



Materia: Metodologías Ágiles para Aplicaciones Web
Integrantes: Benitez Alvarenga, Javier
Gil, Rodrigo
Sosa, Milton
Villarrubia, Mayra

Proyecto: Evaluación del Clima de Conservación (ECCO)

1. Objetivo del proyecto

Desarrollar una aplicación que gestione usuarios, bibliotecas y salas (pertenecientes a una biblioteca), permitiendo que un usuario registrado pueda conocer cuál es el clima ideal de una sala basado en la condición previa a la cual los materiales de dicha sala se aclimataron. Esta información previa se obtiene a través de dataloggers que registran durante cierto tiempo la temperatura y humedad de la sala en cuestión.

2. User Stories desarrolladas

I. Como usuario de una biblioteca monitorizada que aún no ha sido incorporada a sistema (nuevo usuario) deseo registrarme como usuario del sistema, para poder registrar los datos de mi biblioteca.

II. Como nuevo usuario deseo registrar la biblioteca en el sistema para poder luego conocer los datos del clima que se desprendan del análisis que realiza el sistema.

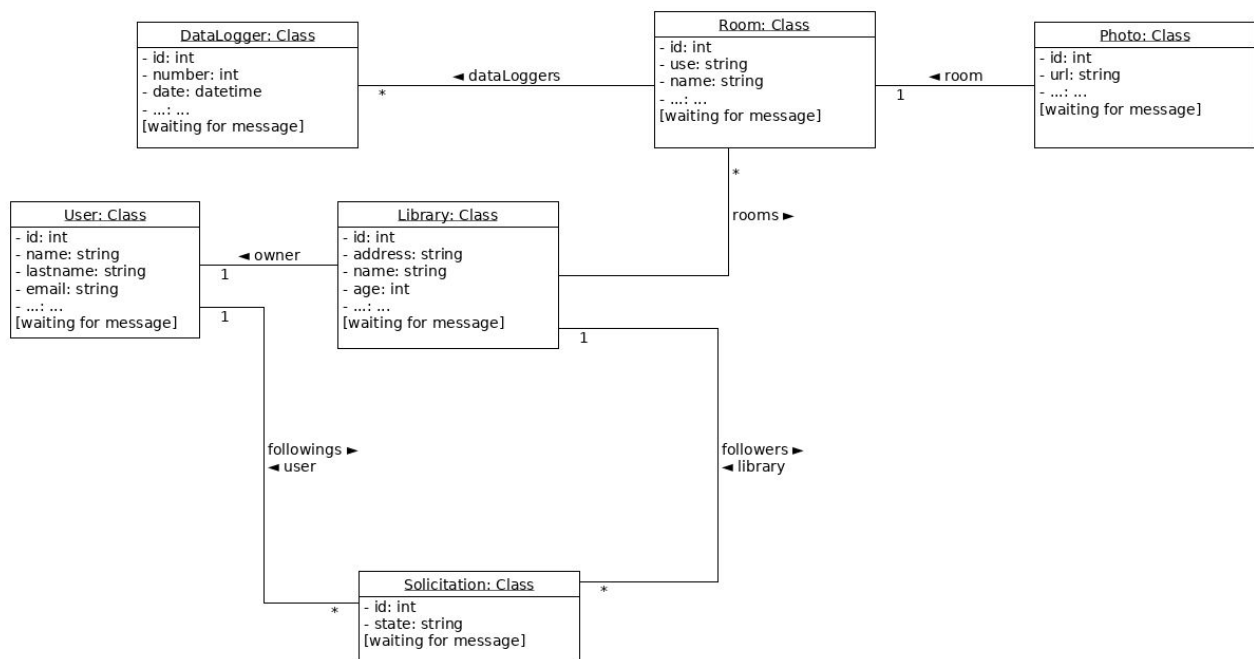
III. Como usuario registrado deseo visualizar para una sala con datos cargados:

A. Media móvil

B. Objetivo climático

IV. Como usuario de una biblioteca registrada en el sistema deseo registrarme como usuario autorizado a acceder a los datos de una biblioteca.

3. Modelo de diseño en UML



4. Tecnologías usadas en el desarrollo y consideraciones de implementación importantes

- Symfony3.4
- Doctrine
- JQuery
- PHP
- MySQL
- [Nginx | Apache2]
- Bootstrap
- AdminLTE
- Highchart

5. Instrucciones de instalación

a) Clonar el repositorio:

SSH: `git clone git@github.com:MeAgPaWeb/cpebbci.git`

HTTPS: `git clone https://github.com/MeAgPaWeb/cpebbci.git`

b) Preparar el proyecto:

- Entrar al directorio y darle permisos a caché, logs y sessions. Correr el siguiente comando:

```
cd cpebbci
```

```
bash # El siguiente filtro corre en bash
```

```
HTTPDUSER=`ps axo user,comm | grep -E
```

```
'[a]pache|[h]ttpd|[_]www|[w]ww-data|[n]ginx' | grep -v root | head -1  
| cut -d" " -f1`
```

```
sudo setfacl -R -m u:"$HTTPDUSER":rwX -m u:`whoami`:rwX var
```

```
sudo setfacl -dR -m u:"$HTTPDUSER":rwX -m u:`whoami`:rwX var
```

- Instalar las dependencias, en este punto se configura parameters.yml:

```
composer update
```

- Crear la bases de datos y las respectivas tablas:

```
php bin/console doctrine:database:create
```

```
php bin/console doctrine:schema:create
```

- Compilar y aplicar filtros de javascript y css:

```
php bin/console assetic:dump
```

- Iniciar el servidor:

```
php bin/console server:start
```

6. Reflexiones sobre usabilidad.

Ver informes

7. Trabajo futuro:

A. User stories que quedaron sin desarrollar:

I. Como usuario registrado deseo visualizar para una sala con datos cargados la resiliencia (estabilidad de clima en un día particular en una época del año particular, la actual o no.

II. Como usuario registrado deseo evaluar la condición actual (cargando los datos de sensado actuales) visualizando:

I. En qué porcentaje cumple con el objetivo climático, informando satisfacción de humedad relativa, satisfacción de temperatura y satisfacción de ambas variables en simultáneo.

II. Gráfico de comparación de la condición actual con el objetivo climático que tiene asignada la biblioteca

III. Como usuario registrado deseo modificar los parámetros de amplitud diaria.

B. Posibles stories que podrían agregarse

I. Gestor de dataloggers por sala. Pensar en la forma de manejar esa información.

II. Aceptar otros formatos de dataloggers

III. Posibilidad de exportar los gráficos a pdf

IV. Mejorar los cálculos para los gráficos. Posibilidad de realizar los cálculos en background.

V. Otros gráficos, aunque para esto sería necesario tener una mayor comprensión del dominio del problema