

RadarCovid

Generado por Doxygen 1.8.20

1 Indice de namespaces	1
1.1 Paquetes	1
2 Indice jerárquico	3
2.1 Jerarquía de la clase	3
3 Índice de clases	5
3.1 Lista de clases	5
4 Indice de archivos	7
4.1 Lista de archivos	7
5 Documentación de namespaces	9
5.1 Paquetes AppService	9
5.2 Paquetes DAO	9
5.3 Paquetes JettyServer	9
5.4 Paquetes Objects	9
6 Documentación de las clases	11
6.1 Referencia de la Clase AppService.AuthGestor	11
6.2 Referencia de la Clase DAO.DAOAuthGestor	11
6.2.1 Descripción detallada	11
6.2.2 Documentación de las funciones miembro	11
6.2.2.1 deleteUser()	12
6.2.2.2 getDAOAuthgestor()	12
6.2.2.3 registerUser()	12
6.2.2.4 selectUsers()	13
6.3 Referencia de la Clase DAO.DAOGestor	13
6.3.1 Descripción detallada	14
6.3.2 Documentación de las funciones miembro	14
6.3.2.1 deleteUser()	14
6.3.2.2 getDAOgestor()	14
6.3.2.3 login()	14
6.3.2.4 main()	15
6.3.2.5 positivesList()	15
6.3.2.6 registerPositive()	15
6.3.2.7 registrarse()	16
6.3.2.8 usersList()	17
6.3.3 Documentación de los datos miembro	17
6.3.3.1 positives	17
6.3.3.2 registered	17
6.3.3.3 userLogged	17
6.3.3.4 users	17
6.4 Referencia de la Clase DAO.DAOPositiveGestor	18

6.4.1 Descripción detallada	18
6.4.2 Documentación de las funciones miembro	18
6.4.2.1 getDAOPositivegestor()	18
6.4.2.2 registerPositive()	18
6.4.2.3 selectPositives()	19
6.5 Referencia de la Clase DAO.DAOStatsGestor	19
6.5.1 Descripción detallada	20
6.5.2 Documentación de las funciones miembro	20
6.5.2.1 getDAOStatsgestor()	20
6.5.2.2 readAllStats()	20
6.6 Referencia de la Clase JettyServer.JettyServer	20
6.6.1 Descripción detallada	21
6.6.2 Documentación de las funciones miembro	21
6.6.2.1 main()	21
6.6.2.2 start()	21
6.6.2.3 stop()	21
6.7 Referencia de la Clase JettyServer.LoginServlet	22
6.7.1 Descripción detallada	22
6.7.2 Documentación de las funciones miembro	22
6.7.2.1 doGet()	22
6.8 Referencia de la Clase Objects.Positive	23
6.8.1 Descripción detallada	23
6.8.2 Documentación del constructor y destructor	23
6.8.2.1 Positive()	23
6.9 Referencia de la Clase JettyServer.PositiveAddServlet	23
6.9.1 Descripción detallada	24
6.9.2 Documentación de las funciones miembro	24
6.9.2.1 doGet()	24
6.10 Referencia de la Clase AppService.PositiveGestor	24
6.10.1 Descripción detallada	24
6.10.2 Documentación de las funciones miembro	25
6.10.2.1 getPositivegestor()	25
6.10.2.2 registerPositive()	25
6.10.2.3 searchPositives()	25
6.11 Referencia de la Clase JettyServer.PositivesServlet	26
6.11.1 Descripción detallada	26
6.11.2 Documentación de las funciones miembro	26
6.11.2.1 doGet()	26
6.12 Referencia de la Clase JettyServer.RegisterServlet	27
6.12.1 Descripción detallada	27
6.12.2 Documentación de las funciones miembro	27
6.12.2.1 doGet()	27

6.13 Referencia de la Clase AppService.StatsGestor	27
6.13.1 Descripción detallada	28
6.13.2 Documentación de las funciones miembro	28
6.13.2.1 getStatsgestor()	28
6.13.2.2 obtainStatistics()	28
6.14 Referencia de la Clase JettyServer.StatsServlet	29
6.14.1 Descripción detallada	29
6.14.2 Documentación de las funciones miembro	29
6.14.2.1 doGet()	29
6.15 Referencia de la Clase Objects.User	29
6.15.1 Descripción detallada	30
6.15.2 Documentación de las funciones miembro	30
6.15.2.1 toString()	30
6.15.3 Documentación de los datos miembro	30
6.15.3.1 admin	30
6.15.3.2 age	31
6.15.3.3 email	31
6.15.3.4 gender	31
6.15.3.5 idCard	31
6.15.3.6 occupation	31
6.15.3.7 password	31
6.15.3.8 username	31
7 Documentación de archivos	33
7.1 Referencia del Archivo src/main/java/AppService/AuthGestor.java	33
7.2 Referencia del Archivo src/main/java/AppService/PositiveGestor.java	33
7.3 Referencia del Archivo src/main/java/AppService/StatsGestor.java	33
7.4 Referencia del Archivo src/main/java/DAO/DAOAuthGestor.java	34
7.5 Referencia del Archivo src/main/java/DAO/DAOGestor.java	34
7.6 Referencia del Archivo src/main/java/DAO/DAOPositiveGestor.java	34
7.7 Referencia del Archivo src/main/java/DAO/DAOStatsGestor.java	34
7.8 Referencia del Archivo src/main/java/JettyServer/JettyServer.java	35
7.9 Referencia del Archivo src/main/java/JettyServer/LoginServlet.java	35
7.10 Referencia del Archivo src/main/java/JettyServer/PositiveAddServlet.java	35
7.11 Referencia del Archivo src/main/java/JettyServer/PositivesServlet.java	35
7.12 Referencia del Archivo src/main/java/JettyServer/RegisterServlet.java	36
7.13 Referencia del Archivo src/main/java/JettyServer/StatsServlet.java	36
7.14 Referencia del Archivo src/main/java/Objects/Positive.java	36
7.15 Referencia del Archivo src/main/java/Objects/User.java	36
Índice alfabético	37

Capítulo 1

Índice de namespaces

1.1. Paquetes

Aquí van los paquetes con una breve descripción (si está disponible):

AppService	9
DAO	9
JettyServer	9
Objects	9

Capítulo 2

Índice jerárquico

2.1. Jerarquía de la clase

Esta lista de herencias esta ordenada aproximadamente por orden alfabético:

AppService.AuthGestor	11
DAO.DAOAuthGestor	11
DAO.DAOGestor	13
DAO.DAOPositiveGestor	18
DAO.DAOStatsGestor	19
JettyServer.JettyServer	20
Objects.Positive	23
AppService.PositiveGestor	24
AppService.StatsGestor	27
Objects.User	29
HttpServlet	
JettyServer.LoginServlet	22
JettyServer.PositiveAddServlet	23
JettyServer.PositivesServlet	26
JettyServer.RegisterServlet	27
JettyServer.StatsServlet	29

Capítulo 3

Índice de clases

3.1. Lista de clases

Lista de las clases, estructuras, uniones e interfaces con una breve descripción:

AppService.AuthGestor	11
DAO.DAOAuthGestor	11
DAO.DAOGestor	13
DAO.DAOPositiveGestor	18
DAO.DAOStatsGestor	19
JettyServer.JettyServer	20
JettyServer.LoginServlet	22
Objects.Positive	23
JettyServer.PositiveAddServlet	23
AppService.PositiveGestor	24
JettyServer.PositivesServlet	26
JettyServer.RegisterServlet	27
AppService.StatsGestor	27
JettyServer.StatsServlet	29
Objects.User	29

Capítulo 4

Indice de archivos

4.1. Lista de archivos

Lista de todos los archivos con descripciones breves:

src/main/java/AppService/AuthGestor.java	33
src/main/java/AppService/PositiveGestor.java	33
src/main/java/AppService/StatsGestor.java	33
src/main/java/DAO/DAOAuthGestor.java	34
src/main/java/DAO/DAOGestor.java	34
src/main/java/DAO/DAOPositiveGestor.java	34
src/main/java/DAO/DAOStatsGestor.java	34
src/main/java/JettyServer/JettyServer.java	35
src/main/java/JettyServer/LoginServlet.java	35
src/main/java/JettyServer/PositiveAddServlet.java	35
src/main/java/JettyServer/PositivesServlet.java	35
src/main/java/JettyServer/RegisterServlet.java	36
src/main/java/JettyServer/StatsServlet.java	36
src/main/java/Objects/Positive.java	36
src/main/java/Objects/User.java	36

Capítulo 5

Documentación de namespaces

5.1. Paquetes AppService

Clases

- class [AuthGestor](#)
- class [PositiveGestor](#)
- class [StatsGestor](#)

5.2. Paquetes DAO

Clases

- class [DAOAuthGestor](#)
- class [DAOGestor](#)
- class [DAOPositiveGestor](#)
- class [DAOStatsGestor](#)

5.3. Paquetes JettyServer

Clases

- class [JettyServer](#)
- class [LoginServlet](#)
- class [PositiveAddServlet](#)
- class [PositivesServlet](#)
- class [RegisterServlet](#)
- class [StatsServlet](#)

5.4. Paquetes Objects

Clases

- class [Positive](#)
- class [User](#)

Capítulo 6

Documentación de las clases

6.1. Referencia de la Clase AppService.AuthGestor

Diagrama de colaboración para AppService.AuthGestor:

6.2. Referencia de la Clase DAO.DAOAuthGestor

Métodos públicos

- String [registerUser](#) (int idCard, String username, String password, String email, int age, String gender, String occupation, boolean admin)
- void [deleteUser](#) ()
- void [selectUsers](#) ()

Métodos protegidos estáticos

- static [DAOAuthGestor](#) [getDAOAuthgestor](#) ()

6.2.1. Descripción detallada

Gestor [DAO](#) que se relaciona con la base de datos remota a traves de Datanucleus para la autenticacion de usuarios.

Autor

Alumno

Definición en la línea 15 del archivo DAOAuthGestor.java.

6.2.2. Documentación de las funciones miembro

6.2.2.1. deleteUser()

```
void DAO.DAOAuthGestor.deleteUser ( )
```

Método que borra el usuario de la base de datos a través de Datanucleus.

Definición en la línea 76 del archivo DAOAuthGestor.java.

Gráfico de llamadas a esta función:

6.2.2.2. getDAOAuthgestor()

```
static DAOAuthGestor DAO.DAOAuthGestor.getDAOAuthgestor ( ) [static], [protected]
```

Método que crea o recupera la instancia del [DAOAuthGestor](#). Sigue el patrón Singleton.

Devuelve

gestorDAO

Definición en la línea 24 del archivo DAOAuthGestor.java.

Gráfico de llamadas a esta función:

6.2.2.3. registerUser()

```
String DAO.DAOAuthGestor.registerUser (
    int idCard,
    String username,
    String password,
    String email,
    int age,
    String gender,
    String occupation,
    boolean admin )
```

Método que registra el usuario en la base de datos a través de Datanucleus.

Parámetros

<i>idCard</i>	
<i>username</i>	
<i>password</i>	
<i>email</i>	
<i>age</i>	
<i>gender</i>	
<i>occupation</i>	
<i>admin</i>	

Devuelve

string (para comprobar si se ha registrado con éxito el usuario o no)

Definición en la línea 44 del archivo DAOAuthGestor.java.

Gráfico de llamadas para esta función: Gráfico de llamadas a esta función:

6.2.2.4. selectUsers()

```
void DAO.DAOAuthGestor.selectUsers ( )
```

Método que selecciona todos los usuarios de la base de datos a través de Datanucleus.

Definición en la línea 108 del archivo DAOAuthGestor.java.

Gráfico de llamadas a esta función:

La documentación para esta clase fue generada a partir del siguiente fichero:

- src/main/java/DAO/DAOAuthGestor.java

6.3. Referencia de la Clase DAO.DAOGestor

Diagrama de colaboración para DAO.DAOGestor:

Métodos públicos

- String [login](#) (int idCard, String password)
- String [registrarse](#) (int idCard, String username, String password, String email, int age, String gender, String occupation, boolean admin)
- void [deleteUser](#) ()
- boolean [registerPositive](#) (double latitude, double longitude, int year, int month, int day)

Métodos públicos estáticos

- static [DAOGestor](#) [getDAOgestor](#) ()
- static void [main](#) (String[] args)
- static void [usersList](#) ()
- static String [positivesList](#) ()

Atributos públicos

- boolean [registered](#) = false

Atributos públicos estáticos

- static ArrayList< [User](#) > [users](#) = new ArrayList<[User](#)>()
- static ArrayList< [Positive](#) > [positives](#) = new ArrayList<[Positive](#)>()
- static [User](#) [userLogged](#) = new [User](#)(1,"","","1","",false)

6.3.1. Descripción detallada

Gestor [DAO](#) que recoge las llamadas del [AppService](#) y delega el trabajo en los demas gestores

Autor

Alumno

Definición en la línea 17 del archivo DAOGestor.java.

6.3.2. Documentación de las funciones miembro

6.3.2.1. deleteUser()

```
void DAO.DAOGestor.deleteUser ( )
```

Metodo para borrar un usuario de RadarCovid. Delega la funcion en el [DAOAuthGestor#deleteUser\(\)](#)

Definición en la línea 131 del archivo DAOGestor.java.

Gráfico de llamadas para esta función: Gráfico de llamadas a esta función:

6.3.2.2. getDAOgestor()

```
static DAOGestor DAO.DAOGestor.getDAOgestor ( ) [static]
```

Metodo que crea o recupera la instancia del [DAOGestor](#). Sigue el patron Singleton.

Devuelve

gestorDAO

Definición en la línea 30 del archivo DAOGestor.java.

Gráfico de llamadas a esta función:

6.3.2.3. login()

```
String DAO.DAOGestor.login (
    int idCard,
    String password )
```

Metodo que comprueba el login con los usuarios de la base de datos. Coteja los usuarios con la lista de

Ver también

[DAOGestor#usersList\(\)](#)

Parámetros

<i>idCard</i>	
<i>password</i>	

Devuelve

el usuario logeado

Definición en la línea 86 del archivo DAOGestor.java.

Gráfico de llamadas para esta función: Gráfico de llamadas a esta función:

6.3.2.4. main()

```
static void DAO.DAOGestor.main (
    String[] args ) [static]
```

Definición en la línea 39 del archivo DAOGestor.java.

Gráfico de llamadas para esta función:

6.3.2.5. positivesList()

```
static String DAO.DAOGestor.positivesList ( ) [static]
```

Metodo para obtener la lista de positivos de RadarCovid. Delega la funcion en el [DAOPositiveGestor#selectPositives\(\)](#)

Devuelve

Definición en la línea 171 del archivo DAOGestor.java.

Gráfico de llamadas para esta función: Gráfico de llamadas a esta función:

6.3.2.6. registerPositive()

```
boolean DAO.DAOGestor.registerPositive (
    double latitude,
    double longitude,
    int year,
    int month,
    int day )
```

Metodo para regitrar positivo en RadarCovid. Delega la funcion en el [DAOPositiveGestor#registerPositive\(double, double, int, int, int\)](#)

Parámetros

<i>latitude</i>	
<i>longitude</i>	
<i>year</i>	
<i>month</i>	
<i>day</i>	

Devuelve

Definición en la línea 159 del archivo DAOGestor.java.

Gráfico de llamadas para esta función: Gráfico de llamadas a esta función:

6.3.2.7. registrarse()

```
String DAO.DAOGestor.registrarse (
    int idCard,
    String username,
    String password,
    String email,
    int age,
    String gender,
    String occupation,
    boolean admin )
```

Método para registrarse en RadarCovid. Delega la función en el [DAOAuthGestor#registerUser\(int, String, String, String, int, String, String, boolean\)](#).

Parámetros

<i>idCard</i>	
<i>username</i>	
<i>password</i>	
<i>email</i>	
<i>age</i>	
<i>gender</i>	
<i>occupation</i>	
<i>admin</i>	

Devuelve

Definición en la línea 115 del archivo DAOGestor.java.

Gráfico de llamadas para esta función: Gráfico de llamadas a esta función:

6.3.2.8. usersList()

```
static void DAO.DAOGestor.usersList ( ) [static]
```

Método que recupera la lista de usuarios de RadarCovid. Delega el trabajo en el [DAOAuthGestor#selectUsers\(\)](#).

Definición en la línea 69 del archivo DAOGestor.java.

Gráfico de llamadas para esta función: Gráfico de llamadas a esta función:

6.3.3. Documentación de los datos miembro

6.3.3.1. positives

```
ArrayList<Positive> DAO.DAOGestor.positives = new ArrayList<Positive>() [static]
```

Definición en la línea 22 del archivo DAOGestor.java.

6.3.3.2. registered

```
boolean DAO.DAOGestor.registered = false
```

Definición en la línea 20 del archivo DAOGestor.java.

6.3.3.3. userLogged

```
User DAO.DAOGestor.userLogged = new User(1, "", "", "", 1, "", "", false) [static]
```

Definición en la línea 23 del archivo DAOGestor.java.

6.3.3.4. users

```
ArrayList<User> DAO.DAOGestor.users = new ArrayList<User>() [static]
```

Definición en la línea 21 del archivo DAOGestor.java.

La documentación para esta clase fue generada a partir del siguiente fichero:

- src/main/java/DAO/[DAOGestor.java](#)

6.4. Referencia de la Clase DAO.DAOPositiveGestor

Métodos públicos

- String [selectPositives](#) ()

Métodos públicos estáticos

- static [DAOPositiveGestor](#) [getDAOPositivegestor](#) ()

Métodos protegidos

- boolean [registerPositive](#) (double latitude, double longitude, int year, int month, int day)

6.4.1. Descripción detallada

Gestor [DAO](#) que se relaciona con la base de datos remota a traves de Datanucleus para la gestion de positivos.

Autor

Alumno

Definición en la línea 12 del archivo DAOPositiveGestor.java.

6.4.2. Documentación de las funciones miembro

6.4.2.1. [getDAOPositivegestor\(\)](#)

```
static DAOPositiveGestor DAO.DAOPositiveGestor.getDAOPositivegestor ( ) [static]
```

Metodo que crea o recupera la instancia del [DAOPositiveGestor](#). Sigue el patron Singleton.

Devuelve

gestorDAO

Definición en la línea 21 del archivo DAOPositiveGestor.java.

Gráfico de llamadas a esta función:

6.4.2.2. [registerPositive\(\)](#)

```
boolean DAO.DAOPositiveGestor.registerPositive (
    double latitude,
    double longitude,
    int year,
    int month,
    int day ) [protected]
```

Metodo que registra el positivo en la base de datos a traves de Datanucleus.

Parámetros

<i>latitude</i>	
<i>longitude</i>	
<i>year</i>	
<i>month</i>	
<i>day</i>	

Devuelve

boolean (si se ha registrado con exito o no)

Definición en la línea 38 del archivo DAOPositiveGestor.java.

Gráfico de llamadas para esta función: Gráfico de llamadas a esta función:

6.4.2.3. selectPositives()

```
String DAO.DAOPositiveGestor.selectPositives ( )
```

Metodo que selecciona los positivos de la base de datos a traves de Datanucleus.

Devuelve

string (lista de positivos)

Definición en la línea 70 del archivo DAOPositiveGestor.java.

Gráfico de llamadas a esta función:

La documentación para esta clase fue generada a partir del siguiente fichero:

- [src/main/java/DAO/DAOPositiveGestor.java](#)

6.5. Referencia de la Clase DAO.DAOStatsGestor

Métodos públicos

- void [readAllStats](#) ()

Métodos públicos estáticos

- static [DAOStatsGestor](#) [getDAOStatsgestor](#) ()

6.5.1. Descripción detallada

Gestor [DAO](#) que se relaciona con la base de datos remota a traves de Datanucleus para la gestion de estadísticas.

Autor

Alumno

Definición en la línea 13 del archivo DAOStatsGestor.java.

6.5.2. Documentación de las funciones miembro

6.5.2.1. `getDAOStatsgestor()`

```
static DAOStatsGestor DAO.DAOStatsGestor.getDAOStatsgestor ( ) [static]
```

Metodo que crea o recupera la instancia del [DAOStatsGestor](#). Sigue el patron Singleton.

Devuelve

gestorDAO

Definición en la línea 42 del archivo DAOStatsGestor.java.

Gráfico de llamadas a esta función:

6.5.2.2. `readAllStats()`

```
void DAO.DAOStatsGestor.readAllStats ( )
```

Metodo que lee todas las estadísticas de la base de datos a traves de Datanucleus.

Definición en la línea 53 del archivo DAOStatsGestor.java.

Gráfico de llamadas a esta función:

La documentación para esta clase fue generada a partir del siguiente fichero:

- `src/main/java/DAO/DAOStatsGestor.java`

6.6. Referencia de la Clase `JettyServer.JettyServer`

Métodos públicos estáticos

- static void [main](#) (String[] args)
- static void [start](#) () throws Exception
- static void [stop](#) () throws Exception

6.6.1. Descripción detallada

Servidor Jetty para poder asociar el HTML y poder recoger las peticiones HTTP que realice el usuario.

Autor

Alumno

Definición en la línea 13 del archivo JettyServer.java.

6.6.2. Documentación de las funciones miembro

6.6.2.1. main()

```
static void JettyServer.JettyServer.main (
    String[] args ) [static]
```

Metodo en el que se inicia la aplicacion. Lanza el metodo JettyServer#start()

Parámetros

<i>args</i>	
-------------	--

Definición en la línea 20 del archivo JettyServer.java.

Gráfico de llamadas para esta función:

6.6.2.2. start()

```
static void JettyServer.JettyServer.start ( ) throws Exception [static]
```

Metodo que lanza el servidor Jetty.

Excepciones

<i>Exception</i>	
------------------	--

Definición en la línea 32 del archivo JettyServer.java.

Gráfico de llamadas a esta función:

6.6.2.3. stop()

```
static void JettyServer.JettyServer.stop ( ) throws Exception [static]
```

Metodo que para el servidor Jetty.

Excepciones

<i>Exception</i>	
------------------	--

Definición en la línea 60 del archivo JettyServer.java.

La documentación para esta clase fue generada a partir del siguiente fichero:

- `src/main/java/JettyServer/JettyServer.java`

6.7. Referencia de la Clase JettyServer.LoginServlet

Diagrama de herencias de JettyServer.LoginServlet

Diagrama de colaboración para JettyServer.LoginServlet:

Métodos protegidos

- `void doGet (HttpServletRequest request, final HttpServletResponse response) throws ServletException, IOException`

6.7.1. Descripción detallada

Servlet asociado al [JettyServer](#) relacionado con el login del usuario.

Autor

Alumno

Definición en la línea 26 del archivo LoginServlet.java.

6.7.2. Documentación de las funciones miembro

6.7.2.1. doGet()

```
void JettyServer.LoginServlet.doGet (
    HttpServletRequest request,
    final HttpServletResponse response ) throws ServletException, IOException [protected]
```

Peticion GET que comprueba el login de un usuario. Delega la funcion en el [AuthGestor#login\(int, String\)](#)

Definición en la línea 31 del archivo LoginServlet.java.

Gráfico de llamadas para esta función:

La documentación para esta clase fue generada a partir del siguiente fichero:

- `src/main/java/JettyServer/LoginServlet.java`

6.8. Referencia de la Clase Objects.Positive

Métodos públicos

- **Positive** ([User](#) patient, double latitude, double longitude, int year, int month, int day)

6.8.1. Descripción detallada

Clase que representa los positivos en la Covid-19.

Autor

Alumno

Definición en la línea 12 del archivo Positive.java.

6.8.2. Documentación del constructor y destructor

6.8.2.1. Positive()

```
Objects.Positive.Positive (
    User patient,
    double latitude,
    double longitude,
    int year,
    int month,
    int day )
```

Definición en la línea 13 del archivo Positive.java.

La documentación para esta clase fue generada a partir del siguiente fichero:

- `src/main/java/Objects/Positive.java`

6.9. Referencia de la Clase JettyServer.PositiveAddServlet

Diagrama de herencias de JettyServer.PositiveAddServlet

Diagrama de colaboración para JettyServer.PositiveAddServlet:

Métodos protegidos

- void [doGet](#) (HttpServletRequest request, final HttpServletResponse response) throws ServletException, IOException

6.9.1. Descripción detallada

Servlet asociado al [JettyServer](#) relacionado con el registro de positivo.

Autor

Alumno

Definición en la línea 21 del archivo PositiveAddServlet.java.

6.9.2. Documentación de las funciones miembro

6.9.2.1. doGet()

```
void JettyServer.PositiveAddServlet.doGet (
    HttpServletRequest request,
    final HttpServletResponse response ) throws ServletException, IOException [protected]
```

Peticion GET que realiza el registro de positivo. Delega la funcion en el [PositiveGestor#registerPositive\(double, double, int, int, int\)](#)

Definición en la línea 25 del archivo PositiveAddServlet.java.

Gráfico de llamadas para esta función:

La documentación para esta clase fue generada a partir del siguiente fichero:

- [src/main/java/JettyServer/PositiveAddServlet.java](#)

6.10. Referencia de la Clase AppService.PositiveGestor

Métodos públicos

- boolean [registerPositive](#) (double latitude, double longitude, int year, int month, int day)
- String [searchPositives](#) ()

Métodos públicos estáticos

- static [PositiveGestor](#) [getPositivegestor](#) ()

6.10.1. Descripción detallada

[AppService](#) que delega las llamadas relacionadas con el registro de positivos al DAOGestor.

Autor

Alumno

Definición en la línea 12 del archivo PositiveGestor.java.

6.10.2. Documentación de las funciones miembro

6.10.2.1. getPositivegestor()

```
static PositiveGestor AppService.PositiveGestor.getPositivegestor ( ) [static]
```

Método que crea o recupera la instancia del [PositiveGestor](#). Sigue el patrón Singleton.

Devuelve

positivegestor

Definición en la línea 20 del archivo PositiveGestor.java.

Gráfico de llamadas a esta función:

6.10.2.2. registerPositive()

```
boolean AppService.PositiveGestor.registerPositive (
    double latitude,
    double longitude,
    int year,
    int month,
    int day )
```

Método para registrar el positivo en RadarCovid. Delega la función en el [DAOGestor#registerPositive\(double, double, int, int, int\)](#).

Parámetros

<i>latitude</i>	
<i>longitude</i>	
<i>year</i>	
<i>month</i>	
<i>day</i>	

Devuelve

boolean (para indicar si se ha registrado con éxito o no)

Definición en la línea 36 del archivo PositiveGestor.java.

Gráfico de llamadas para esta función: Gráfico de llamadas a esta función:

6.10.2.3. searchPositives()

```
String AppService.PositiveGestor.searchPositives ( )
```

Método para seleccionar los positivos registrados en RadarCovid. Delega la función en el [DAOGestor#positivesList\(\)](#).

Devuelve

Definición en la línea 43 del archivo PositiveGestor.java.

Gráfico de llamadas para esta función: Gráfico de llamadas a esta función:

La documentación para esta clase fue generada a partir del siguiente fichero:

- `src/main/java/AppService/PositiveGestor.java`

6.11. Referencia de la Clase JettyServer.PositivesServlet

Diagrama de herencias de JettyServer.PositivesServlet

Diagrama de colaboración para JettyServer.PositivesServlet:

Métodos protegidos

- `void doGet` (`HttpServletRequest request`, `final HttpServletResponse response`) throws `ServletException`, `IOException`

6.11.1. Descripción detallada

Servlet asociado al [JettyServer](#) relacionado con la obtencion de positivos.

Autor

Alumno

Definición en la línea 25 del archivo PositivesServlet.java.

6.11.2. Documentación de las funciones miembro

6.11.2.1. doGet()

```
void JettyServer.PositivesServlet.doGet (
    HttpServletRequest request,
    final HttpServletResponse response ) throws ServletException, IOException [protected]
```

Peticion GET que obtiene los positivos. Delega la funcion en el [PositiveGestor#searchPositives\(\)](#)

Definición en la línea 30 del archivo PositivesServlet.java.

Gráfico de llamadas para esta función:

La documentación para esta clase fue generada a partir del siguiente fichero:

- `src/main/java/JettyServer/PositivesServlet.java`

6.12. Referencia de la Clase JettyServer.RegisterServlet

Diagrama de herencias de JettyServer.RegisterServlet

Diagrama de colaboración para JettyServer.RegisterServlet:

Métodos protegidos

- void [doGet](#) (HttpServletRequest request, final HttpServletResponse response) throws ServletException, IOException

6.12.1. Descripción detallada

Servlet asociado al [JettyServer](#) relacionado con el registro de usuario.

Autor

Alumno

Definición en la línea 21 del archivo RegisterServlet.java.

6.12.2. Documentación de las funciones miembro

6.12.2.1. doGet()

```
void JettyServer.RegisterServlet.doGet (  
    HttpServletRequest request,  
    final HttpServletResponse response ) throws ServletException, IOException [protected]
```

Peticion GET que realiza el registro de un usuario. Delega la funcion en el [AuthGestor#register\(int, String, String, String, int, String, String\)](#)

Definición en la línea 25 del archivo RegisterServlet.java.

Gráfico de llamadas para esta función:

La documentación para esta clase fue generada a partir del siguiente fichero:

- src/main/java/JettyServer/[RegisterServlet.java](#)

6.13. Referencia de la Clase AppService.StatsGestor

Métodos públicos

- String [obtainStatistics](#) ()

Métodos públicos estáticos

- static [StatsGestor](#) [getStatsgestor](#) ()

6.13.1. Descripción detallada

[AppService](#) que delega las llamadas relacionadas con la obtencion de estadisticas al DAOGestor.

Autor

Alumno

Definición en la línea 14 del archivo StatsGestor.java.

6.13.2. Documentación de las funciones miembro

6.13.2.1. [getStatsgestor\(\)](#)

```
static StatsGestor AppService.StatsGestor.getStatsgestor ( ) [static]
```

Metodo que crea o recupera la instancia del [StatsGestor](#). Sigue el patron Singleton.

Devuelve

statsGestor

Definición en la línea 22 del archivo StatsGestor.java.

Gráfico de llamadas a esta función:

6.13.2.2. [obtainStatistics\(\)](#)

```
String AppService.StatsGestor.obtainStatistics ( )
```

Metodo para pasar a string las estadisticas recogidas del [DAOStatsGestor#readAllStats\(\)](#).

Devuelve

string

Definición en la línea 33 del archivo StatsGestor.java.

Gráfico de llamadas para esta función: Gráfico de llamadas a esta función:

La documentación para esta clase fue generada a partir del siguiente fichero:

- src/main/java/AppService/[StatsGestor.java](#)

6.14. Referencia de la Clase JettyServer.StatsServlet

Diagrama de herencias de JettyServer.StatsServlet

Diagrama de colaboración para JettyServer.StatsServlet:

Métodos protegidos

- void [doGet](#) (HttpServletRequest request, final HttpServletResponse response) throws ServletException, IOException

6.14.1. Descripción detallada

Servlet asociado al [JettyServer](#) relacionado con la obtencion de estadísticas.

Autor

Alumno

Definición en la línea 22 del archivo StatsServlet.java.

6.14.2. Documentación de las funciones miembro

6.14.2.1. doGet()

```
void JettyServer.StatsServlet.doGet (
    HttpServletRequest request,
    final HttpServletResponse response ) throws ServletException, IOException [protected]
```

Peticion GET que obtiene las estadísticas. Delega la funcion en el [StatsGestor#obtainStatistics\(\)](#)

Definición en la línea 26 del archivo StatsServlet.java.

Gráfico de llamadas para esta función:

La documentación para esta clase fue generada a partir del siguiente fichero:

- src/main/java/JettyServer/[StatsServlet.java](#)

6.15. Referencia de la Clase Objects.User

Métodos públicos

- String [toString](#) ()

Atributos públicos

- int `idCard`
- String `username`
- String `password`
- String `email`
- int `age`
- String `gender`
- String `occupation`
- boolean `admin` =false

6.15.1. Descripción detallada

Clase que representa los usuarios registrados en RadarCovid.

Autor

Alumno

Definición en la línea 13 del archivo User.java.

6.15.2. Documentación de las funciones miembro

6.15.2.1. toString()

```
String Objects.User.toString ( )
```

Definición en la línea 35 del archivo User.java.

Gráfico de llamadas a esta función:

6.15.3. Documentación de los datos miembro

6.15.3.1. admin

```
boolean Objects.User.admin =false
```

Definición en la línea 32 del archivo User.java.

6.15.3.2. age

```
int Objects.User.age
```

Definición en la línea 26 del archivo User.java.

6.15.3.3. email

```
String Objects.User.email
```

Definición en la línea 24 del archivo User.java.

6.15.3.4. gender

```
String Objects.User.gender
```

Definición en la línea 28 del archivo User.java.

6.15.3.5. idCard

```
int Objects.User.idCard
```

Definición en la línea 18 del archivo User.java.

6.15.3.6. occupation

```
String Objects.User.occupation
```

Definición en la línea 30 del archivo User.java.

6.15.3.7. password

```
String Objects.User.password
```

Definición en la línea 22 del archivo User.java.

6.15.3.8. username

```
String Objects.User.username
```

Definición en la línea 20 del archivo User.java.

La documentación para esta clase fue generada a partir del siguiente fichero:

- `src/main/java/Objects/User.java`

Capítulo 7

Documentación de archivos

7.1. Referencia del Archivo `src/main/java/AppService/AuthGestor.java`

Clases

- class [AppService.AuthGestor](#)

Paquetes

- package [AppService](#)

7.2. Referencia del Archivo `src/main/java/AppService/PositiveGestor.java`

Clases

- class [AppService.PositiveGestor](#)

Paquetes

- package [AppService](#)

7.3. Referencia del Archivo `src/main/java/AppService/StatsGestor.java`

Clases

- class [AppService.StatsGestor](#)

Paquetes

- package [AppService](#)

7.4. Referencia del Archivo src/main/java/DAO/DAOAuthGestor.java

Clases

- class [DAO.DAOAuthGestor](#)

Paquetes

- package [DAO](#)

7.5. Referencia del Archivo src/main/java/DAO/DAOGestor.java

Clases

- class [DAO.DAOGestor](#)

Paquetes

- package [DAO](#)

7.6. Referencia del Archivo src/main/java/DAO/DAOPositiveGestor.java

Clases

- class [DAO.DAOPositiveGestor](#)

Paquetes

- package [DAO](#)

7.7. Referencia del Archivo src/main/java/DAO/DAOStatsGestor.java

Clases

- class [DAO.DAOStatsGestor](#)

Paquetes

- package [DAO](#)

7.8. Referencia del Archivo src/main/java/JettyServer/JettyServer.java

Clases

- class [JettyServer.JettyServer](#)

Paquetes

- package [JettyServer](#)

7.9. Referencia del Archivo src/main/java/JettyServer/LoginServlet.java

Clases

- class [JettyServer.LoginServlet](#)

Paquetes

- package [JettyServer](#)

7.10. Referencia del Archivo src/main/java/JettyServer/PositiveAddServlet.java

Clases

- class [JettyServer.PositiveAddServlet](#)

Paquetes

- package [JettyServer](#)

7.11. Referencia del Archivo src/main/java/JettyServer/PositivesServlet.java

Clases

- class [JettyServer.PositivesServlet](#)

Paquetes

- package [JettyServer](#)

7.12. Referencia del Archivo `src/main/java/JettyServer/RegisterServlet.java`

Clases

- class [JettyServer.RegisterServlet](#)

Paquetes

- package [JettyServer](#)

7.13. Referencia del Archivo `src/main/java/JettyServer/StatsServlet.java`

Clases

- class [JettyServer.StatsServlet](#)

Paquetes

- package [JettyServer](#)

7.14. Referencia del Archivo `src/main/java/Objects/Positive.java`

Clases

- class [Objects.Positive](#)

Paquetes

- package [Objects](#)

7.15. Referencia del Archivo `src/main/java/Objects/User.java`

Clases

- class [Objects.User](#)

Paquetes

- package [Objects](#)

Índice alfabético

admin
 Objects.User, 30
age
 Objects.User, 30
AppService, 9
AppService.AuthGestor, 11
AppService.PositiveGestor, 24
 getPositivegestor, 25
 registerPositive, 25
 searchPositives, 25
AppService.StatsGestor, 27
 getStatsgestor, 28
 obtainStatistics, 28

DAO, 9
DAO.DAOAuthGestor, 11
 deleteUser, 11
 getDAOAuthgestor, 12
 registerUser, 12
 selectUsers, 13
DAO.DAOGestor, 13
 deleteUser, 14
 getDAOgestor, 14
 login, 14
 main, 15
 positives, 17
 positivesList, 15
 registered, 17
 registerPositive, 15
 registrarse, 16
 userLogged, 17
 users, 17
 usersList, 16
DAO.DAOPositiveGestor, 18
 getDAOPositivegestor, 18
 registerPositive, 18
 selectPositives, 19
DAO.DAOStatsGestor, 19
 getDAOStatsgestor, 20
 readAllStats, 20
deleteUser
 DAO.DAOAuthGestor, 11
 DAO.DAOGestor, 14
doGet
 JettyServer.LoginServlet, 22
 JettyServer.PositiveAddServlet, 24
 JettyServer.PositivesServlet, 26
 JettyServer.RegisterServlet, 27
 JettyServer.StatsServlet, 29

email
 Objects.User, 31

gender
 Objects.User, 31
getDAOAuthgestor
 DAO.DAOAuthGestor, 12
getDAOgestor
 DAO.DAOGestor, 14
getDAOPositivegestor
 DAO.DAOPositiveGestor, 18
getDAOStatsgestor
 DAO.DAOStatsGestor, 20
getPositivegestor
 AppService.PositiveGestor, 25
getStatsgestor
 AppService.StatsGestor, 28

idCard
 Objects.User, 31

JettyServer, 9
JettyServer.JettyServer, 20
 main, 21
 start, 21
 stop, 21
JettyServer.LoginServlet, 22
 doGet, 22
JettyServer.PositiveAddServlet, 23
 doGet, 24
JettyServer.PositivesServlet, 26
 doGet, 26
JettyServer.RegisterServlet, 27
 doGet, 27
JettyServer.StatsServlet, 29
 doGet, 29

login
 DAO.DAOGestor, 14

main
 DAO.DAOGestor, 15
 JettyServer.JettyServer, 21

Objects, 9
Objects.Positive, 23
 Positive, 23
Objects.User, 29
 admin, 30
 age, 30

- email, [31](#)
- gender, [31](#)
- idCard, [31](#)
- occupation, [31](#)
- password, [31](#)
- toString, [30](#)
- username, [31](#)
- obtainStatistics
 - AppService.StatsGestor, [28](#)
- occupation
 - Objects.User, [31](#)
- password
 - Objects.User, [31](#)
- Positive
 - Objects.Positive, [23](#)
- positives
 - DAO.DAOGestor, [17](#)
- positivesList
 - DAO.DAOGestor, [15](#)
- readAllStats
 - DAO.DAOSTatsGestor, [20](#)
- registered
 - DAO.DAOGestor, [17](#)
- registerPositive
 - AppService.PositiveGestor, [25](#)
 - DAO.DAOGestor, [15](#)
 - DAO.DAOPositiveGestor, [18](#)
- registerUser
 - DAO.DAOAuthGestor, [12](#)
- registrarse
 - DAO.DAOGestor, [16](#)
- searchPositives
 - AppService.PositiveGestor, [25](#)
- selectPositives
 - DAO.DAOPositiveGestor, [19](#)
- selectUsers
 - DAO.DAOAuthGestor, [13](#)
- src/main/java/AppService/AuthGestor.java, [33](#)
- src/main/java/AppService/PositiveGestor.java, [33](#)
- src/main/java/AppService/StatsGestor.java, [33](#)
- src/main/java/DAO/DAOAuthGestor.java, [34](#)
- src/main/java/DAO/DAOGestor.java, [34](#)
- src/main/java/DAO/DAOPositiveGestor.java, [34](#)
- src/main/java/DAO/DAOSTatsGestor.java, [34](#)
- src/main/java/JettyServer/JettyServer.java, [35](#)
- src/main/java/JettyServer/LoginServlet.java, [35](#)
- src/main/java/JettyServer/PositiveAddServlet.java, [35](#)
- src/main/java/JettyServer/PositivesServlet.java, [35](#)
- src/main/java/JettyServer/RegisterServlet.java, [36](#)
- src/main/java/JettyServer/StatsServlet.java, [36](#)
- src/main/java/Objects/Positive.java, [36](#)
- src/main/java/Objects/User.java, [36](#)
- start
 - JettyServer.JettyServer, [21](#)
- stop
 - JettyServer.JettyServer, [21](#)
- toString
 - Objects.User, [30](#)
- userLogged
 - DAO.DAOGestor, [17](#)
- username
 - Objects.User, [31](#)
- users
 - DAO.DAOGestor, [17](#)
- usersList
 - DAO.DAOGestor, [16](#)