

# **Android App Development Integrating UX/UI Design: Object-Oriented Computer Programming**

A Case Study Presented to

Mr. Leonardo Pabroquez

Department of Physical Sciences and Mathematics

University of the Philippines Visayas

In Partial Fulfillment of the Requirement

in CMSC 22: Fundamentals of Object-Oriented Programming

Erru G. Torculas

BS in Computer Science II

December 19, 2020

## Introduction

With the onset of COVID-19 pandemic, there has been a massive shift to learning. Students were left with their own devices to cope up with the online setting that does not only imply privilege but also the lack of support to those who does not have the capabilities. Hence, digital inclusion is deemed necessary for the education system to progress given that it is a fundamental right and prerogative to every individual for lifelong learning opportunities<sup>1</sup>. With that being said, we can't just turn a blind eye to those students who lag behind the current set up especially those who are residing in far-flung areas that can't be reached by digital and broadcast remote learning. Not to mention, the digital gap between students and teachers has been unprecedented which the solution aims to bridge that dilemma.

The Philippines' Department of Education (DepEd) calculated the student's enrollment for school year 2020-2021 in order to have a grasp of the adverse consequences of the phenomenon. The numbers showed 6 million of Filipino students from last year have failed to enlist<sup>2</sup> (Magsambol, 2020). Consequently, this puts the country to a huge decline in the education sector which indirectly undermines the access to quality education and thrive the existing inequality that the students are facing. Also, some of the students enrolled are forced to enlist knowing they don't want to be left behind with the current setting despite the lack of resources to sustain their selves (e.g. mobile data, phones, laptops, etc.). This backhanded narrative is an alarming issue that amplifies the inconsistency of the education system which only favors the few.

<sup>1</sup> United Nations report on their annual gathering.

<sup>2</sup> For more details about this report, proceed to the references

Furthermore, the Philippines is an archipelagic country where land masses are separated that formed urban and rural areas. Based on the Philippine Statistic Authority urban-rural classification, the rural population is comprised of 54.7% of the country's population<sup>3</sup>. Empirically speaking, that is quite a number of people living in the rural area which also boils down to their means of communication given that these areas are frequently experiencing unreliable connection. The Philippines is dependent on short message service or commonly known as SMS and it is still the viable messaging channel that can offer these areas (Gray, 2020). The prevalence of SMS usage in rural areas is still relevant up-to-date. Hence, the current set-up students who are living in far-flung areas lag behind in terms of mobile data connectivity.

Nevertheless, the solution will pave the way to current mishaps of the education system, if not, alleviate some of the students dilemma with regards to the current setup – online and modular learning. It's a no-brainer that we strive for the betterment of what's within our reach which ultimately suggests every individual to take part of making that aspiration feasible. The egregious shift to a new normal setting is unexpected and that alone should be a factor to thrive to aid those who are left behind. No one should be left behind.

<sup>3</sup> For more details about this report, proceed to the references.

<sup>4</sup> PromoTexter reports the SMS situation in the Philippines

## **Statement of the Problem**

This solution designs to further discuss the integration of UI/UX design through the development of the android application software.

Specifically, this solution seeks to answer the following questions:

1. What is the efficacy of the UX/UI in the user's end?
2. How does the android app development help integrating the UX/UI design?
3. What is the relevance of the SMS-based education platform to the performance of the user?

## **Scope and Limitations**

This solution aims to determine efficacy of SMS-based education platform. It is intended to give knowledge to the beneficiaries on how the students see the solution provided on the users-end.

The study will be conducted among primary students to secondary students in Miagao Central Elementary School with the cooperation of Department of Education Region IV-Milagao District. There will be a survey conducted wherein the researchers seek to find answers regarding the efficacy of the UX/UI design, run-time of the app, and the SMS-based platform per se.

Limitations the researchers will encounter are resources provided on the user-end. The solution is on its prototyping phase – testing phase. Resources such as sim card, keypad mobile phone and microcomputer will be limited ranging forty (40) to fifty (50) students. The solution is only intended to calculate the needed resources to deploy in the near future.

# Design

This chapter will tackle about the principles used, evaluation of patterns and the architecture with regards to the solution at hand.

## Design Pattern

### 1. Creational

- Builder – During the development of the app, builder pattern is evidently crucial to building a complex object using simple objects and approaching this on a step-by-step process. The graphical representation which is the UI design classes set to be created first and the flow proceeds.

### 2. Structural

- Bridge – The overall structure of android development completely relies to this design pattern which cultivates a decoupled abstraction from the actual implementation, hence, providing a bridge structure between these two entities in order to seamlessly work. For instance, the xml classes (graphical representation) needs decouple from with java code of the project in order to grasp the design of the user interface.

### 3. Behavioral

- Mediator – This pattern is used to decipher complex communications between multiple objects and classes. Not only has that, mediator classes normally handled these complex transmissions between classes. Consequently, maintenance eases for the code and algorithm by loose coupling. Android development is inherently dependent to this behavioral

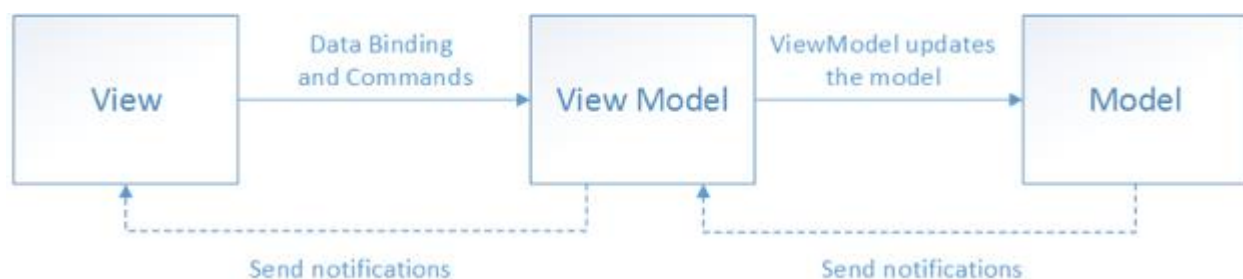
design pattern because of classes being linked with each other. Not to mention, user interface is necessary for the app, so it's essential to have a mediator class to grasp the complex communications that is happening in the algorithm.

## Design Architecture

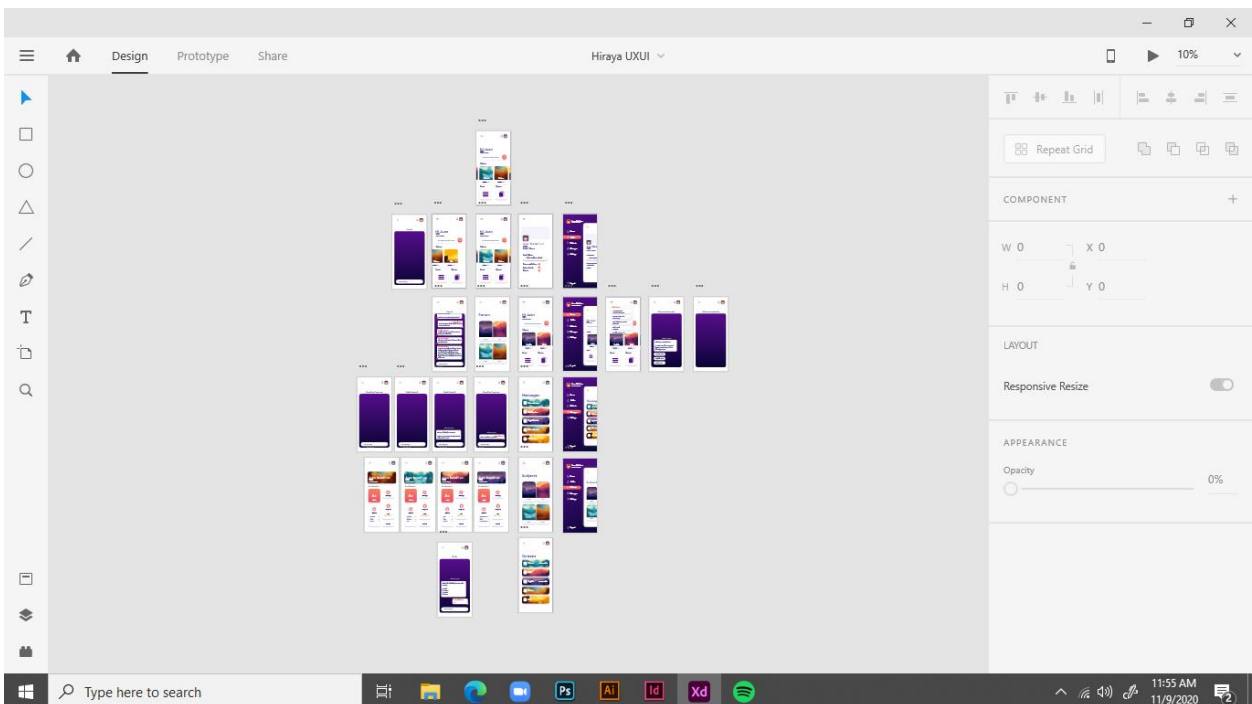
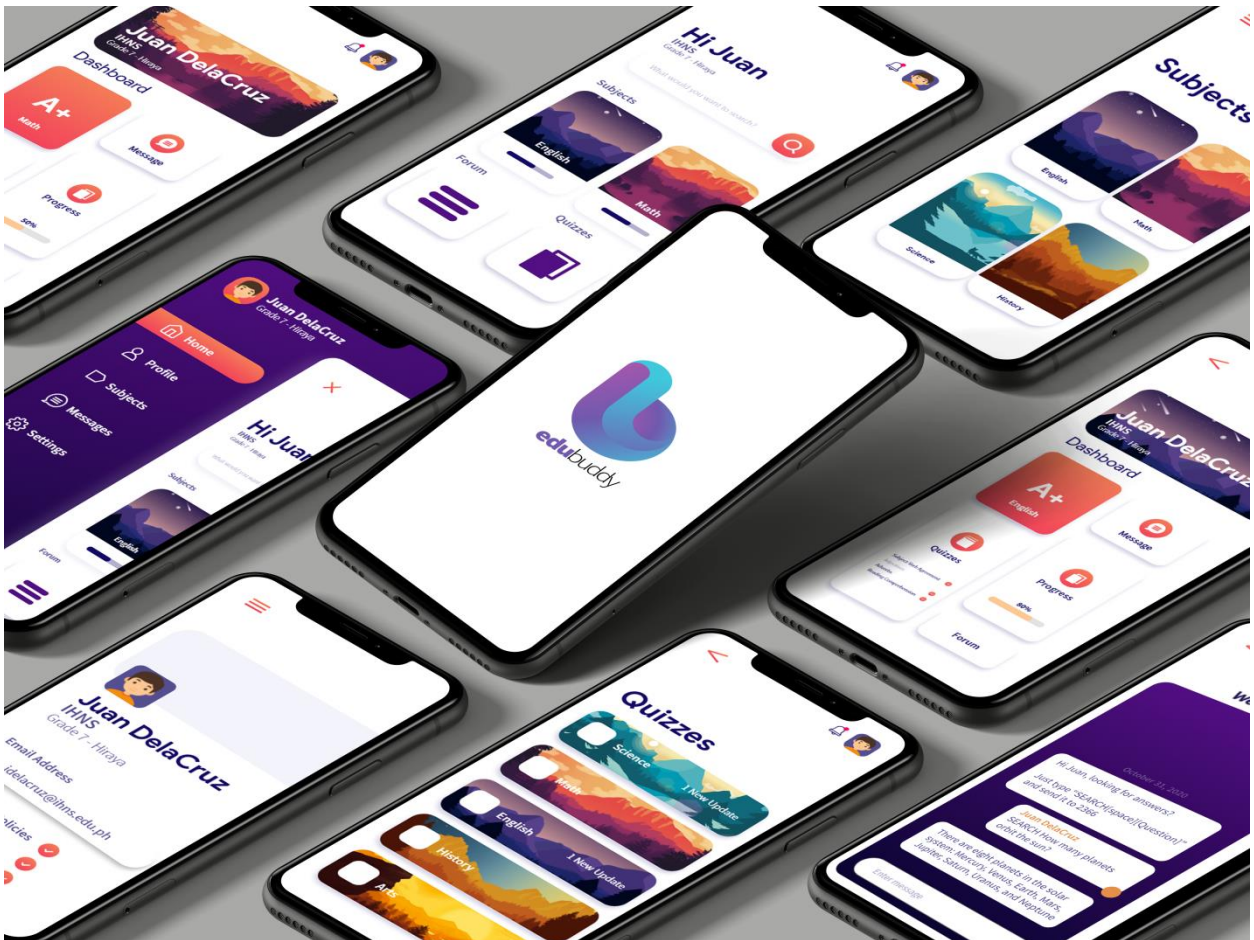
- Model-View-ViewModel Architecture (MVVM)

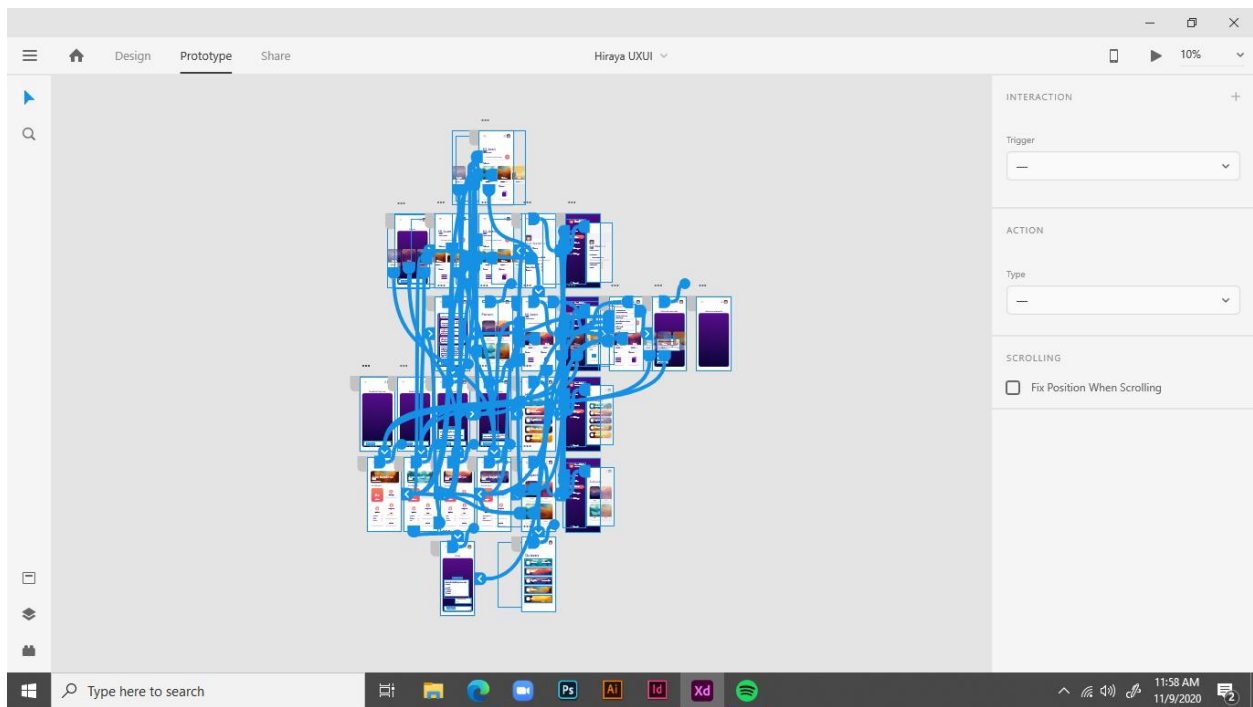
Android development adapts this design architecture because it allows a fast-paced reaction to the design changes. Not only that, it is also easy to test when test-driven-development comes in action. MVVM combines the advantages of separation of concerns while leveraging the advantages of data bindings (Hong, 2019).

Basically the MVVM architecture has three components at play and these are: View, ViewModel, and Model. The first component View is solely responsible for the overall structure of the layout displayed on the user's end. Also, UI logic is mostly executed in this part. Second, ViewModel is the data implementation and commands linked to the View to notify the changes in current state. Consequently, the View receives the message and applies the change. Lastly, Model is more of a non-visual component that has data that is executable.

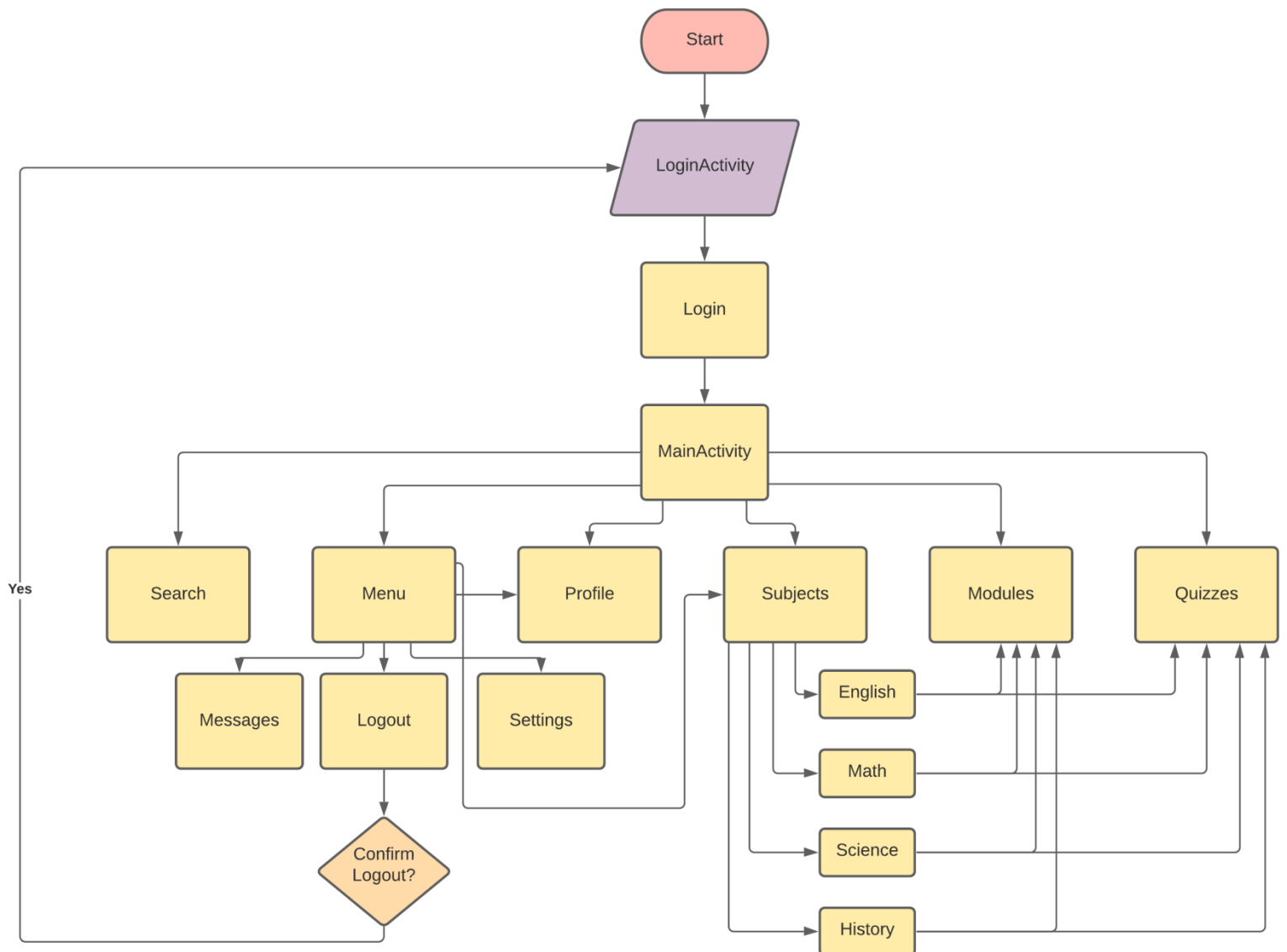


# UX/UI Design





## Flowchart Diagram





## **Development**

This chapter aids to guide the researchers for an in-depth process of the development and adhere to the experiences and discussion related to the android app development.

### **Front End Development**

The whole process of developing the front end is tedious due to some circumstances that left the developer clueless where to start. Not only that, wire framing of the UI design is the first hurdle to overcome – many decisions and hesitations came to play. As soon as the UX research was properly ironed out, following certain rubrics to be mindful, developing the rest will come at ease.

Furthermore, XML was used to develop the front end of the android. It provides a clear database of actions needed while not alienating the principles of design process and principles. However, it was challenging at first to explore the nature of XML and how it works since it is my first time dealing with it in order to delve in the development of application software.

### **Back End Development**

The responsiveness of the android was made possible of back end development which Java programming language's main function throughout the process. Needless to say, Java is the prominent player with regards to the functionality of the application software that brought every widget come to life and do certain actions that deems necessary to the development.

Moreover, the project is not fully optimized due to time constraints. However, the rest of the development relies to back end development to provide a cohesive communication between databases and servers.

## **Test-driven Development**

- Instrumented Unit Test / Instrumentation Test

The development solely focused on instrumented unit tests where it builds and runs through a physical device in order to check the responsiveness of the user interface. However, this test is slow compared to local unit test but it does give a significant evaluation of the app's behavior in the long run against actual device hardware. Basically, it emulates the actions of the end user using the app to further test what might not work, hence, failed test and configuring the code. In other words, instrumentation test provides a functional test approach, in effect; more user-oriented especially it deals with a real device to run tests.

Espresso and JUnit4 frameworks were the main player to TDD of the android app. specifically; Espresso tests the state expectations, interactions, and assertions in a cohesive manner alienating the messy implementations in the background. Not to mention, it runs the UI test optimally fast with ease. Consequently, it simulates the test of the app's behavior through interactions between the view and view model. Basically, it performs UI test in order to grasp the logic, configurations, and limitations behind it.

## **Recommendation**

Based on the observations drawn in the solution during the development of the app, the following are hereby recommended:

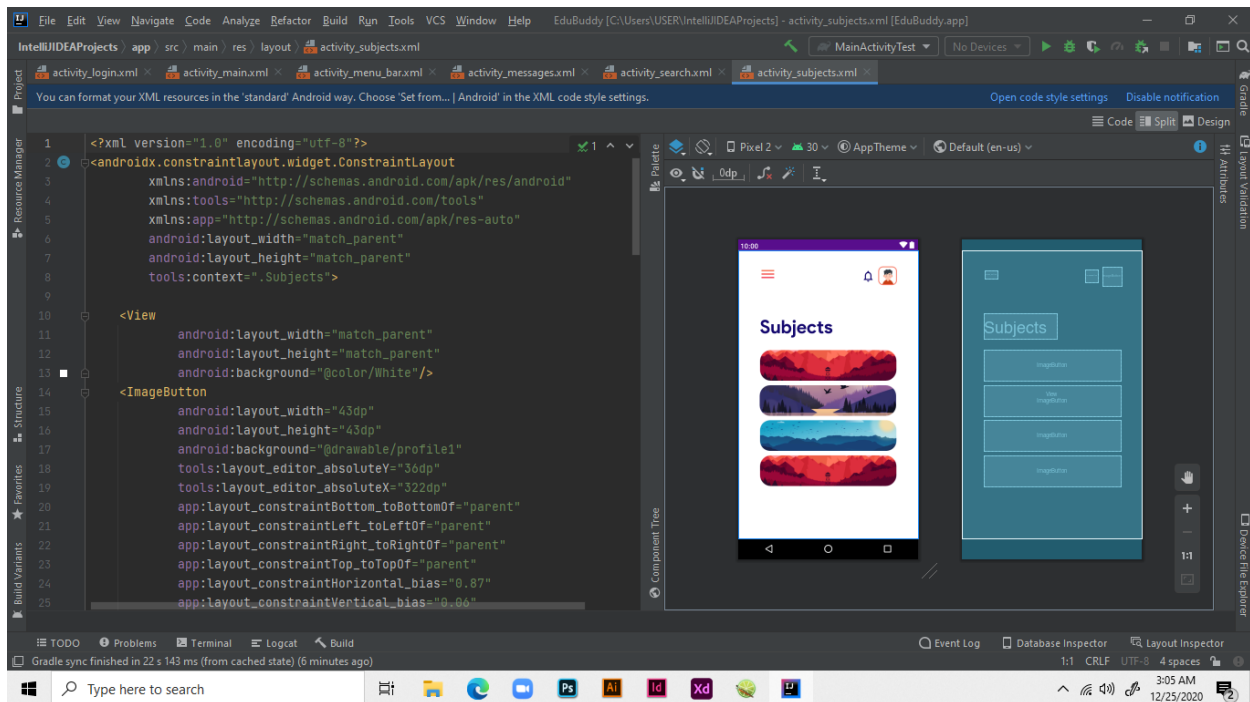
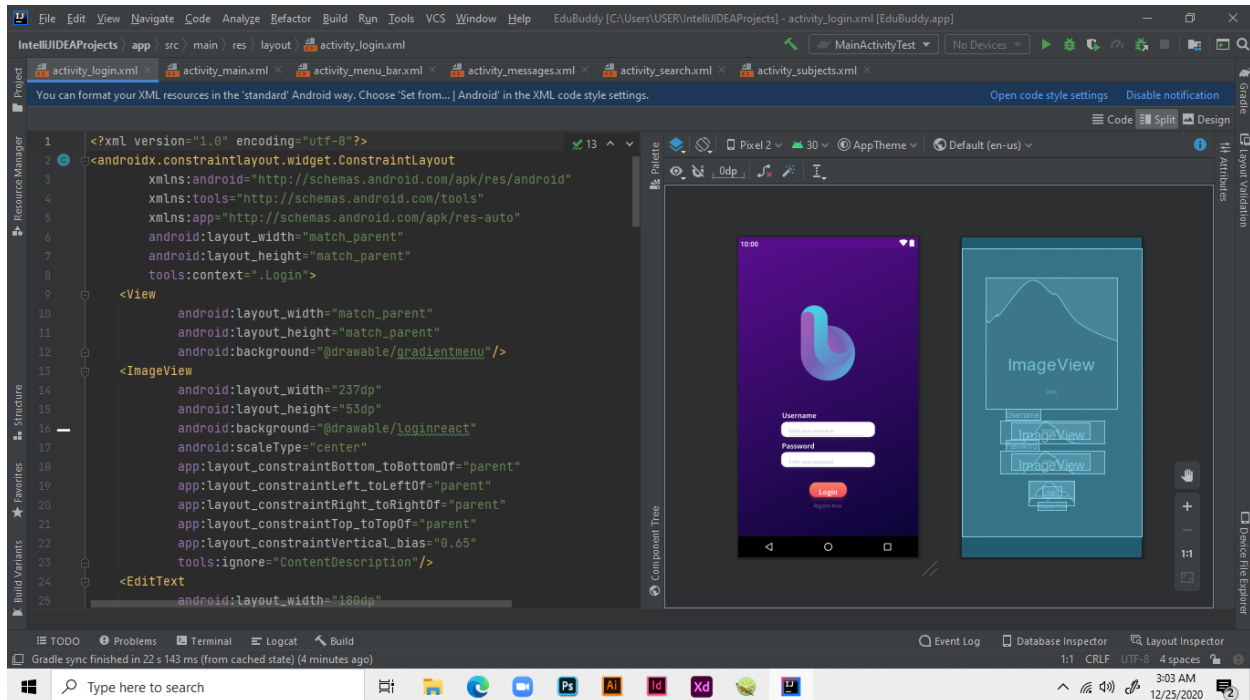
It is better to start off the solution with a clear objective and proper designed UX/UI in order not to complicate during the development process. Not only that, resources should be compiled properly.

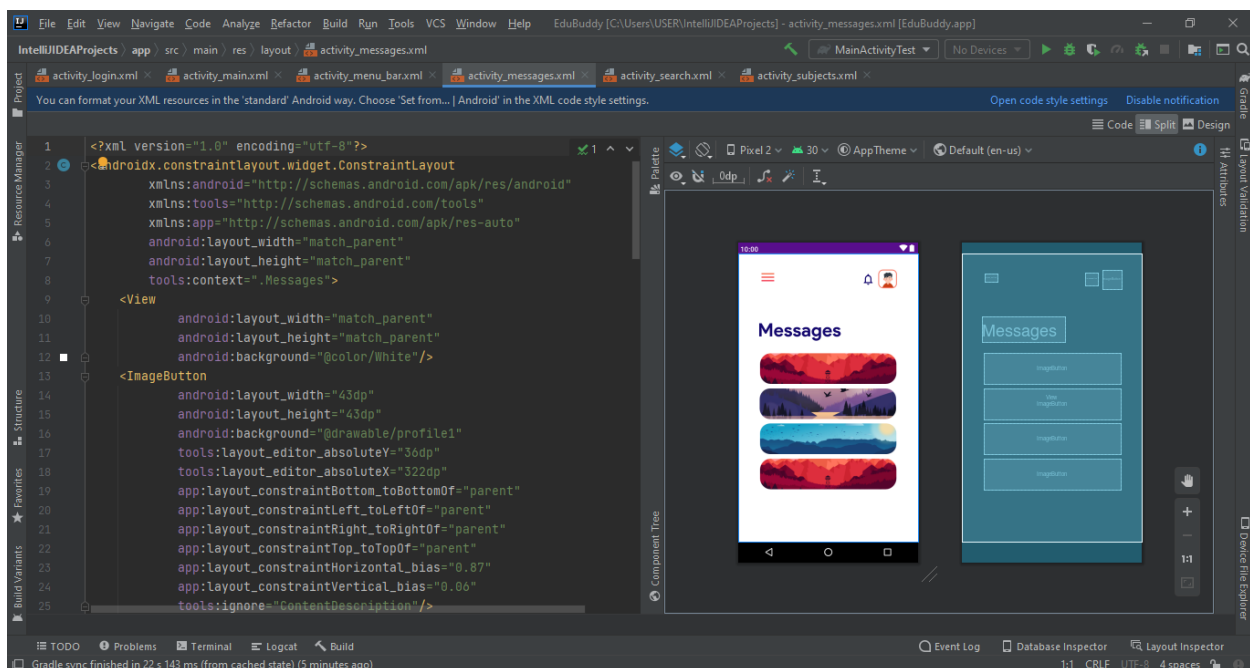
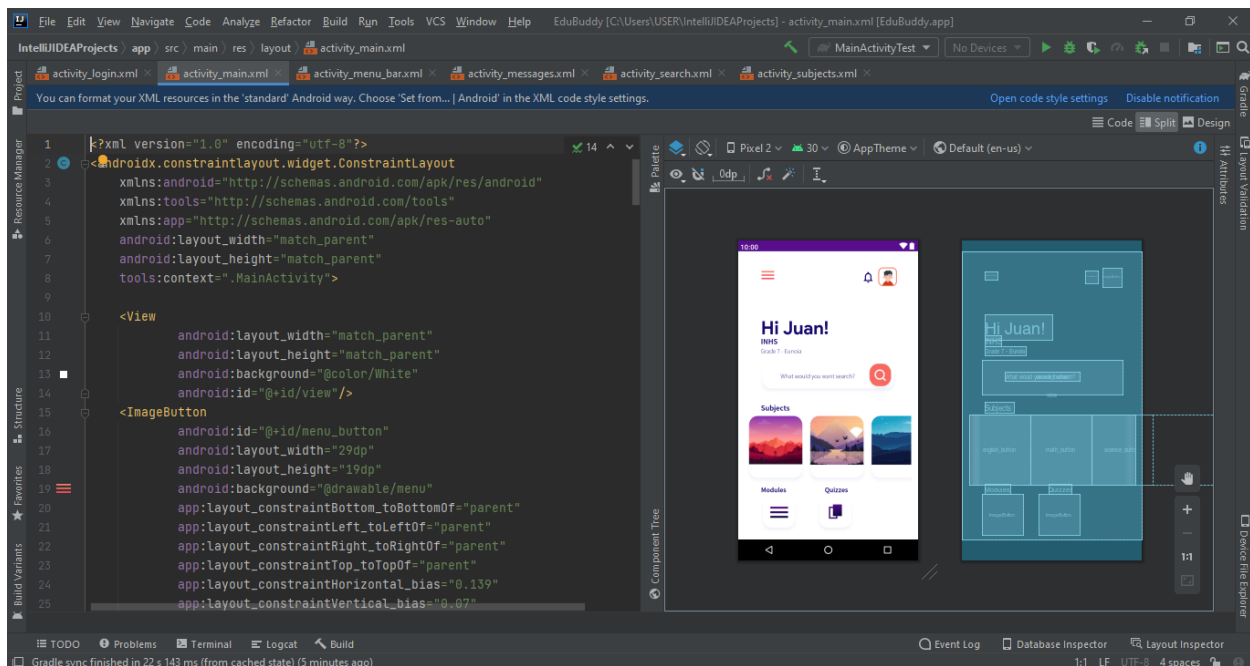
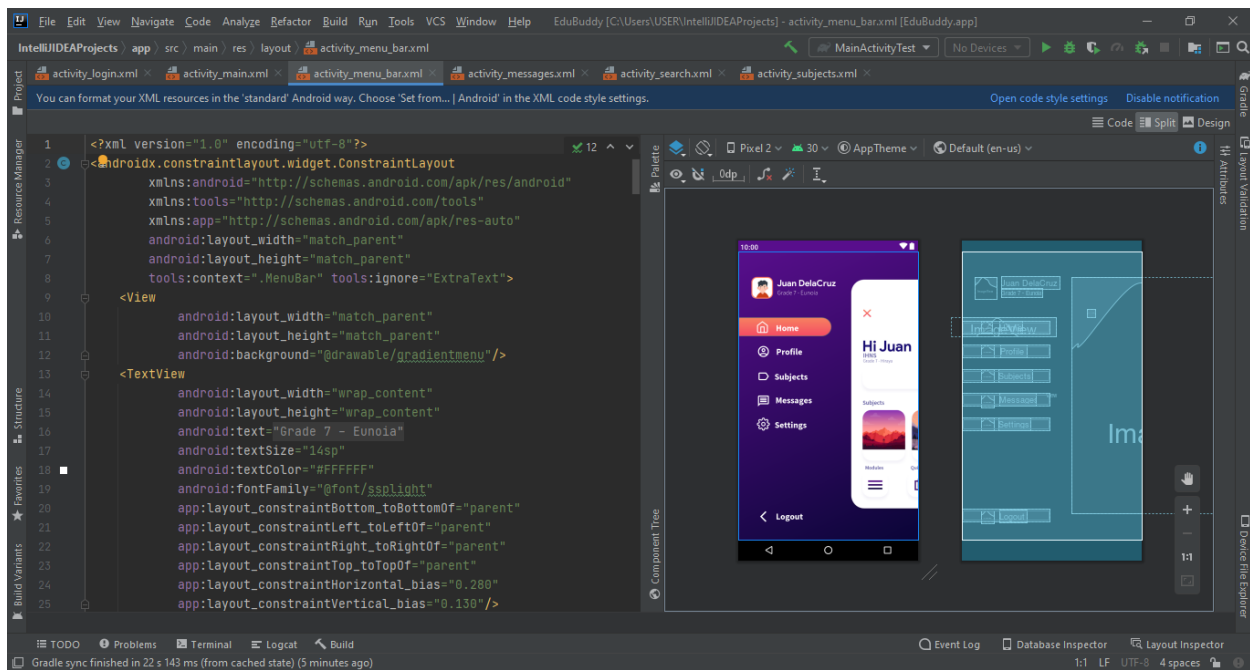
Moreover, small-scaled solutions that are well-designed and well thought out are factors to a feat of successful development. Development of the app should be in Android for the sake of inclusivity on the end-users. However, the demo visuals needed for the projects should be elaborated in IOS since it depicts a sophisticated look of the project.

Next, the UX/UI designs is the holy grail of the solution, therefore it needs to be concise and effective to deliver sustainable and cohesive efficacy of the solution

Lastly, the Department of Education in the Philippines will be able to find better alternatives other than the current setting which is online and modular setup. This will clearly give a heads-up to DepEd of how many students are in dire need of a tangible solution especially the students residing in far-flung areas of the country.

# Appendices





## References

- Gray, P. (2020). *SMS in the Philippines: still the most viable business messaging channel?* Retrieved December 14, 2020, from PromoTexter:  
<https://promotexter.com/sms-in-the-philippines/>
- Hong, B. (2019, April 15). *Create Android app with MVVM pattern simply using Android Architecture Component*. Retrieved December 15, 2020, from Medium:  
<https://medium.com/hongbeomi-dev/create-android-app-with-mvvm-pattern-simply-using-android-architecture-component-529d983eaabe>
- Magsambol, B. (2020, July 18). *More than 6M elementary, high school students fail to enroll during pandemic*. Retrieved December 14, 2020, from Rappler News:  
<https://www.rappler.com/nation/millions-fail-to-enroll-elementary-high-schools-philippines-2020-2021>
- Muntenesco, F. (2016, November 4). *Android Architecture Patterns Part 3: Model-View-ViewModel*. Retrieved December 15, 2020, from Medium:  
<https://medium.com/upday-devs/android-architecture-patterns-part-3-model-view-viewmodel-e7eeee76b73b>
- Nations, U. (2019, July 9). *Review of SDG 4 (Quality Education) implementation and interrelations among goals*. Retrieved December 14, 2020, from Sustainable Development Goals United Nation:  
[https://sustainabledevelopment.un.org/content/documents/23669BN\\_SDG4.pdf](https://sustainabledevelopment.un.org/content/documents/23669BN_SDG4.pdf)
- PSA. (2010). *Urban-rural classification*. Manila: Philippine Statistics Authority.