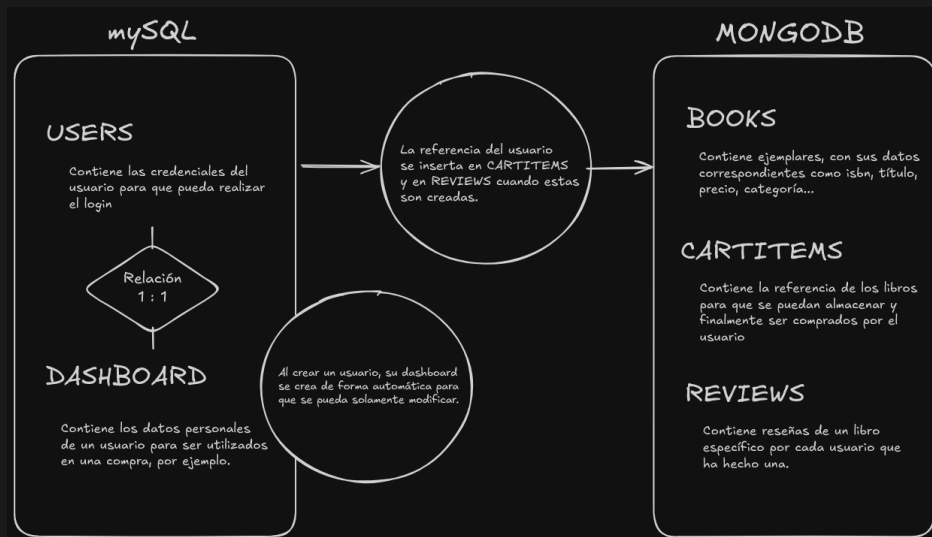


# CULT OF THE READER

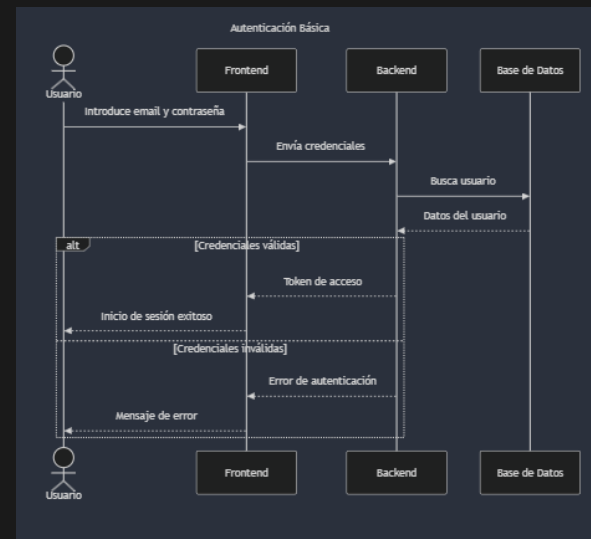
## SERVER-SIDE

### BASE DE DATOS



- Users: guarda las credenciales de los usuarios
- Dashboard: guarda los datos personales de los usuarios
- Books: contiene todos los libros con sus respectivos datos
- CartItems: guarda los ejemplares a comprar por el usuario
- Reviews: guarda las reseñas de cada libro hechas por los usuarios

### FLUJO DE AUTENTICACIÓN



- El usuario escribe su email y contraseña.
- El frontend envía esos datos al backend.
- El backend consulta en la base de datos si el usuario existe.
- La base de datos responde con la información del usuario
- Si las credenciales coinciden: Se permite el acceso
- Si las credenciales no coinciden: Se muestra error.

## ESTRUCTURA

```
server/
├── config/
│   ├── logger.js           # Logging configuration
│   ├── mongo.connection.js # MongoDB connection config
│   └── mysql.connection.js  # MySQL connection config
├── controllers/
│   ├── auth.controller.js   # Authentication controller
│   ├── book.controller.js   # Book operations controller
│   ├── cartitem.controller.js # Cart items controller
│   ├── dashboard.controller.js # Dashboard controller
│   └── review.controller.js  # Reviews controller
├── docs/
│   └── jest
│       └── book.model.test.js # Jest tester for CRUD book
collection
├── jmeter
│   ├── auth_test.jmx        # Template JMeter
│   └── auth.csv              # Template for the JMeter auth test
├── diagram_UML.png          # Simple UML Diagram
├── ER_DDBB.png              # Database flow
├── secuencialLogin.jpg      # Basic Diagram Sequence Login
├── middlewares/
│   └── auth.middleware.js    # Authentication middleware
├── models/
│   ├── book.model.js        # Book data model
│   ├── cartitem.model.js    # Cart item data model
│   ├── dashboard.model.js   # Dashboard data model
│   ├── review.model.js      # Review data model
│   ├── sync.model.js        # Synchronization model
│   └── user.model.js        # User data model
├── mongoDbsCollection/
│   └── cultofreader.books.json # Books collection for MongoDB
├── routes/
│   ├── auth.js              # Authentication routes
│   ├── book.js              # Book-related routes
│   └── cartitem.js           # Cart item routes
├── .env                      # Environment variables
├── .gitignore                # Git ignore configuration
├── coffeconfig.md            # Configuration documentation
├── package-lock.json         # Detailed dependencies
├── package.json              # Project configuration
├── README.md                 # Project documentation
├── server.js                 # Main server file
└── swagger.js                # Swagger API configuration
```



# CLIENT-SIDE