Lab 2: Tutor Dash Product Specification

Brandon Campbell

CS 411W

Team Gold

Professor T.J. Kennedy

08 November 2019

Version 2

Table of Contents

List of Figures

List of Tables

1 Introduction

University campuses provide a unique market with unique challenges. Students are often in need of tutoring services while, at the same time, in a position to offer tutoring services of their own. On a large campus, it can be difficult for students to communicate their need for services and promote tutoring services they may be offering.

Students seeking a tutor in a specific course often run into many obstacles. Many university tutoring centers only offer tutoring services in courses that have high fail rates. If a student is seeking assistance in a course that is not offered, they rely on their peers who have taken that course. The obstacle is in finding that peer.

Providing a central platform with the tools that connect tutors to tutees and make the transaction process simpler will mitigate the many challenges facing those seeking or offering academic assistance at colleges and universities.

1.1 Purpose

Tutor Dash will be a mobile application providing a digital marketplace for students to promote their tutoring services to other students. The application provides a structure to ensure transparency on services offered through a built-in rating and review system and enforces a fair cost of services through the use of pricing algorithms.

Tutor Dash will streamline the process of connecting qualified tutors to tutees, as demonstrated in Figure 1. It does this by eliminating many of the obstacles students face. Its benefit over the existing process is that it is designed specifically for tutoring services. While there are many ways to advertise services, they are not tutoring specific. When a student is in need of tutoring, they will be able to open the Tutor Dash application and have access to all

tutors at their school. They can search this list of tutors by the courses they offer, their cost for services, and the tutors' rating.

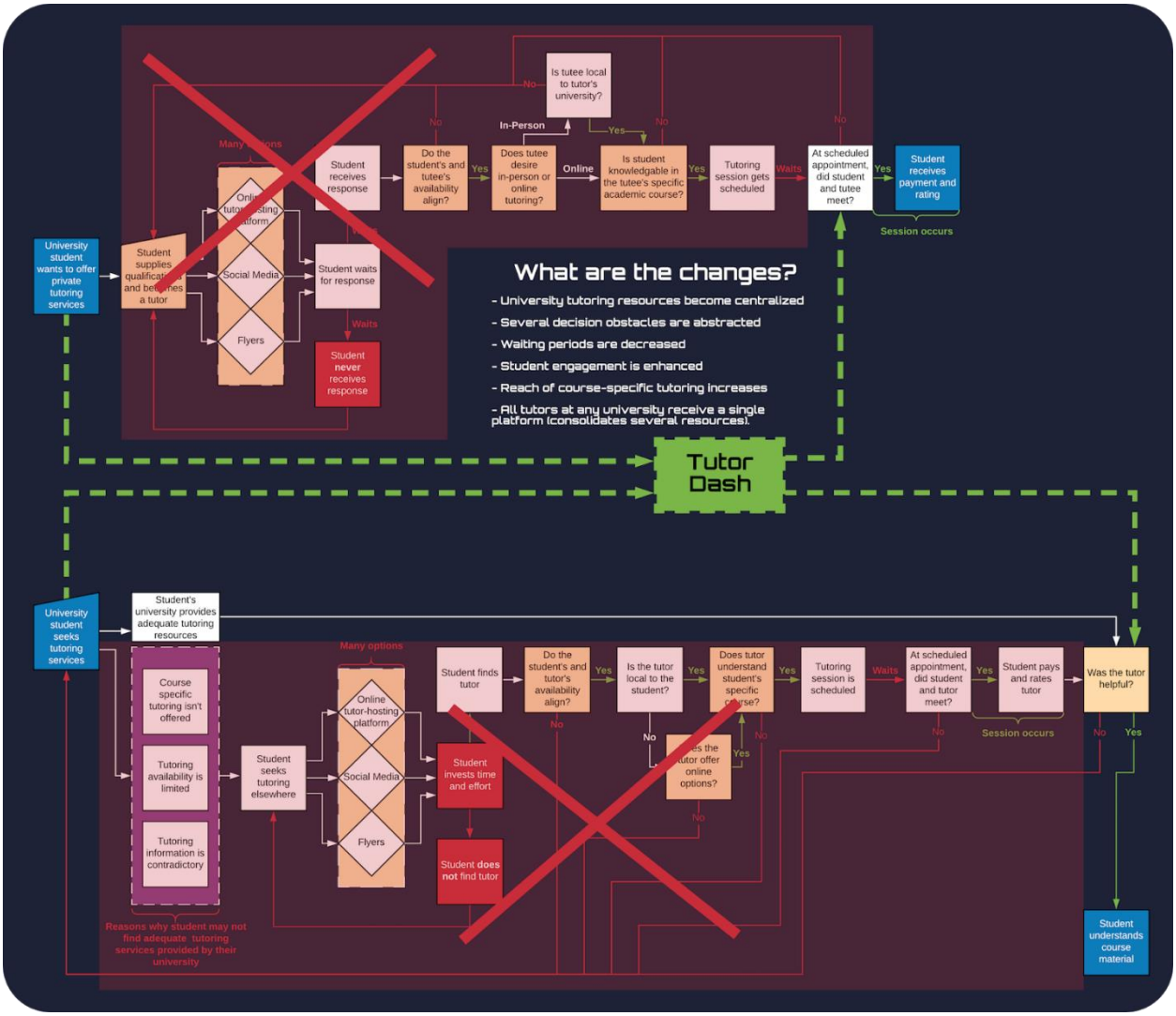

Figure 1. How Tutor Dash Affects the Current Processes

Tutor Dash users will be students who attend a college or university. These are students who are either seeking tutoring services or seeking to offer tutoring services. Those looking to offer tutoring services are likely students who are already tutoring and would like a way to expand their customer base, or they are students who are just looking to earn some extra cash from time to time without commitments.

The schools also stand to benefit from Tutor Dash. They can stand to gain from Tutor Dash being used widely on their campus because it can help alleviate the burden on school provided tutoring services and help improve the performance of the student body. A school's DFWI rate is a main indicator on the quality of the education provided by the school.

1.2 Scope

The objective of Tutor Dash is to make it easier for college students to find qualified tutors amongst their peers at an affordable price. It will fill the gaps in the market, which currently only consists of university resources, private tutors, and online tutoring businesses. This will ultimately lead to more successful students and establish Tutor Dash as the go-to resource for finding tutoring services.

The goal is to make the application simple, intuitive, and reliable so that students will be more likely to use the application and recommend it to their peers. Tutor Dash's modern approach to addressing the tutoring market will help break down the stigma involved in asking for academic assistance. As Tutor Dash grows in popularity, it will be scaled to work at any university.

The initial development of Tutor Dash will be a prototype version. The prototype will be limited in features and scalability when compared to the real world product (RWP). The prototype will be developed for smartphones with the Android operating system. Table 1 outlines the differences between the prototype and RWP.

[This space intentionally left blank]

Feature	RWP	Prototype
On-the-fly tutor qualification based on transcript	Fully-Functional	Fully-Functional
University student verification based on email	Fully-Functional	Fully-Functional
Search results tailored based on tutor/tutee mode	Fully-Functional	Fully-Functional
Real-time scheduling	Fully-Functional	Fully-Functional
Weighted ratings for every course	Fully-Functional	Fully-Functional
Reviews and comments on user profiles	Fully-Functional	Fully-Functional
In-app payments/deposits (any transactions)	Fully-Functional	Fully-Functional
In-app messaging/history of conversations	Fully-Functional	Fully-Functional
Web conference and in-person meeting support	Fully-Functional	Fully-Functional
Relative distance user A is from user B appears in query	Fully-Functional	Fully-Functional
Night mode	Fully-Functional	Eliminated
Automated pay rate calculation for every course	Fully-Functional	Partially Functional - Mean & std. dev. of pay-rates will need to be mocked up
Reporting features	Fully-Functional	Partially Functional - Users can report, but no action will occur
Re-authentication when navigating back into app	Fully-Functional	Fully-Functional - However, this feature may disrupt the user experience
Refunds due to poor experiences	Fully-Functional	Partially Functional - Most likely, this will not be automated, but it still will exist
Free sessions/monetary bonuses	Fully-Functional	Eliminated
Blacklisting of users	Fully-Functional	Partially Functional - Capabilities will be implemented but not used
Support of multiple universities	Fully-Functional	Eliminated
Cross-platform support	Eliminated	Eliminated
Firebase console linked to test suite(s) with mockups	Eliminated	Fully-Functional

Table 1. RWP vs. Prototype

The prototype is meant to provide a proof of concept and demonstrate the application's usefulness. Development of the RWP will require a lot of time, money, and resources. The prototype will demonstrate basic user interaction, transcript parsing for tutoring capabilities, rate calculation algorithm, and the ability to schedule tutoring sessions either in person or web-based.

[This space intentionally left blank]

1.3 Definitions, Acronyms, and Abbreviations

Course-specific tutoring: Academic assistance services provided for a particular course at a particular university.

DFWI: An acronym for Drop/Fail/Withdraw/Incomplete. This relates to university course incompletion status.

DFWI rates: Represents the ratio of university students who do not complete their courses to students who do complete their courses.

Direct Competitor: Another product or company which is solely involved in the same domain space as Tutor Dash.

Entity: A person, object, or external server that serves as a leveraged functional component of the Tutor Dash product.

FERPA: The Family Educational Rights and Privacy Act is a United States federal law that protects the privacy of educational records.

Indirect Competitor: Another product or company which is involved in the same domain space as Tutor Dash but does not focus solely in that space.

Tester: Individuals responsible for testing the quality of the software.

Tutee: A university student seeking academic assistance.

Tutor: A university student offering independent tutoring services that are qualified based off previously taken courses.

Uber: A ride-hailing company that offers the Uber mobile app, which you can use to submit a trip request that is automatically sent to an Uber driver near you, alerting the driver to your location.

1.4 References

"Academic Tutoring in Comprehensive Universities." Hanover Research, 2014. URL: https://www.hanoverresearch.com/wp-content/uploads/2017/08/Academic-Tutoring-in-Comprehensive-Universities.pdf

"Academics." Old Dominion University, 8 Jan. 2019. URL: www.odu.edu/academics

"Campus Tutoring." Old Dominion University, 19 Jan. 2019. URL: www.odu.edu/success/academic/tutoring#tab125=0

"Course-Specific Tutoring." Old Dominion University, 19 Jan. 2019. URL: www.odu.edu/success/academic/tutoring/course-specific

"Courses of Instruction." Old Dominion University, Feb. 2019. URL: catalog.odu.edu/courses/

Ciscell, Galen, et al. "Barriers to Accessing Tutoring Services Among Students Who Received a MidSemester Warning." ERIC, Pacific Lutheran University - Department of Sociology, 2016. URL: files.eric.ed.gov/fulltext/EJ1114513.pdf

Evans MDR, Kelley P and Kelley J (2017). Identifying the Best Times for Cognitive Functioning Using New Methods: Matching University Times to Undergraduate Chronotypes. Front. Hum. Neurosci. 11:188. doi: 10.3389/fnhum.2017.00188. URL: https://www.frontiersin.org/articles/10.3389/fnhum.2017.00188/full?&utm_source=Email_to_authors_&utm_medium=Email&utm_content=T1_11.5e1_author&utm_campaign=Email_publication&field=&journalName=Frontiers_in_Human_Neuroscience&id=239492

"Facebook - Groups." Facebook Help Center, Facebook, 2019. URL: www.facebook.com/help/1629740080681586?helpref=hc_global_nav

"Find a Local In-Home Tutor Today." HeyTutor, HeyTutor LLC. URL: heytutor.com/

Fry, Natalie. "New Research Reveals That College Students Study Best Later in the Day." NevadaToday, University of Nevada, Reno, 11 Apr. 2017. URL: www.unr.edu/nevada-today/news/2017/best-time-of-day-to-study

“Lab 1 - Tutor Dash Product Description.” Team Gold. 26 Sept. 2019. URL: https://www.cs.odu.edu/~411gold

"Peer Assisted Learning" BMC Education, 8 March 2006 URL: https://bmcmededuc.biomedcentral.com/articles/10.1186/1472-6920-6-18

Pierce, Dennis. "Supporting Students Beyond Financial Aid", 2016 URL: http://eds.b.ebscohost.com.proxy.lib.odu.edu/ehost/detail/detail?vid=0&sid=d93df6c4-3729-4b62-8d58-95e25c309878%40sessionmgr102&bdata=JnNpdGU9ZWhvc3QtbGl2ZSZzY29wZT1zaXRl#AN=114789419&db=ehh

Qayyum, Adnan. "Student Help-Seeking Attitudes and Behaviors in a Digital Era." International Journal of Educational Technology in Higher Education, vol. 15, no. 1, 2018, doi: 10.1186/s41239-018-0100-7. URL: https://educationaltechnologyjournal.springeropen.com/articles/10.1186/s41239-018-0100-7

"Skooli Tutors Online." Skooli Online Tutoring, Skooli, Feb. 2019. URL: www.skooli.com/prices/students

"Student as Peer Tutors" BMC Education, 9 June. 2014. URL: https://bmcmededuc.biomedcentral.com/articles/10.1186/1472-6920-14-115https://bmcmededuc.biomedcentral.com/articles/10.1186/1472-6920-14-115

Topping, J. Keith. "Trends in Peer Learning", 19 Jan 2007 URL: https://www.tandfonline.com/doi/full/10.1080/01443410500345172?scroll=top&needAccess=tru

"Tutor Matching Service - How It Works." Tutor Matching Service, Tutor Matching Service, 2019. URL: tutormatchingservice.com/#/about

"Tutor.com - The Princeton Review." Tutor.com, The Princeton Review, URL: www.tutor.com

"Tutors - Care.com." Care.com, Care.com, Feb. 2019. URL: www.care.com/tutors

"Wyzant." Wyzant Resources, Wyzant Inc., Feb. 2019. URL: www.wyzant.com/howitworks/students

[This space intentionally left blank]

1.5 Overview

This product specification outlines the architectural and functional descriptions of the Tutor Dash mobile application prototype. Sections 2 and 3 provide the detailed descriptions of the hardware, software, user, and communication interfaces required for the prototype; the features and performance requirements of the prototype; and the non-functional requirements necessary for the applications development and operation.

[This space intentionally left blank]

2 General Description

Tutor Dash is a mobile application intended to connect college-level tutors and tutees for course specific tutoring assistance. The application will allow students to search for a tutor in their desired course, schedule a tutoring system, handle payment for services, and allow the users to leave a rating and review based on their experience.

The application allows for two types of users, tutors and tutees. Tutees will have the ability to search for tutors at their university for tutoring services. Tutors will be required to upload their transcripts to determine their tutoring eligibility. Both users will have differing features and interactions within the application because of their tutor/tutee status.

2.1 Prototype Architecture Description

Tutor Dash is comprised of the following major components:

- Android mobile phone: this is the main source for user interaction. The mobile device will run the mobile application and must have internet access. The device must have the Android 8 or newer operating system.
- Android application: this is the main software component of Tutor Dash. The application is installed on the user's mobile device. The application performs all functions and features available to the user and provides the graphical user interface. The application performs all of the logical functions necessary for operation.
- Parsing algorithms: there are a few algorithms within the application. The transcript parsing algorithm parses an official university transcript in PDF format in order to determine the courses a tutor is eligible to tutor in. The pay rate algorithm calculates the

rate a tutor can charge for a specific course based on their experience, ratings, and course. The relative distance algorithm determines which tutors offering a specific course is within a selected radius of the tutee.

- Firebase: Firebase is a mobile and web development platform that provides cloud based tools and software for the development and operation of an application. Tutor Dash will use Firebase's Cloud Firestore for its database and Firebase Auth for authenticating users.
- Braintree: Braintree is a mobile and web payment system that will be integrated into Tutor Dash using the Braintree API. Braintree will handle all transactions between users.
- Google G Suite: if users would like to schedule web based tutoring sessions they will need a Google account. Web sessions will be handed off to Google Hangouts.
- Google Maps: The Google Maps API will be used in Tutor Dash to display a map to users showing nearby tutors and tutees.

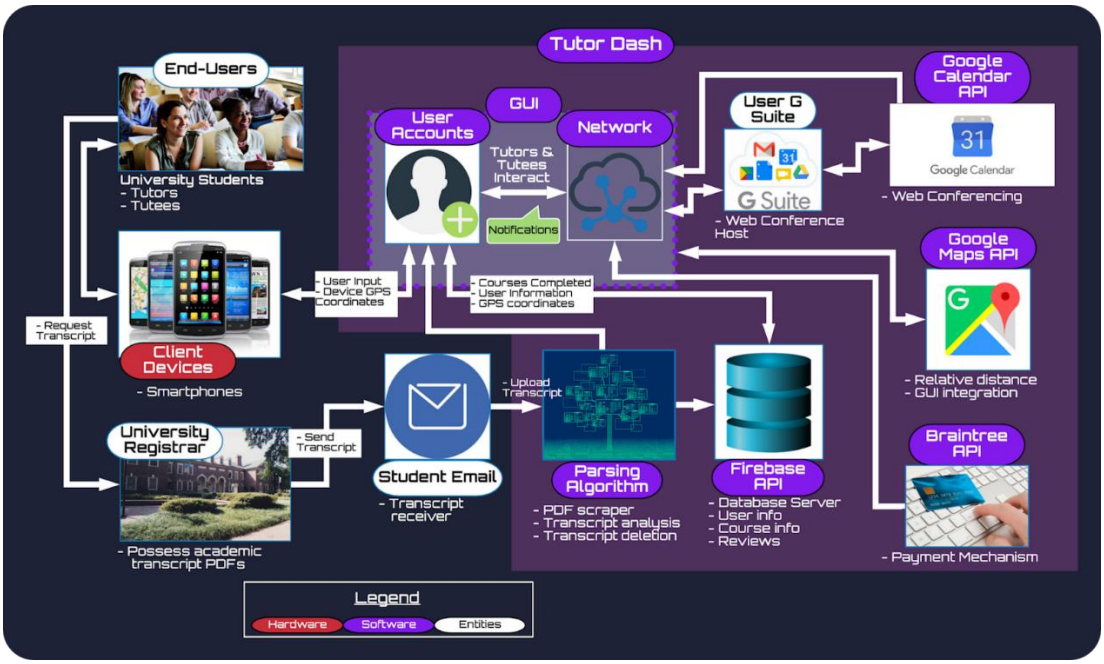

Figure 2. Major Functional Component Diagram

2.2 Prototype Functional Description

Some of the main functions of Tutor Dash include the following:

- User Account Creation and Login: This function prompts the user to login after launching the application. If a user does not have an account, they have the option of creating one. Users will provide some basic information to create the account. If the user chooses to be a tutor, they will be prompted to upload their transcript. (See Figure 5)
- Transcript Upload: Users will be able to upload an official transcript to become designated a tutor. The transcript parser will determine which courses they are eligible to tutor. The prototype will only accept an official Old Dominion University transcript. (See Figure 3)

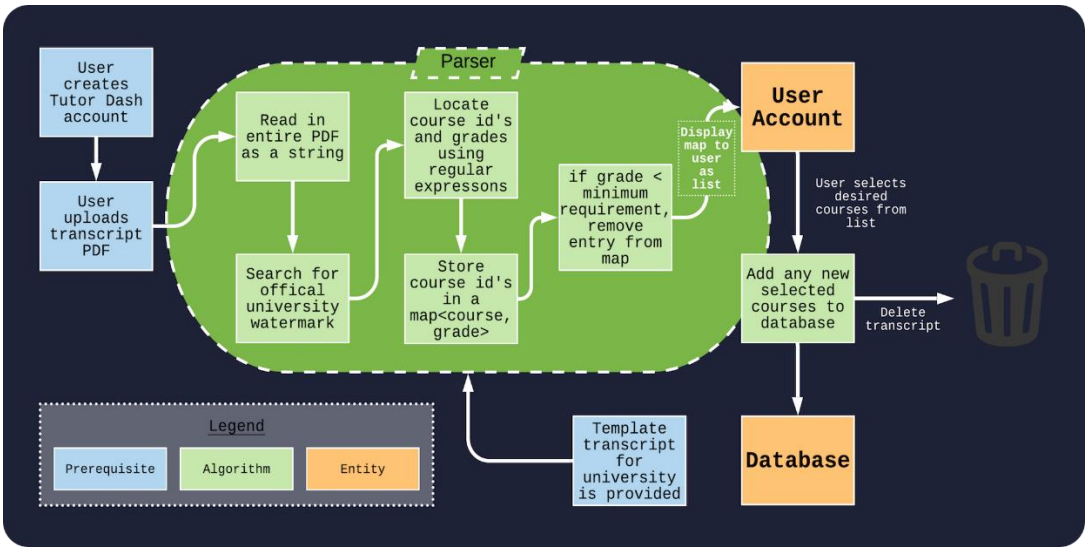

Figure 3. Transcript Parsing Algorithm

- Map Discovery: The map discovery function will show the user a map of their nearby location and any available tutors nearby. (See Figure 5)
- Text Discovery: The text discovery function will show a list of all available tutors nearby. (See Figure 5)
- Search Discovery: The search discovery function will allow a user to search for tutors by a specific course, tutor, or rating. (See Figure 5)
- User Profile: The user profile function will display a user's profile page containing their basic information, courses they're eligible to tutor, and their ratings and reviews. This page will allow users to navigate to a direct message with the user or schedule a tutoring session. (See Figure 5)
- Session Scheduling: This function will allow a tutee to select a date and time for a tutoring session request which will then be sent to the tutor. The tutor will have the option of accepting the request, modify the date/time, or deny the request. (See Figure 5)
- Rating and Review: This function will allow a tutee to rate and review a tutor after a session and vice versa. (See Figure 5)
- Chat Messaging: This function allows users to have real-time, direct messaging. (See Figure 5)
- Settings and Edit Profile: This function will allow a user to modify the settings of their account, profile, and application. (See Figure 5)

- Payment: This function allows users to perform transactions following a tutoring session. The actual processing of the payment is handled by a third-party API, Braintree.
- Pay Rate Calculation: The pay-rate for a specific tutor and a specific course is automatically calculated based on a list of parameters. This will allow for dynamic pay-rates which will meet the demands of the market. (See Figure 4)

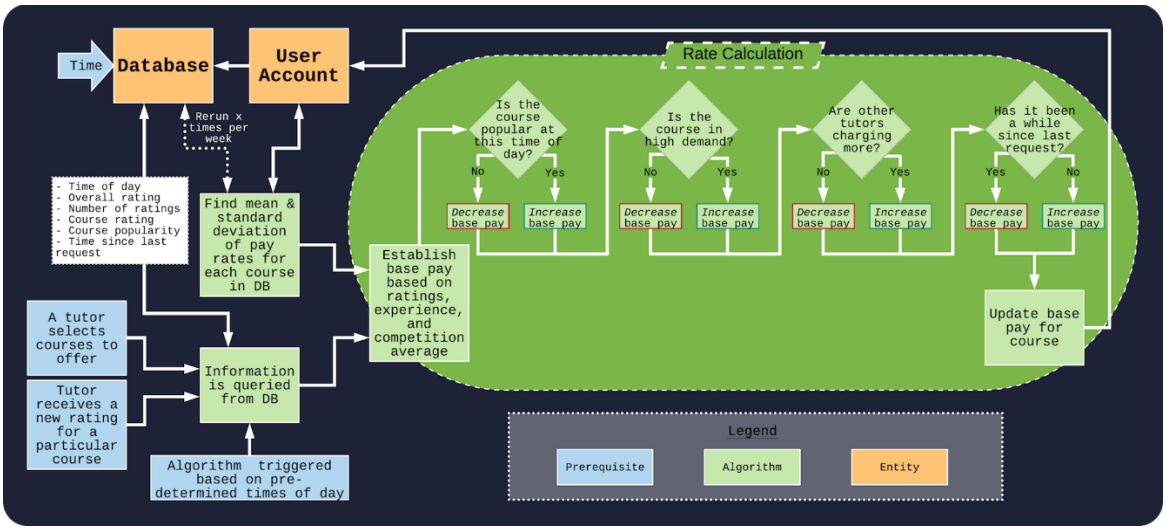

Figure 4. Pay Rate Algorithm

- Alert System: This function will allow tutees to send a request out to all tutors whether they are marked available or not. It will also allow tutors to send out a notification promoting a specific tutoring service.

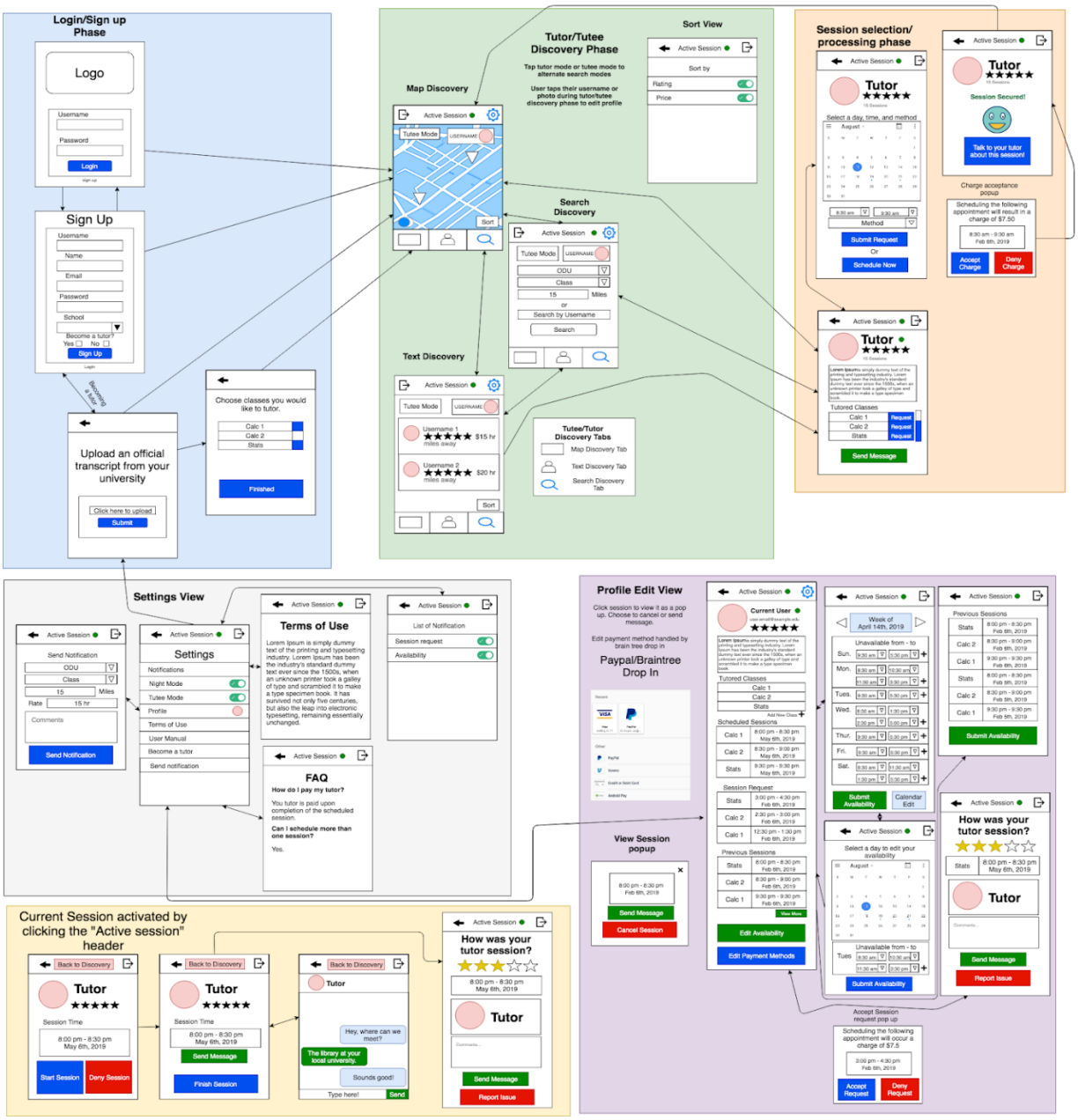

Figure 5. UI/UX Full Implementation

2.3 External Interfaces

The following describes the hardware, software, user, and communication interfaces used in the Tutor Dash application.

2.3.1 Hardware Interfaces

The hardware used in the prototype will be limited to an Android smartphone running the Android operating system. The RWP will be expanded to include an iOS version which can be used on iPhone devices.

Figure 6. Android Smartphone

2.3.2 Software Interfaces

The software interfaces include the Android application, which supports user interaction, and a few backend components such as Cloud Firestore, Firebase Auth, and Braintree.

- Tutor Dash Android Application: This software will be installed on the user's smartphone and will provide all of the interactions and logic necessary for the application.

- Firebase Cloud Firestore: This is a cloud-based software that will provide the main database for the application. The database for the prototype will be limited in storage size and the number of queries that can be made per week.
- Firebase Auth: This is a cloud-based software that will perform all functions necessary for authenticating a user.
- Braintree: This is a third-party software that handles processing payments between users. It is integrated into the Android application through a Java API.

2.3.3 User Interfaces

The user interfaces include the Android Mobile Device and Google Hangouts. These provide the primary means for the user to interact with the application.

- Android Mobile Device: This provides the primary interface available to the user. It must have a display, keyboard, internet access, and GPS.
- Google Hangouts: This will provide the interface for web-based tutoring sessions.

2.3.4 Communications Protocols and Interfaces

Tutor Dash relies on Internet and GPS connectivity in order to connect tutors and tutees. Both communication protocols are necessary for the application to function.

- Internet: The user's device must be capable of connecting to the internet using either network data or Wi-Fi.
- GPS: The user's device must have GPS capabilities.