ELIKSER PEOPLEOPS V3 · MODERNIZATION DOCUMENTATION

Business & Functional Requirements Document (PRD)

Elikser Employee Portal Modernization (V3) · Product Team Specification v1.1 · November 20, 2026

Product Requirements Document (PRD): Elikser Employee Portal Modernization (V3)

Document Version: 1.1 (Approved Target Stack & Environment Baseline)
Date: November 20, 2026
Approved Target Stack: Angular LTS (`web/`) · .NET 10 Web API (`api/`) · Azure SQL (`sqldb-employee-<env>`) · Azure Blob Storage (`stemployee<env>`) · Microsoft Entra ID Corporate SSO
Target Environments & URLs:

- **Interactive Mockups (Live Review):** `https://mockups.employee.elikser.com` (local: `docs/mockups/index.html`)
- **DEV:** `https://employee.dev.elikser.com` · API: `https://service.employee.dev.elikser.com` · DB: `sqldb-employee-dev`
- **QA:** `https://employee.qa.elikser.com` · API: `https://service.employee.qa.elikser.com` · DB: `sqldb-employee-qa`
- **PROD:** `https://employee.elikser.com` · API: `https://service.employee.elikser.com` · DB: `sqldb-employee-prod`

Source Codebases Audited (Scheduled for Read-Only Archival):

- **V1 Generation (`EmployeePortal-Legacy-V1`):** `EmployeePortal-Service` (.NET Core 3.1 API + SQL Snapshot) & `EmployeePortal-FrontEnd` (Angular 8)
- **V2 Generation (`EmployeePortal-Legacy-V2`):** `EmployeePortal-2-Service` (.NET 10 Upgraded Solution) & `EmployeePortal-2-Web` (Angular 8 + MSAL)
- **Migration Baseline:** `DOCS_SESSION_MIGRATION.md`

Companion Deliverables:

- Management Modernization Proposal: `docs/EmployeePortal_V3_Management_Proposal.md`
- Environment & V3 Database Architecture: `docs/Environment_URLs_And_Database_Design_V3.md` · `docs/sql/V3_Database_Schema.sql`
- QA How-To & Testing Guide: `docs/QA_How_To_And_Testing_Guide.md`

---

1. Executive Summary & Modernization Context

1.1 Problem Statement

Elikser operates two overlapping generations of an internal Employee Portal (**V1** and **V2**). Both generations grew organically over cloned starter templates (*SkolHub / Akhor / PAWS*) without consistent architectural standards, server-side authorization, or cross-module data integrity. As a result:

1. **Critical Security Exposure:** Both V1 and V2 accept unsigned Microsoft/Google JWTs without cryptographic verification (`AuthenticationService.cs`), lack role/ownership enforcement on API controllers (allowing any logged-in user to alter their own or others' `BasicSalary` via `POST AppUser/MyProfile` , view any employee's CNIC/bank details, or trigger mass emails via `[AllowAnonymous]` endpoints), and contain plaintext secrets in repository configs.
2. **Feature Fragmentation Between V1 & V2:** V2 modernized authentication (Microsoft Entra ID MSAL), dynamic RBAC (`Role` & `Feature`), Applicant Tracking (ATS), Boarding Checklists, E-Signed Resignation, and Unified Requests—but **dropped** V1's Monthly Payroll engine, Payslip PDF generation, Salary Discrepancy Claims, and Experience Letters.

3. **Broken Core Workflows & High Bug Density:** Multiple production journeys fail or corrupt data—including missing `EmployeeResignationLetter.docx` templates, resignation date pickers blocking future notice dates, HR receiving the resigning employee's acceptance email, V1 Sunday timesheet hours overwriting Wednesday, V2 `EditAttendance` always inserting duplicate rows, Claim approvals hiding `Amount` and `Month`, and files stored as 5 MB-limited Base64 strings inside `localStorage` and SQL `NVARCHAR(MAX)`.

1.2 Business Objectives

- **Consolidate V1 + V2 into a Single Cohesive Platform ("Employee Portal V3"):** Restore Monthly Payroll, Payslips, and Experience Letters alongside V2's ATS, Lifecycle Checklists, Requests, and Knowledge Hub.
- **Zero-Trust Security by Design:** Enforce Microsoft Entra ID JWKS validation, server-side RBAC + resource ownership policies, object-storage document vaulting, and immutable audit trails.
- **Eliminate Workflow Dead-Ends:** Connect ATS Hire -> Employee Onboarding -> Probation Confirmation -> Attendance/Leave/Timesheets -> Payroll -> Resignation/Offboarding into one continuous lifecycle.
- **Modern, Responsive, Accessible UI:** Replace jQuery/Bootstrap 4 modal soup and copy-pasted validation errors with an intuitive, responsive Angular LTS + Tailwind/CDK UI achieving WCAG 2.1 AA contrast and clean state management (no `localStorage` data bridges that break browser refresh).

1.3 V1 vs. V2 vs. Modernized Target Reconciliation Matrix

Capability Domain	Legacy V1 State	Legacy V2 State	V3 Target Specification (This PRD)
Authentication & SSO	Unverified Google `gapi.auth2` (`loginUsingGoogleAuth`); dead password form	Microsoft Entra ID MSAL (`@azure/msal-browser`), unverified JWT parse on server, unfinished `"welcome note here"`	Verified Entra ID JWKS + Tenant validation, full MSAL + storage logout, 4-step guided First-Time Onboarding Wizard
RBAC & User Admin	4 static roles (`Admin` , `HR` , `Lead` , `TeamMember`), split `/admin` & `/employee` shells	Dynamic `Role` + `Feature (1..24)` + `isLead` , cosmetic sidebar `*ngIf` , zero backend RBAC	Server-enforced Policy RBAC (`View` / `Create` / `Edit` / `Approve` / `Export`), resource-ownership guards, separated self-service vs management scopes
Dashboard & Alerts	Plain announcement list only	Check-In/Out (WFH) + basic links + manual-refresh notifications (`EP.Socket` empty)	Role-adaptive KPI widgets, live shift timer, Unified Approvals Queue, SignalR WebSocket push + email notifications
Recruitment & ATS	Basic Interview Records, Q&A evaluation, static Contract/NDA PDF	Multi-stage ATS, Google Calendar sync, 1–5 star scorecards, Question Bank + XLSX import, Offer/NDA + Convert to User	Unified Kanban/Table ATS, inline Question Bank reference, fixed deep-linking & filter bugs, 1-click Hire -> Onboarding pipeline
Employee Directory & Profile	Modal-based profile + separate Resume list	5-tab profile + Checklists (`Onboarding` , `Probation` , `Offboarding`); `childName.split` crash; salary IDOR	Secured 5-tab 360° profile, field-level salary lock, multi-child add/remove, unified CV/Skills & Document Vault
Resignation & Offboarding	Not present	E-Signature resignation (`angular2-signaturepad`); missing `.docx` template, `[max]=today` , wrong email recipient	Future notice-date calculation (`min=today`), valid `.docx` template, employee-directed notifications, auto-triggered Offboarding checklist
Attendance & WFH	Check-In/Out + Edit (`Reason` silently discarded); silent 2 PM auto-checkout	WFH flag + Edit (inserts `attendanceId=0` duplicate); inverted `isWFH` Excel export	Accurate shift state, missing check-out regularization request workflow, in-place edits with mandatory immutable Audit Trail

Capability Domain	Legacy V1 State	Legacy V2 State	V3 Target Specification (This PRD)
Projects & Timesheets	Weekly grid (Sunday->Wednesday bug, hardcoded `"aaa"`, blocked Rejected resubmit)	Weekly + Daily editor (`monHourshours` crash); Lead queue; hidden from Admin/Exec	Validated Mon–Sun + Daily task drawer, Rejected re-submission enabled, unassignable Activity Groups, Lead + Executive escalation routing
Requests (Leave, Letters, Claims, Items)	Leave only (manual days, no balance check)	4 request cards; manual leave days; `data[0]` crash; Claim table hides `Amount` / `Month`; New Items tab bug	Holiday-aware working-day auto-calculator, locked approved leaves, Amount/Month/Receipt in Claim approval, Experience + Bank + Probation + Salary Letters
Payroll & Payslips	**Present:** Monthly generation, Edit (`GrossIncome` cumulative bug), Exec approval, PDF payslips, read-only claims	**Dropped completely**	**Restored & Fixed:** Editable pay periods, holiday-adjusted working hours, idempotent adjustments, Exec approval, PDF Payslip Vault, closed-loop Claim resolution
Knowledge, Ideas & Blogs	Abstracts hidden; alpha-only regex; approved blogs unpublished	Versioned Policies/Benefits; CKEditor Blogs (broken delete `*ngIf`); unpublished approved blogs	Policy Read-Acknowledgment tracking, full readable Abstracts, Upvotable Ideas (creator-only edit), **Published Company Blog Reader Feed**
System Setup & Holidays	Hardcoded 22 activity names	Setup CRUD (`Job`, `Subject`, `ActivityGroup`, `Tool`, `Item` [Item edit broken]); **no Holidays table**	Complete Master Data Console + **Public Holiday Calendar & Leave Accrual Rules** + `.docx` Template Health Validator

2. Target Personas & Unified RBAC Matrix

2.1 Target Personas

Persona	Definition & Scope	Primary Goals
Employee (Team Member)	Every active staff member (`Probation` or `Permanent`), including Leads, HR, Executives, and Admins when acting for themselves.	Clock in/out (Office/WFH), log weekly timesheets, request leaves/letters/claims/items, view payslips, acknowledge policies, submit ideas/blogs, manage personal/family profile, or submit resignation.
Project Lead	Dynamic capability (`isLead = true`) granted to any employee assigned as `ProjectLeadId` on >= 1 active project.	Assign project Activity Groups to team members, review/approve/reject team weekly timesheets, review/regularize team attendance, monitor project utilization.
Interviewer	Employees assigned an Interviewer role (`RoleType = Interviewer`, e.g., Technical Lead, BA Lead, Architecture Panel).	View assigned scheduled interviews, consult Question Bank & Guidelines, submit 1–5 star Soft/Technical scorecards and stage recommendations.
HR Specialist / Manager	Human Resources operations staff.	Manage Recruitment/ATS, issue Offers/NDAs/Contracts/HR Letters, onboard & convert candidates, manage Probation/Offboarding checklists, approve Leaves/Documents/Items/Claims/Resignations, prepare Monthly Payroll, publish Policies/Benefits/Announcements, moderate Blogs.
Executive / Finance Approver	Senior leadership / Managing Director / Finance head.	Review & approve/reject Monthly Payroll batches, review company-wide Timesheet/Utilization/Attendance reports, co-approve escalations (Lead/HR timesheets, senior resignations), broadcast announcements.
System Administrator	IT / Portal Operations administrator.	Provision & deactivate user accounts, manage dynamic Roles, Features, granular Permissions, Master Data Setup, Public Holiday Calendar, and `.docx` Templates.

2.2 Unified Role-Based Access Control (RBAC) Matrix

Architectural Rule: Every employee—including HR, Executive, Project Lead, and Admin—retains **Self-Service (`Self`)** access to their own Attendance, Timesheets, Requests, Payslips, Profile, and Resignation. Elevated permissions apply strictly on the server via `[Authorize(Policy = "...")]` plus resource-ownership checks (`EmployeeId == CurrentUser.EmployeeId`).

Legend: `C` = Create, `R` = Read, `U` = Update, `D` = Soft-Delete, `A` = Approve/Reject, `E` = Export/Download, `(Self)` = Own records only, `(Team)` = Assigned project members only, `(All)` = Organization-wide.

Module	Capability	Employee	Project Lead	Interviewer	HR Manager	Executive	SysAdmin
MOD-01	Auth, Onboarding Wizard & Session	`R,U (Self)`	`R,U (Self)`	`R,U (Self)`	`R,U (Self)`	`R,U (Self)`	`R,U (Self)`
MOD-01	User Provisioning & Role/Feature RBAC	–	–	–	`C,R,U (Invite)`	`R`	`C,R,U,D (All)`
MOD-02	Dashboard & Personal Notifications	`R (Self)`	`R (Self+Team)`	`R (Self)`	`R (Self+All)`	`R (Self+All)`	`R (Self)`
MOD-02	Broadcast Announcements	–	–	–	`C,R,U,D,E`	`C,R,U,D,E`	`R`
MOD-03	Candidates, Scheduling, Offers & Hire	–	–	`R (Assigned)`	`C,R,U,D,E`	`R,A (Offer)`	–
MOD-03	Interview Scorecards & Question Bank	–	–	`C,R,U (Self)`	`C,R,U,D,E`	`R`	–
MOD-04	Employee Directory & 5-Tab Profile	`R,U (Self*)`	`R,U (Self*)`	`R,U (Self*)`	`C,R,U,E (All)`	`R,E (All)`	`R (Basic)`
MOD-04	Basic Salary & HR Document Uploads	`R (Self Docs)`	`R (Self Docs)`	`R (Self Docs)`	`C,R,U,D (All)`	`R (All)`	–
MOD-05	Boarding Checklists & Probation Confirm	`R (Self)`	`R,U (Team)`	–	`C,R,U (All)`	`R`	`U (IT Tasks)`
MOD-05	E-Sign Resignation & Approval	`C,R (Self)`	`C,R (Self)`	`C,R (Self)`	`R,A,E (All)`	`R,A (All)`	–
MOD-06	Check-In/Out, Regularization & Audit	`C,R (Self)`	`C,R,U,A (Team)`	`C,R (Self)`	`C,R,U,A,E (All)`	`R,E (All)`	`R (Audit)`
MOD-07	Projects, Teams & Activity Assignment	`R (Assigned)`	`R,U (Team Act.)`	`R (Assigned)`	`R,E`	`R,E`	`C,R,U,D`
MOD-07	Weekly/Daily Timesheets & Approvals	`C,R,U (Self)`	`A (Team)`	`C,R,U (Self)`	`C,R,U,E (All)`	`A (Lead/HR), E`	`C,R,U (Self)`
MOD-08	Leave, Letters, Claims & Item Requests	`C,R,U,D (Self)`	`C,R,U,D (Self)`	`C,R,U,D (Self)`	`R,A,E (All)`	`A (Esc.), E`	`U (IT Items)`
MOD-09	Monthly Payroll Run, Batch Approval & Claims	`R,E,C-Claim (Self)`	`R,E,C-Claim (Self)`	`R,E,C-Claim (Self)`	`C,R,U,E (Prep)`	`R,A,E (Batch)`	–
MOD-10	Policies & Benefits (Acknowledge / Manage)	`R,Ack`	`R,Ack`	`R,Ack`	`C,R,U,D`	`C,R,U,D`	`R,Ack`
MOD-10	Innovation Ideas & Blogs	`C,R,U,D (Self)`	`C,R,U,D (Self)`	`C,R,U,D (Self)`	`C,R,U,A (Blogs)`	`C,R,A (Blogs)`	`C,R,U,D (Self)`
MOD-11	Master Data, Holidays & `.docx` Templates	`R (Holidays)`	`R (Holidays)`	`R (Holidays)`	`C,R,U,D (HR/Cal)`	`R`	`C,R,U,D (All)`

* `Self*`: *Employees may edit Personal Contact, Address, Bank, Family, and Professional Project Responsibilities—but NEVER* `BasicSalary`, `EmployeeStatusId`, `JoiningDate`, *or* `AnnualLeaves`.

---

3. Detailed Functional Requirements by Module (`MOD-01` – `MOD-11`)

Module 01: Authentication, First-Time Onboarding & RBAC (`MOD-01`)

Mockup: `01-auth-onboarding-rbac.html`

`FR-AUTH-001` — Microsoft Entra ID SSO & Cryptographic Token Validation

- **Trigger:** User clicks `"Sign in with Microsoft"` on `/login`.
- **Pre-Conditions:** User has an `@elikser.com` (or configured tenant) Microsoft Entra ID account and an active `AppUser` record.
- **Detailed Flow:**
  1. Angular `@azure/msal-browser` acquires an Entra `idToken` (supporting popup and redirect fallback).
  2. Frontend posts `idToken` in request body (`POST /api/v1/auth/sso/entra`) — never via URL query string!
  3. Backend validates signature against Microsoft's OIDC JWKS endpoint (`https://login.microsoftonline.com/{tenantId}/discovery/v2.0/keys`), verifies `iss`, `aud` (`ClientId`), `exp`, and `nbf`.
  4. Backend extracts primary email claim (`preferred_username` -> `upn` -> `email`), queries `AppUser JOIN Employee`, verifies `AppUser.IsActive == true && !AppUser.IsDeleted` and `Employee.EmployeeStatusId NOT IN (Resigned, Terminated)`.
  5. Backend dynamically computes `isLead = Project.Any(p ⟹ p.ProjectLeadId == EmployeeId && !p.IsDeleted)`.
  6. Backend issues a short-lived portal access JWT (15–60 min) containing `sub`, `employeeId`, `roles`, `permissions`, `isLead` + `HttpOnly` refresh token cookie, returning user session metadata including `isFirstTimeLogin`.
- **Business Rules & Exceptions:**
  - Invalid/unsigned tokens MUST return `401 Unauthorized` and log a security audit event.
  - Inactive/terminated accounts MUST return `403 Forbidden` with message `"Your account is deactivated. Contact HR at hr@elikser.com."`
- **Legacy Issues Resolved:** Fixes V1/V2 `jwtHandler.ReadJwtToken(Token)` signature bypass, deactivated-user login bypass, and exposing JWT in query string `?Token=`.

`FR-AUTH-002` — Full Session Termination (Logout)

- **Trigger:** User clicks `Log Out` in profile menu or receives unrecoverable `401`.
- **Detailed Flow:** Clears all portal state in `localStorage` / `sessionStorage`, revokes server refresh token (`POST /api/v1/auth/logout`), clears MSAL active account cache (`logoutPopup` / `clearCache`), and redirects cleanly to `/login`.
- **Legacy Issues Resolved:** Fixes V2 `admin.component.ts` `test()` reloading page without clearing `localStorage['Auth']` or MSAL cache.

`FR-AUTH-003` — Guided First-Time Employee Onboarding Wizard

- **Trigger:** User logs in with `isFirstTimeLogin == true`.
- **Detailed Flow:**
  1. Routes to `/onboarding` instead of dashboard.
  2. Displays 4-step wizard: (1) Confirm Personal, Emergency & Bank Details; (2) Review & Acknowledge Mandatory Company Policies (IT Security, Code of Conduct); (3) View Assigned Project/Lead/Team Roster & HR Onboarding Tasks; (4) Interactive Portal Quick Tour.

3. On completion, calls `POST /api/v1/users/me/complete-onboarding` (idempotently flips `isFirstTimeLogin = false` and sends welcome email safely inside non-blocking background job).

- **Legacy Issues Resolved:** Replaces V2's unfinished `"welcome note here"` screen and broken template binding before async load.

`FR-AUTH-004` — User Provisioning & Granular RBAC Matrix

- **Trigger:** Admin opens `/admin/rbac` (`Manage Users`, `Manage Roles`, `Permission Matrix`).
- **Detailed Flow:**
  1. Admin creates or updates a user with `@elikser.com` (domain configurable in appsettings) email, Full Name, Active toggle, and multi-select Roles.
  2. Admin creates/edits Roles specifying `RoleType` (`1 = Portal Access Role`, `2 = Interviewer Stage Role`). Only Portal Access Roles attach module-level permissions (`View`, `Create`, `Edit`, `Approve`, `Export`).
  3. Frontend `RoleGuard` validates route metadata against token permissions; Backend enforces `[Authorize(Policy = "...")]` on every endpoint.
- **Legacy Issues Resolved:** Eliminates `userfeatureId == null` showing all admin menus, fixes `CreateUser` multi-select array/object deselect bug, fixes `"Delete Role"` mislabeled modal when deleting a user, fixes V1 `Admin` self-registration endpoint (`[AllowAnonymous] Register`), and enforces server-side RBAC.

Module 02: Role-Adaptive Dashboard & Real-Time Notifications (`MOD-02`)

Mockup: `02-dashboard-notifications.html`

`FR-DASH-001` — Role-Adaptive Workspace & Live Shift Timer

- **Trigger:** Authenticated user lands on `/dashboard`.
- **Detailed Flow:**
  1. Top bar / Hero displays quick **Clock In / Clock Out** widget with `Work From Home` toggle, live elapsed shift counter (`HH:mm:ss` in `America` / `Asia/Karachi` normalized UTC offset), and previous missing check-out warning if applicable.
  2. **KPI Strip** dynamically adapts to persona:
     - *All Employees:* Remaining Leave Balance (Days), Current Week Timesheet (`Logged / 40.0h`), Pending My Requests, Latest Payslip Net Amount.
     - *Project Lead:* Pending Team Timesheets count, Team Attendance Compliance %.
     - *HR / Executive:* Active Candidates in Pipeline, Pending Leave/Claim/Document/Resignation Requests, Payroll Batch Status.
  3. **Unified Approvals Queue** shows actionable cards with 1-click `Inspect` / `Approve` / `Reject` for items awaiting the current user's role.
- **Legacy Issues Resolved:** Replaces V1's empty notice-only dashboards and V2's static 4-link dashboard; fixes V1 `CheckIn.Date == Now` always-false check-in indicator.

`FR-DASH-002` — Real-Time Notifications (WebSocket/SignalR) & Targeted Announcements

- **Trigger:** System workflow event (leave approval, timesheet rejection, blog review) OR HR creates an announcement.
- **Detailed Flow:**
  1. Backend persists `Notification` row and pushes real-time event over **SignalR** (`/hubs/notifications`) to connected user/group channels.
  2. Topbar Bell icon increments unread badge immediately; drawer supports `Mark Read`, `Mark All Read`, and click-through deep link (`actionUrl`).
  3. HR/Executive creates Broadcast Announcements with `Title`, `Message` (rich text/multiline), `TargetAudience` (`All`, `Role Groups`, `Specific Individuals`), `ValidityFrom`, and `ValidityTo` (`ValidityTo ≥ ValidityFrom`).

4. HR can edit or archive any announcement at any time with audit logging (removing V2's arbitrary 1-hour edit cutoff and individual-only restriction).

- **Legacy Issues Resolved:** Populates V2's empty `EP.Socket` project; fixes V1 duplicate `"Notification Validity From*"` label and singular/plural `formHead` bug; fixes V2 `SendNotificationEmailToSingleUsers` hardcoding `hr@elikser.com`.

Module 03: Recruitment & Applicant Tracking System (`MOD-03`)

Mockup: `03-recruitment-ats.html`

`FR-ATS-001` — Candidate Pipeline & Resume Vault

- **Trigger:** HR opens `/recruitment/candidates` (Kanban or Table view) and clicks `+ Add Candidate` or selects a row.
- **Validations:** `FullName` (3–64 chars), `Email` (RFC-5322, unique active candidate), `ContactNumber`, `JobPositionId` (required), `Experience`, `PreviousSalary` (>= 0), `Degree`, `Discipline`, `City`, Resume upload (PDF/DOCX <= 10 MB via multipart `FormData` to blob storage).
- **Detailed Flow:** Selecting any candidate opens the Candidate 360° drawer fetched cleanly by `/api/v1/candidates/{id}` (supporting browser refresh & shareable URLs) with inline PDF resume viewer and chronological feedback history.
- **Legacy Issues Resolved:** Fixes `SharedDataService` / `localStorage` state loss on page refresh, Base64 `localStorage` quota crashes, and non-PDF CV preview failures.

`FR-ATS-002` — Multi-Interviewer Scheduling & Calendar Integration

- **Trigger:** HR clicks `Schedule Interview` or `Reschedule` on a candidate row/drawer.
- **Detailed Flow:** Selects Stage Role (`Interviewer` role type), 1+ Interviewers (`email` multi-select), `DateTime` (must be strictly in future), `Mode` (`Online / Video` vs `On-Site`), configurable meeting location/link, and notes. Saving checks duplicate active stage, creates/updates Calendar event (configurable OAuth/Service Account via Secret Manager, never hardcoded JSON), and notifies candidate + interviewers.
- **Legacy Issues Resolved:** Fixes V2 `candidate-info.component.ts` missing `router.navigate( ... )` when clicking Interview Date or Post Feedback; replaces hardcoded `STP4 Rawalpindi` address & `hr@elikser.com`.

`FR-ATS-003` — Structured Scorecard Evaluation, Question Bank & Guidelines

- **Trigger:** Assigned Interviewer opens `/recruitment/evaluations/{interviewId}` (from email link, dashboard, or candidate drawer).
- **Detailed Flow:**
  1. Left/main pane captures `SoftSkillsRating` (1–5 stars), `TechnicalSkillsRating` (1–5 stars), `ExpectedSalary`, `Remarks`, and Decision (`Move to Next Stage` + Next Role, `Eligible for Onboarding`, or `Rejected`).
  2. Right collapsible drawer displays filtered **Question Bank** (`Subject` + `QuestionType`, optional multi-filter working cleanly with either or both selected) and downloadable **Interview Guidelines**.
  3. HR manages Questions (single CRUD + `.xlsx` bulk template import with row-by-row validation error preview) and Guidelines separately.
- **Legacy Issues Resolved:** Fixes V2 `EvaluationInfoService.GetAllQuestions` sequential `if` query bug returning 0 rows when `questionTypeID == 0`, and fixes V2 Question delete icon accidentally invoking `removeGuides()`.

`FR-ATS-004` — Offer, NDA, Contract Generation & 1-Click Hire Onboarding

- **Trigger:** Candidate reaches `OnBoarding` / `Eligible` status and HR clicks `Make Offer` -> `Convert to Employee`.
- **Detailed Flow:**
  1. HR enters `FatherName`, `CNIC`, `Address`, `OfferedSalary`, `JoiningDate` (`> today`). System validates `.docx` templates, generates Contract & NDA PDFs via thread-safe document service, previews inline, and dispatches via email with delivery status tracking.

2. HR clicks `Convert to Employee`, which pre-populates `@elikser.com` email, Job Position, Joining Date, Offered Salary, Probation status (`Probation` default), and initial Portal Roles.
3. Backend wraps in a single DB transaction: creates `AppUser` + `AppUserToRole` + `Employee` (copying CNIC, Resume to Document Vault, Contract, NDA) + seeds **Onboarding & Probation Checklists** + initializes **Leave Accrual Balance** + links `Candidate.ConvertedEmployeeId`.

- **Legacy Issues Resolved:** Unifies V1's disconnected Contract/NDA workflow with V2's conversion flow; carries candidate CV/Contract/NDA directly into Employee Document Vault; fixes `CreateUser` multi-select role bug.

Module 04: Employee Directory & 5-Tab 360° Profile (`MOD-04`)

Mockup: `04-employee-directory-profile.html`

`FR-EMP-001` — Searchable Employee Directory

- **Trigger:** HR, Executive, or Admin opens `/employees`.
- **Detailed Flow:** Filterable directory displaying Employee ID, Name, Email, Job Position, Department, Status (`Probation`, `Permanent`, `Notice Period`, `Resigned`, `Terminated`), Joining Date, **Resignation / Last Working Date** (populated properly!), and quick link to `Boarding Checklists`.
- **Legacy Issues Resolved:** Populates V2's blank `Resignation Date` cell (`<td></td>`) and links V1's Resume directory directly into Tab 4/5.

`FR-EMP-002` — 5-Tab 360° Profile & Field-Level Access Protection

- **Trigger:** Employee clicks `My Profile` (`/profile/me`) OR HR clicks an employee (`/employees/{employeeId}`).
- **Tabs & Validation Rules:**
  1. **Personal & Emergency:** `CNIC` (`00000-0000000-0`), `ContactNumber`, `DOB` (`≤ today`), `BloodGroup`, `CurrentAddress`, `SameAsCurrent` sync checkbox, `PermanentAddress`. **`BasicSalary`, `JoiningDate`, `JobPosition`, `Department`, `EmployeeStatusId` MUST be read-only for self-service and editable ONLY via a dedicated HR Compensation endpoint** `[Authorize(Roles="HR")]` with audit reason!
  2. **Bank & Payroll Account:** `AccountTitle`, `AccountNumber`, `BankName`, `BranchCode`, `AccountType` (`Personal` | `Company`), `BankAddress`.
  3. **Family & Dependents:** `FatherName` (req), `FatherNumber`, `FatherCNIC`, `GuardianName`, `GuardianNumber`, `MaritalStatus`. If `Married`, `SpouseName`, `SpouseCNIC`, `SpouseDOB` + dynamic array of Children (`Name`, `CNIC`, `DOB`, `Gender`) with working `+ Add Child` **AND** `Remove Child` controls. Gracefully handles `0` children!
  4. **Secure Document Vault:** Shows all HR-uploaded documents, Contracts, NDAs, Candidate CV, and generated HR Letters (Experience, Bank, Probation, Salary). Only HR/Admin can upload/archive official documents; employees can read/download their own via authorized signed stream URLs.
  5. **Professional Profile & Branded CV Export:** Queries strictly by route `:employeeId` (not caller JWT!). Editable `Qualification` and assigned Project Roles, Responsibilities, & Tech Stack Tools (`Tool` tags), with clean server-side or structured print-PDF Company Resume export.
- **Legacy Issues Resolved:** Closes CRITICAL `MyProfile` `BasicSalary` tampering IDOR and `GetEmployeeProfile` PII leak; fixes `_data.childName.split(',')` null crash; adds missing Remove Child button; fixes Tab 5 ignoring `:Id`; fixes V1 copy-pasted `"Numbers Allowed only"` errors under Name/Email; resolves V1 `UpdateUser` Admin-role overwrite bug.

Module 05: Lifecycle Checklists & E-Sign Resignation (`MOD-05`)

Mockup: `05-lifecycle-checklists-resignation.html`

`FR-LIFE-001` — Onboarding, 90-Day Probation & Offboarding Checklists

- **Trigger:** Auto-seeded on Hire (`Onboarding` + `Probation`), or on Resignation Acceptance (`Offboarding`). Accessible at `/employees/{id}/checklists`.

- **Detailed Flow:** Three tabbed checklists with task owner category (`HR` , `IT` , `Finance` , `Lead`), due date, status (`To Do` | `In Progress` | `Done`), and progress percentage. When all **Probation** tasks reach `Done` , HR sees a 1-click **"Confirm Permanent & Generate End-of-Probation Letter"** action that transitions `EmployeeStatusId: Probation → Permanent` and stores the letter in the Employee's Document Vault.
- **Automated Reminders:** Background scheduled job (protected by internal worker key/managed identity, NOT `[AllowAnonymous]` HTTP GET!) notifies HR/owners of overdue tasks.

`FR-LIFE-002` — Self-Service E-Sign Resignation & HR Clearance

- **Trigger:** Employee opens `/resignation` .
- **Detailed Flow:**
  1. Employee selects `LastWorkingDay ≥ today` (notice-period calculator displays notice duration, e.g., `30 days`), enters `Reason` (min 10 chars), and draws digital signature on HTML5 canvas.
  2. Clicking `Preview Letter` generates PDF using validated `EmployeeResignationLetter.docx` template. Clicking `Submit` creates `EmployeeResignation` (`Status = Pending`) and notifies HR.
  3. **HR Review (`/hr/resignations`):** HR views signed PDF and clicks `Accept` or `Reject` (comment required on reject).
  4. **On Accept:** Sets `ResignationStatus = Accepted` , updates `Employee.ResignationDate` & `EndDate` , seeds **Offboarding Checklist**, and **sends notification & email to the resigning Employee (`Employee.UserId`)** (NOT the HR approver!). On `LastWorkingDay` end-of-day (or immediate termination), background job deactivates `AppUser.IsActive = false` .
- **Legacy Issues Resolved:** Fixes missing `EmployeeResignationLetter.docx` (`FileNotFoundException`), `[max]="todayDate"` preventing future dates, `ChangeResignationLetterStatus` emailing HR instead of the employee, and stateful `AppUserService : EmployeeResignationLetterGenerator` inheritance.

Module 06: Attendance, WFH & Regularization Audit (`MOD-06`)

Mockup: `06-attendance-wfh.html`

`FR-ATT-001` — Daily Check-In / Check-Out & WFH Tracking

- **Trigger:** Employee clicks `Check In` / `Check Out` on Dashboard or Attendance header.
- **Detailed Flow:** Evaluates active open shift (`CheckIn ≠ null && CheckOut == null` for current shift window normalized in company timezone `Asia/Karachi` / UTC). Persists `CheckIn` UTC timestamp, `IsWFH` , client IP, and returns exact active state. Shift durations display formatted (`8h 24m`), never unlabeled raw integer minutes.

`FR-ATT-002` — Missing Check-Out Flagging & Regularization Requests

- **Trigger:** Nightly job detects prior-day shift with `CheckIn ≠ null && CheckOut == null` .
- **Detailed Flow:** Flags the record as `Missing Check-Out` (instead of silently overwriting Check-Out with `14:00:00` as V1 did!) and notifies the employee. Employee submits a **Regularization Request** with actual `CheckOutTime` and `Reason` ; Project Lead or HR approves/rejects it, updating the attendance record and creating an audit log entry.

`FR-ATT-003` — HR/Lead Attendance Management & Immutable Audit Trail

- **Trigger:** Project Lead (for `Team` members only) or HR (`All` employees) opens `/attendance/manage` , edits an existing shift, or adds a missed shift.
- **Detailed Flow:**
  1. Editing an existing record executes `PUT /api/v1/attendance/{attendanceId}` (updating the row **in-place**, never inserting `attendanceId = 0` duplicates!).
  2. Mandatory `Reason` (6–250 chars) is required and persisted alongside `OldCheckIn` , `OldCheckOut` , `NewCheckIn` , `NewCheckOut` , `OldIsWFH` , `NewIsWFH` , `ModifiedByUserId` , and `ModifiedAtUtc` in an immutable `AttendanceAuditLog` table.

3. Excel/CSV exports map `IsWFH` accurately (`Yes` / `No`) and include audit indicators.

- **Legacy Issues Resolved:** Fixes V1 `CheckIn.Date == Now` always-false bug, V1 `RoleId=3` vs `isLead=1` returning 0 employees for Leads, V1 silently discarding `Reason` & `ModifyBy`, V2 `EditAttendance` always setting `attendanceId=0`, V2 inverted `isWFH` in Excel export, and V2 timezone mismatch between server-local `DateTime.Now` and `CurrentDateTimeToPakistan()`.

Module 07: Projects, Teams & Timesheets (`MOD-07`)

Mockup: `07-projects-timesheets.html`

`FR-PRJ-001` — Project, Team & Activity Group Allocation

- **Trigger:** Admin creates/edits project (`/projects`); Admin or Project Lead assigns/unassigns team Activity Groups.
- **Detailed Flow:**
  1. Project fields: `Title`, `ProjectType` (`Billable` | `Non-Billable`), `StartDate`, `EndDate` (**validated** `EndDate ≥ StartDate`), `Description`, `ProjectLeadId` (single active lead), and multi-select `TeamMembers`.
  2. Project Lead (or Admin) can **both Assign AND Unassign** Activity Groups per employee on the project; unassigning preserves historical logged timesheet rows while hiding the activity from future new timecards.
- **Legacy Issues Resolved:** Fixes V1 `dateLessThan('StartDate', 'StartDate')` self-check, V1 `removeEmployee` wrong-index splice bug, V1 `#addpetModal` stray trigger, V1 inability to unassign Activity Groups or add a second assignment without refreshing.

`FR-TS-001` — Weekly & Daily Timesheet Entry, Validation & Re-Submission

- **Trigger:** Any employee opens `/timesheets` (`Admin` and `Executive` are no longer locked out of logging timesheets!).
- **Detailed Flow:**
  1. Employee selects or creates a Monday–Sunday `Timeframe` (calendar week ISO-8601; exactly 7 consecutive days).
  2. Grid renders `Project Title` (real name, never `"aaa"`!) and assigned `Activity Group` rows across `Mon` .. `Sun`.
  3. Each day cell validates `0 ≤ hours ≤ 24`, max `24h` daily aggregate across all projects, with optional step `0.25h` and mandatory task description when `hours > 0`. Clicking a day cell opens the **Daily Task Drawer**; editing a row updates `Mon..Sun` mapped strictly to each day's exact ISO date (`Sunday` maps to `Day + 6`, never Wednesday!).
  4. Employee clicks `Submit Timeframe` (`New` -> `Pending`) or **`Re-Submit Timeframe`** **(`Rejected` -> `Pending`)** while viewing the Lead's Rejection Comment.
- **Legacy Issues Resolved:** Fixes V1 Sunday->Wednesday `wedTaskDate` corruption, V1 hardcoded `<h5>aaa</h5>`, V1 `DateRange` 6-day out-of-bounds bug, V1 blocking re-submission of `Rejected` timecards, V2 `monHourshours` / `tueHourshours` / `wedHourshours` template crash, and restores Timesheet access for Admin/Executive.

`FR-TS-002` — Multi-Level Approval Queue, Automated Reminders & Utilization Reporting

- **Trigger:** Employee submits timecard; Lead reviews in `/timesheets/approvals` (or Executive/HR reviews Lead/HR own timecards).
- **Detailed Flow:**
  1. Team Member timecards route to their `ProjectLeadId` (if multi-project, split approval or primary project lead configurable). Project Lead & HR timecards route to Executive / designated Department Approver (closing the self-approval gap).
  2. Approver clicks `Approve` (`Status = Approved`, locked) or `Reject` (mandatory `RejectionComment`, `Status = Rejected`, editable for re-submission).
  3. **Automated Reminders:** Protected background worker sends individual/BCC reminders (not exposing all employees in `To:`!) on Friday afternoon & Monday morning for unsubmitted weeks.
  4. **Utilization Report:** HR/Executive filters by Date Range, Employee, Project, and Billable Type, drilling into daily entries and exporting `.xlsx`.

Module 08: Unified Employee Requests & HR Approvals Hub (`MOD-08`)

Mockup: `08-requests-approvals-hub.html`

`FR-REQ-001` — Holiday-Aware Leave Requests & Balance Ledger

- **Trigger:** Employee opens `/requests/leaves` ; HR opens `/hr/approvals/leaves` .
- **Detailed Flow:**
  1. Displays real-time Leave Balance card (`Annual Quota` , `Accrued` , `Used` , `Pending` , `Available` — safely defaults to `0` for new employees without crashing!).
  2. Employee selects `LeaveType` (`Annual/Vacation` , `Sick` , `Urgent` , `Short Leave [half-day]` , `Maternity/Paternity` , `Unpaid`), `StartDate` , `EndDate` (`EndDate ≥ StartDate`).
  3. System **automatically computes Working Days** by excluding configured weekends (`Sat` / `Sun`) and **Company Public Holidays** from `MOD-11` (removing manual `TotalDays` tampering!).
  4. Employee can edit/cancel ONLY while `Status == Pending` . Once `Accepted` or `Rejected` , rows are immutable (fixes V2 overwrite bug) and Rejection `Comment` is prominently displayed to the employee.
  5. **On HR** `Accept` **:** Backend transaction checks sufficient balance (unless `Unpaid`), prevents double-approval race conditions, records `LeaveLedger` debit (`-hours`), and marks leave dates on Timesheet/Attendance summary.
- **Legacy Issues Resolved:** Fixes V1 leaves never deducting from balance, V2 `data[0]` crash for new hires, V2 manual `TotalDays` ignoring weekends/holidays, V2 allowing edits to already Accepted/Rejected leaves, and V1 hiding Rejection comments.

`FR-REQ-002` — HR Letters & Certificates (Self-Service + Proactive HR Issuance)

- **Trigger:** Employee requests letter OR HR generates letter proactively (`/requests/letters` or `/hr/letters`).
- **Supported Letter Types:** `Bank Account Opening Letter` , `End of Probation Letter` , `Experience / Service Certificate` (restored from V1!), `Employment & Salary Verification Letter` .
- **Detailed Flow:** Employee selects type & purpose (`Reason`). Upon HR approval (or direct HR issuance), system renders PDF from validated `.docx` template, saves into Employee's **Document Vault** (`MOD-04`), and enables instant download.

`FR-REQ-003` — Expense & Reimbursement Claims

- **Trigger:** Employee submits `Claim` or `Expense` with `Target Month-Year` , `Amount (PKR > 0)` , `Category` , `Description` , and Receipt upload (PDF/JPG/PNG <= 10 MB).
- **Detailed Flow:** HR Approval queue displays **`Employee`** , **`Type`** , **`Target Month-Year`** , **`Amount (PKR)`** , **`Description`** , **`Receipt Preview`** , and `Accept` / `Reject` with mandatory comment. Export generates `.xlsx` named `Expense_Claims_Report.xlsx` (not `"Item Request History.xlsx"` !).
- **Legacy Issues Resolved:** Fixes V2 omitting `Amount` and `Target Month-Year` in HR Claim Approval table and fixes misnamed Excel export.

`FR-REQ-004` — IT & Office Equipment / Item Requests

- **Trigger:** Employee requests `New Equipment` or `Replacement` from IT/Office Catalog (`Setup → Items`) or adds custom `"Other"` item specification.
- **Detailed Flow:** Tab switching between `New Request` and `History` retains focus cleanly (fixing V2 `resetRequestForm` forcing `#requestHistory`). Submitting dispatches notification + email to configured IT/Admin/HR recipients (fixing V2 empty `To` list in `SendItemRequestEmail`).

Module 09: Monthly Payroll, Payslips & Discrepancy Claims (`MOD-09` — Restored from V1)

Mockup: `09-payroll-payslips.html`

`FR-PAY-001` — Holiday-Aware Monthly Payroll Preparation (HR)

- **Trigger:** HR opens `/payroll/run`, selects `PayPeriod` (any `Month + Year` — editable, fixing V1 `readonly` inputs!), and clicks `Generate Draft Payroll`.
- **Calculation Business Rules:**
  1. `WorkingDays = countBusinessDays(Month) - countPublicHolidays(Month)`; `ExpectedHours = WorkingDays * 8`.
  2. `ApprovedHoursWorked` = sum of `Approved` timesheet hours in `PayPeriod` (excluding overtime beyond `ExpectedHours`, or separating overtime per Product Policy §7).
  3. `ApprovedPaidLeaveHours` = sum of `Accepted` paid leave hours in `PayPeriod` (from `MOD-08`, capped at `Shortfall = max(0, ExpectedHours - ApprovedHoursWorked)` — **strictly** `≥ 0`, fixing V1 negative `PaidLeaves` on overtime!).
  4. `HourlyRate = BasicSalary / ExpectedHours`; `BaseEarned = min(BasicSalary, (ApprovedHoursWorked + ApprovedPaidLeaveHours) * HourlyRate)`.
  5. `GrossIncome = BaseEarned + TravelAllowance + Bonus`; `NetSalary = max(0, GrossIncome - TaxDeduction - OtherDeductions)`.
- **Idempotent Adjustments:** HR edits `TravelAllowance`, `Bonus`, `TaxDeduction`, and `DeductionReason` in modal; backend recalculates `GrossIncome` and `NetSalary` from immutable `BaseEarned` (fixing V1 cumulative `GrossIncome += Bonus` inflation and stale `NetSalary`!).

`FR-PAY-002` — Executive Payroll Batch Approval

- **Trigger:** HR submits `Draft` (`New`) payroll batch -> `Pending Executive Approval`; Executive (`[Authorize(Roles="Executive")]`) reviews `/payroll/approvals/{yearMonth}`.
- **Detailed Flow:** Displays total batch spend, employee count, MoM variance highlights, and per-employee breakdown. Executive clicks `Approve & Lock Batch` (locks month from further edits and makes payslips available) or `Reject Batch` (requires comment, returns to HR `Draft`).
- **Legacy Issues Resolved:** Secures V1 unauthenticated `AdminApproval` endpoint and displays `Rejected` status properly.

`FR-PAY-003` — Employee Payslip Vault & Closed-Loop Discrepancy Claims

- **Trigger:** Approved payroll unlocks Payslips under `/payroll/my-payslips`.
- **Detailed Flow:**
  1. Employee views/downloads formatted PDF Payslip for any approved historical month (background PDF service generates isolated memory stream by `PayRollId` — **zero shared** `myWelcome.pdf` **file, zero** `DateTime.Now.Month` **mismatch, zero** `Console.ReadKey()` **!**).
  2. If employee spots an error, they click `Raise Discrepancy Claim` (`Category`, `Description`, optional proof).
  3. HR views open claims in `/payroll/claims`, investigates, and resolves with status `Resolved (Adjusted in Next Cycle / Re-issued)` or `Declined` + `ResolutionNotes`, notifying the employee.

Module 10: Knowledge Base, Innovation Ideas & Company Blog (`MOD-10`)

Mockup: `10-knowledge-ideas-blogs.html`

`FR-KBI-001` — Versioned Policies & Benefits + Read-Acknowledgment Tracking

- **Trigger:** HR uploads/updates a Policy or Benefit (`/knowledge/policies`, `/knowledge/benefits`); Employees browse and acknowledge.
- **Detailed Flow:**
  1. Flexible alphanumeric validation (`3-120 chars`, allows numbers, colons, hyphens — removing V1/V2 `/^[a-zA-Z]*$/`!).

2. Updating metadata (`Title` , `Description` , `Category`) **without uploading a new file preserves the existing attachment** (fixing V1 file wipe!). Uploading a new file increments `Version` (`v1.0 → v2.0`), marks prior version `IsLatest = false` (preserving Version History drawer), and optionally flags `RequiresReAcknowledgment = true` .
3. Employees view **full readable** `Description` / `Abstract` inline, preview PDFs, and click `Acknowledge Policy` (recording `UserId` , `PolicyId` , `Version` , `Timestamp` for HR compliance reporting).

`FR-KBI-002` — Innovation Idea Board

- **Trigger:** Any employee submits an idea (`Title` , full `Abstract` , optional attachment) on `/knowledge/ideas` .
- **Detailed Flow:** Displays full Idea cards (never hiding `Abstract` !), employee upvote counter (`+1`), and status tags (`Submitted` , `Under Review` , `Planned` , `Implemented` editable by HR/Exec). **Strict Creator Ownership:** Edit & Delete controls appear and succeed ONLY for `CreatedBy == CurrentUser.UserId` (or HR/Admin archive), fixing V2 allowing any employee to edit/delete colleagues' ideas.

`FR-KBI-003` — Blog Authoring, Editorial Review & Published Reader Feed

- **Trigger:** Employee drafts/submits blog (or completes HR-assigned topic); HR approves; **All employees read on** `/knowledge/blog` .
- **Detailed Flow:**
  1. Author composes Title, Cover/Tags, sanitized rich-text Content (`Save Draft` | `Submit for Review`). Authors may edit/delete `Draft` (`Saved`) or `Pending` submissions cleanly (fixing V2 contradictory `*ngIf` !).
  2. HR reviews in Editorial Queue (`Approve & Publish` | `On Hold / Request Edits` | `Reject` with comment).
  3. **Published Feed:** Approved blogs (`Status == Published`) appear on a clean Company Blog Reader Feed with author, publish date, reading time, and search!

Module 11: System Master Data, Holiday Calendar & Templates (`MOD-11`)

Mockup: `11-system-setup-master-data.html`

`FR-CFG-001` — Unified Master Data Setup

- **Trigger:** Admin or HR opens `/setup` .
- **Detailed Flow:** Dedicated, cleanly typed CRUD pages/tabs for `Departments & Job Roles` , `Evaluation Subjects` , `Activity Groups` , `Tech Stack Tools` , and `Equipment/Item Catalog` —each backed by its own DTO and route `:id` (explicitly fixing V2 `AddItemComponent` reading `ToolData` !).

`FR-CFG-002` — Company Public Holiday Calendar & Workweek Rules

- **Trigger:** HR/Admin adds or imports annual Public Holidays (`/setup/holidays`).
- **Detailed Flow:** Stores `Date` , `HolidayName` (e.g., *Quaid-e-Azam Day*), `IsRecurring` , `AppliesToLocations` , plus Workweek (`Mon–Fri` , `8h/day`) and Leave Accrual policy (`18 days/year`). Consumed automatically by **MOD-07 (Timesheet targets)**, **MOD-08 (Leave working-day calculator)**, and **MOD-09 (Payroll expected hours)**.

`FR-CFG-003` — Document Template Manager & Placeholder Validator

- **Trigger:** Admin/HR views `/setup/templates` or uploads a `.docx` template.
- **Detailed Flow:** Displays all required templates (`EmploymentContract.docx` , `EmploymentNDA.docx` , `BankOpeningLetter.docx` , `EndOfProbationLetter.docx` , `ExperienceLetter.docx` , `EmployeeResignationLetter.docx` , `PaySlip.docx`), parses XML placeholders (`<EmployeeName>` , `<JoiningDate>` , etc.), and verifies template health at startup and upload time so missing templates fail fast in Setup rather than crashing user workflows!

---

4. User Stories & Gherkin Acceptance Criteria

GHERKIN

```
Feature: Zero-Trust Entra ID Authentication & Deactivation Enforcement
  Scenario: Valid active employee signs in with Microsoft Entra ID
    Given an employee "ayesha.khan@elikser.com" exists with IsActive = true and Status = Permanent
    When she authenticates with Microsoft Entra ID and submits a cryptographically valid idToken
    Then the backend validates the JWT signature against Microsoft OIDC JWKS, issuer, and audience
    And returns a signed portal JWT containing her employeeId, roles, permissions, and isLead flag
    And does not expose any token in query string parameters.

  Scenario: Forged or unsigned JWT is rejected
    Given an attacker constructs an unsigned JWT with preferred_username "admin@elikser.com"
    When the token is sent to POST /api/v1/auth/sso/entra
    Then the backend rejects the request with HTTP 401 Unauthorized
    And logs a security audit warning without issuing a portal token.

  Scenario: Deactivated or resigned user is blocked from login
    Given an employee whose AppUser.IsActive = false or EmployeeStatusId = Resigned
    When they attempt SSO login
    Then the system denies access with HTTP 403 Forbidden and message "Account deactivated. Contact HR."
```

GHERKIN

```
Feature: Profile Field-Level Authorization (Prevent Salary Tampering IDOR)
  Scenario: Team member cannot alter BasicSalary via profile update
    Given "bilal.ahmed@elikser.com" is logged in as TeamMember (RoleId = 4)
    When he sends PUT /api/v1/employees/me/profile with a modified basicSalary = 950000
    Then the backend ignores or rejects the restricted compensation field with HTTP 403 Forbidden
    And his BasicSalary in the database remains unchanged.

  Scenario: Profile loads cleanly for an employee with zero children
    Given an employee "usman.tariq@elikser.com" has MaritalStatus = Single and zero Child records
    When HR or Usman opens the Family & Dependents tab
    Then the tab renders cleanly with an empty children list and an active "+ Add Child" button
    And throws zero JavaScript TypeErrors.
```

GHERKIN

```
Feature: End-to-End E-Sign Resignation & Offboarding Trigger
  Scenario: Employee submits resignation with future notice-period Last Working Day
    Given today is "2026-11-19"
    When an employee selects Last Working Day "2026-12-19" (30 days ahead), enters an exit reason, and signs
the canvas
    And clicks "Preview Resignation Letter"
    Then the system renders the PDF from validated template "EmployeeResignationLetter.docx" without error
    And on Submit saves Status = "Pending" and notifies HR.

  Scenario: HR accepts resignation and the resigning employee receives notification
    Given HR views Usman Tariq's pending resignation
    When HR clicks "Accept Resignation"
    Then the system updates Status = "Accepted", seeds Usman's Offboarding Checklist
    And dispatches the acceptance notification and email to "usman.tariq@elikser.com" (NOT HR).
```

GHERKIN

```
Feature: Attendance Modification & Immutable Audit Trail
  Scenario: HR edits an existing attendance shift in-place
    Given AttendanceId = 4082 has CheckIn "09:45 AM" and CheckOut "06:10 PM"
    When HR updates CheckIn to "09:00 AM", toggles IsWFH = true, enters Reason "Approved early client call",
and clicks Save
    Then the system updates AttendanceId = 4082 in-place via PUT /api/v1/attendance/4082 (no duplicate row
created)
    And appends an immutable row in AttendanceAuditLog recording old/new timestamps, WFH state, HR's UserId,
and the Reason.
```

```gherkin
Feature: Weekly Timesheet Date Integrity & Rejection Re-Submission
  Scenario: Sunday hours persist strictly to Sunday (never Wednesday)
    Given an employee opens timeframe "2026-11-16 to 2026-11-22"
    When the employee logs 4.0 hours on Sunday "2026-11-22" under "FinTech Core API / Development" and saves
    Then the persisted TimesheetDetail record has TaskDate = "2026-11-22"
    And Wednesday "2026-11-18" hours remain unaffected.

  Scenario: Employee edits and re-submits a rejected timesheet
    Given a timesheet is in Status = "Rejected" with Lead comment "Break out QA hours"
    When the employee updates hours and clicks "Re-Submit for Approval"
    Then Status transitions to "Pending" and the Lead receives a re-submission notification.
```

```gherkin
Feature: Holiday-Aware Leave Calculation & Expense Claim Visibility
  Scenario: Leave duration excludes weekends and public holidays automatically
    Given Friday "2026-12-25" is configured as a Public Holiday in Setup
    When an employee selects StartDate = "2026-12-24" (Thu) and EndDate = "2026-12-28" (Mon)
    Then the system auto-calculates TotalWorkingDays = 2.0 days (Thu + Mon, excluding Fri holiday and Sat/Sun)
    And blocks manual override of working days.

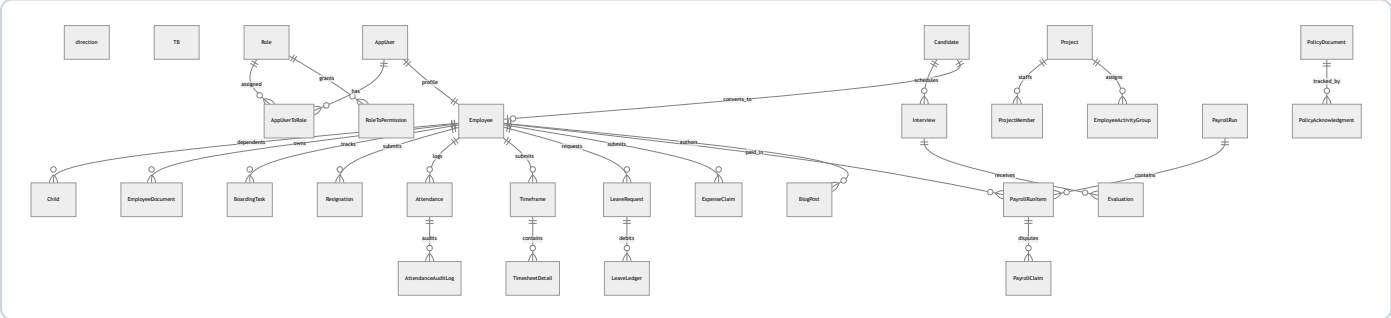
  Scenario: HR inspects Claim Amount and Target Month before approving reimbursement
    Given an employee submitted an Expense Claim for "October 2026" of "PKR 12,500" with a PDF receipt
    When HR opens the Pending Claims Approval table
    Then the table clearly displays "PKR 12,500", "October 2026", Description, and a working Receipt Preview
button.
```

```gherkin
Feature: Non-Destructive Monthly Payroll Run & Claim Resolution
  Scenario: Repeated HR bonus adjustments do not cumulatively inflate GrossIncome
    Given Employee BasicSalary earned for Oct 2026 is "PKR 180,000"
    When HR sets TravelAllowance = 10,000 and Bonus = 15,000 and saves (Gross = 205,000)
    And later edits Bonus to 20,000 and saves again
    Then GrossIncome equals 210,000 (180,000 + 10,000 + 20,000), NOT 225,000
    And NetSalary updates synchronously in the database.
```

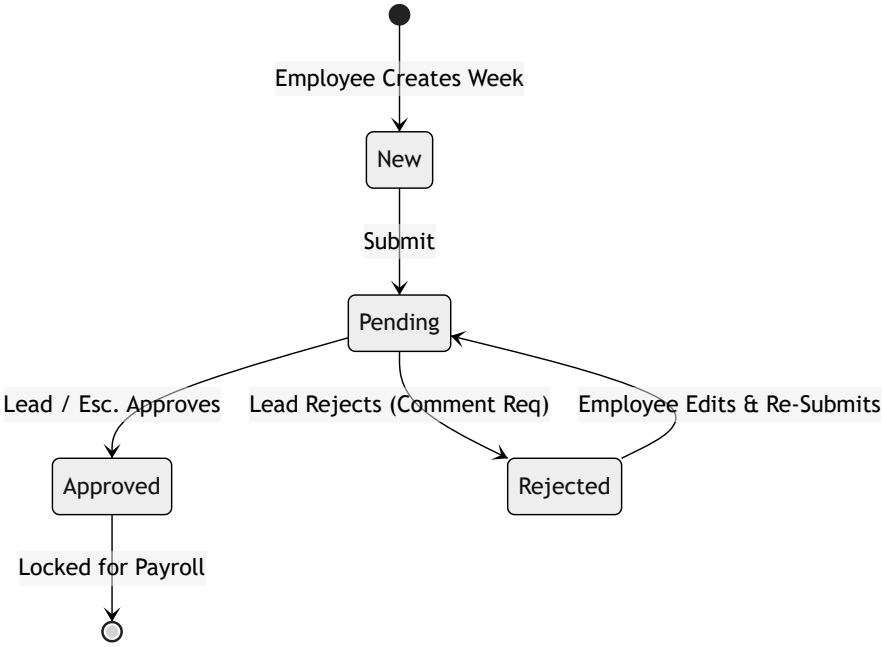
---

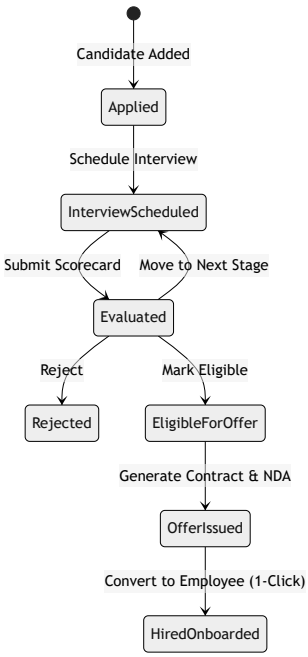
5. Technical Architecture, Data Model & API Contracts

5.1 Core Domain Entity Relationship Diagram



5.2 Standardized Lifecycle State Machines





5.3 Standardized Enumerations (Replacing Mixed Int/String Casing)

- **`ApprovalStatus`**: `Draft (0)`, `Pending (1)`, `Approved (2)`, `Rejected (3)`, `Cancelled (4)`, `OnHold (5)` — applied uniformly across Timesheets, Leaves, Documents, Claims, Items, Resignations, and Payroll Batches.
- **`EmployeeStatus`**: `Probation (1)`, `Permanent (2)`, `NoticePeriod (3)`, `Resigned (4)`, `Terminated (5)`.
- **`ClaimResolutionStatus`**: `Open (1)`, `UnderReview (2)`, `ResolvedAdjusted (3)`, `Declined (4)`.

5.4 RESTful API Contract Highlights (`/api/v1/ ...`)

All endpoints use proper HTTP verbs (`GET`, `POST`, `PUT`, `PATCH`, `DELETE`), return standard JSON envelope `{ "data": ... , "meta": ... , "errors": [] }`, enforce `[Authorize(Policy = " ... ")]`, and separate metadata JSON from file uploads (`multipart/form-data` -> Object Storage + DB URI/Key).

Endpoint	Auth Policy	Key Contract & Validation Rules
`POST /api/v1/auth/sso/entra`	`[AllowAnonymous]`	Body: `{ idToken }`. Validates JWKS/iss/aud; blocks inactive/resigned users (`403`).
`PUT /api/v1/employees/me/profile`	`Employee.Self`	Updates personal/address fields; **rejects** `basicSalary` , `joiningDate` , `statusId` (`403`).
`PATCH /api/v1/employees/{id}/compensation`	`HR.ManageCompensation`	Updates `basicSalary` , `annualLeaves` , `jobPositionId` ; logs Compensation Audit row.
`POST /api/v1/candidates/{id}/convert-to-employee`	`HR.Hire`	Atomic DB transaction creating `AppUser` , `Employee` , Vault Docs, Onboarding Tasks, Leave Quota.
`PUT /api/v1/attendance/{id}`	`Attendance.Manage`	Updates existing shift; requires `reason` (6–250 chars); writes `AttendanceAuditLog` .
`POST /api/v1/timeframes/{id}/resubmit`	`Timesheet.Self`	Allowed when `status IN (Draft, Rejected)` ; validates `0..24h/day` .
`POST /api/v1/requests/leaves/preview-days`	`Employee.Self`	Computes working days from `startDate..endDate` excluding weekends + `PublicHolidays` .
`POST /api/v1/payroll/runs/{yearMonth}/generate`	`Payroll.Prepare`	Computes non-negative `PaidLeaveHours` , `BaseEarned` , `GrossIncome` , `NetSalary` .
`POST /api/v1/payroll/claims/{id}/resolve`	`Payroll.Prepare`	Updates claim status (`ResolvedAdjusted` / `Declined`), stores `resolutionNotes` , notifies employee.
`POST /api/v1/files/upload`	`Authenticated`	Multipart upload (`≤ 10 MB` , magic-byte PDF/DOCX/PNG/JPG check), returns `fileId` .
`GET /api/v1/files/{fileId}`	`File.AuthorizedRead`	Checks resource ownership or role permission; streams binary with real MIME `Content-Type` .

6. Non-Functional, Security, & Audit Requirements

1. **Zero-Trust Security & Secret Hygiene:**
   - All JWT, Azure AD, SMTP, Database, Google Calendar, and PDF rendering credentials MUST reside in Azure Key Vault / environment variables—never in `appsettings.json` or repo files.
   - Strict origin-allowlist CORS (never `AllowAnyOrigin()` + `AllowCredentials()`).
   - Scheduled reminder workers authenticate with an internal Managed Identity / API Key (`POST /api/v1/jobs/*`), replacing all `[AllowAnonymous]` GET email endpoints.
   - All Rich Text (Blogs, Notifications) sanitized with HTML sanitizer (DOMPurify on client + HtmlSanitizer on server) to prevent Stored XSS.
2. **Performance & Scalability:**
   - List APIs (`GET /candidates` , `/employees` , `/policies` , `/ideas` , `/payslips`) MUST use server-side pagination (`page` , `pageSize` , `sort` , `search`) and NEVER include raw file bytes (`VARBINARY(MAX)` / Base64 strings).
   - p95 API latency `< 250 ms` for standard reads/writes; `< 2.0 s` for on-demand PDF rendering.
   - Single `DbContext` lifetime per HTTP request (`Scoped`), eliminating V1's dual undisposed `UnitOfWork` context leak.
3. **Reliability & Concurrency:**

- PDF rendering executes purely in-memory (`MemoryStream`), eliminating V1's shared `"myWelcome.pdf"` race condition and lethal `Console.ReadKey()` / `Environment.Exit(2)` calls.
- Email failures log warnings and enqueue retries without failing the primary DB transaction.

4. **Accessibility & Responsiveness:**
   - Full keyboard navigation, focus-visible outlines, ARIA labels on modals/drawers, WCAG 2.1 AA contrast (`≥ 4.5:1`), responsive layouts down to `390px` mobile width.
5. **Immutable Audit Logging:**
   - Mandatory audit trail rows (`ActorUserId` , `TimestampUtc` , `EntityType` , `EntityId` , `Action` , `BeforeJson` , `AfterJson` , `Reason` , `IpAddress`) for: Auth Failures, Role/Permission Changes, Salary Edits, Attendance Edits, Leave/Payroll Approvals, and Resignation Decisions.

---

7. Open Product Questions for Stakeholder Sign-Off

#	Question	Options / Recommendation	Impacted Module
Q1	How should **Overtime Hours** (`HoursWorked > ExpectedHours`) affect Monthly Payroll?	**(Rec)** Cap `BaseEarned` at `BasicSalary` and track overtime hours separately for optional HR Overtime Bonus approval (never negative `PaidLeaves`).	`MOD-09` Payroll
Q2	Who approves Timesheets & Leave Requests submitted by **Project Leads** and **HR Managers**?	**(Rec)** Route Project Lead & HR requests to their Department Head / Executive Approver (`ApproverUserId` on `Employee`).	`MOD-07` , `MOD-08`
Q3	What is the Annual Leave **Carry-Forward & Encashment** policy at year-end?	Define annual cap (e.g., max 5 days carry-forward, zero automatic encashment, or reset on Jan 1).	`MOD-08` , `MOD-11`
Q4	When a Company Policy receives a **Major Version** update, should all employees be forced to re-acknowledge on next login?	**(Rec)** Yes for Major version bumps (`v2.0`), optional banner for Minor edits (`v1.1`).	`MOD-01` , `MOD-10`
Q5	Should **Contracts & NDAs** support digital candidate e-signature acceptance inside the portal before Hire Conversion?	**(Rec)** Phase 2 enhancement: email secure time-limited signing link; Phase 1 retains HR PDF generation + Email.	`MOD-03` ATS
Q6	How long should an employee have to submit a **Missing Check-Out Regularization** request before escalation?	**(Rec)** 48 business hours self-service window; escalates to Lead/HR after 48 hours.	`MOD-06` Attendance
Q7	What is the exact **Standard Workweek & Public Holiday** calendar basis?	Configure Monday–Friday (8h/day, 40h/week) in `MOD-11` vs Saturday half-day if applicable per department.	`MOD-07` , `MOD-08` , `MOD-09` , `MOD-11`
Q8	Should **Payslip Tax Deductions** be auto-calculated via configurable annual FBR income tax slabs or imported/entered manually by Finance?	**(Rec)** Support configurable tax slab formula with Finance override per employee.	`MOD-09` Payroll