# Physical Data Design for Licensing Self-Certification Portal (LSCP)

**Version 0.1**

**Jan 2025**

## 1. Introduction

This document provides a comprehensive description of the physical data structure and process design for the Licensing Self-Certification Portal (LSCP) Project. This document serves as a blueprint for the implementation of the LSCP database, ensuring a robust and efficient data management system.

The document delves into the detailed relationships between key business entities within the LSCP. A clear and concise diagrammatic representation is provided to visually illustrate the connections between these entities. This section also includes a comprehensive description of each entity, its attributes, and the data types used to represent them. The data relationships are further explained, highlighting the primary keys, foreign keys, and any constraints imposed on the data.

By providing a detailed understanding of the physical data structure, this document serves as a valuable resource for developers, database administrators, and other stakeholders involved in the implementation and maintenance of the LSCP.

## 2. Objectives

The objectives of the proposed Licensing Self-Certification Portal (LSCP) should:

1. Provide user-friendly and meaningful messages to users.
2. Store all electronic and paper submissions from applicant and authorized person (AP)/registered structural engineer (RSE) applications of the requisite safety certificates for registration of non-purpose built schools and child care centres, and applications in related to non-purpose non-local higher and professional education courses.
3. BD departmental portal login for internal users (BD), and provide User ID and password as an alternative.
4. Support the latest web browsers.
5. Comply with the standards of the Government System Architecture and government IT security policy.

## 3. Physical Data Structure Specification

This section documents the data model and its associated descriptions of the required system.

### 3.1 Physical Data Structure

An entity-relationship diagram consists of three basic elements such as entity, relationship, and attribute. Along with these are more components based on their main elements like weak entity, multi-valued attribute, and many more. Notations used to make ERD diagram examples include cardinality and ordinality to define relationships in numbers.

**Entity-Relationship Diagram (ERD):**

*(The ERD diagram from the original document is not directly representable in Markdown. A text-based description follows.)*

The data model includes the following entities and relationships:

* **Application:** Contains core application information.
* **Submission:** Represents a specific submission related to an application.
* **Attachment:** Stores file attachments related to applications and cases.
* **BsBlock:** Contains block ID and BDGIS information.
* **Task:** Represents a task in the workflow.
* **Case:** Represents a specific case related to an application.
* **Eminute:** Stores electronic minutes related to cases.
* **OAuthToken:** Stores OAuth tokens for authentication.
* **User:** Contains user information.
* **Notification:** Stores notifications for users.
* **SysFileRef:** Contains system file reference information.
* **AdrBlkFileRef:** Relates address blocks to file references.

**Relationships:**

* Application 1:N Submission
* Application 1:N Attachment
* Case 1:N Attachment
* Application 1:N Case
* Case 1:N Task
* Case 1:N Eminute

#### 3.1.1 (GCIS) Frontend - Application Forms Submission

*(The diagram from the original document is not directly representable in Markdown. A text-based description follows.)*

This part of the data structure focuses on the information collected through the frontend application forms. It includes entities related to applicant details, premise information, and other application-specific data.

#### 3.1.2 (GCIS) Frontend - OTP login control

*(The diagram from the original document is not directly representable in Markdown. A text-based description follows.)*

This section covers the entities used for managing OTP-based login functionality, including user IDs, OTP codes, and related timestamps.

#### 3.1.3 (BD) Backend - TBC

*(The diagram from the original document is not directly representable in Markdown. A text-based description follows.)*

This section is marked as "To Be Confirmed" and would detail the backend-specific entities and relationships, likely involving internal user data, workflow management, and other administrative aspects.

## 4. Data Entity Description

This section states the conversion rules, the assumptions applied for the physical data design, the names of the physical data tables, the corresponding required system entities and key details to be stored into the database.

The database is a physical store of contract related information and textual data inside a database management system (DBMS). For LSCP, **Microsoft SQL Server 2019** is selected for the database management system. **All the spatial and textual entity will be stored into Microsoft SQL Server 2019.**

The following tables document how the Logical Data Model (LDM) can be mapped onto the physical data design.

**LSCP Frontend**

| Table ID | LSCP Name | LSCP Entity Description | Key Nature | Key Data Item |
| --- | --- | --- | --- | --- |
| T-S-01 | ApplicationCases | Table to store all the application number | PK | Id |
|  |  |  |  | ApplicationNo |
| T-S-02 | SchoolApp\_Infos | Table to store the latest update of the submitted application data as 1 row | PK | Id |
|  |  |  |  | ApplicationNo |
| T-S-03 | SchoolApp\_Submissions | Table to store the submission of each application | PK | Id |
|  |  |  |  | ApplicationNo |
|  |  |  |  | SubmissionId |
| T-S-04 | ApplicationFiles | Table to store all the path of applicant upload files | PK | Id |
|  |  |  |  | ApplicationNo |
|  |  |  |  | SubmissionId |
| T-S-05 | LSCPMasterTable | Table to store meta-data or parameter data for frontend | PK | Id |
|  |  |  |  | Code |
|  |  |  |  | Type + Code |
| T-S-06 | GenOtp | Table to store generated OTP code and the usage status | PK | Id |
|  |  |  |  | ApplicationNo + userId + Otp |
| T-S-07 | AdrBlk | Table to store addresses that import from BCIS | PK | ADR\_BLK\_ID |
| T-S-08 | SYS\_META\_DATA | Table to store meta data that import from BCIS | PK | SYS\_META\_DATA\_ID |
|  |  |  |  | REC\_TYPE |
|  |  |  |  | CODE |
| T-S-09 | Aprse | Table to store AP / RSE information that import from MWMS 2.0 | PK | Id |
|  |  |  |  | Name + RegistrationNumber |

**LSCP Backend**

| Table ID | LSCP Name | LSCP Entity Description | Key Nature | Key Data Item |
| --- | --- | --- | --- | --- |
| T-S-01 | ApplicationCases | Table to store all the application number | PK | Id |
|  |  |  |  | ApplicationNo |
| T-S-02 | SchoolApp\_Infos | Table to store the latest update of the submitted application data as 1 row | PK | Id |
|  |  |  |  | ApplicationNo |
| T-S-03 | SchoolApp\_Submissions | Table to store the submission of each application | PK | Id |
|  |  |  |  | ApplicationNo |
|  |  |  |  | SubmissionId |
| T-S-04 | ApplicationFiles | Table to store all the path of applicant upload files | PK | Id |
|  |  |  |  | ApplicationNo |
|  |  |  |  | SubmissionId |
| T-S-05 | LSCPMasterTable | Table to store meta-data or parameter data for frontend | PK | Id |
|  |  |  |  | Code |
|  |  |  |  | Type + Code |
| T-S-06 | SYS\_META\_DATA | Table to store meta data that import from BCIS | PK | SYS\_META\_DATA\_ID |
|  |  |  |  | REC\_TYPE |
|  |  |  |  | CODE |
| T-S-07 | SYS\_META\_DATA | Table to store meta data that import from BCIS | PK | SYS\_META\_DATA\_ID |
|  |  |  |  | REC\_TYPE |
|  |  |  |  | CODE |

\*\*\* End of document\*\*\*