

Design Document

We thought of an idea we wanted to implement and then created the sketch, wireframe, and prototype for it.

Figma link:

https://www.figma.com/file/WaL_fyfQbmmSWwvQKS8xQur/Wireframe?type=design&node-id=38-415&mode=design&t=pDGV3SM9tjsXe3t1-0

Developmental process / implementing the design:

1. We created the 3 pages that displays the trending(main.dart), favorited(favorites_page.dart), and previously watched videos (History_page.dart)
2. Then we created the bottom navigation bar to access these pages with three icons

```
onTap: (int index) {  
  setState(() {  
    _currentPageIndex = index;  
  });  
  if (index == 0) {  
    // Trending page  
    Navigator.push(  
      context,  
      MaterialPageRoute(builder: (context) => const TrendingPage()),  
    );  
  } else if (index == 1) {  
    // Favorites page  
    Navigator.push(  
      context,  
      MaterialPageRoute(  
        builder: (context) => FavoritesPage(favoritedVideos: favoritedVideos), // MaterialPageRoute  
      ),  
    );  
  } else if (index == 2) {  
    // History page  
    Navigator.push(  
      context,  
      MaterialPageRoute(  
        builder: (context) => HistoryPage(clickedVideos: clickedVideos), // MaterialPageRoute  
      ),  
    );  
  }  
}
```

3. Added functionality to play videos(used onpress to access the videos and onTap to favorite the videos.) in all three pages

```
onTap: () {  
  clickedVideos.add(video);  
  VideoPage videoPage = VideoPage(  
    videoId: video.videoId,  
    title: video.title,  
    channelName: video.channelName,  
    viewCount: video.views,  
    channelId: '',  
  ); // VideoPage  
  Navigator.push(  
    context,  
    MaterialPageRoute(  
      builder: (context) => VideoPlayerPage(videoPage: videoPage),  
    ), // MaterialPageRoute  
  );  
},
```

4. Display favorited videos on the favorite page whenever the heart icon on the right side of the video is clicked. Whenever a user clicks a video, it is displayed on the history page

5. Added the search bar and logic to search for videos(uses the fetchSearchVideo() method to fetch for videos using a query and an api key from google cloud called Youtube data api v3). We understand that the api key should not be distributed, but it is being shown here as an example.

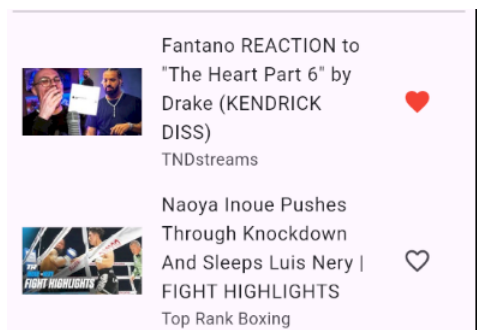
```
List<Video> clickedVideos = [];  
List<Video> favoritedVideos = [];  
String key = 'AIzaSyBbJ8_htCFUtXNPV6zERYgjAV_zzQwtyZk';
```

The catch to show the snackBar if no data is displayed

```
try {  
    results = await youtubeDataApi.fetchSearchVideo(query, key);  
} catch (e) {  
    print('Error fetching search results: $e');  
    ScaffoldMessenger.of(context).showSnackBar(  
        const SnackBar(  
            content: Text('No search results found'),  
        ), // SnackBar  
    );  
}
```

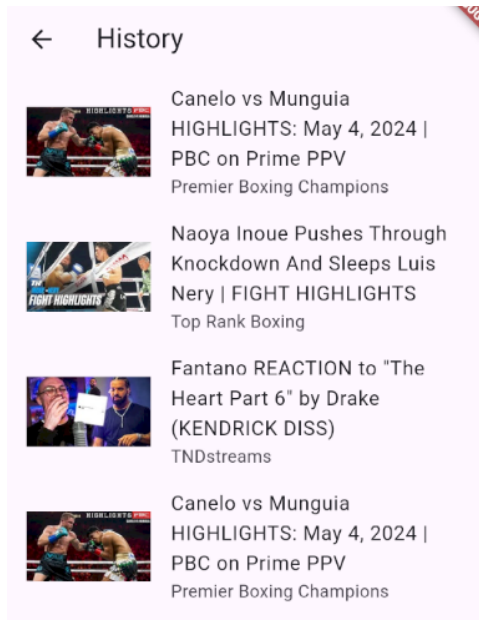
Summary of features:

1. Videos based on location: trending page displays videos that are popular in the users location.
2. Navigation Bar: allows users to switch between the three pages by clicking on each icon.
3. Favorites button: user is able to favorite a video by clicking on the favorite button. The favorite button is displayed on the right side of each video. Favoriting a video will display it on the favorites page. You can then unfavorite the videos on the favorites page if needed.

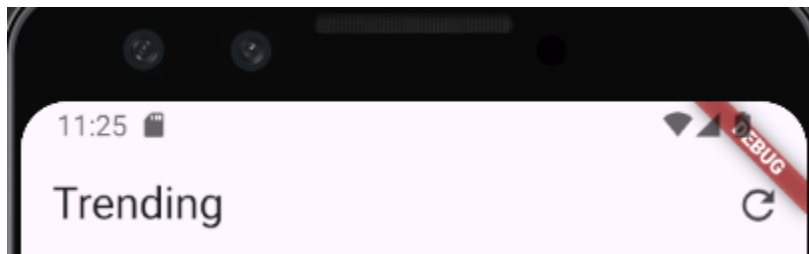


4. History page: whenever a user views a video, it is automatically sent to the history page. Unlike youtube, it does not move the video up from the history

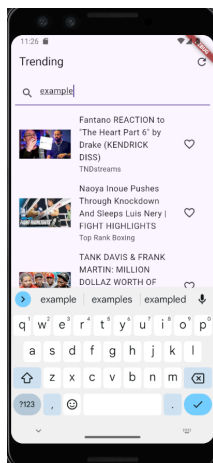
page. Ex. if you watched a video that was three videos ago and you click on it again, it will display again in the history.



5. Video player: when the user clicks a video, they are sent to another page(video_player_page.dart) where the video plays.
6. Refresh button: refreshes the data on the trending page.



7. Search bar: allows users to search for a specific video.



8. Page orientation: app can be viewed from portrait and landscape orientation
9. Vibration: device vibrates when a video is favorited or unfavorited.

Successes:

1. Successfully created the navigation and page functionalities(global variables to store the data so it does not disappear when switching between pages and .push to navigate to different screens)
2. Refresh button was simple to implement and refreshes the data as needed
3. Good time management within the team, good communication, met deadlines, good code with error exceptions.
4. Provided error handling. Ex. In the search functionality if no data appears for that search the snack bar appears displaying “results not found”

Issues:

1. Had difficulty getting video data in order to play the video and display its information(fetching from youtube data api pub dev) we got the channel name, view count, and title to display.
2. Search bar was difficult to implement as it could not always display information based on the users' search. If we wanted to search it would only work after clicking the trending page again.

Future development:

Could add original youtube functionality for future development

- a. Add more Like/Dislike button functionality like display information and create another page for like and dislike(similar to favorites)
- b. Subscription functionality(have a page where it displays channels you're subscribed to)
- c. Display Like/Dislike Count, date, description, and amount of subscribers