

UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

Colegio de Ciencias e Ingenierías

Music Match: Connect Through Music

Carol Mariasol Pineda Sotomayor

Ingeniería en Ciencias de la Computación

Trabajo de fin de carrera presentado como requisito
para la obtención del título de
Ingeniero en Ciencias de la Computación

Quito, 11 de diciembre de 2020

UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

Colegio de Ciencias e Ingenierías

**HOJA DE CALIFICACIÓN
DE TRABAJO DE FIN DE CARRERA**

Music Match: Connect Through Music

Carol Mariasol Pineda Sotomayor

Nombre del profesor, Título académico

Daniel Riofrío, Ph.D.

Quito, 11 de diciembre de 2020

© DERECHOS DE AUTOR

Por medio del presente documento certifico que he leído todas las Políticas y Manuales de la Universidad San Francisco de Quito USFQ, incluyendo la Política de Propiedad Intelectual USFQ, y estoy de acuerdo con su contenido, por lo que los derechos de propiedad intelectual del presente trabajo quedan sujetos a lo dispuesto en esas Políticas.

Asimismo, autorizo a la USFQ para que realice la digitalización y publicación de este trabajo en el repositorio virtual, de conformidad a lo dispuesto en la Ley Orgánica de Educación Superior del Ecuador.

Nombres y apellidos: Carol Mariasol Pineda Sotomayor

Código: 00132674

Cédula de identidad: 1104441918

Lugar y fecha: Quito, 11 de diciembre de 2020

ACLARACIÓN PARA PUBLICACIÓN

Nota: El presente trabajo, en su totalidad o cualquiera de sus partes, no debe ser considerado como una publicación, incluso a pesar de estar disponible sin restricciones a través de un repositorio institucional. Esta declaración se alinea con las prácticas y recomendaciones presentadas por el Committee on Publication Ethics COPE descritas por Barbour et al. (2017) Discussion document on best practice for issues around theses publishing, disponible en <http://bit.ly/COPETheses>.

UNPUBLISHED DOCUMENT

Note: The following capstone project is available through Universidad San Francisco de Quito USFQ institutional repository. Nonetheless, this project – in whole or in part – should not be considered a publication. This statement follows the recommendations presented by the Committee on Publication Ethics COPE described by Barbour et al. (2017) Discussion document on best practice for issues around theses publishing available on <http://bit.ly/COPETheses>.

RESUMEN

El presente proyecto busca ofrecer una solución para realizar conexiones entre músicos, a partir de sus gustos musicales, su instrumento de ejecución, su experiencia con el mismo y su edad. Para este propósito se utilizó Python3 integrado con Django para ofrecer una aplicación web donde los usuarios modifican los parámetros de los integrantes que desean para conformar su ensamble. El sistema usa un algoritmo, que, basado en las distancias de gustos musicales, experiencia y edad devuelve al usuario los músicos más cercanos a los planteados por el usuario. A través de este proyecto se aplicaron e integraron varios conceptos fundamentales dentro de la ingeniería en ciencias de la computación, de una manera satisfactoria. Se produjo una aplicación con dos funcionalidades principales; en primer lugar, como un registro de músicos y sus datos presentados a modo de una agenda, y por otra parte el usuario puede usar la aplicación para generar grupos o ensambles de mínimo 2 y máximo 10 personas. Sin embargo, se plantean varias mejoras a futuro, como la integración de tablas dinámicas para la selección de usuarios, además de la integración de un API con un servicio de *streaming* de música como Spotify para segmentar de una manera real los gustos musicales del usuario.

Palabras clave: Python3, Django, desarrollo web, Bootstrap, ensamble, músicos.

ABSTRACT

This Project aims to offer a solution for musicians, by building connections between users based upon music genre preferences, instrument, experience, and age. For this matter, a web application was developed using Python 3 and Django which lets users modify the parameters previously described for each member of the ensemble. The system uses an algorithm that calculates the distance between musical genres, experience, and age, sums it up and returns the closest musician according to the given characteristics. Throughout this project, multiple concepts were used, implemented, and integrated successfully. The result is a web application with two main functionalities; first, this works as a musician's registry for the users, displayed as a yearbook. On the other hand, the user can create ensembles from 2 up to 10 members. However, there are several upgrades proposed for a future work. For example, the usage of dynamic tables for ensemble member selection, and the integration of accounts of streaming music services such as Spotify through an API, to track the real musical preferences of each user.

Key words: Python3, Django, web development, Bootstrap, ensemble, musicians.

TABLA DE CONTENIDO

Introducción	11
Objetivo General.....	12
Objetivos Específicos.....	12
Desarrollo del Tema.....	13
Análisis de Requerimientos	13
Diseño de la aplicación	26
Implementación.....	31
Resultados	44
Problemas Conocidos.....	45
Conclusiones	46
Conclusiones	46
Trabajo futuro	46
Referencias bibliográficas	48

ÍNDICE DE TABLAS

Tabla 1. Sistema de calificación para aristas del grafo de géneros musicales.....	28
Tabla 2. Herramientas de desarrollo y sus versiones.....	32

ÍNDICE DE FIGURAS

Figura 1. Casos de uso para usuario no autenticado	14
Figura 2. Casos de uso para usuario autenticado de Music Match	14
Figura 3. Pantalla de Inicio Music Match.....	15
Figura 4. Pantalla de manual Music Match a) Parte 1, b) Parte 2.....	16
Figura 5. Formulario para registro de perfil.....	17
Figura 6. Registro exitoso de usuario.	18
Figura 7. Mensaje de sistema en caso de encontrar errores en el formulario de registro.	18
Figura 8. Formulario para ingreso de un usuario ya registrado.	19
Figura 9. Página de inicio para usuario registrado.....	19
Figura 10. Página de login luego de que los datos ingresados no coincidieran.....	20
Figura 11. Formulario para crear un ensamble.	21
Figura 12. Página de ensamble.	22
Figura 13. Mensaje emitido por el sistema en caso de no seleccionar ningún músico.....	23
Figura 14. Resultado de búsqueda de perfiles, con la cadena de caracteres “Sol”.....	24
Figura 15. Pantalla de perfil detallado.	24
Figura 16. Resultado de búsqueda de perfiles, con el año de nacimiento “1852”.....	25
Figura 17. Pantalla All Ensembles.....	25
Figura 18. Pantalla My Ensembles.	26
Figura 19. Presentación de super-géneros de MusicMap	27
Figura 20. Grafo de representación de géneros musicales.....	29
Figura 21. Matriz de distancias de géneros musicales.....	30
Figura 22. Diagrama de funcionamiento del modelo MVC.	31

Figura 23. Diagrama de base de datos.	32
Figura 24. Seudocódigo de algoritmo de creación de ensambles.	35
Figura 25. Prueba 1 con estilo y nivel predeterminado.	36
Figura 26. Ensamble de resultado de la prueba 1.	37
Figura 27. Prueba 1 con nivel predeterminado.	38
Figura 28. Resultado de ensamble de la prueba 1 con nivel predeterminado.	39
Figura 29. Prueba 1 con nivel y estilo aleatorio.	39
Figura 30. Resultado de ensamble de la prueba 1 con nivel y estilo aleatorio.	40
Figura 31. Prueba 2 con estilo y nivel predeterminado.	41
Figura 32. Resultado de ensamble de prueba 2 con estilo y nivel predeterminado.	41
Figura 33. Prueba 2 con nivel predeterminado.	42
Figura 34. Resultado de ensamble de la prueba 2 con nivel predeterminado.	43
Figura 35. Prueba 2 con nivel y estilo aleatorio.	43
Figura 36. Resultados de ensamble de la prueba 2 con nivel y estilo aleatorio.	44

INTRODUCCIÓN

Las conexiones interpersonales son una necesidad básica para los seres humanos, al introducir el factor de ejecución musical como posible potenciador de dichas conexiones, se obtienen emparejamientos mucho más estimulantes para los actores involucrados, tanto en el ámbito creativo como en su bienestar personal. El presente proyecto sugiere posibles grupos de personas basado en compatibilidad de instrumentos, gustos musicales y experticia.

En primer lugar, es necesario mencionar la importancia de promover la música y su práctica dentro de la sociedad. Un estudio realizado sobre la “efectividad de la Experiencia en un ensamble de Música de Cámara” sugiere que, aunque varios de los participantes no tuvieron una mejora significativa en sus habilidades psicomotrices para ejecutar el instrumento que escogieron, al final del estudio mostraron una actitud mucho más favorable hacia la música y la participación en la misma (Zorn, 1973). En este caso, la sugerencia de un ensamble facilitaría el espacio propicio para la inmersión musical.

En segundo lugar, la ejecución de música en grupo supone un beneficio para los músicos como individuos. Al tratar de sincronizar musicalmente con otros, incluso si esa persona no es visible o no se encuentra en el mismo lugar físico, se presenta una actitud positiva hacia el otro con el que se coordinan los movimientos (Launay, 2014). Dicha coordinación está conectada con la liberación de químicos cerebrales como la dopamina (Tarr, 2014). Además, un estudio se realizó en humanos probando que, cantar por 30 minutos aumenta significativamente los niveles de oxitocina, sin importar cuán felices o tristes los hizo esta experiencia (Grape, 2003).

En tercer lugar, las preferencias musicales constituyen una herramienta para realizar conexiones interpersonales. Individuos que descubren a personas con los mismos gustos

musicales que sí mismos, tienden a pensar mejor de ellos, asocian estos gustos con valores específicos e influencia su percepción de manera positiva hacia entablar una amistad con estas (Boer, 2011). Este documento detalla el proceso de desarrollo de una herramienta digital que provee un medio facilitador para realizar posibles conexiones con personas que compartan nuestros gustos musicales. Con quienes, además, se puede desarrollar una relación profesional y de colaboración creativa.

Objetivo General

Crear y presentar una herramienta que sugiera a sus usuarios posibles grupos para ejecutar música, basándose en parámetros de; géneros musicales preferidos, instrumento practicado, nivel de experiencia y edad.

Objetivos Específicos

- Diseñar e implementar una base de datos que contenga la información de los músicos o usuarios.
- Implementar un grafo para caracterizar a los diferentes géneros musicales.
- Programar una heurística para sugerir posibles grupos de usuarios.
- Diseñar e implementar una página web que muestre las conexiones de cada usuario.
- Comparar las diferentes predicciones del modelo y su utilidad.

DESARROLLO DEL TEMA

Análisis de Requerimientos

Es necesario plantear los requerimientos de la aplicación en función al problema que esta intenta resolver. Tomando en cuenta que la misma busca conectar a músicos y ofrecerles opciones de ensambles. Por lo tanto, el prototipo contiene las siguientes funcionalidades:

- Registro y mantenimiento de usuarios por medio de una base de datos.
- Customización de datos del perfil de cada usuario, es decir, edad, instrumento, género musical preferido, además de su nivel de experiencia.
- Algoritmo de generación de ensambles de músico en base a instrumentos, género musical, nivel de experiencia y edad.

Tomando en cuenta los requerimientos expuestos anteriormente se utilizó Python3 como lenguaje de programación, debido a la gran cantidad de bibliografía disponible para el desarrollo de los distintos componentes de la aplicación.

Casos de Uso

Los casos de uso que se expondrán a continuación describen el comportamiento esperado del usuario frente a la aplicación, también se describe el curso alternativo de eventos en caso de no realizar los pasos inicialmente programados. Para este propósito el usuario será llamado Músico. A continuación, en la figura 1, se detallan todos los casos de uso para el músico autorizado y, en la figura 2, los casos de uso para un músico sin autenticar al acceder a Music Match.

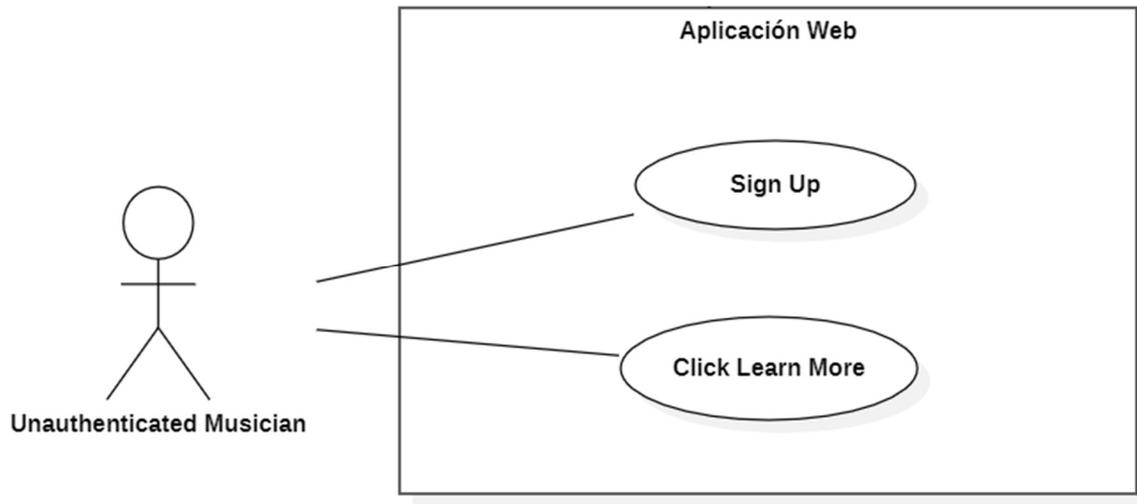


Figura 1. Casos de uso para usuario no autenticado

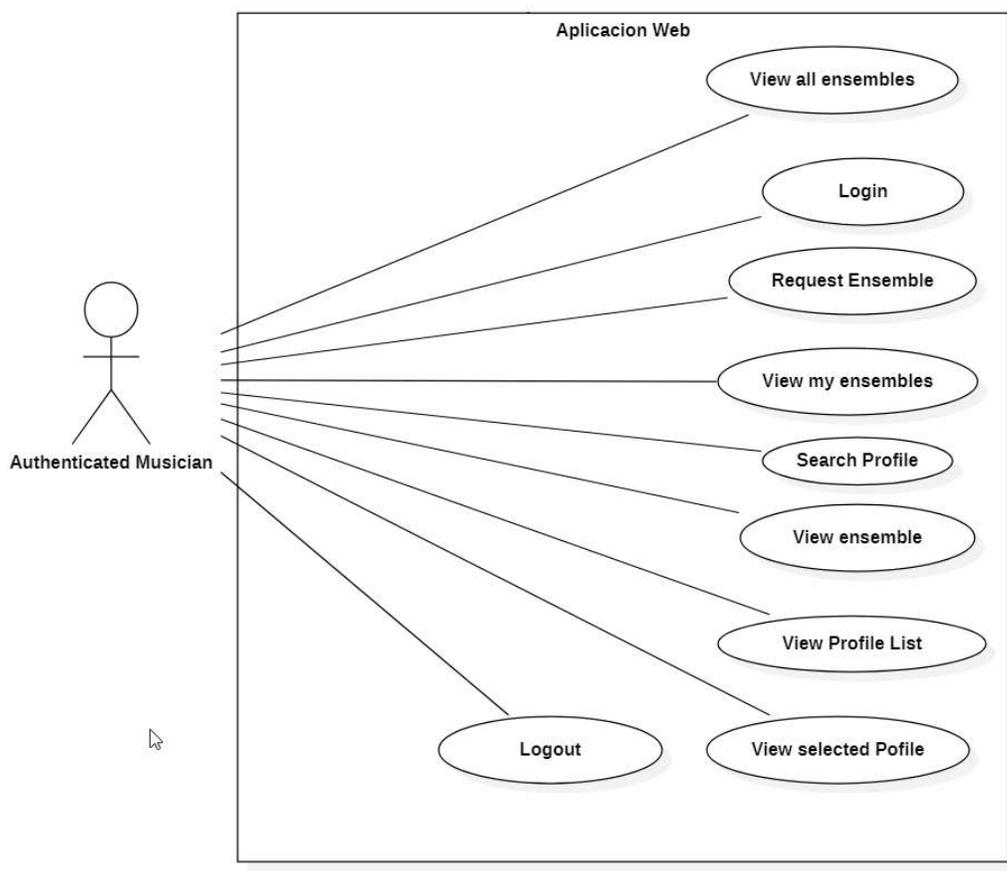


Figura 2. Casos de uso para usuario autenticado de Music Match

Caso de Uso: Clic Learn More

Curso Normal: El músico ingresa a la aplicación y se muestra una pantalla de bienvenida ilustrada en la figura 3. El músico presiona el botón “Learn More”, el sistema lo redirige a la pantalla de manual donde se muestra el grafo y se explica el funcionamiento de la aplicación, como muestra la figura 4a. Como indica la figura 4b al final de esta pantalla se muestra un botón de “Sign Up”, el cual, al ser presionado por el usuario lo redirige a la página de creación de perfil y sigue el curso descrito en el caso de uso “Sign Up”.

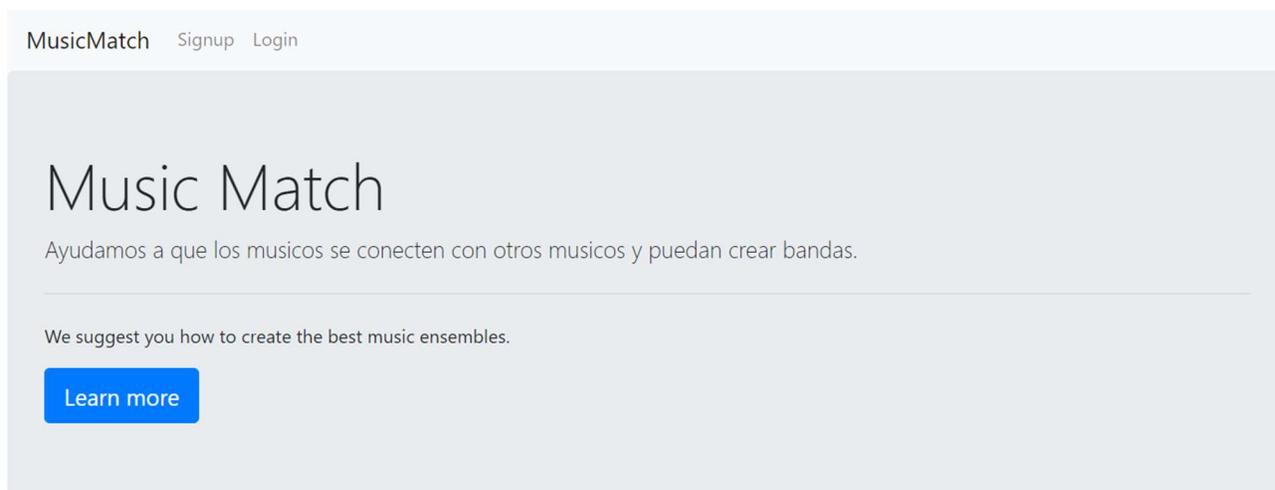


Figura 3. Pantalla de Inicio Music Match.

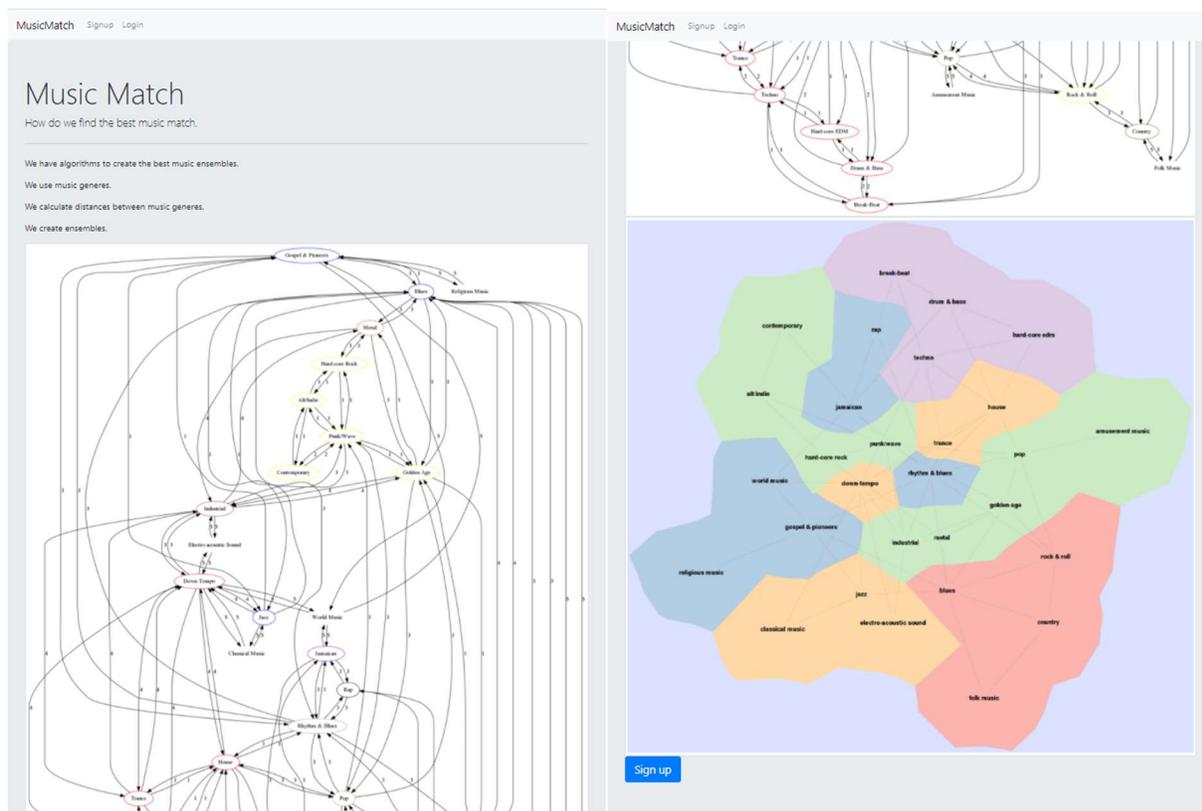


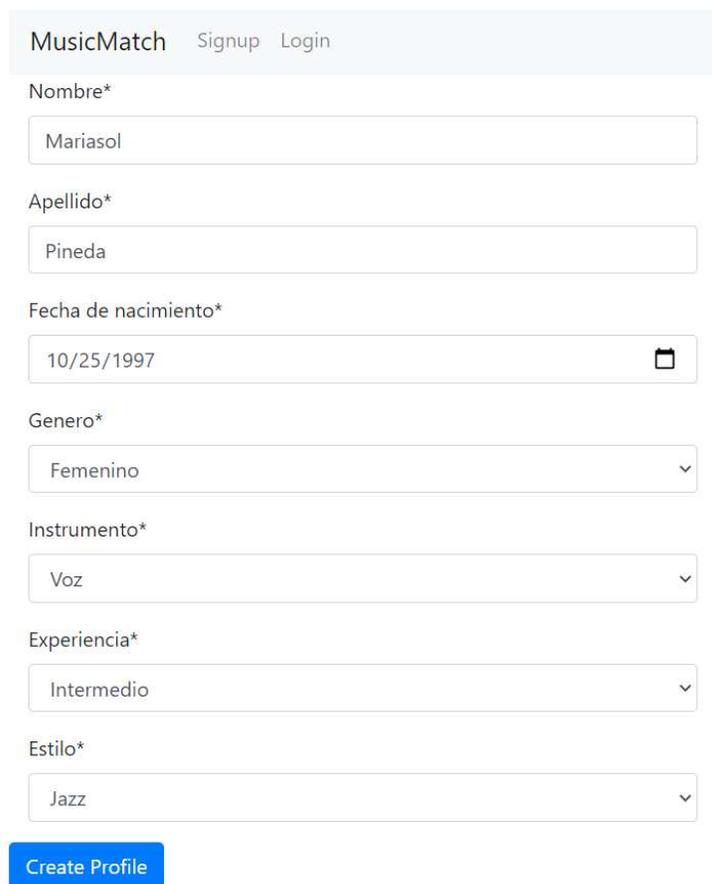
Figura 4. Pantalla de manual Music Match a) Parte 1, b) Parte 2.

Caso de Uso: Sign Up

Curso Normal: en la página de inicio, el músico presiona el botón “Sign Up”, el sistema lo redirecciona hacia la página para crear su usuario. Se muestra un formulario en la que el músico llena sus datos para crear el perfil, como se muestra en la figura 5. Se comprueba que los datos introducidos en el formulario se encuentren en el formato esperado. El sistema crea un perfil y guarda los datos asociados. El sistema redirecciona al músico a la página de inicio y muestra otros botones dentro de la barra de navegación. Además, aparece un pop up con el nombre de usuario y claves generados para dicho usuario, conformados por el nombre y apellido del usuario y su edad, como se muestra en la figura 6.

Curso Alterno:

El nombre, apellido o edad del músico se encuentra vacío: Como muestra la figura 7, el sistema emite un mensaje informando al músico que dichos campos no pueden estar vacíos. Refresca la misma página.



The image shows a web form for creating a profile on MusicMatch. At the top, there is a navigation bar with the text "MusicMatch" and links for "Signup" and "Login". Below this, the form consists of several fields, each with an asterisk indicating it is required:

- Nombre***: A text input field containing the name "Mariasol".
- Apellido***: A text input field containing the surname "Pineda".
- Fecha de nacimiento***: A date picker field showing "10/25/1997" with a calendar icon to its right.
- Genero***: A dropdown menu with "Femenino" selected and a downward arrow.
- Instrumento***: A dropdown menu with "Voz" selected and a downward arrow.
- Experiencia***: A dropdown menu with "Intermedio" selected and a downward arrow.
- Estilo***: A dropdown menu with "Jazz" selected and a downward arrow.

At the bottom of the form is a blue button labeled "Create Profile".

Figura 5. Formulario para registro de perfil.

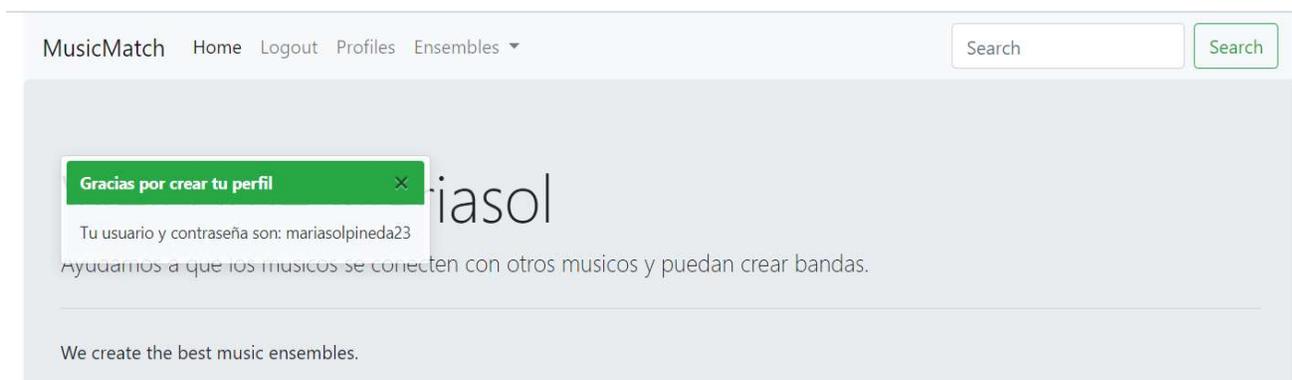


Figura 6. Registro exitoso de usuario.

 The screenshot shows the MusicMatch registration form. The navigation bar includes "MusicMatch", "Signup", and "Login". The form has three fields: "Nombre*" (Name) with the value "Mariasol", "Apellido*" (Last name) which is empty, and "Fecha de nacimiento*" (Date of birth) with a placeholder "mm/dd/yyyy" and a calendar icon. A red error message box points to the "Apellido*" field, stating "Please fill out this field."

Figura 7. Mensaje de sistema en caso de encontrar errores en el formulario de registro.

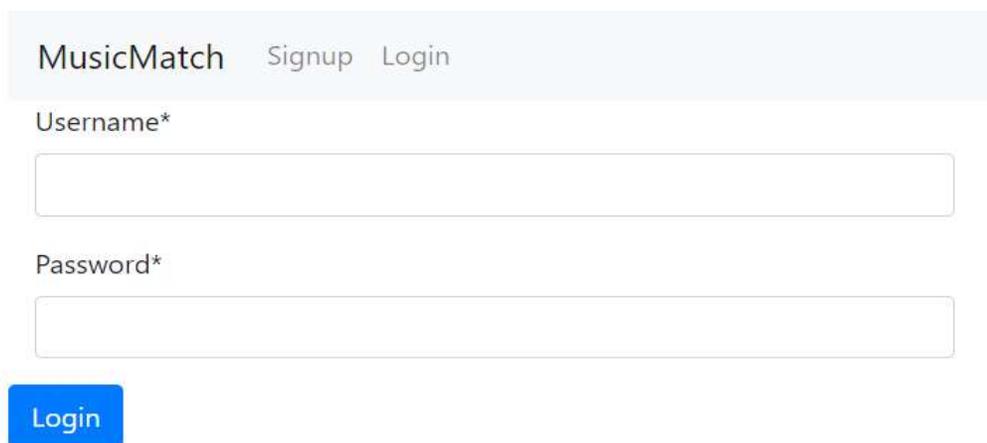
Caso de Uso: Login

Curso Normal: en la página de inicio, el músico presiona el botón "Login", el sistema lo redirecciona hacia la página de login donde se muestra el formulario para rellenar el nombre de usuario y la contraseña, ilustrado en la figura 8. El sistema verifica la concordancia de los datos ingresados y redirecciona al usuario a la página de inicio, con el mensaje "Welcome Back Username" como muestra la figura 9.

Cursos Alternos:

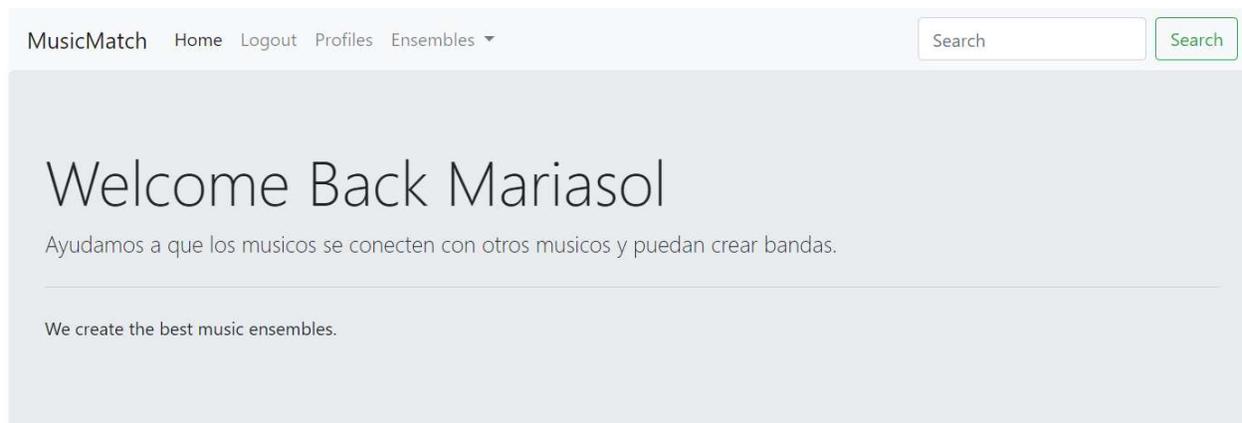
El nombre de usuario o la contraseña del músico se encuentra vacío: El sistema emite un mensaje informando al músico que dichos campos no pueden estar vacíos.

El nombre de usuario no coincide con la contraseña ingresada: Como muestra la figura 10 el sistema refresca la página e informa al músico que el nombre de usuario y la contraseña no coinciden.



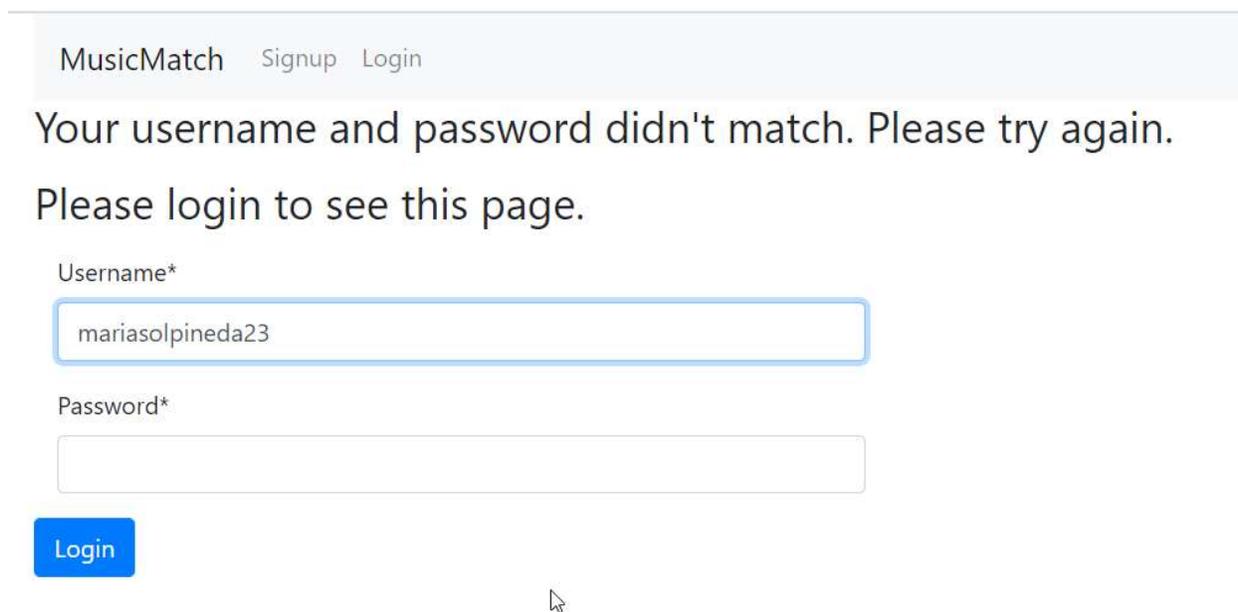
The screenshot shows a login form for MusicMatch. At the top left, the text "MusicMatch" is followed by "Signup" and "Login" links. Below this, there are two input fields: "Username*" and "Password*", each with a corresponding empty text box. A blue "Login" button is positioned below the password field.

Figura 8. Formulario para ingreso de un usuario ya registrado.



The screenshot shows the home page of MusicMatch for a registered user. The top navigation bar includes "MusicMatch", "Home", "Logout", "Profiles", and "Ensembles" with a dropdown arrow. On the right side of the navigation bar, there is a search input field with a "Search" button. The main content area features a large heading "Welcome Back Mariasol" and a subheading "Ayudamos a que los músicos se conecten con otros músicos y puedan crear bandas." Below this, there is a horizontal line and the text "We create the best music ensembles."

Figura 9. Página de inicio para usuario registrado.



MusicMatch Signup Login

Your username and password didn't match. Please try again.
Please login to see this page.

Username*

Password*

Login

Figura 10. Página de login luego de que los datos ingresados no coincidieran.

Caso de Uso: Request Ensemble

Curso normal: el músico ha ingresado al sistema y presiona el botón “Request Ensemble”. El sistema le presenta el formulario para que lo rellene con sus preferencias para cada uno de los músicos. Como se muestra en la figura 11, este puede seleccionar el tipo de instrumento, género musical y nivel de experiencia para cada uno de los integrantes del ensamble. El sistema admite una selección mínima de 1 persona y máximo 9 personas. Luego de haber seleccionado los campos deseados, el sistema escoge los músicos que se aproximan más a los requerimientos propuestos por el creador del ensamble y se muestra la pantalla del ensamble creado como el ejemplo de la figura 12.

Curso alterno:

No se ha seleccionado ningún músico: Como se ilustra en la figura 13, en caso de no seleccionar ningún músico el sistema muestra un pop up indicando al usuario que el espacio del primer músico debe estar lleno.

MusicMatch Home Logout Profiles Ensembles ▾ Search Search

Nombre*

Prueba

Instrumento 2	Guitarra ▾	Pop ▾	Intermedio ▾	<input checked="" type="checkbox"/> *
Instrumento 3	Bajo ▾	Alt/Indie ▾	Avanzado ▾	<input checked="" type="checkbox"/>
Instrumento 4	Voz ▾	Country ▾	Avanzado ▾	<input checked="" type="checkbox"/>
Instrumento 5	Contrabajo ▾	Blues ▾	Experto ▾	<input checked="" type="checkbox"/>
Instrumento 6	Bateria ▾	Jazz ▾	Avanzado ▾	<input checked="" type="checkbox"/>
Instrumento 7	Piano ▾	Classical Music ▾	Experto ▾	<input checked="" type="checkbox"/>
Instrumento 8	Flauta ▾	Jazz ▾	Avanzado ▾	<input checked="" type="checkbox"/>
Instrumento 9	Violin ▾	Classical Music ▾	Experto ▾	<input checked="" type="checkbox"/>

Request Ensemble

Figura 11. Formulario para crear un ensemble.

MusicMatch Home Logout Profiles Ensembles ▾

Search Search

Prueba **171**

Mariasol Pineda2020-12-18

 <p>Adrián Fernández Contrabajo Rhythm & Blues Avanzado 18</p>	 <p>Ana Paz Guitarra Pop Intermedio 20</p>	 <p>Andrés Cao Piano Classical Music Principiante 53</p>
 <p>David Jiménez Bajo Pop Intermedio 26</p>	 <p>José Armendariz Voz Blues Experto 31</p>	 <p>Mariasol Pineda Voz Jazz Intermedio 23</p>
 <p>María López Flauta Jazz Intermedio 35</p>	 <p>Salvador Alvarado Violín Electro-acoustic Sound Experto 78</p>	 <p>Sara Capilla Batería Blues Avanzado 64</p>

Figura 12. Página de ensamble.

MusicMatch Home Logout Profiles Ensembles ▾

Search Search

Nombre*

Prueba

Instrumento 2	Guitarra ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 3	Bajo ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 4	Voz ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 5	Contrabajo ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 6	Bateria ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 7	Piano ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 8	Flauta ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 9	Violin ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>

Request Ensemble

! Please check this box if you want to proceed.

Figura 13. Mensaje emitido por el sistema en caso de no seleccionar ningún músico.

Caso de Uso: Search Profile

Curso normal: El músico ha ingresado al sistema, en la barra de navegación se muestra un widget para buscar perfiles. El músico puede introducir cualquier término de búsqueda, ya sea nombre, edad, año de nacimiento, instrumento y género musical. El sistema realiza la búsqueda en la base de datos y presenta una lista de perfiles que coincidan con la búsqueda como muestra la figura 14. A su vez, se puede seleccionar uno de estos perfiles, y aparecerá la pantalla que muestra un perfil específico como lo ilustra la figura 15.

Caso alterno:

Ningún perfil coincide con la búsqueda realizada: Como se muestra en la figura 16 el sistema muestra una página en blanco únicamente con la barra de navegación.

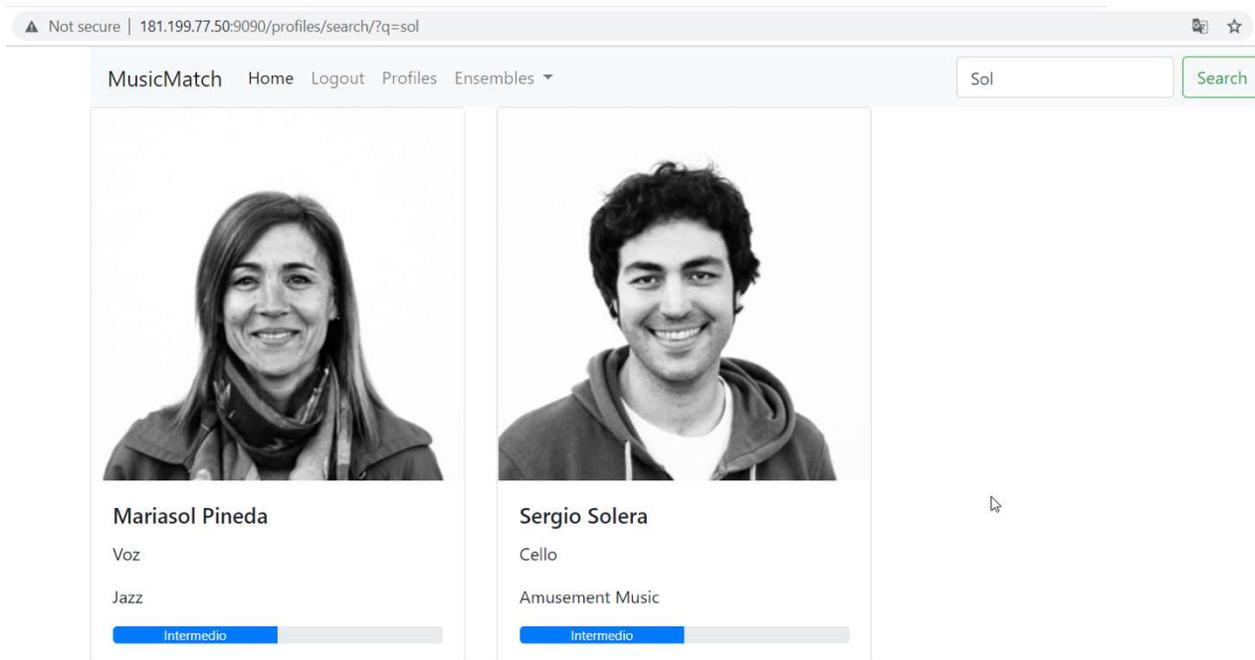


Figura 14. Resultado de búsqueda de perfiles, con la cadena de caracteres “Sol”.



Figura 15. Pantalla de perfil detallado.

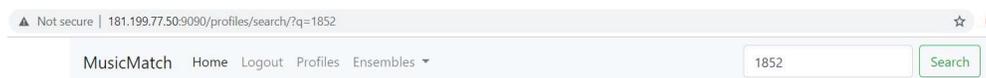


Figura 16. Resultado de búsqueda de perfiles, con el año de nacimiento “1852”.

Caso de Uso: View All Ensembles

Curso normal: El usuario ha ingresado al sistema y oprime el botón “All Ensembles”. A continuación, el sistema muestra los ensambles creados por todos los usuarios. Se muestran 6 ensambles por página, donde se incluye el nombre del ensamble, puntuación del ensamble, el músico que lo creó, la fecha de creación y los instrumentos que lo conforman seguido del nivel de experiencia de cada uno, como se muestra en la figura 17. El músico puede seleccionar cualquiera de los ensambles para ver la página de un ensamble en específico donde se incluye un hipervínculo hacia el perfil de cada uno de los integrantes, en el formato mencionado en el caso de uso “Request Ensemble”.

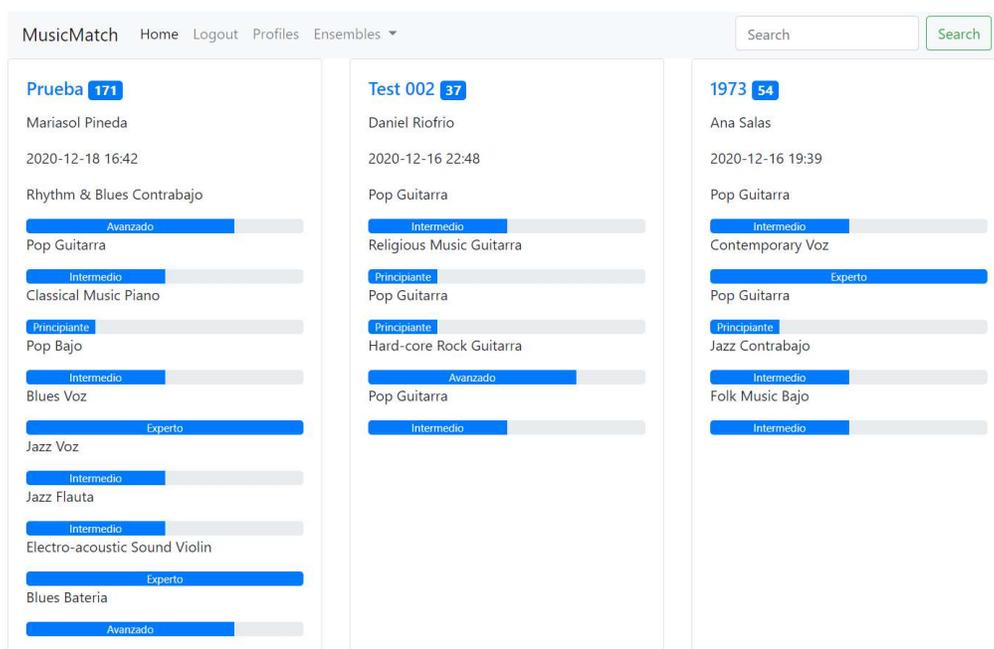


Figura 17. Pantalla All Ensembles.

Caso de Uso: View My Ensembles

Curso normal: El usuario ha ingresado al sistema y oprime el botón “*My Ensembles*”. Como se muestra a continuación en la figura 18 el sistema carga la pantalla que muestra únicamente los ensambles que fueron creados por el músico que se encuentra logueado, en el mismo formato mencionado en el caso de uso “*View All Ensembles*”

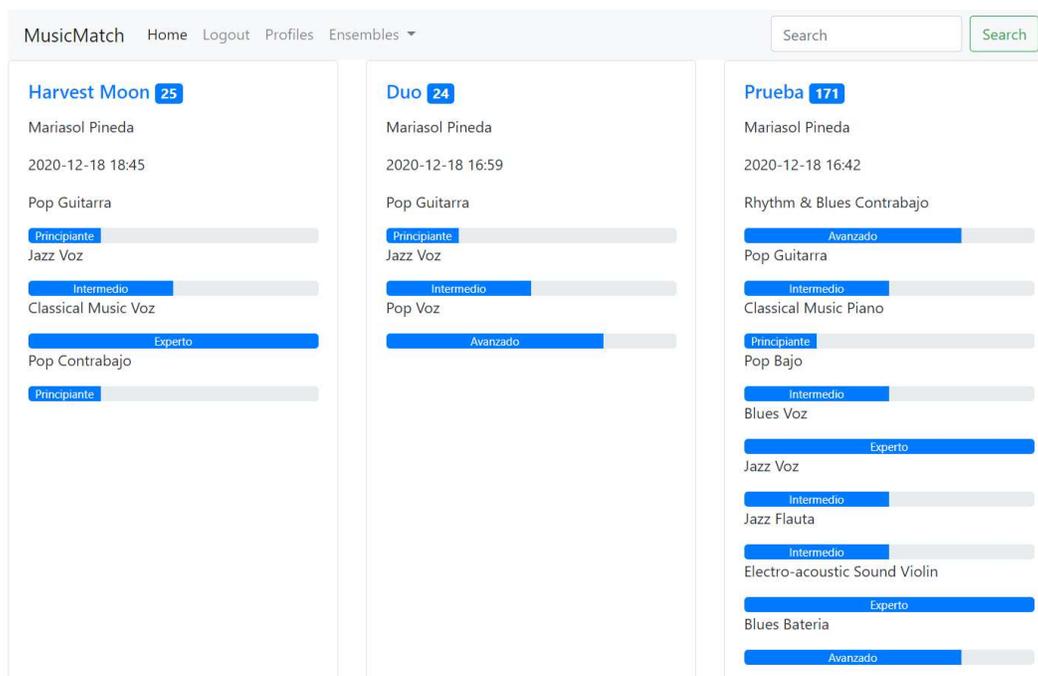


Figura 18. Pantalla My Ensembles.

Diseño de la aplicación

Para categorizar a los géneros musicales se ha utilizado el *framework* presentado por Music Map, donde se han recopilado alrededor de 230 géneros musicales. Sin embargo, para simplificar su modelo, proponen la creación de “*Súper-géneros*”, es decir, áreas conocidas y delimitadas dentro de la red de géneros musicales y de la cual se derivan varios otros géneros. Se delimitan 3 súper géneros; Rock, Dance/EDM y Blue Note, además se caracteriza las influencias más

importantes con líneas continuas y las menos importantes con líneas entrecortadas (Musicmap, s/f).

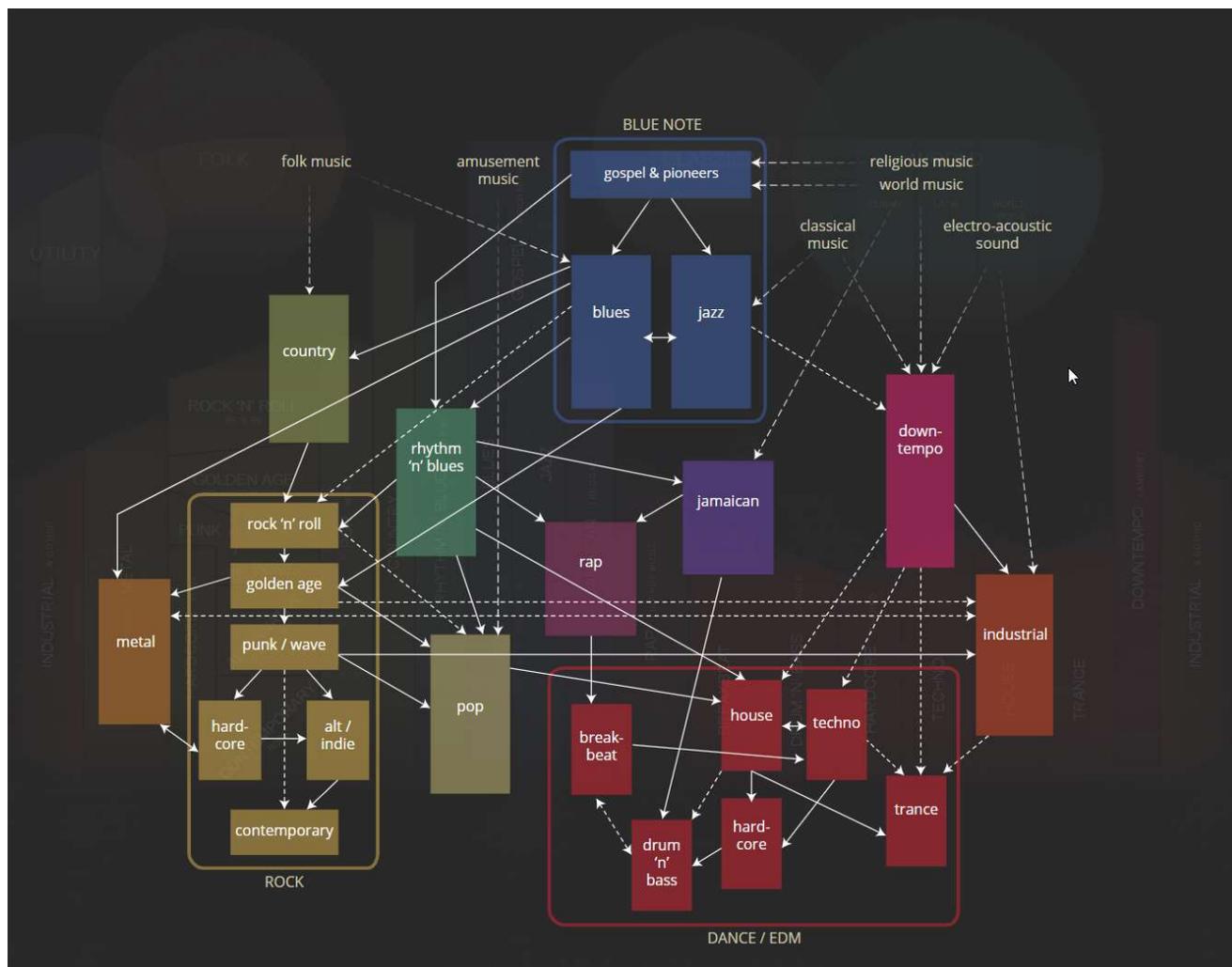


Figura 19. Presentación de super-géneros de MusicMap

Para traducir esta aproximación se colocaron diferentes pesos a cada una de las aristas dependiendo su figuración (entrecortada o continua), y la pertenencia del género o nodo a un súper grupo, además de sus relaciones entre sí. A continuación, en la tabla 1 se detalla los pesos asignados a cada arista.

Tipo de arista	Peso asignado
Línea continua dentro del mismo súper-género	1
Línea entrecortada dentro del mismo súper-género	2
Línea continua con no súper-género	3
Línea continua entre súper-géneros	4
Línea entrecortada entre súper-géneros	5

Tabla 1. Sistema de calificación para aristas del grafo de géneros musicales.

Representación de grafo de géneros musicales

Una vez construido el grafo descrito en la figura 20, se encontró la distancia entre todos los géneros musicales. Para hacerlo primero se repitió cada una de las aristas en sentido contrario con el mismo peso y luego se encontró el camino más corto entre cada par de nodos. El camino más corto fue encontrado utilizando el algoritmo de Dijkstra. Las mencionadas distancias se encuentran en la figura 21. Para la creación de los grafos y la generación de distancias se utilizó la librería NetworkX y para la visualización de dichos grafos se utilizó el programa Graphviz.

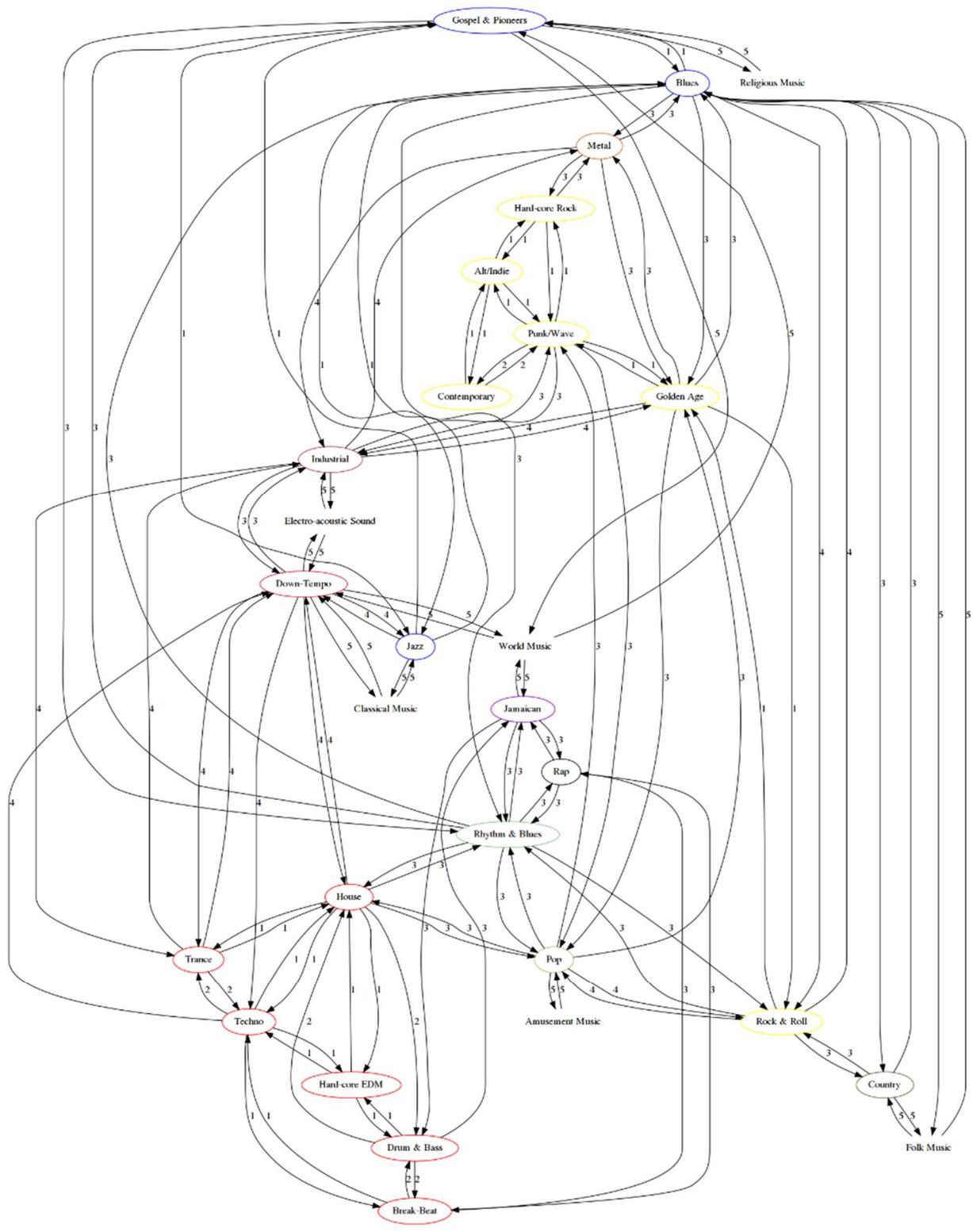


Figura 20. Grafo de representación de géneros musicales

	Gospel & Pioneers	Blues	Jazz	Rock & Roll	Golden Age	Punk/Wave	Hard-core Rock	Alt/Indie	Contemporary	Break-Beat	House	Techno	Drum & Bass	Hard-core EDM	Trance	Folk Music	Amusement Music	Religious Music	World Music	Classical Music	Electro-acoustic Sound	Country	Metal	Rhythm & Blues	Pop	Rap	Jamaican	Down-Tempo	Industrial
Gospel & Pioneers	0	1	1	5	4	5	6	6	7	8	6	7	8	7	7	6	11	5	5	6	10	4	4	3	6	6	6	5	8
Blues	1	0	1	4	3	4	5	5	6	8	6	7	8	7	7	5	11	6	6	6	10	3	3	3	6	6	6	5	7
Jazz	1	1	0	5	4	5	6	6	7	9	7	8	9	8	8	6	12	6	6	6	5	9	4	4	4	7	7	4	7
Rock & Roll	5	4	5	0	1	2	3	3	4	8	6	7	8	7	7	8	9	10	10	10	10	3	4	3	4	6	6	8	5
Golden Age	4	3	4	1	0	1	2	2	3	8	6	7	8	7	7	8	8	9	9	9	9	4	3	4	3	7	7	7	4
Punk/Wave	5	4	5	2	1	0	1	1	2	8	6	7	8	7	7	9	8	10	10	10	8	5	4	5	3	8	8	6	3
Hard-core Rock	6	5	6	3	2	1	0	1	2	9	7	8	9	8	8	10	9	11	11	11	9	6	3	6	4	9	9	7	4
Alt/Indie	6	5	6	3	2	1	0	1	9	7	8	9	8	8	10	9	11	11	11	11	9	6	4	6	4	9	9	7	4
Contemporary	7	6	7	4	3	2	1	0	10	8	9	10	9	9	11	10	12	12	12	10	7	5	7	5	10	10	8	5	
Break-Beat	8	8	9	8	8	8	9	9	10	0	2	1	2	2	3	13	10	13	10	10	11	11	5	5	3	5	5	7	
House	6	6	7	6	6	6	7	7	8	2	0	1	2	1	1	11	8	11	9	9	9	9	9	3	5	4	5	4	5
Techno	7	7	8	7	7	7	8	8	9	1	1	0	2	1	2	12	9	12	9	9	9	10	4	4	4	5	4	5	6
Drum & Bass	8	8	9	8	8	8	9	9	10	2	2	0	1	3	13	10	13	8	11	11	11	11	5	5	5	3	6	7	6
Hard-core EDM	7	7	8	7	7	7	8	8	9	2	1	1	1	0	2	12	9	12	9	10	10	10	10	4	4	5	4	5	6
Trance	7	7	8	7	7	7	8	8	9	3	1	2	3	2	0	12	9	12	9	9	9	10	8	4	4	6	6	4	4
Folk Music	6	5	6	8	8	9	10	10	11	13	11	12	13	12	12	0	16	11	11	11	15	5	8	8	11	11	11	10	12
Amusement Music	11	11	12	9	8	8	9	9	10	10	8	9	10	9	9	16	0	16	16	17	16	12	11	8	5	11	11	12	11
Religious Music	5	6	6	10	9	10	11	11	12	13	11	12	13	12	12	11	16	0	10	11	15	9	9	8	11	11	11	10	13
World Music	5	6	6	10	9	10	11	11	12	10	9	9	8	9	9	11	16	10	0	10	10	9	9	8	11	8	5	5	8
Classical Music	6	6	5	10	9	10	11	11	12	10	9	9	11	10	9	11	17	11	10	0	10	9	9	9	12	12	5	8	
Electro-acoustic Sound	10	10	9	10	9	8	9	9	10	10	9	9	11	10	9	15	16	15	10	10	0	13	9	12	11	13	14	5	5
Country	4	3	4	3	4	5	6	6	7	11	9	10	11	10	10	5	12	9	9	9	13	0	6	6	7	9	9	8	8
Metal	4	3	4	3	4	3	4	3	4	5	11	9	10	11	10	8	11	9	9	9	6	0	6	6	6	9	9	7	4
Rhythm & Blues	3	3	4	3	4	5	6	6	7	5	3	4	5	4	4	8	8	8	8	9	12	6	0	3	3	3	7	8	
Pop	6	6	7	4	3	3	4	4	5	5	3	4	5	4	4	11	5	11	11	12	11	7	6	3	0	6	6	7	6
Rap	6	6	7	6	7	8	9	9	10	3	5	4	5	5	6	11	11	11	8	12	13	9	9	3	6	0	3	8	10
Jamaican	6	6	7	6	7	8	9	9	10	5	5	5	3	4	6	11	11	11	5	12	14	9	9	3	6	3	0	9	10
Down-Tempo	5	5	4	8	7	6	7	7	8	5	4	4	6	5	4	10	12	10	5	5	5	8	7	7	7	8	9	0	3
Industrial	8	7	7	5	4	3	4	4	5	7	5	6	7	6	4	12	11	13	8	8	5	8	4	8	6	10	10	3	0

Figura 21. Matriz de distancias de géneros musicales.

Para la interacción con los músicos y la gestión de la base de datos referentes a los mismos se utilizó Django, el cual es un *framework* de desarrollo web basado en el paradigma Model-Template-View (MTV). Cabe destacar que MTV es un caso específico del paradigma de desarrollo de software Modelo-Vista-Controlador (MVC). El MVC es un patrón en el cual se separan los datos y los módulos que permiten manipularlos y observarlos, como se muestra en la figura 22. En este caso, la integración con Django los componentes del patrón de diseño son:

- Modelo: La base de datos de músicos y la base de datos de ensambles construidos.
- Controlador/Template: Las funciones implementadas para interactuar con dichas bases de datos, es decir, creación de perfiles de músicos, solicitudes de ensambles, consultas de las bases de datos, etc.
- Vista: Presentación de la información contenida en las bases de datos.

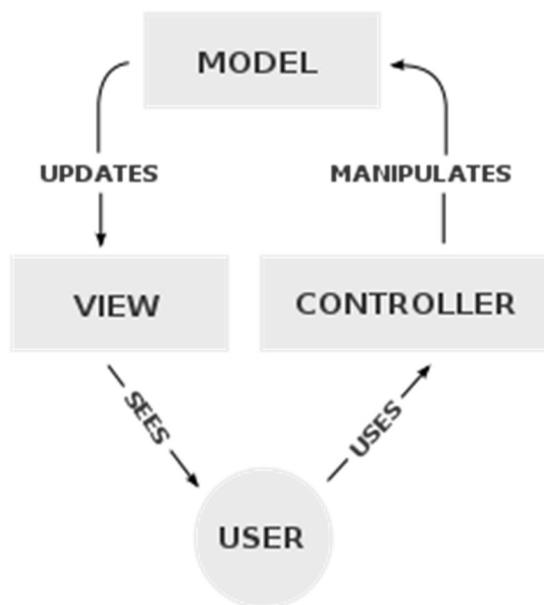


Figura 22. Diagrama de funcionamiento del modelo MVC.

Nota. Adaptado de Diagram of interactions within the MVC pattern [Fotografía], por Wikipedia, 2010, Commons (<https://commons.wikimedia.org/wiki/File:MVC-Process.svg>).

Django, al tratarse de un entorno de trabajo para el desarrollo de aplicaciones web, las componentes Template y View están integradas de manera tal que se genere código HTML de forma dinámica utilizando el lenguaje Django Template Language (DTL), al cual puede agregarse elementos de estilo web.

Para la generación de dichas vistas web se utilizó el mencionado DTL y Bootstrap, que es un entorno de trabajo para el desarrollo de sitios web, que implementa elementos de diseño basados en HTML y CSS.

Implementación

Herramientas

El desarrollo de la aplicación se realizó en Visual Studio Code en Ubuntu/Linux-GNU 18.04LTS y se utilizaron las siguientes herramientas:

FRAMEWORK	VERSIÓN
Python	Python 3.6.9
Django	Django 3.1.3
Bootstrap	Bootstrap 4.5
Graphviz	Graphviz 2.40
NetworkX	NetworkX 2.3

Tabla 2. Herramientas de desarrollo y sus versiones.

Implementación

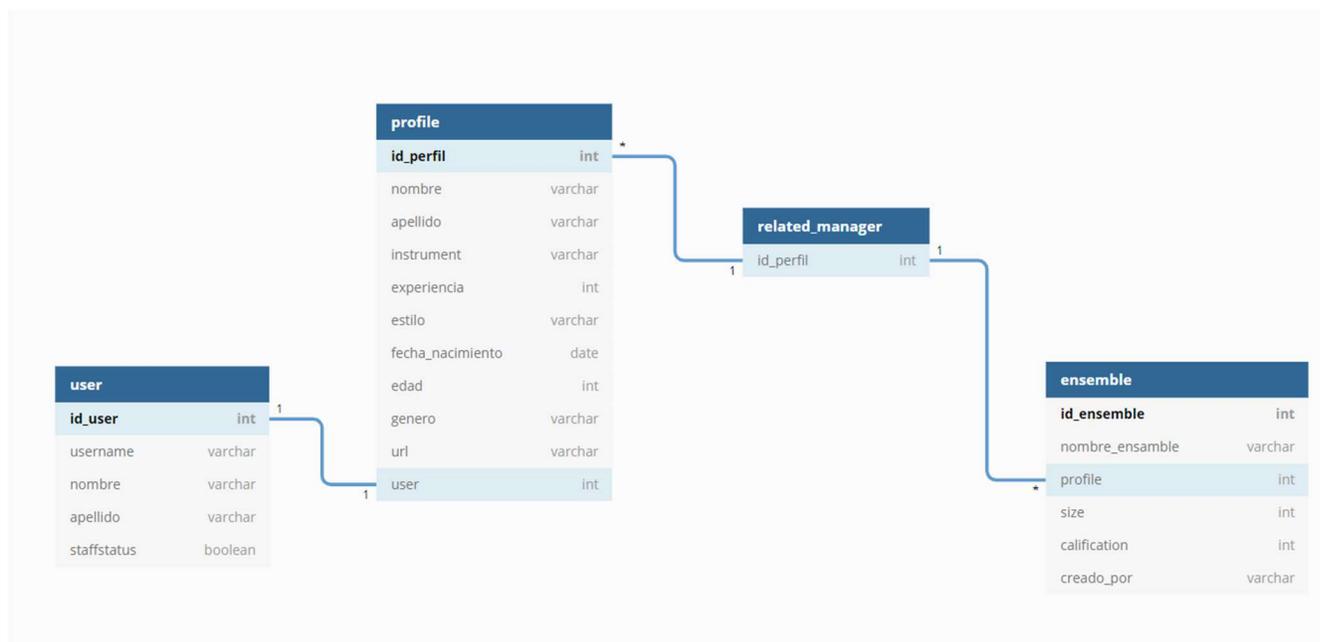


Figura 23. Diagrama de base de datos.

Como se muestra en la figura 22, se implementaron dos bases de datos. La primera correspondiente a los perfiles de los músicos y la segunda a los ensambles de músicos. Además, se utilizó una base de datos propia de Django para el manejo de usuarios. Se creó una relación de uno a uno entre un músico y un usuario. Gracias a esto, se puede gestionar sesiones y comprobar que los músicos estén autenticados para acceder a las funciones de la aplicación. De manera similar, es posible presentar una pantalla de inicio específica para los músicos anónimos que aún no han creado perfiles e incitarlo a unirse a la aplicación. Una vez que un músico crea un perfil, la creación de un usuario es automática; el músico nunca ingresa una contraseña ni dirección de correo. La generación automática de usuarios es una decisión tomada para restringir el acceso a la aplicación sin crear inconvenientes al músico. Además, este método de autenticación puede ser reemplazado a futuro.

Algoritmo de selección de ensambles

Como se expuso anteriormente en los casos usos, existen dos tipos de músicos. Únicamente los músicos autenticados, es decir, aquellos para los cuales existe una entrada en la tabla de usuarios y perfiles, pueden solicitar ensambles. Luego de que un músico solicita un ensamble, habiendo llenado el formulario satisfactoriamente, se recupera el perfil del músico que solicita el ensamble, el número total de músicos solicitado, y para cada uno de ellos los respectivos instrumentos, géneros musicales y niveles de experiencia.

Para generar los ensambles se utiliza un algoritmo de tipo glotón o *greedy*, el cual resuelve el problema de generar un ensamble, un músico a la vez. El músico que solicita la creación del ensamble es el primer músico seleccionado. Para facilitar la explicación este músico se llamará Músico A. Luego, para cada uno de los músicos solicitados se procede de la siguiente manera:

- Del conjunto de todos los músicos disponibles en la base de datos se excluye a los músicos ya seleccionados.
- Para cada uno de los músicos restantes se dispone en valor de la distancia entre el género musical del músico y el género musical solicitado por el Músico A. Esta cantidad será referida como distancia de estilo.
- Para cada uno de los músicos restantes se calcula el valor absoluto de la diferencia entre el nivel de experiencia del músico y el nivel de experiencia solicitado por el Músico A. Esta cantidad será referida como diferencia de experiencia.
- Para cada uno de los músicos restantes se calcula el valor absoluto de la diferencia entre la edad del músico y la edad del Músico A. Dicha cantidad será referida como diferencia de edad.
- Del conjunto de todos los músicos disponibles, se seleccionan solamente los músicos que tiene el instrumento solicitado por el Músico A.
- Se ordenan en forma ascendente todos los músicos restantes en función de:
 1. La distancia de estilo
 2. La diferencia de experiencia
 3. La diferencia de edad

De esta forma los músicos más compatibles se encuentran al principio de una lista ordenada.

- De esta lista ordenada de músicos y restringida por instrumento, se selecciona el músico que se encuentra en la primera posición, el cual es el más compatible en función de los campos requeridos por el Músico A.

- Finalmente, la calificación del ensamble se calcula sumando la distancia de estilo, la diferencia de experiencia, y la diferencia de edad para cada uno de los músicos seleccionados.

Como consecuencia del algoritmo utilizado los mejores ensambles tienen calificaciones más bajas. La mejor calificación posible es 0, correspondiente a un ensamble en el cual todos los músicos seleccionados cumplen con los requisitos del Músico A, y además tienen la misma edad en años que el Músico A. A continuación, en la figura 22, se muestra un pseudocódigo que explica el funcionamiento del algoritmo descrito.

```

MusicoA = User.objects.get(username = HTTP.request.user.username).musico
musicos_of_ensemble.append(MusicoA)
queryset = Musico.objects.SQL.exclude(id=MusicoA.id)
for musico in HTTP.form[musicos_requested]:
    if HTTP.form.cleaned_data[musico.es_requerido]:
        instrumento = HTTP.form.cleaned_data[musico.instrumento]
        estilo      = HTTP.form.cleaned_data[musico.estilo]
        experiencia = HTTP.form.cleaned_data[musico.experiencia]
        queryset = queryset.SQL.annotate(
            diferencia_experiencia = Abs(SQL.Field('experiencia') - experiencia),
            diferencia_edad       = Abs(SQL.Field('edad') - MusicoA.edad))
        musico_seleccionado = queryset.SQL.filter(
            instrumento__exact=instrumento).SQL.order_by(
                'distancia_estilo',
                'diferencia_experiencia',
                'diferencia_edad').SQL.exclude(
                    id__in=musicos_of_ensemble).SQL.first()
        musicos_of_ensemble.append(musico_seleccionado)
        distancia_estilo.append(musico_seleccionado.distancia_estilo)
        distancia_experiencia.append(musico_seleccionado.diferencia_experiencia)
        distancia_edad.append(musico_seleccionado.diferencia_edad)
    calificacion = reduce(add, distancia_estilo) +
        reduce(add, distancia_experiencia) +
        reduce(add, distancia_edad)
Ensemble.objects.SQL.create(
    nombre      = HTTP.form.cleaned_data[nombre],
    musicos     = musicos_of_ensemble,
    calificacion = calificacion
    fecha      = datetime.datetime.now()
    creador_por = MusicoA
)

```

Figura 24. Seudocódigo de algoritmo de creación de ensambles.

Casos de Estudio

Para la realización de pruebas se crearon 200 perfiles aleatorios. El nombre apellido y sexo de cada músico se recupera de archivo de texto y el resto de las características son generadas aleatoriamente. Además, se han creado alrededor de 20 músicos de forma manual. Usando los usuarios de dichos músicos se ha ingresado a la aplicación y se ha solicitado los siguientes ensambles.

Prueba 1

Para esta prueba se tomó el perfil de Mariasol Pineda, con instrumento voz, género musical Jazz, una experiencia intermedia y 23 años. Para dicha prueba se seleccionaron los instrumentos; guitarra, piano, batería y bajo, ya que son los instrumentos usados en la mayoría de las bandas populares dentro de la industria musical. En la primera iteración se seleccionaron los instrumentos mencionados y tanto en experiencia como en género musical se mantienen los valores seleccionados por default, es decir, pop y principiante, como se ilustra en la figura 25.

The screenshot shows the MusicMatch application interface. At the top, there is a navigation bar with 'MusicMatch', 'Home', 'Logout', 'Profiles', and 'Ensembles'. Below this is a form for requesting an ensemble. The form has a 'Nombre*' field containing 'Prueba1.0'. Below the name field is a table with 9 rows, each representing an instrument. Each row has three dropdown menus: 'Instrumento', 'Género', and 'Nivel', followed by a checkbox. The first four rows (Instrumento 2-5) have their checkboxes checked, and the last five rows (Instrumento 6-9) have their checkboxes unchecked. A blue 'Request Ensemble' button is located at the bottom left of the form.

Instrumento	Género	Nivel	Seleccionado
Instrumento 2	Guitarra	Pop	<input checked="" type="checkbox"/> *
Instrumento 3	Bajo	Pop	<input checked="" type="checkbox"/>
Instrumento 4	Bateria	Pop	<input checked="" type="checkbox"/>
Instrumento 5	Piano	Pop	<input checked="" type="checkbox"/>
Instrumento 6	Bateria	Pop	<input type="checkbox"/>
Instrumento 7	Piano	Pop	<input type="checkbox"/>
Instrumento 8	Flauta	Pop	<input type="checkbox"/>
Instrumento 9	Violin	Pop	<input type="checkbox"/>

Figura 25. Prueba 1 con estilo y nivel predeterminado.

De acuerdo con la figura 26, en esta iteración de la prueba, el puntaje obtenido fue de 82 puntos. En este caso el puntaje se debe en su gran mayoría a la diferencia de edad entre Manuel López (73 años) y Mariasol Pineda (23 años), es decir, 50 puntos. De la misma manera, la diferencia de edad con David García (46 años), aumenta 23 puntos al puntaje. Sin embargo, los géneros musicales de la mayoría de los integrantes se encuentran en el pop o Rock & Roll los cuales son cercanos.

MusicMatch Home Logout Profiles Ensembles ▾ Search Search

Prueba1.0 **82**

Mariasol Pineda2020-12-21

 <p>David García Batería Pop Principiante 46</p>	 <p>David Jiménez Bajo Pop Intermedio 26</p>	 <p>Jose Murillo Guitarra Pop Principiante 20</p>
 <p>Manuel López Piano Rock & Roll Intermedio 73</p>	 <p>Mariasol Pineda Voz Jazz Intermedio 23</p>	

Figura 26. Ensamble de resultado de la prueba 1.

En la segunda iteración de la prueba 1 se cambió únicamente los géneros musicales, mientras que la experiencia permaneció con los datos predeterminados como se muestra en la figura 27.

MusicMatch Home Logout Profiles Ensembles ▾

Nombre*

Prueba1.1

Instrumento 2	Guitarra ▾	Blues ▾	Principiante ▾	<input checked="" type="checkbox"/> *
Instrumento 3	Bajo ▾	Jazz ▾	Principiante ▾	<input checked="" type="checkbox"/>
Instrumento 4	Bateria ▾	Rock & Roll ▾	Principiante ▾	<input checked="" type="checkbox"/>
Instrumento 5	Piano ▾	Classical Music ▾	Principiante ▾	<input checked="" type="checkbox"/>
Instrumento 6	Bateria ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 7	Piano ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 8	Flauta ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 9	Violin ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>

Request Ensemble

Figura 27. Prueba 1 con nivel predeterminado.

Como se ilustra en la figura 28, el puntaje de la segunda iteración de la prueba 1 tiene un puntaje más alto que la primera iteración, lo que quiere decir que los integrantes de este ensamble se encuentran más alejados entre sí. Esta diferencia se debe, a más de la edad es que únicamente dos integrantes del ensamble cumplen la petición del género musical.

Para la tercera iteración de la prueba 1, como se muestra en la figura 29 se utilizan diferentes niveles de experiencias, conservando los mismos géneros de la anterior iteración.

El resultado de esta iteración tiene un puntaje aún más alto que la primera iteración, debido a la diferencia en los niveles de experiencia, como se muestra en la figura 30. De acuerdo con estos parámetros los músicos escogidos cambian, por lo tanto, también su edad y su género, lo cual, en este caso aumenta el puntaje.

MusicMatch Home Logout Profiles Ensembles ▾

Prueba1.1 **87**

Mariasol Pineda 2020-12-21

 <p>Andrés Cao Piano Classical Music Principiante 53</p>	 <p>Antonio Romo Guitarra Rock & Roll Principiante 23</p>	 <p>Ignacio Tejedor Bajo Classical Music Intermedio 52</p>
 <p>Manuel Guedes Bateria Rock & Roll Avanzado 45</p>	 <p>Mariasol Pineda Voz Jazz Intermedio 23</p>	

Figura 28. Resultado de ensamble de la prueba 1 con nivel predeterminado.

MusicMatch Home Logout Profiles Ensembles ▾

Nombre*

Prueba1.2

Instrumento 2	Guitarra ▾	Blues ▾	Avanzado ▾	<input checked="" type="checkbox"/> *
Instrumento 3	Bajo ▾	Jazz ▾	Intermedio ▾	<input checked="" type="checkbox"/>
Instrumento 4	Bateria ▾	Rock & Roll ▾	Experto ▾	<input checked="" type="checkbox"/>
Instrumento 5	Piano ▾	Classical Music ▾	Avanzado ▾	<input checked="" type="checkbox"/>
Instrumento 6	Bateria ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 7	Piano ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 8	Flauta ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 9	Violin ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>

Request Ensemble

Figura 29. Prueba 1 con nivel y estilo aleatorio.



MusicMatch Home Logout Profiles Ensembles ▾

Prueba1.2 104

Mariasol Pineda2020-12-21

 Andrés Cao Piano Classical Music Principiante 53	 Ignacio Tejedor Bajo Classical Music Intermedio 52	 Mariasol Pineda Voz Jazz Intermedio 23
 Mercedes Ligeró Guitarra Gospel & Pioneers Avanzado 43	 Silvia Santana Batería Rock & Roll Experto 44	

Figura 30. Resultado de ensamble de la prueba 1 con nivel y estilo aleatorio.

Prueba 2

Para esta prueba se tomó el perfil de Ana Salas, con instrumento guitarra, género musical Pop, una experiencia intermedia y 30 años. Como se muestra en la figura 31, al igual que en la prueba uno, se seleccionaron los instrumentos más comunes, es decir, voz, bajo, batería, y piano.

MusicMatch Home Logout Profiles Ensembles ▾

Nombre*
Prueba2.0

Instrumento 2	Voz ▾	Pop ▾	Principiante ▾	<input checked="" type="checkbox"/> *
Instrumento 3	Bajo ▾	Pop ▾	Principiante ▾	<input checked="" type="checkbox"/>
Instrumento 4	Bateria ▾	Pop ▾	Principiante ▾	<input checked="" type="checkbox"/>
Instrumento 5	Piano ▾	Pop ▾	Principiante ▾	<input checked="" type="checkbox"/>
Instrumento 6	Bateria ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 7	Piano ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 8	Flauta ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 9	Violin ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>

[Request Ensemble](#)

Figura 31. Prueba 2 con estilo y nivel predeterminado.

En esta iteración el puntaje fue más alto que en la prueba anterior, aunque este no es extremadamente bajo, un factor importante es la menor diferencia de edad entre los participantes como se muestra en la figura 32.

MusicMatch Home Logout Profiles Ensembles ▾

Prueba2.0 **68**

Ana Salas 2020-12-21

 Ana Salas Guitarra Pop Intermedio 30	 Daniela Costa Voz Pop Intermedio 25	 David García Bateria Pop Principiante 46
 Manuel López Piano Rock & Roll Intermedio 73	 Marta Aguilera Bajo Pop Intermedio 30	

Figura 32. Resultado de ensamble de prueba 2 con estilo y nivel predeterminado.

Como se muestra en la figura 33, en esta segunda iteración se modificaron los géneros, mientras que la experiencia se mantuvo constante como la predeterminada.

MusicMatch Home Logout Profiles Ensembles ▾

Nombre*

Prueba2.1

Instrumento 2	Guitarra ▾	Blues ▾	Principiante ▾	<input checked="" type="checkbox"/> *
Instrumento 3	Bajo ▾	Jazz ▾	Principiante ▾	<input checked="" type="checkbox"/>
Instrumento 4	Voz ▾	Rock & Roll ▾	Principiante ▾	<input checked="" type="checkbox"/>
Instrumento 5	Contrabajo ▾	Classical Music ▾	Principiante ▾	<input checked="" type="checkbox"/>
Instrumento 6	Batería ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 7	Piano ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 8	Flauta ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 9	Violín ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>

Request Ensemble

Figura 33. Prueba 2 con nivel predeterminado.

Como en el caso anterior, la mayor diferencia se da debido a la distancia de edad, sin embargo, está muy cercano al puntaje de la misma iteración dentro de la prueba 1, como se ilustra en la figura 34.



MusicMatch Home Logout Profiles Ensembles ▾ Search Search

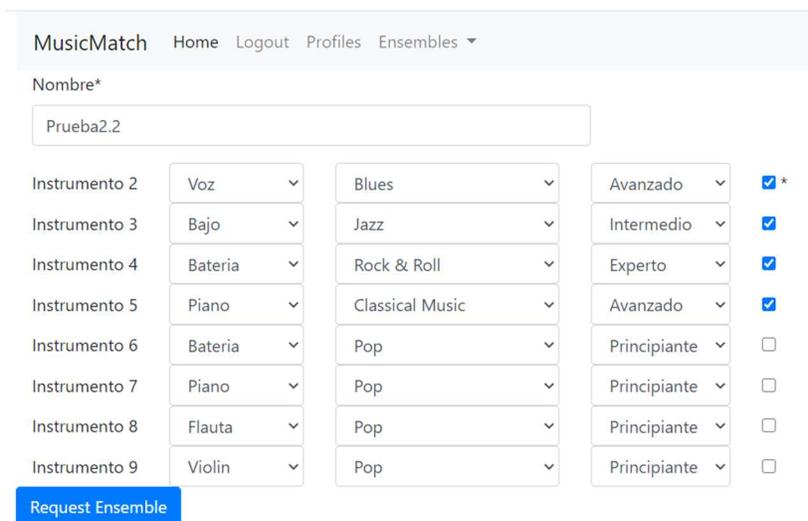
Prueba2.1 85

Ana Salas 2020-12-21

 Ana Salas Guitarra Pop Intermedio 30	 Antonio Romo Guitarra Rock & Roll Principiante 23	 Carlos Mayo Contrabajo Classical Music Avanzado 66
 Ignacio Tejedor Bajo Classical Music Intermedio 52	 Mercedes Cuesta Voz Pop Avanzado 42	

Figura 34. Resultado de ensamble de la prueba 2 con nivel predeterminado.

Como se muestra en la figura 35, en la última iteración de la prueba 2, se tomaron en cuenta géneros y experiencias aleatorias.



MusicMatch Home Logout Profiles Ensembles ▾

Nombre*

Prueba2.2

Instrumento 2	Voz ▾	Blues ▾	Avanzado ▾	<input checked="" type="checkbox"/> *
Instrumento 3	Bajo ▾	Jazz ▾	Intermedio ▾	<input checked="" type="checkbox"/>
Instrumento 4	Bateria ▾	Rock & Roll ▾	Experto ▾	<input checked="" type="checkbox"/>
Instrumento 5	Piano ▾	Classical Music ▾	Avanzado ▾	<input checked="" type="checkbox"/>
Instrumento 6	Bateria ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 7	Piano ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 8	Flauta ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>
Instrumento 9	Violin ▾	Pop ▾	Principiante ▾	<input type="checkbox"/>

Request Ensemble

Figura 35. Prueba 2 con nivel y estilo aleatorio.

Esta iteración fue la más exitosa entre ambas pruebas ya que arrojó un resultado de 64 puntos. Siendo este el mejor puntaje dentro de las pruebas. Como se ve ilustrado en la figura 36. Esto se debe a que los participantes se encuentran mucho más cercanos en edad, además de que los géneros que se seleccionaron eran muy cercanos a los requeridos.



Figura 36. Resultados de ensamble de la prueba 2 con nivel y estilo aleatorio.

Resultados

Como se pudo constatar en ambas pruebas, uno de los factores más importantes a considerar es la edad, ya que es esta la que más peso tiene dentro del puntaje. Parámetros como el género o el nivel de experiencia tienen un menor peso en el cálculo tomado en cuenta por el algoritmo. Aunque los datos fueron generados de manera aleatoria, existe una mayor cantidad de perfiles con edades avanzadas, por esta razón, si un usuario se encuentra cerca de la moda de los datos, sus puntajes

de acuerdo con los ensambles serán mucho mejores. De acuerdo con la experiencia con el uso de la aplicación, otro de los componentes importantes es el género o estilo con el que se encuentra registrado el perfil del usuario, ya que, dependiendo de los usuarios del sistema y cuál es el género musical más popular el puntaje cambiará de acuerdo con sus requerimientos.

Problemas Conocidos

Uso de Tablas Dinámicas con JavaScript para agregar nuevos músicos al ensamble

Se considera que la implementación de la forma HTTP en la cual el músico solicita un ensamble, debería generarse con código HTTP dinámico, es decir, usando JavaScript. Es evidente que el formato actual en el que se presenta un número específico de músicos a seleccionar funcionaría mejor si el músico que solicita el ensamble también especifica cuántos músicos requiere.

Para la implementación de dicha funcionalidad, se debe modificar el código del servidor y el código del cliente. En la parte del servidor es necesario crear un nuevo modelo "*MusicoSolicitado*" que tendrá los campos solicitados por el usuario. Los ensambles tendrán una relación de muchos a muchos con este nuevo modelo. La forma HTTP y la función asociada a esta se encarga de crear este nuevo conjunto de músicos solicitados y luego, la creación del ensamble se deberá realizar dentro del modelo Ensemble.

En la parte del cliente dentro de la *template* asociada se debe implementar un lazo que itere sobre el conjunto de músicos solicitados, además se debe implementar un botón para añadir al músico solicitado que corresponderá a aumentar un músico solicitado en el respectivo conjunto del ensamble, de igual forma un botón para eliminar el músico solicitado.

CONCLUSIONES

Conclusiones

- La solución encontrada en la primera iteración, al problema planteado es satisfactoria. La aplicación cumple con los requerimientos y entrega al músico un ensamble aproximado de acuerdo con los datos ingresados por el usuario. Sin embargo, al tratarse de un prototipo aún es posible aumentar y mejorar funcionalidades.
- El desarrollo de la presente aplicación web ha sido un proyecto en el cual se han aplicado varios temas estudiados durante la carrera de Ingeniería en Ciencias de la Computación, como la creación y manejo de bases de datos, la generación de grafos, y algoritmos para solucionar problemas que se pueden modelar con dicha herramienta matemática, los protocolos de transferencia de hipertexto, la autenticación de usuarios, y la redirección de puertos de redes, entre otros.
- El aprendizaje del entorno de trabajo Django ha sido satisfactorio, debido al paradigma MTV, el código escrito es modular y puede reutilizarse para otros proyectos futuros.

Trabajo futuro

La clasificación de géneros musicales es una aproximación simplificada, por lo tanto, esta podría ser mejorada para obtener un resultado mucho más preciso.

Por otra parte, la creación de usuarios y su autenticación son parte de un módulo propio de Django el cual es fácilmente reemplazable.

Además, al tratarse de un proyecto en el que el principal elemento es la música se prevé utilizar la API del servicio de *streaming* de música Spotify. En dicho caso la autenticación de usuarios se la realizará con la cuenta de Spotify. Entonces, una vez vinculada una cuenta se podrá disponer de

los gustos musicales precisos de los músicos. Con esta información se mejora el método para la creación de ensambles.

Por otro lado, al ser una primera iteración en el desarrollo del algoritmo que crea los ensambles, es evidente que puede mejorarse su funcionamiento. Al momento, debido a la distribución de las edades de los músicos, la diferencia de edades contribuye excesivamente a la calificación del ensamble construido. Una manera de solucionar dicho problema es normalizar los valores de estas diferencias de edad con una normalización min-max, es decir, la máxima diferencia de edades entre todos los músicos disponibles tomará un valor de 1 y la mínima de 0, luego todas las diferencias tomarán valores entre 0 y 1.

Así mismo, en los perfiles de los músicos se podrá incluir un campo que indique una calificación asignada, la cual corresponderá a la validación de su experiencia con un instrumento. Luego, los músicos con calificaciones superiores sean prioritarios durante la construcción del ensamble.

Además, se debería incluir la posibilidad de que un músico toque más de un instrumento, en varios estilos de música. Incluso sería conveniente considerar una clasificación de instrumentos musicales para asignar distancias entre ellos y construir los ensambles tratando de minimizar estas distancias al igual que se lo hace con las distancias entre estilos de música.

REFERENCIAS BIBLIOGRÁFICAS

- Boer, D., Fischer, R., Strack, M., Bond, M. H., Lo, E., & Lam, J. (2011). How shared preferences in music create bonds between people: values as the missing link. *Personality & social psychology bulletin*, 37(9), 1159–1171. <https://doi.org/10.1177/0146167211407521>
- Grape, C., Sandgren, M., Hansson, L. O., Ericson, M., & Theorell, T. (2003). Does singing promote well-being?: An empirical study of professional and amateur singers during a singing lesson. *Integrative physiological and behavioral science: the official journal of the Pavlovian Society*, 38(1), 65–74. <https://doi.org/10.1007/BF02734261>
- Launay, J., Dean, R. & Bailes F. (2014). Synchronising movements with the sounds of a virtual partner enhances+ partner likeability. *Cognitive processing*, 15(4), 491–501. <https://doi.org/10.1007/s10339-014-0618-0>
- Tarr, B., Launay, J. & Dunbar, R. (2014). Music and social bonding: “self-other” merging and neurohormonal mechanisms. *Frontiers in Psychology*. Obtenido el 16 de octubre de 2020, desde <https://doi.org/10.3389/fpsyg.2014.01096>
- Zorn, J. (1973). Effectiveness of Chamber Music Ensemble Experience. *Journal of Research in Music Education*, 21(1), 40-47. Obtenido el 16 de octubre de 2020, desde <http://www.jstor.org/stable/3343977>