AVISHKAR '21

SOFTABLITZ 2K21

Brought to you by the Avishkar Team

PROBLEM STATEMENTS:

"SEEDHA PADHNE MAT BAITHO, PEHELE CHITRA DEKHO"

BASIC

- E - Padhai
- pow(2,n)
- WatchList

INTERMEDIATE

- Chitra Maker
- Covi-Katana
- OffCoder
- Fast Frenzy

ADVANCED

- Fair and Simple
- CodeDoc
- Concerto Virtuale
- BuddyFilm

BASIC INSTRUCTIONS

- A team can consist of at most 3 members.

- All branches are allowed, but each team must have at least 1 member from either CSE or IT. Cross-year teams are not allowed.

- Plagiarism in any form will result in elimination.

- Teams are allowed to attempt a maximum of two problem statements. If your first problem statement is complete then only we will consider your second problem statement for scoring. In short we will not be considering 2 incomplete projects for scoring. In any case we will be evaluating all your work and giving feedback.

- Bonus points will also be awarded if the team implements features other than the one specified.

- Use of Git and GitHub is compulsory and holds good weightage of score.

SUGGESTIONS

- Firstly read each and every problem statement and then decide which one suits best with your current skill set.

- Try to build a complete solution for the problem that you are attempting. In this way you will have a project worth mentioning on your resume.

- The 4th point on the left-side clearly states that even for winning you need to have at least one complete project with all the features mentioned.

- First try to implement as many features as you can. In case you get time after this, then only spend time polishing UI. Remember an application with very good UI but minimal functionality is useless. It is just like, "Naam bade par darshan chote". You won't be getting any marks for UI although we may use it as tie breaker.

- Remember to keep the spirit of Softablitz ablaze by not using libraries to implement features unless specified to do so, any libraries/packages/JARs used which are not mentioned by us will lead to negative points.

- Last but not the least for any doubt regarding problem statements contact coordinators as we are always here to help.
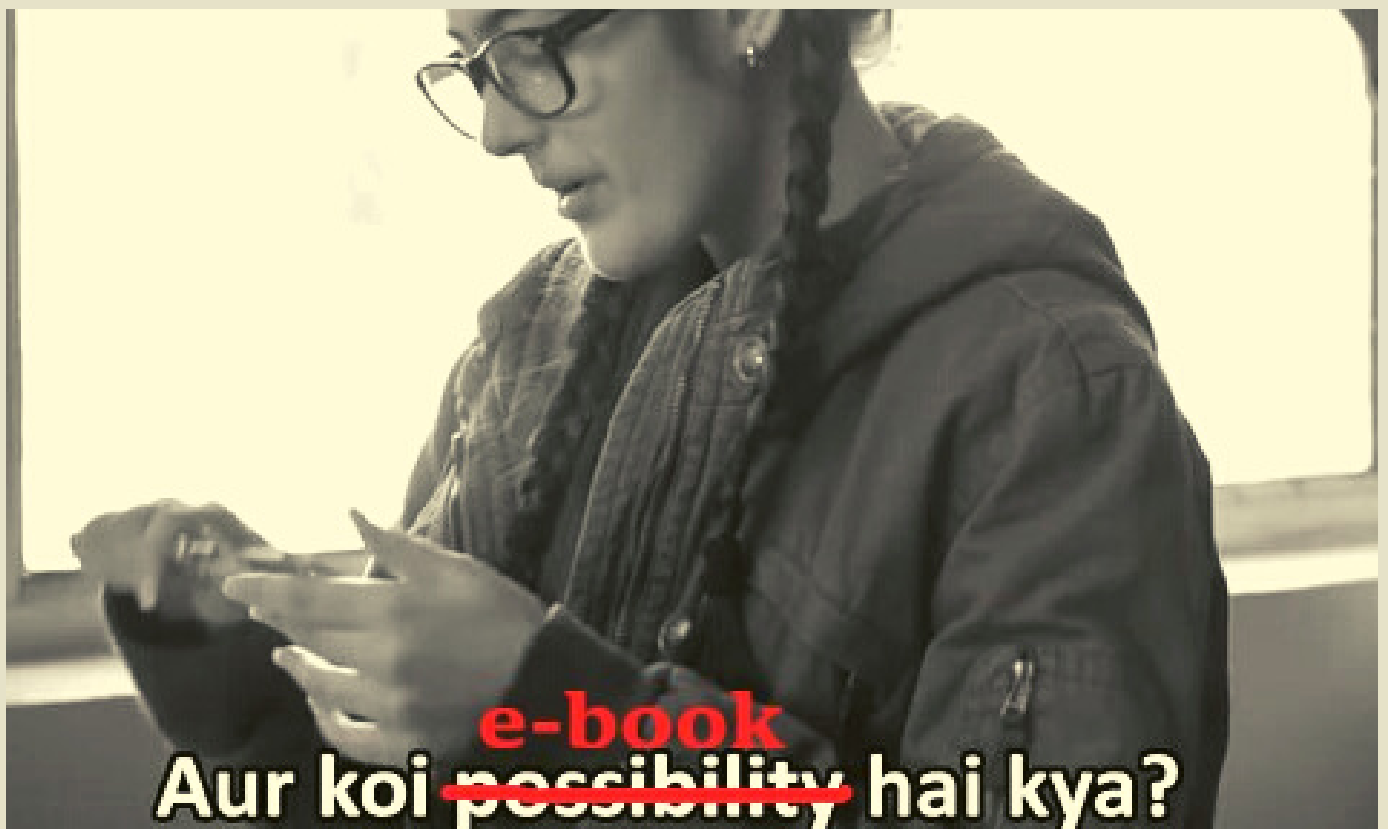
E - PADHAI

Difficulty : Easy

⭐

Meenal is Prodigy's topper. As expected, she is a major book nerd. But now poor Meenal is locked at home and cannot visit the library, so she decides to satisfy her book needs digitally.

Now she has ended up with a massive digital stash of e-books and is in a dire need of an application to organise them for her, so she can focus on her studies. Help her by creating an application following the below guidelines, and you might just end up in the topper's good books.

BASIC

1. Scan filesystem to add books (Books can be of different formats , i.e. .epub, .mobi, .pdf, .azw, etc.) [Added books can be kept at a separate location]
2. Search single directory
3. Search sub-directories
4. Search zip files
5. Show books in a catalogue (With metadata and cover image if present)
6. Edit metadata (Title, Author, Date, Rating, Tags, Publisher, etc.)
7. Option to sort by title, author, date, etc.
8. Option to filter by tags
9. Can create and edit different read-lists

ADVANCED

1. Fetch books from various free e-book sites (e.g. Library Genesis, archive.org, etc.). You can use their APIs or run scraping.
2. Can send/recommend books to friends via email.
3. Users can choose to encrypt their individual lists using a pass-key.
4. The platform should support multiple users and keep their data separated.

POW(2,N)

Difficulty : Easy

⭐

Uday is always fascinated by mobile games one such game is 2048 but he has made a promise to Meenal and Vaibhav that he won't be using mobile phone during study, but you know Uday he can't concentrate for 6 hours straight . So you had to help him by making an alternate version of 2048 for Windows PCs which will have below features in addition to the original one.

BASIC

1. Users can create an account and log in.
2. User's history of scores with hours they played should be saved locally.
3. Score Bound gameplay should be available.
4. Time-bound gameplay should be available.
5. Users should be able to play without having an account.
6. Game difficulty:
   a. Easy - Infinite undo
   b. Medium - No 2 consecutive undos
   c. Hard - No undo
7. Users should be able to save the winning screenshot.
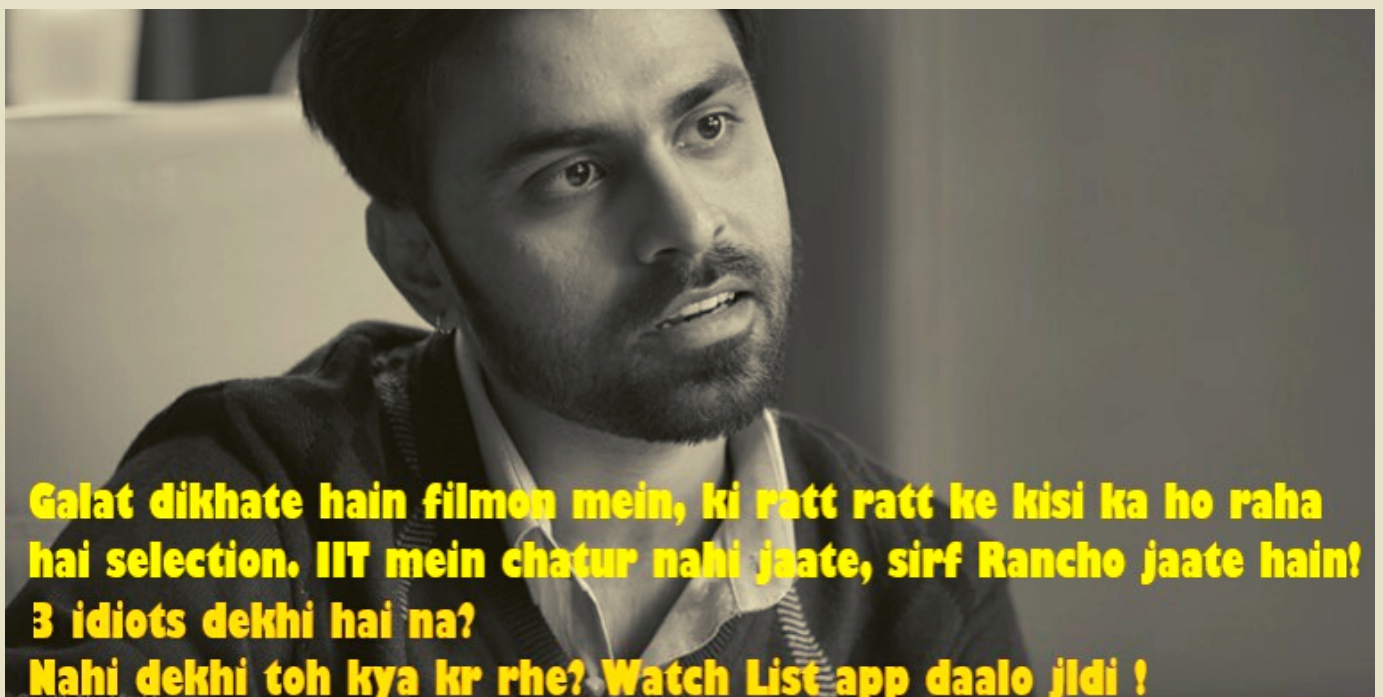
ADVANCED

1. Users can upload their photos, add their social media account to their profile and should be able to update that later on.
2. User's history of scores should be updated online whenever connected to the internet.
3. Users can see other user's profiles by searching their userId(userId should be unique).
4. Users can follow other users.
5. A global chart of users with their highest score and time required to score it should be visible.
6. Users should be able to save the game status and continue from where they left.
7. Dummy credit system to purchase new tile skins and game skins.

WATCHLIST

Difficulty : Easy

⭐

Uday loves watching movies when he's free i.e. not studying and most of the time he's free so he already watched a lot of movies and now when he plans to watch a movie most of his time goes in searching for a new movie he'll like to watch. Meena being a caring friend of him wants Uday to waste less time so he discussed with Vaibhav that if there had been an application which could predict which movies Uday would like to watch next it would save his lot of time. Meena and Vaibhav being busy in their JEE preparation wants you to design and develop an application which would recommend movies that the user would like to watch based on their previous usage of application.

Galat dikhate hain filmon mein, ki ratt ratt ke kisi ka ho raha hai selection. IIT mein chatur nahi jaate, sirf Rancho jaate hain! 3 idiots dekhi hai na? Nahi dekhi toh kya kr rhe? Watch List app daalo jldi !

BASIC

1. Users can create an account and log in (Age compulsory for movie recommendation).
2. On the first run, the user should be asked about preferred languages and genres
3. Should render organised dashboard with movies recommended on the basis of users' preferences (based on his previously liked movies and app history)
4. Should fetch and update movies using TMDB API. (API should be used only for fetching metadata of movies)
5. Movies should be rendered on the UI elegantly with proper metadata and its poster.
6. Users should be able to like, dislike movies and this should be the key factor for future recommendations.
7. Users can create, edit and share different watchlists.

ADVANCED

1. Play trailer if available
2. Users can add reviews for movies that should be visible to other users
3. Users can sort according to various attributes (like genre, year of release, etc)
4. Users can search for other users by username or user-ID
5. Users can send friend requests to others.
6. Users can choose to make their watchlists public or available to friends only.
7. Add reset option (Delete all user usage data) to have a fresh start.

CHITRA MAKER

Difficulty : Intermediate

★★

As we know all the students of prodigy classes are preparing well for JEE Main & Advanced , but it's time to fill JEE Main form so they need a concise photo to fill out the form. but they don't have proper image editor to make a photo concise so they asked jeetu bhaiya to help them, so jeetu bhaiya decided to make a image editor for all the prodigy students , but jeetu bhaiya is busy in teaching physics so jeetu bhaiya wants you to make an image editor.

BASIC

1. The Chitra maker should be able to open an image from a system
2. After editing, the user should be able to save the image into the system.
3. Able to zoom in, zoom out, undo, crop.
4. Able to scale the image.
5. Able to add text and emoji on image
6. Basic adjust (brightness, contrast)
7. Add area selector (lasso, rectangle)
8. Add brush and colour selector
9. Users can create a collage of multiple images
10. Transform image - Rotate, Horizontal/Vertical Inversion
11. Users can add frames

ADVANCED

1. Add filters (raining, sunshine, smoothing)
2. Red-eye correction feature.
3. Compress the file size of an image.
4. Add background changer and frame selection for an image
5. Add functionality of layers Preserve ratio by blurring sides. Example - https://drive.google.com/file/d/1Qr7IOV9XrPuSh7S1BzVrGuhVcxUuB7Sd/view?usp=sharing

COVI-KATANA

Difficulty : Intermediate

★★

Maheshwari classes has started the new batch and it has decided to conduct the classes in offline as well as online modes. COVID has still not gone completely and students need to be vaccinated, so they can give their attention towards JEE preparation. For this, Maheshwari Classes decided to have vaccination camps, and for that, it needs a software where students can register and book vaccination slots. Help them to make such a software such that the new batch does not have to suffer the online classes as we do.

Beta hum toh yahaan tabse hain jab yeh Institute ke classes,
Online nahi Offline hua karta tha.

BASIC

1. Separate login for government officials and people.
2. OTP/Email based authentication
3. Show available slots by location, pin and area.
4. Users can add documents as their ID proof.
5. Users should be able to book open slots.
6. The vaccinator should be able to update the vaccination status.
7. Update on email/phone on vaccination.
8. Users should be able to download certificates.
9. Vaccination dashboard showing vaccination stats of every day.

ADVANCED

1. Charts showing area wise vaccination stats. (API can be used)
2. Charts showing current covid stats of each area with various filters. (API can be used)
3. Waitlist for vaccination in case slots is not available.
4. Show vaccination stats on Map. (API can be used)
5. Fetch nearest vaccination centre using location and display it on Google map. (API can be used)
6. Vaccination slots booking should be thread-safe (concurrency control).

Note: APIs should be used only to fetch the data.

OFFCODER

Difficulty : Intermediate

★★

While growing up Meena didn't have a steady internet connection and he was very much fascinated by the craze of competitive programming. He wanted to become a FAAD Coder. Now after the JEE exams are finished. He wants to develop an alternative to an online coding platform, which will help those who don't have a good internet connection. A platform in which one can download questions, set up contests, submit answers and later on when connected to the internet can see the final result. Will you be able to help Meena ???

Par *OffCoder* toh rating improve karti hai na?

BASIC

1. Users should be able to pick questions randomly as well as specifically from any one platform (codeforces, leetcode, etc) and be able to download it with its sample test case. (external libraries may be used for scraping as well as APIs.)
2. Users can schedule a contest with a chosen problem or randomly selected problem.
3. Users should be able to submit the code in various languages (like C, C++, Java, Python).
4. The platform should be able to compile the code and display the verdict against the sample test case.
5. When the user is connected to the internet the platform on its own should be able to submit the user code and give the user a final verdict on its submission.
6. Users should be able to make personalised lists for various questions that they have solved with solutions (e.g. 'My fav DP questions')
7. If an Editorial is published it should be available to the user.
8. The platform should have a history of submitted questions.

1. The external libraries used in the previous section should be replaced with your own scraper algorithm.
2. The platform should have its own editor with proper syntax highlighting, indentation, etc.
3. Users must get their projected new rating and rank for the offline contest based on the concerned platform's rating mechanism.
4. The platform should have an elegant dashboard for statistics of user's performance through graphs, pie charts, etc.
5. The platform should have an automatic download feature for favourite topics selected previously. The download should happen in a periodic manner whenever connected to the Internet.
6. The platform can suggest questions similar to the current question.

FAST FRENZY

Difficulty : Intermediate

★★

When Vaibhav got the lowest batch, A10 at Prodigy, Jeetu bhaiya had to help him. To test his skills and dedication, he has asked you to make a car game which he can give to Vaibhav to test him. Vaibhav's score in that game will determine which batch he goes to. Jeetu bhaiya has given you some specifications and features to keep in mind while making the game.

BASIC

1. Car is controlled by keys/mouse
2. Provision of power-ups (invincibility, gold rush etc.) and coins
3. Game speed increases as the game proceeds
4. Obstacles appear which need to be dodged, else health is reduced
5. Previous score history is stored for users
6. Login / Signup
7. Guest session system
8. Background music and sound effects
9. Changeable background
10. Changeable difficulty

ADVANCED

1. Provision of cheat codes
2. Purchasable car skins (from in-game currency, which is collected from games, like coins)
3. Background music can be controlled and changed
4. 2 players can play on a single laptop (this game mode will be called race, in which the person who reaches the finish line first wins. If the health of both players becomes 0, then the person who is ahead in the race wins.) with split-screen function. 1st player drives with WASD keys, and 2nd player uses arrow keys to drive. The first person
5. Car takedown possible in race mode - if player 1 hits 2nd player with a speed higher than a threshold, 2nd player's health is reduced significantly and he slows down, but 1st player doesn't lose any health. If speed is lesser than the threshold, player 1 slows down and player 2 speeds up a bit
6. Additional Power-ups for race modes (like Nitrous boost, missile launch, shield etc)
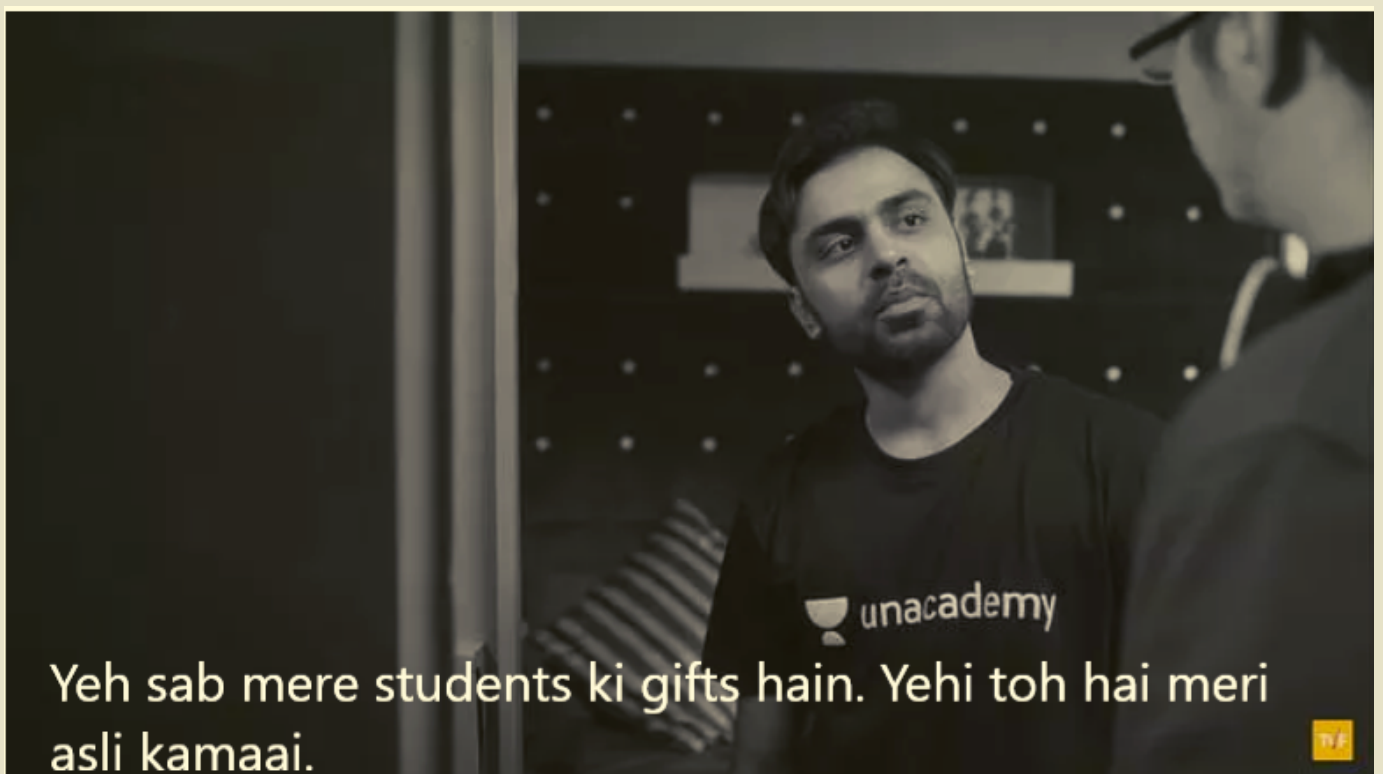7. Play with friends (LAN Play)

Note - Jeetu bhaiya loves good looking and creative UI. He will award you more for a good UI. Don't disappoint him.

FAIR AND SIMPLE

Difficulty : Advanced

★★★

Jeetu Bhaiya wants to conduct a practice exam in the online situation. But he knows that many of the students will resort to cheating in exams which will not be helpful for them at all. They may be able to get their batches changed in Prodigy but what use would that be for the real JEE examination. Help Jeetu Bhaiya in creating a fair online examination system so that his students can qualify JEE and reach great places. (After all, he doesn't want to lose the awesome gifts his students send him later ;-))

Yeh sab mere students ki gifts hain. Yehi toh hai meri asli kamaai.

BASIC

1. Different register and login portals for teachers and students.
2. Option to create classroom system (like in schools) / course wise team (like in colleges)
3. Make a proctoring system
4. Make a system for storing submissions by students

   Teacher Portal:

   a. Add new student, new course/ make a new class
   b. Add question paper (MCQ type) with exam time
   c. Send notification in the class of exam creation and 15 minutes before the exam starts. Take final attendance 10 minutes before the exam.
   d. Check if at least 1 proctor is available 15 minutes before the exam for each paper

   Student's Portal:

   a. Display schedule of all of the exams of that particular student
   b. Display timer during the examination
   c. The application should go full screen when the exam starts. If any other application is open in the background, notify the student and don't allow to give the exam till all applications are closed.
   d. Students should be able to see their checked submissions along with grades/marks.

ADVANCED

1. Question Paper can hold any type of question subjective/ one word/ order oriented/ diagram type
2. Conduct multiple exams at the same time
3. Allow students to clarify their doubts after the paper on a comment section type area
4. In case of any network or other kinds of issues from the student's side, the proctor should be able to manually make the required changes.

CODEDOC

Difficulty : Advanced

★★★

Vaibhav and Vartika are in the same college. Vaibhav and his friends are good at coding and we know that Vaibhav is always ready to help Vartika but due to some college restrictions they cannot meet after 8PM. Vaibhav makes a call to Jeetu Bhaiya and asks for his help and as always Jeetu Bhaiya comes up with a great idea of code collaboration platform. As Vaibhav is busy in practicing DSA, he won't be able to develop this platform. So, can you join him as Uday, Meena, and Meenal and help him make this collaboration a success? So come up with your squad and gear up for this collab and and take up Jeetu Bhaiya's advice on "21 din ki mehnat".

BASIC

1. Users can signup/login/logout.
2. User should be able to fetch previous records
3. Users can create/edit/save CodeDoc.
4. Users can allow other users to edit/see in their CodeDoc and after giving permission can block the user from further updating the code.
5. When any user updates in the CodeDoc the same change must be reflected in every other user's window, working on the same CodeDoc.
6. Coding area should have highlighting/autocomplete/recommendation according to language. (Library)
7. Should allow writing/compilation in at least 4 languages.
8. Coding area should show line numbers.
9. Users can run programs by providing input and can have output.

ADVANCED

1. Recommendation feature should be implemented by own. (Can use trie for optimization)(In any one language)
2. Separate Chat section where users can send/receive messages both privately and to everyone.
3. Coding area features in point 6 of basic can be implemented from scratch = Extra marks.
4. Write cursors for users should display their respective names (like Google Docs).
5. Audio-video communication between users (Can use external libraries for this, e.g. WebRTC)
6. Multiple tabs for each session.

CONCERTO VIRTUALE

Difficulty : Advanced

★★★

Vaibhav and all his friends got 10 days of holidays before their reshuffling test. And since Jeetu bhaiya told them "Thoda battery charge karlo, padhai ke sath sath thoda relax karoge to burn out nahi hoge", all of them decided to host a musical evening. But since Vaibhav, Meena and Uday are not allowed to go to girls PG, they decided to host this concert virtually. Help them bring out the hidden 'Bathroom singer' in them!

BASIC

1. Users can live record their music and save it on the local machine and on the server.
2. Users can upload an already recorded music file in the software.
3. Users can generate new music using virtual versions of various instruments within the software. (Bonus points for implementing more than 2 instruments).
4. Users can do basic editing on the audio file like audio mixing, apply audio effects, cut/paste audio parts, amplify, etc.
5. Users can signup and login.
6. Users can share their audio files with one another while online.
7. Users can freshly download the edited audio file or replace the original file.
8. Spectators can be allowed when a solo concert is being held.

ADVANCED

1. Users can apply special effects on the music file like reverb, noise cancellation, silence removal, etc.
2. Users can share their audio files with one another and the file uploads and downloads in the background even when the software is not in use.
3. Multiple users can create a room and play their musical instruments live together.
4. Cheering, clapping options can be made available for the spectators.
5. Judging and scoring options can be made available for music competitions.
6. Spectators can be allowed when a group concert is being held.
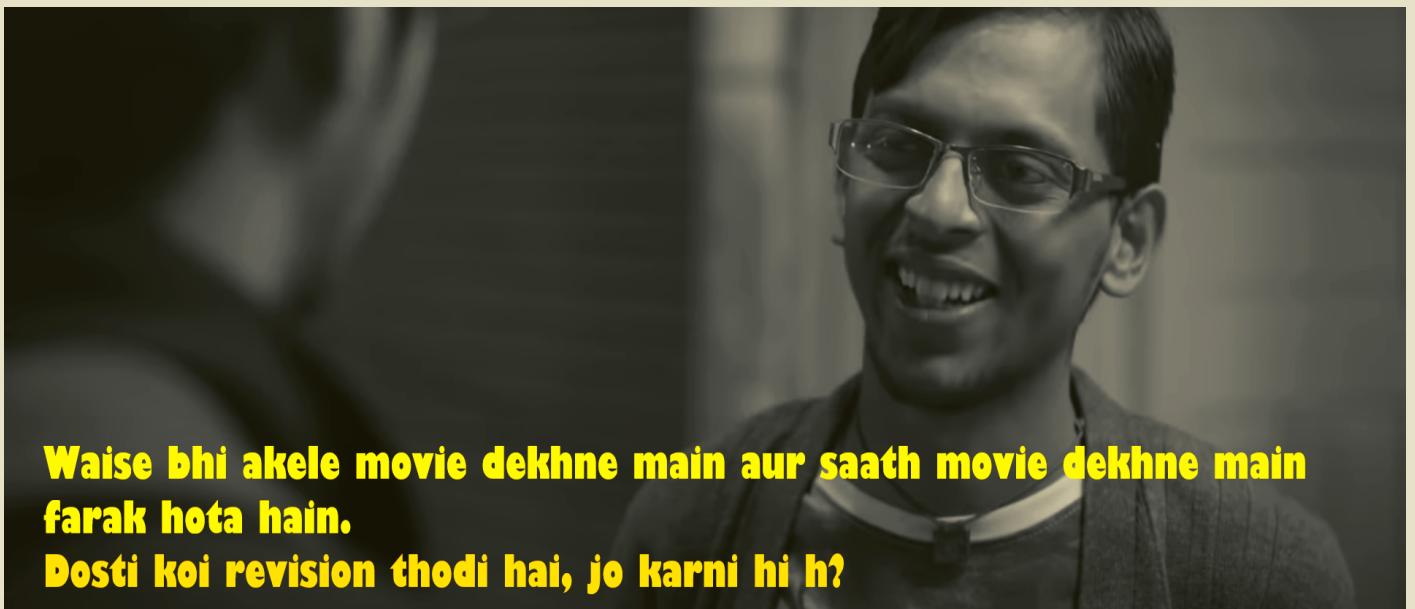
BUDDYFILM

Difficulty : Advanced

★★★

All work and no play makes Vaibhav and friends a dull squad.

A day spent skimming through the hectic classes surely drains the energy of our squad's fellas. So, what better than binge watching some crazy drama Or intense thriller Or just light hearted rom coms with the homies?

We are sure you all would love the idea of binge watching shows together with your squad just like Vaibhav and friends do, after a day consumed by online classes and keeping a keen check on maintaining one's attendance!!!

So, here's the challenge: pull in your Uday, Meena, and Meenal and make a P2P desktop app to watch movies together!

Waise bhi akele movie dekhne main aur saath movie dekhne main farak hota hain.
Dosti koi revision thodi hai, jo karni hi h?

BASIC

1. Users can log in/log out and have their respective app profiles (Username, Real Name, Email, Bio, PFP).
2. Users should be able to send friend requests to others. Only 'friends' can watch each others' streams.
3. One can search other users by their usernames and/or real names.
4. A user should be able to start a group movie session by selecting a movie from their system and streaming it.
5. Other users should be able to join the stream through a 'Join Stream' option present in the streamer's profile page (provided the streamer is a 'friend').
6. Both streamer and joined users should be able to play/pause the stream. Only streamers can stop the stream.
7. Basic chat feature both outside and during streams, with one friend or group of friends.

ADVANCED

1. More feature-rich chatting during viewing sessions, with timestamps and emoji reactions.
2. All users can additionally seek through buffered video (seek for themselves or seek for all). Seek can be done through seek bar or timestamps. [Seek for all would be basically pausing the video for all, seeking to position by self, then sending timestamp for position to all in chat]. Also, store the seek position of streamer (current live seek)
3. Add a role-based playback control in streaming sessions, where only people with a specified role can play/pause or seek.
4. Users should be able to communicate through audio (mic)
5. Users can mute their audio
6. One can also choose to disable another user's audio for themselves.
7. Option to disable audio communication when the stream is playing. (Available to streamer only)
8. Users can allow the app to scan through the user's movie library and make it available for display on their profile page.
9. Users can request their friends to start streaming a specific movie and send a notification to them.
10. Users can schedule a movie stream at any time and share its notification to all friends or a selected group of friends.
11. Users can choose to generate a secret code for a movie stream which can be entered by others to directly join it.

Libraries allowed for use:
Xuggler, Webcam Capture API (https://github.com/sarxos/webcam-capture), OpenCV, Vlcj
The streaming algorithm should be built from scratch. Any external libraries used for streaming (WebRTC, pre-built RTP libraries) would result in deduction of marks. [Well-synced audio and video would fetch higher marks. Use some syncing techniques for the same.]

OPEN THEME

Reached till here? None of our problems was appealing enough?

Go ahead pick something from around you and surprise us with what you can brainstorm. You are free to choose any idea and build a software around it.

But wait! That's not all, once decided(i.e. submitted in the abstract) you have to follow through with that idea and surprise us with what you build.