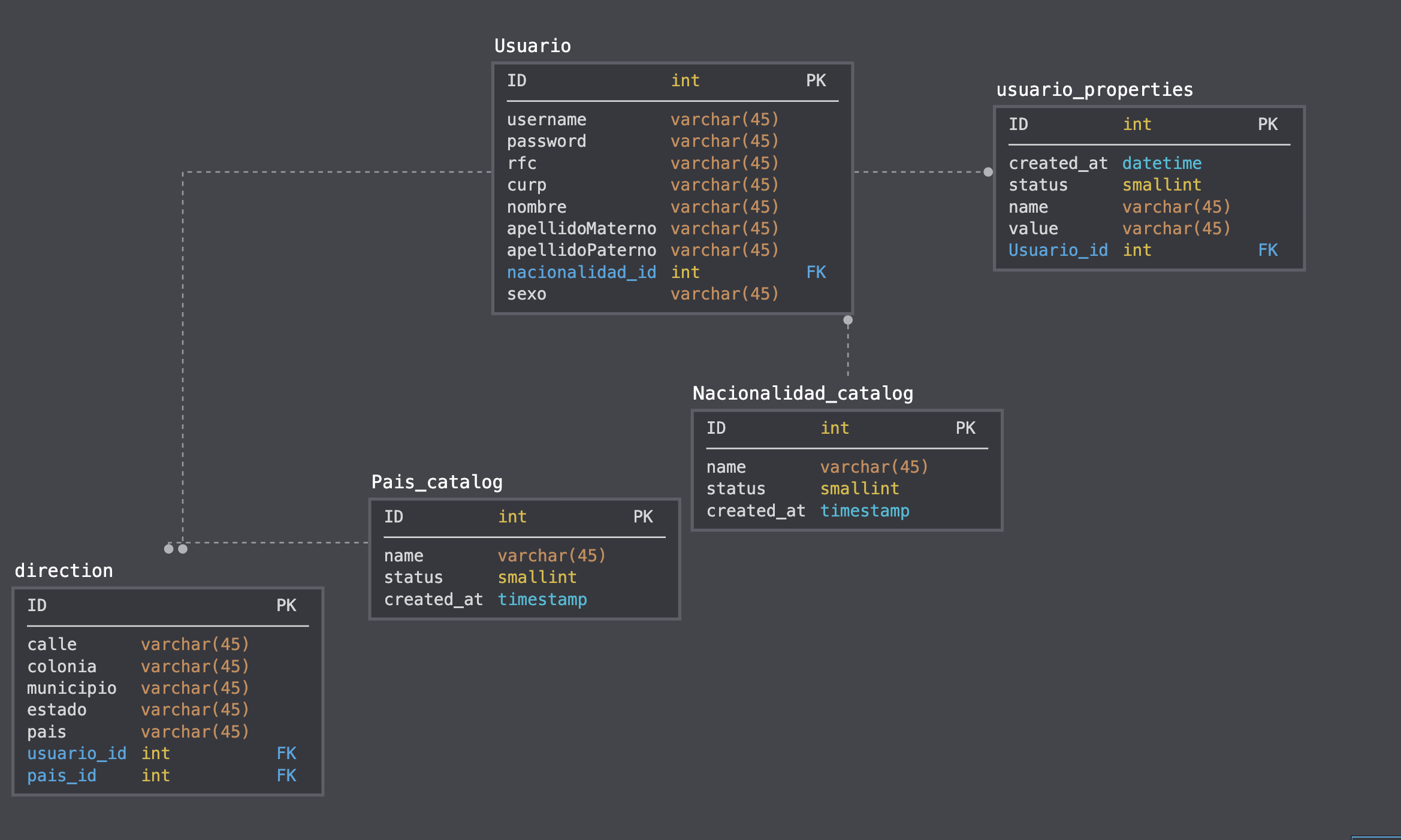
Aguirre Olvera Luis Fernando

Base de datos

1.-

|  |  |
| --- | --- |
| SQL | No SQL |
| No permite la redundancia de datos | Permite redundancia de datos |
| Manejo sobre un esquema estricto | No obliga a tener una esquema estático de los datos |
| Complicaciones sobre cambios en las tablas | Hay bases documentales como es Mongo DB |
| La mayoría de su lógica se encuentra en la base de datos | La mayoría de la lógica parte del código del programador o equipo |
| Almacenamiento estructurado | Recomendados en el manejo de datos masivos o Big data |
| Consistencia en los datos | Alta disponibilidad de los datos |
|  | Consultas más rápidas |

2-



3.-

{  
 "nombre": "string",  
 "apellidoM": "string",  
 "curp": "string",  
 "rfc": "string",  
 "direction": [{  
 "calle" : "string",  
 "colonia" : "string",  
 "municipio" : "string",  
 "estado" : "string",  
 "pais" : "string"  
 }],  
 "nacionalidad": "string",  
 "username": "string",  
 "password": "string"  
}

4.- Un DDL es un lenguaje de definición que proporcionado por el gestor de base de datos, el cual nos ayuda a hacer la estrutuca de nuestra de base de datos , por ejemplo son los Create , Drop y Alter , que directamente se centra en la estructura de los datos y tablas que los contienen

5.- Un DML es un lengual de manipulación de los datos que es al igual es proporcionado por el gestor de base de datos , el cual nos ayuda a interactuar, manipular y consultar los datos que contienen la base de datos, por ejemplo Select, Insert, Update , Delete.

6. – Las funciones de agrupaciones son las que utilizamos para poder consultar datos de diferentes tablas pero que entre estas mismas tienen una llave de relación la cual nos permiten agrupar los datos y poder consultar los datos necesarios, por ejemplo son InnerJoin , Leftjoin, RightJoin entre otros.

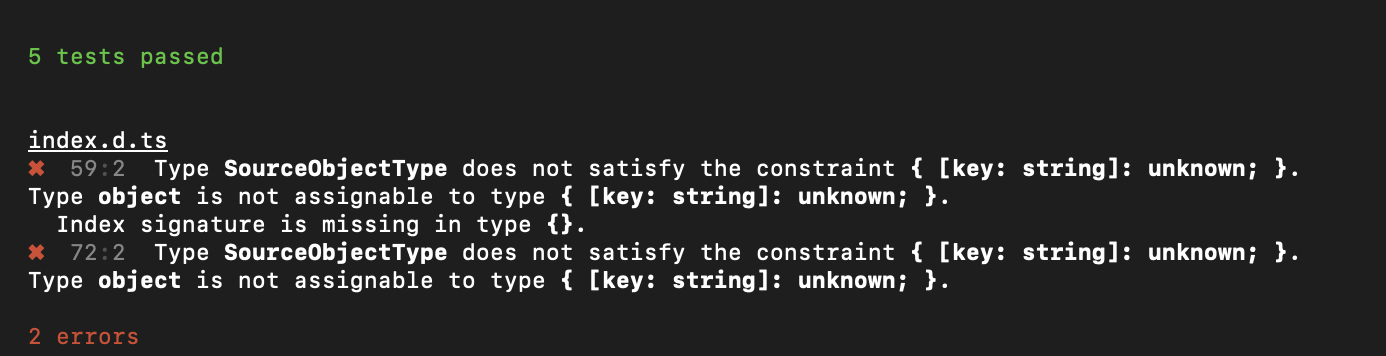
Las funciones de agregación son las que utlizamos pricipalmente para poder separar la informacion necesaria mediante uno o mas filtros , donde mas se utliza es un NoSql, ejemplo de ello son los aggregate de MongoDB, el cual nos permite agregar y agrupar documentos por algunos de sus campos , y otro ejemplo de eso son los consolidados, que nos permiten hacer estadistica de forma mas rapida en las consultas , ya que se consolidan documentos por campos iguales o parecidos.

7.- El sistema continuará funcionando cuando se produzcan particiones de red, eso nos permite que si hay una interrumpcion en alguno de los nodos , podamos seguir funcionando.

8.- Al menos un nodo funciona y debe estar disponible par apoder servir datos, lo que nos permite una alta disponibilidad como lo brinda Mongo DB

**Testing**

3.- Al correr las pruebas unitarias, nos damos cuenta que sale una excepcion de tipo SourceObject, y al leer el archivo de test, nos damos cuenta que la parte que falto probar, es la de isObject , que su función es decirnos si una variable es de tipo Object y asi decirle al algoritmo que pueda trabajar con esta variable



4.-

* Prueba de asignacion correcta de grupo para aprobar
* Prueba de formato correcto de solicitud
* Prueba de estatus de la solicitud
* Prueba de estatus de solicitud aceptada
* Prueba de estatus de solicitud denegada
* Prueba de resultados de Financial System
* Prueba de trasnferencia correcta
* Prueba de almacenaje correcto

Preguntas

1.- Validacion es una evaluacion final que nos ayuda a comprobar que se cumple con el valor deseado o acordado desde un inicio, por otro lado verficación es comprobar que el proceso se esta realizando correctamente y que se cumpla el requisito inicialmente acordado.

2.- Cuando se cubre la mayoria de los casos funcionales en las pruebas unitarias , una aseguración del codigo y la calidad del mismo para que pueda cumplir las requerimientos iniciales y tener un mejor nivel de aceptación y menor nivel de error

3.- Es una prueba basda en ejecución, revisión para una retroalimentacion del software , estas pruebas se diseñan para probarse en el momento del desarrollo , y tambien se consideran pruebas de tipo caja negra.

4.-Las pruebas de integración sirven para probar de una manera sistematica un proyecto como un todo , a diferencia que las pruebas unitarias solamente esta probando sobre partes del sistema.

5.- Las diferencias principales son que QA es un sistema de prevención que se enfoca principalmente a predecir fallos sobre el softwarem y genera medidas correctivas y evita que el producto llegue con defectos a la parte productiva y por lo tanto podremos decir que QA esta orientado al proceso. Por otro lado QC es un sistema de correción y de mejora de la calidad y comportamiento del software, y este se ve implementado al final del proceso donde ya tenemos el producto finalizado ,su enfoque principal esta orientado al producto.

6.- Son pruebas de rendimiento la cual nos ayuda a ver que tipo de fallas o ventajas tenemos al momento de hacer codigo , al igual existeb prueba de stress que nos permiten llevar nuestro producto a una demanda muy fuerte y ver el punto de rompimiento y poder tomar decisiones basado en ello.