



TUMANGA

TFG 2022

Trabajo realizado por Adrián Arnaiz Fernández

DAW

tumanga.arnaizdev.com

INDEX

1. Introducción
2. Objetivos
3. Tecnologías
4. Despliegue
5. BBDD
 - a. Colecciones
6. Backend
7. Frontend
8. Pasos
9. Funcionalidades
10. Anexo

URL

<https://tumanga.arnaizdev.com>

1. Introducción

El objetivo de este proyecto es crear una aplicación lo más realista posible sobre una tienda de mangas online. Practicar, aprender y poner en marcha todos los conocimientos necesarios que encontré para poder hacer la página lo más profesional posible, simulando que esa aplicación se va a desplegar en una empresa de verdad.

2. Objetivos

- Comprar artículos de una manera lo más rápida posible
- Ver la información de tu cuenta lo más detallada posible
- Vender los artículos fácilmente en el lado de la empresa

3. Tecnologías

Para este proyecto, he usado estas tecnologías:

Backend	Frontend	BBDD
NodeJS	ReactJS	MongoDB

4. Despliegue

En este paso use:

Backend	Frontend	BBDD
Render	Netfily	MongoDB Atlas

Estuve pensando en desplegarlo en AWS pero al final decidí no hacerlo ya que fui subiendo las partes individualmente y estas webs serán las que más usare en el futuro.

5. Base de datos

Como mencione anteriormente, he usado MongoDB y MongoDB Atlas. Este es una bbdd no relacional. Esta estructura esta formada por JSON.

El nombre de la bbdd es TuManga.

5.1. Colecciones

Direcciones

```
● ● ● Uploaded using RayThis Extension

var Address = new Schema({
  _id : {
    type : Schema.Types.ObjectId,
    auto : true
  },
  user_id : {
    type: Schema.Types.ObjectId,
    ref : "User",
    required : true,
    unique : false
  },
  name : {
    type : String,
    unique : false,
    required: true
  },
  number : {
    type : Number,
    required : false,
    unique : false
  },
  floor : {
    type : Number,
    required : false,
    unique : false
  },
  name_person : {
    type : String,
    required : true,
    unique : false
  },
  location : {
    type : String,
    unique : false,
    required : true
  },
  telephone: {
    type: Number,
    unique: false,
    required: true
  },
  created_date: {
    type: Date,
    default: Date.now
  }
});
```

Facturación

Uploaded using RayThis Extension

```
var Billing = new Schema({
  _id : {
    type : Schema.Types.ObjectId,
    auto : true
  },
  user_id : {
    type : Schema.Types.ObjectId,
    ref : "User",
    required : true,
    unique : false
  },
  card_name : {
    type : String,
    required : true,
    unique : false
  },
  expiration_date : {
    type : Date,
    required : true,
    unique : false
  },
  last_4_digits : {
    type : Number,
    unique : false,
    required : true
  },
  encrypt_card : {
    type: String,
    unique : false,
    required: true
  },
  type : {
    type: String,
    unique: false,
    required: true
  },
  image : {
    type: String,
    unique: false,
    required : true
  },
  created_date : {
    type: Date,
    default: Date.now
  }
});
```

Comentarios

```
Uploaded using RayThis Extension

var Comment = new Schema({
  _id : {
    type: Schema.Types.ObjectId,
    auto : true
  },
  user_id : {
    type : Schema.Types.ObjectId,
    ref : "User",
    required : true,
    unique : false
  },
  message : {
    type : String,
    required: true,
    unique : false
  },
  product_id : {
    type : Schema.Types.ObjectId,
    ref : "Product",
    required : true,
    unique : false
  },
  score : {
    type : Number,
    unique : false,
    required : true
  },
  date : {
    type: Date,
    default : Date.now
  },
  name: {
    type: String,
    required: true,
    unique: false
  },
  product_name : {
    type: String,
    required: true,
    unique: false
  }
});
```

Pedidos

Uploaded using RayThis Extension

```
var ItemCart = new Schema({
  product_id: {
    type: Schema.Types.ObjectId,
    ref: "Product",
    unique: false,
    required: false
  },
  quantity : {
    type: Number,
    min: 1,
    required: false,
    unique: false
  },
  price : {
    type: Number,
    required: false,
    unique: false
  },
  name: {
    type: String,
    required: false,
    unique: false
  },
  image : {
    type: String,
    required: false,
    unique: false
  },
  total_price: {
    type: Number,
    required: false,
    unique: false
  }
})
```



```
var Order = new Schema({
  _id: {
    type: Schema.Types.ObjectId,
    auto: true
  },
  id_client: {
    type: Schema.Types.ObjectId,
    ref: "User",
    required: true,
    unique: false
  },
  order_date: {
    type: Date,
    default: Date.now
  },
  send_date: {
    type: Date,
    unique: false
  },
  products: [ItemCart],
  /* products: [
    {
      type: Schema.Types.ObjectId,
      ref: "Product",
      unique: false,
      required: true
    }
  ], */
  delivery_address: {
    type: Schema.Types.ObjectId,
    ref: "Address",
    unique: false,
    required: false
  },
  billing: {
    type: Schema.Types.ObjectId,
    ref: "Billing",
    unique: false,
    required: false
  },
  state: {
    type: String,
    unique: false,
    required: true
  },
  total: {
    type: Number,
    required: true,
    unique: false
  }
});
```

Producto

Uploaded using RayThis Extension

```
var Product = new Schema({
  _id: {
    type: Schema.Types.ObjectId,
    auto: true
  },
  name: {
    type: String,
    unique: false,
    required: true
  },
  price: {
    type: Number,
    unique: false,
    required: true
  },
  description: {
    type: String,
    unique: false,
    required: true
  },
  short_description: {
    type: String,
    unique: false,
    required: true
  },
  upload_date: {
    type: Date,
    default: Date.now
  },
  state: {
    type: String,
    unique: false,
    required: true
  },
  stock: {
    type: Number,
    unique: false,
    required: true
  },
  categories: [String],
```

```
type: {
  type: String,
  unique: false,
  required: true
},
image: {
  type: String,
  unique: false,
  required: false,
  default: null
},
number_sales: {
  type: Number,
  unique: false,
  required: true
},
authors: {
  type: String,
  unique: false,
  required: false
},
editorial: {
  type: String,
  unique: false,
  required: false
},
series: {
  type: String,
  unique: false,
  required: false
},
comments: [{
  type: Schema.Types.ObjectId,
  ref: "Comment",
  unique: false,
  required: false
}]
});
```

Roles

Uploaded using RayThis Extension

```
var Role = new Schema({
  _id : {
    type: Schema.Types.ObjectId,
    auto : true
  },
  name : {
    type : String,
    required : true,
    unique : false
  }
});
```

Usuarios

```
var User = new Schema({
  _id : {
    type: Schema.Types.ObjectId,
    auto : true
  },
  name : {
    type : String,
    unique : false,
    required : false
  },
  last_name : {
    type : String,
    unique : false,
    required : false
  },
  email : {
    type : String,
    unique : true,
    required : true
  },
  password_hash : {
    type : String,
    unique : false,
    required : true
  },
  register_date : {
    type : Date,
    default : Date.now
  },
  state : {
    type : String,
    unique : false,
    required : true,
    default : "Active"
  },
},
```

User2

```
billing : [
  {
    type : Schema.Types.ObjectId,
    ref : "Billing",
    unique : false,
    required : true
  }
],
cart : [
  {
    type : Schema.Types.ObjectId,
    ref : "Orders",
    unique : false,
    required : true
  }
],
address : [
  {
    type : Schema.Types.ObjectId,
    ref : "Address",
    unique : false,
    required : true
  }
],
role : {
  type : Schema.Types.ObjectId,
  ref : "Role",
  unique : false,
  required: true
},
comments : [
  {
    type : Schema.Types.ObjectId,
    ref : "Comment",
    unique: false,
    required : true
  }
],
tokenRecover: {
  type: String,
  required: false,
  unique: false
}
});
```

