

Memoria Práctica Integradora – Grupo 2

Autores: GRUPO 2

- Vicente Álvarez
- Diego Recio
- Alicia San Julián

Tiempo empleado en el desarrollo:

- Vicente: horquilla de 150-160 horas (JPA, repositorios y Servicios asociados, principalmente)
- Diego: 140 horas (aproximadamente)
- Alicia: 135 horas

Diagrama de Gantt

Como parte de una de las exigencias de la práctica, hemos realizado un diagrama de Gantt. Hemos utilizado la funcionalidad de GitHub que permite realizar un diagrama de Gantt vinculado a los *issues* del repositorio de Git en el que se desarrolla. Aquí está el enlace a la web del diagrama:

<https://github.com/users/Koderonin/projects/2/views/4>

Entornos de desarrollo (IDE)

- IntelliJ Ultimate Edition
- Visual Studio Code

Navegadores en los que se ha probado

- Google Chrome
- Mozilla Firefox

Sistemas operativos utilizados

- MV Ubuntu virtualizada sobre Windows
- MV Ubuntu virtualizada sobre MAX
- WSL Linux sobre windows

Ruta de inicio de la aplicación

- Por el que se entra a la aplicación (no endpoint como tal, sino URL)
<http://apache.da2.dva/index.html>
- Primer endpoint de la parte de Tomcat (se accede a través del anterior)
<http://tomcat.da2.dva:8080/login/paso1>

Credenciales

Hay varios usuarios creados, con y sin clientes asociados. Aquellos que tienen, cada uno por su lado, todo completo son:

- Usuario administrador: admin@integradora.jp
Clave administrador: Administra1!

Memoria Práctica Integradora – Grupo 2

- Usuario cliente: cliente@integradora.jp
Clave cliente: Clientillo1!
- Usuario cliente 2: cliente2@integradora.jp
Clave cliente 2: Clientillo2!

Existe también un usuario cliente que no dispone de una cuenta de cliente vinculada a su usuario, y sus credenciales son las que siguen:

- Usuario cliente 3: cliente3@integradora.jp
Clave cliente 3: Clientillo3!

Instalación y otras dependencias

La aplicación requiere que en la máquina en la que se vaya a alojar la aplicación tenga instalado Docker Compose, en una versión igual o superior a la 3.3. Una vez instalado, mediante el siguiente comando de Bash ejecutado en el mismo directorio del archivo Docker-compose.yml, la aplicación se levanta y despliega automáticamente.

```
docker compose -f ./docker-compose.yml -p practicaintegradora up  
-d
```

En la configuración de la aplicación Tomcat viene incluida su relación con los contenedores de persistencia. Todo ataque a la base de datos se realiza a través de dicha aplicación.

En cualquier caso, si se desea uno conectar directamente a los contenedores para pedir datos:

contiene_mysql ip_acceso_cluster:3306 User: root Pass: root	contiene_mongo ip_acceso_cluster:27017 User: root Pass: root (se utiliza como authSource la db admin)
--	---

Memoria Práctica Integradora – Grupo 2

Métodos existentes en los controladores

LoginController	@RequestMapping("login")	@GetMapping("paso1") public ModelAndView login(HttpSession sesion)
		@PostMapping("paso1") public ModelAndView login(@RequestParam("usuario") String email, HttpSession sesion)
		@GetMapping("paso2") public ModelAndView clave(HttpSession sesion)
		@PostMapping("paso2") public ModelAndView clave(@RequestParam("clave") String clave, HttpSession sesion, @CookieValue(name = "accesosUsuarios", defaultValue = "none") String contenidoCookie, HttpServletResponse response)
RegistroUsuarioController	@RequestMapping("registro/usuario")	@GetMapping("/") public ModelAndView registroUsuario(@ModelAttribute("usuario") UsuarioCliente usuario)
		@PostMapping("/") public ModelAndView registroUsuario(@Valid @ModelAttribute("usuario") UsuarioCliente usuario, BindingResult resultado)

Memoria Práctica Integradora – Grupo 2

RegistroClienteController	@RequestMapping("registro/cliente")	@GetMapping("paso1") public ModelAndView registroCliente(@ModelAttribute("cliente") Cliente cliente, HttpSession sesion, BindingResult result)
		@PostMapping("paso1") public ModelAndView registrar(@Validated(DatosPersonales.class) @ModelAttribute("cliente") Cliente cliente, BindingResult resultado, HttpSession sesion)
		@GetMapping("paso2") public ModelAndView paso2(@ModelAttribute("cliente") Cliente cliente, HttpSession sesion, BindingResult result)
		@PostMapping("paso2") public ModelAndView registrar2(@Validated(DatosContacto.class) @ModelAttribute("cliente") Cliente cliente, BindingResult resultado, HttpSession sesion)
		@GetMapping("paso3") public ModelAndView paso3(@ModelAttribute("cliente") Cliente cliente, HttpSession sesion, BindingResult result)
		@PostMapping("paso3") public ModelAndView registrar3(@Validated(DatosCliente.class) @ModelAttribute("cliente") Cliente cliente, BindingResult resultado, HttpSession sesion)
		@GetMapping("paso4") public ModelAndView resumen(@ModelAttribute("cliente") Cliente cliente, HttpSession sesion, BindingResult result)
		@PostMapping("paso4") public ModelAndView paso4(HttpSession sesion)
AreaClienteController	@RequestMapping("/cliente")	@GetMapping("/area-cliente") public ModelAndView areaCliente(HttpSession sesion)

Memoria Práctica Integradora – Grupo 2

DireccionController	@RequestMapping("/cliente")	@GetMapping("/nueva-direccion") public ModelAndView agregarDireccion(@ModelAttribute("direccion") Direccion direccion, HttpSession sesion)
		@PostMapping("/nueva-direccion") public ModelAndView insertarTarjeta(@Valid @ModelAttribute("direccion") Direccion direccion, HttpSession sesion, BindingResult result)
TarjetaController	@RequestMapping("/cliente")	@GetMapping("/nueva-tarjeta") public ModelAndView agregarTarjeta(@ModelAttribute("tarjeta") TarjetaCredito tarjeta, HttpSession sesion)
		@PostMapping("/nueva-tarjeta") public ModelAndView insertarTarjeta(@Valid @ModelAttribute("tarjeta") TarjetaCredito tarjeta, HttpSession sesion, BindingResult resultado)
AdministracionController	@RequestMapping("/admin")	@GetMapping("/area-admin") public ModelAndView devuelveAreaAdmin()
AlmacenController	@RequestMapping("/admin/almacen")	@GetMapping("/add-producto") public ModelAndView addProducto(HttpSession sesion)
AdminClienteControllerREST	@RequestMapping("admin/api/cliente")	@GetMapping("listado_ampliado") public Object listadoClientes()
		@GetMapping("/{id}") public Cliente getCliente(@PathVariable("id") String id)
		@GetMapping("listado") public Object listadoClientesFiltrado()

Memoria Práctica Integradora – Grupo 2

		@GetMapping("/verify") public List<Object> verify(HttpSession sesion)
		@PostMapping(value = "/update", consumes = "application/json") public void actualizarCliente(@RequestBody Cliente cliente)
AdminProductoControllerREST	@RequestMapping("admin/api/producto")	@PostMapping(consumes = "multipart/form-data") public void createProducto(@RequestPart("formularioProducto") Document formularioProducto, @RequestPart(value = "imagenes", required = false) MultipartFile[] imagenes)
		@GetMapping("listado") public List<Document> listadoProyectos()
		@GetMapping("/detalle/{codigo}") public Document detalleProductoByCodigo(@PathVariable("codigo") String codigo)
		@PostMapping(value = "/update", consumes = "application/json") public void updateProduct(@RequestBody Document product)
AdminUsuarioControllerREST	@RequestMapping("admin/api/usuario")	@GetMapping("listado") public Object listadoUsuarios()
		@GetMapping("/{id}") public Usuario getCliente(@PathVariable("id") String id)
		@PostMapping(value = "/update", consumes = "application/json") public ResponseEntity<String> actualizarUsuarioCliente(@RequestBody UsuarioCliente usuarioActualizado)
ClienteControllerREST	@RequestMapping("/api/cliente")	@GetMapping("listado") public List<Cliente> listadoClientes()

Memoria Práctica Integradora – Grupo 2

ProductoControllerREST	@RequestMapping("/api/producto")	@GetMapping("listado") public List<Document> listadoProyectos()
		@GetMapping("/detalle/{codigo}") public Document detalleProductoByCodigo(@PathVariable("codigo") String codigo)
		@GetMapping("/{id}") public Document detalleProductoById(@PathVariable("id") String id)
SesionControllerREST	@RequestMapping("/api/sesion")	@GetMapping("/cliente") public Cliente detalleClienteByUsuario(HttpSession session)
		@GetMapping("/usuarioEsAdmin") public boolean usuarioEnSesionEsAdmin(HttpSession session)
		@GetMapping("/paginas-visitadas") public int paginasEnSesion(HttpSession session)
		@GetMapping("/agregar-visita") public void agregarVisita(HttpSession session)
InvalidateSesionController	@RequestMapping("logout")	@GetMapping("invalidate") public String logout(HttpSession session)
		@GetMapping("remove") public String logout2(HttpSession session)
AdminNominasControllerREST	@RequestMapping("admin/api/nomina")	@GetMapping("listado") public Object listadoClientes()
		@GetMapping("listado_completo") public Object listadoNominas()
		@GetMapping("/{id}") public List<Nomina> getNominasByCliente(@PathVariable("id") String id)

Memoria Práctica Integradora – Grupo 2

AdminDireccionControllerREST	@RequestMapping("admin/api/direccion")	@PostMapping(value = "direccion/update", consumes = "application/json") public void actualizarDireccion(@RequestBody Direccion direccion)
------------------------------	--	--