The ui/ux of the project needs to be heavily influenced by Microsoft Fluent Design paradigm. The official site of fluent design paradigm can be accessed at:

https://developer.microsoft.com/en-us/windows/apps/design

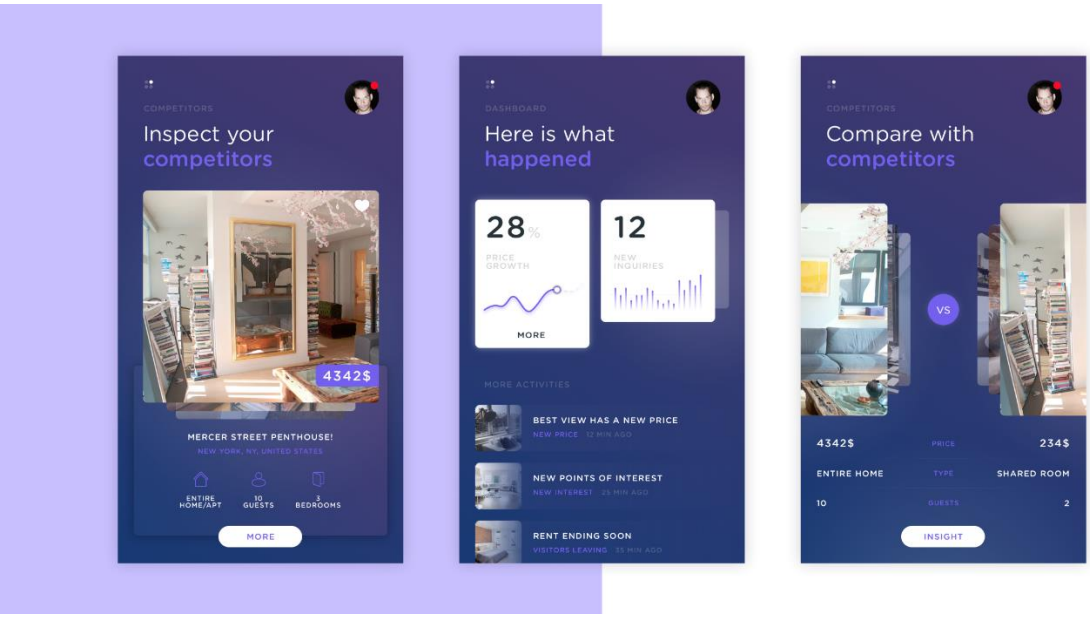
The fluent design elements toolkit for various designing software can be downloaded at:

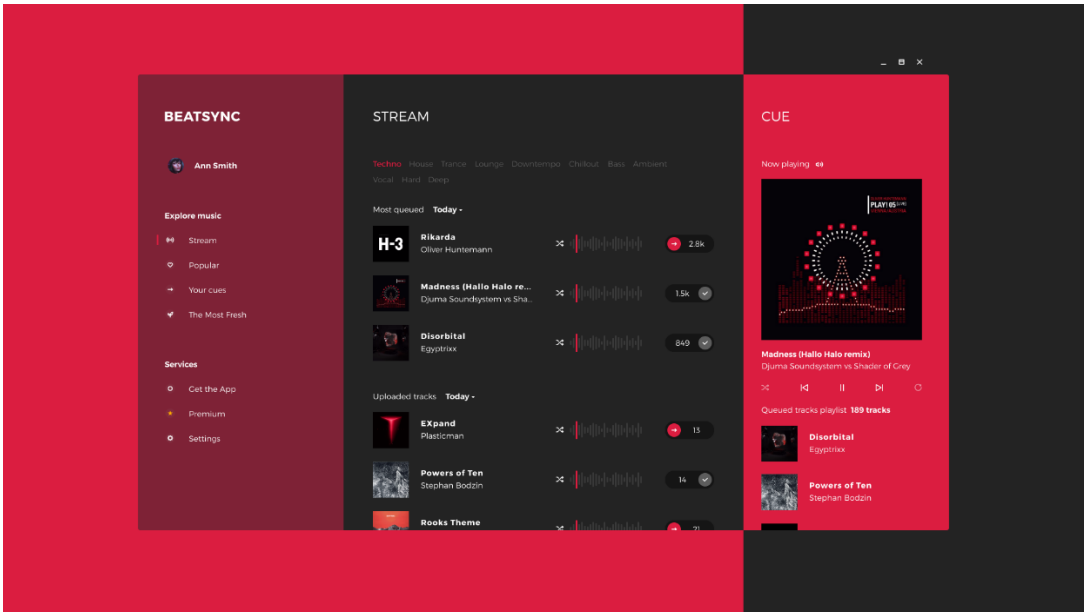
https://docs.microsoft.com/en-us/windows/uwp/design-downloads/index

Various examples of how fluent design looks and feels for mobile and web user interfaces can be seen at:

https://dribbble.com/search?q=fluent

While there is no better alternative than checking out all the dribbble shots returned by the above link, I am enclosing few snipped images from dribbble for quick overview.



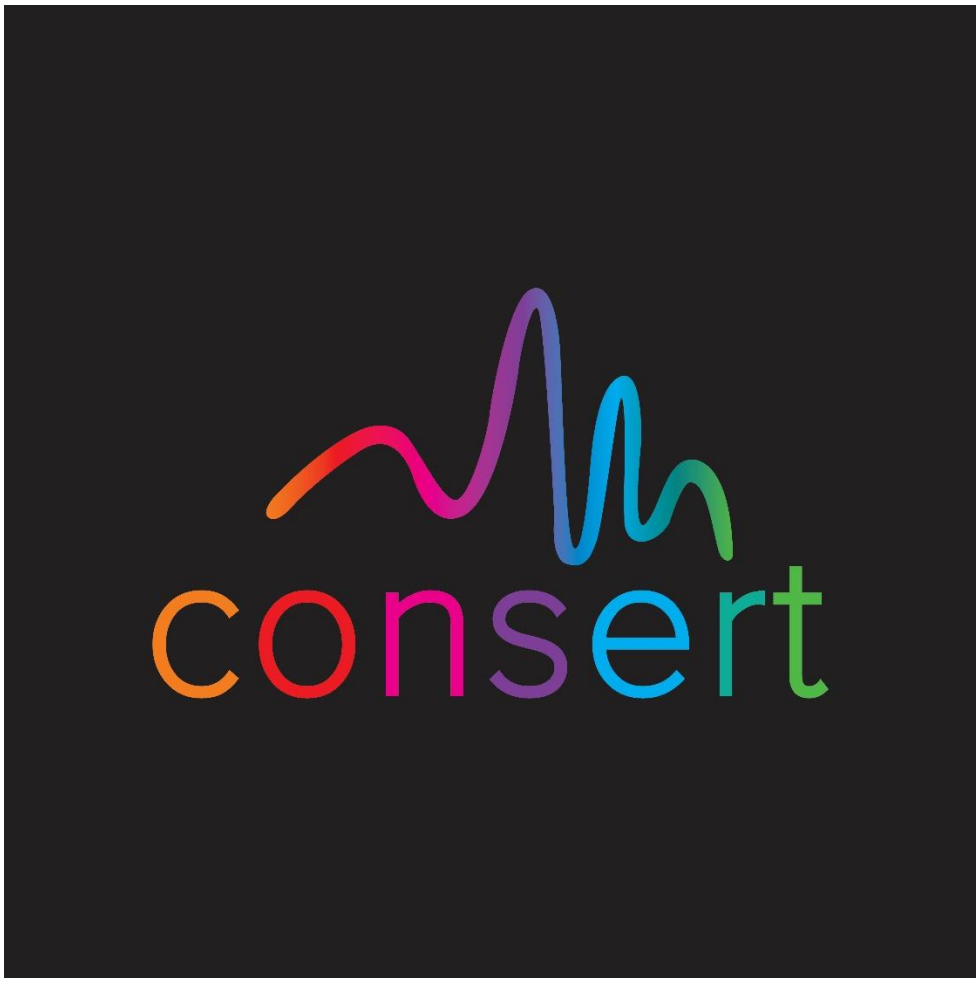


I would definitely recommend the designer to get acquainted with the concepts introduced in fluent design for seamless communication.

Also, one of the **mandatory** requirements is to represent/mark all the design elements that will be used in ui composition with their respective **control and pattern types** defined in UWP app (fluent deisgn) guidelines. The **same constraints** apply to **layout** and **styles** of those elements. These restrictions are imposed to facilitate high fidelity reproduction of ui in real application, and thus, frontend coding.

Now, some basic requirements for the web app are mentioned below:

- Font: PROXIMA NOVA
- Iconography: Custom icons included in the delivery or MDL2
- Color **Theme** / Background Color: **Dark/Black**
- Accent Colour: Choose those accent colors from our multicolored logo that are well-suited for a dark ui. Our logo, currently in the late stage of prototyping, is shown below:



- The design needs to be responsive in nature. So, the ui should atleast be designed for 3 different device screen sizes profile i.e. mobile, tablet, pc/laptop.
- This ui is meant to be **run in web-browsers** in all execution environments i.e. **no native mobile** apps, only mobile/tablet/pc web apps.

Features

Let's start with a high-level features overview of our web application:

name: consert

A dedicated **social** web app that suggests users past, happening, and upcoming **live concert videos** based on their Spotify history and preferences.

- The only way to access our web application catalogue is through **spotify signup/login.**
- As the web application is focused on suggesting videos, the **primary content** our application will be **videos**.
- Users can perform certain actions on the videos, and they are:
    1. **Play/Stop/Resume** the video from the feeds.
    2. **Play** the video on a separately dedicated page.
    3. **Like/Save/Favourite** the video.
    4. **Add** any video to Play Queue/ Up Next.
    5. **Remove** any video from Play Queue/ Up Next.
    6. **Add** a video to his **playlist**.
    7. **Share** a video to his social network account such as fb, twitter etc.
    8. **Repost/Share** a video on our website.
    9. **Repost/ Share a** playlist of videos on our web app, or on his social networks.
    10. **Mark video** as **not Interested**.
    11. Users can start **artist or track based radio station** for any given video.
    12. Users can **comment** on a video.
- Additionally, user can **search** for videos, and **filter the search results** by popularity, artists, tags etc.
- User can **visit** any other **user/artist profile** page.
- User can **follow** another user/artist.
- User can **share/repost** any other user profile on our web app or social media.
- User can view **related artist/users** and can also get **suggestions about who to follow** etc.
- The **listening history** of user will be saved.
- User must be able to **choose from palette of accent colours** options provided by default in our dark themed web app.
- Our web application will suggest **multiple categories of video feeds** (similar to 9gag's hot, trending, and reddit's hot, new, rising) to the users.
- The user must have capability to **minimize the currently playing video in a side panel** and continue browsing our web app for other videos that can also be added to play queue/up next.

Primary Content Look and Feel

As **video is our primary content**, it is natural to question how that content should be displayed in our web app. Below are some guidelines about how a video thumbnail might display in our app:

- The video content, thumbnails etc. must appear as immersive as possible.
- The video thumbnails should also have inline command bar to perform actions such as like, share, repost etc..

- The above shot shows a minimalistic view of video thumbnails with video properties such as name is appearing on the thumbnail itself. Adding an inline command bar in a creative manner to such thumbnails will give our video content immersive, modern, and minimalistic look.
- Also, to preserve the consistency in the immersive nature, different size of devices must show different no. of columns of thumbnails in a row. For example, a mobile phone can show only one thumbnail is a row, tablet can two, and laptops/pc can show three thumbnails.
- Playing a video expands its thumbnail to cover the width of the device, and pin it to the top edge of the screen, while reflowing the other video thumbnail suggestions simultaneously. The same thing will occur when a user accesses the video with its dedicated video url either from social media links or from somewhere else, but the suggestions in that case will not be videos from the feed, but will be the videos related to the current video.
- Search, scroll, and navigation must automatically pin the currently playing video to the sidebar.

- On the whole, the tasks of navigating, searching, playing, liking etc. must be fluid and smooth.
- CSS **Flexbox** grid should be used to manage content reflow, resize etc.

Navigation

Navigation Panel is the starting point of user experience. It's how users find the content and features they're interested in. Keeping it simple, clean, and consistent is our primary motive. Navigation Panel can be implemented with Navigation View class in uwp fluent design. It also provides a very nice opportunity to use Acrylic material, reveal, parrallax and other stylistic components provided by the fluent design paradigm.

The top-level navigation panel must be properly aligned to top or bottom of the window as per as the device profile. Some general tips for layout design for different screen sizes can be seen at:

https://docs.microsoft.com/en-us/windows/uwp/layout/screen-sizes-and-breakpoints-for-responsive-design

The **transition** across various **navigational items and content** must create a **compelling and intuitive** navigational experience. The **norms** of **motion and animation** defined in fluent design paradigm can be read at:

https://docs.microsoft.com/en-us/windows/uwp/style/connected-animation

I suggest the designer to follow those design guidelines in my web application to **avoid unnecessary revisions**.

The top-level navigation menu must be implemented with **acrylic** navigation view panes, and navigation items must use **reveal** to illuminate important elements.

I can visualize at least the following navigation items to exist in top-level navigation pane:

a) Feeds/Home/Browse
b) Library/Collection/Your Music
c) Search
d) Account/Profile/Settings

Each menu items, which contain sub-items, are explained further in this document.

a) Feeds/Home/Browse Menu Item

This is the default section that will be open on successful login. The **Feeds** section will contain Pivot and Tabs one of which will be preselected on user login/signup. Pivot and Tabs are generally top aligned with an underline in accent colour representing the activated tab.

The different tabs and their content is elucidated below:

Stream/Live Feed:

The contents of **Stream** Feed are:

- Videos of **currently live** broadcasts that suit user music taste.
- Thumbnails showing some of the **up-coming live** video broadcasts according to user music preferences.
- Social activities, happening in our web app, of the users/artists followed such as likes, reposts etc.
- The main attribute of this feed group will be **high-velocity of video content recommendations** by our machine learning algorithms i.e. content will refresh faster in these feed tab compared with others.

Spotlight/Highlight/Featured Feed:

As the name suggests, this feed group will have:

- Top **past live performances videos** taking user preferences into account.
- Suggested **new artist/user to follow**.
- **Site wide popular/trending live performance videos** filtered as per as user musical taste.
- **Videos** preferred by the users having similar musical inclinations as current user.
- This feed group will have **medium-velocity content refresh**.

Genres/Mood Feed:

- This tab will have **past live video** performances categorized as per as their genres, moods etc.
- As the listings are genre categorized, this tab will have one more level of nesting to view the video catalogued in a genre i.e. a user can click/tap a genre and navigate to view the video recommendations in that genre.
- The **order** of the **genre suggestion categories** will be based on user history and preferences.

Discover/Stumble Feed:

It will help user to discover new live music videos:

- Based on Artists.
- Based on Songs.
- Based on Playlists.

b) Your Music/Library/ Collection Menu Item

The Pivot and Tabs contained in this section are:

a) Likes/ Favourite Videos.
b) Playlists (Liked and Self-Created).
c) Stations that were started earlier (Both Artist and Track Based)
d) Listening History.
e) Following & Followers(user/artist).

All the tabs of this menu item are pretty much self-explanatory.

c) Search Menu Item

- Search experience can be designed according to fundamentals mentioned on:

  https://docs.microsoft.com/en-us/windows/uwp/controls-and-patterns/search

- The **results to show on zero input canvas** i.e. when text input field is activated but no text has been entered can be recent search history, trending searches, contextual search suggestions. **Designer must decide** about the type of results that can be displayed in such case.
- **Auto-suggest** will kick in as soon as any character is type in the input field.

The **search result page** has to be well-designed.

d) Account/Profile/Settings Menu Item

Provide navigation options to the following pages/actions:

- **Profile** i.e. the web page that will be seen by other users of our web app.
- **About us**
- **Logout**
- **Help**
- **Terms of Use**

That summarizes the major navigational options and features defined in our application.

Miscellaneous

This section tries to cover all the remaining functionality and screens of our web app that were either not covered thoroughly, or remained untouched in the previous section:

- **404 resource not found** error page.
- **500 internal server** error screen.
- **user/artist** profile pages.
- When a user selects to play track/artist based **radio** from the video's inline command bar or from an artist's profile page, the station will open on a new page with video playing in the expanded mode **on an altogether new page layout**. Below that expanded player will be the suggestions of similar videos recommendations of the radio.
- When a live video link is shared on other social media accounts such as fb, twitter etc, clicking on that link will open a page of our web app showing the expanded video player **similar to** the previously mentioned **radio page**. But, if a user is **not** registered/logged in, those **video suggestions on the bottom will be little blurred**, and the user can play only the **original social media shared video** that is open in the expanded mode. For playing suggested videos, he/she has to login or signup. So, a spotify login/signup button will appear at the bottom of that page.