ERD Designs and diagrams

Name of student

Name of professor

University

Course

Date

**Introduction**

In the previous exercise and activity, the database project selection was made on an Estate security application. The objective of the application is too able to integrate all the other components /classes of the application by helping to report and cub insecurity in the concerned regions and areas of the estate.

The application allows the user to submit /report an insecurity concern through the application by either, recording voice, uploading document or media through the application. Once this is done, the system admin dashboard is able to view all records that have been submitted by the users and appropriate actions can be taken by the authorities concerned.

Based on the above analogy, an analyst can then be able to draw certain use case classes belonging to this application by defining the relevant table classes needed to execute the project. These classes include:

* Tenants class
* Partners class
* Crime class
* Payments class
* Transactions class
* Location class
* Estate class

These tables can then be broken down to illustrate the ways each and every table here would look like:

**Tenants table;**

|  |  |  |  |
| --- | --- | --- | --- |
| Tenant\_id | Tenant\_name | Tenant\_status | Crime\_id |
| PRIMARY KEY (PK) |  |  | FOREIGN KEY (FK) |
|  |  |  |  |

**Partners table**

|  |  |  |  |
| --- | --- | --- | --- |
| Patner\_id | Patner\_name | Patner\_estate | Estate\_id |
| PRIMARY KEY (PK) |  |  | FOREIGN KEY (FK) |
|  |  |  |  |

Crime table;

|  |  |  |
| --- | --- | --- |
| Crime\_id | Crime\_name | Crime\_location |
| PRIMARY KEY (PK) |  | FOREIGN KEY (FK) |
|  |  |  |

Payments class

|  |  |  |  |
| --- | --- | --- | --- |
| Payment\_id | Payment\_name | Payment\_amount | Transaction\_id |
| PRIMARY KEY (PK) |  |  | FOREIGN KEY (FK) |

Trnsanctions class

|  |  |  |  |
| --- | --- | --- | --- |
| Trsnaction\_id | Trsnaction\_name | Transaction\_amount | Payment\_id |
| PRIMARY KEY(PK) |  |  | FOREIGN KEY (FK) |
|  |  |  |  |

Location class

|  |  |  |  |
| --- | --- | --- | --- |
| Location\_id | Location\_name | Estate\_id | Location\_status |
| PRIMARY KEY(PK) |  | FOREIGN KEY (FK) |  |
|  |  |  |  |

**Estate class**

|  |  |  |  |
| --- | --- | --- | --- |
| Estate\_id | Estate\_name | Location\_id | Estate\_status |
| PRIMARY KEY(PK) |  | FOREIGN KEY (FK) |  |
|  |  |  |  |

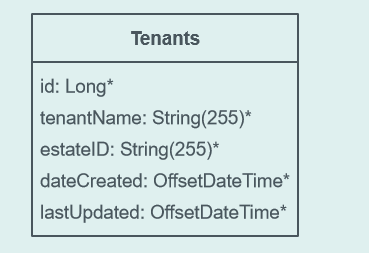
**Rules:**

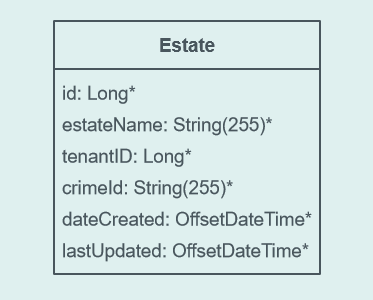
During the database design implementation, the primary keys inherit as foreign keys in other tables, allowing the application to request resources from another class using one primary key and the associated foreign key in the other table. Bearing this in mind, the implementation of the database rules can then be implemented:

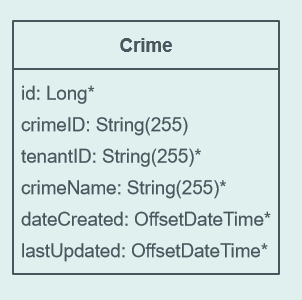
* ONE tenant can have MANY transactions;
* ONE transaction can only belong to ONE user
* MANY tenants can be in MANY estates
* ONE location can have MANY estates
* ONE crime belongs to ONE estate
* ONE crime can be reported by MANY tenants
* MANY partners can belong to MANY estates

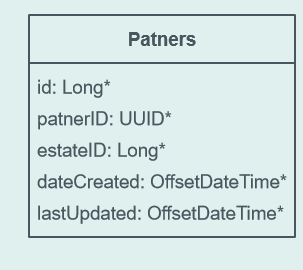
**ERD diagrams**

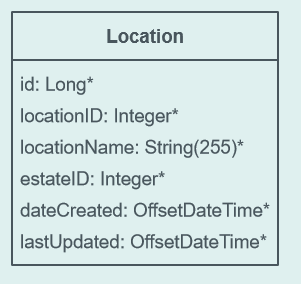
The associated ERD diagrams for these look like below;

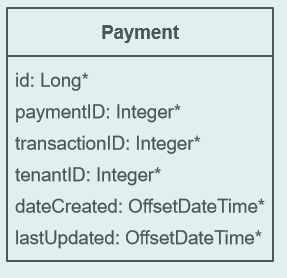












**Table class relationships**

