|  |
| --- |
| **Template 1** |
| **Devon database design report** |
| |  |  |  |  | | --- | --- | --- | --- | | **ORGANISATION NAME** | **ORGANISATION NAME: International Skills Institute (ISI)** | | | | **CONTACT NAME** | Colin | **ADDRESS** | **Level 5, 123 Education Boulevard Sydney, NSW 2000 Australia** | |
| **1. Business overview**  **The International Skills Institute (ISI) is a vocational education provider specializing in skills development and compliance training. Our mission is to deliver industry-relevant education to domestic and international students across key sectors including IT, business, healthcare, and trades.**  **Key Operational Areas:**   * **Vocational certificate and diploma programs** * **Industry compliance training** * **Corporate upskilling programs** * **Apprenticeship pathways**   **Database-Supported Core Functions:**   * **Student enrollment and progression tracking** * **Course delivery management** * **Financial operations and payment processing** * **Assessment and certification management** |
| **2. Current Database Design and Security Plan**  **Database Design Summary**  **Schema Overview**:   * **Dimension Tables**: **dim\_student**, **dim\_trainer**, **dim\_course**, **dim\_time**, **dim\_department** * **Fact Tables**: **fact\_enrollment**, **fact\_attendance** * **Relationships**: Star schema design with clear foreign key relationships * **Technology Stack**: MySQL 8.0 hosted on InfinityFree with PHP frontend   **Data Flow**:   1. Student enrollment → **fact\_enrollment** 2. Daily attendance → **fact\_attendance** 3. Periodic reporting → Aggregated from fact tables   **Security Plan**  **Current Measures**:   * Basic username/password authentication * Table-level permissions * Manual weekly backups   **Identified Gaps**:   * No encryption for sensitive data * Absence of role-based access control * Lack of audit logging * No multi-factor authentication |
| **3.** **Big Data Applications for ISI**  **Application 1: Predictive Student Analytics**  **Implementation**:   * Use historical **fact\_attendance** and **fact\_enrollment** data to predict at-risk students * Machine learning models to identify patterns leading to dropouts   **Tools**: Python (scikit-learn), Tableau for visualization **Benefits**: 15-20% improvement in student retention rates  **Application 2: Dynamic Curriculum Optimization**  **Implementation**:   * Analyze industry job postings and skills demand * Cross-reference with course completion data from **dim\_course**   **Tools**: Hadoop for processing, Power BI for dashboards **Benefits**: 30% faster curriculum adaptation to market needs |
| **4. Design vs. Security Plan Comparison**   | **Aspect** | **Current State** | **Future Needs** | | --- | --- | --- | | **Data Volume** | Handles 1,000 students | Scale to 10,000+ | | **Security** | Basic access controls | RBAC + encryption | | **Integration** | Standalone system | API integrations | | **Compliance** | Minimal | GDPR/APP ready |   **Critical Gaps**:   * No protection against SQL injection * Missing data encryption at rest * Inadequate backup verification |
| **5. Identify the database security requirements based on security plan.**  **Technical Controls**:   * Implement AES-256 encryption for PII * Deploy role-based access (RBAC) with:   + Admin: Full access   + Trainers: Read/write to attendance   + Students: Read-only self-data * Install intrusion detection system (IDS)   **Procedural Measures**:   * Quarterly security audits * Mandatory staff training on phishing * Incident response plan development   **Compliance**:   * Map controls to ISO 27001 Annex A * Implement APP 11 (security safeguards) |
| **6.** **Technical Specifications**  **Infrastructure Upgrades**:   * Migrate to AWS RDS (PostgreSQL compatible) * Implement read replicas for reporting   **Performance Optimization**:   * Add composite indexes on:   sql  CREATE INDEX idx\_enrollment ON fact\_enrollment(student\_id, course\_id);   * Partition **fact\_attendance** by year   **Backup Strategy**:   * Automated daily encrypted backups * 7-day retention with monthly archives * Test restoration procedures quarterly   **Monitoring**:   * Prometheus for database metrics * Alerting on failed login attempts |