

Pruebas y resultados

CAPÍTULO 5

CAPÍTULO 5 – Pruebas y resultados

En este capítulo se van a mostrar las pruebas que se realizaron sobre el sistema y los resultados obtenidos. Las pruebas contempladas principalmente son tres: pruebas de integración de clases, pruebas de caja negra y prueba de valores frontera.

5.1 Pruebas de integración de clases

La prueba de integración fue realizada en la fase de desarrollo y se aplicó a el *Model* que es modulo principal del software desarrollado. En el *Model* se encuentra la lógica aplicativa del programa, por lo cual se decidió aplicar pruebas de integración sobre éste, es decir, a las clases que tienen interacción directa con *Hibernate*. Se desarrolló el módulo del modelo (ClasificacionModel.java, ProgramaModel.java, UsuarioModel.java e Hibernate.java) y se probaron las clases de manera aislada hasta no encontrar fallos en la respuesta del sobre los servicios prestados. Posteriormente se integró a la aplicación y la interacción que tuvo cubrió con la especificación debido a que tuvo un comportamiento definido.

5.2 Pruebas de caja negra

Las pruebas de caja negra se centraron a la parte funcional. Se verificó que se obtuvieran resultados esperados como respuesta de entradas apropiadas estas salidas involucraban la consistencia en la base de datos. La prueba consistió en evaluar diez funcionalidades del sistema como: registro de un usuario, guardar un programa, crear y borrar un usuario y clasificación, entre otras como a continuación se detalla:

Función probada	Resultado esperado		Resultados obtenido
1. Registro de usuario:	Información del usuario almacenada adecuadamente en la base de datos, permitiendo el acceso al sistema.	✓	
2. Guardar un programa	Generar la compilación del código fuente, indexar, almacenar la información en la base de datos así como el código fuente en la ruta deseada dentro del servidor para su adecuada consulta.	✓	
3. Visualizar código fuente colorado	El programa mostrado debe de corresponder al programa solicitado por el usuario.	✓	
4. Borrar un programa	El programa debe de ser borrado físicamente del servidor así como todo lo relacionado con el en la base de datos e índice de Lucene.	✓	
5. Crear una clasificación	Se debe de generar las clasificación en la base de datos para poder asignar programas a ésta.	✓	
6. Borrar una clasificación	Se debe de borrar la tupla de la base de datos así como los programas relacionados con la misma	×	Se borró la tupla de la base de datos pero los programas no.
7. Crear un nuevo usuario admin.	Se debe almacenar la información del nuevo para permitir su acceso al sistema.	✓	
8. Borrar un usuario	Se debe de borrar la tupla de la base de datos así como los programas relacionados con la misma	×	Se borró la tupla de la base de datos pero los programas no.

9. Realizar consulta por texto	Se debe de poder verificar que las palabras proporcionadas al sistema se encuentran en los resultados de la consulta.	✓	
10. Consulta por categoría	Los programas recuperados deben de pertenecer a la categoría solicitada.	✓	

La tabla anterior se observa como en dos ocasiones no se obtuvieron los resultados deseados. Los problemas fueron resueltos y se aplicó la prueba nuevamente y se obtuvieron los resultados deseados en todos los casos, dando así por terminada la prueba.

5.3 Prueba de valores frontera

Se aplicó para probar situaciones extremas o inusuales, específicamente para dos casos:

- En las palabras que se reciben en la consulta: La prueba consistió en pedir a cinco usuarios introdujeran palabras sin sentido, espacios, signos raros. Las palabras, espacios y algunos signos si fueron soportados por el sistema, pero los siguientes signos: ?, “, (,) provocaron que el sistema enviara una excepción, la cual fue corregida y posteriormente se aplico la misma prueba y no se encontró alguna excepción el sistema. Dando por terminada la prueba.
- En el archivo a subir. La prueba consistió en pedir a cinco usuarios trataran de subir algún tipo de archivo diferente al aceptado por la aplicación. Primero se seleccionó un archivo que no era .java: al no tener la extensión Java el sistema no intento subir el archivo al servidor en todas las ocasiones, e informo al usuario de este error La segunda prueba consistió en no seleccionar ningún archivo en el cual el sistema informo al usuario de este error. Finalmente se le cambio la extensión a un archivo de imagen a .java el sistema intento subir el archivo por coincidir la extensión con la esperada, pero debido a que las imágenes son archivos muy grandes y el sistema limita

el tamaño de los archivos que va a subir al sistema a 10KB es por lo cual los usuarios, no pudieron subir el archivos. Se consideró que el usuario pudiera subir un archivo con otro contenido que no fuera un programa de Java, pero tendría que engañar al sistema cambiando la extensión del archivo y que su extensión fuera a lo mucho de 10KB, pero a pesar de poder lograrlo el sistema trataría de compilarlo y al no lograrlo el sistema borraría el archivo automáticamente. En esta prueba se concluyo como satisfactoria debido a que se tienen consideradas las posibles entradas al sistema para el archivo.