ELIKSER PEOPLEOPS V3 · MODERNIZATION DOCUMENTATION

Executive Modernization & Architecture Proposal

Elikser Employee Portal V3 · Management Review & Sign-Off Baseline · November 20, 2026

Elikser Employee Portal V3 — Executive Modernization & Architecture Proposal

Prepared For: Elikser Executive Management & Engineering Leadership
Prepared By: Bilal Azam (`bilal.azam@elikser.com`)
Date: November 20, 2026
Version: 1.0 (Management Review & Sign-Off Baseline)
Live Interactive Prototype: https://mockups.employee.elikser.com
Companion Specifications:

- Business & Functional PRD (for Product Team): `PRD_Employee_Portal_Modernization.md`
- Verified V3 Azure SQL Schema: `sql/V3_Database_Schema.sql`
- QA Playbook & Regression Matrix: `QA_How_To_And_Testing_Guide.md`

---

1. Executive Summary

Elikser's internal Employee Portal currently exists as **four fragmented legacy projects** split across two unfinished generations (**V1** on .NET Core 3.1 + Angular 8, and **V2** on .NET + Angular 8). A comprehensive technical and product audit uncovered severe security vulnerabilities, broken cross-department workflows, and lost business modules (V2 dropped V1's Monthly Payroll, Payslip Vault, and Experience Letters).

Rather than patching brittle legacy Angular 8 templates and a 45-table blob-bloated database, this proposal establishes **Employee Portal V3**—a clean, secure, single-monorepo platform aligned 1-for-1 with the cloud governance, DevOps pipeline standards, and low-cost Azure footprint proven on **Elikser CustomerPortal**.

Strategic Value for Management

1. **Eliminates Critical Security & Payroll Risks:** Closes unverified JWT authentication bypasses, self-service salary tampering (`BasicSalary` IDOR), unauthenticated PII/document downloads, and cumulative payroll calculation bugs.
2. **Unifies All 11 HR, Delivery & Finance Workflows:** Connects Recruitment (ATS) -> 1-Click Onboarding -> Probation Confirmation -> Attendance/WFH -> Project Timesheets -> Holiday-Aware Leave & Claims -> Monthly Payroll & Executive Batch Approval -> E-Signed Resignation & Offboarding in one platform.
3. **Standardized Low-Cost Azure Footprint:** Reuses Elikser's proven Azure architecture (Azure Static Web Apps + Azure Container Apps Consumption + Azure SQL + Azure Blob Storage + Azure Managed SSL) with automated `Develop` -> `QA` -> `Prod` CI/CD gates.

---

2. Legacy Assessment: Why a Clean Rebuild Is Required

Risk Category	Findings Across Legacy V1 & V2	Business Impact if Unaddressed
Authentication & RBAC (Critical)	Backend parsed SSO JWTs without validating Microsoft cryptographic signatures; API controllers lacked role/ownership checks; any authenticated user could modify `BasicSalary` or read colleague CNIC/bank records.	Full account takeover risk and unauthorized access to confidential compensation & PII data.

Risk Category	Findings Across Legacy V1 & V2	Business Impact if Unaddressed
Dropped & Disconnected Modules	V2 added ATS, Checklists, and Resignation, but **dropped Monthly Payroll, Payslips, Salary Discrepancy Claims, and Experience Letters** from V1. No Company Public Holiday calendar existed in either version.	HR and Finance forced onto manual spreadsheets; inaccurate leave and working-hour deductions.
High Defect Density in Core Flows	V1 Sunday timesheets overwrote Wednesday; V2 `EditAttendance` inserted duplicate shifts; missing `EmployeeResignationLetter.docx` crashed exit flows; Resignation form blocked future notice dates (`max=today`); HR Claim table hid claim `Amount` and `Month` .	Operational friction, payroll disputes, and recurring developer firefighting.
Database & Storage Anti-Patterns	45+ redundant tables storing multi-MB CVs, Contracts, Policies, and Payslips as inline Base64 `NVARCHAR(MAX)` / `VARBINARY(MAX)` blobs; 5 MB browser `localStorage` upload crashes.	Slow list queries, bloated SQL storage/backup costs, and broken document uploads.
Repository & Secret Sprawl	4 separate folders with legacy TFVC bindings (`.tfignore` , `.vspscc`), `EmployeePortal.kdbx` , and hardcoded SQL, SMTP, Entra, and Google ServiceAccount secrets in source files.	Credential exposure risk and inability to run clean automated CI/CD gates.

3. Approved Target Technology Stack

Aligned with Elikser engineering standards and `CustomerPortal` operational tooling:

Layer	Approved V3 Technology	Rationale
Frontend Web (`web/`)	**Angular LTS** (Standalone Components, Signals, Angular CDK, Tailwind CSS, `@azure/msal-browser` , esbuild/Vite builder)	Preserves team Angular domain expertise while eliminating all deprecated Angular 8 / jQuery / Bootstrap 4 technical debt.
Backend API (`api/`)	**.NET 10 Web API** + EF Core 10 + SignalR (`/hubs/notifications`)	High-performance, strongly-typed REST API with policy-based RBAC, real-time push alerts, and `xUnit` security/unit test gates (`api.tests/`).
Identity & SSO	**Microsoft Entra ID** (Elikser Corporate Tenant `719456a1-069e-4f5a-a577-85431acfb7aa`)	Single corporate SSO (`*@elikser.com`) with strict server-side OIDC JWKS signature, issuer, audience, and active-employee verification.
Relational Database	**Azure SQL Database** (`sqldb-employee-{dev,qa,prod}`)	Clean 3NF schema across 9 logical schemas (`core` , `cfg` , `hr` , `ats` , `work` , `req` , `pay` , `hub` , `audit`) with engine-enforced financial invariants.
Document & File Vault	**Azure Blob Storage** (`stemployee{dev,qa,prod}` / container `portal-files`)	Replaces all inline SQL `VARBINARY` /Base64 columns; supports streaming uploads, SHA-256 integrity hashes, and authorized downloads.
Document Rendering	In-Memory OpenXML / Templated PDF Engine	Thread-safe in-memory generation for Contracts, NDAs, HR Letters, Resignations, and Payslips with startup `.docx` placeholder validation.

4. Standardized Environments, URLs & Azure Resource Matrix

Follows the exact DNS (`[service.]<app>.[<env>.]elikser.com`) and Azure resource naming standard established by `CustomerPortal` , secured with **Azure Managed SSL Certificates**:

Environment	Git Branch	Azure Resource Group	Portal Web URL (Angular LTS)	Backend API URL (.NET 10)	Azure SQL Database	Azure Blob Storage
Mockups *(Review)*	Static	`employee-portal-dev`	`https://mockups.employee.elikser.com`	*(Interactive Stateful Prototype)*	N/A	N/A

Environment	Git Branch	Azure Resource Group	Portal Web URL (Angular LTS)	Backend API URL (.NET 10)	Azure SQL Database	Azure Blob Storage
DEV	`Develop`	`employee-portal-dev`	`https://employee.dev.elikser.com` (`swa-employee-dev`)	`https://service.employee.dev.elikser.com` (`service-employee-dev`)	`sqldb-employee-dev`	`stemployeedev`
QA	`QA`	`employee-portal-qa`	`https://employee.qa.elikser.com` (`swa-employee-qa`)	`https://service.employee.qa.elikser.com` (`service-employee-qa`)	`sqldb-employee-qa`	`stemployeeqa`
PROD	`Prod`	`employee-portal-prod`	`https://employee.elikser.com` (`swa-employee-prod`)	`https://service.employee.elikser.com` (`service-employee-prod`)	`sqldb-employee-prod`	`stemployeeprod`

---

5. Repository Governance, Archival & Secret Hygiene Plan

To prevent carrying historical plaintext secrets and TFVC clutter into the new platform:

1. **Create One Clean Active Git Monorepo (`EmployeePortal`):**
   - Hosted in Azure DevOps (`https://dev.azure.com/elikser/Employee%20Portal/_git/EmployeePortal`).
   - Standardized layout: `web/` (Angular LTS), `api/` (.NET 10), `api.tests/` (xUnit), `docs/`, `pipelines/` (path-filtered Web & Service YAML pipelines), and `scripts/`.
   - Protected branch promotion flow: `feature/<id>-<slug>` / `bugfix/<id>-<slug>` -> PR to **`Develop`** -> PR to **`QA`** -> Approval-gated PR to **`Prod`**.
2. **Archive Legacy Repositories Read-Only:**
   - Snapshot V1 into **`EmployeePortal-Legacy-V1`** (`v1-archive` tag) and V2 into **`EmployeePortal-Legacy-V2`** (`v2-archive` tag), then mark both **Disabled / Read-Only** in Azure DevOps Project Settings (and lock `$/Employee Portal/_Archive` if TFVC is present).
3. **Credential Rotation & Key Vault Isolation:**
   - Rotate all legacy exposed SQL, SMTP, Entra, and Google Calendar credentials; store runtime secrets exclusively in Azure Container App Secrets / Key Vault (`kv-employee-<env>`).

---

6. Re-Designed V3 Database Architecture Summary

The legacy 45+ fragmented tables have been redesigned into **9 clean 3NF domain schemas** (verified against SQL Server with `0` errors in `V3_Database_Schema.sql`):

Schema	Responsibility	Key Architectural Improvements
`core`	Users, Roles, Granular Permissions, File Pointers, Notifications	Replaces hardcoded `FeatureId 1..24` and consolidates 10+ Base64/blob columns into `core.Files` -> Azure Blob.
`cfg`	Departments, Job Positions, Unified Catalogs, **Public Holidays**, `.docx` Templates	Adds company holiday calendar consumed automatically by Leave & Payroll; validates `.docx` templates upfront.
`hr`	Employees, Bank Accounts, Family/Emergency Contacts, **Compensation History**, Checklists, Resignations	Isolates salary into `hr.EmployeeCompensations` (blocking profile IDOR); merges 3 duplicate checklist tables into `hr.LifecycleTasks`.
`ats`	Candidates, Interviews, Panelists, Star Scorecards, Question Bank	Links `ConvertedEmployeeId` for atomic 1-Click Candidate-to-Employee onboarding.
`work`	Attendance Shifts, Regularizations, Projects, Members, Activities, Weekly/Daily Timesheets	Enforces ISO Monday weeks, `0..24h` cell bounds, unique `(Week, Activity, Date)` cells, and non-destructive activity unassignment.

Schema	Responsibility	Key Architectural Improvements
`req`	Leave Requests, **Append-Only Leave Ledger**, Document Requests, Expense Claims, Equipment	Filtered unique index guarantees single leave-balance debit per request; Claims enforce positive `Amount` & receipt pointer.
`pay`	Monthly Payroll Runs, Payroll Items, Discrepancy Disputes	`PERSISTED` **computed** `GrossIncome` **&** `NetSalary` **columns** + `PaidLeaveHours ≥ 0` make double-addition and negative-leave bugs impossible.
`hub`	Versioned Policies/Benefits, Read-Acknowledgments, Upvotable Ideas, **Published Blog Feed**	Tracks mandatory policy sign-offs per version and publishes approved employee blogs to a company reader feed.
`audit`	Append-Only Security, Compensation & Attendance Audit Trail	Database `CHECK` constraint enforces a mandatory reason (`≥ 6 chars`) on every attendance edit and salary change.

7. Functional Scope & Phased Delivery Roadmap

All 11 modules (`MOD-01` through `MOD-11`) are interactively prototyped at `https://mockups.employee.elikser.com` :

Phase	Target Modules	Key Milestones & Exit Criteria
Phase 1: Foundation, Identity & Core HR	`MOD-01` Auth/RBAC, `MOD-04` Directory & 360° Profile, `MOD-11` Master Setup & Holidays	Clean `EmployeePortal` monorepo + CI/CD live on `DEV` / `QA` ; Entra JWKS auth & policy RBAC active; `sqldb-employee-*` & Azure Blob vault deployed.
Phase 2: Daily Operations & Self-Service	`MOD-02` Dashboard & SignalR Alerts, `MOD-06` Attendance/WFH, `MOD-07` Projects & Timesheets, `MOD-08` Unified Requests	Live check-in/out + regularization audit, Mon–Sun/Daily timesheets with Lead/Exec approvals, holiday-aware Leave Ledger, Letters, Claims & Equipment.
Phase 3: Talent Lifecycle & Knowledge Hub	`MOD-03` Recruitment (ATS), `MOD-05` Checklists & E-Sign Resignation, `MOD-10` Policies, Ideas & Company Blog	End-to-end ATS pipeline -> Offer/NDA -> 1-Click Hire, Onboarding/90-Day Probation/Offboarding checklists, E-Sign exit flow, compliance policy acks & blog feed.
Phase 4: Payroll Engine, Data Migration & Cutover	`MOD-09` Monthly Payroll, Executive Batch Approval, Payslip Vault & Claims + Legacy Cutover	Reconciled payroll run parallel-tested against legacy cycle, historical employee/leave/document migration verified, legacy repos locked, `PROD` cutover.

8. Management Decisions & Sign-Off Summary

Decision Item	Proposed / Confirmed Direction	Status
Environment URL Standard	`employee.{dev.,qa.,}elikser.com` + `service.employee.{dev.,qa.,}elikser.com` + `mockups.employee.elikser.com`	**Confirmed**
Frontend Framework	Modern **Angular LTS** (`web/` inside single `EmployeePortal` monorepo)	**Confirmed**
Backend & Cloud Data Stack	**.NET 10 Web API** + **Azure SQL** (`sqldb-employee-<env>`) + **Azure Blob Storage** (`stemployee<env>`)	**Confirmed**
SSL / Custom Domains	**Azure Managed Certificates** + IONOS DNS CNAME/TXT records	**Confirmed**
Legacy Repo Strategy	Fresh `EmployeePortal` Git monorepo; lock `EmployeePortal-Legacy-V1` & `V2` read-only; rotate legacy secrets	**Confirmed**
Product Policy Sign-Off	Product Team review of interactive prototype (`mockups.employee.elikser.com`) & 8 policy items in PRD §7	**Ready for Review**