

- 1) MOMENTO 1 PASO 8: ¿Qué paso con el directorio db?
- 2) MOMENTO 2 PASO 6: ¿Se guardo lo importante?
- 3) MOMENTO 4 PASO 1: ¿Esta bien el nombre del commit?
- 4) MOMENTO 5 PASO 3: ¿Qué hace este paso?
- 5) MOMENTO 5 PASO 6: ¿Qué pasó acá?

- 1- Se agrego al comit
2. Se guardo todo menos la modificacion de app.js.
3. Se guarda todo el proyecto.
4. Restaura la eliminacion de producto_db.js
5. Comitea el proyecto con el archivo config.js eliminado.

1. mkdir Ecommerce
2. cd Ecommerce
3. touch app.js
4. **touch config.js**
5. mkdir db
6. git init
7. git add .
8. git commit -m "Primer Commit"

1. cd db/
2. touch producto_db.js
3. touch carrito_db.js
4. cd ..
5. se modifica el archivo app.js
6. git commit -m "Guardando lo importante"

1. `git add .`

1. git commit -m "Guardando una parte del proyecto"

1. cd db
2. rm producto_db.js
3. git restore .
4. rm ../config.js
5. git add .
6. git commit -m "Sacando la configuración"

1-Escritorio
3- Ecommerce

____2-db

5 Ecommerce

U	Untracked
A	Added
M	Modified
D	Deleted

controladores

rutas

db

E-Commerce

```

app.js
productos_controller.js
config.js
carritos_controller.js
producto_db.js
carrito_db.js

```

PLUS

MOMENTO 1

MOMENTO 2

MOMENTO 3

MOMENTO 4

MOMENTO 5

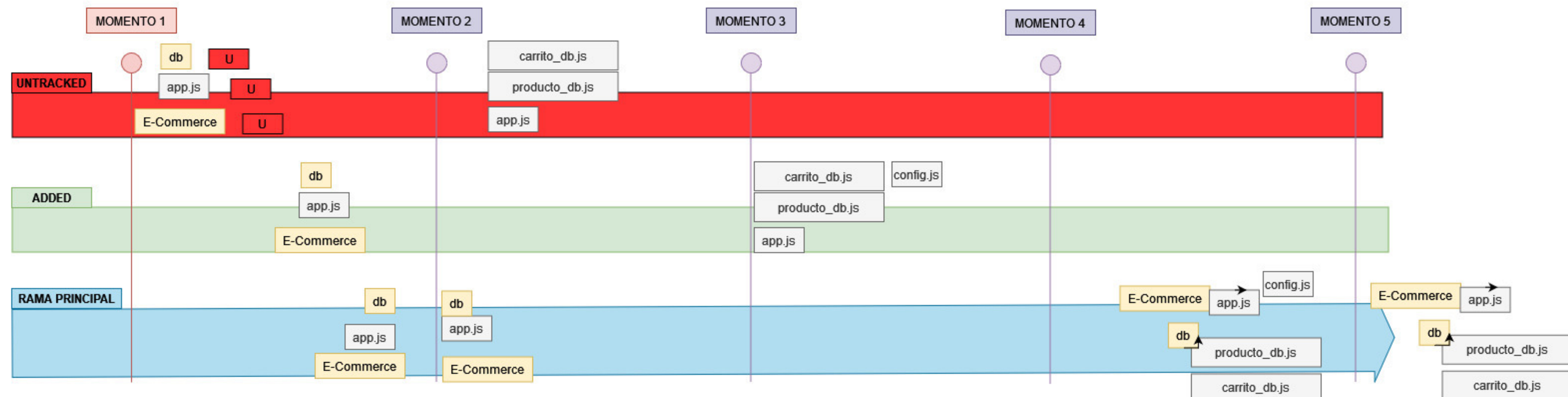
MOMENTO 1

MOMENTO 2

MOMENTO 3

MOMENTO 4

MOMENTO 5



```
graph TD
    ECommerce[E-Commerce]
    appjs[app.js]
    configjs[config.js]
    db[(db)]
    ECommerce --- appjs
    ECommerce --- configjs
    ECommerce --- db
```

```

graph TD
    E-commerce --> app.js
    app.js --> config.js
    config.js --> producto_db.js
    config.js --> carrito_db.js
    producto_db.js --> db
    carrito_db.js --> db
  
```

```

graph TD
    ECommerce[E-Commerce] --> app_js[app.js]
    app_js --> config_js[config.js]
    config_js --> producto_db_js[producto_db.js]
    producto_db_js --> carrito_db_js[carrito_db.js]
    app_js --> db[db]
    config_js --> db
    producto_db_js --> db
    carrito_db_js --> db
  
```

```
graph TD; ECommerce[E-Commerce] --> appjs[app.js]; appjs --- configjs[config.js]; db[db] --> productodbjs[producto_db.js]; db --> carritodbjs[carrito_db.js];
```

The diagram illustrates the relationship between the E-Commerce application and its database components. The E-Commerce application (yellow box) is connected to the app.js file (white box), which in turn is connected to the config.js file (white box). The db (yellow box) is connected to the producto_db.js (white box) and the carrito_db.js (white box).

```

graph TD
    ECommerce[E-commerce] --> appjs[app.js]
    appjs --> configjs[config.js]
    configjs --> D[D]
    D --> productodbjs[producto_db.js]
    productodbjs --> db[db]
    productodbjs --> carrito_dbjs[carrito_db.js]
  
```

The diagram illustrates the structure of an E-commerce application. It shows a hierarchy of dependencies: 'E-commerce' depends on 'app.js', which depends on 'config.js', which in turn depends on 'D'. 'D' depends on 'producto_db.js', which depends on both 'db' and 'carrito_db.js'.