|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Idea 1** | **Idea 2** | **Idea 3** |
| Jorge Blanco Rey | The software for having classes must let the organizer have a total control of the “room”, like managing participants, mute them…. | ~~When a participant asks for a meeting filling up all the required fields, a notification must be sent to all students of the group, so they get informed, instead of having to enter the app.~~ | ~~The software should display all the free students (students that does not have team yet), so any other student can select him to send an invitation.~~ |
| ~~Pablo 2~~ | ~~Each room may have different profile settings, such as, for a conference room all participants are allowed to open their mics and for an exposition room only the how is exposing is allowed to open his mic.~~ | ~~Only “instant” notifications are sent, no email ones, because they are annoying.~~ | The application may include a student’s status which can be set by the student. |
| Rodrigo 3 | Having a “ride” feature that the organizer of the call can use to move all the people and himself to another meeting without the participants needing to do anything. | ~~Advanced filtering for the notifications, only allowing certain words or from certain groups.~~ | Automatically check if both student’s timetables are compatible, so that they can attend the same practice group. |
| Angel 3 | To have a system that interprets the meetings like files so each student is an identifier that can be moved from one file to another. | ~~An auto filling system that enrols the student in all his subjects just introducing his course number.~~ | ~~Doesn’t show the students that already have a pair in the selection area~~ |
| Pablo Soetard | The application must have a file sharing system to share files among students and teachers. | It must have an intuitive and modern UI with as little bugs as possible. | It may add a statistics/analytics section where the teacher could see the performance of each student |
| Jorge 2 | The file sharing system must have the possibility that students can edit the document at the same time. It has to have also control version, so they can undo the changes of the document up to one point. | The notification to the meetings, the participant can select if he wants them to be sent by email, sms or other option. | Statistics should be detailed, showing the time each participant has been in the meeting, time with the micro opened, for example… |
| Rodrigo 4 | You can do branches to the document, like gits. | Allowing the user a more advanced UI or simpler, depending on how knowledgeable he is. | The gamification of those statistics give points for more participation, or more efficiency. |
| Angel 4 | ~~You can send draft deliverables to keep it in the app but change the delivery after that. (Always before the deadline)~~ | Use some little gifs or hover elements to lead the user in the app. | The student could generate reports of his current marks. |
| Angel  Casanova | The application should have a high-resolution video, because otherwise it would be impossible to read a shared screen. | It has to take care of the delay in order to provide a face-to-face conversation felling. | It should be user friendly, we should provide an agile implementation for saving time to the users . |
| Rodrigo 2 | It needs a variable bitrate, so that it has good voice quality and video when internet is good and can work on slower connections too. | It should use some kind of voice filtering. | Advanced search. |
| Pablo 3 | Allow input video from other sources than the camera, ie: other screens or mobile phones | For more informal meetings voice filters such as movie character voices may be funny | Anti-cheat mecanism to avoid copies. |
| Jorge 3 | After each meeting, teams will have the option to fill up a template where they can write which objectives must they accomplish before the next meeting. The points discussed that was already entered when the meeting was created, will already be filled in the template and it could be modified. The minutes of the meeting ad participants is also filled automatically. This information could be also downloaded in case it is needed to deliver it. | Students should have the option to put some filter on the background when they activate their cameras during meetings. | When creating a meeting, if someone in the team is occupied (it already has a meeting), the application will tell to the student who is trying to create the new meeting that X student is not available on that date, and that he/she should talk with him/her or try another date. (Students can ask for changing the date or deleting of a meeting in case they cannot attend or at the end it is not necessary. All students must confirm for the first case) |
| Rodrigo Juez | Make available an API so that students can practice programming and automate tasks. | Version control on deliveries so that when you upload an assignment and then correct things the teacher can see your progress. | Being able to control other people’s computers during calls and meetings. |
| Angel 2 | Make a Bot script that gives feedback about the status of the uploads. | Add a repository system to the page, so you can edit your deliveries anywhere. | Make an VPN in order to share content among students in the easiest way possible. |
| Jorge 4 | In code assignments it could be a progress bar that can tell which percentage of the work has been already done. It could be filled by having each point of the statement in a checklist, for example. (example: Integrated Trello) | Whiteboards used during a meeting are saved automatically and can be checked already after finishing the call. | If the pre-established minutes for the meeting are about to finish and more time is required, this could be set (specifying the additional minutes) as many times as needed. |
| Pablo 3 | An auto checker script could be implemented by the teachers to allow the student to run their code against it and retrieve the number of use cases that their code has successfully passed, with or without knowing which cases where those. | Integrated “post-it" technology to allow students to plan their work with software development methodologies such as SCRUM (integrated ideaFlip) | Create a virtual machine environment where students can run test and code simultaneously. |