

Se crearon un total de 7 apis:

- login - api/login - POST

Los parámetros que espera son el usuario y el password :

```
{
  "usuario": "usr",
  "password": "pass"
}
```

Y la respuesta es la siguiente

```
{
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJodHRwOi8vMTI3LjAuMCA4xOjgwMDAvYXBpL2xvZ2luIiwiaWF0IjoxNjY3NzY4MDM1LCJleHAiOjE2Njc3NzE2MzUsIm5iZiI6MTY2Nzc2ODAzNSwianRpIjoibjhhbnVlbiNHMzcjZudlJrdCIsInN1YiI6IjEiLCJwcnYiOiIyM2JkNWM4OTQ5ZjYwMGFkYjM5ZTcwMWM0MDA4NzJkYjdhNTk3NmY3In0.BnewIID4Xu9EqMS2hTOLrt9rSk40DmfGsmPwDJyV0uc",
  "user": {
    "id": 1,
    "nombre": "rob",
    "usuario": "robe",
    "created_at": "2022-11-06T18:42:43.000000Z",
    "updated_at": "2022-11-06T18:42:43.000000Z"
  }
}
```

- registro - api/registro - POST

Los parámetros que espera son el nombre, usuario y password:

```
{
  "nombre": "rob",
  "usuario": "onts",
  "password": "1234567"
}
```

Y la respuesta es la siguiente :

```
{
  "message": "Usuario creado.",
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJodHRwOi8vMTI3LjAuMCA4xOjgwMDAvYXBpL3JlZ2lzdHJvIiwiaWF0IjoxNjY3NzY4MDM1LCJleHAiOjE2Njc3NzE2MzUsIm5iZiI6MTY2Nzc3NTM1NSwianRpIjoibjhhbnVlbiNHMzcjZudlJrdCIsInN1YiI6IjEiLCJwcnYiOiIyM2JkNWM4OTQ5ZjYwMGFkYjM5ZTcwMWM0MDA4NzJkYjdhNTk3NmY3In0.BnewIID4Xu9EqMS2hTOLrt9rSk40DmfGsmPwDJyV0uc",
  "user": {
    "id": 1,
    "nombre": "rob",
    "usuario": "onts",
    "password": "1234567",
    "created_at": "2022-11-06T18:42:43.000000Z",
    "updated_at": "2022-11-06T18:42:43.000000Z"
  }
}
```

```

    "nombre": "rob",
    "usuario": "etg",
    "updated_at": "2022-11-06T22:55:55.000000Z",
    "created_at": "2022-11-06T22:55:55.000000Z",
    "id": 12
  }
}

```

- **logout – api/logout - POST**
- **guardar - api/guardar - POST**
Espera 3 parametros articulo,descripción, id_usuario

```

{
  "articulo" : "prueba",
  "descripcion": "descripcion prueba",
  "id_usuario": 1
}

```

Y su respuesta es la siguiente

```

{
  "message": "Bien creado",
  "user": {
    "id": 1,
    "nombre": "rob",
    "usuario": "robe",
    "created_at": "2022-11-06T18:42:43.000000Z",
    "updated_at": "2022-11-06T18:42:43.000000Z"
  }
}

```

- **actualizar – api/actualizar/{id} – PUT**
Espera 4 parametros el primero seria el id del bien el cual se actualizara y los demás serian el articulo,descripción y el id del usuario

```

{
  "articulo" : "prueba",
  "descripcion": "descripcion prueba",
  "id_usuario": 1
}

```

Y la respuesta seria la siguiente

```

{
  "message": "Bien actualizado",
  "user": {
    "id": 1,
    "nombre": "rob",

```

```

        "usuario": "robe",
        "created_at": "2022-11-06T18:42:43.000000Z",
        "updated_at": "2022-11-06T18:42:43.000000Z"
    }
}

```

- eliminar – api/eliminar/{id} – DELETE

Solo espera el id del articulo y la respuesta es la siguiente

```
{"message": "Bien eliminado"}
```

- detalle - api/detalle?id={ids} – GET

Espera los ids ya se uno o varios, se dividen por comas(,) y la respuesta es un array con la información requerida

```

{
  "detalle": [
    {
      "id": 4,
      "articulo": "prueba",
      "descripcion": "descripcion prueba",
      "id_usuario": 1
    },
    {
      "id": 5,
      "articulo": "registro 5",
      "descripcion": "descripcion",
      "id_usuario": 1
    }
  ]
}

```

Para la base de datos, solo es necesario crear la bd en este caso la llame **nextia**, para crear las tablas solo es necesario correr las migraciones con las que cuenta laravel

```
php artisan migrate --path=/database/migrations/2014_10_12_000000_create_users_table.php
```

```
php artisan migrate --path=/database/migrations/2022_11_06_011240_create_bienes_table.php
```

se tienen que correr en ese orden para evitar conflictos