1. Esquema Relacional

Cada relación cuenta con sus dependencias funcionales en caso de no ser triviales.

1. Address(id, country, state, street, pc, city, district)
   a) pc country → city
   b) pc country → district
   c) pc country → state
   d) pc country district street → city
   e) city → state
   f) state → country
2. School(id, code, name, address_id, created_at, updated_at, deleted_at)
3. School_Email(school_id, school_code, email)
4. School_Telephone(school_id, school_code, telephone)
5. Address_School(address_id, school_id)
6. Reservation(id, school_code, school_id, principal_teacher, visit_date, visit_hour, method, created_at, updated_at, deleted_at)
7. Visit_Type(id, code, name, description, created_at, updated_at, deleted_at)
8. Product_Visit(id, visit_type_code, visit_type_id, cost, is_active, created_at, updated_at, deleted_at)
9. Reserve_Zone(id, reservation_id, visit_type_id, visit_type_code, principal_teacher, grade, created_at, updated_at, deleted_at)
10. Reserve_Zone_Teacher(reserve_zone_id, teacher)
11. Student(reserve_zone_id, name, created_at, updated_at, deleted_at)
12. Ticket(id, code, product_visit_id, student_id, created_at, updated_at, deleted_at)
13. Employee(rfc, salary, names, father_last_name, mother_last_name, email, telephone, day_off, created_at, updated_at, deleted_at)
14. Address_Employee(address_id, employee_id)
15. Police(rfc, radio_code, is_armed)
16. Scientist(rfc, license)
17. Veterinary(rfc, specialty)
18. Quartermaster(rfc, prof_license)
19. Driver(rfc, license_number, license_expires_at)
20. Vehicle(id, manufacturer, model, type, capacity, fuel_type, is_automomous, is_pc_interective, details, created_at, updated_at, deleted_at)
    a) manufacturer model → type
    b) manufacturer model → capacity

   c) manufacturer model → fuel_type
   d) manufacturer model → is_automomous

21. Drive(driver_employee_rfc, vehicle_id)
22. Zone(code, type, created_at, updated_at, deleted_at, security_rfc, scientist_rfc, veterinary_rfc, quatermaster_rfc)
23. Zone_Vehicle(zone_code, vehicle_zone)
24. Visit(reserve_zone_id, zone_id)
25. Vegetation_Type(id, detail)
26. Dinosaur(id, zone_id, name, class, phylum, kingdom, distribution, surface, feed_type, avg_weight, avg_height, vegetation_type_id, created_at, updated_at, deleted_at)
    a) name → surface
    b) name → class
    c) class → phylum
    d) phylum → kingdom
27. Dinosaur_Safety_Requirements(dinosaur_id, safety_requirement)

2. Esquema Relacional Normalizado

El esquema anterior ya se encuentra en **1FN** desde la declaración del esquema relacional.
El esquema anterior ya se encuentra en **2FN** desde la declaración del esquema relacional.
El esquema siguiente se encuentra en **3FN** despues de generar las relaciones nuevas y así el esquema perdio sus dependencias fucionales que dependian de un atributo no llave.

1. Country(id, name)
2. State(id, country_id, name)
3. City(id, state_id, name)
4. Address(id, city_id, number, street, pc, district)
5. School(id, code, name, address_id, created_at, updated_at, deleted_at)
6. School_Email(school_id, school_code, email)
7. School_Telephone(school_id, school_code, telephone)
8. Address_School(address_id, school_id, school_code)
9. Reservation(id, school_code, school_id, visit_date, visit_date, method, created_at, updated_at, deleted_at)
10. Visit_Type(id, code, name, description, created_at, updated_at, deleted_at)
11. Product_Visit(id, visit_type_code, visit_type_id, cost, is_active, created_at, updated_at, deleted_at)
12. Reserve_Zone(id, reservation_id, visit_type_id, visit_type_code, principal_teacher, grade, created_at, updated_at, deleted_at)
13. Reserve_Zone_Teacher(reserve_zone_id, teacher)
14. Student(reserve_zone_id, name, created_at, updated_at, deleted_at)

15. Ticket(id, code, product_visit_id, student_id, created_at, updated_at, deleted_at)
16. Employee(rfc, salary, names, father_last_name, mother_last_name, email, telephone, day_off, created_at, updated_at, deleted_at)
17. Address_Employee(address_id, employee_id)
18. Police(rfc, radio_code, is_armed)
19. Scientist(rfc, license)
20. Veterinary(rfc, specialty)
21. Quartermaster(rfc, prof_license)
22. Driver(rfc, license_number, license_expires_at)
23. Vehicle_Model(id, manufacturer, model,type, capacity, fuel_type, is_automomous)
24. Vehicle(id, vehicle_model_id, is_pc_interective, details, created_at, updated_at, deleted_at)
25. Drive(driver_employee_rfc, vehicle_id)
26. Zone(code, type, created_at, updated_at, deleted_at, security_rfc, scientist_rfc, veterinary_rfc, quatermaster_rfc)
27. Zone_Vehicle(zone_code, vehicle_zone)
28. Visit(reserve_zone_id, zone_id)
29. Vegetation_Type(id, detail)
30. Tax_Kingdom(id, name)
31. Tax_Phylum(id, tax_kingdom_id, name)
32. Tax_Class(id, tax_phylum_id, name)
33. Dinosaur(id, zone_id, tax_class_id, name, distribution, surface, feed_type, weight, height, vegetation_type_id, created_at, updated_at, deleted_at)
34. Dinosaur_Safety_Requirements(dinosaur_id, safety_requirement)

3. SMBD

La elección fue MySQL por:

- OpenSource y tambien cuenta con una versión privada.
- Comunidad.
- Multiplataforma.
- Amplio subconjunto de SQL.
- De los mejores rendimientos del mercado.