



GRUPO 8

COMPATIFY



Participantes:

Emilio Gómez Esteban, emige19@uma.es

Fernando Javier López Cerezo, lopezalhaurin@uma.es

Enrique Pérez Haro, enriqueperezharo@uma.es

Alberto Trigueros Postigo, albertotrigueros22@uma.es

Jesús Fuentes Moya, jesusfuentesm4@uma.es

María Peinado Toledo, mariapt02@uma.es

Antonio Trujillo Reino, antoniotrujillo@uma.es

Juan José Rodríguez Hernández, jjrodriguezhernandez@uma.es



TABLA DE CONTENIDOS

Introducción	3
Asignación de roles.....	4
Gestión de riesgo	5
Límite de peticiones.....	5
Falta de experiencia	5
Enfermedad o baja del personal	6
Restricción de usuarios	6
Cierre de la API Spotify Developers.....	6
Planificación	7
Power-ups	8
Requisitos	10
Requisitos Funcionales	10
Requisitos No Funcionales	12
Casos de uso	13
Modelo del dominio	18
Diagramas de secuencias.....	19
Plan de pruebas	24
Herramientas software.....	27



Hoy en día la música se ha convertido en algo indispensable para la vida diaria y Spotify es la principal aplicación para ello. ¿Te gustaría conocer gente afín con tus gustos musicales? ¿Nunca has pensado comparar tus canciones favoritas con las de tus amigos? Con este propósito nace Compatify, software online de uso gratuito.

Nuestra aplicación se trata de un software que estima la compatibilidad musical entre dos individuos. Cada usuario se registra en la aplicación, añade su cuenta de Spotify, y mediante su historial de reproducción se podrá obtener el porcentaje de similitud con otro usuario de Compatify, usando aspectos como artistas y canciones. Finalmente, se mostrarán los artistas en común, canciones y porcentaje de compatibilidad, con la posibilidad de reproducir las canciones desde la web.

En cuanto a los posibles roles que pueden adoptar los participantes en este proyecto tendremos en cuenta los siguientes:

- Analista: especifica la documentación independiente del modelo de implementación.
- Diseñador: especifica la documentación dependiente del modelo de implementación.
- Programador: implementa el programa en los lenguajes y marcos de trabajo escogidos.
- Jefe de proyecto: coordina las operaciones y se responsabiliza de las decisiones tomadas.
- Ingeniero de pruebas: diseña e implementa los casos de prueba unitarios, de integración, etc.

Que un integrante tenga un rol significa que será responsable de esa tarea, sin embargo, todos los participantes estarán informados de todas las tareas realizadas en el proyecto. De esa manera, la distribución de roles en primera instancia puede quedar de la siguiente manera:

Analista	Diseñador	Programador	Jefe de proyecto	Ingeniero de pruebas
Jesús Fuentes	Emilio Gómez	Alberto Trigueros	Fernando Javier López	Alberto Trigueros
Juan José Rodríguez	Jesús Fuentes	Enrique Pérez	Enrique Pérez	Juan José Rodríguez
Fernando Javier López	María Peinado	María Peinado	Antonio Trujillo	Antonio Trujillo
		Emilio Gómez		



Se identificarán los riesgos, es decir, los posibles contratiempos que afecten al desarrollo del proyecto y preparar un plan que minimice sus efectos.

Se abordarán los siguientes tipos de riesgo:

- Los riesgos del proyecto que afectan a la planificación y a los recursos.
- Los riesgos del producto que afectan a la calidad y al rendimiento del software bajo desarrollo.
- Los riesgos del negocio que afectan a la organización que desarrolla el software.

Límite de peticiones

- Descripción: La API de Spotify establece un límite al número de peticiones que se pueden hacer en una franja de treinta segundos. Si se excede el límite, se reciben respuestas de error 429. Spotify no especifica cuál es este límite.
- Tipo: Producto y negocio
- Probabilidad: Moderada
- Efectos: Serio
- Estrategia de gestión: Intentar optimizar el número de peticiones que se realizan. Spotify tiene para ello algunas recomendaciones en su página web.

Falta de experiencia

- Descripción: Somos un equipo con poca experiencia en el desarrollo de software, lo cual puede provocar una mayor dificultad y retraso para elaborar el código.
- Tipo: Proyecto
- Probabilidad: Alta
- Efectos: Tolerable
- Estrategia de gestión: Recurrir a recursos webs y tutoriales, así como preguntar a los profesores.

Enfermedad o baja del personal

- Descripción: Algún miembro del equipo puede caer enfermo o abandonar el proyecto, lo que provocaría un retraso en la ejecución del proyecto.
- Tipo: Proyecto
- Probabilidad: Baja
- Efectos: Tolerable
- Estrategia de gestión: Todo el equipo debe saber qué tareas realizan los demás para que pueda haber mayor flexibilidad y se pueda retomar el trabajo hecho por otro integrante. También es buena idea llevar a cabo buenas prácticas de programación como puede ser comentar adecuadamente el código.

Restricción de usuarios

- Descripción: Spotify tiene dos modos para las apps que usan su API, el modo en desarrollo y el modo de cuota extendida. Las aplicaciones inicialmente están en el primer modo, el cual requiere que los usuarios estén en una whitelist y limita el número de usuarios autenticados a 25.
- Tipo: Negocio
- Probabilidad: Baja (Poca probabilidad de que la app sea usada por más de 25 usuarios distintos)
- Efectos: Serio
- Estrategia de gestión: En caso de que queramos lanzar la aplicación al mercado y pueda ser usada por un mayor número de usuarios, habría que rellenar un formulario para que el equipo de Spotify apruebe que la aplicación pase al modo de cuota extendida, que no tiene límite de usuarios.

Cierre de la API Spotify Developers

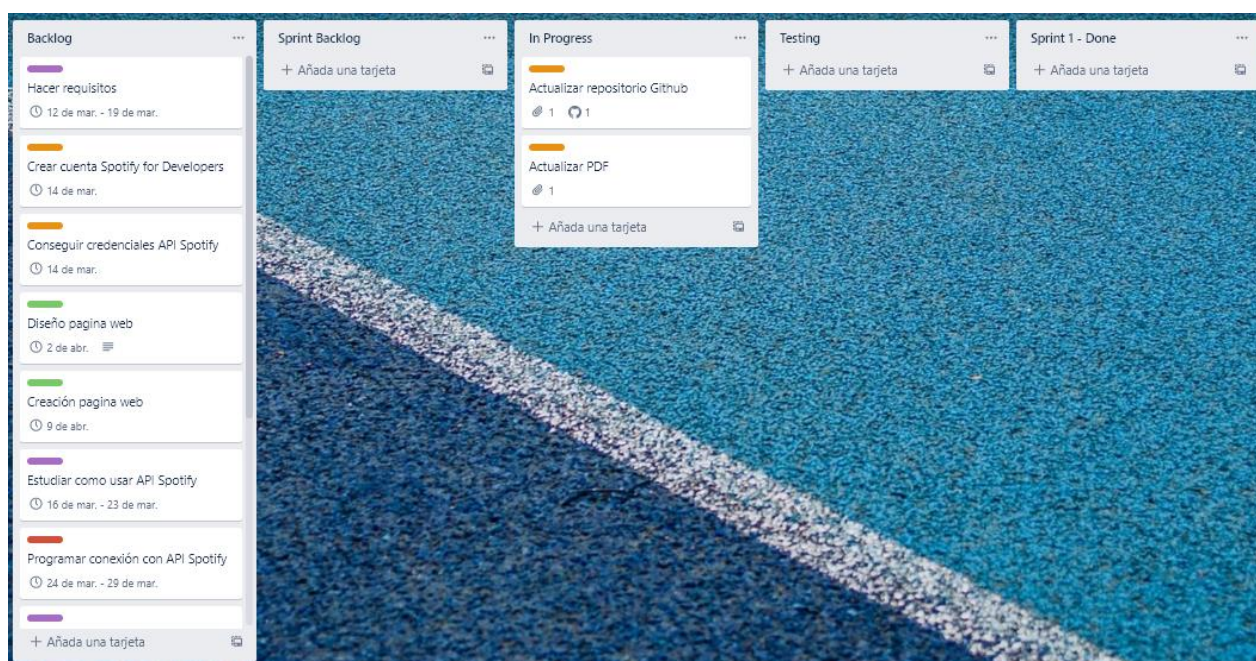
- Descripción: El cierre temporal o definitivo de la API Spotify por la empresa. Esto nos lleva a perder el acceso a la información de cada usuario y quedaría la actividad paralizada.
- Tipo: Producto
- Probabilidad: Muy baja (No ha sido anunciado tal suceso)
- Efectos: Catastrófico
- Estrategia de gestión: Contacto con el equipo de marketing de Spotify con el fin de exponer la idea de proyecto y que se nos permita el acceso a dichos datos, implementado como extensión de la aplicación o como proyecto propio.



Nos enfrentamos a un proyecto del que no conocemos mucho al ser novatos en este tipo de desarrollo, con lo que inevitablemente en ciertos momentos es muy probable que haya incertidumbre. Por ello, necesitamos un proceso ágil que nos permita en esos momentos ser capaces de rectificar, adaptarnos y priorizar de nuevo los procesos. Por todo esto hemos elegido la metodología “Scrum”, ya que tras haber analizado los diferentes modelos de proceso software existentes, este es el más flexible y el que más se adapta al cambio. Esto se debe a que “Scrum”:

- Fomenta la comunicación entre los miembros.
- Optimiza el tiempo.
- Adaptable a cambios y cambios de prioridades.
- Ciclos cortos lo cual disminuye los riesgos.
- Todo el equipo participa en todas las fases.

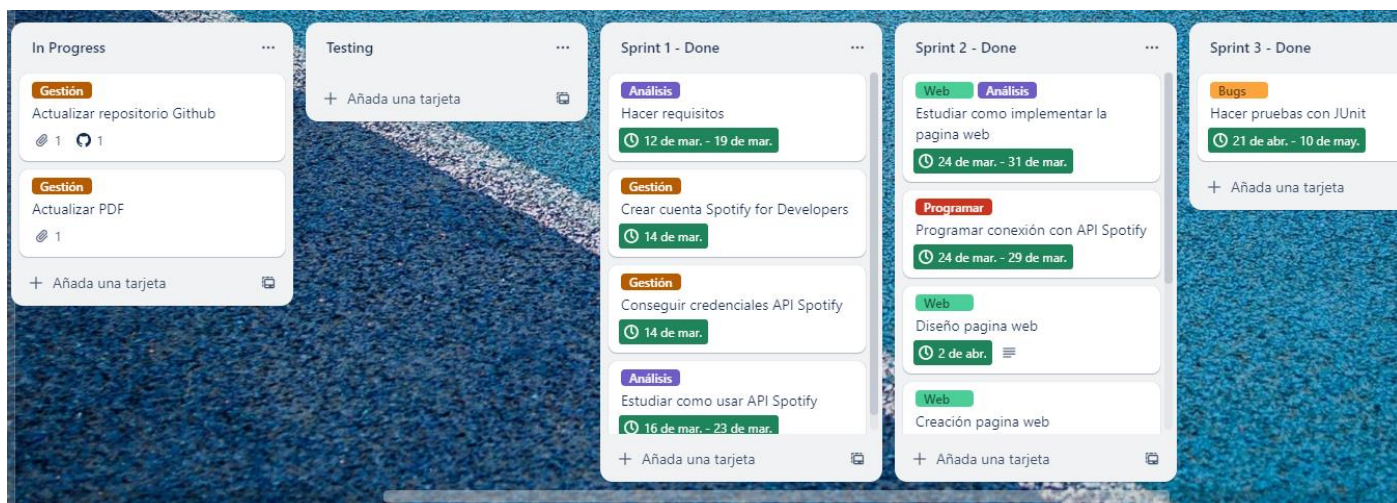
Por todo esto hemos decidido que este método es el óptimo para nuestro proyecto.



En la anterior imagen podemos observar la captura inicial del proyecto en “Trello”, un software de planificación que nos ayudará durante todo el proceso de desarrollo. Se compone de diferentes tareas y columnas que, al ser “Scrum” la metodología empleada, se componen de:

- **Backlog:** Documento a alto nivel del proyecto que reúne todos las tareas o requisitos.
- **Sprint backlog:** Subconjunto de requisitos que serán desarrollados en el siguiente sprint. Hemos escogido una duración de sprint de dos semanas.
- **In progress:** Requisitos desarrollados en este momento del actual sprint.
- **Testing:** Pruebas y revisión de las tareas antes de darlas por terminadas.
- **Sprint 1-Done:** Tareas que se han completado en el primer proceso de sprint.
- **Sprint 2 - Done:** Tareas que se han completado en el segundo sprint.
- **Sprint 3 - Done:** Tareas que se han completado en el último sprint, finalizando así la planificación del proyecto.

A continuación, mostramos la imagen de cómo ha terminado nuestro repositorio “Trello”, tras todos los sprints realizados.



Power-ups

Una de las funcionalidades de Trello, integraciones que se añaden al tablero central para mejorar la experiencia del usuario, como pueden ser gráficos o conexiones con distintas aplicaciones. Los usados en el proyecto son los siguientes:

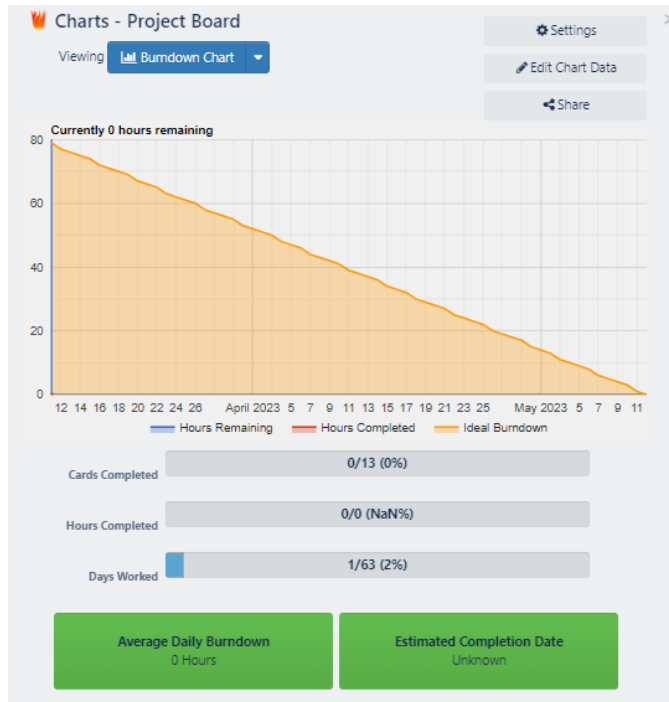
- **Git**



- **Google Drive**



- Burndown Chart





A continuación, describiremos los requisitos de nuestra aplicación. Estos se tratan de los servicios que proporcionamos y las restricciones que hay en la operación de la App.

Requisitos Funcionales

Consideramos como requisitos funcionales aquellos que describen los servicios que se espera que el sistema suministre a los usuarios.

RF1: El sistema proporcionará visores adecuados.
Como desarrollador,
se quiere que una vez hecha la comparativa entre dos usuarios de spotify, los resultados sean mostrados de forma clara, con % y artistas/canciones comunes.

Pruebas de aceptación:

- Se harán pruebas con distintos compañeros del equipo con alta, media y baja compatibilidad musical esperando grandes diferencias en los porcentajes y cantidad de artistas/canciones mostrados.

RF2: Crear cuenta
Como usuario,
se quiere poder crear cuenta en Compatify para acceder a la página web.

Pruebas de aceptación:

- Introducir información en el formulario de creación de cuenta (nombre de usuario, correo electrónico, fecha de nacimiento, género y contraseña) y comprobar que se guarda la información del usuario en la base de datos.

RF3: Asociar cuenta
Como usuario,
se puede iniciar sesión en Spotify desde Compatify para que se descarguen los datos de la cuenta en la aplicación

Pruebas de aceptación:

- Se pulsa el botón "Asociar", se inicia sesión en Spotify y se observa que en la base de datos se ha guardado la información del usuario de Spotify en cuestión.

RF4: Botón actualizar
Como usuario,
se quiere poder actualizar la información de la cuenta Spotify
para poder realizar las comparaciones con información más reciente.

Pruebas de aceptación:

- Se pulsa el botón "actualizar" y se observa que la base de datos se ha actualizado con la información más reciente del usuario de Spotify.

RF5: Búsqueda de Usuario
Como usuario,
se quiere poder buscar otro usuario que pertenezca a la base de datos.

Pruebas de aceptación:

- Se crean dos usuarios de Compatify y desde la cuenta de uno se intenta buscar al otro, tras lo cual éste aparece como resultado de búsqueda.

RF6: Iniciar sesión
Como usuario,
se quiere poder iniciar sesión en Compatify y que conecte correctamente con la cuenta

Pruebas de aceptación:

- Se inicia sesión con una cuenta ya creada tras lo cual se accede exitosamente al perfil de dicha cuenta.

RF7: Comparativa
Como usuario,
se quiere poder realizar una comparación de mi historial musical con cualquier otro usuario.

Pruebas de aceptación:

- Un usuario con cuenta de Spotify asociada intenta compararse con otro que también tiene cuenta de Spotify asociada y se obtiene los artistas y canciones en común, así como el porcentaje de similitud.

Requisitos No Funcionales

Consideramos como requisitos no funcionales aquellos que describen las restricciones sobre los servicios ofrecidos por el sistema.

RNF1: Seguridad

Como usuario, quiero que no se pueda acceder a mi cuenta al menos que se conozca el nombre de usuario y contraseña de la cuenta para que así mi información esté segura.

RNF2: Facilidad de uso

Como usuario, se quiere que la aplicación sea fácil e intuitiva de usar para no que no se necesite mucho tiempo en aprender a usarla.

RNF3: Rapidez

Como usuario, quiero que la página web funcione de forma eficiente para no tener que esperar mucho tiempo a que cargue.

RNF4: Fiable

Como usuario, quiero que ocurran la menor cantidad de fallos en las funcionalidades.

RNF5: Calidad de las comparaciones

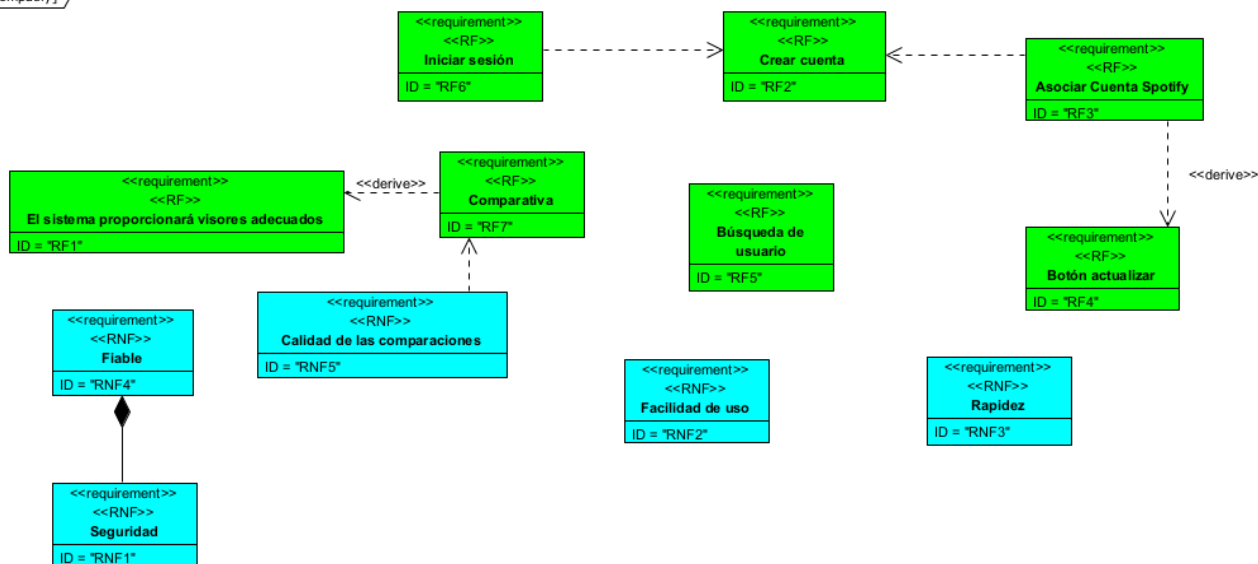
Como usuario, quiero que al realizar comparaciones con otros usuarios se refleje si realmente tenemos gustos parecidos o no.

Pruebas de aceptación:

- Hacer una comparación y que no refleje como gusto en común un artista que apenas se ha escuchado.

Con ayuda del software Visual Paradigm el cual proporciona diagramas para requisitos con sus respectivas relaciones, podemos verlos visualmente como sigue:

req [Compatibility]





- **Identificador único**

- CU1. Crear cuenta de Compatify

- Contexto de uso

- Cuando un usuario quiere usar nuestro software por primera vez, podrá crearse una cuenta.

- Precondiciones y activación

- El usuario está conectado al sistema y en el formulario de creación de cuenta donde introducirá sus datos.

- Garantías de éxito / Postcondición

- Se logra crear un nuevo usuario en el sistema y el usuario es redirigido al inicio de sesión.

- Escenario principal

1. El usuario pulsa la opción “Suscríbete a Compatify”.
2. El sistema muestra un formulario para crear una nueva cuenta.
3. El usuario introduce sus datos, los cuales son un nombre de usuario no utilizado, un correo, una contraseña de 8 o más caracteres, una fecha de nacimiento correcta y su género.
4. El usuario pulsa la opción “Regístrate”.
5. La nueva cuenta se guarda en la base de datos y se redirige al usuario a la página de inicio sesión.

- Escenarios alternativos

- El nombre de usuario introducido es vacío o ya existe un usuario con ese nombre, mostrando un mensaje de error.
- El correo está vacío o ya está usado y se muestra un mensaje de error.
- La contraseña no tiene 8 o más caracteres y se muestra un mensaje de error.
- La fecha introducida no tiene formato numérico, no existe, es una fecha futura o anterior a 1900, mostrando error.

- **Identificador único**

- CU2. Iniciar sesión en Compatify

- Contexto de uso

- Cuando un usuario quiere iniciar sesión con su cuenta de Compatify.

- Precondiciones y activación

- El usuario se encuentra en el formulario de inicio de sesión.

- Garantías de éxito / Postcondición

- Se inicia sesión y se redirige al usuario a la página de perfil.

- Escenario principal

1. El usuario pulsa en “Iniciar sesión”.
2. El sistema muestra el formulario de inicio de sesión.
3. El usuario introduce su nombre de usuario y contraseña.
4. El usuario es redirigido a la página de perfil.

- Escenarios alternativos

- El usuario introduce un nombre de usuario que no existe y se muestra un mensaje de error.
- El usuario introduce un nombre de usuario existente, pero una contraseña errónea, mostrando un mensaje de error.

- **Identificador único**

CU3. Cambiar contraseña.

- Contexto de uso

Cuando un usuario quiera cambiar su contraseña, podrá cambiar su contraseña en nuestra página web.

- Precondiciones y activación

El usuario ha iniciado sesión y está en el formulario de cambiar contraseña.

- Garantías de éxito / Postcondición

Se cambia la contraseña de acceso.

- Escenario principal

1. El usuario se dirige a la pestaña “Cambiar contraseña”.
2. El usuario introduce la contraseña antigua.
3. El usuario introduce una nueva contraseña, la cual tiene 8 caracteres o más.
4. El usuario vuelve a introducir la nueva contraseña.
5. El usuario pulsa en “Establecer nueva contraseña”.
6. El sistema modifica la contraseña en la base de datos.
7. El sistema redirige al usuario a la página del perfil.

- Escenarios alternativos

- La contraseña nueva no tiene 8 caracteres o más y se muestra un mensaje de error.
- La contraseña antigua no es correcta y se le comunica al usuario.
- La contraseña nueva y su confirmación no coinciden, por lo que se muestra un mensaje de error al usuario.

- **Identificador único**

CU4. Actualizar perfil de Compatify

- Contexto de uso

Cuando un usuario quiere cambiar sus datos personales de su cuenta de Compatify.

- Precondiciones y activación

El usuario está conectado en el sistema y en el formulario de actualización de información.

- Garantías de éxito / Postcondición

Se actualiza la información personal del sistema.

- Escenario principal

1. El usuario pulsa la opción “Editar perfil” de la cuenta.
2. El sistema muestra un listado de información con campos rellenos con los datos personales del usuario (correo electrónico, nombre de usuario...).
3. El usuario cambia los campos deseados con los nuevos datos.
4. El usuario pulsa la opción de Actualizar datos.
5. Se actualizan los datos del usuario en la base de datos.
6. Se redirige al usuario al perfil.

- Escenarios alternativos

- Se intenta cambiar el nombre de usuario a uno que ya existe o a uno vacío y se muestra un error.
- El campo de correo se deja vacío o se modifica a uno que ya está usado, tras lo cual se muestra un error.
- La fecha de nacimiento introducida no es correcta y se muestra un error.

- **Identificador único**

CU5. Asociar cuenta de Spotify

- Contexto de uso

Cuando un usuario quiere enlazar su cuenta de Spotify con la de Compatify.

- Precondiciones y activación

El usuario está conectado en el sistema y en la página de actualizar cuenta.

- Garantías de éxito / Postcondición

Se asocia a la cuenta de Compatify la información del usuario de Spotify con el que se quiere enlazar.

- Escenario principal

1. El usuario se dirige a la pestaña “Actualizar cuenta”.
2. El usuario pulsa en “Asociar”.
3. El usuario es redirigido a Spotify.
4. El usuario inicia sesión en Spotify en caso de no tener sesión iniciada y concede los permisos solicitados a nuestra aplicación.
5. Spotify devuelve un código de autorización.
6. Se solicita a Spotify el token de acceso con el código recibido.
7. Spotify devuelve el token de acceso.
8. Se hacen llamadas a la API de Spotify para obtener la información del usuario de Spotify, como su nombre de usuario, correo, canciones más escuchadas y artistas más escuchados.
9. Se guarda esta información en la base de datos asociándose al usuario de Compatify.
10. Se redirige al usuario al perfil.

- Escenarios alternativos

- El usuario de Spotify no está registrado en el dashboard de la app y se obtiene un error de la API al intentar obtener información del usuario con el token.

- **Identificador único**

CU6. Actualizar información de Spotify

- Contexto de uso

Cuando un usuario quiere actualizar los datos de su cuenta podrá hacerlo desde nuestro software.

- Precondiciones y activación

El usuario está conectado en el sistema con la sesión iniciada.

- Garantías de éxito / Postcondición

Se actualiza la información en el sistema.

- Escenario principal

1. El usuario entra en la opción “Actualizar cuenta”.
2. El usuario pulsa la opción “Actualizar”.
3. Se solicita a la API los artistas y canciones más escuchados mediante el token de acceso.
4. Se actualiza dicha información en la base de datos.
5. El usuario es redirigido al perfil.

- Escenarios alternativos

- El token ha caducado y se usa el token de refresco para solicitar un nuevo token y repetir las llamadas a la API.
- Si el token de la cuenta y el de refresco han caducado, se redirige al usuario al perfil sin llevar a cabo la actualización.

- **Identificador único**

CU7. Comparar con otro usuario de Compatify

- Contexto de uso

Cuando un usuario quiere comparar sus gustos musicales con otro usuario de Compatify.

- Precondiciones y activación

El usuario ha iniciado sesión con su cuenta de Compatify.

El usuario ha buscado al usuario con el que quiere compararse.

- Garantías de éxito / Postcondición

Se elabora satisfactoriamente un informe que refleja el resultado de la comparación entre los usuarios.

- Escenario principal

1. El usuario selecciona el usuario con el que quiere compararse de entre los resultados de su búsqueda.
2. El usuario pulsa sobre el botón de comparar.
3. Se realiza una comparación de los artistas y canciones más escuchados de ambos usuarios.
4. Se muestra el porcentaje de compatibilidad, así como los artistas y canciones comunes, con la posibilidad de reproducir dichas canciones.

- Escenarios alternativos

- Si el usuario que quiere compararse con otro o el seleccionado en la búsqueda no tienen cuenta de Spotify asociada se muestra un error.

- **Identificador único**

CU8. Buscar usuario de Compatify

- Contexto de uso

Cuando un usuario quiere buscar a otro usuario mediante el buscador de Compatify.

- Precondiciones y activación

El usuario ha iniciado sesión con su cuenta de Compatify.

- Garantías de éxito / Postcondición

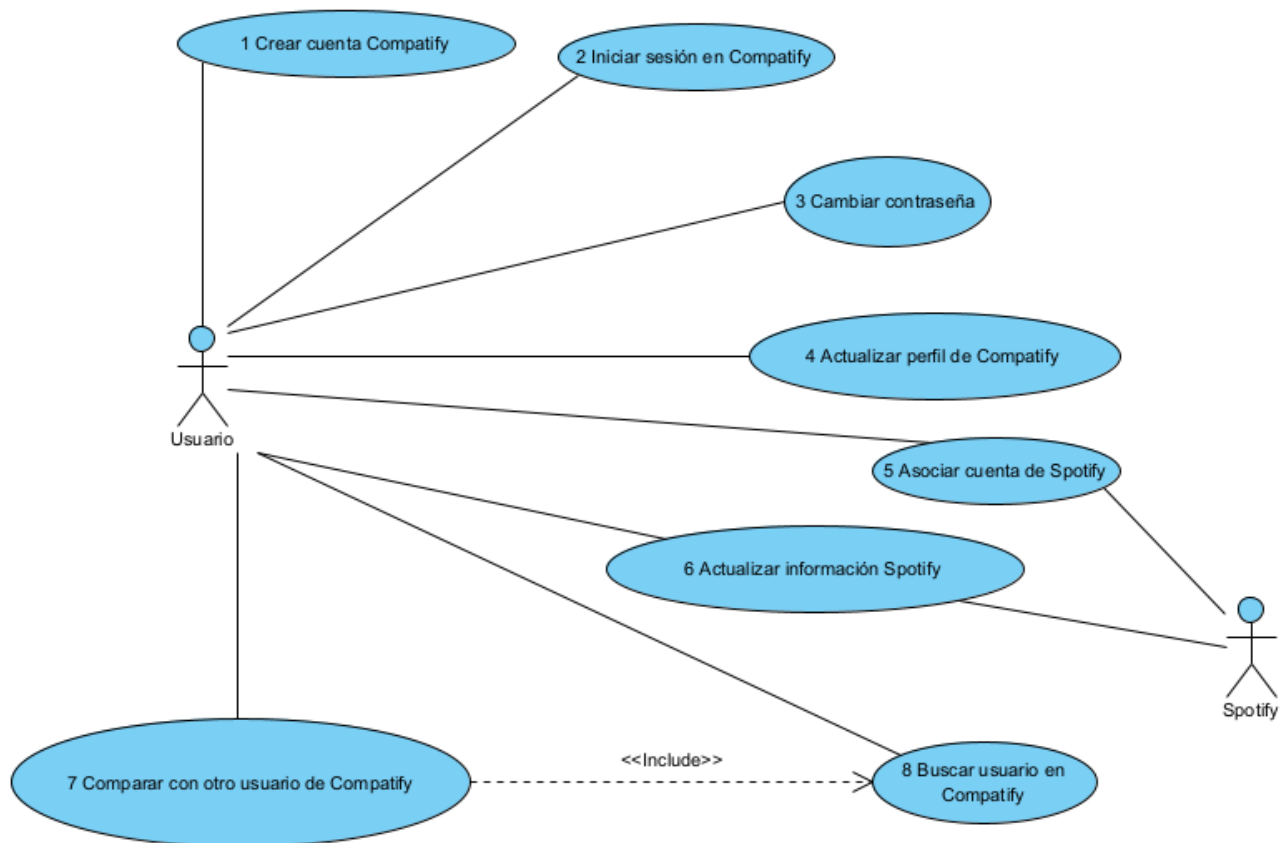
Se realiza la búsqueda, mostrando los usuarios de Compatify que incluyan en su nombre de usuario la cadena introducida en el buscador.

- Escenario principal

1. El usuario pulsa sobre la sección "Comparar"
2. El usuario pulsa en el campo "Introduzca un nombre"
3. El usuario escribe una cadena de caracteres.
4. El usuario presiona la tecla Enter.
5. Se muestran los usuarios que contienen dicha cadena en su nombre.

- Escenarios alternativos

- El resultado de la búsqueda es vacío, mostrando el error correspondiente.



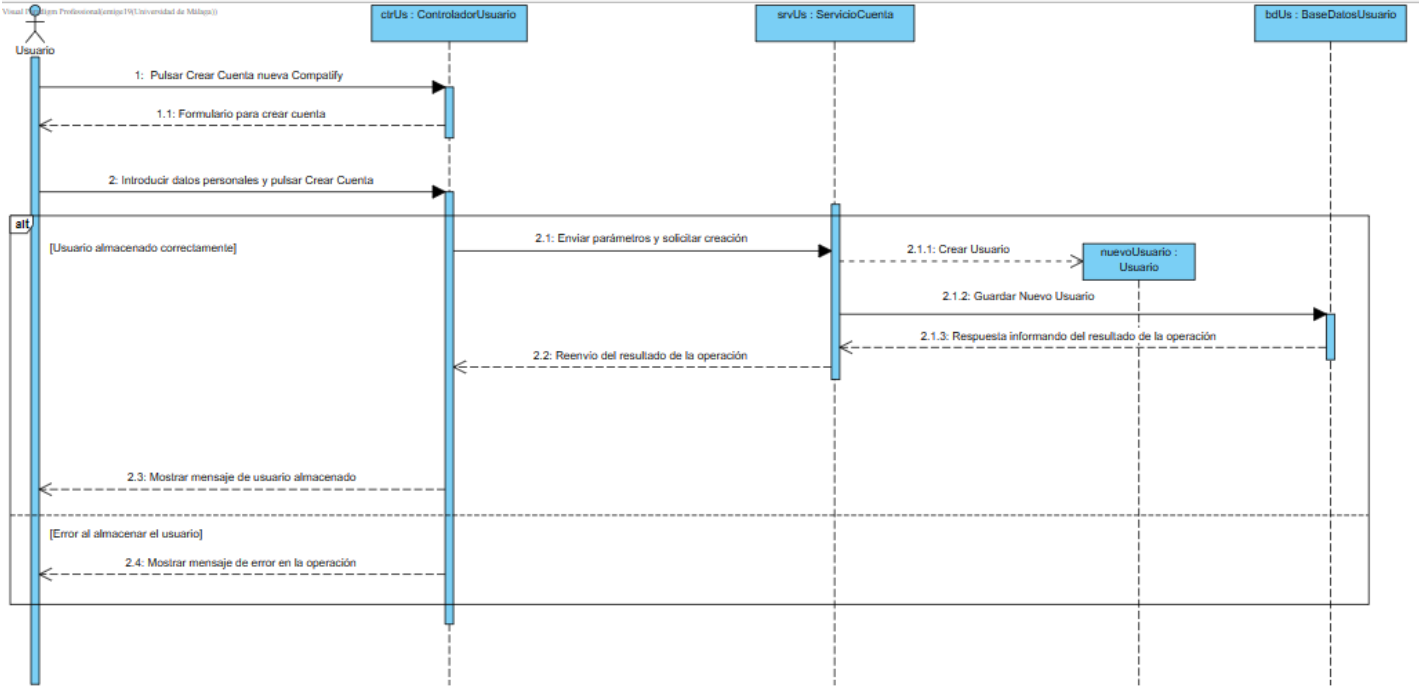




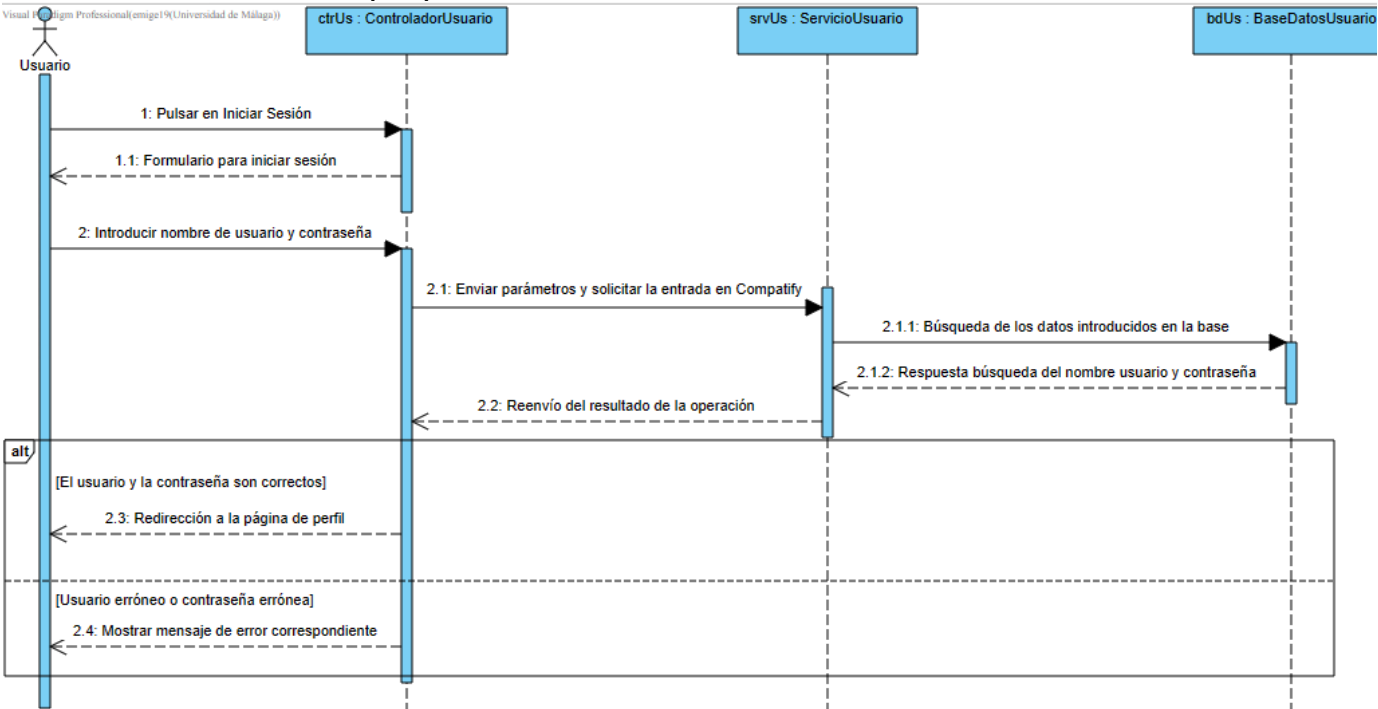
DIAGRAMAS DE SECUENCIA



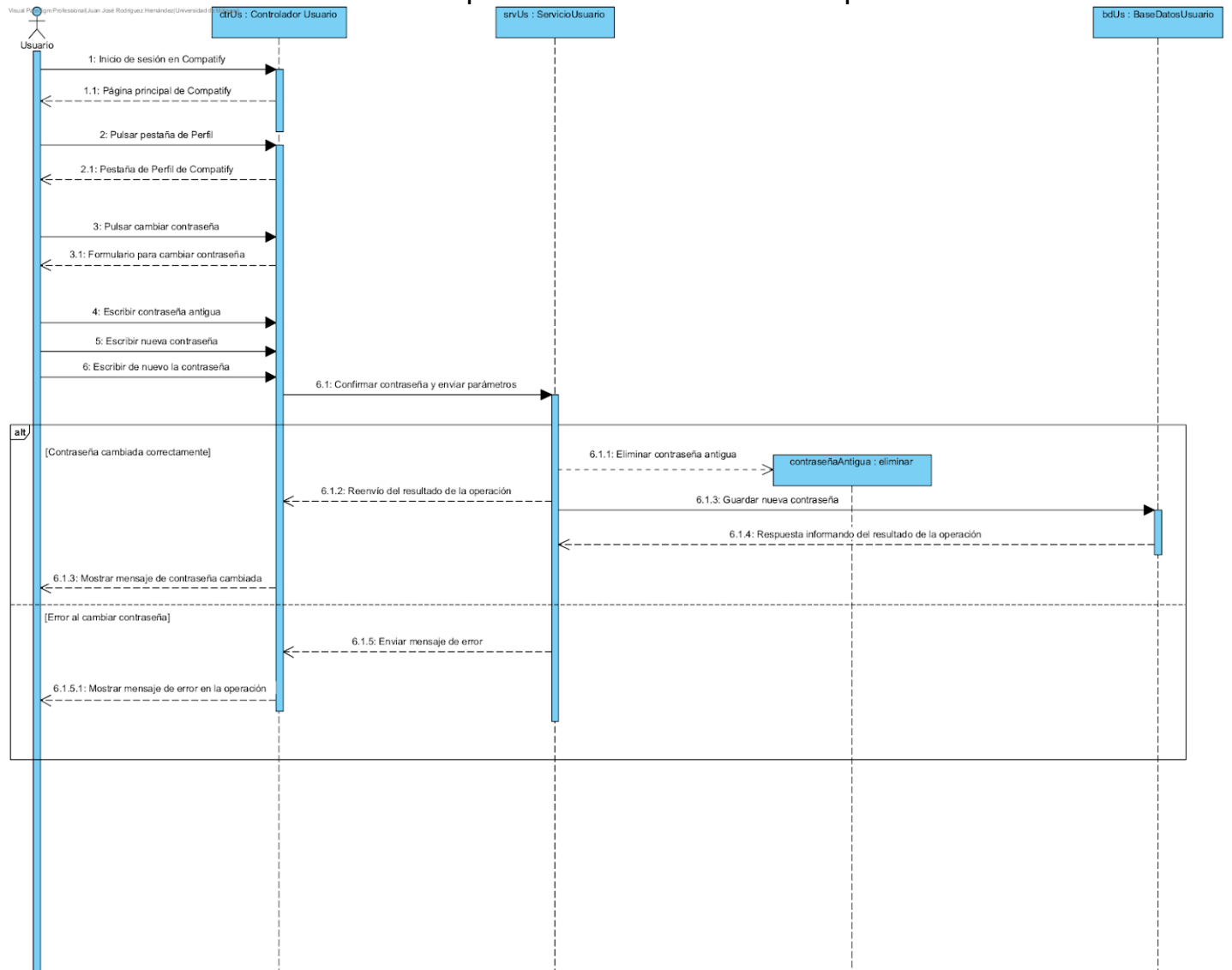
Crear cuenta de Compatify: este diagrama refleja los pasos que tendría que realizar un nuevo usuario de Compatify que decide entrar en la página y crear una cuenta en nuestro sistema. La persona introducirá sus datos y nuestro sistema se encargará de ponerlo todo en orden y comunicar a todas las partes del proyecto. Se han tenido en cuenta que pueden darse los escenarios alternativos y que en ese caso se informaría correctamente al usuario, al igual que en los casos donde todo se ejecuta con normalidad.



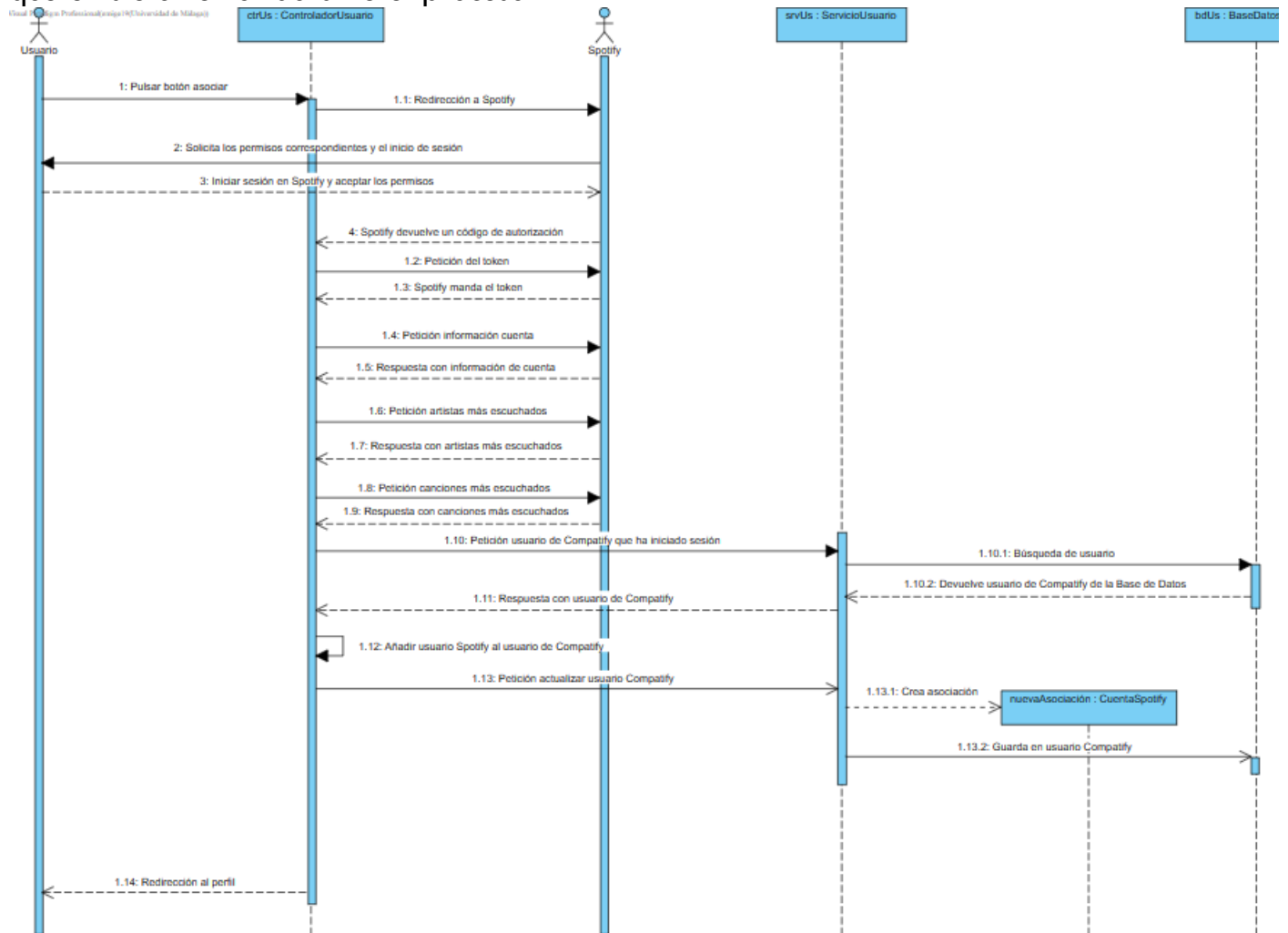
Iniciar sesión en Compatify: el siguiente diagrama mostrará la secuencia de trabajo que se sigue cuando un usuario que ya ha creado una cuenta de Compatify desea iniciar sesión con sus datos (nombre de usuario y contraseña). Como en el caso anterior se han tenido en cuenta los escenarios adversos que pueden suceder en una de estas situaciones.



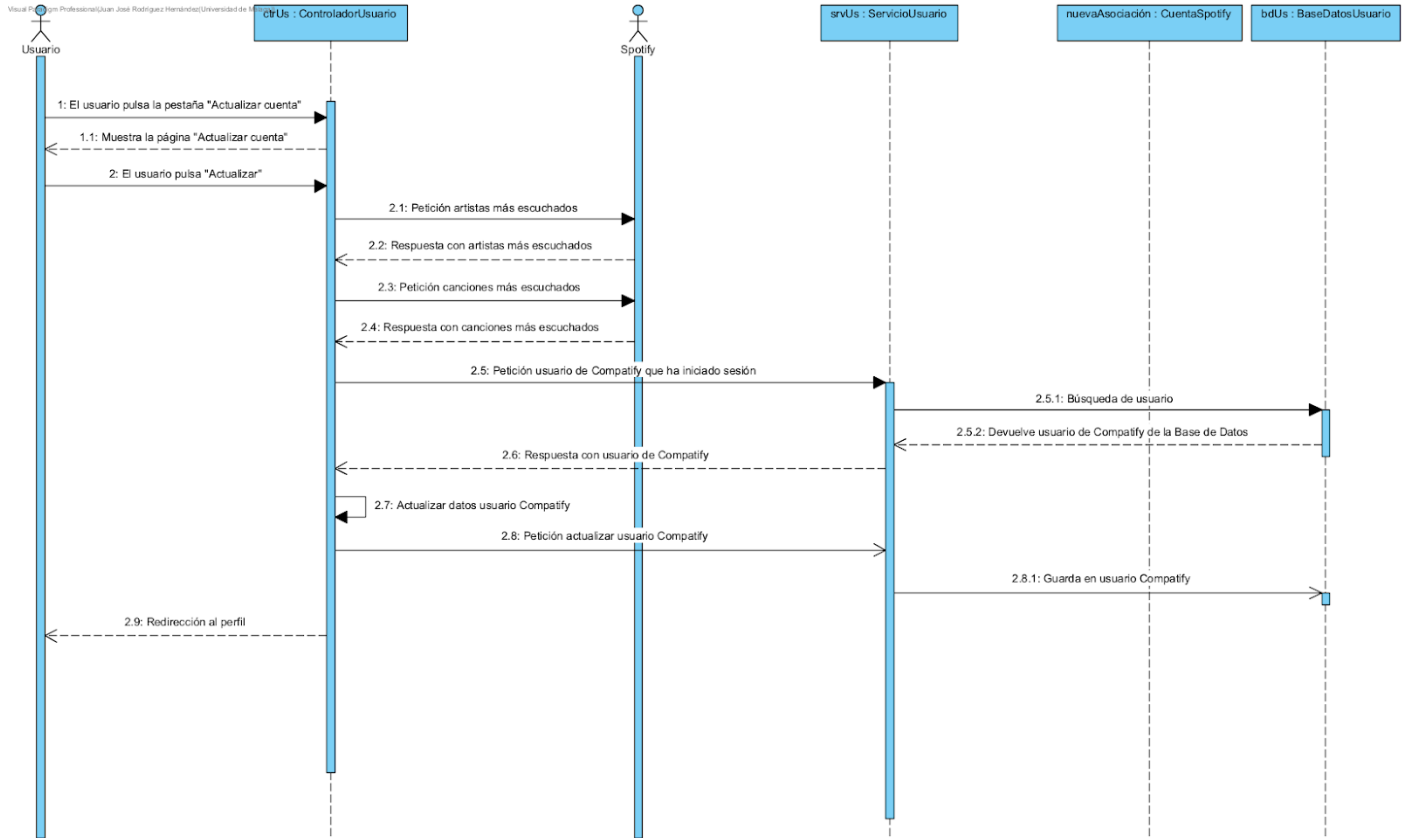
Cambiar contraseña: para este caso se intenta mostrar gráficamente el proceso que conlleva cuando un usuario desea cambiar su contraseña de Compatify. Además de introducir la nueva contraseña, se deberá introducir la anterior para poder cambiarla en nuestra base de datos de manera rápida. Nuevamente se tienen en cuenta tanto los casos en los que se cambia correctamente como en los que existe un error durante el proceso.



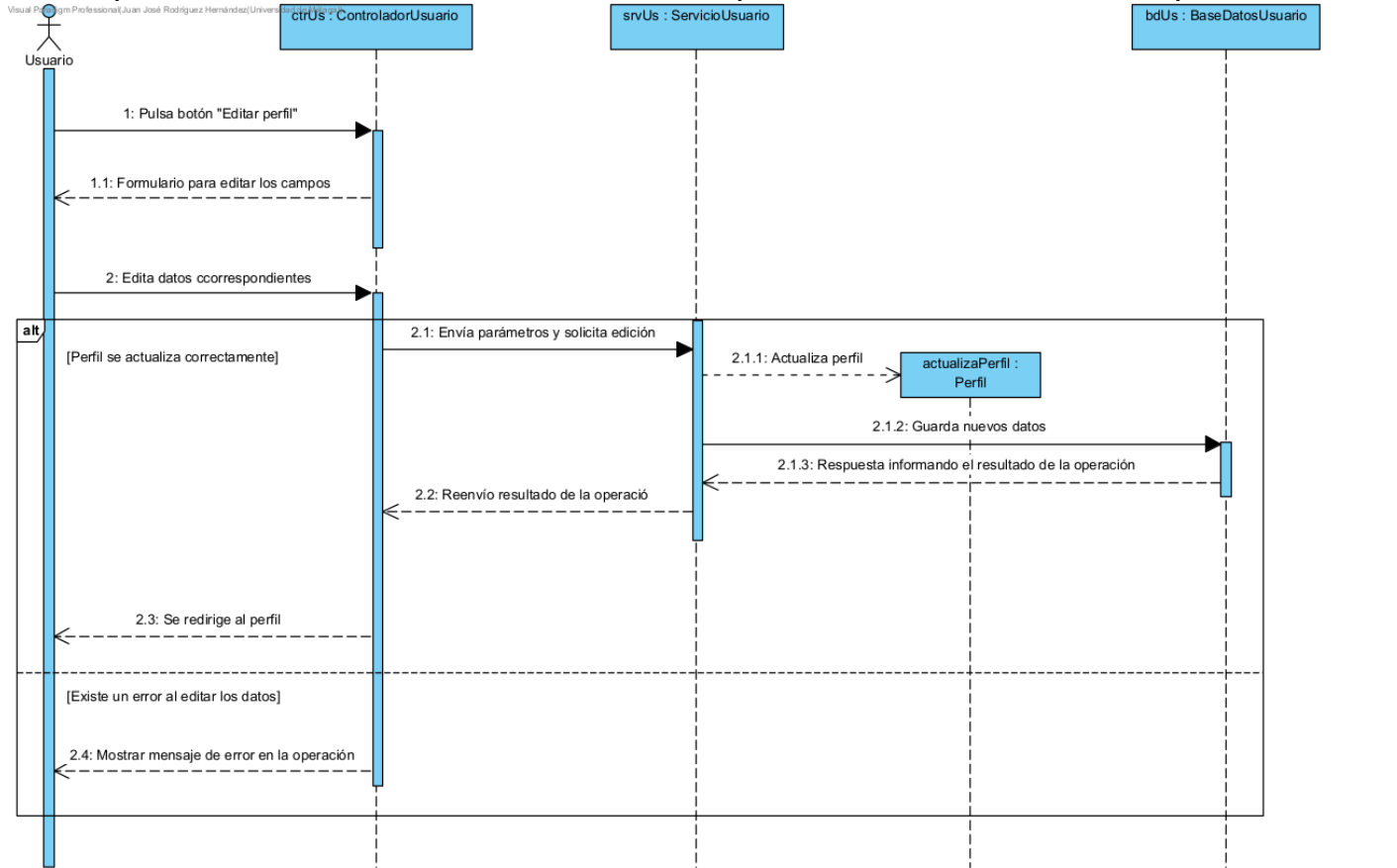
Asociar cuenta: para este caso se intenta mostrar gráficamente el proceso que conlleva cuando un usuario desea enlazar su cuenta de Spotify. Una vez se le redirige a la página de Spotify, deberá rellenar sus datos para iniciar sesión, y se descargará y guardará su historial. Se tienen en cuenta tanto los casos en los que se cambia correctamente como en los que existe un error durante el proceso.



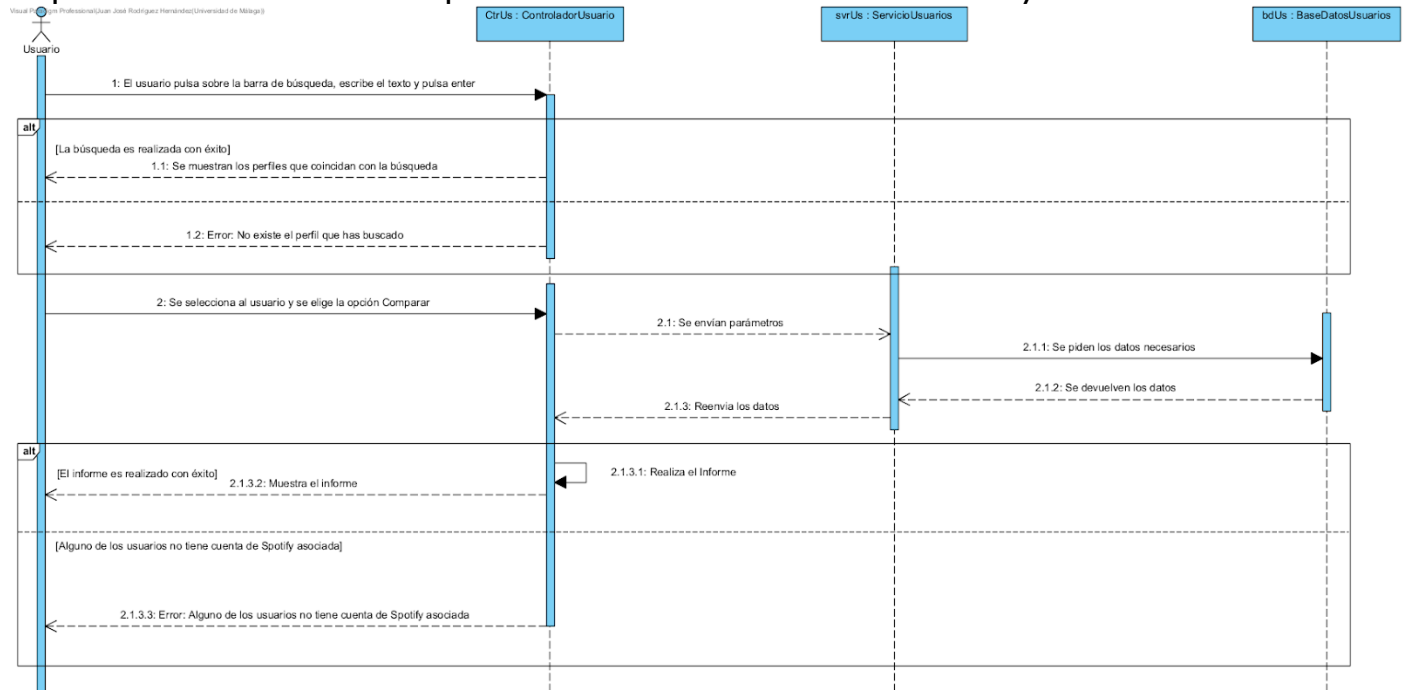
Actualizar Información de Spotify: En el siguiente diagrama, se muestra gráficamente el proceso de cuando un usuario quiere actualizar sus datos de Spotify y así poder disponer de una información más actualizada.



Actualizar perfil: para este caso se intenta mostrar gráficamente el proceso que conlleva cuando un usuario desea cambiar su perfil de Compatify. Deberá editar los campos que desee actualizar y, posteriormente, guardar la operación. Se tienen en cuenta tanto los casos en los que se cambia correctamente como en los que existe un error durante el proceso.



Comparar con otro usuario de Compatify: En este diagrama, mostramos como un usuario puede comparar sus gustos musicales con otro. Para ello, lo primero será buscar el perfil del otro usuario y una vez que se encuentre, solicitar la comparación en nuestro sistema. Cuando es posible realizar dicha comparación se mostrarán todos los datos y detalles de esta.





En esta sección de nuestro proyecto se tratará de redactar una breve descripción de las pruebas que vamos a implementar. Así, se podrá comprobar que nuestra aplicación funciona debidamente. A continuación, describiremos las pruebas en el orden en el que se harán:

1. Crear varios usuarios de Compatify y comprobar que se guardan los datos correctamente en la base de datos:

1.1 Introducir un nombre de usuario que ya esté en la base de datos o una cadena de caracteres vacía, y que se muestre un mensaje de error.

1.2 Introducir un correo electrónico que ya esté en la base de datos o una cadena de caracteres vacía, y que se muestre un mensaje de error.

1.3 Introducir una contraseña que contenga menos de 8 caracteres y que se muestre un mensaje de error.

1.4 Introducir una fecha de nacimiento errónea (fecha que no exista) y que se muestre el mensaje correspondiente.

Se crearán varios usuarios con nombres, contraseñas y otros datos generados aleatoriamente, se introducirán en la base de datos y llamaremos a esta para comprobar si se han guardado los datos introducidos de manera correcta. Si todo funciona bien, limpiamos la base de datos borrando estos usuarios.

2. Iniciar sesión con un usuario ya creado de Compatify

2.1 Introducir un nombre de usuario inexistente y que se muestre un mensaje de error.

2.2 Introducir una contraseña que no se corresponde con el nombre de usuario y que se muestre un mensaje de error.

Se tratará de iniciar sesión con una cuenta no registrada en la base de datos y comprobaremos que la página web no inicie sesión. A continuación, se intentará iniciar sesión con un nombre de usuario, pero introduciendo una contraseña distinta creada aleatoriamente. Se comprobará si se genera el mensaje de error que hemos implementado, y en ese caso que la contraseña es incorrecta y el usuario no puede iniciar sesión, como debía ocurrir.

3. Iniciar sesión con una cuenta de Compatify e intentar editar alguno de los datos del perfil:

3.1 Introducir un nombre de usuario que ya esté en la base de datos o una cadena de caracteres vacía, y que se muestre un mensaje de error.

3.2 Introducir un correo electrónico que ya esté en la base de datos o una cadena de caracteres vacía, y que se muestre un mensaje de error.

3.3 Introducir una fecha de nacimiento errónea (fecha que no exista) y que se muestre el mensaje correspondiente.

Teniendo una cuenta de Compatify ya creada e iniciando sesión en la web, trataremos de cambiar alguno de los datos de nuestro perfil. Se comprobará que se generen los mensajes de error correspondientes si uno de estos ocurre.

- 4. Iniciar sesión con una cuenta de Compatify e intentar cambiar la contraseña:**
 - 4.1 Introducir de manera errónea la contraseña anterior y que el sistema lo detecte.**
 - 4.2 Introducir una contraseña que contenga menos de 8 caracteres y que se muestre un mensaje de error.**
 - 4.3 Introducir una nueva contraseña y que al repetir la nueva contraseña estas no sean idénticas, mostrando así un error.**

Teniendo una cuenta de Compatify ya creada e iniciando sesión en la web, trataremos de cambiar la contraseña del perfil. Se comprobará que se generen los mensajes de error correspondientes si uno de estos ocurre.

- 5. Tratar de asociar una cuenta de Spotify a nuestro usuario de Compatify:**
 - 5.1 Si se solicita el token, que este se reciba correctamente.**
 - 5.2 Comprobar el funcionamiento del token.**
 - 5.3 Al solicitar la asociación Spotify responde con la información de la cuenta.**
 - 5.4 Al solicitar una lista con los artistas más escuchados se nos devuelve una lista del tamaño que hemos solicitado.**
 - 5.5 Al solicitar una lista con las canciones más escuchadas se nos devuelve una lista del tamaño que hemos solicitado.**

Cuando ya tenemos nuestra cuenta de Compatify creada correctamente y queremos asociar una cuenta de Spotify, debemos pedir un token y que se devuelva correctamente. Además, Spotify deberá devolver los datos de la cuenta para asociarlos en la base de datos. Una vez obtenido el token de Spotify comprobaremos que este nos permite acceder a la información de Spotify. Además de poder importar la información que Spotify nos permite exportar desde su API.

Las dos pruebas siguientes consisten en verificar que la lista de artistas más escuchados o canciones más escuchadas que nos muestra por pantalla para el usuario tiene la longitud del tamaño solicitado (en este caso consideramos 50 en ambas).

- 6. Tratar de actualizar la información de la cuenta de Spotify asociada al usuario de Compatify:**
 - 6.1 Si solicitas renovar el token, se recibe el nuevo token.**
 - 6.2 Al solicitar una nueva lista con los artistas más escuchados se nos devuelve una lista del tamaño que hemos solicitado.**
 - 6.3 Al solicitar una nueva lista con las canciones más escuchadas se nos devuelve una lista del tamaño que hemos solicitado.**

En la implementación de nuestro proyecto, cuando queremos renovar el token, llamamos a la base de datos en busca del token de refresco, el cual nos proporcionará el nuevo token. Trataremos de comprobar que a la hora de renovar el token de Spotify el actual sea “machacado” por el nuevo y que se guarde correctamente. Las dos pruebas siguientes consisten en verificar que la lista de artistas más escuchados o canciones más escuchadas que nos muestra por pantalla para el usuario tiene la longitud del tamaño solicitado (en este caso consideramos 50 en ambas).

7. Se intenta buscar un usuario para comparar con él y su nombre de usuario no existe:

Una vez que un usuario se encuentra en la web e intenta comparar sus gustos musicales con otro usuario, tratará de buscar en base a su nombre de usuario. Si no se encuentra ninguno, se mostrará un mensaje de error.

8. No se puede comparar dos usuarios sin cuenta asociada de Spotify:

Se creará una cuenta y sin haber establecido un previo enlace con Spotify se tratará de comparar con otro usuario del dominio. Se comprobará que el usuario no dispone de un historial musical puesto que no hay nada en la base de datos relacionado a su cuenta, y se mostrará un mensaje de error advirtiéndole de que debe iniciar sesión en Spotify.

9. Al solicitar la comparación muestre por pantalla el porcentaje y canciones/artistas correspondiente:

9.1 Comprobar que si al método de comparar usuarios se le pasan dos listas de canciones con varias canciones en común, se devuelven dichas canciones.

9.2 Comprobar que si al método de comparar usuarios se le pasan dos listas de artistas con varios artistas en común, se devuelven dichos artistas.

9.3 Comprobar que el porcentaje sea significativo (esto es, que sea mayor cuantos más artistas/canciones se tengan en común).

9.4 Si dos usuarios no tienen nada en común que se muestre un mensaje especificando la situación.

Se realizarán múltiples comparaciones en las que veremos si aparecen en los informes los porcentajes y los artistas y canciones comunes, con posibilidad de reproducción. Se intentará ver en caso de error el motivo de su ausencia.

Se crea una clase usuario compatify y se llama al método comparar usuarios de esa clase, pasándole como parámetros dos arrays, uno de artistas y otro de canciones, de forma que haya alguna coincidencia con los arrays guardados en dicha clase. Se comprueba que el resultado devuelve los artistas y canciones que coinciden.

Trataremos de comparar con distintos usuarios de la base de datos y ver que si hay más coincidencias en gustos musicales con uno que con otro, el porcentaje de compatibilidad sea mayor. En caso de que dos usuarios no tengan nada en común, la compatibilidad será de 0% y se mostrará en pantalla lo ocurrido.



HERRAMIENTAS SOFTWARE

- a. **Google docs/Google drive:** Usaremos este software para la elaboración de forma colaborativa de los documentos relacionados con el proyecto.
- b. **Paquete Office:** Lo usaremos para redacción y apuntes del proyecto.
- c. **Whatsapp:** Software empleado para la comunicación no instantánea para la organización y planteamientos de diferentes ideas sobre el proyecto.
- d. **Discord:** software usado para la comunicación instantánea a distancia. Es decir, para mantener una conversación mientras hacemos el proyecto.
- e. **GitHub:** Repositorio remoto para trabajar de forma colaborativa sobre el código de la aplicación.
- f. **Trello:** Software para la planificación y organización de las diferentes tareas y subtareas del proyecto.
- g. **Visual Paradigm:** Software para la creación de los requisitos, diagramas de casos, y establecer relaciones entre ellos.
- h. **Visual Studio Code/Eclipse:** Programas utilizados para el desarrollo del código.