

REDUCING EDUCATIONAL APP FRAGMENTATION: A COMPREHENSIVE STUDENT PLATFORM WITH AI AND PEER COLLABORATION

Meihan Liu ¹, Yu Cao ²

¹ Westridge for girls school, 324 Madeline Dr, Pasadena, CA 91105

² California State Polytechnic University, Pomona, CA, 91768

ABSTRACT

CCampus's goal is to provide easy access education to students across the world. Modern day apps usually focus on something specific, such as: a calculator app, a calendar app, a note sharing app, etc. While these specifics are helpful to students, they may also be a hassle to find and may also apply a financial burden on students. CCampus that focuses on combining all the necessities a student needs into one app. Instead of downloading ten apps for your daily student life, you can download CCampus—an all in one app for students of all ages[1]. Furthermore, CCampus supports friendship and peership. By utilizing our note feature, students in CCampus can help each other while sharpening their own skills. The three key systems in CCampus are: AI Chat, Calendar, and Note sharing [2]. While features such as the calculator and the profile are important, AI Chat, Calendar, and note sharing are the core features to this app. The AI Chat allows students to receive quick and precise answers from an AI while the note sharing feature lets students work together and discuss a question from another peer. The calendar feature helps students manage their time. During the design of CCampus, we had multiple problems regarding the details of a page. When looking at other apps, they usually contain vibrant colors and images, but upon discussing the idea of our app, we decided to go with a simplistic design. Our simplistic design makes it easy for students to stay on task. In the future, CCampus is planning on expanding the specifics while still maintaining a clean design. We plan to implement a more advanced AI, a mentorship between students, a customizable home screen, and online courses [7]. These functions may help students learn better, allowing them to learn one-on-one with each other or with a course. The customizable screen implements an element of fun within the app.

KEYWORDS

Education App, AI Learning, Peer Support, Productivity

1. INTRODUCTION

Free apps for student life have always been a problem for me and my peers. Many academic apps require payment, thus inaccessible for students who are struggling financially. Money should not limit a student's access to knowledge and potential to learn.

Education is a right, but UNESCO's most recent statistics show that almost 250 million minors do not have access to education which means almost 11% of minors worldwide do not have the access to a basic human right.

David C. Wyld et al. (Eds): CRYPTIS, CBIoT, CAIML, NLCA, NC, WiMo, ICAIT, ICDIPV, ITCSE – 2025

pp. 325-336, 2025. CS & IT - CSCP 2025

DOI: 10.5121/cs.it.2025.151725

I recognized this recurring problem in school. I was unable to find any app that provided free and high quality educational tools. Multiple of my peers also struggled with this problem [8]. As a result, I used my programming skills to make an app that caters to students of all financial backgrounds. With the costs of tutors skyrocketing, getting immediate assistance on a subject can be difficult. CCampus isn't just an educational app, it is an app with community and one that allows you to learn from and teach others. In CCampus, students can exchange knowledge on multiple subjects, allowing the overall community in this app to prosper educationally and socially.

All three of the apps mentioned in Section 5 are all very specific apps. They all focus on one topic: Task/Time management [3]. These three apps all seek to help students through their school life. All the three apps we compared tended to focus on one specific area, they had multiple features and all different ways of approaching this idea. Although all three of the apps showcased precision, thoroughness, and care in handling the specifics, they all shared a common trend: inconsistent and difficult to navigate UI [4]. When handling a specific topic such as Task management, ideas can often overlap and become unnecessary features. The inconsistent UI combined with the feature overlap can make students using it become confused and frustrated. CCampus's task management handles the specifics while also ensuring its features are simple and straightforward without much clutter, allowing students to easily navigate the app without much confusion. Along with the simplicity, CCampus also provides a guide for new users, showing them how to use the app.

CCampus offers an accessible education app catered towards students which promotes structural learning and building community. Nowadays, many educational apps demand payment for use and contain a strict structure, only allowing students to communicate with AI or verified teachers. CCampus strives to allow access to quality learning without limiting those who do not have the funding to buy the app. Instead of only communicating with an AI or adult, you can communicate with your peers [8]. CCampus's note sharing feature lets students get feedback from fellow students along with allowing students to work with one another on unique projects and concepts. Other solutions towards this same problem usually contain too much filler or ads along with a lack of resources for students. CCampus is a simplistic app which focuses on implementing the necessary resources for students. Unlike many other solutions, CCampus provides two ways to solve your problem.

While still allowing AI help, CCampus focuses on peership, fostering communication between students.

I did a survey that asked 15 college students and 5 highschool students a list of questions regarding how CCampus has benefited their student life. I wanted to test if my app was truly effective. First, I found the students I wanted to survey, I then compiled a list of questions that covered the accessibility, functionality, and effectiveness of the app. Some of my most significant findings for my experiment was that my app overall helped participants with their academics, however, my app lacked the functionality I thought it had. My calculator did not provide the necessary mathematical functions an average student needs for higher level math. My calculator focused more on basic calculation rather than complex ones which caused the participants to prefer physical calculators over the built in one. Although my calculator lacked complexity, the scores overall for my app remained positive, with most participants agreeing that they would recommend this app to their peers. My questionnaire was successful, it gave me feedback on what I could improve and also showed me that my app was indeed successful, fulfilling its intended purpose. The 20 overall students scored my app an average of 3.895, illustrating how CCampus makes a positive impact towards student life.

2. CHALLENGES

In order to build the project, a few challenges have been identified as follows.

2.1. AI-Powered Learning with Peer Sharing

Our app focuses on simplicity and accessibility. We offer AI chats where users can quickly get answers to their questions. One skepticism could be that AI chat prevents the students from connecting with each other, which is wrong. Our app, while offering students the choice to obtain answers from AI, also contains a notes sharing platform that gives students the opportunity to share their knowledge and learn from other people. Students are able to share their notes with title, text, and images. They can also view and comment on other people's posts if they find others' resources helpful.

2.2. Smart Learning and Task Management

Our app also offers task management where users are able to add tasks and manage them inside our app. One skepticism could be that our app doesn't offer a way for the user to efficiently manage their tasks, which is incorrect. Users' tasks can have different repeat types, from every day, every week, every month, to every year. To view their pending tasks, users can go to the calendar page which contains an interactive calendar displaying all of their tasks. They can easily see how many tasks they have today and for the upcoming days. They can easily mark them as complete or delete those tasks if finished.

2.3. Tag-Based Learning and Note Sharing

Our app offers users the ability to post their notes for other people to learn from. One skepticism could be that the users would have a hard time finding relative posts among all of them, which such an issue does not present in our app. When posting their notes, students need to go to a specific subject's note forum, like "Math", "Science", "English", etc. In addition, when adding their notes, the user will also be prompted to select a couple subcategories inside the subject. For example, if the user decides to upload any notes about math, they can select from a variety of tags like "Algebra", "Geometry", "Calculus", etc. When viewing other people's notes, users can filter by these tags to easily find relative information and notes they are looking for.

3. SOLUTION

The first screen upon opening CCampus is the login/signup screen. The user should make an account using their email and make a custom password, if the user has an account, they will be automatically signed in upon opening the app. If in any instance the user is signed out, they can simply click the "sign in" button and use their email and password to log in. When opening the app with a new account, a tutorial will automatically initiate. The tutorial introduces you to all the features within the app and how to use them. Campus contains five sections: a calculator, a calendar, an AI assistant, note sharing function, and a profile page [5]. The calculator provides the user quick access when they need to solve math problems so that they don't have to pull out their physical calculator. The calendar helps users manage their day. Users can simply add a task and set a "due date/time" and receive a notification once the task is due. The AI assistant provides accurate and concise answers to whatever question the user may have. Note sharing allows users to share their notes with one another and improve together. When opening the note sharing page, you will be introduced to different topics such as: Math, Art, English, etc. Upon clicking on these topics, you can use a search filter to search for the notes you need. By using the

comment section inside of each user's note post, you can give feedback or submit questions. The profile page is a page where you can manage your name, description, and a profile picture.



Figure 1. Overview of the solution

The tutorial's purpose is to introduce the features of the app to new users and how to use those features. The tutorial sessions will only appear the first time the user enters the page. The sessions consist of multiple instructions that can be skipped by users if they want to.



Figure 2. Screenshot of the APP

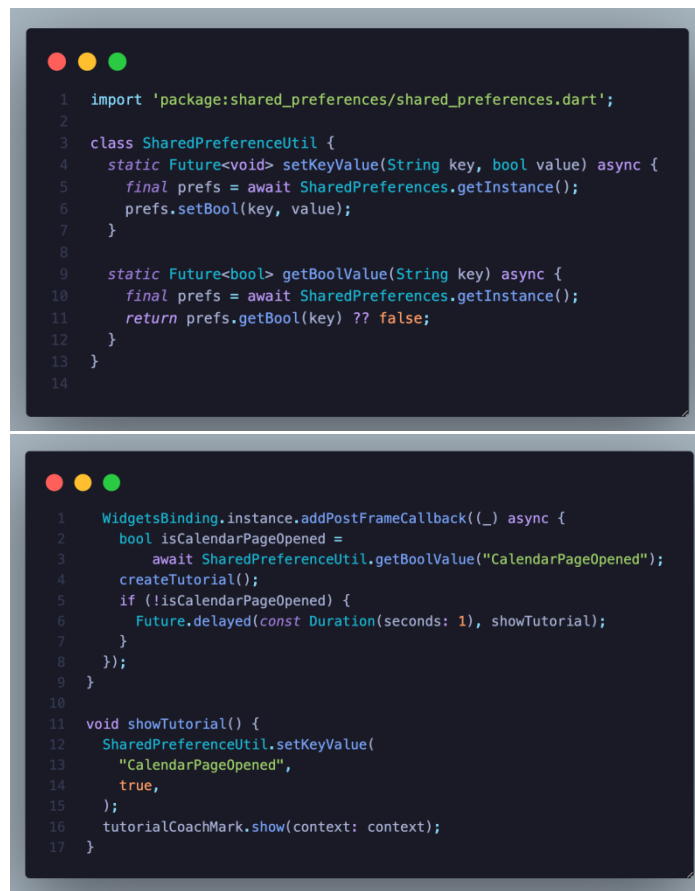


Figure 3. Screenshot of code 1

The first code block displays a `SharedPreferenceUtil` class which is a helper class for `SharedPreferences`, a flutter package for local storage. Because we only want the tutorial page to show up the first time the user enters the page, we need a way to store whether the user has first entered the page inside the device. A local storage solution is perfect for this type of need. The `SharedPreferenceUtil` class contains two functions: “`setKeyValue`” and “`getBoolValue`”. These functions are triggered in the page in the second code block. Then when the page starts, we check if the “<pageName>opened” key exists in the local storage. If so, the user had already visited the page before, otherwise, display the tutorial and mark the “<pageName>opened” key as true. We also want to delay one second before we show the tutorial to ensure that the page itself is fully built, thus no abnormalities in the UI are displayed [6].

The calendar's purpose is to help users organize their tasks. After creating the tasks, the users will be able to view the tasks on the calendar to get a better overview of their tasks. The tasks are also repeatable including repeating every day, every week, every month, and every year. This code block handles displaying tasks on the calendar.



Figure 4. Screenshot of the calendar

```

1  eventLoader: (day) {
2    final tasks = snapshot.data!.docs.where((element) {
3      final dueDate =
4        element["dueDate"].toDate() as DateTime;
5      final repeat = element["repeatFrequency"] as String;
6      if (element["completed"]
7        .contains(Timestamp.fromDate(day))) {
8        return false;
9      }
10     if (repeat == "Every Day") {
11       return day.isBefore(dueDate) ||
12         isSameDay(day, dueDate);
13     } else if (repeat == "Every Week") {
14       final curWeekday = day.weekday;
15       final dueWeekday = dueDate.weekday;
16       return (day.isBefore(dueDate) ||
17         isSameDay(day, dueDate)) &&
18         curWeekday == dueWeekday;
19     } else if (repeat == "Every Month") {
20       return (day.isBefore(dueDate) ||
21         isSameDay(day, dueDate)) &&
22         day.day == dueDate.day;
23     } else if (repeat == "Every Year") {
24       return (day.isBefore(dueDate) ||
25         isSameDay(day, dueDate)) &&
26         day.month == dueDate.month &&
27         day.day == dueDate.day;
28     } else {
29       return isSameDay(dueDate, day);
30     }
31   }).toList();
32   return tasks;
33 },

```

Figure 5. Screenshot of code 2

In the function, we need to return a list of values which the length of the returned list will be the number of dots on the calendar for that particular day. The code processes every task and checks if they will need to be displayed. This is done by the `where()` function which serves as a filtering mechanism.

We first extract the `dueDate` and `repeatFrequency` from the task for later use. We then check if the task is completed, and if so we return false because we do not want to display completed tasks. We then handle the repeat frequency separately. If it repeats every day, we only want to display the task if it is before the due date. If it repeats every week, then we also need to check if the weekday matches. Similarly, if the repeat frequency is every year, we would check if the day matches. For example, if the task is created on the 10th, then the task will show up every month on the 10th. Lastly, for tasks that repeat every year, we only return true if it is before the due date, and day and month also match.

The calculator serves to help users make fast calculations regarding basic math. This function allows people to have an easy everyday calculator in their phone without needing to carry one around. You can use the calculator to perform calculations with addition, subtraction, multiplication, division, and percentages.

The calculator logic includes parsing the user inputs into an abstract syntax tree and evaluating the values separately [9]. It is also challenging allowing user to input values by clicking buttons on the screen as the value inserted needs to be at the right place. The code sample below shows how that is implemented.

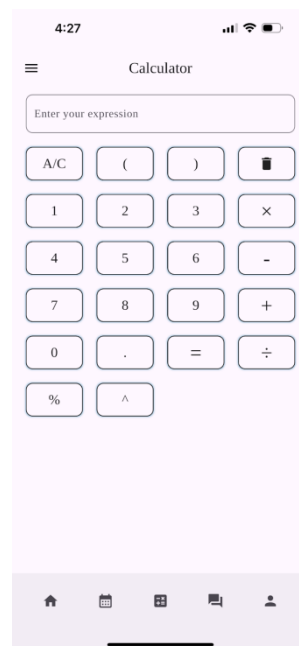


Figure 6. Screenshot of calculator



```

1 void insertCharacter(String character) {
2     final text = calculatorTextFieldController.text;
3     final selection = calculatorTextFieldController.selection;
4     if (selection.start == -1) {
5         calculatorTextFieldController.text = text + character;
6         return;
7     }
8     final newText =
9         text.replaceRange(selection.start, selection.end, character);
10    final newPosition = selection.start + character.length;
11    calculatorTextFieldController.value = TextEditingValue(
12        text: newText,
13        selection: TextSelection.collapsed(offset: newPosition),
14    );
15 }
16
17 void deleteCharacter() {
18     final text = calculatorTextFieldController.text;
19     final selection = calculatorTextFieldController.selection;
20     if (selection.start == -1) {
21         calculatorTextFieldController.text = text.substring(0, text.length - 1);
22         return;
23     }
24     final newText = text.replaceRange(selection.start - 1, selection.end, "");
25     final newPosition = selection.start - 1;
26    calculatorTextFieldController.value = TextEditingValue(
27        text: newText,
28        selection: TextSelection.collapsed(offset: newPosition),
29    );
30 }
31
32 void calculate() {
33     final input = calculatorTextFieldController.text.replaceAll(' ', '');
34     try {
35         final parser = ExpressionParser(input);
36         final ast = parser.parse();
37         final result = ast.eval();
38         setState(() {
39             calculatorTextFieldController.text = result.toString();
40             calculatorTextFieldController.selection = TextSelection.collapsed(
41                 offset: calculatorTextFieldController.text.length);
42         });
43     } catch (e) {
44         ScaffoldMessenger.of(context).showSnackBar(
45             SnackBar(content: Text('Invalid expression')),
46         );
47     }
48 }

```

Figure 7. Screenshot of code 3

The code block contains the three functions for the calculator: insertCharacter, deleteCharacter, and calculate.

The insertCharacter function is called whenever a button is clicked to input values. The function itself takes in a character (string type) as the input. We first get the current calculator field text, and then get the location we want to insert, and then we either add (selection.start and selection.end are the same), or replace the user selected text with the new character [10]. Then we update the value of the calculator's textfield with the new value.

Similarly, deleteCharacter function implements similar logic except we would use "-1" to signal deletion.

Lastly, the calculate function calculates the expression the user enters and displays it on the screen. It first removes any extra spaces the user might put inside the expression. Then it creates a parser to parse the expression into an abstract syntax tree and calls the eval function to obtain the result. It then proceed to update the values of the textfield.

4. EXPERIMENT

CCampus aims to improve student productivity by offering features like Task Management, AI Chat, Note Sharing, etc. It is vital to test how effective our app is in terms of helping students' productivity. In order to measure the increase in productivity, we created a survey and asked 15 college students and 5 high school students to inquire about different students on their experience using our app on a scale of 1 to 5, 1 being strongly disagreeing, and 5 being strongly agreeing. With the result of the app, we can test out whether our app is effective and integrate on the design and logic of the app.

1. Do you find the app easy to use?

2. Do you believe that CCampus has increased your timeliness
3. Do you find it easier to manage and view your tasks using the calendar page?
4. Do you find the AI chat useful in helping you answer questions in a speedy manner?
5. Does the note sharing function help you learn from other students?
6. Do you find the calculator app useful in replacing physical calculators?
7. Would you recommend CCampus to other people?
8. Does this app help you with your grade?
9. Would you upload your own notes on CCampus?
10. Do you believe this app is easily accessible?

We believe that these questions very well represent the aim of our app CCampus – “Simplicity”, “Accessibility”, “Education”, and “community”.

Questions	Answers
1. Do you find the app easy to use?	5 - 9 people 4 - 8 people 3 - 3 people 2 - 0 people 1 - 0 people
2. Do you believe that CCampus has increased your timeliness	5 - 6 people 4 - 5 people 3 - 7 people 2 - 1 person 1 - 1 person
3. Do you find it easier to manage and view your tasks using the calendar page?	5 - 10 people 4 - 4 people 3 - 5 people 2 - 1 person 1 - 0 people
4. Do you find the AI chat useful in helping you answer questions in a speedy manner?	5 - 3 people 4 - 5 people 3 - 7 people 2 - 3 people 1 - 2 people
5. Does the note sharing function help you learn from other students?	5 - 12 people 4 - 3 people 3 - 4 people 2 - 0 people 1 - 1 person
6. Do you find the calculator app useful in replacing physical calculators?	5 - 4 people 4 - 2 people 3 - 3 people 2 - 7 people 1 - 4 people
7. Would you recommend CCampus to other people?	5 - 16 people 4 - 2 people 3 - 1 person 2 - 1 person 1 - 0 people
Questions	Answers
8. Does this app help you with your grade?	5 - 5 people 4 - 7 people 3 - 7 people 2 - 1 people 0 - 0 people
9. Would you upload your own notes on CCampus?	5 - 4 people 4 - 8 people 3 - 6 people 2 - 2 people 1 - 0 people
10. Do you believe this app is easily accessible?	5 - 11 people 4 - 7 people 3 - 2 people 2 - 0 people 1 - 0 people

Figure 8. Table of experiment

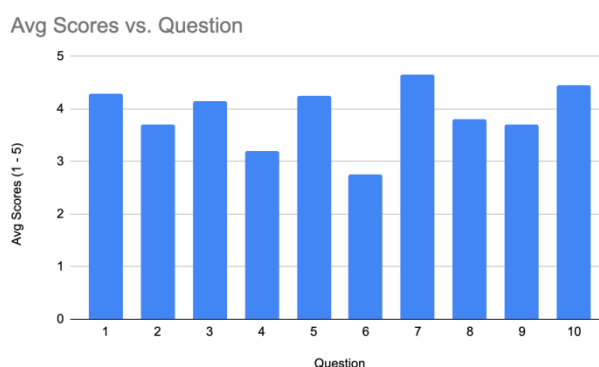


Figure 9. Figure of experiment

The question which showed the highest score was: “Would you recommend CCampus to other people?” The overall average for this question was 4.65, representing how most participants would recommend this to their peers. This data may also suggest that the app—overall, is well rounded and ready for basic student use.

The question which showed the lowest score was: “Do you find the calculator app useful in replacing physical calculators?” The overall average for this question was 2.8. Many participants disagree with this question mainly because our calculator lacks advanced mathematical functions used in high level math. Our calculator is structured towards basic calculations, leading many participants to prefer a physical calculator which has most if not all the mathematical functions a student needs.

Looking at the survey as a whole, we have an average score of 3.895, indicating that CCampus can indeed help students with their academic life. With functionalities like the calendar, AI chat, note sharing, etc, CCampus was proven to be effective in its original purpose — provide easy access education to students across the world.

5. RELATED WORK

Julia’s app, similar to ours, contains a task management system with calendar, task management, repeats, due dates, etc [11]. Different from our app, her app contains extra features like priority and tags. However, compared to her app, our app offers extra features to assist with student’s learning including AI chats and a notes forum. Moreover, we also offer a built in calculator to help with students’ productivity in school. While her app specializes in the task management area, our app provides a broader set of features more suitable for general student development. Furthermore, our app has a more consistent design with a consistent color and UI style overall.

Ronald’s app is also a mobile app built in Flutter that contains a student planner feature [12]. Both our app and his app contain a task management/planner feature that includes a calendar view. Ronald’s app contains a calendar heatmap feature which allows users to quickly view dates with heavy workloads. However, our tasks allow more customizability with features like repeating tasks. Our app also contains systems like notes sharing, calculator, and AI chat. Lastly, our app focuses on accessibility with a consistent UI layout style.

Ashwini’s “SnoopMe” app is built to help students manage their schedules [13]. CCampus and SnoopMe both seek to help students organize their schedule and time. Ashwini’s app contains

notification alerts, a timer, and task history. While his app is excellent for students who want to keep track of their tasks, CCampus provides notifications and set times. SnoopMe has a complex UI which makes it hard for users to see their tasks on a specific day. CCampus provides an adjustable calendar for users' tasks, allowing users to simply click on a day to view and complete their tasks. Ashwini's app also contains an unconventional design with confusing color choices and button layouts which may confuse users and lead to future frustration. On the contrary, CCampus's provides an easy-to-navigate and consistent layout across the entire app. Ashwini's "SnoopMe" provides detailed time management, but CCampus focuses on simplicity and accessibility overall while still containing the necessary tools for students.

6. CONCLUSIONS

If given more time to develop CCampus, I would add features such as: Online courses, customizable AI, Mentorship, and customizable GUI [14]. By implementing online courses, CCampus encourages independent studying amongst students. Students can find subjects that their school may not offer and can use our online courses to learn more about these subjects. Along with online courses, it would also be beneficial to include a more customizable AI. While our current AI can answer questions efficiently and properly, we recognize that many students may have a certain learning style that our current AI cannot adapt to. Ensuring quick and efficient responses is one of our top priorities. A core principle of CCampus is to ensure that students from all over the world can exchange educational information with one another, mentorship would allow students to work together and teach one another [15]. Customizable GUI would be a feature that allows students to customize the color palette for their app. It is a feature that can allow students to adjust the app aesthetic according to their likes. Countless unresponded emails and expensive online courses will no longer be a problem for students. CCampus enables accessible and effective learning while building community—it is an app which students can rely on to be their personal study buddy. In the foreseeable future, I would like to implement these features to CCampus in order to improve the quality of the app overall and to truly be a one-in-all app for students. If I were to start over from the beginning, I would have made a clearer layout for what I wanted and focused more on functions rather than aesthetics.

REFERENCES

- [1] Meurisch, Christian, et al. "Capturing daily student life by recognizing complex activities using smartphones." *Proceedings of the 14th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*. 2017.
- [2] Haristiani, Nuria. "Artificial Intelligence (AI) chatbot as language learning medium: An inquiry." *Journal of Physics: Conference Series*. Vol. 1387. No. 1. IOP publishing, 2019.
- [3] MahmoudZadeh, Somaiyeh, David MW Powers, and Reza Bairam Zadeh. "Mission Planning in Terms of Task-Time Management and Routing." *Autonomy and Unmanned Vehicles: Augmented Reactive Mission and Motion Planning Architecture*. Singapore: Springer Singapore, 2018. 55-71.
- [4] Rose, Meghla Mehnaz, and Masitah Ghazali. "Improving navigation for blind people in the developing countries: A UI/UX perspective." *Journal of Human Centered Technology* 2.1 (2023): 1-10.
- [5] Yin, Meng, Shiyao Jiang, and Xiongying Niu. "Can AI really help? The double-edged sword effect of AI assistant on employees' innovation behavior." *Computers in Human Behavior* 150 (2024): 107987.
- [6] Ferreira, Jennifer, James Noble, and Robert Biddle. "Agile development iterations and UI design." *Agile 2007 (AGILE 2007)*. IEEE, 2007.
- [7] Zhai, Xuesong, et al. "A Review of Artificial Intelligence (AI) in Education from 2010 to 2020." *Complexity* 2021.1 (2021): 8812542.
- [8] Kruger, Ann Cale. "Peer collaboration: Conflict, cooperation, or both?." *Social Development* 2.3 (1993): 165-182.

- [9] Privat, Michael, and Robert Warner. "Handling User Input." *Beginning OS X Lion Apps Development*. Berkeley, CA: Apress, 2011. 73-117.
- [10] Schelin, Charles W. "Calculator function approximation." *The American Mathematical Monthly* 90.5 (1983): 317-325.
- [11] Trujillo, Julia Castillo. "Designing A Time Management App For And With Informatics Students." Undergraduate Thesis, University of Edinburgh, Edinburgh (2021).
- [12] Mulero, Akeem. Technical Report: Requirements for a NCI Student Mobile App Development. Diss. Dublin, National College of Ireland, 2016.
- [13] Dalvi, Ashwini, and Irfan Siddavatam. "Snoopme-interactive task scheduler mobile application for students." 2019 International conference on nascent technologies in engineering (ICNTE). IEEE, 2019.
- [14] Haque, Md Emdadul, et al. "IoT based home automation system with customizable GUI and low cost embedded system." 2019 international conference on sustainable technologies for industry 4.0 (sti). IEEE, 2019.
- [15] Frankie, Gordon W., et al. "Education, information transfer, and information exchange." *Advances in urban pest management* (1986): 163-184.