

# BiciUcm

**Daniel Fernández-Pacheco Santamaría**  
**Marcos Padilla Alonso**

**3B**

# Preparación de proyecto

En nuestro proyecto ,hemos utilizado mongoDB con php. Para que todo esté bien unido y las consultas con mongoDB en php funcionen y accedan/modifiquen la bd, hemos seguido una serie de procedimientos.

MongoDB: Nos lo descargamos junto a su extensión de mongoDBCompass y ejecutamos. El compass es una interfaz de mongo que nos servirá para ver los cambios creados en la bd de una forma más visual y ergonómica. Pero este trabajo se ha hecho con creación de colecciones y consultas como hemos visto en clase y permite hacer cambios en la aplicación desde la consola de mongo. Es importante quedarnos con la versión de la aplicación. Añadimos al path la carpeta donde hayamos guardado el mongo

# Preparación del proyecto

Xampp: Nos descargamos el xampp(en nuestro caso una versión avanzada la 8.0.5.). Ejecutamos y preparamos según los pasos que nos indica el propio programar. Posteriormente nos descargamos el driver más reciente de vuestra versión de mongo para php en esta pagina:<https://pecl.php.net/package/mongodb> .

Pulsamos en DLL de la versión *estable* dicha.

(Nosotros ac la 1.8.1 que era

la más reciente por aquel entonces, mi recomendacion seria que siguierais estos pasos ).

	and PHP, implementing only fundamental and performance-critical components necessary to build a fully-functional MongoDB driver.
Homepage	<a href="http://docs.mongodb.org/ecosystem/drivers/php/">http://docs.mongodb.org/ecosystem/drivers/php/</a>

[ Latest Tarball ]

[ Changelog ]

[ View Statistics ]

[ Browse Source ]

[ Package Bugs ]

[ View Documentation ]

Available Releases					
Version	State	Release Date	Downloads		
<a href="#">1.10.0alpha1</a>	alpha	2021-04-09	<a href="#">mongodb-1.10.0alpha1.tgz</a> (1293.7kB)	<a href="#">DLL</a>	[ Changelog ]
<a href="#">1.9.1</a>	stable	2021-04-07	<a href="#">mongodb-1.9.1.tgz</a> (1255.8kB)	<a href="#">DLL</a>	[ Changelog ]
<a href="#">1.9.0</a>	stable	2020-11-25	<a href="#">mongodb-1.9.0.tgz</a> (1269.9kB)	<a href="#">DLL</a>	[ Changelog ]
<a href="#">1.9.0RC1</a>	beta	2020-11-10	<a href="#">mongodb-1.9.0RC1.tgz</a> (1269.6kB)	<a href="#">DLL</a>	[ Changelog ]
<a href="#">1.8.2</a>	stable	2020-11-05	<a href="#">mongodb-1.8.2.tgz</a> (1262.3kB)	<a href="#">DLL</a>	[ Changelog ]
<a href="#">1.8.1</a>	stable	2020-10-06	<a href="#">mongodb-1.8.1.tgz</a> (1262.2kB)	<a href="#">DLL</a>	[ Changelog ]
<a href="#">1.8.0</a>	stable	2020-07-31	<a href="#">mongodb-1.8.0.tgz</a> (1262.6kB)	<a href="#">DLL</a>	[ Changelog ]
<a href="#">1.8.0RC1</a>	beta	2020-07-17	<a href="#">mongodb-1.8.0RC1.tgz</a> (1261.3kB)	<a href="#">DLL</a>	[ Changelog ]
<a href="#">1.7.5</a>	stable	2020-07-08	<a href="#">mongodb-1.7.5.tgz</a> (1195.2kB)	<a href="#">DLL</a>	[ Changelog ]
<a href="#">1.8.0beta2</a>	beta	2020-06-10	<a href="#">mongodb-1.8.0beta2.tgz</a> (1247.4kB)	<a href="#">DLL</a>	[ Changelog ]
<a href="#">1.8.0beta1</a>	beta	2020-04-14	<a href="#">mongodb-1.8.0beta1.tgz</a> (1223.4kB)	<a href="#">DLL</a>	[ Changelog ]
<a href="#">1.7.4</a>	stable	2020-03-11	<a href="#">mongodb-1.7.4.tgz</a> (1194.7kB)	<a href="#">DLL</a>	[ Changelog ]
<a href="#">1.7.3</a>	stable	2020-02-25	<a href="#">mongodb-1.7.3.tgz</a> (1194.7kB)	<a href="#">DLL</a>	[ Changelog ]
<a href="#">1.7.2</a>	stable	2020-02-13	<a href="#">mongodb-1.7.2.tgz</a> (1194.7kB)	<a href="#">DLL</a>	[ Changelog ]
<a href="#">1.7.1</a>	stable	2020-02-04	<a href="#">mongodb-1.7.1.tgz</a> (1194.6kB)	<a href="#">DLL</a>	[ Changelog ]

# Preparación de proyecto

Se os abrirá esta pantalla donde os descargais la versión más avanzada: 7.4 Thread Safe x64(depends su SO). La recomendación de la descarga de una versión avanzada de xampp viene porque al hacerla con la 7.4.4, en este paso no nos salía. Probar si con vuestra versión funciona pero con la 8.0 aseguramos que sí . Al desargarlo nos vamos a la carpeta ext /php/ xampp donde extraemos ahí el archivo “php\_mongodb.dll” de lo descargado previamente. Después buscamos en /php un archivo llamado “php.ini”, lo editamos y añadimos esta extensión: subrayada de la imagen y guardamos

```
; The MIBS data available in the PHP distribut:
; See http://www.php.net/manual/en/snmp.instal.
;extension=snmp

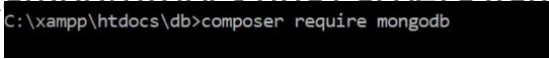
;extension=soap
;extension=sockets
;extension=sodium
;extension=sqllite3
;extension=tidy
;extension=xsl
extension=php_mongodb.dll

;zend_extension=opcache
```

DLL List	
PHP 7.4	<a href="#">7.4 Non Thread Safe (NTS) x64</a> <a href="#">7.4 Thread Safe (TS) x64</a> <a href="#">7.4 Non Thread Safe (NTS) x86</a> <a href="#">7.4 Thread Safe (TS) x86</a>
PHP 7.3	<a href="#">7.3 Non Thread Safe (NTS) x64</a> <a href="#">7.3 Thread Safe (TS) x64</a> <a href="#">7.3 Non Thread Safe (NTS) x86</a> <a href="#">7.3 Thread Safe (TS) x86</a>
PHP 7.2	<a href="#">7.2 Non Thread Safe (NTS) x64</a> <a href="#">7.2 Thread Safe (TS) x64</a> <a href="#">7.2 Non Thread Safe (NTS) x86</a> <a href="#">7.2 Thread Safe (TS) x86</a>

# Preparación de proyecto

Por último nos descargamos el composer para complementar la conexión de php y mongo. Nos lo descargamos y lo guardamos en c/xampp/php/php.exe.

Creamos en el xampp/htdocs una carpeta llamada db, accedemos a ella desde el cmd y escribimos esto  una vez ejecutada, se habrán creado 3 nuevos archivos y estará todo listo.

Para arrancar todo, activamos apache del xampp, iniciamos servidor de mongo.

Posteriormente, si queremos utilizar las bd de mongo, hay que crearlas junto a sus colecciones. Utilizando use “namebd” y db.createCollection(“nameCollection”) . La base de datos usada en el proyecto se llama bicis y sus colecciones users, reservas y bicicletas.

# Funcionamiento »»Resumen general

Hemos implementado una pagina donde esta el login el registro.  
Para iniciar sesión ,el usuario tiene que estar registrado en la bd,  
en el caso de que no lo esté; tiene la opción de registrarse  
en la misma página escribiendo su contraseña y email en un  
formulario que se incluirá en la bbdd.

Hay dos tipos de usuario , el administrador y el user .

Cada tipo de usuario tiene acceso a unas funciones específicas, que realizan distintas consultas en la bd.

Para añadir el primer administrador, hay que registrar a un usuario normal desde la interfaz web y cambiar su rol a “admin” en la bbdd.



# Funcionamiento »»Login:

Hemos hecho una página para darse de alta y registrarse en nuestra web .

Tiene implementado sus excepciones:

- No está registrado(no está guardado en la bd)
- Contraseña/usuario incorrecto (el hash de la contraseña no coincide)

Todo depende de que esté guardado en la base de datos. En el caso de que no estén guardado, abajo en la misma página se encuentra el registro de nuevos usuarios

# Funcionamiento»Admin

Los administradores registrados tienen acceso a la página que mostramos a continuación:

Apartados:

**Gestionar bicis:** donde se le muestran todas las bicis de la bd, su estado de reserva. También, al ser admin, puede añadir bicis con su precio particular y eliminarlas de la bd



Añadir bici

Bicis disponibles			
Id	Precio	Estado	Eliminar
bicl_1622047911721	4 créditos	reservada	Eliminar
bicl_1622047918047	6 créditos	reservada	Eliminar
bicl_1622047921226	7 créditos	reservada	Eliminar
bicl_1622047924830	8 créditos	disponible	Eliminar
bicl_1622047928157	9 créditos	disponible	Eliminar
bicl_1622047932949	10 créditos	disponible	Eliminar



# Funcionamiento»Admin

**Gestionar usuarios:** Este apartado nos muestra los usuarios registrados además de su información.

Permite al admin cambiarle su rol en la aplicación, user a admin y viceversa , al igual que eliminarlos y guardar los cambios en la bd.

**Gestionar reservas:** Se muestran los datos cada reserva en curso (la bicicleta reservada y el usuario). El administrador también tiene la opción de anular reserva.

Inicio

Gestionar Usuarios

Gestionar Bicis

Gestionar Reservas

Sair

Administradores

Nombre	Email	Cambiar Rol	Eliminar
Iola	Iola@gmail.com	<div>Administrador <input checked="" type="radio"/></div> <div>General <input type="radio"/></div> <div>Actualizar</div>	Eliminar

Usuarios

Nombre	Email	Saldo	Cambiar Rol	Eliminar
pepe	pepe@gmail.com	0	<div>Administrador <input type="radio"/></div> <div>General <input checked="" type="radio"/></div> <div>Actualizar</div>	Eliminar
Juan	Juan@gmail.com	5	<div>Administrador <input type="radio"/></div> <div>General <input checked="" type="radio"/></div> <div>Actualizar</div>	Eliminar

Bicis reservadas			
Id	Usuario	Estado	Anular Reserva
bici_1622047911721	user@gmail.com	reservada	Anular reserva
bici_1622047921226	user@gmail.com	reservada	Anular reserva
bici_1622047918047	user@gmail.com	reservada	Anular reserva

# Funcionamiento»User

Los usuarios tienen acceso a la siguiente página:

Apartados:

**Catálogo de bicis:** Se muestran todas las bicis disponibles (añadidas por el admin) para reservar. Si el usuario tiene los créditos suficientes al hacer click la reserva, en caso contrario salta un mensaje de error



# Funcionamiento»User

**Mis reservas:** Se muestra al usuario información sobre las bicicletas que ha reservado así como la posibilidad de anular cada reserva.



The screenshot shows a web interface with a dark navigation bar at the top containing 'Inicio', 'Catálogo de Bicis', and 'Mis Reservas'. On the right of the bar are two buttons: '1 créditos' (yellow) and 'Salir' (red). Below the navigation bar, the main content area has a light green background. It features a table titled 'Bicis reservadas' with three columns: 'Id', 'Estado', and 'Anular Reserva'. The table contains three rows of data, each with a bicycle ID, the state 'reservada', and an 'Anular reserva' button.

Bicis reservadas		
Id	Estado	Anular Reserva
bici_1622047911721	reservada	Anular reserva
bici_1622047921226	reservada	Anular reserva
bici_1622047918047	reservada	Anular reserva

**Añadir saldo:** Al pulsar sobre los créditos disponibles, se abre una interfaz mediante la cual el usuario puede añadir más créditos a su cuenta.



The screenshot shows the same web interface as before, but with a modal form open in the center. The modal has a white background and a light green border. It is titled 'Añadir saldo' and contains a text input field and a green 'Añadir' button.

Inicio Catálogo de Bicis Mis Reservas

1 créditos Salir

Añadir saldo

Añadir

# Ficheros de interfaz

En la carpeta principal del proyecto se encuentran los siguientes archivos, que contienen las interfaces básicas que se encargarán de llamar a los distintos scripts:

**gestionarBicis.php** → Página del administrador que muestra la interfaz de gestión de bicicletas.

**gestionarUsers.php** → Página del administrador que muestra la interfaz de gestión de usuarios.

**home.php** → Página principal que muestra un mensaje de bienvenida al usuario.

**index.php** → Página de login y registro.

**reservas.php** → Página en la que al usuario se le muestran sus reservas en curso y al administrador las de todos los usuarios, con posibilidad de anularlas.

**saldo.php** → Página del usuario general que muestra la interfaz para añadir créditos a su cuenta.

**showBicis.php** → Página del usuario general que muestra en una cuadrícula las bicis disponibles para reservar.

**style.css** → Contiene las hojas de estilo que dan formato a los distintos elementos de cada página.

# Scripts

Dentro de la carpeta includes se encuentran los distintos scripts php que hacen consultas a la bbdd de mongodb y devuelven los datos a las interfaces:

**actualizarBici.php** → Actualiza el estado de una bici de la colección bicicletas o elimina un documento de la colección reservas

**actualizarRol.php** → Cambia el rol de un usuario de la colección users.

**addBici.php** → Añade un nuevo documento con los datos de una nueva bici (id con milisegundos en tiempo unix, precio, y estado) a la colección bicicletas.

**addSaldo.php** → Actualiza el saldo del usuario con la sesión iniciada en la colección users

**config.php** → Establece la conexión a la bbdd e inicia la sesión del usuario

**eliminarBici.php** → Elimina de la colección bicicletas el documento asociado a una bici con un id dado (en caso de que no esté reservada)

**eliminarUser.php** → Elimina de la colección users el documento asociado a un usuario con un id dado

**listaBicis.php** → Hace una consulta a la base colección bicicletas y devuelve una tabla con la información de cada bici

# Scripts

**listaReservas.php** → Hace una consulta a la colección reservas y devuelve una tabla con las reservas en curso de todos los usuarios (en caso del admin), o del propio usuario (en caso del usuario general).

**listaUsers.php** → Hace dos consultas a la colección users y devuelve una tabla con la información de los administradores y otra con la de los usuarios normales.

**logout.php** → script para cerrar la sesión del usuario.

**procesarLogin.php** → script para procesar el login, comprueba en la colección users si existe el usuario y si el hash de la contraseña coincide con el hash de la introducida.

**procesarRegistro.php** → procesa el registro, valida los datos introducidos, comprueba en la colección users que no hay un usuario con el mismo email y de ser así introduce en dicha colección un documento con los datos del nuevo usuario.

**reservarBici.php** → reserva una nueva bici, se consultan y actualizan las colecciones bicicletas, reservas y users, si el usuario tiene saldo suficiente se le resta el precio de la bici, a la bici se le cambia su estado a reservada y se añade un documento con el id de la bici y el email del usuario en la colección reservas.

**sesionUser.php** → script con las funciones de login y logout. Inicia y mantiene la sesión del usuario en el navegador y la borra cuando se llama a logout

**topBar.php** → contiene la barra superior de navegación con las opciones específicas de cada tipo de usuario