KIITE CAFE: A WEB SERVICE FOR GETTING TOGETHER VIRTUALLY TO LISTEN TO MUSIC

Kosetsu Tsukuda **Keisuke Ishida** **Masahiro Hamasaki** **Masataka Goto**
National Institute of Advanced Industrial Science and Technology (AIST), Japan
{k.tsukuda, ksuke-ishida, masahiro.hamasaki, m.goto}@aist.go.jp

ABSTRACT

In light of the COVID-19 pandemic making it difficult for people to get together in person, this paper describes a public web service called *Kiite Cafe* that lets users get together virtually to listen to music. When users listen to music on Kiite Cafe, their experiences are characterized by two architectures: (i) visualization of each user's reactions, and (ii) selection of songs from users' favorite songs. These architectures enable users to feel social connection with others and the joy of introducing others to their favorite songs as if they were together in person to listen to music. In addition, the architectures provide three user experiences: (1) motivation to react to played songs, (2) the opportunity to listen to a diverse range of songs, and (3) the opportunity to contribute as curators. By analyzing the behavior logs of 1,760 Kiite Cafe users over about five months, we quantitatively show that these user experiences can generate various effects (e.g., users react to a more diverse range of songs on Kiite Cafe than when listening alone). We also discuss how our proposed architectures can continue to enrich music listening experiences with others even after the pandemic's resolution.

1. INTRODUCTION

Unlike listening to music alone, listening to music with others adds the qualities of feeling social connection and letting others listen to one's favorite songs. For example, the former quality occurs when attending a live concert and sharing the experience with other audience members [1, 2], while the latter quality occurs when people introduce others to their favorite songs [3–5].

These qualities have become hard to enjoy since the COVID-19 pandemic has made it difficult to get together in person and listen to music with others. Instead of attending a live concert, people can listen to the same music at the same time via TV, radio, or live streaming on the web. However, such media provide a poor alternative because the former quality of social connection requires audiences to get together in the same place so that they can see each other's reactions to the music. Similarly, instead of directly introducing others to favorite songs, people can post URL links to them (e.g., YouTube videos of songs) to social networking services (SNSs) such as Twitter and Facebook. Even if many SNS users react to a song post (e.g., with a "thumbs up"), there is no guarantee that they actually listened to the song and liked it. Rather, the latter quality of sharing a favorite song with others requires knowing that people who react actually listened to the song.

In light of the above, we propose a web service called *Kiite Cafe* [1] [2] that enables people to get together virtually to listen to music without losing the above qualities. Kiite Cafe is characterized by the following two architectures: (i) when users listen to songs on Kiite Cafe, each user's reactions are visualized, and (ii) songs played on Kiite Cafe are selected from users' favorite songs. To facilitate an intuitive understanding of the user experiences provided by these architectures, we give the following example.

Suppose that Emily is a Kiite Cafe user. One day, she logs in to Kiite Cafe and finds that 14 users are logged in. Each user is identified by his/her own icon. The users, including Emily, can simultaneously listen to the same song, which is automatically selected and played. Even if the played song has a different mood from songs that Emily usually listens to, if she likes it, she can add it to her list of favorite songs (i.e., her *favorites list*). Because she has encountered a new favorite song, she feels happy to listen to a diverse range of songs. Moreover, when the currently played song is added to her favorites list, architecture (i) visualizes her reaction by displaying a heart symbol on her icon. Because other users' reactions are also visualized, she can see their reactions to feel social connection. For example, one of Emily's favorite songs is played when it is automatically selected by architecture (ii). While her favorite song is playing, she is pleased to notice that a heart symbol is displayed on another user's icon. Then, other users also react to the song, and eventually the heart symbol is displayed on eight users' icons. Architecture (i) thus enables Emily to see the moments when other users start liking one of her favorite songs. This experience makes her feel happy and want other users to listen to another of her favorite songs. Thus, Emily looks forward to another favorite song being played; until then, she stays on Kiite Cafe and enjoys other users' favorite songs.

Our contributions can be summarized as follows.

- We propose two architectures for enabling people to simultaneously listen to the same music online while achieving the qualities of social connection and the joy of introducing other people to favorite songs.

[1] "Kiite" means "Listen" in Japanese.
[2] https://cafe.kiite.jp

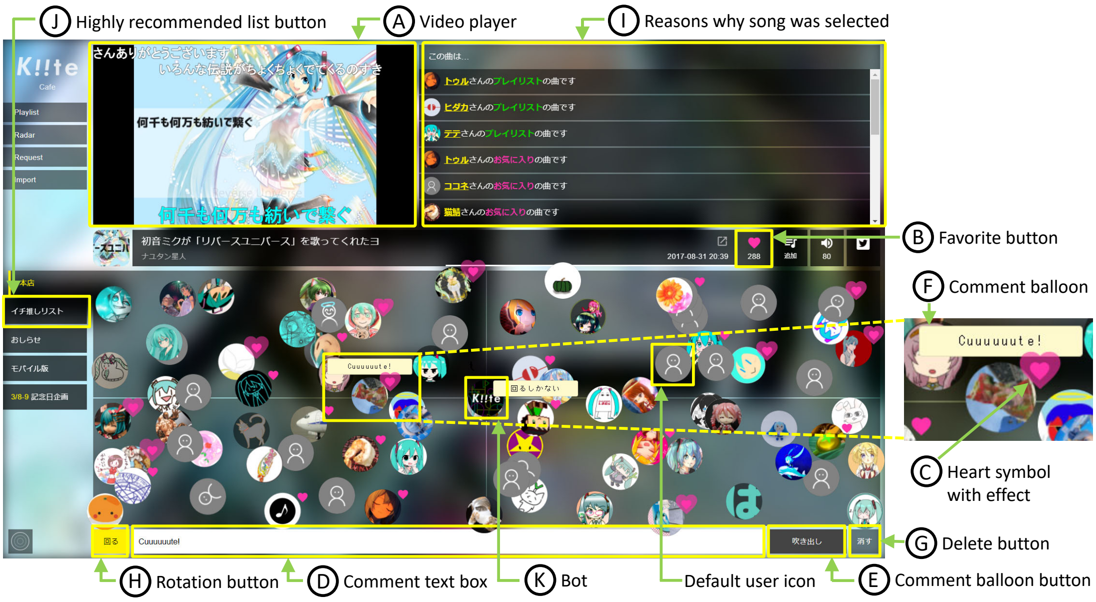

Figure 1. Screenshot of Kiite Cafe.

- We implemented and released a web service, called Kiite Cafe, that applies these architectures.
- We describe three user experiences in which users (1) are motivated to react to songs, (2) can listen to a diverse range of songs, and (3) can contribute as curators; we also discuss the effects of these experiences on users as a result of the two proposed architectures.
- By analyzing logs of user behavior on Kiite Cafe, we show that the architectures do provide the above effects. Specifically, users (1) react to songs more actively as the number of users on Kiite Cafe increases, (2) react to a more diverse range of songs on Kiite Cafe than when they listen to songs alone, and (3) stay on Kiite Cafe longer when they contribute more as curators.

2. OVERVIEW OF KIITE CAFE

Kiite Cafe is implemented as a novel function on Kiite, which is an existing web service for exploring and discovering music. Any Kiite user can use Kiite Cafe. Below, we introduce Kiite's functions related to Kiite Cafe and then give an overview of Kiite Cafe.

2.1 Kiite

Song data on Kiite are routinely collected from *Nico Nico Douga*, which is one of the most popular video sharing services in Japan. On Nico Nico Douga, it is quite popular for both amateur and professional musicians to upload songs created with singing voice synthesizer software called *VOCALOID* [6]. As of the end of Mar. 2021, more than 220,000 songs can be played back on Kiite. When a Kiite user listens to a song, its video clip is played on Kiite by an embedded video player [3] .

Kiite enables users to effectively find favorite songs by providing novel functions such as exploration of songs based on their emotions and continuous listening to only the choruses of multiple songs. A registered user can set her own icon image, add songs to her favorites list, create playlists, listen to other users' playlists, and so on.

[3] On Nico Nico Douga, all songs are uploaded as music videos.

2.2 Kiite Cafe

Fig. 1 shows an overview of Kiite Cafe. When a user logs in, her icon is displayed at a random position in a two-dimensional space that also displays other logged-in users. All of the users listen to the same song played in a video player (Ⓐ in the figure) at the same time, like a live concert. As mentioned in section 1, Kiite Cafe has two architectures, for visualizing users' reactions and selecting songs to play from users' favorite songs. In the rest of this section, we describe the details of each architecture.

2.2.1 *Architecture (i): User Reaction Visualization*

We visualize the following four kinds of reactions so that users can see each other's reactions to a played song.

Favorite. When a user likes a played song, she can add it to her favorites list by clicking the "favorite" button Ⓑ. When the button is clicked, a heart symbol with an animation effect is displayed at the top right of the user's icon while the song is playing (Ⓒ). This enables users to quickly see how many users like a song. When the user had already added the played song to her favorites list, the heart symbol is displayed without the effect.

Comment. When a comment is entered in a text box Ⓓ and the "comment balloon" button Ⓔ is clicked, a comment balloon is displayed above the user's icon for 90 seconds (Ⓕ). The user can also manually delete her comment by clicking the "delete" button Ⓖ. Users can thus use this function to express their impressions of a played song or have simple communication with each other.

Rotation. A user can rotate her icon by clicking the "rotation" button Ⓗ. The icon then rotates clockwise at a uniform rate until the played song ends. The user can also manually stop the rotation by clicking the "rotation" button again. Users can use this function to express feelings like a sense of excitement. However, note that Kiite Cafe does not provide any guidance on when users should use this function, because we want them to use it as they please.

Move. By clicking an arbitrary position in the two-dimensional space, a user can move her icon to the clicked position. The icon is animated to move to the position in a straight line at a uniform rate. Kiite Cafe does not display any meaning for the quadrants and axes in the two-dimensional space. Instead, as with the *Rotation* function, we leave the usage of the *Move* function to users.

2.2.2 Architecture (ii): Song Selection from Users' Favorite Songs

Let U denote a set of users who are logged in to Kiite Cafe. For each user u, we define S_u as the set of songs included in u's favorites list or playlists. A played song is selected from $\bigcup_{u \in U} S_u$. The automatic song selection process is invoked before the end of the currently played song, and it consists of the following two steps: (1) selection of a user and (2) selection of a song from the user's favorite songs.

In the first step, if there are biases toward certain selected users, then the selected songs may also be biased. Moreover, some users may become frustrated if their favorite songs are not selected at all. To avoid such biases and satisfy every user, we developed an algorithm that can randomly but fairly select users and thus diversify the played songs. Suppose that user u is selected in the first step, such that every user has an equal chance to be selected. When song $s \in S_u$ is randomly selected in the second step, the reason for its selection is displayed, e.g., "This song is in u's playlist" (first row of Ⓘ in Fig. 1). If s is also among other users' favorite songs, that information is displayed (second and later rows of Ⓘ) so that those users can notice that one of their favorite songs is being played. Moreover, a user can set one of her playlists as a "highly recommended list" by clicking a button Ⓙ. When the selected user sets a list as "highly recommended," a song in that list is randomly but preferentially selected in the second step. By setting such a list, a user can specify the songs that she wants other users to listen to.

Note that the implemented selection algorithm described above for our service is just an example, and other algorithms can be used as long as they balance the fairness and randomness of selecting both users and songs.

In addition, we created a bot account Ⓚ that is always logged in. The bot periodically creates playlists according to a daily/weekly popularity ranking of VOCALOID songs on Nico Nico Douga. The bot is treated as one of the users, and songs in its playlists can also be selected by the song selection process. This gives a user a chance to listen to the latest popular songs and find new favorite songs even when no other human users are logged in. Note that the bot does not show any reactions to played songs.

3. USER EXPERIENCES AND EFFECTS

As mentioned in section 1, the proposed architectures add two qualities: social connection and the joy of introducing others to favorite songs. In addition, the Kiite Cafe architectures provide three kinds of user experiences. This section describes those experiences and their effects on users.

3.1 Motivation to React to Songs

Although many studies have been conducted on enabling users to listen to music together, most of them have focused on visualizing the song selection process or proposing methods for that process [7–11]. A system that can show a summary of listeners' feedback on a song (total numbers of likes and dislikes) has been proposed [12]; however, little attention has been paid to visualizing each user's reactions. In contrast, Kiite Cafe visualizes users' reactions via their icons, as in section 2.2.1. By sharing all the users' reactions with each other, Kiite Cafe motivates them to react to the currently played song. Accordingly, we expect that, the more people get together on Kiite Cafe, the more meaningful it will be to show their reactions, and the more actively they will react to songs. In the long term, this would enable users to develop the habit of actively listening to music and enrich their listening experiences [13].

3.2 Diversification of Song Listening

Many studies have sought to play songs that match the musical preferences of as many users as possible [7, 9, 14]. In the short term, this approach may be able to increase users' satisfaction. In the long term, however, as is known from the negative effects of a filter bubble [15, 16], this approach could narrow users' musical interests. On the other hand, because Kiite Cafe plays songs selected from various users' favorite songs, it may not be able to always match most users' musical preferences. However, listening to a more diverse range of songs enables users to find not only songs that match their preferences well but also unexpected or serendipitous songs [17] that do not match their usual preferences. In other words, we expect that a user will react to a more diverse range of songs on Kiite Cafe than when she listens alone. In the long term, this experience would expand the user's horizons.

3.3 Contribution as Curators

According to architecture (ii), suppose that a song in user u's playlist is selected and played on Kiite Cafe. Because of architecture (i), u can see the moment when other users start liking or show interest in that song (e.g., u can see when other users add the song to their favorites list or rotate their icon). In substance, for other users, u plays a role as a curator. That is, the two architectures enable every user to naturally contribute as a curator. We expect that when a user experiences the joy of contributing as a curator, she will look forward to the curation opportunity when another of her favorite songs is played and thus increase her dwell time on Kiite Cafe. It has been reported that acting as a curator increases music listening activity (e.g., listening to more songs and making playlists for curation) [18]. Therefore, in the long term, this experience would promote users' daily music listening activity.

4. EXPERIMENT

We launched the Kiite Cafe service on Aug. 5, 2020. In this section, we evaluate the three expected effects discussed in the previous section. To this end, we analyzed user behavior logs for the period between Aug. 5, 2020 and Jan. 14, 2021. The number of unique users who logged in during this period was 1,760. The *Favorite*, *Comment*, *Rotation*, and *Move* reactions were used 29,127, 9,826, 59,983, and 45,353 times, respectively.

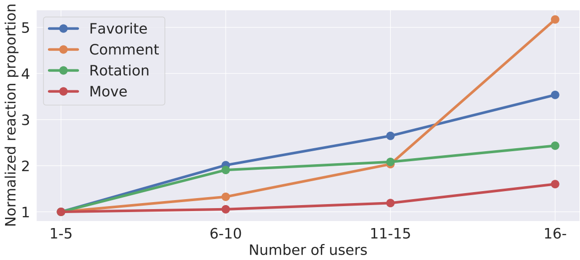

Figure 2. Relation between the number of users on Kiite Cafe and the normalized reaction proportion.

4.1 Frequency of User Reactions

As a result of users sharing their reactions with each other on Kiite Cafe, we expect that they will be more motivated to react as the number of users increases. To verify this effect, we evaluated the following research question: *Does a user react to a played song more frequently as the number of users on Kiite Cafe increases? (**RQ1**)*

Settings. We considered the four kinds of reactions: $R = \{Favorite, Comment, Rotation, Move\}$. First, for each played song, we obtained U_s, the set of users except the bot who were on Kiite Cafe when song s started playing. According to the number of users (i.e., $|U_s|$), we categorized songs into four classes (C_1: $1 \leq |U_s| \leq 5$; C_2: $6 \leq |U_s| \leq 10$; C_3: $11 \leq |U_s| \leq 15$; C_4: $16 \leq |U_s|$). To answer **RQ1**, for each reaction, we compared the average proportion of users who reacted to a song among the classes. More formally, let S_{C_i} denote a list of songs in C_i $(1 \leq i \leq 4)$[4]. Given song $s \in S_{C_i}$ and reaction $r \in R$, let U_s^r denote the set of users who gave r as a reaction to s. Then, the proportion of such users is given by $ratio(s, r) = \frac{|U_s^r|}{|U_s|}$. Finally, the average proportion over S_{C_i} was computed as follow.

$$avgratio(S_{C_i}, r) = \frac{1}{|S_{C_i}|} \sum_{s \in S_{C_i}} ratio(s, r).$$

Results. Fig. 2 shows the results. For visibility, $avgratio(S_{C_i}, r)$ was normalized by $avgratio(S_{C_1}, r)$ for each reaction. For all the reactions, the reaction proportion monotonically increased as the number of users increased; thus, the answer to **RQ1** is "Yes." Because the *Favorite* function was obviously used to add a song to a user's favorites list, we discuss how the users used the other three functions. Regarding the *Rotation* function, although Kiite Cafe does not explain its purpose, we searched Kiite Cafe users' tweets on Twitter[5] and found that a number of users used it to express their feelings of excitement. Next, by analyzing tweets about the *Move* function, we found that it was used mainly for two purposes. First, users moved their icons as if they were dancing. Second, users regarded the top left of the two-dimensional space (i.e., near the video player) as the front row at a live concert venue and moved there when their favorite songs were played. It is interesting that such a culture was created by the users and spread among them. Finally, regarding the *Comment* function,

[4] Because the same song can be played multiple times on Kiite Cafe, the same song can appear multiple times in S_{C_i}.

[5] We assumed that Twitter users who tweeted about the function were Kiite Cafe users.

Reaction r	$\lvert U^r \rvert$	$avgdiv(S_u^{org})$	$avgdiv(S_u^r)$	p-value
Favorite	130	10.493	10.960	1.99×10^{-6}
Comment	56	10.384	10.920	4.40×10^{-3}
Rotation	118	10.502	10.918	5.80×10^{-6}
Move	110	10.559	11.050	8.21×10^{-9}

Table 1. Diversity of musical preferences.

although the average length of the played songs was 237 seconds, 10.1% of comments were posted within the first 15 seconds of a song. In such comments, users often expressed the joy of having their favorite songs played (e.g., "Come ooooon!" and "Yeeeees!"). This was similar to the phenomenon at live concerts in which the audience gets excited when a favorite song starts. In summary, as the number of users increased, they were more likely to express their excitement and behave as if they were attending a live concert.

4.2 Diversity of Reacted Songs

Because Kiite Cafe enables users to listen to songs that do not always match their musical preferences, we expect that they will react to a more diverse range of songs. To verify this effect, we evaluated the following research question: *Does a user react to a more diverse range of songs on Kiite Cafe as compared to her musical preferences before she started using the service? (**RQ2**)*

Settings. Let t_u denote the time when user u initially logged in to Kiite Cafe. We assumed that songs added to u's favorites list before t_u (i.e., before using Kiite Cafe), denoted by S_u^{org}, represented u's original musical preferences; these were collected on the original Kiite service, which was launched on Aug. 30, 2019, and described in section 2.1. Moreover, we assumed that songs for which u gave reaction r, denoted by S_u^r, represented u's musical preferences in terms of r after starting to use Kiite Cafe. To answer **RQ2**, we compared the diversity of S_u^r with that of S_u^{org} for each reaction. Formally, the diversity was computed as the intra-list diversity [19]. In the case of S_u^{org},

$$div(S_u^{org}) = \frac{\sum_{s_i \in S_u^{org}} \sum_{s_j \in S_u^{org} \setminus \{s_i\}} dist(s_i, s_j)}{|S_u^{org}|(|S_u^{org}| - 1)},$$

where $dist(s_i, s_j)$ is the Euclidean distance between the feature vectors of s_i and s_j. For the diversity, we obtained a song's feature vector by using OpenL3 [20]. For each reaction r, we considered only users who had more than nine songs in both S_u^{org} and S_u^r so that we could appropriately measure the users' musical preferences[6]. Let U^r denote the set of such users. Then, given r, the average diversity of S_u^{org} was computed as

$$avgdiv(S_u^{org}) = \frac{1}{|U^r|} \sum_{u \in U^r} div(S_u^{org}).$$

Similarly, $avgdiv(S_u^r) = \frac{1}{|U^r|} \sum_{u \in U^r} div(S_u^r)$.

Results. Table 1 lists the results. We can say that for all reactions, the diversity of songs producing reactions statistically increased in comparison to the diversity of favorite songs before starting to use Kiite Cafe; thus, the answer to **RQ2** is "Yes." These results indicate that Kiite Cafe is also

[6] Because we released a beta version of Kiite Cafe on May 1, 2020, users who logged in to Kiite Cafe for the first time between May 1, 2020 and Aug. 4, 2020 were not included in this analysis.

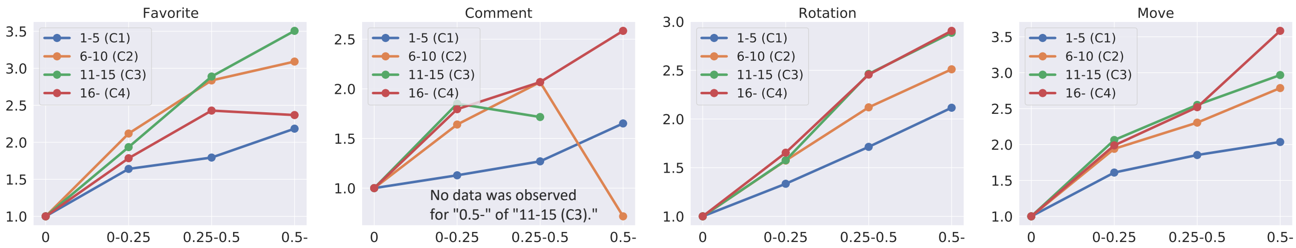

Figure 3. Relation between the proportion of users who gave reactions (x-axis) and the normalized dwell time (y-axis).

useful as a service for users to find songs that are different from their daily musical preferences.

4.3 Dwell Time

Because Kiite Cafe enables users to experience the joy of contributing as curators, we expect that they will stay longer as their contributions increase. To verify this, we evaluated the following research question: *Does a user stay on Kiite Cafe for a longer time as the proportion of users who react to her favorite songs increases? (**RQ3**)*

Settings. We define user u's session on Kiite Cafe as the duration between u's login and logout times. In u's kth session, suppose that three of u's favorite songs were played, and that 0%, 40%, and 16% of users gave reaction r to those songs. Following the assumption that the maximum percentage (in this case, 40%) influenced u's dwell time, we categorized the maximum value of $\mathit{ratio}(s,r)$ (defined in section 4.1) into four classes (G_1: $\mathit{ratio}(s,r)=0$; G_2: $0<\mathit{ratio}(s,r)\leq 0.25$; G_3: $0.25<\mathit{ratio}(s,r)\leq 0.5$; G_4: $0.5<\mathit{ratio}(s,r)$). However, for a proportion of 0.4, eight reacting users among 20 users would have a higher impact on u than two reacting users among five users. Therefore, we also considered the classes of $|U_s|$ as in section 4.1. That is, to answer **RQ3**, given a class of the number of users, we compared the average session lengths between the reaction proportion classes for each reaction. Formally, let D_u denote a list of u's sessions. The kth session $T_{u,k}\in D_u$ represents a list of songs from u's favorite songs (S_u) that were played in the session. For that session, we selected the song $s^{max}_{u,k}\in T_{u,k}$ that had the highest proportion of users who gave reaction r (i.e., $s^{max}_{u,k}=\arg\max_{s\in T_{u,k}}\mathit{ratio}(s,r)$). Given C_i and G_j, we define the set of $s^{max}_{u,k}$ belonging to C_i and G_j in all users' sessions as $S_{i,j}=\{s^{max}_{u,k}|u\in U\wedge 1\leq k\leq |D_u|\wedge s^{max}_{u,k}\in T_{u,k}\wedge s^{max}_{u,k}\in C_i\wedge s^{max}_{u,k}\in G_j\}$. Let $\mathit{len}(u,k)$ denote the length in seconds of u's kth session. Then, the average session length was computed as

$$\mathit{avglen}(S_{i,j})=\frac{1}{|S_{i,j}|}\sum_{s^{max}_{u,k}\in S_{i,j}}\mathit{len}(u,k).$$

Results. Fig. 3 shows the results; $\mathit{avglen}(S_{i,j})$ was normalized by $\mathit{avglen}(S_{i,1})$ for each reaction for visibility. For the *Favorite*, *Rotation*, and *Move* functions, we can see that the dwell time tended to increase as the proportion of users who gave that reaction increased. In these graphs, the line for the class of 1-5 users is located at the lowest position among the four classes (C_1 - C_4). Especially for *Rotation* and *Move*, at each reaction proportion, the normalized dwell time tended to increase with the number of users. These results indicate that not only the proportion of users who gave a reaction but also the absolute number of such users influenced the dwell time. On the other hand, no clear tendency was observed for the *Comment* function when the number of users was 6-10 or 11-15. Still, it is possible that *Comment* also had a positive effect on the dwell time, because it monotonically increased when the number of users was 1-5 or at least 16. Detailed analysis with more user behavior logs will be required to verify this effect, and we leave that for a future work. In summary, the answer to **RQ3** is "Yes" for *Favorite*, *Rotation*, and *Move*.

5. DISCUSSION

In section 3, we described the user experiences provided by Kiite Cafe and their effects. We believe that Kiite Cafe has even more potential to diversify and enrich users' music listening experiences. In this section, to demonstrate that potential, we discuss three themes.

5.1 Application Examples for Online Events

Kiite Cafe has been used for several online events including VOCALOID-related events. At an event on Aug. 29, 2020, for example, a famous creator of VOCALOID songs made a special playlist that consisted of songs that the creator liked or had created. During the one-hour event, as many as 140 Kiite Cafe users enjoyed simultaneously listening to the songs in the playlist, and used the reaction functions of Kiite Cafe to communicate with the creator in real time. For another event on Feb. 11, 2021, a questionnaire was conducted on favorite songs related to winter or snow in the VOCALOID event. During the 90-minute event, 77 users enjoyed listening to songs in a playlist created according to the questionnaire answers.

Although it has become difficult for people to get together in person and communicate with each other and with artists because of the COVID-19 pandemic, we demonstrated a new style of online music events through these examples. Moreover, even after the pandemic's resolution, we believe that this kind of online event will be valuable for users who cannot easily attend physical events for reasons such as geographic remoteness.

5.2 Additional Service Functions

Although all users on Kiite Cafe get together in one online space, it would be interesting to provide additional spaces for different purposes in the future (we could call the main space and additional spaces the "main cafe" and "branches," respectively). For example, for a branch on the theme of "time," we could put a higher priority on songs related to time (e.g., playing night-related songs at night) by analyzing song lyrics if they are available in the song selection process. We could also consider a function that allows any user to conduct a questionnaire by displaying possible responses in each quadrant of the two-dimensional space. For example, a user might ask "Who would you like to listen to the played song with?" and assign responses of "family," "lover," "friend," and "other" to the quadrants. Other users could answer this question by moving their icons. This function would provide a good opportunity to see how other users perceive a song.

5.3 Reusable Insights

The reusable insights can be summarized as follows.

- Through the experiments, we verified that the two architectures are effective in promoting users' music listening activity. These architectures can be helpful for other researchers and companies to develop interfaces that enable users to listen to music together.
- The examples of successful online events showed that the architectures can offer new ways to enjoy music with other people even during the COVID-19 pandemic. We thus opened up a new research theme to support interactions among creators, audiences, and music.
- We clarified the value of visualizing the moments when users start liking a song. In contrast to traditional curation on an SNS, the approach of Kiite Cafe guarantees that users who liked a user's favorite song did listen to it. This insight could also be beneficial in designing other music listening systems or services.

6. RELATED WORK

6.1 Music Listening Systems for Group of Users

Music listening systems for a single user were reviewed by Goto and Dannenberg [21] and Knees *et al.* [22]. In contrast, systems for a group of users can be classified into two types. The first type aims to enable users to listen to music at the same time. Most studies on this type assume that users get together in person at a public space such as a fitness center [7], a party [10], a bar [9], or a room [8]. In MusicFX [7] and Flytrap [8], the system reads users' musical preferences from each user's device, and songs stored in the system are played by taking those preferences into account, while in Jukola [9], PartyVote [10], and WePlay [12], users nominate songs to be played, like a jukebox. In the second type of group listening system, users share songs with other users. Sharing music with others is an important activity to expand listeners' horizons [4]. Studies on this type do not assume that users listen to a song at the same time. Push!Music [4] and tunA [3] are mobile music players that let users share songs via Wi-Fi with others who are nearby. The user studies on those systems showed that users are comfortable sharing their favorite songs with others whether they are friends or strangers. It has also been reported that users share songs mainly because they want to recommend songs that others would like, disseminate their favorite songs, talk about shared songs with others, and so on.

Some applications designed for listening to music together have also been released (e.g., Group Session by Spotify [23] and JQBX [24]). In these applications, any user can let other users listen to her favorite songs by acting like a DJ. Users can also communicate with each other via a text chat system while listening to songs.

Our study is different from the above studies and services in that we introduce the two architectures for reaction visualization and song selection from users' favorite songs. In most of the above cases, because users' reactions are not visualized or are visualized only when chatting with text messages, it is difficult for users to feel social connection with each other. On the other hand, because the first architecture on Kiite Cafe visualizes four kinds of reactions, users can more strongly feel that they are enjoying music with others. In addition, existing systems require users to actively nominate or share songs or act like a DJ, but some users may hesitate to do that, especially if there is a large audience. In contrast, the second architecture on Kiite Cafe enables a user's favorite songs to be automatically played. This lets any user share her favorite songs with other users and see the moment when they start liking those songs.

6.2 Group Recommendation Algorithms

Various song recommendation methods for a single user have been proposed [25–32]. One of the biggest differences between the methods for a single user and those for a group of users is that the latter methods need to take multiple users' preferences into account. A general approach is to aggregate each user's preferences by, for example, merging recommendation results generated for each user according to voting strategies [14, 33]. However, such an approach cannot always reflect minority preferences.

To solve this problem, a concept of *fairness* has been recently introduced into group recommendation algorithms [34–38]. The basic idea of fairness is that a list of items recommended to a group is fair when each user in the group can find at least one item in the list that she finds satisfying. In the context of music recommendation, existing studies have only considered the fairness for users as audiences. On the other hand, fairness for users as curators as well as audiences is achieved by Kiite Cafe because of the second architecture in which each user's favorite songs are fairly selected and played as described in section 2.2.2. In particular, the "highly recommended list" plays an important role in achieving fairness as curators. When a user's favorite and/or recommended song can be listened to with other users, the user is satisfied from audience and curator viewpoints.

7. CONCLUSION

In this paper, we described Kiite Cafe, a web service that enables users to communicate while listening to the same music online. Kiite Cafe is characterized by two proposed architectures for visualizing each user's reactions and selecting played songs from users' favorite songs. Our experimental results quantitatively showed three effects provided by the proposed architectures. We believe that these architectures are also useful for different types of music interfaces, including a three-dimensional interface where user avatars could listen to the same music in a virtual reality (VR) venue.

8. ACKNOWLEDGMENTS

We thank Crypton Future Media, INC. for developing Kiite with us and Nico Nico Douga for initially tacitly (later explicitly) encouraging us to work on this research. We also would like to extend our appreciation to users of Kiite and Nico Nico Douga, creators of VOCALOID songs, creators of popularity rankings of VOCALOID songs, and all people who have created, supported, and enjoyed the VOCALOID culture. This work was supported in part by JST ACCEL Grant Number JPMJAC1602, JST CREST Grant Number JPMJCR20D4, and JSPS KAKENHI Grant Number 20K19934, Japan.

9. REFERENCES

[1] K. Sedgman, "Coughing and clapping: investigating audience experience," *Cultural Trends*, vol. 24, no. 4, pp. 324–326, 2015.

[2] S. C. Brown and D. Knox, "Why go to pop concerts? The motivations behind live music attendance," *Musicae Scientiae*, vol. 21, no. 3, pp. 233–249, 2017.

[3] A. Bassoli, J. Moore, S. Agamanolis, and H. C. Group, "tunA: Local music sharing with handheld Wi-Fi devices," in *Proceedings of the 5th Wireless World Conference 2004*, ser. WWC '04, 2004.

[4] M. Håkansson, M. Rost, and L. E. Holmquist, "Gifts from friends and strangers: A study of mobile music sharing," in *Proceedings of the 10th European Conference on Computer-Supported Cooperative Work*, ser. ECSCW '07, 2007, pp. 311–330.

[5] M. Håkansson, M. Rost, M. Jacobsson, and L. E. Holmquist, "Facilitating mobile music sharing and social interaction with Push!Music," in *Proceedings of the 40th Annual Hawaii International Conference on System Sciences*, ser. HICSS '07, 2007, pp. 87–96.

[6] H. Kenmochi and H. Ohshita, "VOCALOID - commercial singing synthesizer based on sample concatenation," in *Proceedings of the 8th Annual Conference of the International Speech Communication Association*, ser. INTERSPEECH '07, 2007, pp. 4009–4010.

[7] J. F. McCarthy and T. D. Anagnost, "MusicFX: An arbiter of group preferences for computer supported collaborative workouts," in *Proceedings of the 1998 ACM Conference on Computer Supported Cooperative Work*, ser. CSCW '98, 1998, pp. 363–372.

[8] A. Crossen, J. Budzik, and K. J. Hammond, "Flytrap: Intelligent group music recommendation," in *Proceedings of the 7th International Conference on Intelligent User Interfaces*, ser. IUI '02, 2002, pp. 184–185.

[9] K. O'Hara, M. Lipson, M. Jansen, A. Unger, H. Jeffries, and P. Macer, "Jukola: Democratic music choice in a public space," in *Proceedings of the 5th Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques*, ser. DIS '04, 2004, pp. 145–154.

[10] D. Sprague, F. Wu, and M. Tory, "Music selection using the PartyVote democratic jukebox," in *Proceedings of the Working Conference on Advanced Visual Interfaces*, ser. AVI '08, 2008, pp. 433–436.

[11] G. Popescu and P. Pu, "What's the best music you have? Designing music recommendation for group enjoyment in GroupFun," in *Proceedings of the CHI '12 Extended Abstracts on Human Factors in Computing Systems*, ser. CHI EA '12, 2012, pp. 1673–1678.

[12] F. Vieira and N. Andrade, "Evaluating conflict management mechanisms for online social jukeboxes," in *Proceedings of the 16th International Society for Music Information Retrieval Conference*, ser. ISMIR '15, 2015, pp. 190–196.

[13] M. Goto, "Active music listening interfaces based on signal processing," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, ser. ICASSP '07, 2007, pp. IV–1441–IV–1444.

[14] M. Kompan and M. Bielikova, "Group recommendations: Survey and perspectives," *Computing and Informatics*, vol. 33, no. 2, pp. 446–476, 2014.

[15] E. Pariser, *The filter bubble: What the Internet is hiding from you.* Penguin Press, 2011.

[16] M. Taramigkou, E. Bothos, K. Christidis, D. Apostolou, and G. Mentzas, "Escape the bubble: Guided exploration of music preferences for serendipity and novelty," in *Proceedings of the 7th ACM Conference on Recommender Systems*, ser. RecSys '13, 2013, pp. 335–338.

[17] Y. C. Zhang, D. O. Séaghdha, D. Quercia, and T. Jambor, "Auralist: Introducing serendipity into music recommendation," in *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining*, ser. WSDM '12, 2012, pp. 13–22.

[18] J. Fuller, L. Hubener, Y. Kim, and J. H. Lee, "Elucidating user behavior in music services through persona and gender," in *Proceedings of the 17th International Society for Music Information Retrieval Conference*, ser. ISMIR '16, 2016, pp. 626–632.

[19] C.-N. Ziegler, S. M. McNee, J. A. Konstan, and G. Lausen, "Improving recommendation lists through topic diversification," in *Proceedings of the 14th International Conference on World Wide Web*, ser. WWW '05, 2005, pp. 22–32.

[20] J. Cramer, H.-H. Wu, J. Salamon, and J. P. Bello, "Look, listen, and learn more: Design choices for deep audio embeddings," in *Proceedings of the 2019 IEEE International Conference on Acoustics, Speech and Signal Processing*, ser. ICASSP '19, 2019, pp. 3852–3856.

[21] M. Goto and R. B. Dannenberg, "Music interfaces based on automatic music signal analysis: New ways to create and listen to music," *IEEE Signal Processing Magazine*, vol. 36, no. 1, pp. 74–81, 2019.

[22] P. Knees, M. Schedl, and M. Goto, "Intelligent user interfaces for music discovery," *Transactions of the International Society for Music Information Retrieval*, vol. 3, no. 1, pp. 165–179, 2020.

[23] "Group session - Spotify," https://support.spotify.com/us/article/group-session/.

[24] "JQBX - Listen Together. DJ Online. Discover New Music." https://www.jqbx.fm/.

[25] K. Yoshii, M. Goto, K. Komatani, T. Ogata, and H. G. Okuno, "Hybrid collaborative and content-based music recommendation using probabilistic model with latent user preferences," in *Proceedings of the 7th International Conference on Music Information Retrieval*, ser. ISMIR '06, 2006, pp. 296–301.

[26] M. Tiemann, S. Pauws, and F. Vignoli, "Ensemble learning for hybrid music recommendation," in *Proceedings of the 8th International Conference on Music Information Retrieval*, ser. ISMIR '07, 2007, pp. 179–180.

[27] K. Yoshii and M. Goto, "Continuous pLSI and smoothing techniques for hybrid music recommendation," in *Proceedings of the 10th International Society for Music Information Retrieval Conference*, ser. ISMIR '09, 2009, pp. 339–344.

[28] Z. Xing, X. Wang, and Y. Wang, "Enhancing collaborative filtering music recommendation by balancing exploration and exploitation," in *Proceedings of the 15th International Society for Music Information Retrieval Conference*, ser. ISMIR '14, 2014, pp. 445–450.

[29] A. Vall, M. Skowron, P. Knees, and M. Schedl, "Improving music recommendations with a weighted factorization of the tagging activity," in *Proceedings of the 16th International Society for Music Information Retrieval Conference*, ser. ISMIR '15, 2015, pp. 65–71.

[30] D. Liang, M. Zhan, and D. P. W. Ellis, "Content-aware collaborative music recommendation using pre-trained neural networks," in *Proceedings of the 16th International Society for Music Information Retrieval Conference*, ser. ISMIR '15, 2015, pp. 295–301.

[31] R. S. Oliveira, C. Nóbrega, L. B. Marinho, and N. Andrade, "A multiobjective music recommendation approach for aspect-based diversification," in *Proceedings of the 18th International Society for Music Information Retrieval Conference*, ser. ISMIR '17, 2017, pp. 414–420.

[32] O. Gouvert, T. Oberlin, and C. Févotte, "Matrix cofactorization for cold-start recommendation," in *Proceedings of the 19th International Society for Music Information Retrieval Conference*, ser. ISMIR '18, 2018, pp. 792–798.

[33] L. Baltrunas, T. Makcinskas, and F. Ricci, "Group recommendations with rank aggregation and collaborative filtering," in *Proceedings of the Fourth ACM Conference on Recommender Systems*, ser. RecSys '10, 2010, pp. 119–126.

[34] S. Qi, N. Mamoulis, E. Pitoura, and P. Tsaparas, "Recommending packages to groups," in *Proceedings of the IEEE 16th International Conference on Data Mining*, ser. ICDM '16, 2016, pp. 449–458.

[35] D. Serbos, S. Qi, N. Mamoulis, E. Pitoura, and P. Tsaparas, "Fairness in package-to-group recommendations," in *Proceedings of the 26th International Conference on World Wide Web*, ser. WWW '17, 2017, pp. 371–379.

[36] L. Xiao, Z. Min, Z. Yongfeng, G. Zhaoquan, L. Yiqun, and M. Shaoping, "Fairness-aware group recommendation with pareto-efficiency," in *Proceedings of the Eleventh ACM Conference on Recommender Systems*, ser. RecSys '17, 2017, pp. 107–115.

[37] D. Sacharidis, "Top-N group recommendations with fairness," in *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, ser. SAC '19, 2019, pp. 1663–1670.

[38] M. Stratigi, J. Nummenmaa, E. Pitoura, and K. Stefanidis, "Fair sequential group recommendations," in *Proceedings of the 35th Annual ACM Symposium on Applied Computing*, ser. SAC '20, 2020, pp. 1443–1452.