name, location, or ID number...
All Status
All Positions
Employee Applications
1 of 1 employees
EMAIL
PERSONAL NUMB
FIELD CLEARANCE
STATUS
SUBMITTED
ACTIONS
012601-052
kzyrellyan@gmail.com
09974431161
Level 5
Reviewing
Feb 12, 2026
Details
Render ID
Remove
View ID Gallery
Export Data
Employee Details
ORIGINAL PHOTO
AI PHOTO
ID PHOTO
Remove Background
EMPLOYEE NAME
Kzyrell D. Paz
ID NICKNAME
Kzyrell
ID NUMBER
012601-052
ID BARCODE
POSITION
Field Officer
BRANCH/LOCATION
Batangas
EMAIL
kzyrellyan@gmail.com
PHONE
09974431161
STATUS
Remove
Close
Render ID


ID System
Dashboard
ID Gallery
Home
Logout
ID Gallery
View and approve employee ID cards
Back to Dashboard
1
Rendered
0
Approved
0
Sent to POC
Search by name or ID number...
All Positions
Approve All Rendered
Send All to POCs
Kzyrell D. Paz
Kzyrell D. Paz
2 TEMPLATES
Rendered
012601-052 • Batangas
Preview
Approve


ID System
ID Gallery
View and approve employee ID cards
Back to Dashboard
ID Card Preview
SPMC ID Card
Front
Back
Kzyrell D. Paz
Close


ID System
ID Gallery
View and approve employee ID cards
Back to Dashboard
ID Card Preview
Field Office ID Card
Front
Back
KZYRELL
D. PAZ
LEGAL OFFICER
Kzyrell D. Paz
Close

18. Frequently Asked Questions (FQAs)

This section provides clear and concise answers to common questions related to the Employee ID Automation system. It is intended to help employees, HR personnel, and stakeholders understand how the system works, what to expect during usage, and how certain scenarios are handled. The FAQs aim to reduce confusion, minimize repetitive inquiries, and support smoother adoption of the system across the organization.

18.1 End User (Employee) FAQs

Q1. Who can use the Employee ID Automation system?

The system can be used by employees who are required to request or update their company ID. Access is limited to authorized users to ensure data accuracy and security.

Q2. How do I submit an ID request?

Employees submit an ID request by completing the online form, providing the required personal details, uploading a photo and signature, and reviewing the live ID preview before submission.

Q3. Do I need to edit or resize my photo before uploading?

No. The system automatically enhances the uploaded photo and removes the background to meet ID photo standards.

Q4. Can I preview my ID before submitting the form?

Yes. A real-time ID preview is available while filling out the form, allowing you to verify details before final submission.

Q5. Can I edit my submission after it has been sent?

Once submitted, the request is forwarded for HR review. Any required changes must be coordinated with HR.

Q6. Will the system work on mobile devices?

Yes. The system is responsive and can be accessed using both desktop and mobile web browsers.

18.2 HR User FAQs

Q1. Who can access the HR Dashboard?

Only authorized HR personnel are allowed to access the HR Dashboard. Authentication is required to ensure controlled access to employee records and ID requests.

Q2. How does HR review employee ID requests?

HR users can view submitted ID requests through the dashboard, review employee information, and proceed with approval or further processing as needed.

Q3. Can HR manage multiple ID requests at once?

Yes. The HR Dashboard is designed to handle multiple submissions, allowing HR users to efficiently manage and track ID requests.

Q4. Where is employee data stored after submission?

Employee data is stored within the system' s configured database. Optional integrations allow data to be synchronized with external platforms when enabled.

Q5. What happens if an external integration fails?

If an optional integration becomes unavailable, the system continues to operate normally and ensures that employee submissions are still recorded.

19. In Scope

This section outlines all functionalities, system behaviors, integrations, and operational responsibilities that are included within the scope of the Employee ID Automation project. These items define what the system is designed to support and deliver as part of its approved implementation.

19.1 Functional Scope

The system includes the following functional capabilities:

- Submission of Employee ID requests through a centralized web-based form
- Collection of required employee information, including:
  - Full name and identifier details
  - Position and department
  - Uploaded photo and signature
- Automatic validation of required form fields prior to submission
- Real-time ID preview that updates dynamically based on user inputs
- AI-assisted image enhancement to standardize employee photos
- Automated background removal for uploaded or generated photos
- Generation of a standardized digital ID layout based on approved templates
- Storage of employee submissions for tracking and review
- Status tagging of submissions (e.g., submitted, under review, approved)

- Controlled access for HR personnel to review employee ID requests
- HR capability to view, manage, and process multiple ID requests
- Export of individual or multiple ID records for operational use

19.2 Technical Scope

The following technical components are included:

- Backend application built using a Python-based web framework
- Frontend interface implemented using web technologies (HTML, CSS, JavaScript)
- Local database support for development and testing environments
- Cloud-compatible deployment using serverless infrastructure
- Secure image storage using third-party cloud services
- Integration with AI services for image generation and enhancement
- Optional integration with external data platforms (e.g., spreadsheets or databases) when configured
- Environment-based configuration using environment variables
- Logging and error-handling mechanisms for operational visibility

19.3 Security and Access Scope

Security-related inclusions consist of:

- Session-based authentication for HR access
- Restricted HR Dashboard access based on authorized user credentials
- Controlled exposure of employee data to authorized roles only
- Secure handling of uploaded images and personal information
- Use of environment variables for sensitive configuration values

19.4 Operational Scope

The operational responsibilities covered include:

- Deployment of the system in local and cloud environments
- Basic system monitoring through logs and error messages
- Manual oversight by HR for approval and decision-making
- Configuration and maintenance of environment variables
- Optional setup and management of third-party integrations

- Support for iterative enhancements based on internal feedback

19.5 User Scope

The system explicitly supports the following user groups:

- **End Users (Employees)** submitting ID requests
- **HR Users** responsible for reviewing and approving ID requests

20. Out of Scope

Defined here are the functionalities, responsibilities, and system behaviors that are explicitly excluded from the scope of the Employee ID Automation project. Items listed here are not part of the current implementation and would require separate approval, planning, and resources if considered in the future.

20.1 Functional Exclusions

The following functionalities are not included:

- Physical printing and distribution of employee ID cards
- Issuance, tracking, or lifecycle management of physical ID hardware
- Automatic deactivation or revocation of employee IDs upon resignation or termination
- Payroll, attendance, or timekeeping integration
- Biometric authentication (e.g., fingerprint, facial recognition for access control)
- Real-time synchronization with enterprise HR systems beyond configured integrations
- Automated approval logic without human HR review
- Employee role-based permission management beyond defined HR access

20.2 Technical Exclusions

The following technical elements are excluded:

- Custom mobile application development (Android or iOS)
- Offline system usage or local-first synchronization
- High-availability clustering or multi-region redundancy
- Performance load testing beyond development validation
- Data migration from legacy HR systems
- Automated database failover or disaster recovery infrastructure
- Long-term archival or cold storage management

- Custom analytics dashboards outside the HR portal

20.3 Operational and Support Exclusions

The following operational responsibilities are excluded:

- 24/7 system monitoring or on-call production support
- Dedicated helpdesk or ticketing system integration
- End-user training programs or formal onboarding materials
- SLA-backed uptime guarantees
- Automated incident response or escalation workflows
- Ongoing system administration beyond basic configuration

20.4 Third-Party and Cost Exclusions

The project does not cover:

- Subscription costs for third-party services beyond free or trial tiers
- Licensing fees for paid AI, storage, or database platforms
- Vendor support agreements
- Procurement or management of external infrastructure
- Future pricing changes of integrated services