

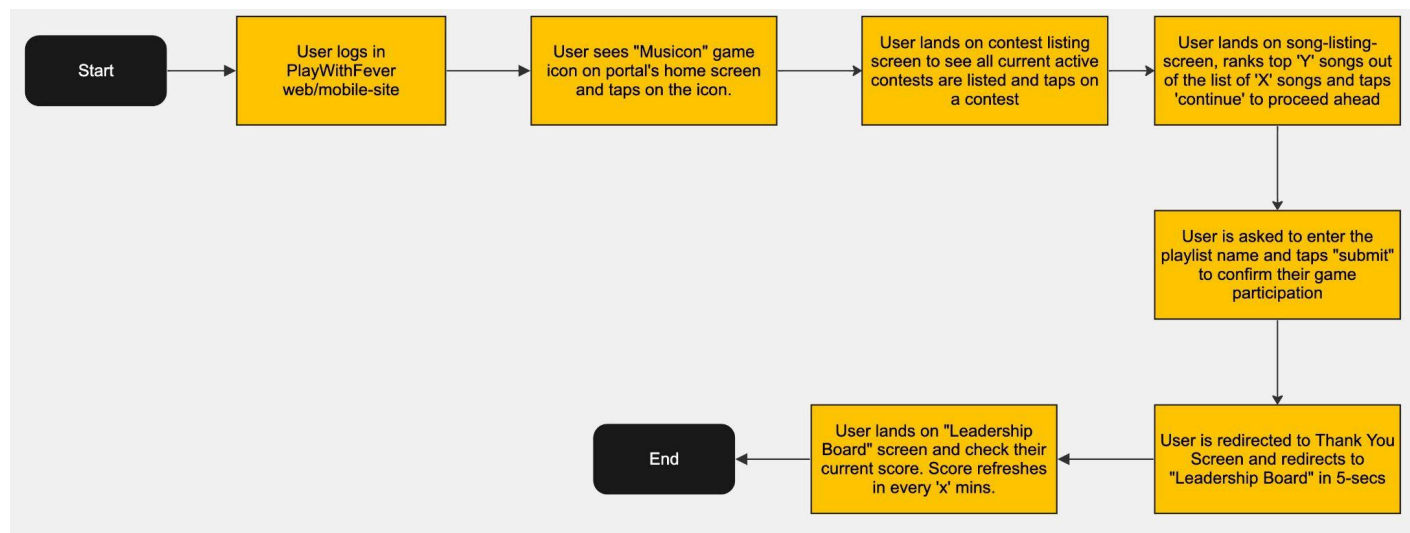
Motivation & Goal:

In a world where GRPs & TRPs of ads(TV & radio) are in decline, brand solutions that offer better results on any one of these metrics would surely attract spending from brands looking for performance. Today brands are willing to increase their advertising spending on digital solutions that could drive better top-of-the-mind recall as a result of higher attention span.

With this need, Play-With-Fever strives to become a platform that offers a bouquet of such digital solutions driving different brand goals: higher engagement, higher conversion, higher UGC(user-generated content), higher recall, etc.

Musicon is one such upcoming property, on Play-with-fever, that bridges the gap between radio and digital.

“Musicon” User Flow:



Primary Scenarios:

❖ Musicon Game Workflow

- **Song Listing Screen** is where users would see the **list of X(eg:20)** songs out of which they **must pick their Y(eg:5) favorite** songs to proceed further
- It should be verified that this list of 20 songs is configurable from the **Game Management Console(GMS)** such that for each contest song listings could vary. The count of offered songs(X) & favorite picks(Y) should be configurable from the Game Management Console(admin portal)
- It should be verified that users are **able to listen to each of the listed songs** in order to pick their favorite X picks out of the list of Y songs
- It should be verified that for each song - **Song Title, Singer Name, Album Thumbnail & Album Name** - is available on the listing screen
- It should be verified that as soon as users select their 5 favorite songs flow auto scrolls to the bottom fold where **“Continue” CTA** is present in **activated state**. The count of 5 should be configurable from GMS
- It should be verified that on tapping “Continue”, in prev step, users are taken to the **Song Ranking Screen** where they are asked to rank their selected songs in order of preference from **rank-1 to rank-5**. As soon as all ranks get assigned, “Continue” CTA at the bottom of the screen activates
- It should be verified that on tapping “Continue”, in prev step, users are taken to the **Playlist Submission Screen** where they are asked to **assign a name to the playlist**. Tapping on “Submit” on this step should enroll that user for the contest
- It should be verified that on tapping “Submit”, in prev step, users are congratulated for their participation and subsequently **redirected to the “Leadership Board”** which showcases their current game score & ranking
- Verify that this screen UX/UI is in line with the communicated designs
- Verify that the whole experience is mobile first and screen renders correctly across all major mobile models and display sizes

❖ Musicon Player Scoring Logic

- It should be verified that each game participant is given a score based on the top 'Y' songs in the system. Verify that this score refreshes in every 'z' mins. 'z' should be configurable from GMS
- It should be verified that game scoring logic works for all combinations of Offered songs (X) & favorite picks (Y)
- It should be verified that in every 'z' mins, top 'Y' songs in the system is determined based on a rank scoring logic where each song 'pick' by a player leads to increase the song score by (1 / 'pick' rank) as shown below:

Players join the game and rank the songs as follows:

Player 1	Player 2	Player 3
Song 3	Song 5	Song 2
Song 4	Song 3	Song 4
Song 1	Song 4	Song 3
Song 2	Song 1	Song 5
Song 9	Song2	Song 7

Song scores are determined, based on above picks, as follows:

Scheduled List	Song Score	
Song 1	$1/3+1/4$	0.58
Song 2	$1/4+1/5+1$	1.45
Song 3	$1+1/2+1/3$	1.83
Song 4	$1/2+1/3+1/2$	1.33
Song 5	$1+1/4$	1.25
Song 6	0	0.00
Song 7	$1/5$	0.20
Song 8	0	0.00
Song 9	$1/5$	0.20
Song 10	0	0.00

To determine top Y(eg:5) songs in the system as:

Song 3
Song 2
Song 4
Song 5
Song 1

It should be verified that player scores are updated, using weight multiplication, as soon as top 5 songs are updated as shown below:

Player 1	Player 2	Player 3		Top 5 Songs		Rank Weight
Song 3	Song 5	Song 2		Song 3		Y(eg:5)
Song 4	Song 3	Song 4		Song 2		Y-1(eg:4)
Song 1	Song 4	Song 3		Song 4		Y-2(eg:3)
Song 2	Song 1	Song 5		Song 5		Y-3(eg:2)
Song 9	Song2	Song 7		Song 1		1(eg:1)

Max Score: $5 \times 5 + 4 \times 4 + 3 \times 3 + 2 \times 2 + 1 = 55$

Player	Scoring Logic	Score
Player 1	$[(5 \times 5 + 4 \times 3 + 3 \times 1 + 2 \times 4 + 0) / 55] \times 1000$	873
Player 2	$[(5 \times 2 + 4 \times 5 + 3 \times 3 + 2 \times 1 + 1 \times 4) / 55] \times 1000$	818
Player 3	$[(5 \times 4 + 4 \times 3 + 3 \times 5 + 2 \times 2 + 0) / 55] \times 1000$	927

❖ 'Musicon' Leadership Board

- Leadership board is the place where players can check out
 - Their **score and relative performance** as compared to the overall user base that participated in the contest, under **"Rankings" tab**
 - Song performance indicated as a percentage of points earned from total allotted points to all songs, under **"Scorecard" tab**
- Leadership board should reside inside the game
- It should be verified that players are ranked based on their scores and top **30 ranks should be loaded in the first instance** with page load. **Lazy loading** should be employed to render ranks afterwards
- It should be verified that a player is able to tap on other players and see their song selection from the "Rankings" tab
- Verify that this screen UX/UI is in line with the communicated designs
- Verify that the whole experience is mobile first and screen renders correctly across all major mobile models and display sizes

❖ 'Musicon' Game History

- Game History is the place where players would come to check their performance in 'Musicon' contests in which they had participated in the past
- It should be verified that the section lists out all the contests(in which player had participated or in-progress) along with the Contest name and their end date. In-progress contests should be marked as active
- It should be verified that tapping on any contest card takes the user to the leadership board for that contest
- Game's history should reside inside the 'Musicon' game
- It should be verified that contests are listed in reverse chronological order

❖ Game Management System

- Game Management System(GMS) is an admin portal which would be used to manage games on PlayWithFever
- It should be verified that admin is able "Create" a 'Musicon' game contest from GMS

- It should be verified that in order to create a 'Musicon' contest admin mandatorily provisions the following:
 - Contest name
 - Game start timestamp
 - Game end timestamp
 - Count of offered song choices (X)
 - Count of offered favorite picks (Y)
 - Winning Criteria (Top 'x' rank)
 - Winner Reward Type (Coupon / Passes / Others)
 - Winner Reward Text
 - Coupon List Upload / Download Links Upload via excel
 - Score Refresh Frequency ('z' mins)
- It should be verified that the admin must add the mandatory song's metadata before (s)he can schedule it for the next period. The song's metadata should include: Song Title, Singer Name, Album Thumbnail & Album Name
- It should be verified that once admin 'Create' the game, it should start showing up in the portal right away. But verify that playlist and user flow on playwithfever is activated only post the start timestamp and deactivates post end timestamp
- It should be verified that the admin is able to create multiple contests at once.

Secondary Scenarios:

❖ Login Screen

- The login screen is the screen that users land on post typing(or clicking from search) www.playwithfever.com
- Verify that logged-in users are directly taken to the game listing screen
- Verify that the login screen has both "Login" & "Register" as options
- Verify that the user who opts to "Register" is asked to fill in the phone Number & OTP
- Verify that the user who has forgotten the password can click the "Forgot Password" link for OTP login

- Verify that once the user has logged in then they remain logged-in on subsequent visits until the browser cache is cleared
- Verify that the whole experience is mobile first and screen renders correctly across all major mobile models and display sizes

❖ Game Listing Screen

- The game listing screen is the homescreen of the Play-With-Fever platform and users land on this screen post registering or logging in
- Verify that the “Musicon” game is listed on this screen
- Verify that this screen UX/UI is in line with the communicated designs
- Verify that the whole experience is mobile first and screen renders correctly across all major mobile models and display sizes

❖ Game Reward Screen

- The reward screen should show all the rewards that a user had won in their journey on play with fever
- It should be verified that if a player is a winner, among top ‘x’ ranks, then their reward is shown under this section, in tile format, in reverse chronological order
- It should be verified that tapping on the reward tile in above step opens a pop-up displaying the winning text along with coupon code(for ‘coupon’ reward type), download link(for ‘Pass’ reward type) or email id prompt(for ‘others’ reward type)
- Verify that not 2 winners get the same coupon or download entry from the list that was uploaded at the time of contest creation.
- Verify that this screen UX/UI is in line with the communicated designs
- Verify that the whole experience is mobile first and screen renders correctly across all major mobile models and display sizes

❖ Game Guide Screen

- The guide screen should show all the participation guidelines, as shared by the play with fever team
- Verify that this screen UX/UI is in line with the communicated designs

- Verify that the whole experience is mobile first and screen renders correctly across all major mobile models and display sizes

❖ **User Profile & Game Data (Existing Fever Data Platform i.e, API link)**

- Verify that the admin can retrieve game & contest wise report from the GMS that should have the following details:
 - Game Name
 - Contest Name
 - Player Name
 - Player Mobile
 - Player Email
 - Player Rank

Non-Functional Requirements:

- Verify that the “Musicon” game is optimized to ensure a smooth gaming experience for at least 150K concurrent users.
- Verify that Page Loading Speeds are very high in general with use of skeleton screens to ensure less perceived loading times.
- Verify that gameplay screen experience is as good as native and only dynamic elements reload, and static elements should not require reload post first page load.