

1. Identificar y extraer las historias de usuario, y tareas si procediera, que derivan de la conversación reflejada en la introducción

Acceso a la aplicación: <http://incidence2.eu-gb.mybluemix.net/>

Usuario:										
Identificador	Resumen incidencia	Criticidad	Estado final	Usuario	Fecha reporte		Id - Resumen			
IN-0006	El usuario us001 incapaz de loguearse	MEDIA	FEEDBACK	gcastro	08/12/2015 13:00	Ver detalle	IN-0004 - El color de fondo del fondo provoca que no se visualice			
IN-0008	Sqj excepcion al generar el listado	BAJO	FEEDBACK	gcastro	08/12/2015 09:00	Ver detalle	Es necesario cambiar el fondo en algunos dispositivos no se visualiza			
							Estado	Comentario	Fecha	
IN-0004	El color de fondo del fondo provoca que no se visualice	BAJO	CERRADA	gcastro	09/12/2015 12:00	Ver detalle	ABIERTA		12/12/2015 12:45	
IN-0001	La url de la aplicación sin acceso	CRITICO	DUPLICADA	gcastro	11/12/2015 12:00	Ver detalle	RESOLVIENDO	se cambia el color por uno más claro	12/12/2015 13:45	
IN-0007	Null pointer excepcion	MEDIA	FEEDBACK	gcastro	08/12/2015 09:00	Ver detalle	RESUELTA	se integra el cambio	12/12/2015 16:45	
IN-0005	Al hacer click dos veces a la vez al botón se bloquea	BAJO	CERRADA	gcastro	09/12/2015 12:00	Ver detalle	FEEDBACK	se visualiza correctamente se puede cerrar	12/12/2015 17:45	
IN-0003	Pruebas unitarias de back-end app error	CRITICO	CERRADA	rgarcia	10/12/2015 12:00	Ver detalle	CERRADA	se cierra	13/12/2015 16:45	
IN-0002	Posicionamiento de la ventana incorrecto en dispositivo	CRITICO	CERRADA	rgarcia	10/12/2015 12:00	Ver detalle				
IN-0000	Error javascript	CRITICO	RESOLVIENDO	rlopez	11/12/2015 12:00	Ver detalle				

2. Aplicación web basada en la pila tecnológica: AngularJS que cubra al menos el lado cliente/frontend de la aplicación, con gestión en memoria de las entidades

Se ha desarrollado toda la capa front-end sobre angularJS, esto conlleva las siguientes funcionalidades:

- la construcción de la tabla.
- las llamadas get que recuperan Json de las entidades.
- la vinculación con los botones.

1-Desarrollo de backend con algún servicio en la nube.

Para la base de datos se ha empleado una base de datos distribuida no relacional no sql.

- Instancia Bd:
<https://51122191-f4f2-43bc-aa83-bd0878dcc1f1-bluemix.cloudant.com/incidencemg/>
- Instancia App, conectada a la bd y que accede al servicio de bd:
<http://incidence2.eu-gb.mybluemix.net/>
<http://incidence2.eu-gb.mybluemix.net/api/favorites> (acceso al json de listado sobre el que se construye la tabla principal)

Desde bluemix que es una plataforma cloud, se despliega por un lado la app java web la cuál accede al servicio cloudant el cuál conecta con la bd. Y la cuál contiene también la parte front sobre la que se visualiza la aplicación.

Proyecto web está compuesto fundamentalmente de los siguientes ficheros:

Parte java:

- ResourceServlet, provee el listado de incidencias en formato Json mediante la conexión CloudantClientMgr
- CloudantClientMgr, accede al servicio cloudant y provee el api a la conexión.

Parte web:

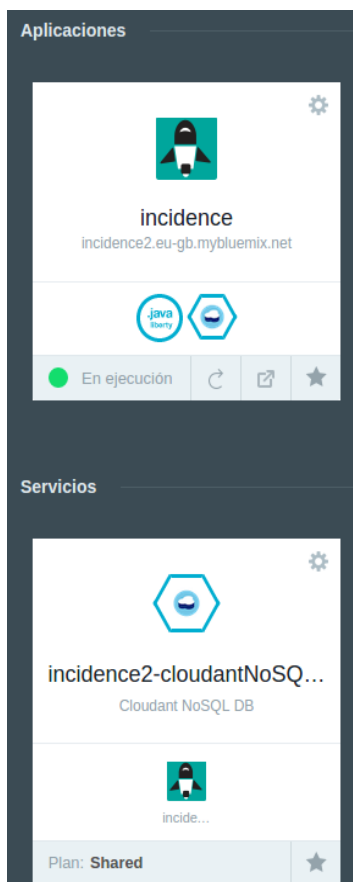
- index.html, constuye la interfaz html
- style.css, estilos necesarios

Despliegue la app en Bluemix. Sobre el código desarrollado se sube a la nube mediante los siguientes comandos, aunque parece que también hay conexión con git hub, pero esta opción no se ha usado.

- cf api https://api.eu-gb.bluemix.net
- cf login -u gustavo.castro@hotmail.es -o gustavo.castro@hotmail.es -s dev
- cf push incidence

Para poblar la base de datos se ha utilizado el plugin restclient para chrome. Se adjunta documento que contiene los documentos con los que se ha poblado la base de datos.

Se adjunta la imagen en el que se ve el esquema actual de la configuración.



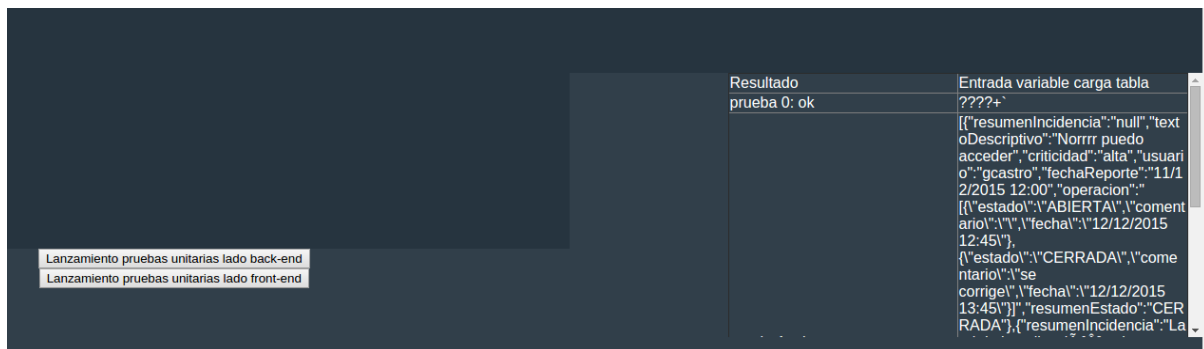
2-Diseño responsive o adaptativo

Se ha procedido a configurar todas las capas en relación a una proporción de la pantalla para que de esta forma sea dinámico para cada pantalla de cada dispositivo.

3-Pruebas unitarias del lado front

Se han desarrollado varias pruebas del lado front para poner a prueba el código javascript de carga de la tabla del listado de incidencias con varias entradas. Puesto que esta funcionalidad es la más crítica e importante del lado front.

Esta prueba se puede accionar desde el botón que se ve en la siguiente imagen. Y los resultados de la prueba a la derecha.



4- Pruebas unitarias del lado back-end

Se han desarrollado pruebas unitarias en la capa back-end que consisten en

- Probar que la conexión a cloudant del cliente java funciona correctamente
- Probar que se recuperan los documentos que refieren a las incidencias correctamente
- Probar que cada documento relacionado con la incidencia contiene el formato correcto para que sea visualizado sin causar problemas en la capa front

Si estas pruebas son correctas, cuando se integre los resultados de esta capa con la capa front los resultados serán los esperados.

Esta prueba se puede accionar desde el botón que se ve en la siguiente imagen.



5- Enganchar con algún servicio gratuito en la nube de integración continua.

Se ha procedido a utilizar travis CI, mediante conexión al repositorio git hub. Esta herramienta detecta que se han realizado una actualización en el repositorio. En cuanto lo detecta lanza la tarea ant test localizada en el fichero build.xml.

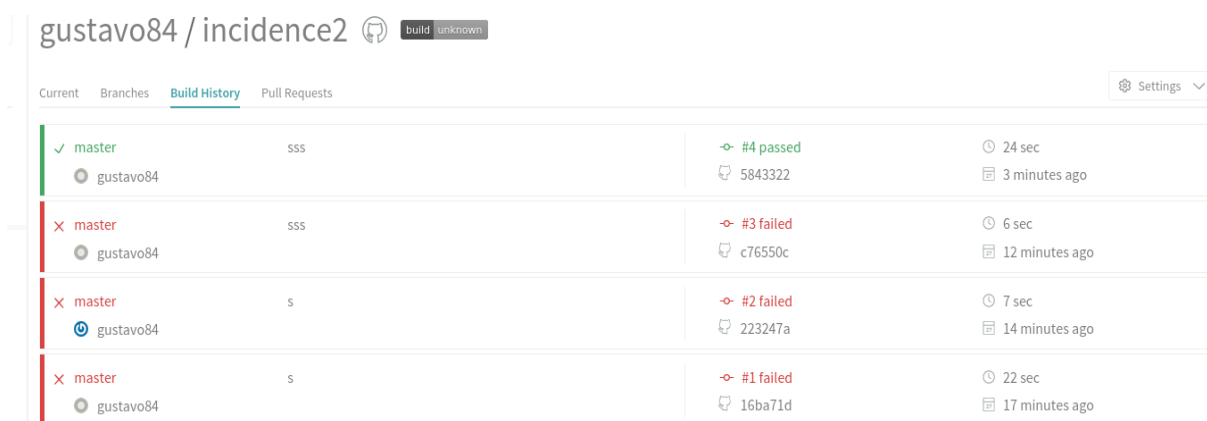
Se ha subido el código al repositorio siguiente de github, mediante línea de comandos git.

<https://github.com/gustavo84/incidence2>

<https://travis-ci.org/gustavo84/incidence2/builds>

En esta captura se muestra los push que se han realizado a github y se han detectado y ha lanzado la tarea build.xml de test.

En esta tarea se produce un compilado del proyecto. Por lo que se puede comprobar fácilmente si ha sido una subida correcta y es fácil y de forma automática detectar errores.



En la siguiente imagen se visualiza como al entrar en un item el resultado de la tarea ant.



gustavo84 / incidence2



build unknown

Current Branches Build History Pull Requests > **Build #4**

4

✓ master sss

Commit 5843322

Compare c76550c..5843322

gustavo84 authored and committed

~ #4 passed

⌚ Elapsed time 24 sec

🕒 less than a minute ago

```
1 Using worker: worker-linux-docker-lafd9ae3.prod.travis-ci.org:travis-linux-13
2
3 Build system information
67
68 $ git clone --depth=50 --branch=master https://github.com/gustavo84/incidence2.git gustavo84/incidence2
77
78 This job is running on container-based infrastructure, which does not allow use of 'sudo', setuid and setgid executables.
79 If you require sudo, add 'sudo: required' to your .travis.yml
80 See http://docs.travis-ci.com/user/workers/container-based-infrastructure/ for details.
81 $ jdk_switcher use oraclejdk8
82 Switching to Oracle JDK8 (java-8-oracle), JAVA_HOME will be set to /usr/lib/jvm/java-8-oracle
83 $ java -Xmx32m -version
84 java version "1.8.0_31"
85 Java(TM) SE Runtime Environment (build 1.8.0_31-b13)
86 Java HotSpot(TM) 64-Bit Server VM (build 25.31-b07, mixed mode)
87 $ javac -J-Xmx32m -version
88 javac 1.8.0_31
89 $ true
90 $ ant test
91 Buildfile: /home/travis/build/gustavo84/incidence2/build.xml
92
93 clean:
94 [delete] Deleting directory /home/travis/build/gustavo84/incidence2/bin
95 [delete] Deleting: /home/travis/build/gustavo84/incidence2/JavaCloudantApp.war
96
97 init:
98 [mkdir] Created dir: /home/travis/build/gustavo84/incidence2/bin
99
```