



MANUAL TÉCNICO:

Proyecto: Página web vivienda compartida

Jonathan Payan

Sergio Aguirre

Juan Felipe Duque

Hanner Obando Ramírez

Programación III

Universidad de Caldas

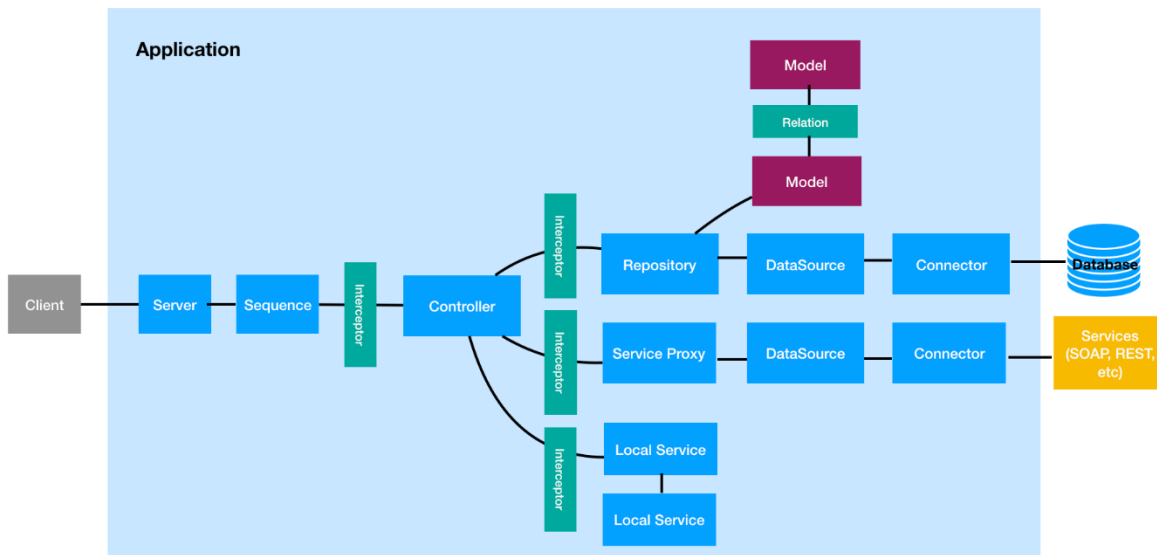
Ingeniería en sistemas y computación

CONTENIDO

1. Arquitectura API REST.....	3
2. Base de datos.....	4
2.1. Servicio para la base de datos.....	4
2.2. Diagrama de la base de datos.....	5
2.3. Diccionario de datos.....	6

1. Arquitectura API REST:

Backend desarrollado en Node JS mediante el framework LoopBack siguiendo la siguiente arquitectura:



Nuestro **cliente** (página web) realiza una solicitud al **servidor** y este efectúa una **secuencia** (solicitud) al controlador, estos dos últimos se comunican a través de un **interceptor** que según su patrón lógico decidirá si retorna al cliente o continua con la petición.

El **controlador** expone las acciones CRUD (Crear, leer, actualizar y borrar) y según sea la solicitud del cliente y cumpliendo con un interceptor, el **repositorio** accederá:

- Al **modelo** y **modelos relacionados** a través de una id.
- A la **fuentes de datos** que esta por medio de un **conector** (MongoDB) accederá a la **base de datos**.

Finalmente efectuara la acción CRUD correspondiente.

2. Base de datos:

2.1. Servicio para la base de datos:



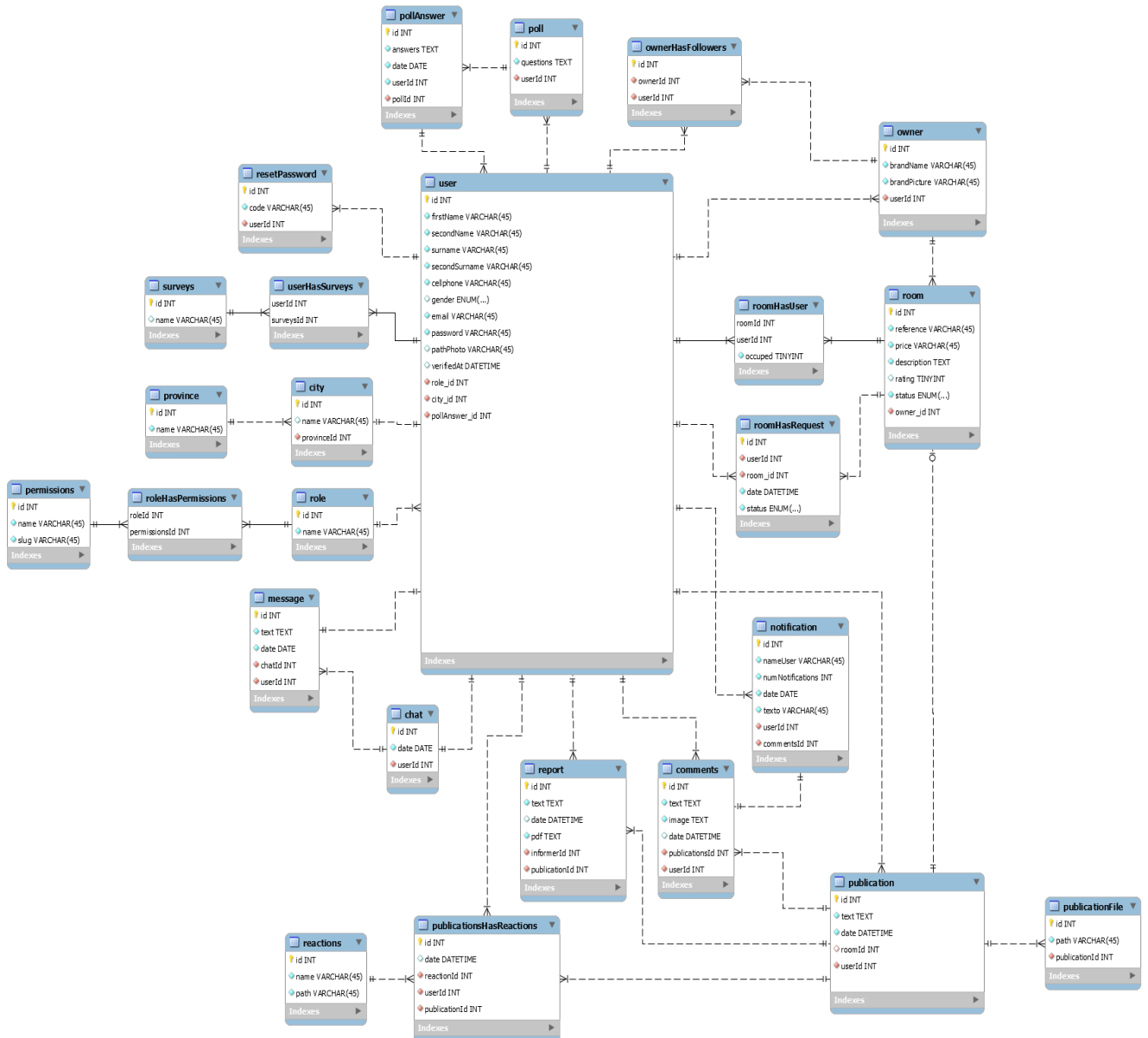
Base de datos no-relacional (NOSQL) con colecciones orientadas a documentos basados en JSON pero almacenados como BSON. La base de datos es operada y mantenida por MongoDB a través de su servicio en la nube que nos proporciona MongoDB Atlas.

MongoDB Atlas nos brinda:

- Servicio escalable al ser un entorno cloud.
- Entorno seguro con controles de acceso a través de usuarios en whitelist mediante ip y con respectivos roles.
- Alta disponibilidad al ofrecernos zonas distintas para los diferentes servidores.
- Protección de datos gracias al respaldo con copias de seguridad.

2.2. Diagrama de la base de datos:

Para la creación de la base de datos se llevó acabo un análisis con el cual se desarrolló el siguiente diagrama de base datos que sirvió de guía para la creación del backend para la página web:



2.3. Diccionario de datos:

Diccionario de datos desarrollado según los modelos creados en el backend de la pagina web:

chat											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
id	INT	✓	✓					✓			
date	DATE		✓								
userId	INT		✓							propietario del chat	

city											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
id	INT	✓	✓								
name	VARCHAR(45)										
provinceId	INT		✓							provincia a la que pertenece	

comments											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
id	INT	✓	✓					✓			
text	TEXT		✓								
image	TEXT		✓								
date	DATETIME										
publicationsId	INT		✓							publicación a la que pertenece	
userId	INT		✓							usuario que comenta	

message											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
id	INT	✓	✓					✓			
text	TEXT		✓								
date	DATE		✓								
chatId	INT		✓							chat al que pertenece	
userId	INT		✓							Propietario del mensaje Si el chat.user_id == message.user_id => emisor Si el chat.user_id != message.user_id => receptor	

notification											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
id	INT	✓	✓					✓			
nameUser	VARCHAR(45)		✓								
numNotifications	INT		✓								
date	DATE		✓								
texto	VARCHAR(45)		✓								
userId	INT		✓								
commentsId	INT		✓								

owner											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
id	INT	✓	✓					✓			
brandName	VARCHAR(45)		✓								
brandPicture	VARCHAR(45)		✓								
userId	INT		✓								

ownerHasFollowers											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
id	INT	✓	✓								
ownerId	INT		✓								
userId	INT		✓								

permissions											
Column name	DataType	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment	
id	INT	✓	✓					✓			
name	VARCHAR(45)		✓								
slug	VARCHAR(45)		✓								

poll

Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
id	INT	✓	✓							
questions	TEXT		✓							array con las preguntas
userId	INT		✓							dueño de la encuesta

pollAnswer

Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
id	INT	✓	✓							
answers	TEXT		✓							array con las respuestas
date	DATE		✓							
userId	INT		✓							usuario que responde
pollId	INT		✓							id de la encuesta

province

Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
id	INT		✓	✓						
name	VARCHAR(45)		✓							

publication

Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
id	INT	✓	✓					✓		
text	TEXT		✓							
date	DATETIME		✓							
roomId	INT									
userId	INT		✓							

publicationFile

Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
id	INT	✓	✓					✓		
path	VARCHAR(45)		✓							
publicationId	INT		✓							

publicationsHasReactions

Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
id	INT	✓	✓							
date	DATETIME									
reactionId	INT		✓							
userId	INT		✓							
publicationId	INT		✓							

reactions

Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
id	INT	✓	✓							
name	VARCHAR(45)		✓							
path	VARCHAR(45)		✓							

report

Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
id	INT	✓	✓					✓		
text	TEXT		✓							
date	DATETIME									
pdf	TEXT		✓							archivo con pruebas
informeId	INT		✓							usuario que reporta
publicationId	INT		✓							publicación reportada

resetPassword

Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
id	INT	✓	✓					✓		
code	VARCHAR(45)		✓							
userId	INT		✓							

role

Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
id	INT	✓	✓					✓		
name	VARCHAR(45)		✓							

roleHasPermissions

Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
roleId	INT	✓	✓							
permissionsId	INT	✓	✓							

room

Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
id	INT	✓	✓					✓		
reference	VARCHAR(45)		✓							
price	VARCHAR(45)		✓							
description	TEXT		✓							
rating	TINYINT									
status	ENUM('AVAILABLE', 'UNAVAILABLE')		✓						'AVAILABLE'	
owner_id	INT		✓							

roomHasRequest

Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
id	INT	✓	✓					✓		
userId	INT		✓							
room_id	INT		✓							
date	DATETIME		✓							
status	ENUM('PENDING', 'APPROVED', 'REJECTED')		✓						'PENDING'	

roomHasUser

Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
roomId	INT	✓	✓							
userId	INT	✓	✓							
occupied	TINYINT		✓							

surveys

Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
id	INT	✓	✓							
name	VARCHAR(45)									

user

Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
id	INT	✓	✓							
firstName	VARCHAR(45)		✓							
secondName	VARCHAR(45)		✓							
surname	VARCHAR(45)		✓							
secondSurname	VARCHAR(45)		✓							
cellphone	VARCHAR(45)		✓							
gender	ENUM('MALE', 'FEMALE')									
email	VARCHAR(45)		✓							
password	VARCHAR(45)		✓							
pathPhoto	VARCHAR(45)									
verifiedAt	DATETIME									
role_id	INT		✓							
city_id	INT		✓							
pollAnswer_id	INT		✓							

userHasSurveys

Column name	Data Type	PK	NN	UQ	BIN	UN	ZF	AI	Default	Comment
userId	INT	✓	✓							
surveysId	INT	✓	✓							