

MEMORIA PROYECTO

‘GRUPO 1’

Fecha Entrega: 06/03/2020

Integrantes:

Manuel Esquer Cerezo

Ana María Aguilar García

1. - Memoria Descriptiva

1.1 Repositorio

- REPOSITORIO: - <https://github.com/FormadoresEoi/bancaonline>
- RAMA DEVELOP: - **g1_develop**
 - RAMA MASTER: - **g1_master**

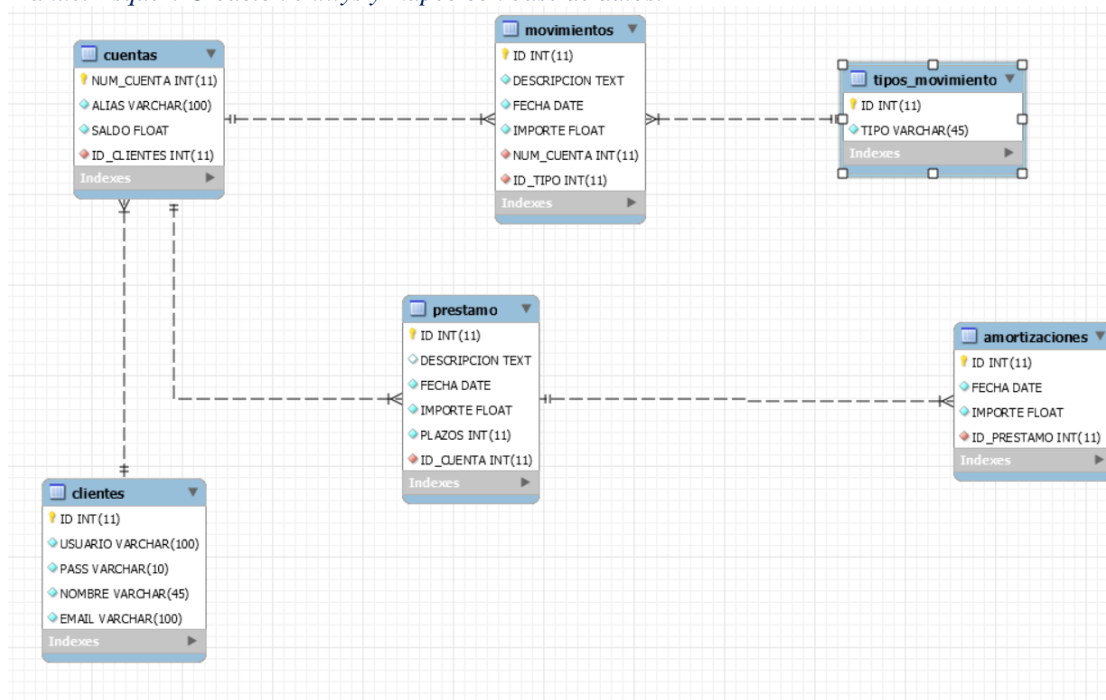
1.2 Roles:

Los roles que se han establecido han sido adaptados a las situaciones del proyecto en cada situación y de una manera equitativa. Manuel Esquer ha dirigido en ciertas ocasiones a la hora de realizar ciertas acciones y donde enfocarnos en distintas situaciones del proyecto.

1.3 Agenda:

DÍA 1 (02/03/2020)

- Ana María: Construcción SCRIPTS DDL en MySQL con Workbench
- Manuel Esquer: Creación entitys y mapeo con base de datos.



DÍA 2 (03/03/2020)

- Ana Maria : Función buscar cuentas de clientes
- ALUMNO 2: CRUD cuentas y clientes y alguna función más de cliente

DÍA 3 (04/03/2020)

- Ana María : Función log-in clientes y comienzo de crear prestamos
- Manuel Esquer: Funciones get deudoras, movimientos y prestamos

DÍA 4 (05/03/2020)

- Ana María : Función crear prestamos acabada
- Manuel Esquer: Funciones prestamos Vivos y Amortizados y crear ingresos y pagos

DÍA 5 (06/03/2020)

- Ana María : Arreglar conflictos y comenzar Amortizaciones Diarias
- Manuel Esquer: Comprobación y corrección de las funciones del código y memoria

1.4 **Investigación**

- *Resolución problemas de identificación en Git. Para solucionar este problema ha sido necesario eliminar las credenciales almacenadas de Windows. Además hemos aprendido a utilizar Git con distintas herramientas como GitKraken, Tortoise, Consola de Comandos y Eclipse.*
- *Buscar cómo realizar un Mapper de listas de objetos.*
- *Solucionar problema de entity no serializable.*
- *Como pasar un date sin horas minutos y segundos*

1.5 **Objetivo y Resultado**

Todas las funcionalidades de cliente controller y cuenta controller han sido desarrolladas menos la funcionalidad de ejecutarAmortizacionesDiarias. Tampoco se ha realizado ninguna de las funcionalidades de Reports Controller

1.6 **Conclusión**

El proyecto tiene una buena dificultad y complejidad las cuales han puesto a prueba nuestras habilidades y capacidades resolutivas. Consideramos que su extensión es más larga de la que se puede desarrollar en el tiempo estimado pero ha sido muy motivador y divertido realizar este proyecto.

2. - Memoria Técnica

2.1 *Directorio de Entrega:*

Contenido de la entrega:

- *Memoria Proyecto G1.pdf*
- *BancoBBDD.sql*
- *GRUPO_X_Source_Develop.zip*
- *GRUPO_X_Source_Master.zip*

2.2 **Documentos Generados**

Ninguno.

2.3 **API REST**

2.3.1 Servicio Web – Obtención de préstamo por código.

CLIENTE CONTROLLER

- **POST – Crear Cliente**

Entrada:

POST /clientes crearCliente

Parameters Cancel

Name	Description
dto required (body)	dto Example Value Model <pre>{ "email": "javi@gmail.com", "nombre": "Javi", "pass": "pass", "usuario": "Salpichon" }</pre>

Salida:

Server response

Code	Details
200	Response body <pre>{ "id": 3, "usuario": "Salpichon", "nombre": "Javi", "email": "javi@gmail.com" }</pre> Download

- **GET –Clientes**

Entrada:

cliente-controller Cliente Controller

GET /clientes mostrarCliente

Parameters Cancel

No parameters

Execute Clear

Salida:

Server response

Code	Details
200	<p>Response body</p> <pre>[{ "id": 1, "usuario": "Manuel", "nombre": "Manuel", "email": "m@gmail.com" }, { "id": 2, "usuario": "Ana Maria", "nombre": "Ana", "email": "a@gmail.com" }, { "id": 3, "usuario": "Salpichon", "nombre": "Javi", "email": "javi@gmail.com" }]</pre> <p>Download</p>

- **GET – Cliente por id**
Entrada:

GET /clientes/{id} selectCliente

Parameters Cancel

Name	Description
id * required integer(\$int32) (path)	id <input type="text" value="3"/>

Execute

Salida:

Code	Details
200	<p>Response body</p> <pre>{ "id": 3, "usuario": "Salpichon", "nombre": "Javi", "email": "javi@gmail.com" }</pre> <p>Download</p>

- **POST – Login**
Entrada:

POST /clientes/login findByUsuarioAndPass

Parameters Cancel

Name	Description				
login * required (body)	<p>login</p> <table border="1"> <thead> <tr> <th>Example Value</th> <th>Model</th> </tr> </thead> <tbody> <tr> <td>{ "pass": "pass", "usuario": "Ana Maria" }</td> <td></td> </tr> </tbody> </table>	Example Value	Model	{ "pass": "pass", "usuario": "Ana Maria" }	
Example Value	Model				
{ "pass": "pass", "usuario": "Ana Maria" }					

Salida:

Code	Details
200	<p>Response body</p> <pre>{ "id": 2, "usuario": "Ana Maria", "nombre": "Ana", "email": "a@gmail.com" }</pre> <p>Download</p>

- **GET – Cuentas por Cliente**
Entrada:

GET /clientes/{id}/cuentas findAllById_Clientes

Parameters Cancel

Name	Description
id * required integer(\$int32) (path)	id <input type="text" value="1"/>

Salida:

Server response

Code	Details
200	<p>Response body</p> <pre>[{ "alias": "caprichos", "saldo": 951, "id_cliente": 0 }, { "alias": "chiquillos", "saldo": 20, "id_cliente": 0 }]</pre> <p>Download</p>

- **UPDATE – Update email Cliente**

Entrada:

PUT /clientes/update/{id} updateCliente

Parameters Cancel

Name	Description
email * required string (query)	email <input type="text" value="manuel@gmail.com"/>
id * required integer(\$int32) (path)	id <input type="text" value="1"/>

Salida:

Server response

Code	Details
200	<p>Response body</p> <pre>{ "id": 1, "usuario": "Manuel", "nombre": "Manuel", "email": "manuel@gmail.com" }</pre> <p>Download</p>

CUENTAS

- **POST – Crear Cuentas**

Entrada:

POST /cuentas crearCuenta

Parameters Cancel

Name	Description
dto * required (body)	<p>dto</p> <p>Example Value Model</p> <pre>{ "alias": "Vivir", "id_cliente": 2, "saldo": 15000 }</pre>

Salida:

Code	Details
200	<p>Response body</p> <pre>{ "num_cuenta": 5, "alias": "Vivir", "saldo": 15000, "cliente": { "id": 2, "usuario": "Ana Maria", "nombre": "Ana", "email": "a@gmail.com" } }</pre> <p>Download</p>

- **GET – Cuentas**
Entrada:

GET	/cuentas mostrarCuentas
Parameters	
No parameters	
<div>Execute</div> <div>Clear</div>	

Salida:

Code	Details
200	<p>Response body</p> <pre>[{ "num_cuenta": 1, "alias": "caprichos", "saldo": 951, "cliente": { "id": 1, "usuario": "Manuel", "nombre": "Manuel", "email": "manuel@gmail.com" } }, { "num_cuenta": 2, "alias": "chiquillos", "saldo": 20, "cliente": { "id": 1, "usuario": "Manuel", "nombre": "Manuel", "email": "manuel@gmail.com" } }, { "num_cuenta": 4, "alias": "Comida", </pre> <p>Download</p>

- **GET – Buscar Cuenta por id**
Entrada:

GET	/cuentas/{id} buscarCuenta				
Parameters					
<table border="1"> <thead> <tr> <th>Name</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>id * required integer(\$int32) (path)</td> <td>id</td> </tr> </tbody> </table>		Name	Description	id * required integer(\$int32) (path)	id
Name	Description				
id * required integer(\$int32) (path)	id				
<div>Execute</div> <div>Clear</div>					

Salida:

Code	Details
200	<p>Response body</p> <pre>{ "num_cuenta": 1, "alias": "caprichos", "saldo": 951, "cliente": { "id": 1, "usuario": "Manuel", "nombre": "Manuel", "email": "manuel@gmail.com" } }</pre> <p>Download</p> <p>Response headers</p>

- **GET – Cuentas deudoras**

Entrada:

GET /cuentas/deudoras buscarCuentasDeudoras

Parameters Cancel

No parameters

Execute Clear

Salida:

Server response

Code	Details
200	<p>Response body</p> <pre>[{ "num_cuenta": 6, "alias": "Lol", "saldo": -150, "cliente": { "id": 3, "usuario": "Salpichon", "nombre": "Javi", "email": "javi@gmail.com" } }]</pre> <p>Download</p>

- **PUT – Update Alias**

Entrada:

PUT /cuentas/update/{id} updateCuenta

Parameters Cancel

Name	Description
alias * required string (query)	alias <input type="text" value="Universidad"/>
id * required integer(\$int32) (path)	id <input type="text" value="2"/>

Execute Clear

Salida:

Server response

Code	Details
200	<p>Response body</p> <pre>{ "num_cuenta": 2, "alias": "Universidad", "saldo": 20, "cliente": { "id": 1, "usuario": "Manuel", "nombre": "Manuel", "email": "manuel@gmail.com" } }</pre> <p>Download</p>

- **GET – Movimiento por id**

Entrada:

GET /cuentas/{id}/movimientos buscarMovimientosCuenta

Parameters Cancel

Name	Description
id * required integer(\$int32) (path)	id <input type="text" value="1"/>

Execute Clear

Salida:

Server response

Code	Details
200	<p>Response body</p> <pre>[{ "id": 1, "descripcion": "string", "fecha": "2020-03-06T00:00:00.000+0000", "importe": 500, "tiposmovimiento": { "id": 2, "tipo": "PAGO" } }, { "id": 2, "descripcion": "string", "fecha": "2020-03-06T00:00:00.000+0000", "importe": 500, "tiposmovimiento": { "id": 2, "tipo": "PAGO" } }]</pre>

• POST – Crear Prestamo

Entrada:

POST /cuentas/{id}/prestamos CrearPrestamo

Parameters Cancel

Name	Description
dto * required (body)	dto Example Value Model <pre>{ "descripcion": "Casa", "importe": 100000, "plazos": 10 }</pre>

Salida:

Server response

Code	Details
200	<p>Response body</p> <pre>{ "descripcion": "Casa", "importe": 100000, "plazos": 10 }</pre> Download

• GET – Prestamos Vivos

Entrada:

GET /cuentas/{id}/prestamosVivos buscarPrestamosVivos

Parameters Cancel

Name	Description
id * required integer(\$int32) (path)	id <input type="text" value="4"/>

Execute Clear

Salida:

Server response

Code	Details
200	<p>Response body</p> <pre>[{ "id": 8, "descripcion": "coche", "fecha": "2020-03-05", "importe": 10000, "plazos": 2, "listAmort": [{ "id": 22, "fecha": "2020-04-05T00:00:00.000+0000", "importe": 5000 }, { "id": 23, "fecha": "2020-05-05T00:00:00.000+0000", "importe": 5000 }] }]</pre> <p>Download</p>

- **GET – Prestamos Amortizados**

Entrada:

GET /cuentas/{id}/prestamosAmortizados buscarPrestamosAmortizados

Parameters Cancel

Name	Description
id * required integer(\$int32) (path)	id <input type="text" value="1"/>

Execute Clear

Salida:

Server response

Code	Details
200	<p>Response body</p> <pre>[{ "id": 1, "descripcion": "1", "fecha": "0111-11-11", "importe": 1, "plazos": 1, "listAmort": [{ "id": 1, "fecha": "2010-11-10T00:00:00.000+0000", "importe": 200 }] }, { "id": 2, "descripcion": "string", "fecha": "2010-10-10", "importe": 10, "plazos": 2, "listAmort": [] }, { "id": 3, "descripcion": "string", "fecha": "2010-11-11", "importe": 10, "plazos": 2, "listAmort": [] }]</pre> <p>Download</p>

- **POST – Crear Ingreso**

Entrada:

POST /cuentas/{id}/ingresos crearingreso

Parameters Cancel

Name	Description
id * required integer(\$int32) (path)	id <input type="text" value="1"/>
ingresodto * required (body)	ingresodto Example Value Model <pre>{ "description": "pension", "importe": 150 }</pre>

Salida:

Server response

Code	Details
200	<p>Response body</p> <pre>{ "id": 8, "description": "pension", "fecha": "2020-03-05T23:00:00.000+0000", "importe": 150, "tiposmovimiento": { "id": 1, "tipo": "INGRESO" } }</pre> Download

- **POST – Crear pago**

Entrada:

POST /cuentas/{id}/pagos crearPagos

Parameters Cancel

Name	Description
id * required integer(\$int32) (path)	id <input type="text" value="1"/>
pagodto * required (body)	pagodto Example Value Model <pre>{ "description": "pago", "importe": 200 }</pre>

Salida:

Server response

Code	Details
200	<p>Response body</p> <pre>{ "id": 9, "description": "pago", "fecha": "2020-03-05T23:00:00.000+0000", "importe": 200, "tiposmovimiento": { "id": 2, "tipo": "PAGO" } }</pre> Download

2.3.2 Servicio Web N