

SOEN 490 - Capstone Software Engineering Design Project  
Version 1

# Lock & Learn Milestone Document

Fall 2023

Instructor: Dr. Peter Rigby  
Date: 29/09/23

Student ID	Student Name
40176279	Alice Chen
40177866	Fatema Akther
40175616	Leon Zhang
40176360	William Chong
40175335	Julie Trinh
40174210	Hao Yi Liu
40178711	Anna Hsu
40175423	Ryan Kim
40175034	Georgia Pitic

Lock & Learn	3
Team Members	3
Project Summary	3
Risk	3
Legal and Ethical Issues	4
Economic	4
Contractor Estimate	4
Velocity	4
Overall Architecture and Class Diagram	5
Figure 1: Component diagram	6
Figure 2: Use Case diagram	7
Figure 3: Domain Model diagram	8
Figure 4: Class diagram	9
Infrastructure	10
Name Conventions	10
Code	10
Testing and Continuous Integration	10

## Lock & Learn

### Team Members

Name and student ID	GitHub id	Number of story points that member was an <b>author</b> on.
<b>Fatema Akther</b> (40177866)	fatema-a	
Leon Zhang (40175616)	LeonZhang7	
Julie Trinh (40175335)	julietrinh1	5
William Chong (40176360)	Chovv	5
Georgia Pitic (40175034)	georgia1515	
Ryan Kim (40175423)	RIGNITE	5
Alice Chen (40176279)	Aykchen	8
Hao Yi Liu (40174210)	Niniishere	8
Anna Hsu (40178711)	AnnaHsu1	8

### Project Summary

Lock & Learn provides an efficient tool for parents to have control over their child's digital life/device usage by implementing a lock mechanism on their device. Once the child completes and successfully passes the lesson selected by the parent, the system will unlock to allow the child to continue using their device. Our application can easily be tailored to help kids focus on their studies without having any digital distractions through their tablets, laptops and phones. This platform gives peace of mind to parents knowing that their child can maximize their academic potential while not having the need to regularly monitor them.

### Risk

The following are the highest risks:

- Locking the device: We want the locking system to work properly otherwise the app will not work as intended. We came up with multiple solutions and will be primarily focusing on encrypting sensitive data related to the locking mechanism to prevent tampering and password protection on changing the locking mechanism. See [Issue 17](#) for more information.
- Unlocking the device: We do not want permanent or unplanned locking as it would cause serious problems. To mitigate this, we will have an emergency code to unlock the device. See [Issue 20](#) for more information.
- Security for payment methods: Payment processes are sensitive to security issues since the user's personal information will be found there. We will be using Stripe SDK to ensure a secure transaction for the users. See [Issue 4](#) for more information.
- Uploading study material: There is a risk of offensive or inappropriate content being posted on the platform. There are several ways to counter this risk. We will mainly be focusing on developing our own explicit content detection software or using a library to identify unwanted images. See [Issue 1](#) for more information.

## Legal and Ethical Issues

**Privacy Concerns:** Locking and monitoring a child's device usage may raise privacy concerns, as it involves tracking and controlling their digital activities. This needs to be managed carefully to ensure compliance with privacy regulations, we need consent from children and parents.

**Ethical Considerations:** Locking children out of their devices for extended periods may lead to questions about the ethical implications of such control. Striking a balance between parental guidance and personal freedom is crucial. We need to make sure there is a limit for the study time so parents won't abuse the system.

**Environmental Impact:** No large component, as of now, can have an environmental impact due to increased energy consumption. This should be taken into account and minimized where possible.

**Discrimination Concerns:** The application's usage should be fair and not discriminate against any minority groups. It should not be used to target or discriminate based on race, gender, or other characteristics. We should monitor content to make sure there is nothing discriminatory.

If using libraries, the team has to ensure to use open source libraries in case the application is published on the marketplace.

**Impact on Jobs/Careers:** Depending on the extent of adoption, Lock & Learn could potentially disrupt existing careers related to child education, as it provides an alternative means for parents to manage their children's digital activities. Tutoring services might be less needed. However, this impact is likely to vary based on the business model given to us by the stakeholders.

## Economic

Parents purchasing study material will allow instructors to get paid and we will get a percentage of this amount.

## Contractor Estimate

No charge will be applied for release/iteration work. We will be working free of charge and in exchange, our group will have a 50% stake in the project.

## Velocity

*Project Total: 26 stories, 125 points over 26 weeks*

### Iteration 1 (2 stories, 13 points)

The main achievements are regarding setting up Github and the base of our application as well as installing Docker, CI/CD and its emulators. Stories were listed with their labels, risks, task distribution and points (SP, priority, risk). Documentations such as the project proposal, project milestones, software testing, product vision document, SAD (overview of architecture and design) were

established) and MOU were completed. We also progressed in the features of signing up/logging in and uploading study material which will be carried onto iteration 2 for further development.

#### [Iteration 2](#) ( 3 stories, 21 points)

The main achievement from this iteration will be: that users will be able to sign in based on their user type in a secure manner. Instructors will be able to upload study packages as quizzes and study material as well as to choose their formats. Users will also be able to lock their devices (within some constraints).

The rest of the iterations will be discussed as we go further within this project based on the priorities of our stakeholders.

#### **Overall Architecture and Class Diagram**

In our GitHub stories, we have started to include more in-depth diagrams to see their architecture more in-depth. We will create these diagrams as we go. As seen in [Issue 26](#) where we have a use case for that specific scenario and [Issue 1](#) where we have a state diagram and a use case diagram.

In Figure 1, we have a component diagram illustrating the main components of our system. It shows how the users, system and backend server connect with each other.

In Figure 2, we have a use case diagram displaying how users will use the application and components it interacts with, payment and AI correction tool (potential view from our client, which is not yet concrete).

In Figure 3, we have the domain model to help our stakeholders understand the structure and dynamics of the system we will create.

In Figure 4, we have a class diagram that illustrates the structure of the system. It shows our different classes, their attributes, their methods, and the different relations the classes have between them. It will help us understand the different components and how they interact together. More information will be added regarding the diagram in relation to the transaction after a discussion with the stakeholders.

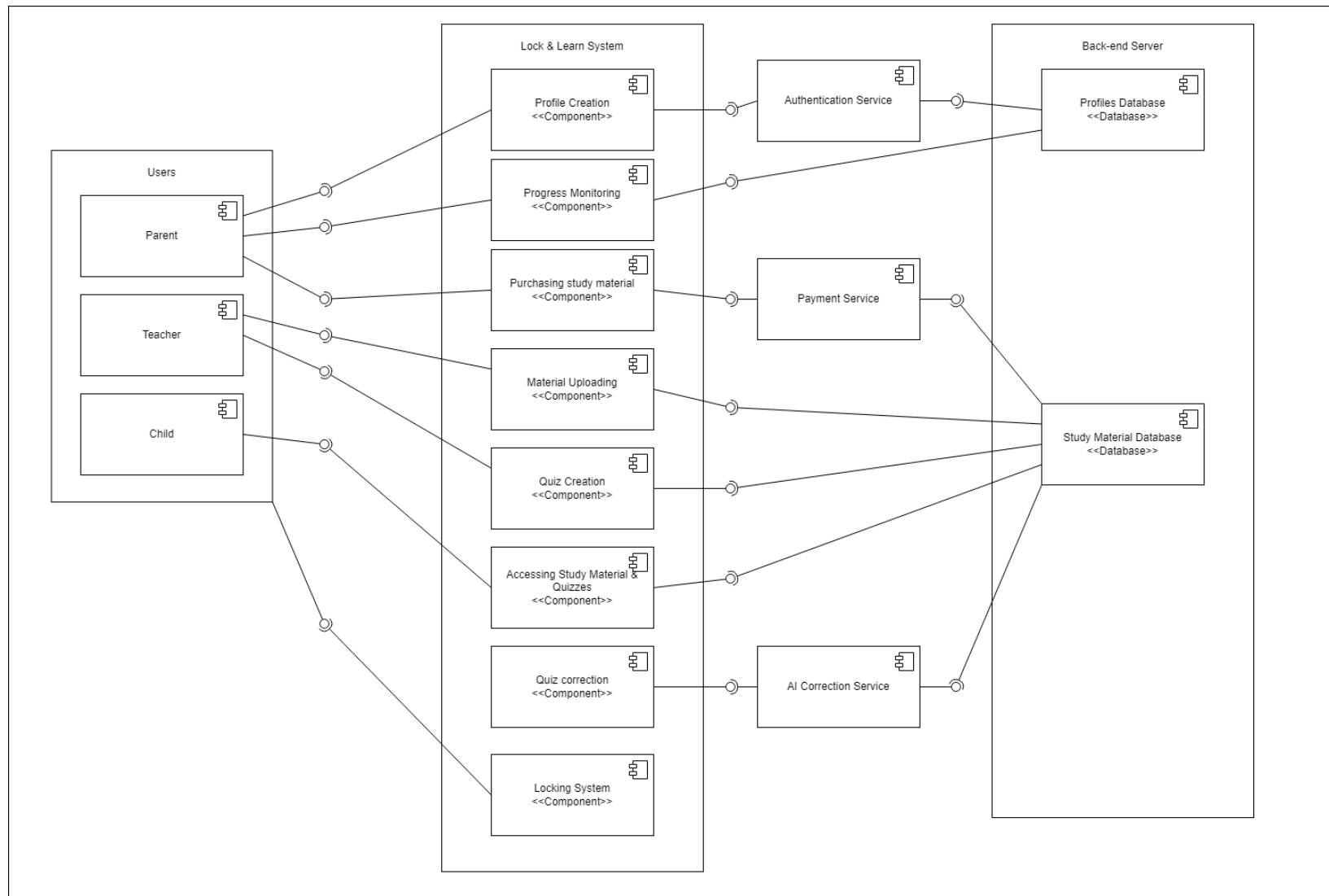


Figure 1: Component diagram

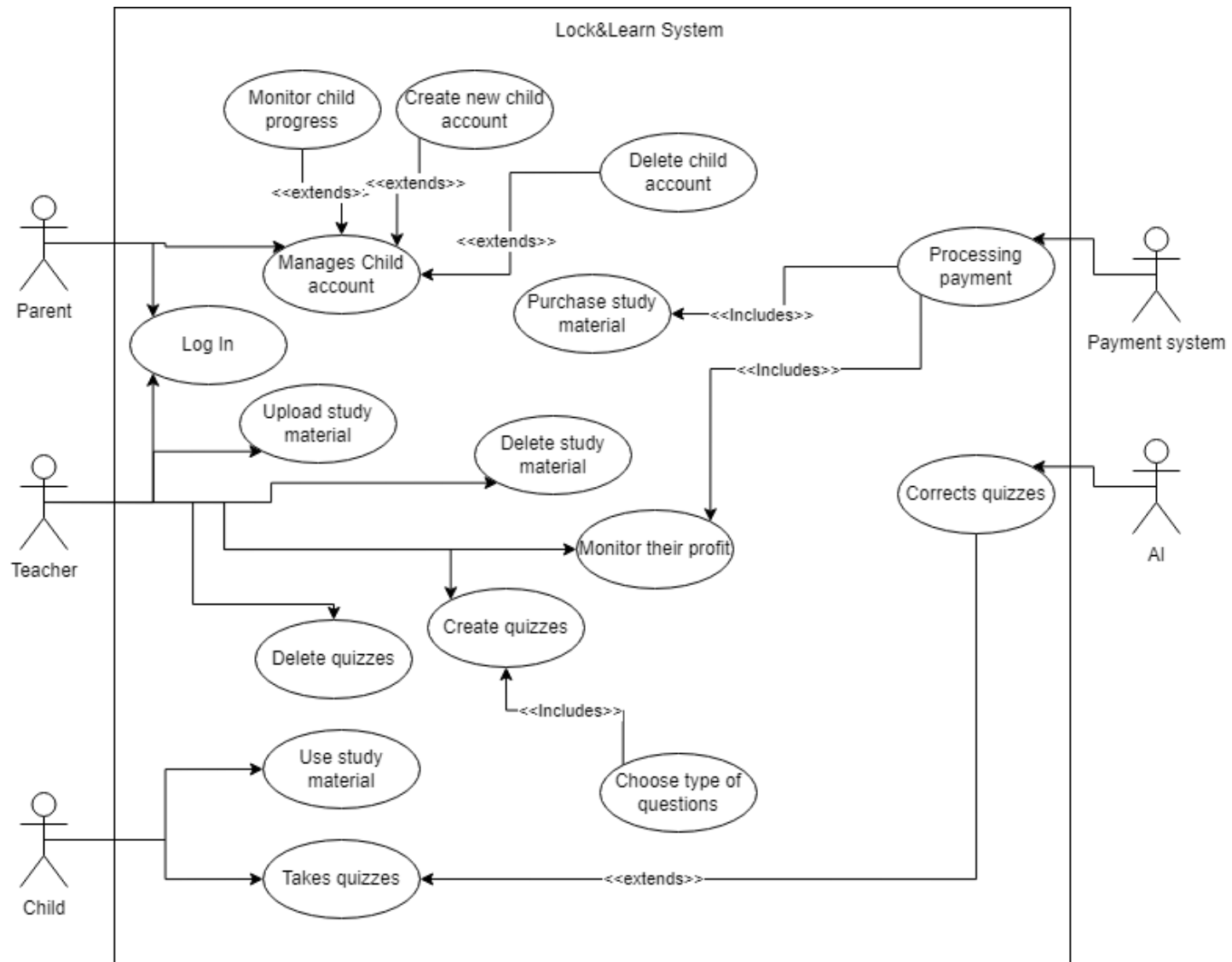


Figure 2: Use Case diagram

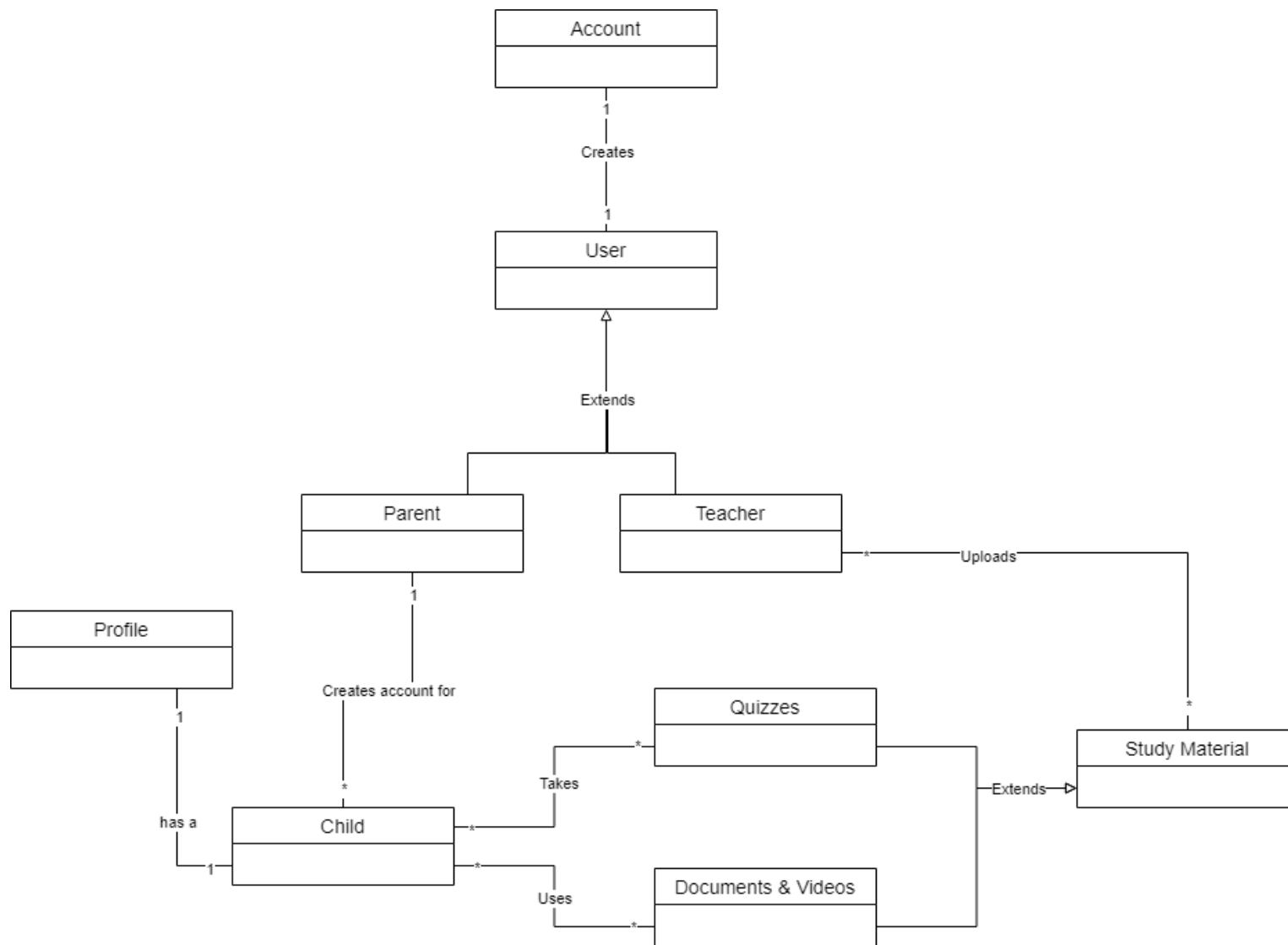


Figure 3: Domain Model diagram



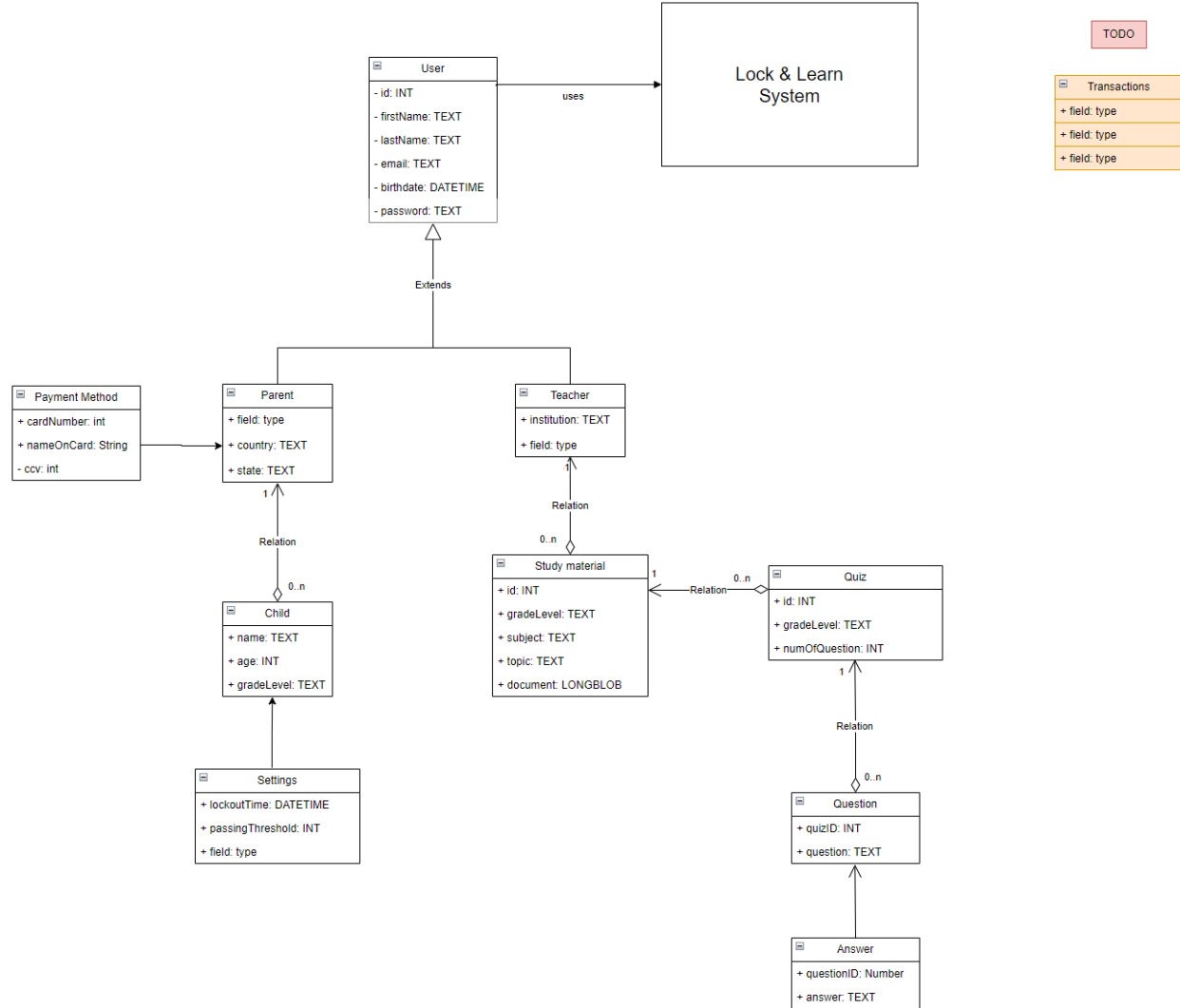


Figure 4: Class diagram

## Infrastructure

We are using different frameworks, which suit best for the team and for developing a mobile and web application:

- React Native for the UI
- Javascript for the backend
- MySQL for the database
- Docker for the container
- Jest for the unit and integration testing
- Detox for end-to-end testing

## Name Conventions

We will be using the CamelCase naming convention. To see how it is used refer to the following link:

[https://en.wikipedia.org/wiki/Camel\\_case](https://en.wikipedia.org/wiki/Camel_case)

## Code

File path with clickable GitHub link	Purpose
<a href="https://github.com/RIGNITE/LockAndLearn/blob/main/LockAndLearn/screens/UploadScreen.js">https://github.com/RIGNITE/LockAndLearn/blob/main/LockAndLearn/screens/UploadScreen.js</a>	[prototype] Screen where tutors will upload their education material
<a href="https://github.com/RIGNITE/LockAndLearn/blob/main/LockAndLearn/screens/LoginScreen.js">https://github.com/RIGNITE/LockAndLearn/blob/main/LockAndLearn/screens/LoginScreen.js</a>	[prototype] Screen where the user will log in
<a href="https://github.com/RIGNITE/LockAndLearn/blob/main/LockAndLearn/screens/SignupScreen.js">https://github.com/RIGNITE/LockAndLearn/blob/main/LockAndLearn/screens/SignupScreen.js</a>	[prototype] Screen where the user will sign up
<a href="https://github.com/RIGNITE/LockAndLearn/blob/main/LockAndLearn/screens/HomeScreen.js">https://github.com/RIGNITE/LockAndLearn/blob/main/LockAndLearn/screens/HomeScreen.js</a>	[prototype] Home screen of the application after login in
<a href="https://github.com/RIGNITE/LockAndLearn/blob/main/LockAndLearn/screens/EditUploadScreen.js">https://github.com/RIGNITE/LockAndLearn/blob/main/LockAndLearn/screens/EditUploadScreen.js</a>	[prototype] Screen for tutors to edit the content that they are uploading

## Testing and Continuous Integration

Test File path with clickable GitHub link	What is it testing
<a href="https://github.com/RIGNITE/LockAndLearn/blob/main/LockAndLearn/tests/App.test.js">https://github.com/RIGNITE/LockAndLearn/blob/main/LockAndLearn/tests/App.test.js</a>	This test validates 3 elements: if the screen can be rendered correctly, if the expected text is present and if the snapshot matches with the component.

This will be completed in the following iterations. More tests will be added for the screens without unit testing as the screens are still in development.

Description of continuous integration environment:

[Create node.js.yml template · RIGNITE/LockAndLearn@f815c4c \(github.com\)](#)

For the continuous integration, we are using GitHub's actions. Every merged pull request is run through continuous integration. This latter builds the code with the required dependencies and tests the added code/commits using Jest.