|  |  |
| --- | --- |
|  | 2814ICT – Data Management  7003ICT – Database Design  School of Information & Communication Technology  Trimester 1, 2020  **Assignment Part 1:**  **Designing a Database for Commonwealth Transport Services** |

**ASSIGNMENT TITLE: Commonwealth Transport Services Database Project**

|  |  |  |
| --- | --- | --- |
| **Student 1** | **s-number: s5098442** | **Full name: Travis Jacob** |
| **Student 2** | **s-number: s5168360** | **Full name: Atif Hossain** |
| **Student 3** | **s-number: s5054158** | **Full name: Damian Garcia** |
| **Course Code: 2814ICT** | | **Workshop/Lab day & time: Friday 12pm** |
| **Tutor’s name: Nosheen Munir** | | **Date submitted:** |

**Marks obtained:** \_\_\_\_\_\_\_\_. [For marker to fill up.]

**PLAGIARISM**

Plagiarism: occurs when the work of another is represented, intentionally or unintentionally, as one's own original work, without appropriate acknowledgement of the author or the source. See more at <https://www.griffith.edu.au/academic-integrity/information-for-students/what-is-plagiarism>.

Plagiarism is a serious offence. Refer to the following document on Student Academic Misconduct:

<http://policies.griffith.edu.au/pdf/Student%20Academic%20Misconduct%20Policy.pdf>.



|  |  |  |
| --- | --- | --- |
| Declaration  Except where appropriately acknowledged, this assignment is our own work, has been expressed in our own words and has not previously been submitted for assessment. We have also retained a copy of this assessment piece for our own records. | | |
| **Student 1:** | **Student 2:** | **Student 3:** |
| Name: Travis Jacob  Signature:  Date: 7.4.20 | Name: Atif Hossain  Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Date: 8.4.20 | Name: Damian Garcia    Signature: \_\_\_\_\_\_\_\_\_\_\_\_  Date: 9.4.20 |

**Note: All students in the group must sign this first page, scan the signed page, and then place at the beginning of the assignment.**

# **Table of Contents**

[Generate a table of content with page numbers.]

# 

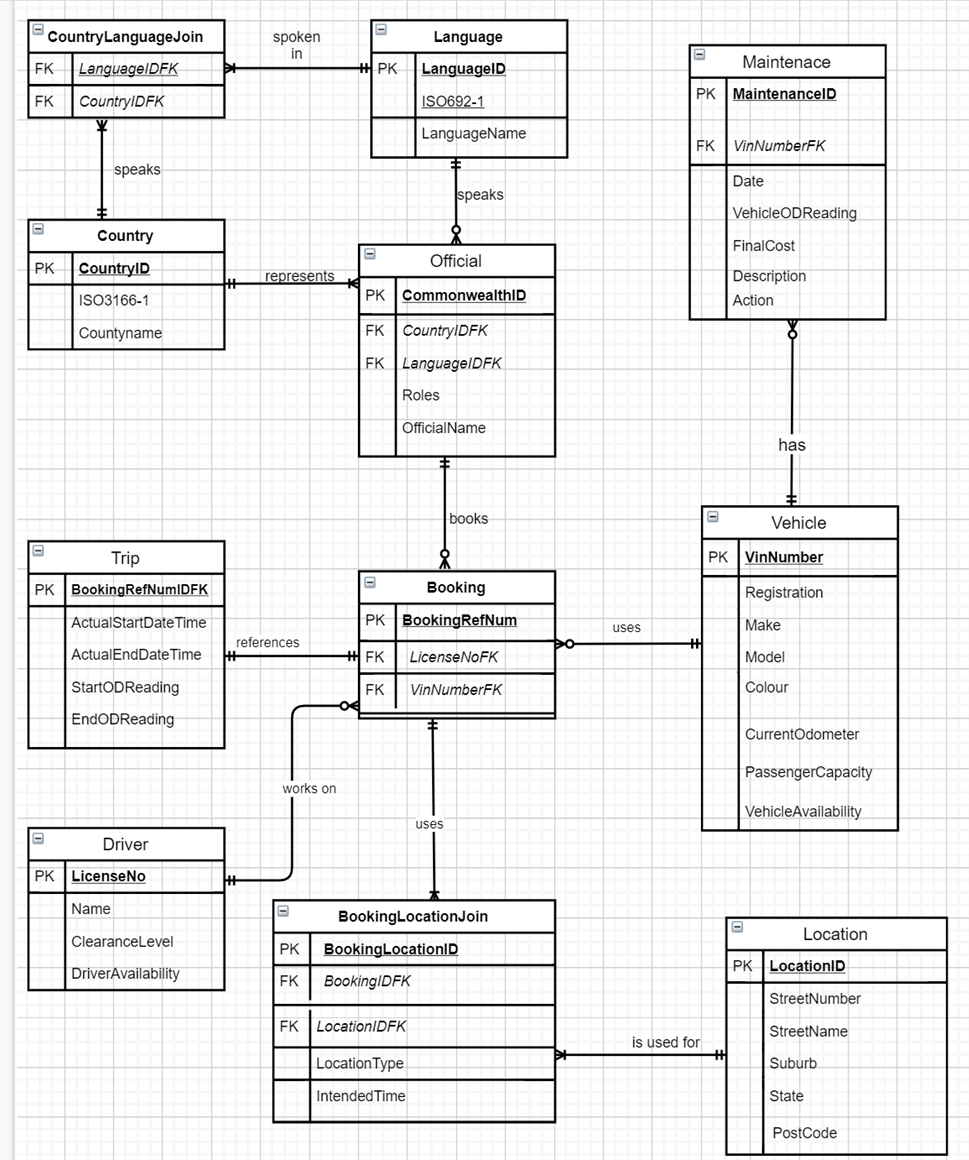
# **List of Illustrations**

[Generate a list of figures and tables with page numbers.]

# **Acknowledgements:**

1. Tutor – Nosheen Munir

# **Entity Relationship Diagram**



# **Assumptions**

* + Only one driver is needed for every trip, and only one vehicle is used
  + Every country needs one official
  + The location type and intended time being put into the join table allow for multiple locations to be added to each booking, so that there is not just one pickup and one drop off, with intended times for both. Each booking can have many BookingLocationJoins which allow there to be multiple locations with a time for each. There is also a field called LocationType which can be set to be pickup or drop off for each location in the booking so the records can be referenced as either type.

# **Normalisation**

#### Relation Schema

1. CountryLanguageJoin(*LanguageIDFK, CountryIDFK*)
2. Language(**LanguageID,** ISO692-1, LanguageName)
3. Country(**CountryID**, ISO3166-1, Countryname)
4. Official(**CommonwealthID**, *CountryIDFK, LanguageIDFK,* Roles, OfficialName)
5. Maintenance(**MaintenanceID**, *VinNumberFK*, Date, VehicleODReading, FinalCost, Description, Action)
6. Vehicle(**VinNumber**, Registration, Make, Model, Colour, CurrentOdometer, PassengerCapacity, VehicleAvaliability)
7. Booking(**BookingRefNum**, *LicenseNoFK, VinNumberFK*)
8. Trip(**BookingRefNumIDFK**, ActualStartDateTime, ActualEndDateTime, StartODReading, EndODReading)
9. Driver(**LicenseNo**, *LanguageIDFK*, Name, ClearanceLevel, DriverAvaliability)
10. BookingLocationJoin(**BookingLocationID**, *BookingIDFK, LocationIDFK*, LocationType, IntendedTime)
11. Location(**LocationID**, StreetNumber, StreetName, Suburb, State, Postcode)
12. **Normalisation**

CountryLanguageJoin(*LanguageIDFK, CountryIDFK*)

This relational data structure is in 1st NF:

* *LanguageIDFK*
* *CountryIDFK*
* In 1st NF due to identifying foreign keys with no dependency

Language(**LanguageID ISO692-1**, LanguageName)

This relational data structure is in 3rd NF:

* LanguageID -> ISO692-1, LanguageName

Country(**CountryID**, ISO3166-1, Countryname)

This relational data structure is in 3rd NF:

* **CountryID** -> ISO3166-1, CountryName

Official(**CommonwealthID**, *CountryIDFK, LanguageIDFK*, Roles, OfficialName)

This relational data structure is in 3rd NF:

* **CommonwealthID** -> *CountryIDFK, LanguageIDFK*, Roles, OfficialName

Maintenance(**MaintenanceID**, *VinNumberFK*, Date, VehicleODReading, FinalCost, Description, Action)

This relational data structure is in 2nd NF:

* **MaintenanceID** -> Date, Description, Action, FinalCost
* *VinNumberFK* -> VehicleODReading
* This is in 2nd NF due to still having transparent dependancy

Vehicle(**VinNumber**, Registration, Make, Model, Colour, CurrentOdometer, PassengerCapacity, VehicleAvaliability)

This relational data structure is in 3rd NF:

* **VinNumber** -> Registration, Make, Model, Colour, CurrentOdometer, PassengerCapacity, VehicleAvaliability

Booking(**BookingRefNum**, *LicenseNoFK, VinNumberFK*)

This relational data structure is in 3rd NF:

* **BookingRefNum** -> *LicenseNoFK, VinNumberFK*

Trip(**BookingRefNumIDFK**, ActualStartDateTime, ActualEndDateTime, StartODReading, EndODReading)

This relational data structure is in 3rd NF:

* **BookingRefNumberIDFK** -> ActualStartDateTime, ActualEndDateTime, StartODReading, EndODReading

Driver(**LicenseNo**, LanguageIDFK, Name, ClearanceLevel, DriverAvaliability)

This relational data structure is in 3rd NF:

* **LicenseNo** -> *LanguageIDFK*, Name, ClearanceLevel, DriverAvaliability

BookingLocationJoin(**BookingLocationID**, BookingIDFK, LocationIDFK, LocationType, IntendedTime)

This relational data structure is in 3rd NF:

* **BookingLocationID** -> *BookingIDFK, LocationIDFK,* LocationType, IntendedTime

Location(**LocationID**, StreetNumber, StreetName, Suburb, State, Postcode)

This relational data structure is in 3nd NF:

* **LocationID** -> StreetNumber, StreetName, Postcode, Suburb, State

# **Relational Database Schema**

|  |  |  |  |
| --- | --- | --- | --- |
| Table Name | Field | Type | Description |
| BOOKING | BookingRefNum | INT (30) | PRIMARY KEY |
|  | LicenseNoFK | INT (9) | FOREIGN KEY REFERENCES DRIVER(LicenseNo) |
|  | VinNumberFK | INT (17) | FOREIGN KEY REFERENCES VEHICLE(VinNumber) |
| TRIP | BookingRefNumIDFK | INT (30) | PRIMARY KEY, FOREIGN KEY REFERENCES BOOKING(BookingRefNum) |
|  | ActualStartDateTime | DATETIME |  |
|  | ActualEndDateTime | DATETIME |  |
|  | StartODReading | INT (7) |  |
|  | EndODReading | INT (7) |  |
| DRIVER | LicenseNo | INT (9) | PRIMARY KEY |
|  | Name | VARCHAR (40) |  |
|  | ClearanceLevel | INT (1) | Integer from 1 - 4 |
|  | DriverAvailability | BIT | Binary 0 or 1 |
| LOCATION | LocationID | INT (30) | PRIMARY KEY |
|  | StreetNumber | INT (4) |  |
|  | StreetName | VARCHAR (50) |  |
|  | Suburb | VARCHAR (40) |  |
|  | State | VARCHAR (40) |  |
|  | PostCode | INT (5) |  |
|  |  |  |  |
| BOOKINGLOCATIONJOIN | BookingLocationID | INT (30) | PRIMARY KEY |
|  | BookingIDFK | INT (30) | FOREIGN KEY REFERENCES BOOKING(BookingRefNum) |
|  | LocationIDFK | INT (30) | FOREIGN KEY REFERENCES LOCATION(LocationID) |
|  | LocationType | VARCHAR (7) | pickup or dropoff |
|  | IntendedTime | DATETIME |  |
| VEHICLE | VinNumber | INT (17) | PRIMARY KEY |
|  | Registration | VARCHAR (10) |  |
|  | Make | VARCHAR (20) |  |
|  | Model | VARCHAR (20) |  |
|  | Colour | VARCHAR (20) |  |
|  | CurrentOdometer | INT (7) |  |
|  | PassengerCapacity | INT (2) |  |
|  | VehicleAvailability | BIT | 0 for unavailable, 1 for available. Front end system can control value |
| MAINTENANCE | MaintenanceID | INT (30) | PRIMARY KEY |
|  | Date | DATE |  |
|  | VehicleODReading | INT (7) |  |
|  | FinalCost | VARCHAR (10) |  |
|  | Description | VARCHAR (500) |  |
|  | Action | VARCHAR (1) | M for maintenance, R for repair |
|  | VinNumberFK | INT (17) | FOREIGN KEY REFERENCES VEHICLE(VinNumber) |
| OFFICIAL | CommonwealthID | VARCHAR (8) | PRIMARY KEY |
|  | Roles | VARCHAR (50) |  |
|  | OfficialName | VARCHAR (40) |  |
|  | LanguageIDFK | INT (30) | FOREIGN KEY REFERENCES LANGUAGE(LanguageID) |
|  | CountryIDFK | INT (30) | FOREIGN KEY REFERENCES COUNTRY(CountryID) |
| COUNTRY | CountryID | INT (30) | PRIMARY KEY |
|  | CountryName | VARCHAR (50) |  |
|  | ISO3166-1 | VARCHAR (2) | 2 character country code |
| LANGUAGE | LanguageID | INT (30) | PRIMARY KEY |
|  | LanguageName | VARCHAR (40) |  |
|  | ISO629-1 | VARCHAR (2) | 2 character language code |
| COUNTRYLANGUAGEJOIN | CountryLanguageID | INT (30) | PRIMARY KEY |
|  | LanguageIDFK | INT (30) | FOREIGN KEY REFERENCES LANGUAGE(LanguageID) |
|  | CountryIDFK | INT (30) | FOREIGN KEY REFERENCES COUNTRY(CountryID) |

# **Appendices**

[Add any additional work other than what has been requested. Your marker may not look at and mark the content in this section.]

# **Bibliography**

W3Schools. (n.d.). *SQL Data Types for MySQL, SQL Server, and MS Access*. Retrieved from https://www.w3schools.com/sql/sql\_datatypes.asp