The Harmony Project

Bringing People Together Through Music

COS597A Advanced Topics in Computer Science: Long Term Memory in AI - Vector Search and Databases
Final Project Paper Writeup

Author: Yagiz Devre
Advisors: Prof. Dr. Edo Liberty, Prof. Dr. Matthijs Douze & Dr. Nataly Brukhim

1 Introduction

In our modern world, the digital era has significantly transformed how we interact with each other and maintain social connections, which is now maintained mostly through mobile applications such as Instagram, Snapchat and BeReal. What made BeReal different was the ability to see what your friends were doing in real time, the realness of the social network got people's attention. That was the reason why BeReal became a viral hit in a matter of weeks and maintained a strong user-base, due to the users' "hunger" for different ways to interact with people. Just like BeReal, The Harmony Project is inspired to become something unique, capitalizing on this trend by offering a platform where music, a universal language that is a deeply personal interest, becomes the key point of social interaction.

Unlike conventional social media like Instagram and Tinder, Harmony provides an innovative way for users to discover and connect with others who resonate with their musical tastes. Unlike most of the social networking platforms, in Harmony, it does not matter how many people you are following, or how many people are following you. It is all about individualization and individual personalized experience of the users. It is not a social app that you would use to gain followers, it is an avenue for users to find friends and relationships based on the most universal concept of our civilization: Music.

Harmony's uniqueness lies in the approach to utilize user preferences through Spotify for an in-depth analysis of users' musical behavior. The concept of Harmony is the idea that musical preferences are reflective of individual personalities, a possible medium for establishing meaningful connections that was overlooked for many years even though we have countless music streaming platforms.

By integrating comprehensive user data from Spotify, including top artists, songs, and playlists, Harmony crafts a distinct user personality(as a vector) for every user. Furthermore, users can create new friendships but also deepen their appreciation of music. This innovative approach of using music as a key point in social interactions is set to revolutionize online interactions. In summary, Harmony is not just another social networking platform; it's a community brought together by the universal language of music, designed to create lasting connections in an increasingly digital world.

2 Problem Statement and Hypothesized Solution

2.a Problem Statement

As mentioned earlier, we have countless number of music streaming applications such as Spotify, Apple Music, Youtube Music etc. Yet, a social networking application that is centralized through user's music preferences is still waiting to be innovated. To effectively design a functional social networking experience for Harmony, there is a key problem that needs to be addressed: efficient representation of user preferences and songs, which will be the main point of this paper.

Music-based social networking, as mentioned earlier, has the challenge of effective representation and analysis of songs to facilitate meaningful user interaction. Traditional methods of categorizing music by genre or artist and generating recommendations based on a series of *if statements* is too broad and will eventually fail to capture the level of deepness of individual musical preferences of the users. This limits the ability to accurately match users with users and users with songs based on their specific taste in music at a given time. The problem lies in the complexity of musical preferences, which often contains wide array of genres, artists, styles, tones, and moods, therefore requiring a more sophisticated approach for representation and analysis instead of pure data representation.

2.b Hypothesized Solution

The solution to this challenge is quite simple and intuitive: representing songs and user preferences as vectors and utilizing a vector database for efficient and accurate matching through queries based on how "similar" two vectors are to each other through *cosine* relation. This can be done in two separate steps for both user and song vectors.

Vector Generation: To generate these vectors, Harmony uses an advanced method: utilizing OpenAI's vector embeddings and text generation models. Using a text generation model for the data collected from Spotify and then turning it into vectors enables Harmony to represent Spotify in such a way that it is easier to identify patterns and preferences, which are then translated into numerical values in the vector of fixed length and dimension. This considers the diversity and range of the user's musical interests.

Storage of Vectors for Matching: Once the user and song vectors are created, they are stored in the Pinecone Vector Database. When the platform performs a matching operation through a query request, Pinecone is used to compare the vectors of different users or the users with songs. Users whose vectors are closer in the multi-dimensional vector space are considered to have more similar music tastes and are thus more likely to be matched. This vector-based matching is more dynamic and precise compared to traditional matching methods, as it reflects a deep understanding of each user's unique music preference.

By representing users and tracks as vectors, Harmony offers a highly personalized and unique recommendations of songs and matches of users based on a detailed understanding of individual preferences. User vectors are updated as the user preferences change over time, specifically every 4-6 weeks, ensuring that the platform remains responsive to evolving changes of the user's whereas song vectors stay constant but increase in quantity as more and more users bring new songs that are vectorized and stored in the database. This makes Harmony a scalable solution as well due to the fact that we do not need to get all the 1.2 Million songs on Spotify and vectorize each of the songs since it would take over $600 just for the vector generation. Instead, since we are adding approximately 500 new songs to the database each time a user signs up to the application. To conclude, representing users as vectors is a sophisticated and effective way to represent the musical tastes, enabling Harmony to provide personalized and accurate matches based on shared musical interests.

3 Methodology

3.a Technical Stack

Fundamentally Harmony is designed to be a mobile application for the ease of use. Furthermore, it is built on top of a React Native architecture since code written on React Native powers both iOS and Android devices through the same codebase, therefore enabling accessibility to everyone regardless of their operating system. Likewise, React Native uses Java Script and Type Script, and its wide-ranging support for various APIs and services was another key decision maker for my choice on using the framework. Harmony, through the JS API calls to multiple endpoints, creates an integration of the different components required for the platform which can be described as follows.

Spotify Web API: As the primary data collection API of Harmony's music-based features such as fetching user data and also playing tracks, the Spotify Web API provides access to a list of very important and useful user music data such as top artists, songs, and playlists that can be used to model the user's preferences. Custom API for React Native built on top of the Web API was developed and utilized for Harmony specifically.

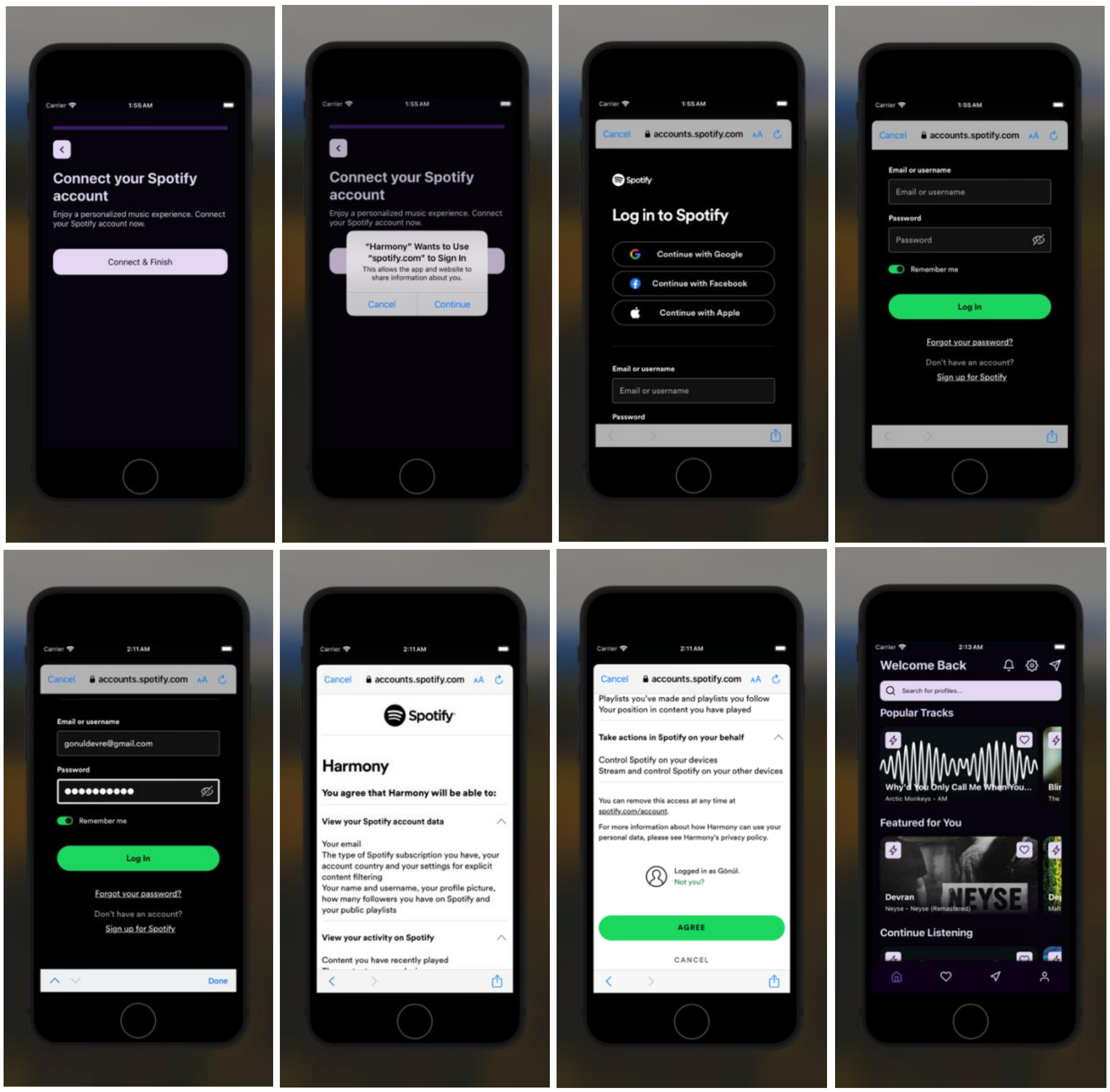
Pinecone API: Perhaps the second most important element of the entire application, Pinecone is used for handling and storing high-dimensional user data represented as vectors of 1536 dimensions. This API plays a crucial role in storing and retrieving user and song vector embeddings based on similarity between User-User and User-Song vector pairs which is essential for the platform's matching algorithms.

OpenAI API(GPT3.5 Turbo Instruct and ADA2 Vector Embeddings): Leveraging the power of OpenAI's text generation and vector embeddings, Harmony manages to create vectors of fixed dimensions for advanced analysis of user profiles. This component is critical for converting user music preferences and song details into detailed vectors for a precise matching.

Firebase Database: Harmony uses a No-SQL database along with Pinecone database. The main reason for this decision was that Firebase provides a scalable backend solution for email authentication and user creation which is highly favorable from the security perspective. Likewise, it is very affordable and highly efficient to store profile images and match information in the real-time database for storing user-specific data.

3.b Data Collection

As mentioned earlier Harmony works with the Spotify accounts of the user through a custom API for React Native that I developed on top of the Web API. Since there is no custom integration with React Native, an API system on top of the Spotify Web API that functions with React Native was developed. This API system effectively acts as a connection between the user and the application. The user data is fetched through this API at the last step of Signup Flow described in *Appendix A*. The detailed flow of the connection can be illustrated as follows:



Enjoy a personalized music experience. Connect your Spotify account now.

This allows the app and website to share information about you.

You agree that Harmony will be able to:

View your Spotify account data

Your email
The type of Spotify subscription you have, your account country and your settings for explicit content filtering
Your name and username, your profile picture, how many followers you have on Spotify and your public playlists

View your activity on Spotify

Content you have recently played

Playlists you've made and playlists you follow
Your position in content you have played

Take actions in Spotify on your behalf

Control Spotify on your devices
Stream and control Spotify on your other devices

You can remove this access at any time at spotify.com/account.

For more information about how Harmony can use your personal data, please see Harmony's privacy policy.

***Figure 1:** Spotify Connection Through API*

As seen in the *Figure 1*, when the user reaches to the 16th step in signup flow, described in *Appendix A*, they are asked to connect their Spotify accounts to Harmony and finish the signup process. Then, they are directed to a login screen within the application to connect their Spotify accounts. If the user already has Spotify as an application in their phone, the API redirects the user to the Spotify Application instead of the *accounts.spotify.com* link and shows the 6th screen instead of requiring for an email and password combination. Finally, when the user agrees the data usage agreement, they are redirected to Harmony instantly.

Meanwhile, on the backend, a bearer access token is generated that allows fetching user information that is valid for 60 minutes and a refresh token to generate a new access token after 60 minutes both stored in the user's respective document as separate fields in the database as follows:

Access Token: BQBXNTamjBs…beeU3WGCnDDoJUsxw3ZV1DNy
Refresh Token: AQDHzi19aVN…KLnKA45mQqefG4mlw8nCIB7XE

Whenever the user's access token times out, the refresh token generates a new access token and updates the already existing token in the database automatically in real time. Finally, the API uses the following scopes:

scopes: [*'user-top-read'*, *'user-library-read'*, *'user-read-playback-position'*,
'playlist-read-private', *'user-read-currently-playing'*, *'user-read-recently-played'*,
'user-read-playback-state', *'user-modify-playback-state'*, *'streaming'*,
'user-read-email', *'user-read-private'*,
],

These scopes enable Harmony to securely retrieve user data such as top 50 artists short(4 week period), medium(6 month period) and long term(several years) along with their genres, top 50 songs listened by the user in short, medium and long term, 50 songs liked by the user recently, 50 songs listened by the user most recently(just like browser history), currently listened song, user's public and private playlists and the songs in those playlists, the 50 artists that the user follows and top 10 songs of each of those artists.

Furthermore, instead of just reading data through GET API calls, Harmony can also use PUT and DELETE API calls as well which enable the users to like a song in Harmony (thus adding the song to their liked songs in Spotify) without ever leaving the application. This can be shown in the following figure set:

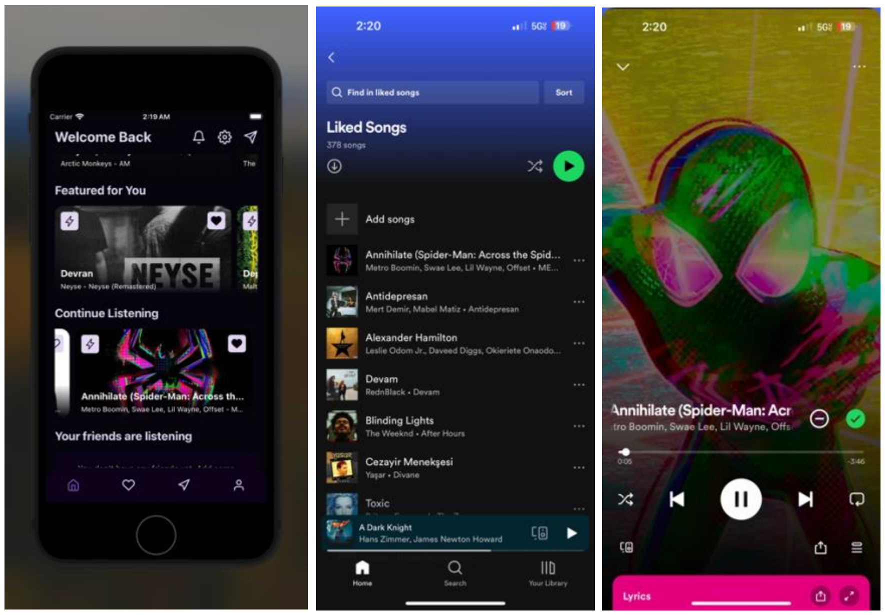


Figure 2: *Realtime Spotify Connection*

As illustrated in the *Figure 2*, when the user likes a song by pressing the heart-icon on the top right corner of the song tile (in the case of *Figure 2*, the song demonstrated is *Annihilate by Metro Boomin*), the song is directly added to their Liked Songs list in Spotify. Likewise, when the user presses the tile image, the song automatically starts playing in Spotify in the background therefore the user does not need to ever leave the application and open Spotify to search for the song that they liked. Finally, the Zap icon on the top left corner allows users to listen to the preview, the overall most listened part, of the song therefore, instead of playing the entire song they can just listen to the preview.

3.c Song Representation as Vectors

Each song is represented as a multi-dimensional vector. Specifically, through the use of Spotify API, a unique text is generated using the OpenAI GPT-3.5 Turbo Instruct API, and the text will then be converted into a vector through the OpenAI Vector Embeddings API, ADA 002. This process can be illustrated as follows:

```
{
“Album Name”: “AM”,
“Artist”: “Arctic Monkeys”,
“ArtistID”: “7Ln80lUS6He07XvHI8qqHH”,
“ID”: “5FVd6KXrgO9B3JPmC8OPst”,
“Name”: “Do I Wanna Know?”,
“Popularity”: 90,
“Acousticness”: 0.186,
“Danceability”: 0.548,
“Energy”: 0.532,
“Instrumentalness”: 0.000263,
“Liveness”: 0.217,
“Loudness”: -7.596,
“Speechiness”: 0.0323,
“Tempo”:85.03,
“Valance”: 0.405
}
```

Arctic Monkeys’ **"Do I Wanna Know?"** is another highly popular track from their album "**AM**" boasting a popularity score of **90**. The song has a moody and introspective vibe with an **acousticness of 0.186**, giving it a somewhat stripped-down feel compared to some of their other tracks. Its **danceability rating of 0.548** and **energy level of 0.532** make it a **steady and contemplative piece**. The instrumentalness, though modest at 0.000263, adds depth to the composition. With a tempo of 85.03**, it sets a deliberate pace**, and the liveness of 0.217 **suggests a live ambiance**. The track’s lyrics, coupled with Alex Turner’s signature delivery with a speechiness of 0.0323, contribute to its emotional depth. "Do I Wanna Know?" is a standout piece in Arctic Monkeys’ discography, **striking a balance between raw emotion and musical finesse.**

[-0.0192117076, -0.0152770886, 0.0103684273, -0.0302978512, -0.030707974, 0.0191732589, ... -0.0245945, -0.0227105711]
Vector of 1536 Dimensions

As seen above, the text generated through GPT-3.5 Turbo Instruct effectively involves key features of the data collected through Spotify(shown in bold font) and thus the vector also encapsulates various attributes of the song, such as liveness, loudness, tempo, artist, and more, which is more informative than traditional genre-based classifications. The final vector generated by Ada Text Embedding 002 effectively guaranties that the vector for every song will be 1536 dimensions, therefore ensuring a scalable and reliable metric for similarity between vectors through Pinecone database queries.

3.d User Preference Representation as Vectors

Just like the song vectors, in Harmony, users’ musical preferences are also captured through a process that is very similar to the representation of songs as vectors. Fundamentally, Harmony again uses GPT-3.5 Turbo Instruct to generate a text about the user and then Ada Text Embedding 002 converts that to a vector of size 1536. However, the data used for text generation is primarily about the user’s Top Artists, Top Songs, Top Genres instead of song audio features of the songs. This ensures a nuanced representation tastes and since Ada Text Embedding 002 is utilized to generate the vectors, the generated vectors have the same dimensions as the song vectors, enabling a powerful recommendation system between User-Song interaction as well. An example vector generation can be illustrated as follows:

{
"**Top 1 Artist**": "Eminem",
"**Top 2 Artist**": "Trent Reznor and Atticus Ross",
"**Top 3 Artist**": "Taylor Swift",
"**Top 4 Artist**": "Kenan Doglu",
"**Top 5 Artist**": "Ahmet Kaya",
"**Most Recent Favorite Artist**": "J-Cole",
"**Top Genre**": Pop,
"**Top 2 Genre**": Hip Hop,
.
.
.
"**10th Most Liked Song**": Shake it Off
}

My musical tastes are eclectic, encompassing a wide array of genres that resonate with my diverse interests. At the core of my preferences lies a deep appreciation for **soundtracks** and **hip hop** music. Among my **favorite** artists are the iconic duo **Trent Reznor and Atticus Ross,** known for their evocative film scores, and **Eminem**, whose **sharp lyrics** and **dynamic rhythms** never fail to captivate me. **Taylor Swift's** storytelling prowess and catchy melodies hold a special place in my playlist, along with the soulful tunes of **Ahmet Kaya** and the vibrant beats of **Kenan Doğulu** from the **Turkish pop scene**. **J-Cole's** thoughtful rap narratives are another source of inspiration. Additionally, I find myself drawn to the unique sounds of **British soundtracks**, which transport me to different worlds, as well as the energetic beats of **southern hip hop** and the catchy rhythms of **pop rap**. This eclectic mix of genres beautifully encapsulates the vast spectrum of my musical journey, reflecting the varied moods and moments I cherish in my life.

[0.00128538581, -0.0182422344, -0.00562177878, -0.048601184, -0.00458406657, ..., 0.0213370956, -0.0105625903, 0.0149796735, -0.0429655723]
Vector of 1536 Dimensions

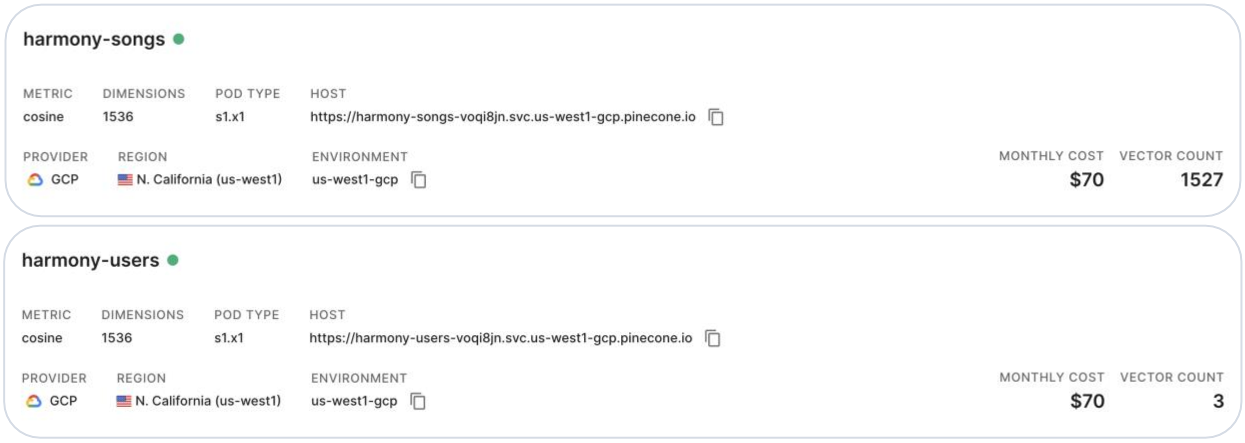
As seen above, the text generated through GPT-3.5 Turbo Instruct effectively encapsulates key features of the Spotify data. This enables Harmony to generate resulting vectors in Ada 002 Text Embedding and represent various attributes of the user's musical preferences, such as favored artists, listening frequency, genre diversity, and more. Just like song representations, this representation goes beyond the genre-based classifications to a more in-depth, preference-based metric.

3.e Database Structure

As mentioned in the 3.a Technical Stack section, Harmony uses two database structures. These database architectures are designed to efficiently handle both the complex vector data using Pinecone and user management using Firebase. Pinecone is used for vector storage and similarity matching, and Firebase is used for user data storage and authentication processes.

3.e.i Pinecone

Pinecone is the selected vector database for Harmony, chosen for its ability to manage and query high-dimensional data. Fundamentally, Harmony uses 2 indices for storing vectors: *harmony-songs* and *harmony-users* which hold song vectors and user vectors respectively as shown in the *Figure 3*.


harmony-songs ●

METRIC	DIMENSIONS	POD TYPE	HOST
cosine	1536	s1.x1	https://harmony-songs-voqi8jn.svc.us-west1-gcp.pinecone.io

PROVIDER	REGION	ENVIRONMENT	MONTHLY COST	VECTOR COUNT
GCP	N. California (us-west1)	us-west1-gcp	$70	1527

harmony-users ●

METRIC	DIMENSIONS	POD TYPE	HOST
cosine	1536	s1.x1	https://harmony-users-voqi8jn.svc.us-west1-gcp.pinecone.io

PROVIDER	REGION	ENVIRONMENT	MONTHLY COST	VECTOR COUNT
GCP	N. California (us-west1)	us-west1-gcp	$70	3

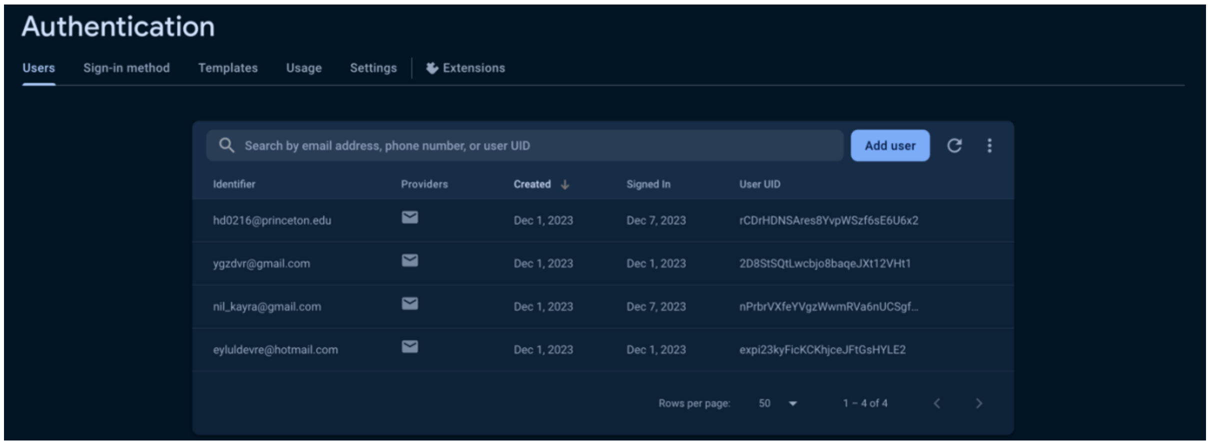
Figure 3: *Pinecone Indices for Harmony*

These indices are central to the platform's recommendation and matching features. The structure of Pinecone is specifically optimized for similarity searches based on *cosine* similarity, allowing for accurate retrieval of vectors.

This vector database functionality is key to delivering personalized content and connections to users based on their unique music tastes. The use of Pinecone also ensures consistency, efficiency and reliability in the matching algorithm described in *Section 4: Algorithm*. Since this is a vector database, fast retrieval of vectors is possible in real-time.

3.e.ii Firebase

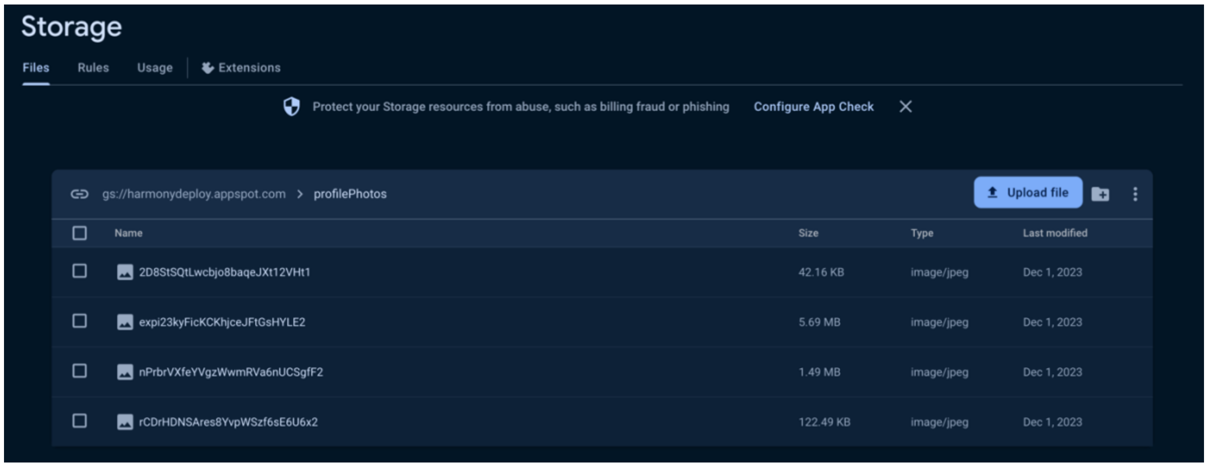
Firebase is used to handle the non-vector aspects of Harmony's data, including user profiles, authentication credentials, profile pictures, and various metadata. As a NoSQL database, Firebase operates in real-time and facilitates instantaneous updates across all user devices, such as updating popular songs, updating matches, friend requests etc. all in real time. This enhances the user experience since the data in the app is constantly evolving and changing to best represent the user that is using the application. Firebase's comprehensive tools, including a secure authentication service for email password combination, enables Harmony to maintain a secure and efficient user management system. Harmony uses 3 of Firebase tools which can be illustrated as follows:



Identifier	Providers	Created	Signed In	User UID
hd0216@princeton.edu	✉	Dec 1, 2023	Dec 7, 2023	rCDrHDNSAres8YvpWSzf6sE6U6x2
ygzdvr@gmail.com	✉	Dec 1, 2023	Dec 1, 2023	2D8StSQtLwcbjo8baqeJXt12VHt1
nil_kayra@gmail.com	✉	Dec 1, 2023	Dec 7, 2023	nPrbrVXfeYVgzWwmRVa6nUCSgf...
eyluldevre@hotmail.com	✉	Dec 1, 2023	Dec 1, 2023	expi23kyFicKCKhjceJFtGsHYLE2

Rows per page: 50 1 – 4 of 4

Figure 4: *Firebase Authentication*

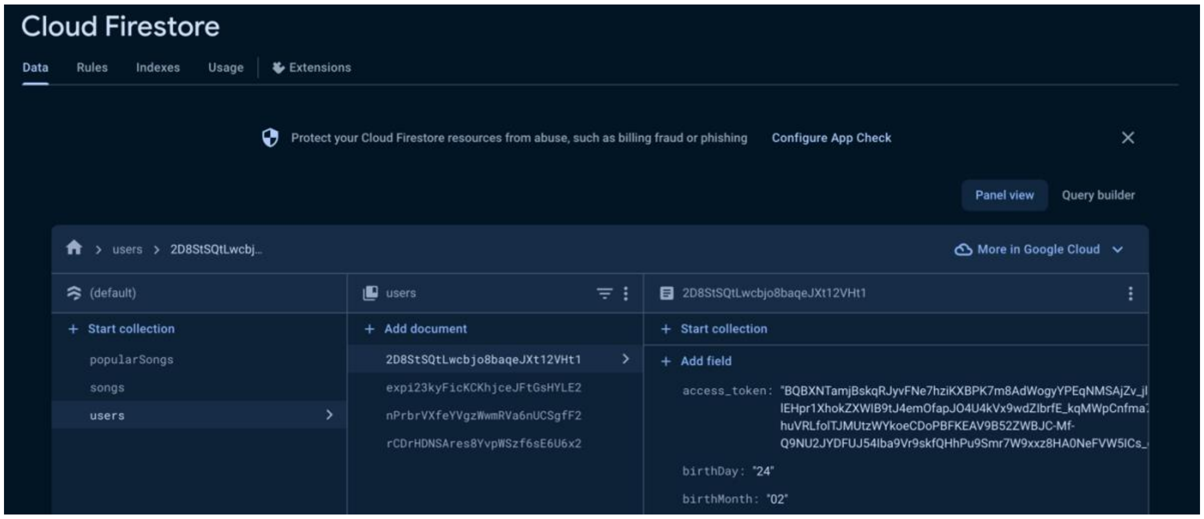


Protect your Storage resources from abuse, such as billing fraud or phishing

gs://harmonydeploy.appspot.com > profilePhotos

Name	Size	Type	Last modified
2D8StSQtLwcbjo8baqeJXt12VHt1	42.16 KB	image/jpeg	Dec 1, 2023
expi23kyFicKCKhjceJFtGsHYLE2	5.69 MB	image/jpeg	Dec 1, 2023
nPrbrVXfeYVgzWwmRVa6nUCSgfF2	1.49 MB	image/jpeg	Dec 1, 2023
rCDrHDNSAres8YvpWSzf6sE6U6x2	122.49 KB	image/jpeg	Dec 1, 2023

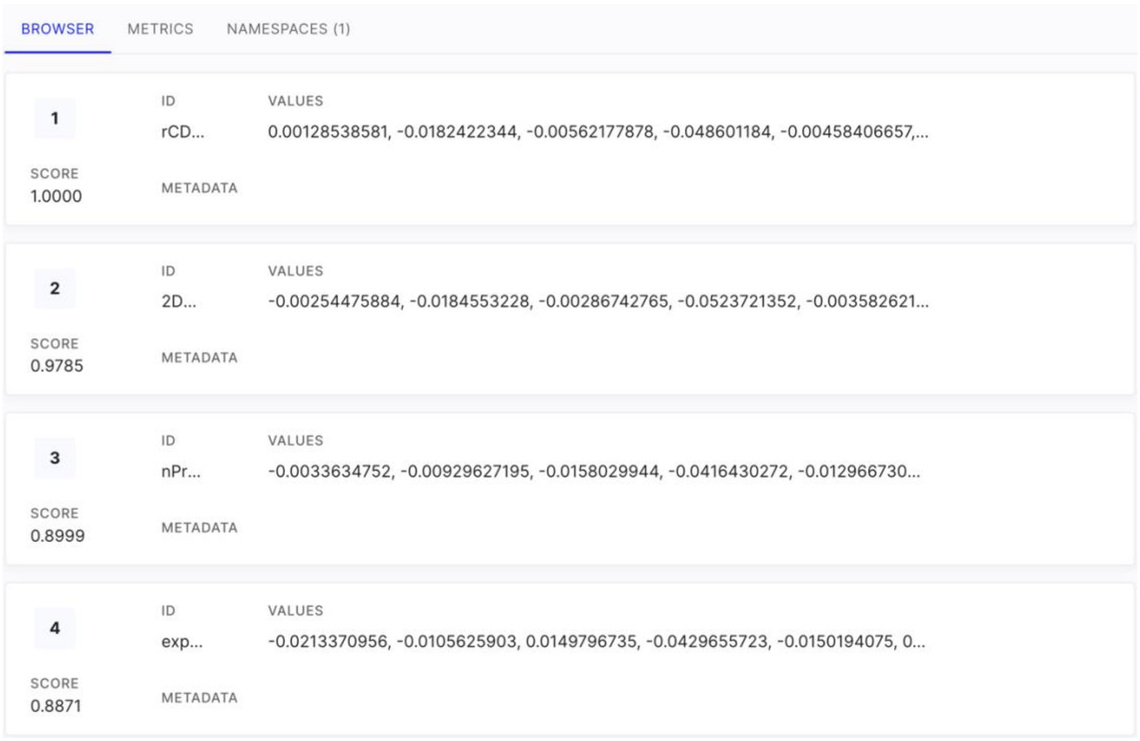
Figure 5: *Firebase Storage*



```
(default)          users                           2D8StSQtLwcbjo8baqeJXt12VHt1
  popularSongs       2D8StSQtLwcbjo8baqeJXt12VHt1    access_token: "BQBXNTamjBskqRJyvFNe7hziKXBPK7m8AdWogyYPEqNMSAjZv_jl
  songs              expi23kyFicKCKhjceJFtGsHYLE2                  lEHpr1XhokZXWIB9tJ4emOfapJO4U4kVx9wdZIbrfE_kqMWpCnfma
  users              nPrbrVXfeYVgzWwmRVa6nUCSgfF2                  huVRLfoITJMUtzWYkoeCDoPBFKEAV9B52ZWBJC-Mf-
                     rCDrHDNSAres8YvpWSzf6sE6U6x2                  Q9NU2JYDFUJ54Iba9Vr9skfQHhPu9Smr7W9xxz8HA0NeFVW5ICs_
                                                     birthDay: "24"
                                                     birthMonth: "02"
```

Figure 6: *Firebase Cloud Firestore*

As seen in the *Figures 4, 5, and 6*, the users are stored with a unique ID(UID) which is consistent throughout all the features of Firebase. The unique ID of the user generated in the Authentication section is used to also store the users as distinct documents in the "users" collection in Firestore represented in *Figure 6.* Likewise, the profile photos of the users are also categorized and separated by the same distinct UID as shown in *Figure 5.* Finally, every user is also stored with this same UID in Pinecone database as well in *harmony-users* index which can be illustrated as follows.



	SCORE	ID	VALUES	METADATA
1	1.0000	rCD...	0.00128538581, -0.0182422344, -0.00562177878, -0.048601184, -0.00458406657,...	
2	0.9785	2D...	-0.00254475884, -0.0184553228, -0.00286742765, -0.0523721352, -0.003582621...	
3	0.8999	nPr...	-0.0033634752, -0.00929627195, -0.0158029944, -0.0416430272, -0.012966730...	
4	0.8871	exp...	-0.0213370956, -0.0105625903, 0.0149796735, -0.0429655723, -0.0150194075, 0...	

Figure 7: *Pinecone UIDs*

Together, Pinecone and Firebase provide Harmony with a sophisticated database infrastructure. Pinecone's specialized vector handling capabilities pair with Firebase's data and authentication management to form the backbone of Harmony's service, ensuring that both the music matching precision and user experience are maintained at the highest standard.

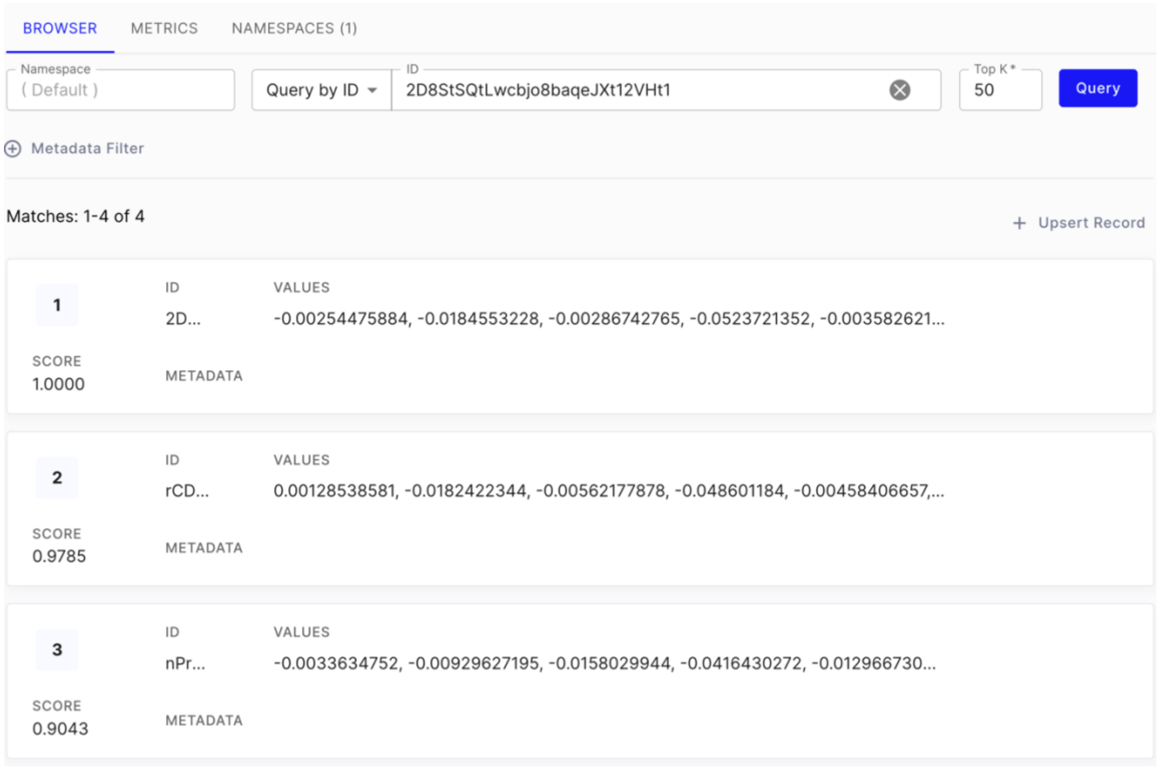
4 Algorithm

Harmony's algorithmic model is designed in such a way that allows the users to match with other users but also see song recommendations based on their music preferences. This algorithm is effectively an analysis within a 1536-dimensional vector space of 2 indices, where both songs and user preferences are represented as vectors and the problem we are solving is effectively choosing the most similar vectors for a given query. These vectors are compared using *cosine* similarity to get the degree of likeness, which informs the matching process across various domains of the application.

4.a User-User Matching

Harmony's most special feature, the User-User matching using taste of music uses an algorithm that is an efficient process that determines the similarity between two user vectors in a 1536-dimensional space. Utilizing *cosine* similarity, the algorithm calculates the *cosine* of the angle between two vectors, which serves as a measure of orientation and not magnitude, making it well-suited for high-dimensional data.

This measured score ranges from -1 to 1, where a *cosine* similarity of 1 implies perfect alignment of preferences, and thus a potential match. To optimize the experience, the algorithm employs a "top-k retrieval" approach, where only the highest scoring vectors, i.e., the most similar users, are considered as potential matches for a given user. As an example, look at the *Figure 8*.



Namespace: (Default)
Query by ID
ID: 2D8StSQtLwcbjo8baqeJXt12VHt1
Top K *: 50
Query

Matches: 1-4 of 4

	ID	VALUES
1	2D...	-0.00254475884, -0.0184553228, -0.00286742765, -0.0523721352, -0.003582621...
SCORE 1.0000	METADATA	
2	rCD...	0.00128538581, -0.0182422344, -0.00562177878, -0.048601184, -0.00458406657,...
SCORE 0.9785	METADATA	
3	nPr...	-0.0033634752, -0.00929627195, -0.0158029944, -0.0416430272, -0.012966730...
SCORE 0.9043	METADATA	

Figure 8: *Top-K Retrieval Approach*

As seen in *Figure 8*, a similarity query for the user with the UID *2D8StSQtLwcbjo8baqeJXt12VHt1* returns an ordered list of vectors and the UIDs that map to the vectors representing a user and their preference. To test the accuracy of the methodology Harmony uses, the vector generation algorithm was run for two Spotify accounts and thus generated 3 users with 3 distinct vectors. Two of those 3 distinct users used the same Spotify data but due to the randomness factor in GPT-3.5 Turbo Instruct, 2 very similar but non-identical vectors were obtained which was crucial to test User-User matching. In the ideal case those 2 very similar but non-identical vectors would score the highest since they represent the same user behavior with just slight differences, which is what is observed in *Figure 8* as well.

The top scorer vector is obviously the vector itself since it perfectly aligns with itself. But the other 2 vectors have scores less than 1.0, meaning that they are different vectors. One of the accounts specifically performs very well and aligns perfectly with the queried user, with a score of 0.9785. This user is the duplicate user generated to test the validity of the approach used and proved that the algorithm works and returns best matching users based on music preferences. This ensures that each user is presented with a curated set of potential connections that best mirror their own music tastes.

Finally, these "top-k" matches is filtered based on the user's preferences that were asked in *Appendix A: Signup Flow* such as gender, interested gender, mode selection etc. For instance, if the current user is not looking for a "date", the users in the matched array who actually is looking for a "date" is removed from the array and the match list is updated with the next highest scoring user that is not already in the array. These matches are then presented to the user in real-time in Harmony Match Screen shown in *Figure 9*. This entire process ensures a more personalized matching experience for the users as well.

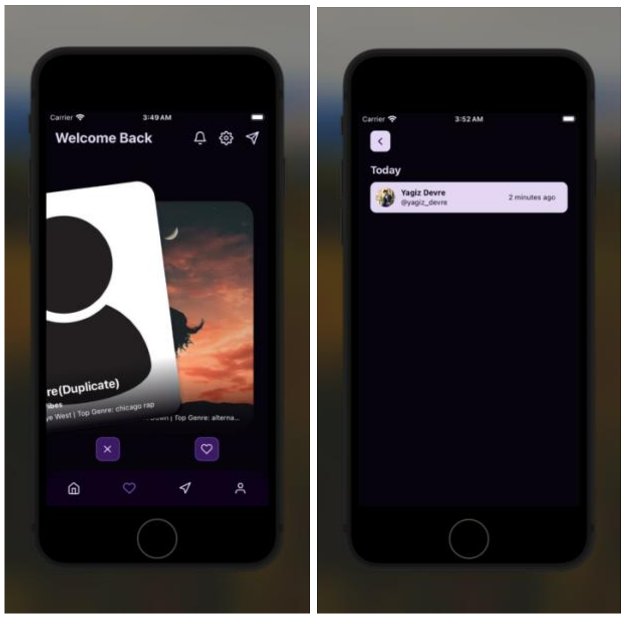
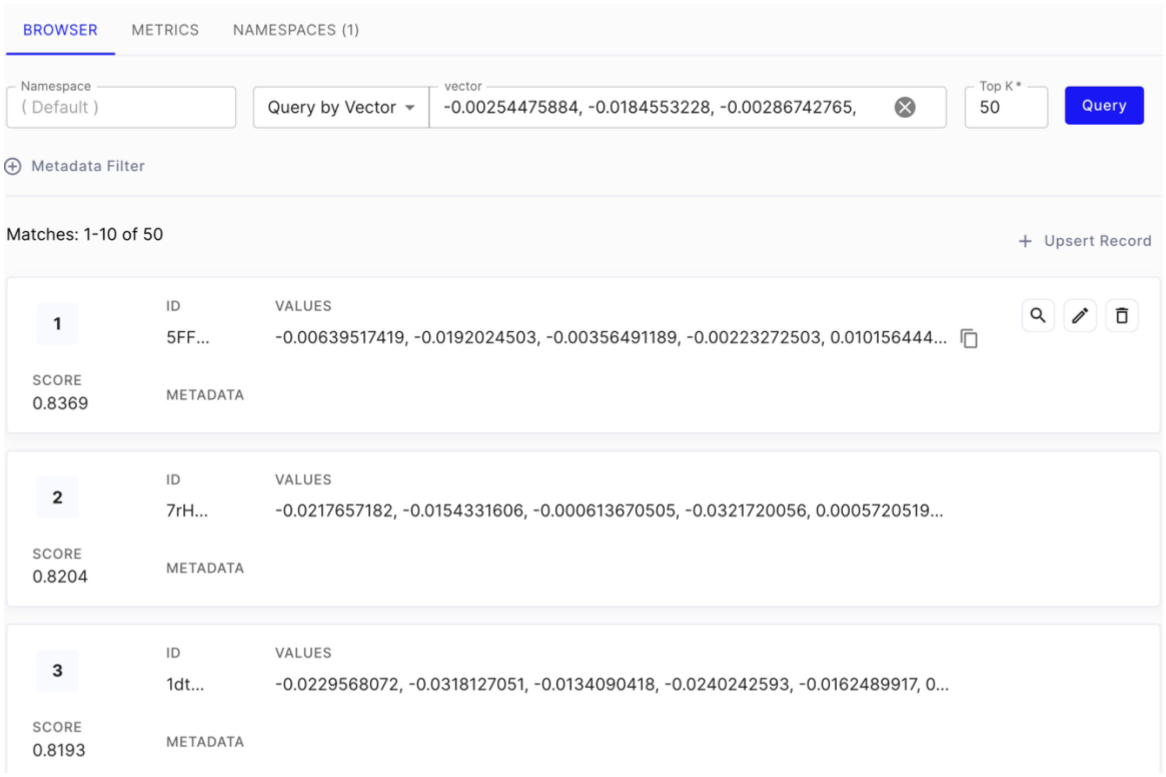


Figure 9: *Real Time Match Screen*
(A left swipe adds the swiped user to passed users array, a right pass sends a friend request to the other user automatically in real-time as in the second image)

4.b User-Song Matching

User-Song matching utilizes the principles of *cosine* similarity to the relationship between users and songs just like the User-User matching. Each song's similarity vector stored in *harmony-songs* index in relation to a user's preference vector stored in *harmony-users* index is computed to find the *cosine* similarity score. The algorithm then sorts these scores and applies a "top k" strategy to present the user with a selection of songs that exhibit the highest similarity scores. This method is particularly effective in recommending songs that align well with the user's current musical taste profile, as it captures their preferences across a spectrum of musical songs, attributes, and genres. As an example *Figure 10* can be used.



BROWSER METRICS NAMESPACES (1)

Namespace (Default) | Query by Vector | vector -0.00254475884, -0.0184553228, -0.00286742765, | Top K* 50 | Query

Metadata Filter

Matches: 1-10 of 50

Upsert Record

	ID	VALUES
1	5FF...	-0.00639517419, -0.0192024503, -0.00356491189, -0.00223272503, 0.010156444...
SCORE 0.8369	METADATA	
2	7rH...	-0.0217657182, -0.0154331606, -0.000613670505, -0.0321720056, 0.0005720519...
SCORE 0.8204	METADATA	
3	1dt...	-0.0229568072, -0.0318127051, -0.0134090418, -0.0240242593, -0.0162489917, 0...
SCORE 0.8193	METADATA	

Figure 10: *User Vector Querying in Songs Index Pinecone*

As the figure shows, the vector representing the user with the UID *2D8StSQtLwcbjo8baqeJXt12VHt1* that is stored in *harmony-users* index can be queried in *harmony-songs* index to generate top matching User-Song pairs. For instance, the first 3 songs are all represent Turkish Rock, a genre which the user with *2D8StSQtLwcbjo8baqeJXt12VHt1(*this was actually my personal Spotify account that I created a while ago for free in Turkey before coming to US and opening my current Spotify Premium account) listened a lot. Finally, these songs are then transferred to Harmony and added to the "Featured for You" section to add one more layer of personalization as seen in *Figure 11*.

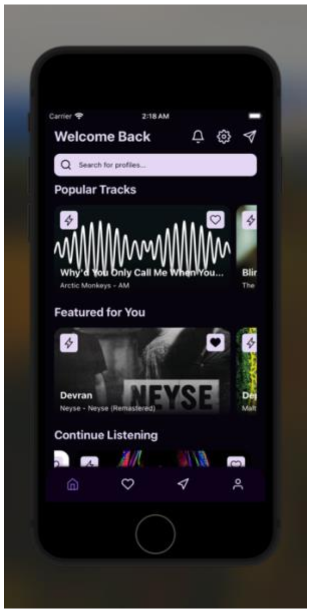


Figure 11: *Featured for You section in the Home Screen*

4.c User-User-Song Matching

The user-user-song matching algorithm is perhaps the most complex algorithm in Harmony's features, as it merges the principles applied in user-user and user-song matchings. The algorithm starts by calculating the *cosine* similarity between two user vectors to establish a baseline of shared musical interest.

From there, it proceeds to identify song vectors that maintain high *cosine* similarity to both users concurrently. The algorithm does this by taking the mean of the two user vectors as follows.

Suppose we have two vectors for representing two users $\boldsymbol{A}$ and $\boldsymbol{B}$ where $A = (a_1, a_2, \ldots, a_{1536})$ and $\boldsymbol{B} = (b_1, b_2, \ldots, b_{1536})$.

Since they are both the user vectors, these vectors are of the same dimension, 1536 to be precise. The "Harmony Vector" that blends the preferences of these two users therefore can be calculated by the mean of these two vectors, denoted as H(H is for Harmony), is calculated by taking the average of the corresponding elements of A and B. More formally, and mathematically, H can be represented as:

$$H = \left(\frac{a_1 + b_1}{2}, \frac{a_2 + b_2}{2}, \ldots, \frac{a_{1536} + b_{1536}}{2}\right)$$

This new vector is then used for a "top-k" retrieval from the *harmony-songs* index. The "top-k" results from this computation suggest songs that are at the intersection of both users' preferences, effectively capturing the musical commonness between the two users. These results are then represented in the Harmony Screen as shown in *Figure 12*.

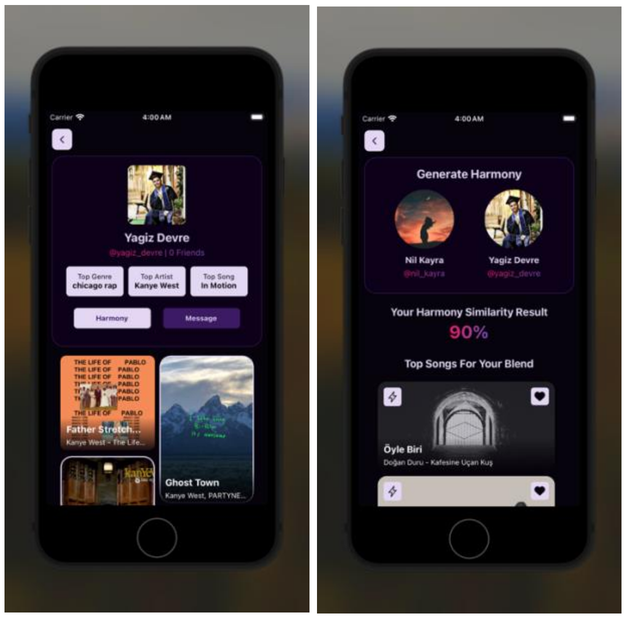


Figure 12: *Song Retrieval based on the Harmony Vector of Two Users*

The averaging part of the algorithm ensures that the songs recommended are likely to enhance the social bond by effectively reaching both users. The use of 1536-dimensional vectors and *cosine* similarity offers a state-of-the-art to music-based social networking, enabling Harmony to operate with efficiency thanks to the reliable and effective Pinecone Database while keeping precision and personalization at its core.

5. Future Extensions

Based on the feedback I got after the presentation on 1st of December, I believe launching Harmony as a downloadable product on Apple App Store and Google Play Store is a highly favorable trajectory for the future of the project. However, before the official launch, Harmony still needs some key features to become the leading music-based social networking platform. Primarily, I have identified 3 key updates that needs to be utilized. These envisioned future extensions are selected to align with the core philosophy of the platform while introducing innovative features based on user needs.

Algorithmic Refinement
An algorithmic refinement in terms of a "Feed" page is a critical part of Harmony's future roadmap. As mentioned, Harmony is all about making true and valuable connections with your friends over the concept of music. Therefore adding a "Feed/Explore" section where you see what your friends have been doing will be essential. Therefore allowing users to share a new concept called "memories" (instead of posts) will be a future development goal. As the field of artificial intelligence continues to advance, there is an opportunity to refine this feed algorithm further and sort the "memories" shared by your friends using vector databases.

Likewise, currently I am experimenting the implementation of Ticketmaster API as well. If successful, as a future extension, users will be able to see musical events that align with their musical preferences as well and even purchase tickets in Harmony directly. These refinements will allow for more personalized user experience where you will be able to see and experience the most meaningful memories and events according to your musical preferences.

Enhanced Social Features
One key feature that is missing is the messaging feature. You can add friends, see what they are listening, but none of this does not matter unless you have an effective communication system between you and your friends. Therefore, a more interactive and engaging way for users to connect over music and interact with each other can be done through implementing a messaging feature. A Third-Party messaging API such as Stream Messaging would be sufficient. Considering that Stream Messaging API also allows video and audio calls, these features can also be added to the application.

Along with messaging, other enhanced social features could include the introduction of shared listening sessions, where users can experience music synchronously. Likewise, allowing users to send songs and playlists directly through messaging feature as a widget can bring a lot of value to the application. That way, users would be able to share mixtapes that they made for their "special ones" just like in 90s.

Finally, a referral system can be implemented to create a "selective" feeling in the application. By enriching the platform's social connectivity tools, users can form deeper bonds and create shared musical experiences with each other.

Monetization Strategies
Finally, to ensure the platform's sustainability and continued innovation, a monetization strategy is needed. I personally believe that advertisement strategies and a freemium option would be beneficial for the continuation of the application. I recognize the importance of developing a monetization model that is both ethical and adds value to the user experience.

Potential strategies could include premium features for an enhanced user experience, such as exclusive access to certain musical content, unlimited daily matches, and being able to see who viewed your profile. Additionally, partnerships with artists, record labels, and event promoters could offer users unique opportunities and benefits while providing a revenue stream for the platform.

These extensions represent just a fraction of the potential growth scenarios Again, considering the ethics, morals and ethos of Harmony, the implementation of these features will be approached thoughtfully, with a continuous focus on user privacy, security, and maintaining the details of what makes Harmony a unique space for music lovers.

6. Conclusion

To conclude, I have started this journey to bring two things that I love in this life: Music and Technology. I believe, my mission to redefine social connectivity through the universal language of music shows a lot of promise. With its innovative use of AI vector databases and a user-centric approach, Harmony promises a new way we form social interactions online. Especially the underlying algorithm that enables a representation of users and songs as distinct vectors is truly and only possible thanks to the state-of-the-art vector database that is used: Pincone.

As I look ahead, I am personally excited about the potential expansions that promise to enrich the platform even further, enhancing the way users discover music and connect with one another. While the current landscape of Harmony is promising, there are several features that is needed for a more functional social networking platform. The journey ahead is filled with opportunities for growth. In conclusion, Harmony is not just an application; it is a community, a new social canvas waiting to be painted with countless memories.

7. Acknowledgments

The Harmony Project stands as the perfect example of technology and music, two things that were essential to my life. I would like to special thank to the course instructors of COS597A AI Vector Databases and my advisors in this project Prof. Dr. Edo Liberty, Prof. Dr. Matthijs Douze and Dr. Nataly Brukhim for their insightful feedback and support throughout the development of the project. I would like to thank them again for such an amazing semester filled with inspirational and innovative concepts which broadened my understanding of vector search and vector databases. I have truly learned a lot about vector databases along with project management and application development throughout this course.

I also appreciate the valuable input and feedback from the initial user base of Harmony (first 7 users of the application), whose engagement and feedback have been crucial in refining Harmony's user experience, debugging the algorithms mentioned and making a presentable application. Lastly, I extend my thanks to the music lovers who will use Harmony in the future, whose passion for music will inspire and drive the continuous evolution of the platform.

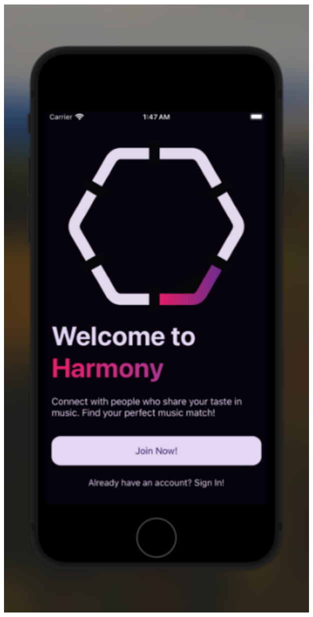
8. Code and Contributions

All the code, proposal, presentation, and final paper related to The Harmony Project can be found in the following open source GitHub Repository:

https://github.com/ygzdvr/Harmony

Appendix

Appendix A: Signup Flow

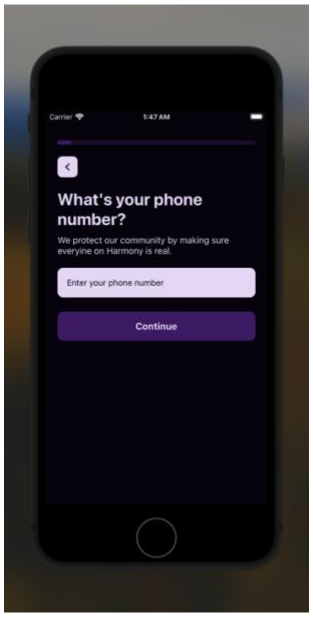

Carrier 1:47 AM

Welcome to Harmony

Connect with people who share your taste in music. Find your perfect music match!

Join Now!

Already have an account? Sign In!

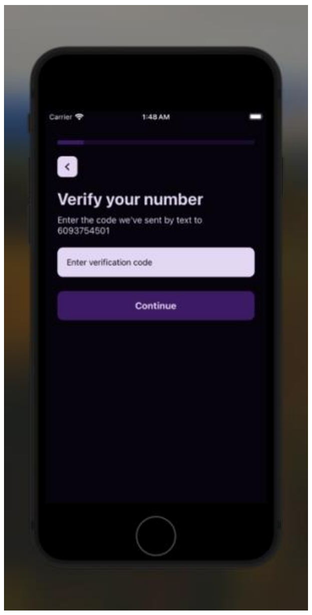

Carrier 1:47 AM

What's your phone number?

We protect our community by making sure everyine on Harmony is real.

Enter your phone number

Continue

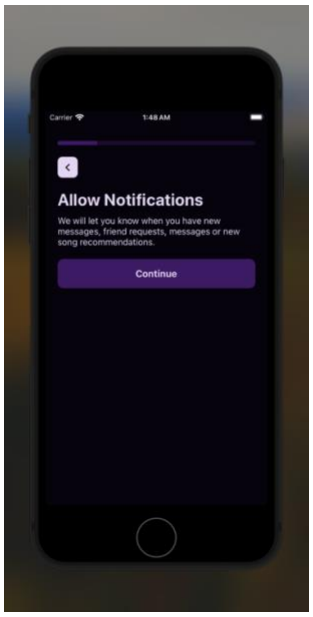

Carrier 1:48 AM

Verify your number

Enter the code we've sent by text to 6093754501

Enter verification code

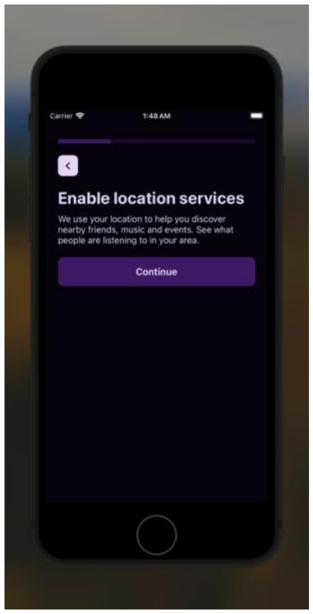
Continue


Carrier 1:48 AM

Allow Notifications

We will let you know when you have new messages, friend requests, messages or new song recommendations.

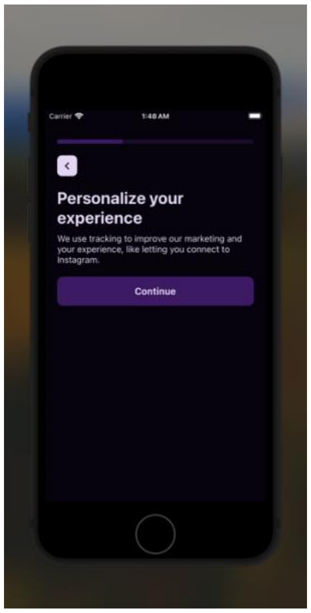
Continue


Carrier 1:48 AM

Enable location services

We use your location to help you discover nearby friends, music and events. See what people are listening to in your area.

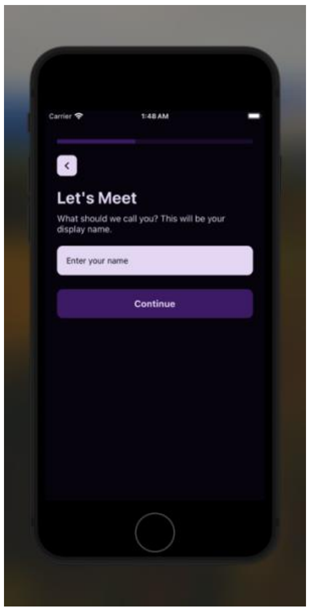
Continue


Carrier 1:48 AM

Personalize your experience

We use tracking to improve our marketing and your experience, like letting you connect to Instagram.

Continue

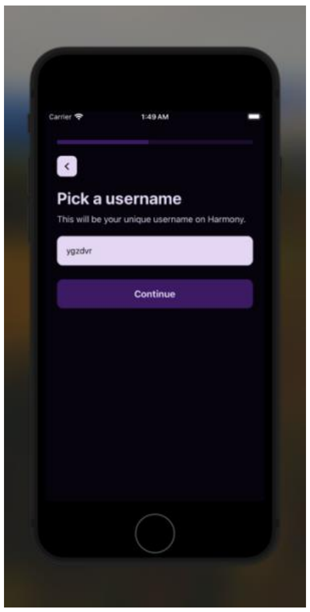

Carrier 1:48 AM

Let's Meet

What should we call you? This will be your display name.

Enter your name

Continue

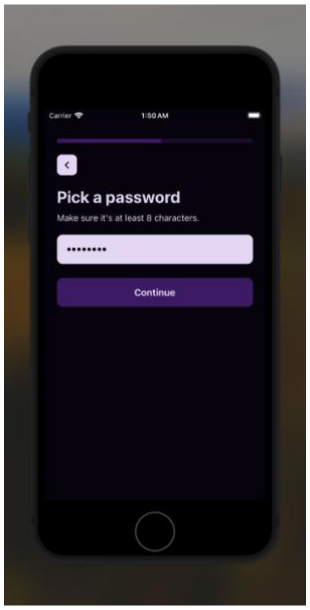

Carrier 1:49 AM

Pick a username

This will be your unique username on Harmony.

ygzdvr

Continue

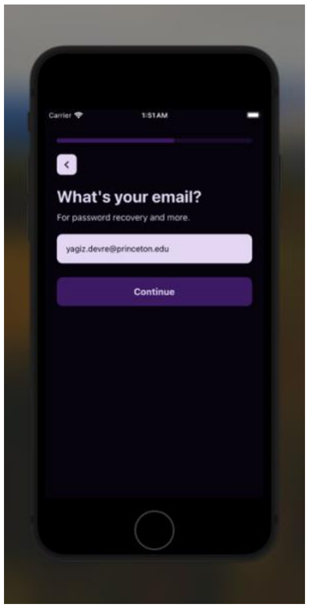

Carrier 1:50 AM

Pick a password

Make sure it's at least 8 characters.

••••••••

Continue

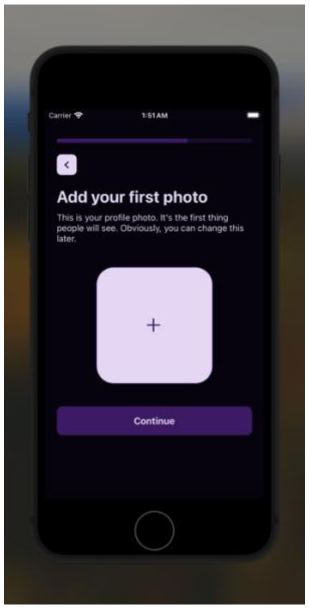

Carrier 1:51 AM

What's your email?

For password recovery and more.

yagiz.devre@princeton.edu

Continue

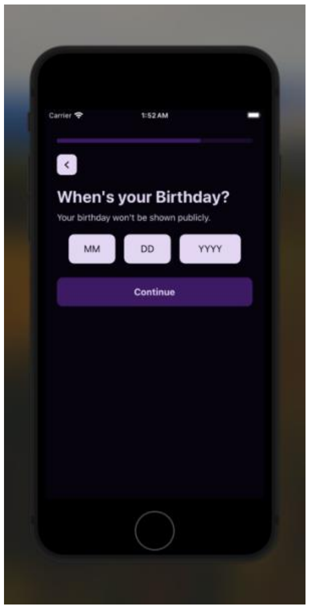

Carrier 1:51 AM

Add your first photo

This is your profile photo. It's the first thing people will see. Obviously, you can change this later.

+

Continue

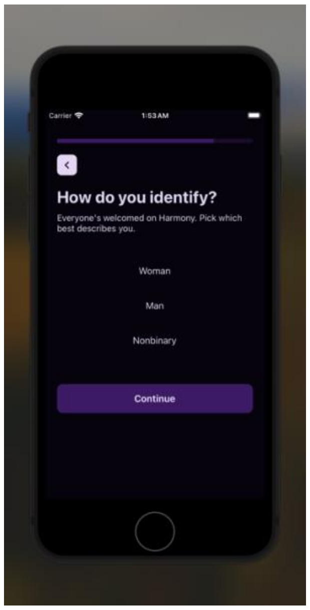

Carrier 1:52 AM

When's your Birthday?

Your birthday won't be shown publicly.

MM DD YYYY

Continue


Carrier 1:53 AM

How do you identify?

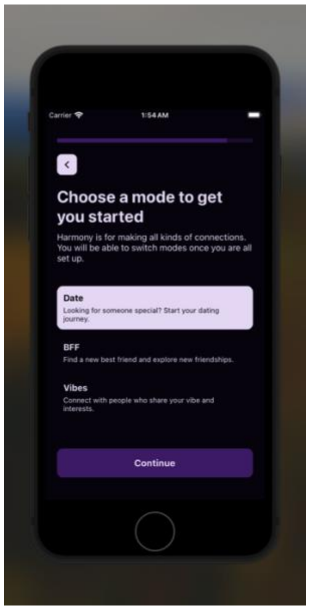
Everyone's welcomed on Harmony. Pick which best describes you.

Woman

Man

Nonbinary

Continue


Carrier 1:54 AM

Choose a mode to get you started

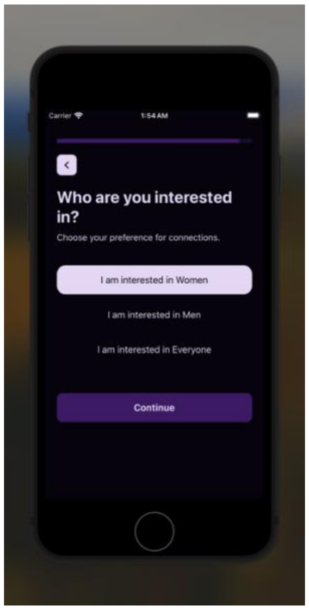
Harmony is for making all kinds of connections. You will be able to switch modes once you are all set up.

Date
Looking for someone special? Start your dating journey.

BFF
Find a new best friend and explore new friendships.

Vibes
Connect with people who share your vibe and interests.

Continue


Carrier 1:54 AM

Who are you interested in?

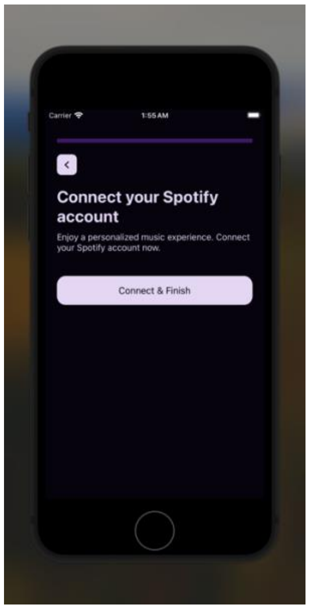
Choose your preference for connections.

I am interested in Women

I am interested in Men

I am interested in Everyone

Continue

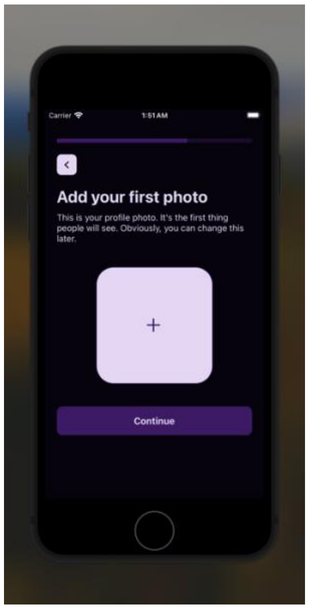

Carrier 1:55 AM

Connect your Spotify account

Enjoy a personalized music experience. Connect your Spotify account now.

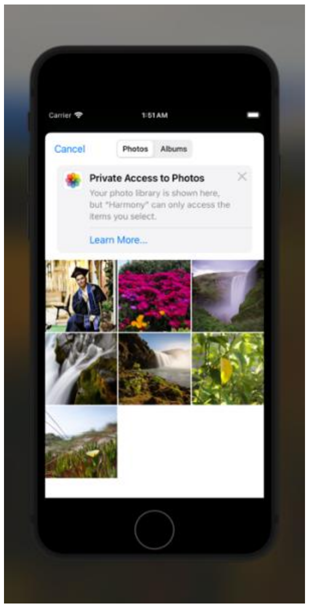
Connect & Finish

Appendix B: Profile Photo Selection



Add your first photo

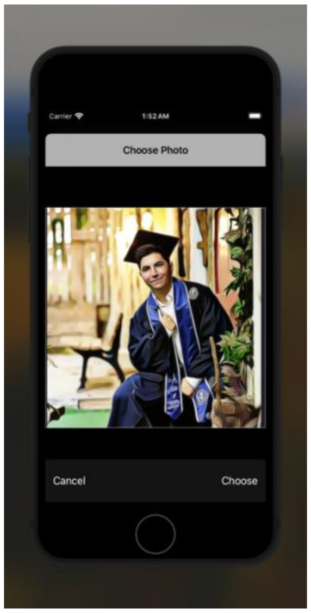
This is your profile photo. It's the first thing people will see. Obviously, you can change this later.

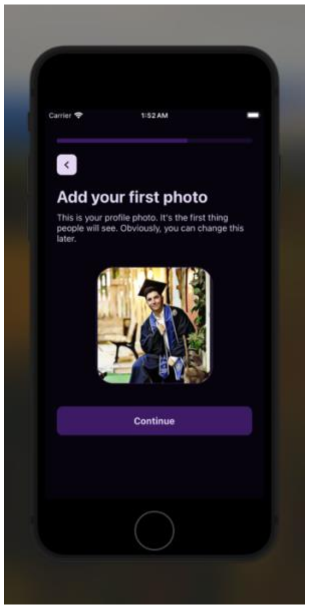


Private Access to Photos

Your photo library is shown here, but "Harmony" can only access the items you select.

Learn More...

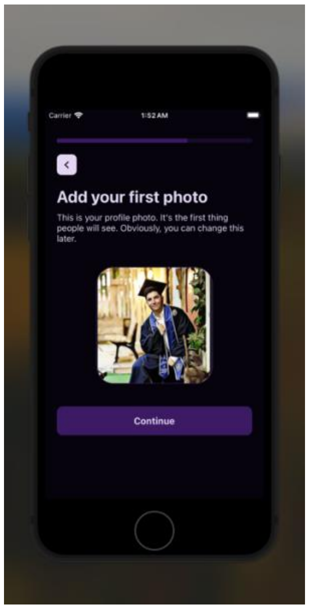




Add your first photo

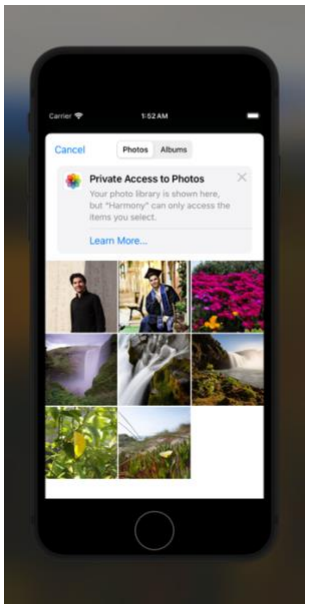
This is your profile photo. It's the first thing people will see. Obviously, you can change this later.

Appendix C: Change Selected Profile Photo



Add your first photo

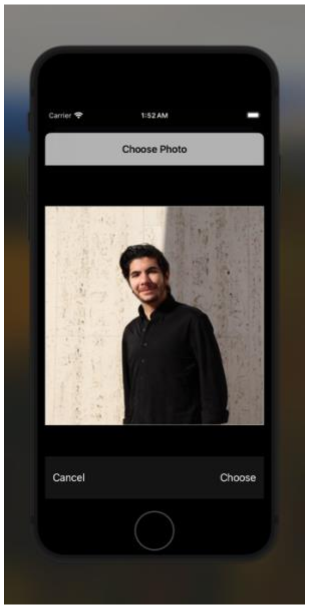
This is your profile photo. It's the first thing people will see. Obviously, you can change this later.

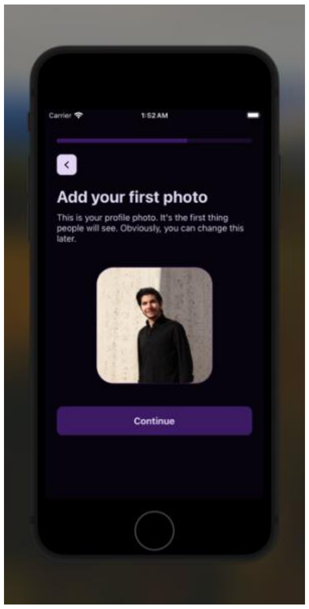


Private Access to Photos

Your photo library is shown here, but "Harmony" can only access the items you select.

Learn More...

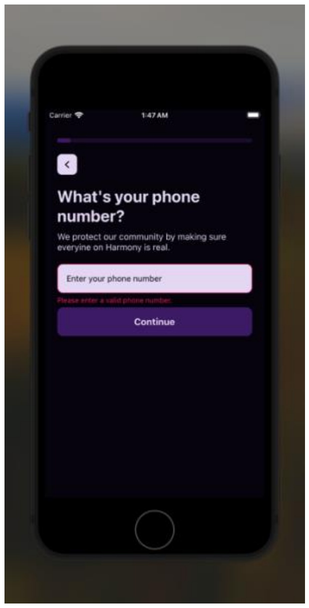




Add your first photo

This is your profile photo. It's the first thing people will see. Obviously, you can change this later.

Appendix D: Error Handling


Carrier 1:47 AM

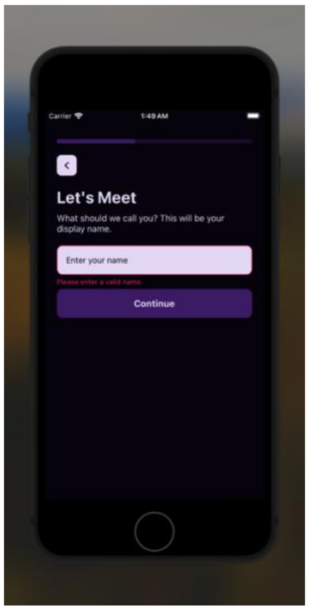
What's your phone number?

We protect our community by making sure everyine on Harmony is real.

Enter your phone number

Please enter a valid phone number.

Continue


Carrier 1:49 AM

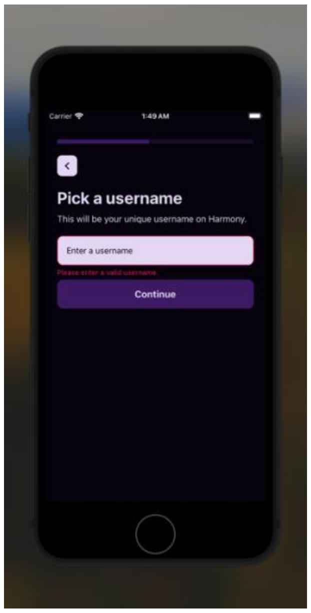
Let's Meet

What should we call you? This will be your display name.

Enter your name

Please enter a valid name.

Continue


Carrier 1:49 AM

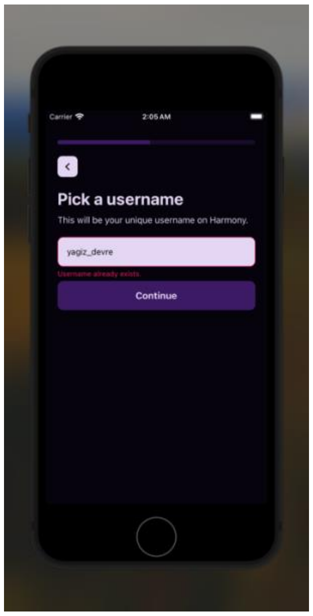
Pick a username

This will be your unique username on Harmony.

Enter a username

Please enter a valid username.

Continue


Carrier 2:05 AM

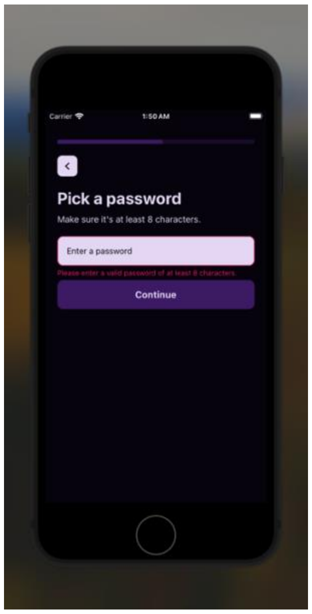
Pick a username

This will be your unique username on Harmony.

yagiz_devre

Username already exists.

Continue


Carrier 1:50 AM

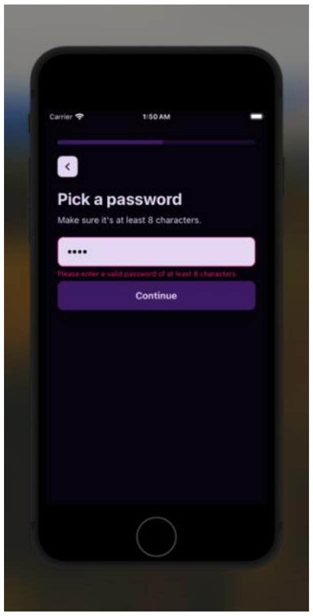
Pick a password

Make sure it's at least 8 characters.

Enter a password

Please enter a valid password of at least 8 characters.

Continue


Carrier 1:50 AM

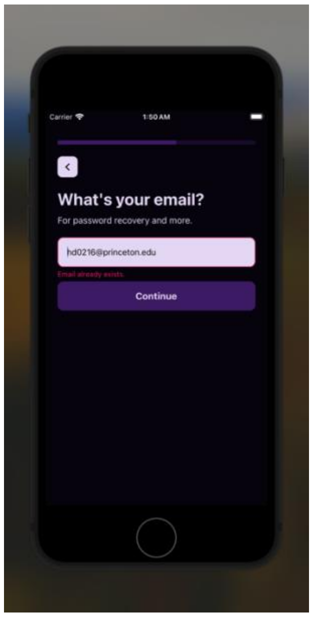
Pick a password

Make sure it's at least 8 characters.

••••

Please enter a valid password of at least 8 characters.

Continue


Carrier 1:50 AM

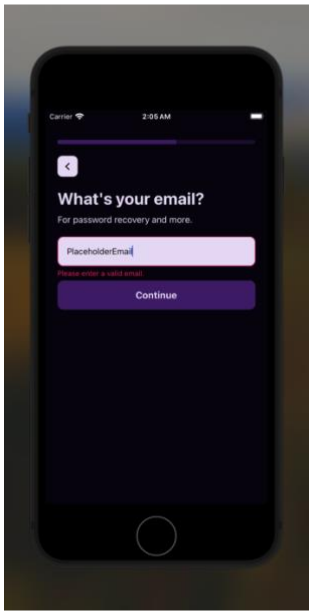
What's your email?

For password recovery and more.

hd0216@princeton.edu

Email already exists.

Continue


Carrier 2:05 AM

What's your email?

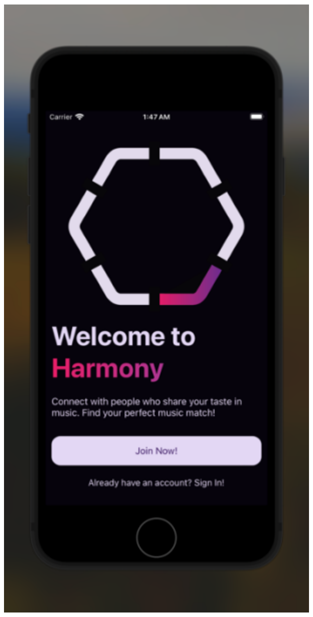
For password recovery and more.

PlaceholderEmail

Please enter a valid email.

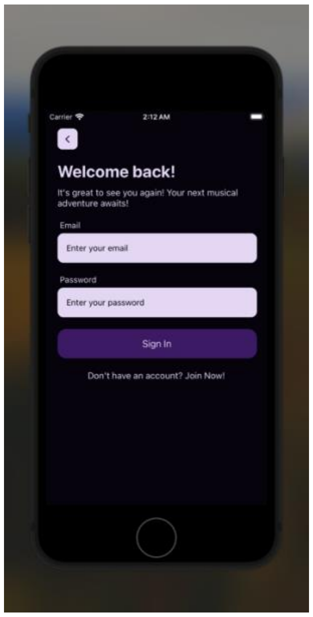
Continue

Appendix E: Sign In Flow



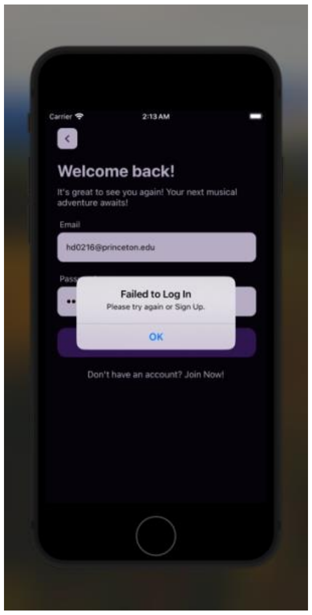
Welcome to Harmony

Connect with people who share your taste in music. Find your perfect music match!



Welcome back!

It's great to see you again! Your next musical adventure awaits!



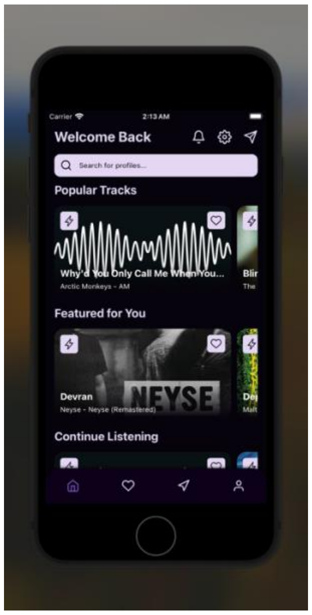
Welcome back!

It's great to see you again! Your next musical adventure awaits!

Email: hd0216@princeton.edu

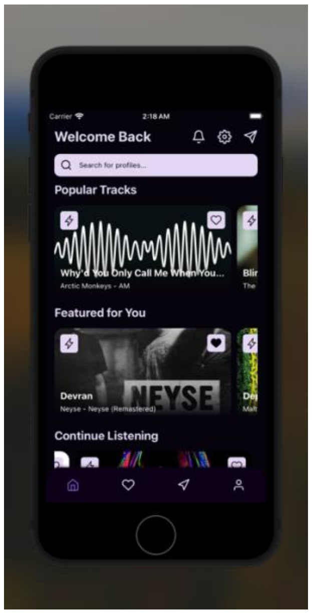
Failed to Log In

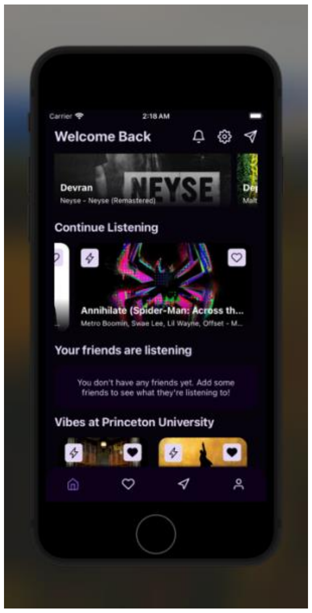
Please try again or Sign Up.



** The third image shows a user trying to login with incorrect credentials which results in an error message, forth image shows a user login with correct credentials*

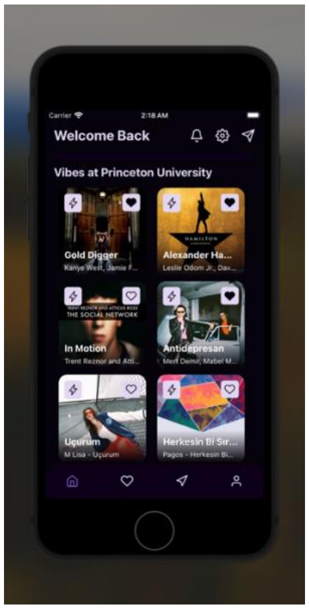
Appendix F: Home Screen



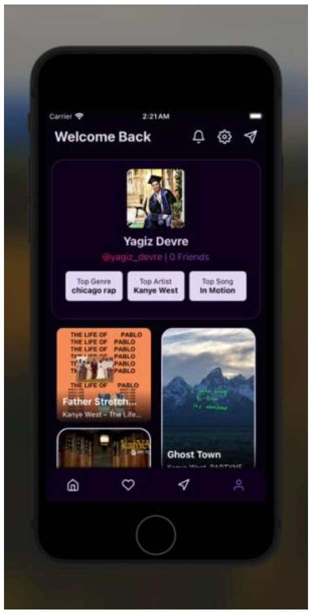


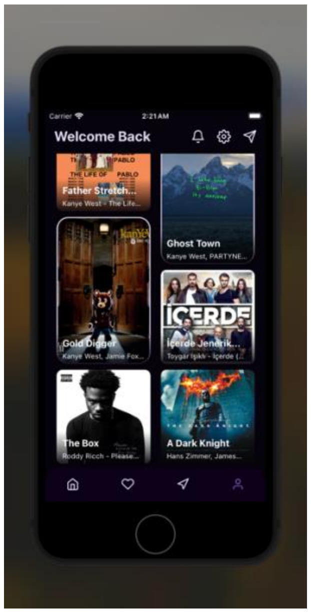
You don't have any friends yet. Add some friends to see what they're listening to!

Carrier
2:18 AM
Welcome Back
Devran
Neyse - Neyse (Remastered)
Continue Listening
Annihilate (Spider-Man: Across th...
Metro Boomin, Swae Lee, Lil Wayne, Offset - M...
Your friends are listening
Vibes at Princeton University

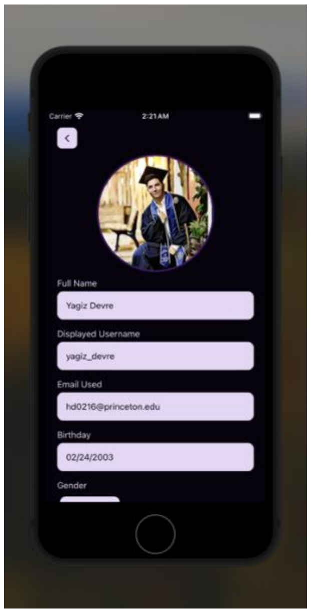


Appendix G: Profile Screen





Appendix H: Setting View


Carrier 2:21 AM

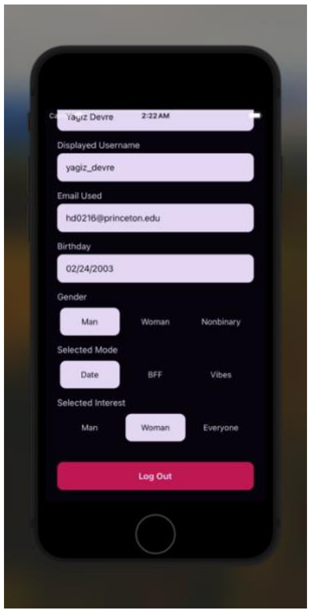
Full Name
Yagiz Devre

Displayed Username
yagiz_devre

Email Used
hd0216@princeton.edu

Birthday
02/24/2003

Gender


Yagiz Devre 2:22 AM

Displayed Username
yagiz_devre

Email Used
hd0216@princeton.edu

Birthday
02/24/2003

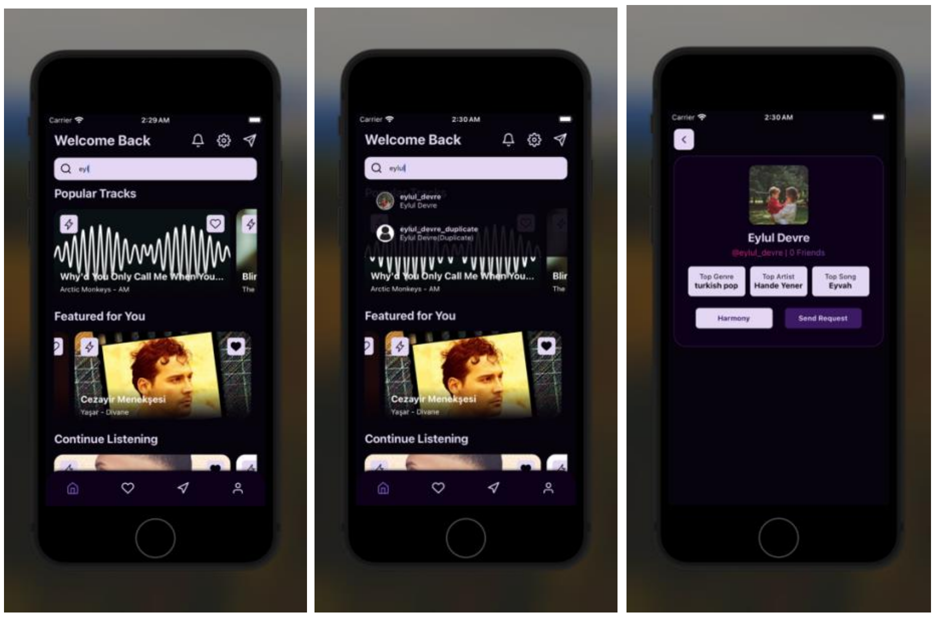
Gender
[✓ Man] Woman Nonbinary

Selected Mode
[✓ Date] BFF Vibes

Selected Interest
Man [✓ Woman] Everyone

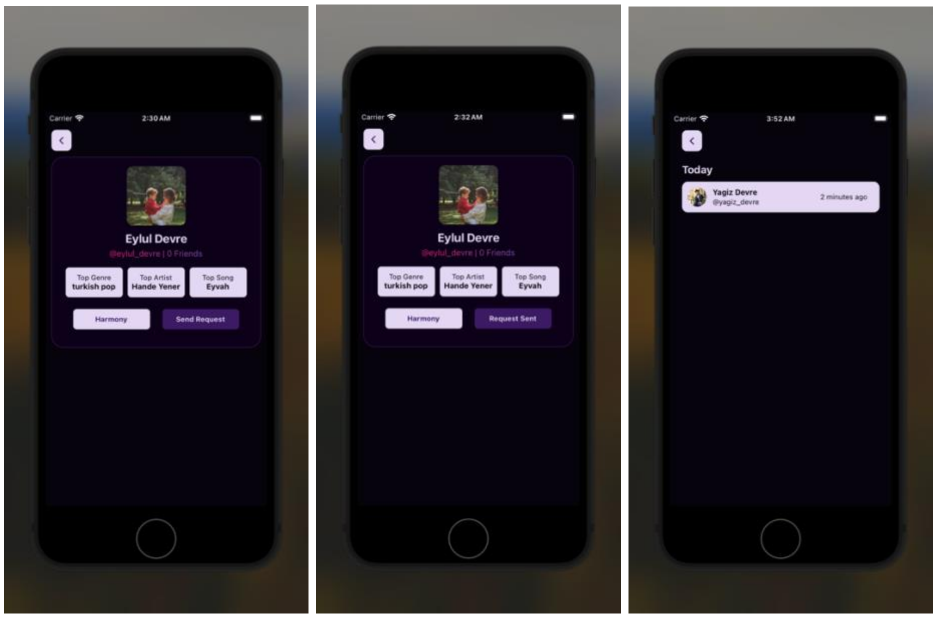
** The user can change their preferences and information as they want in this screen. The only information that cannot be changed is the username. Likewise, users can log out of the app through this screen.*

Appendix I: Profile Search



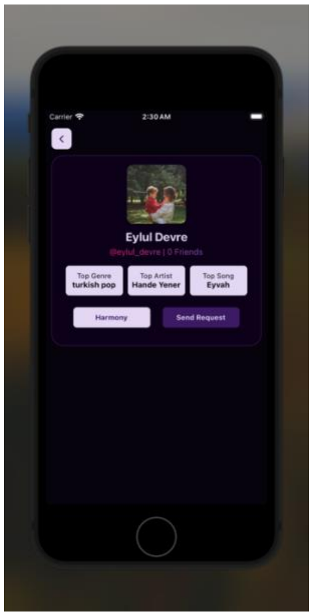
** The search is done in real-time by each keystroke and updates the search results in real-time. The search also starts checking for users after 4 or more characters are inputted to the search bar for efficiency and reduce the number of database calls since each username is already at least 4 characters long*

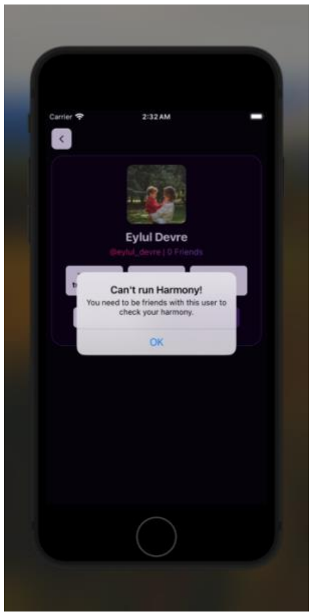
Appendix J: Friend Requests



* Users can send friend requests either through match screen, or directly by searching profiles. As seen in the second picture, when a request is sent, the text is updated to "Request Sent" and when we log into the other account(eylul_devre's account) and go to the notifications screen, we see the request that we sent 2 minutes ago from the user yagiz_devre.

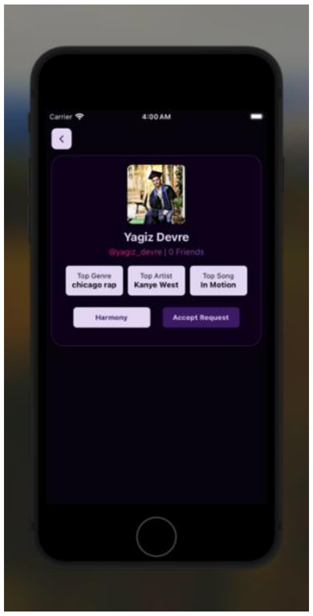
Appendix K: Harmony

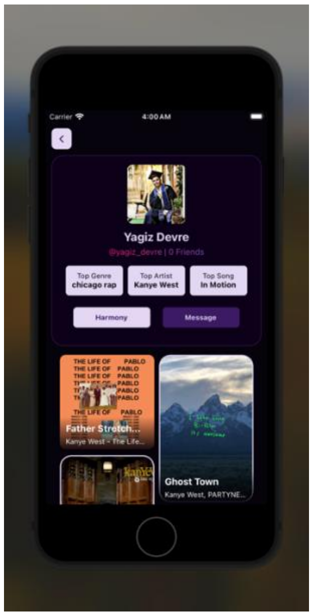


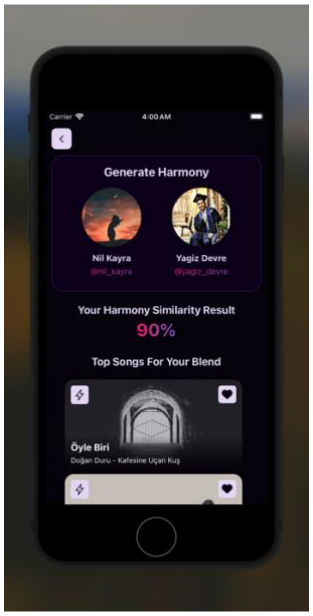


Can't run Harmony!
You need to be friends with this user to check your harmony.

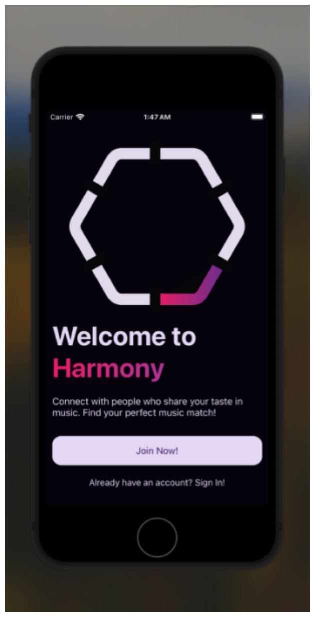
* If two users are not friends, they can not generate a Harmony Playlist





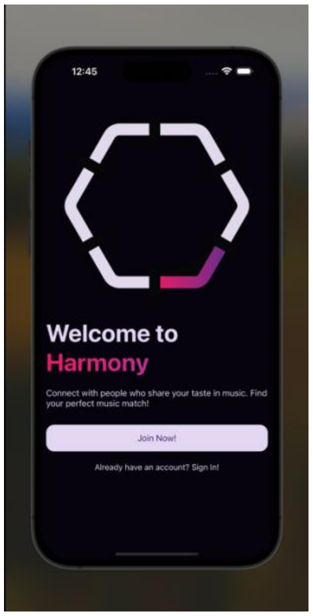


__Appendix L: Responsiveness__



Welcome to Harmony

Connect with people who share your taste in music. Find your perfect music match!



Welcome to Harmony

Connect with people who share your taste in music. Find your perfect music match!

* Harmony works and functions perfectly in larger screens as well such as iPhone 15 Pro Max. It also works in Android devices as well.