Tutor Dash

"Find a tutor. Be a tutor."

Prototype Design

Old Dominion University
CS410 Spring 2019
Team Gold
May 8, 2019

Table of Contents

Team Gold

Alex Wojtowicz
Project Manager
DB/Algorithms Developer

Brandon Campbell
Database Manager
Back-End Developer

John Hessefort
UI/UX Developer/Tester
Domain Expert
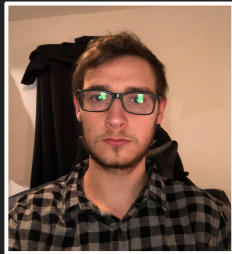
Duncan Holterhaus
Back-End Developer
Algorithms Developer

Jamauni Taylor
Webmaster
UI/UX Developer
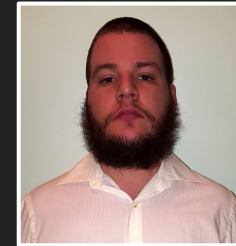
Dwight Owings
Tester
Quality Assurance

Availability of University Tutoring Programs

- University students study at no specific time of the day.

- Studies even suggest that more students study at night (as opposed to daytime).

Distribution of university students' study times
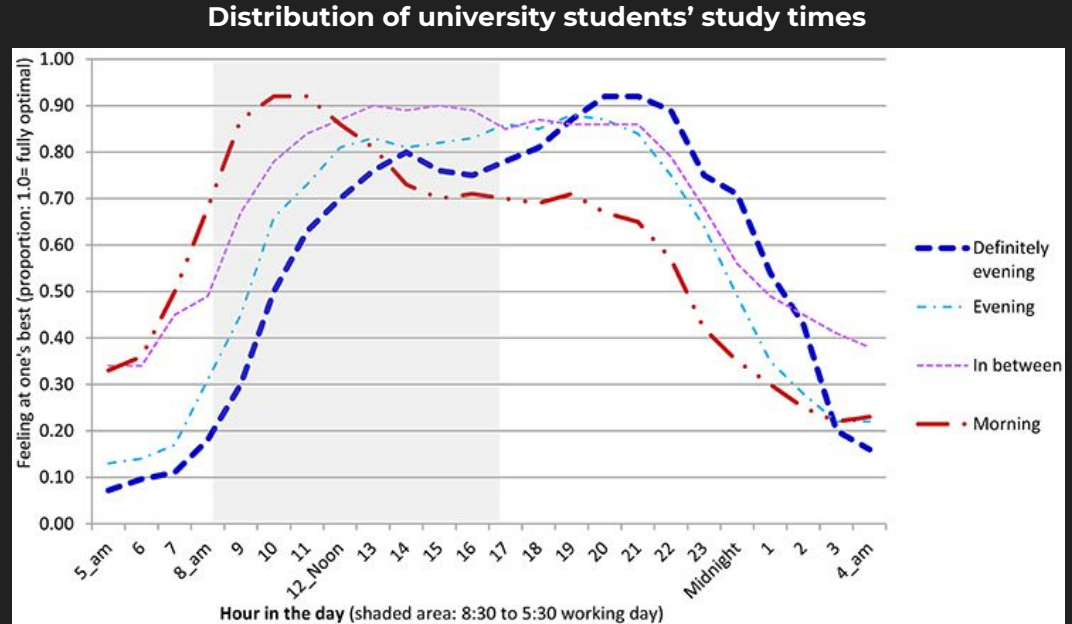
Source: *Evans MDR, Kelley P and Kelley J (2017)*

The Private Tutoring Market

- Private tutoring is a growing industry.

- However, there is no centralized network for tutors.

- Most tutor-hosting services only tend to focus in the online tutoring market.

The market will be **ACCELERATING** growing at a **CAGR** of more than

CAGR = Compound Annual Growth Rate

7%

INCREMENTAL GROWTH ▶

$5.80 bn

2017   2022

The year-over-year growth rate for **2018** is estimated at

▲ 6.48%

The market is **EXTREMELY FRAGMENTED** with many players occupying the market share

72% of the growth will come from the **ONLINE LEARNING** segment

One of the **KEY DRIVERS** for this market will be the growing student enrollments

READ THE REPORT:

PRIVATE TUTORING MARKET IN THE US 2018-2022

10,000+ reports covering niche topics
EDUCATION
Read them at:
www.technavio.com

technavio

Source: *technavio.com*

University Students as Tutors

- Student-to-student interactions are effective; tutees learn more cognitively through bonds with their peers.

- PAL is a peer mediated instructional program where students tutor their peers.

- Peer Assisted Learning Programs (PAL) proves that students tutoring other students is effective.

- University students are willing to tutor people what they know in their free time if it is easy to find people to tutor.

The Problems

1. Tutoring services available to university students are limited in scope, do not provide flexibility, and lack a centralized platform for promotion.

2. Students willing to provide tutoring services lack a tool to promote those services.

Problem Characteristics

- Limited Scope; not all courses offer tutoring

- Tutoring services do not encompass all study times.

- Information inconsistency; university tutoring information is spread out

- Students can't always trust private tutors to understand their course-specific material

- Students do not have an adequate platform to advertise their tutoring services.

Current Process - Tutees Seeking Tutors
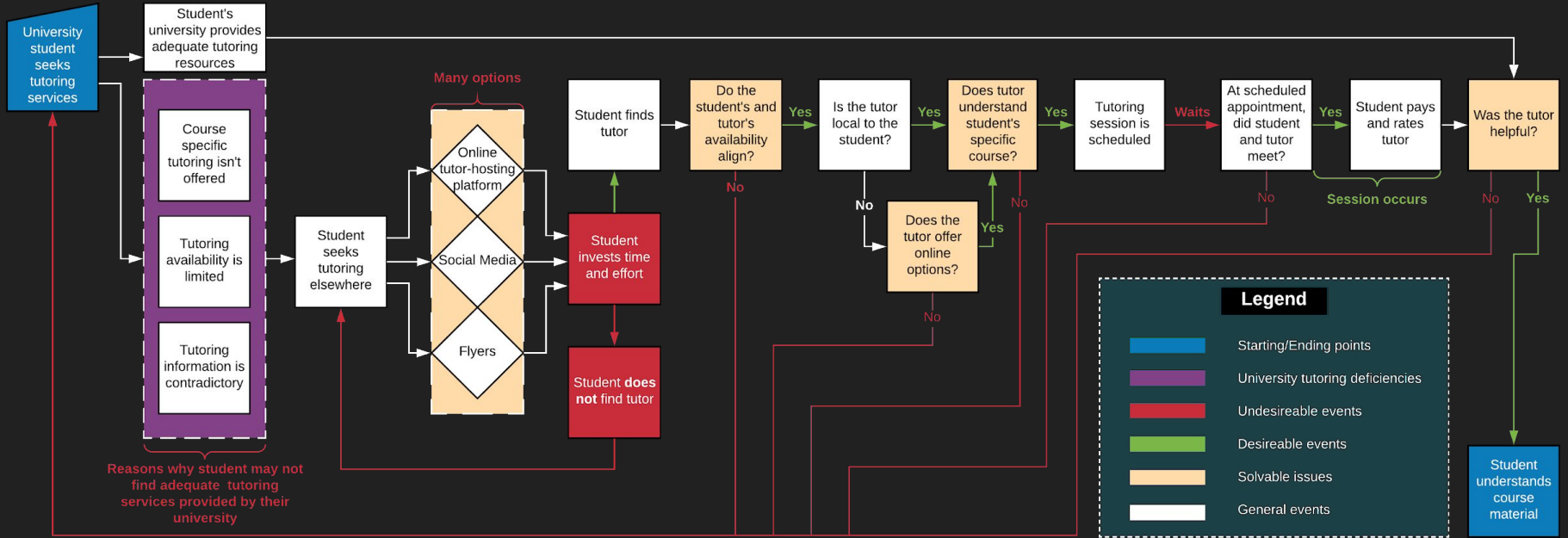
Current Process - Tutors Seeking Tutees

University student wants to offer private tutoring services
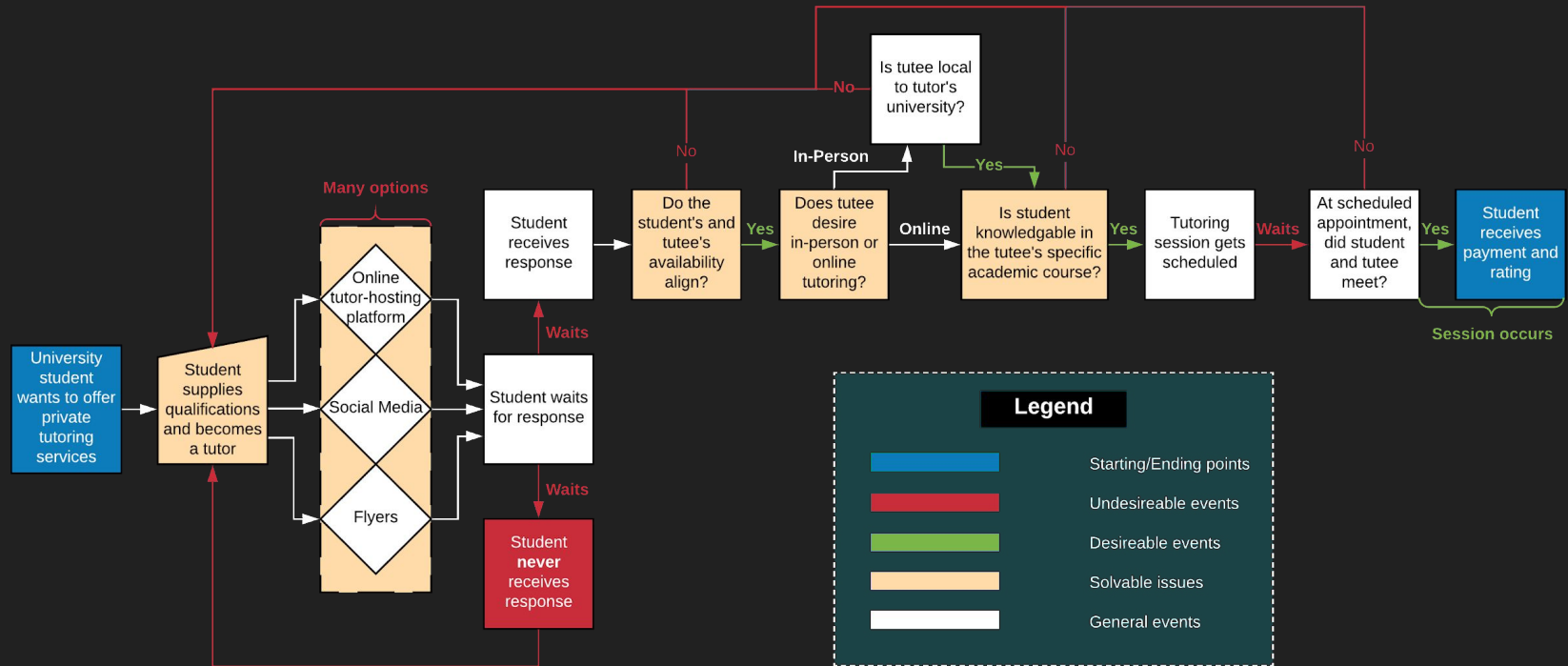
Student supplies qualifications and becomes a tutor

Many options

Online tutor-hosting platform

Social Media

Flyers

Student receives response

Waits

Student waits for response

Waits

Student **never** receives response

Do the student's and tutee's availability align?

Yes

No

Does tutee desire in-person or online tutoring?

In-Person

Online

Is tutee local to tutor's university?

No

Yes

Is student knowledgable in the tutee's specific academic course?

No

Yes

No

Tutoring session gets scheduled

Waits

At scheduled appointment, did student and tutee meet?

No

Yes

Student receives payment and rating

Session occurs

Legend

|	
(blue)	Starting/Ending points
(red)	Undesireable events
(green)	Desireable events
(orange)	Solvable issues
(white)	General events

Proposed Solution - **Tutor Dash**

- Tutor Dash is a mobile application-based service that unifies university students who are interested in tutoring with university students who desire course-specific tutoring in real-time.

- Tutor Dash provides an extension of any university's current tutoring services so that students can receive help for all courses 24/7.

University

Students Seeking Academic Assistance

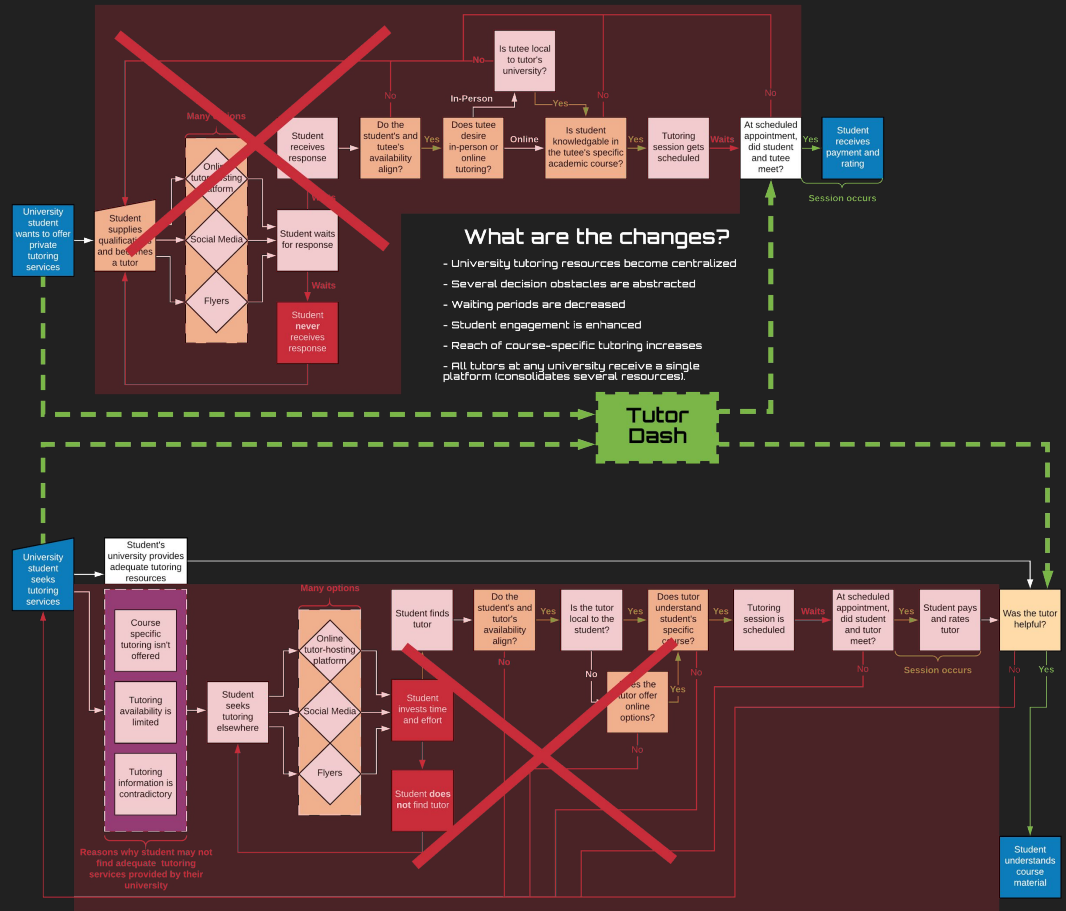Students Seeking Tutoring Advertisement
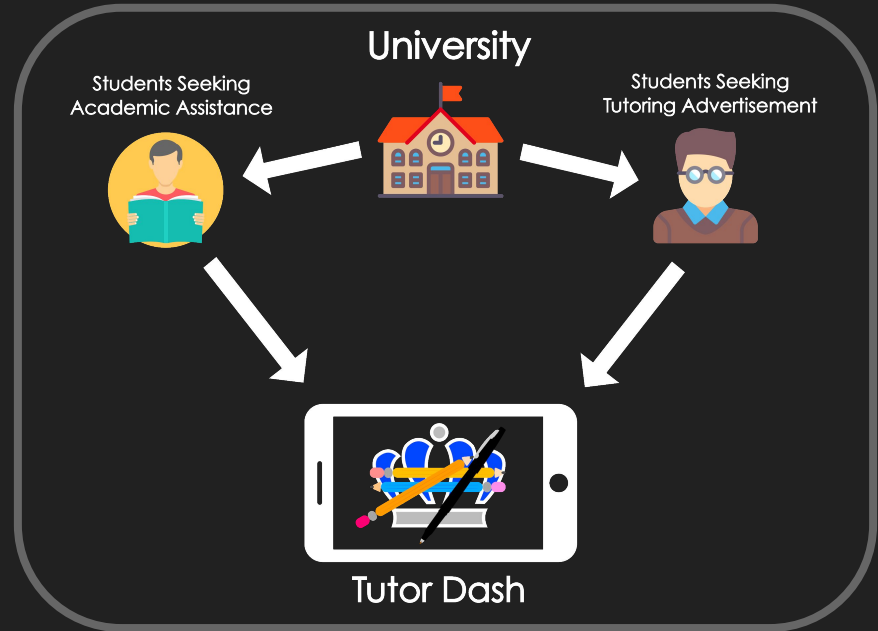
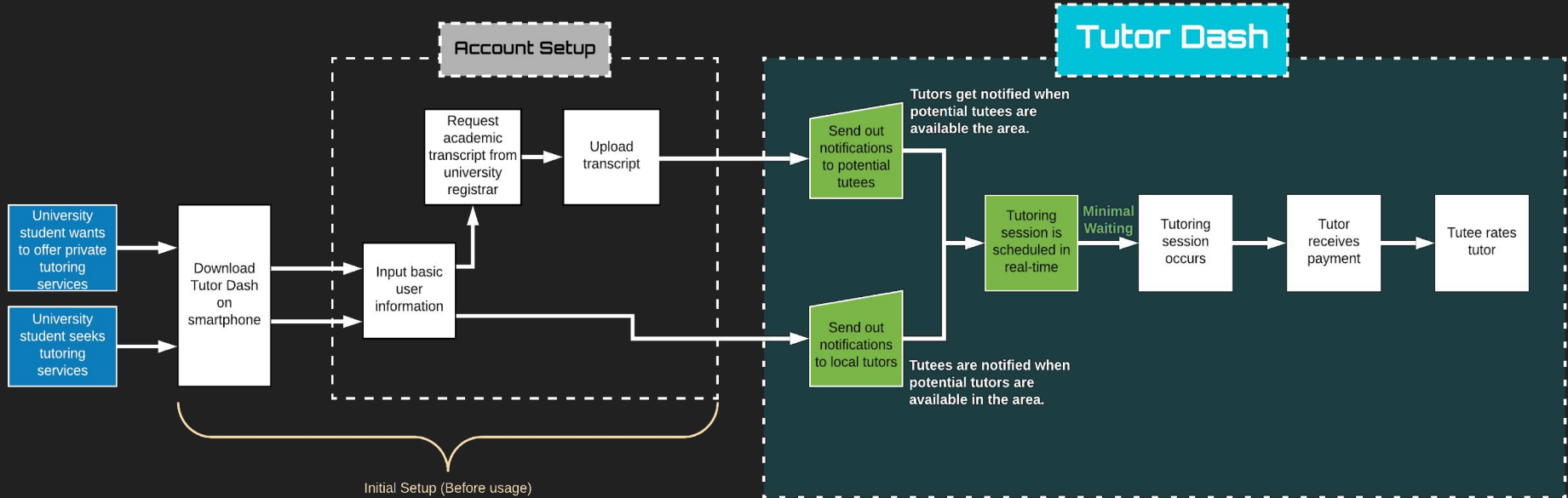Tutor Dash

Characteristics of Tutor Dash

- User-base constrained to university students

- Every tutor verified based on their previously taken classes

- Rating system for both tutors *and* tutees

- Qualified tutors can tutor in any class available at their respective university

- Tutors and tutees can be available at any time

- Notifications sent to connect participating tutors and tutees

- Allows for in-person and online meetings

- Mechanism for payment within application

How Tutor Dash Affects Current Processes

- Creates a more obvious connection between tutors and tutees at universities

- Niche tool for participating tutors and tutees

- More centralization → Less options → More opportunity

What are the changes?

- University tutoring resources become centralized
- Several decision obstacles are abstracted
- Waiting periods are decreased
- Student engagement is enhanced
- Reach of course-specific tutoring increases
- All tutors at any university receive a single platform (consolidates several resources).

Current Process Flows & Tutor Dash

Account Setup

Tutor Dash

University student wants to offer private tutoring services

University student seeks tutoring services

Download Tutor Dash on smartphone

Input basic user information

Request academic transcript from university registrar

Upload transcript

Send out notifications to potential tutees

Tutors get notified when potential tutees are available the area.

Send out notifications to local tutors

Tutees are notified when potential tutors are available in the area.

Tutoring session is scheduled in real-time

Minimal Waiting

Tutoring session occurs

Tutor receives payment

Tutee rates tutor

Initial Setup (Before usage)

What Tutor Dash Will Not Do

- Not an "official" university tutoring resource, rather a tool for presenting information on a unique platform

- Will not make any money off of existing university tutoring resources

- Will not interfere with current university tutoring resources; only improve advertisement to extend reach

- Will not violate the Family Education Rights and Privacy Act (FERPA)

- Cannot guarantee there is a tutor for everyone online 24/7

- Cannot prevent students from voluntarily violating the honor code

- Will not be a hosting platform for established tutoring businesses

Competition

Features	Tutor Dash	Tutor.com	Tutor Matching Service	Skooli	Wyzant	HeyTutor	Care.com	Public Facebook Group
Offers various university course-specific tutoring	✔	✔	✔	✖	✖	✖	✖	✖
Allows qualified undergraduate students to be tutors	✔	✖	✖	✖	✖	✖	✖	✖
Sends notifications about local tutors/tutees	✔	✖	✖	✖	✖	✖	✖	✖
Qualified tutors are constrained to university communities	✔	✖	✖	✖	✖	✖	✖	✖
Provides real-time scheduling capabilities	✔	✔	✖	✖	✖	✖	✖	✔
Available as mobile application	✔	✔	✔	✖	✔	✔	✔	✔
Includes online tutoring options	✔	✔	✔	✔	✔	✖	✖	✔
Includes in-person tutoring options	✔	✖	✔	✖	✔	✔	✔	✔
Provides tutor ratings	✔	✔	✔	✔	✔	✔	✔	✖
Establishes hourly rate ceilings	✔	✔	✔	✖	✖	✖	✔	✖
Requires tutor verification/validation	✔	✔	✔	✔	✔	✔	✖	✖
Does not require subscription commitment	✔	✖	✖	✖	✖	✖	✔	✔
Provides 24/7 scheduling	✔	✔	✔	✔	✔	✔	✔	✔

Customers

- University students seeking tutoring services (academic help)

- University students interested in tutoring (providing academic help)

End Users

University student bodies

- University students looking for tutors
  - Seeking immediate help
  - Seeking future help

- University students looking for tutees
 - Seeking network growth
  - Seeking extra money

User Roles

Tutee	Private Tutor	Tester
• University student • Seeks academic assistance	• University student • Qualified to tutor previously taken courses • Offers tutoring services • Works independently	• Team member • Responsible for quality of software • Uses tools to write tests for test automation

User Stories - Tutee

As a tutee, I need…
1. …the opportunity to receive tutoring in any class which I am enrolled in.
2. …the ability to receive/request tutoring at any time of the day.
3. …the ability to message tutors in real-time before hiring them.
4. …to search for classes I need help in and receive a list of tutors.
5. …payments to be secure and only charged when I can confirm that the meeting did, in fact, occur.
6. …the ability to leave meaningful ratings and reviews based on my experiences with tutors.
7. …the ability to report malicious activity regarding authorized tutors' activities.
8. …the ability to send out alerts so that tutors will be notified when I am looking for them.
9. …the ability to confirm that a session did, in fact, occur.
10. …to see some indication that tutors are currently available to hire right now.
11. …to see the distance a tutor is away from me if I am, in fact, seeking an in-person meeting.
12. …the ability to receive tutoring in-person.
13. …the ability to receive tutoring online.
14. …the ability to search for tutors tutoring any course at my university.
15. …the ability to reconnect with a tutor I liked so I can rehire them in the future.
16. …tutor profiles to be public when I search for them.
17. …sensitive account information to be private and secure.
18. …reassurance that the person I am meeting is the person they say they are.
19. …registered tutors to be qualified and authorized to tutor me.
20. …to search for tutors who attend my university within a certain radius.
21. …the ability to report malicious activity regarding authorized tutors' activities.
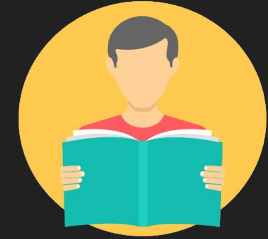
As a tutee I wish…
1. …for an appealing visual display both for daytime AND nighttime hours.
2. …for my payment methods to be remembered.
3. …that if I am searching for a course with no tutors in the system, I will be alerted.
4. …for a way of informing Tutor Dash if my desired course is not in the system, so they can alert potential tutors.
5. …for a refund on my online session if there are technical difficulties preventing my session from occurring properly.

User Stories - Private Tutor

As a private tutor, I need...

1. ...my qualifications to be based on my previous coursework.
2. ...to receive notifications when potential tutees message/hire me.
3. ...my pay-rate to be as competitive as possible so I don't wait too long to get hired.
4. ...the option to receive alerts when potential tutees in the network are seeking tutoring in courses I tutor.
5. ...a calendar to maintain and update my availability at any time, which potential tutees can publicly view.
6. ...the ability for potential tutees to view my user profile.
7. ...the ability for potential tutees to message me before scheduling a session.
8. ...my ratings to be given only by tutees which I have tutored.
9. ...a mechanism for confirming a session did, in fact, occur.
10. ...payment to be handled within the application.
11. ...my charging pay-rate to be based partly on my experience.
12. ...the ability to challenge poor/negative ratings and/or reviews.
13. ...a mechanism for confirming a session did, in fact, occur.
14. ...the ability to rate tutees based on my experiences with them.
15. ...the ability to refuse service.
16. ...a means of toggling my availability in real-time.
17. ...web-conferencing and in-person tutoring options.
18. ...sensitive account information to be private and secure.

As a private tutor, I wish...

1. ...for a log to keep track of my session and payment history.
2. ...the option to turn off location services if I am only offering online sessions.
3. ...for alerts that tell me classes I don't tutor, but am eligible to tutor are in high demand.
4. ...an alert every X months to remind me to upload an updated transcript so I can tutor more courses.
5. ...for a bonus to tutor courses in high demand if there is a low supply of tutors.
6. ...for an alert before a scheduled session occurs if payment for that session fails to go through.

User Stories - Tester

As a tester, I need...

1. ...to create a variety of mocked up user accounts with semi-automated decision capabilities to simulate an interactive experience.
2. ..a visual log that shows me all the attributes and results of my interactive simulation based on events that I initiate.
3. ...to simulate the signing up both a tutor AND a tutee.
4. ...to simulate a tutee searching for a tutor.
5. ...to simulate a tutor searching for a tutee.
6. ...to mock up data/accounts to simulate various tutors in various courses that exist at ODU.
7. ...to mock up data/accounts to simulate various tutees who would be using the app.
8. ...to design a series of test cases where a tutor/tutee is rated at various different times.
9. ...to design a plan to ensure that pay-rates are affected by weighted ratings, course demand, and time.
10. ...to simulate a tutee requesting to hire a tutor.
11. ...to simulate a tutor denying AND a tutor accepting a tutee's request.
12. ...to simulate a tutee who has sent an alert into the network looking for a tutor.
13. ...to simulate a tutee who cancels after hiring a tutor.
14. ...to simulate the transactions that take place before and after an appointment (deposit and payment respectively).
15. ...to supply a series of mocked up transcripts to the transcript parser that register as official to observe tutoring eligibility based on academic history.
16. ...to supply fake transcripts to ensure the security of the transcript parser.
17. ...to simulate a tutor AND tutee query with both inactive and active users existing in the network.
18. ...to simulate a tutee requesting both an in-person and online meeting.
19. ...to simulate what happens with payments in the case of when both a tutor and tutee agree to a web conference, but there are technical difficulties.
20. ...to simulate a user authenticating themselves via email.
21. ...to simulate the re-authentication process triggered by when a user navigates back into the application from an outside view.
22. ...to simulate a conversation among two users at two distinctive points in time to make sure chat history is retrieved.
23. ...to simulate a series of exact queries in which users who are active and appear in the result are moving away/towards the user searching for them.
24. ...to simulate a meeting in which either (but not both) the tutee or tutor never acknowledge the start of the meeting.
25. ...to simulate the event in which a tutee gets refunded based on a poor experience.
26. ...to simulate a scheduling conflict due to the overlap of calendar events.

As a tester, I wish...

1. ...to provide a test case for when users are currently using the application, and the database fails.
2. ...to run my test suites for every unit of work alongside every build.
3. ...for a tool to aid in the automation of database querying.
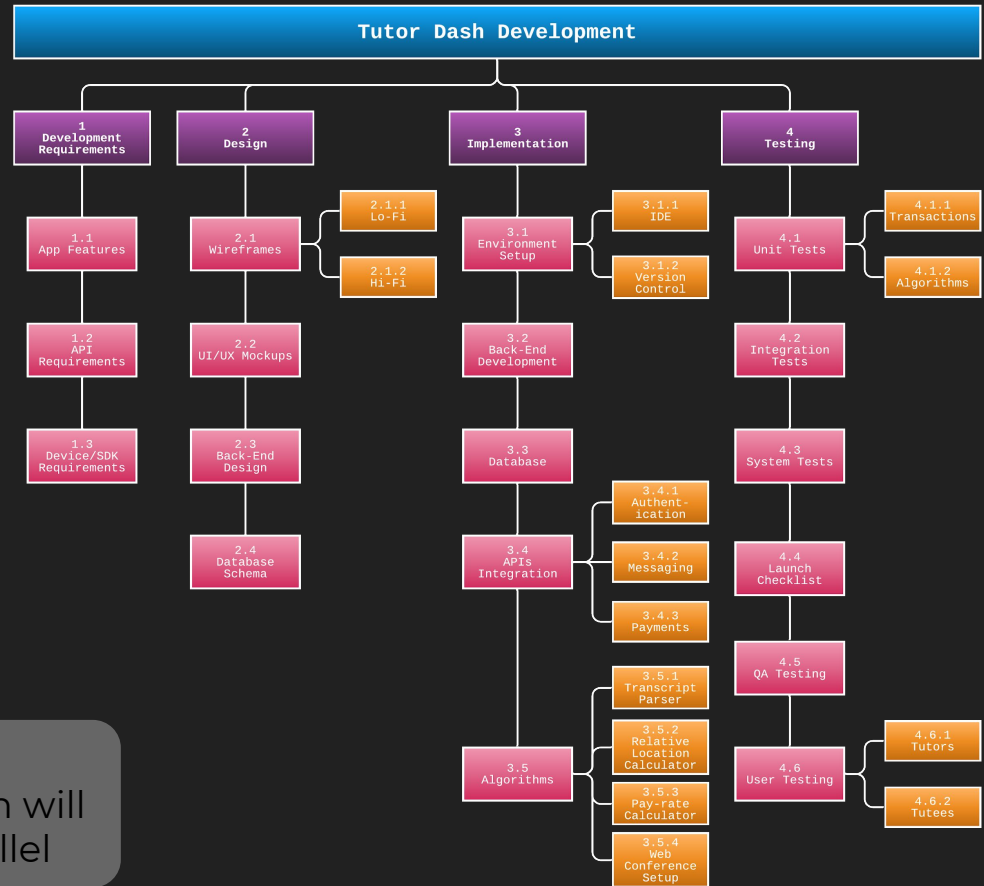
Major Functional Components
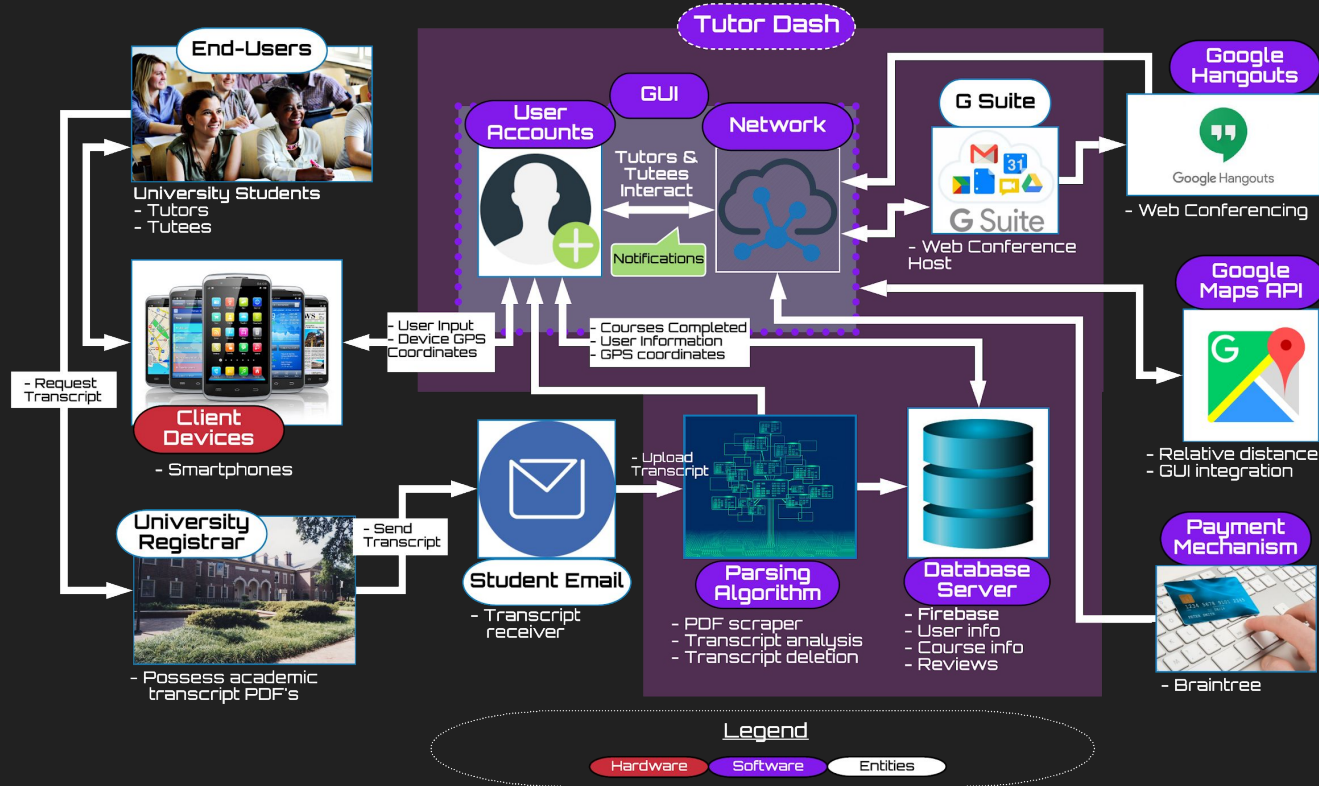
Development Tools

Component	Tool (Android)
Language	Java
IDE	Android Studio
UI/UX	Android Studio
Database	Firebase
Payment Mechanism	Braintree
Build Manager	Gradle
Version Control	GitLab
Continuous Integration/Deployment	GitLab
Issue Tracker	Trello
Testing Framework(s)	JUnit/Espresso/Firebase

Work Breakdown Structure

1. Development Requirements

2. Design

3. Implementation

4. Testing
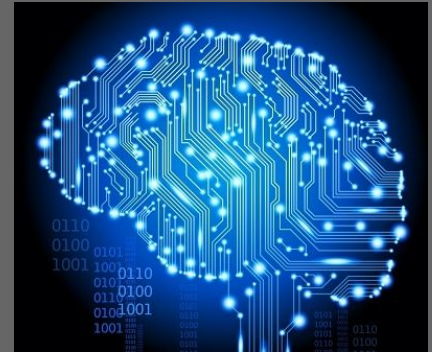
* Testing and Implementation will be done in parallel

Tutor Dash Development

1 Development Requirements
2 Design
3 Implementation
4 Testing

1.1 App Features
2.1 Wireframes — **2.1.1** Lo-Fi / **2.1.2** Hi-Fi
3.1 Environment Setup — **3.1.1** IDE / **3.1.2** Version Control
4.1 Unit Tests — **4.1.1** Transactions / **4.1.2** Algorithms

1.2 API Requirements
2.2 UI/UX Mockups
3.2 Back-End Development
4.2 Integration Tests

1.3 Device/SDK Requirements
2.3 Back-End Design
3.3 Database
4.3 System Tests

2.4 Database Schema
3.4 APIs Integration — **3.4.1** Authent-ication / **3.4.2** Messaging / **3.4.3** Payments
4.4 Launch Checklist

4.5 QA Testing

3.5 Algorithms — **3.5.1** Transcript Parser / **3.5.2** Relative Location Calculator / **3.5.3** Pay-rate Calculator / **3.5.4** Web Conference Setup
4.6 User Testing — **4.6.1** Tutors / **4.6.2** Tutees

Core Components

UI/UX

Database

Algorithms

UI/UX

UI/UX Breakdown

Tutor Dash's UI/UX design is broken down into 4 phases and 2 views:

- **Phases**
  1. Login/Sign up
  2. Tutor/Tutee Discovery
  3. Session Selection
  4. Active Session

- **Views**
  1. Settings
  2. Profile Edit

"Phase" implies sequential ordering, while "View" implies no such ordering.

UI/UX Big Picture

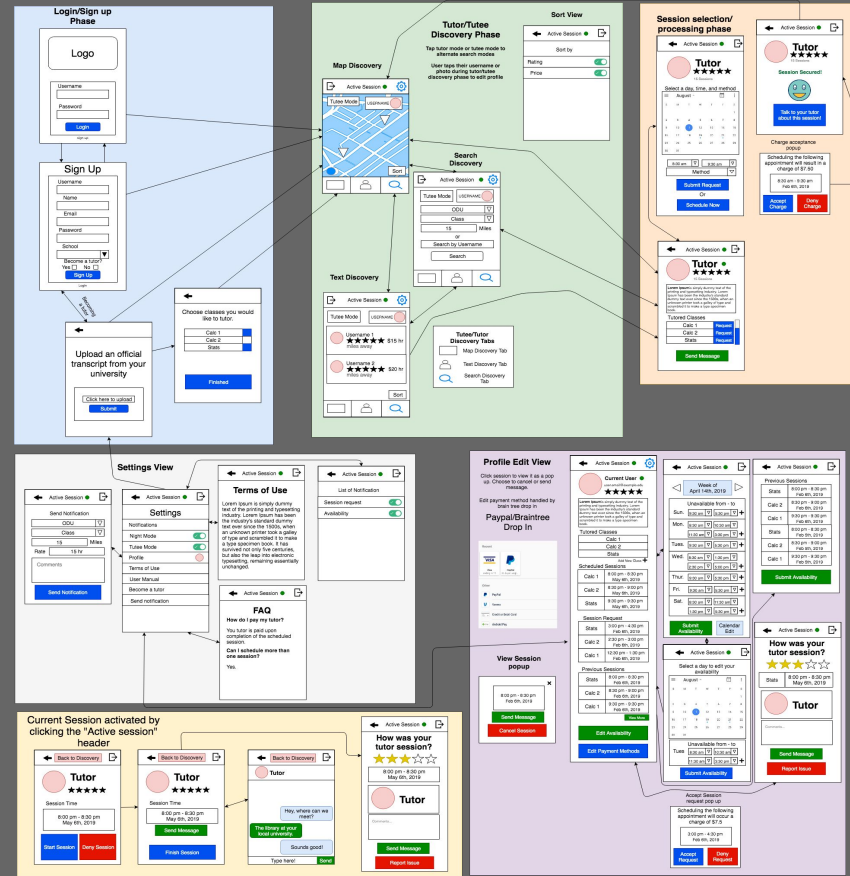| Blue	Login/Sign Up
Green	Tutor/Tutee Discovery
Orange	Session Selection
Yellow	Active Session
Grey	Settings
Purple	Profile Edit

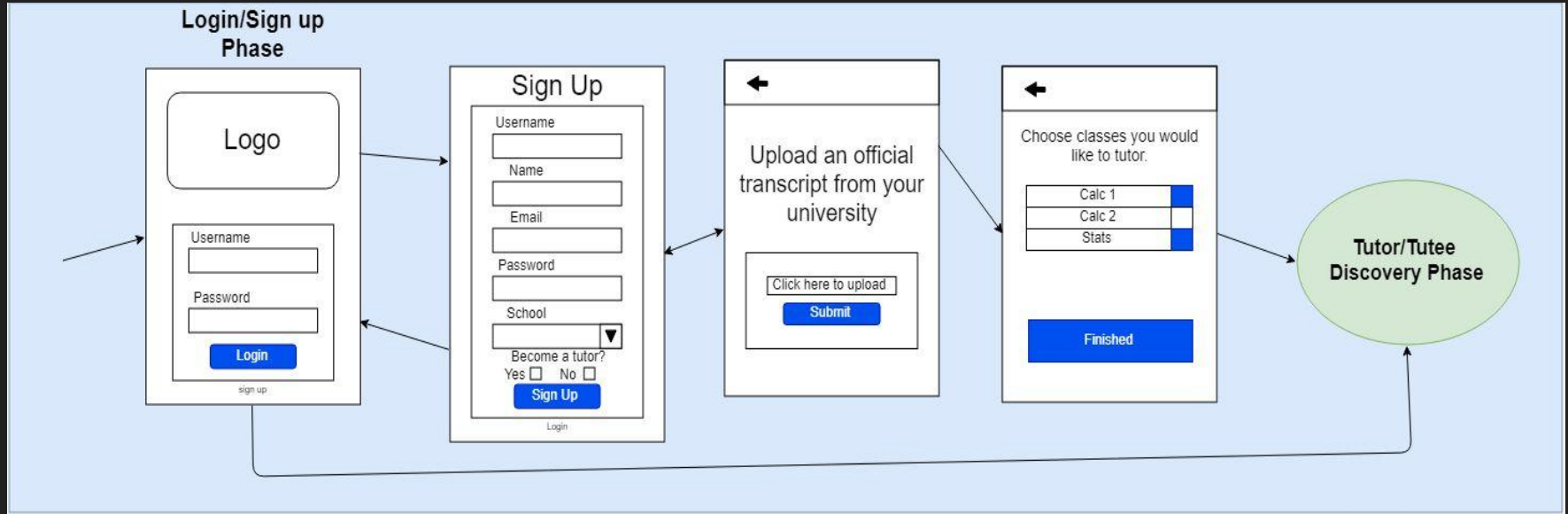CLICK HERE to view enlarged image

Phase 1 - Login/Sign Up

The Login/Sign-Up Phase allows the user to do the following:

- Sign up and register as a user (either tutee or tutee and tutor)

- Login to registered account

- Upload transcript for tutor authentication

- Initialize personal list of tutored classes from list of all permitted classes

Phase 1 - Login/Sign Up UI

Phase 2 - Tutor/Tutee Discovery

The Tutor/Tutee Discovery Phase allows users to do the following:

- Search for tutors or tutees based on mode through map or text

- Change view modes from map/text

- Change or sort content of map or text view through search criteria

- Search for a specific tutor in search view

- Change status to or from active
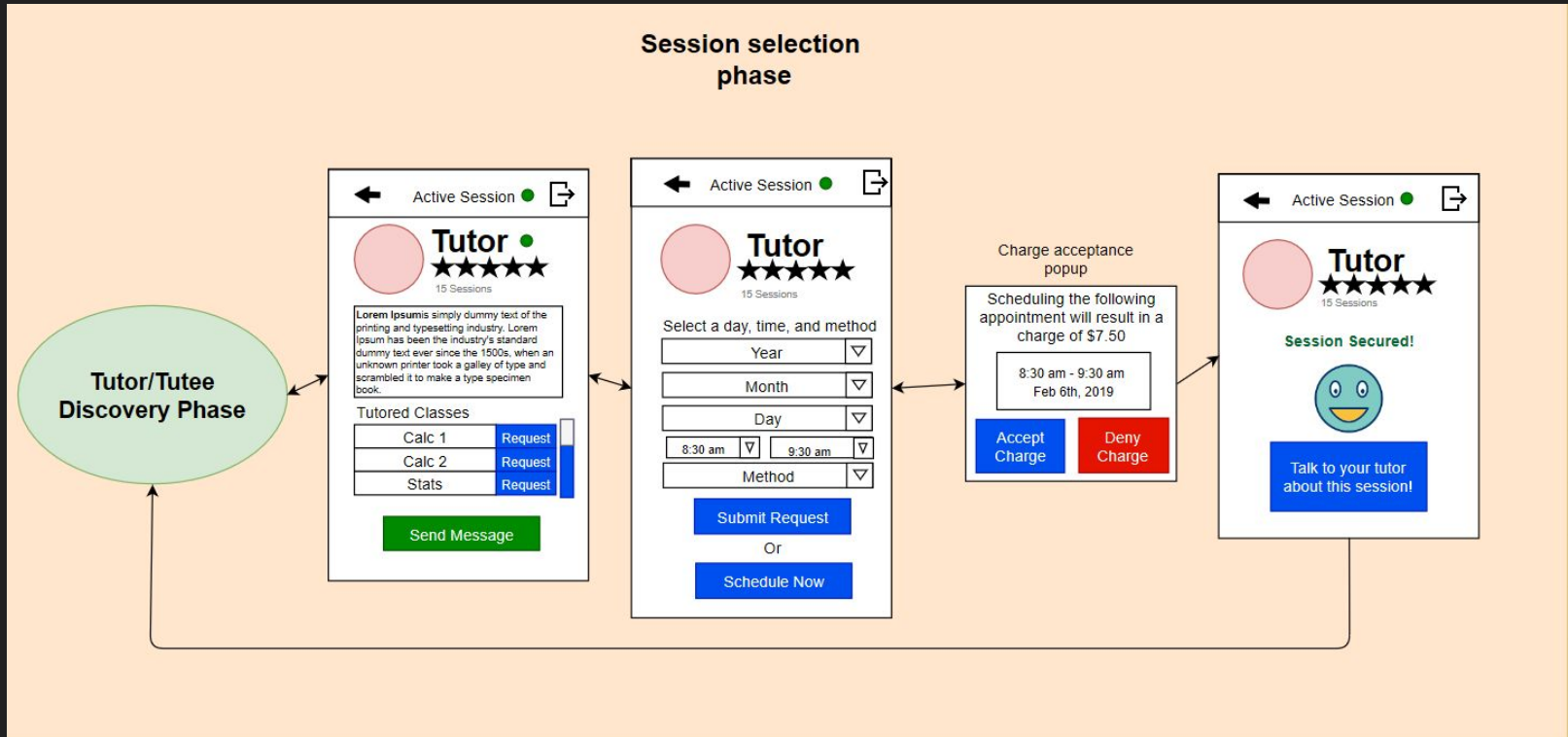
Phase 2 - Tutor/Tutee Discovery UI
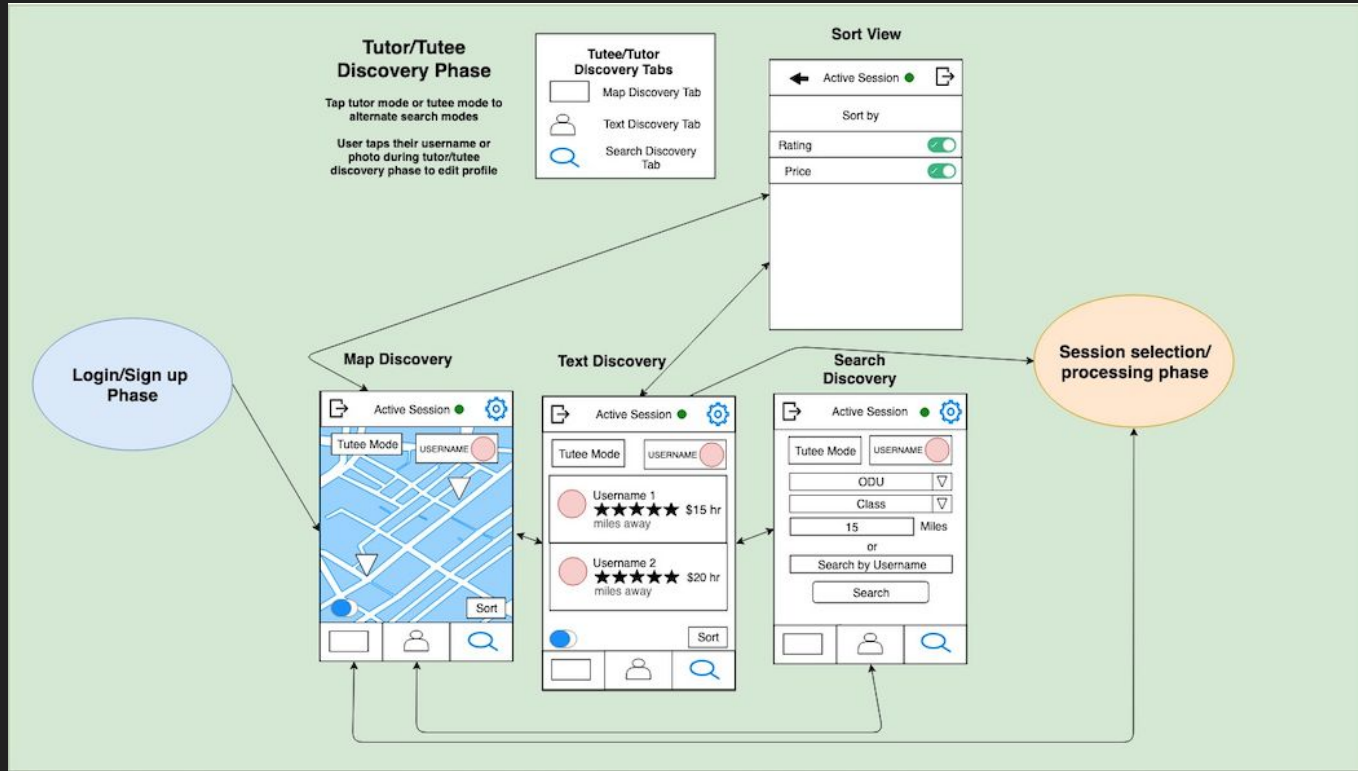
Phase 3 - Session Selection

The Session Selection Phase gives the user the ability to do the following:

- View user information:
    - Username
    - Ratings
    - Biography
    - Classes available for tutoring

- Send messages to other users

- Schedule sessions from a tutor's available times

- Schedule sessions ASAP (in real-time)

Phase 3 - Session Selection UI

Phase 4 - Active Session (Part A)

- The Active Session Phase is accessible from any screen where the "Active Session" beacon is presented.

- The beacon's availability is dependent on the user's scheduled sessions.

- Users may toggle in and out of their active sessions to and from the phase/screen they entered from.

Active Session Beacon
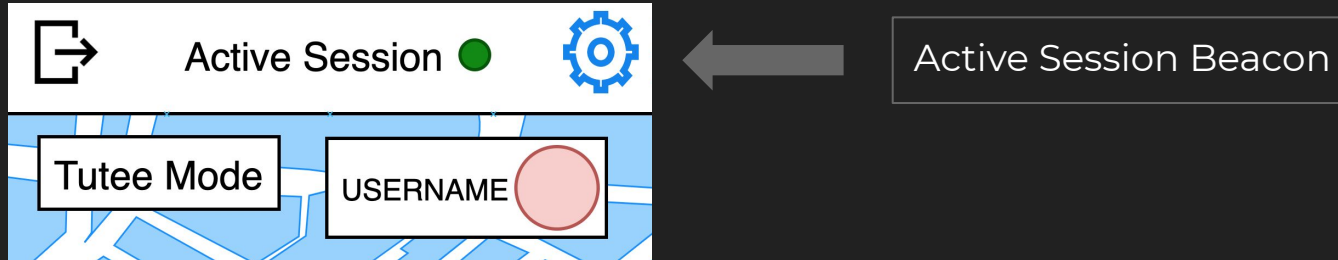
Phase 4 - Active Session (Part B)
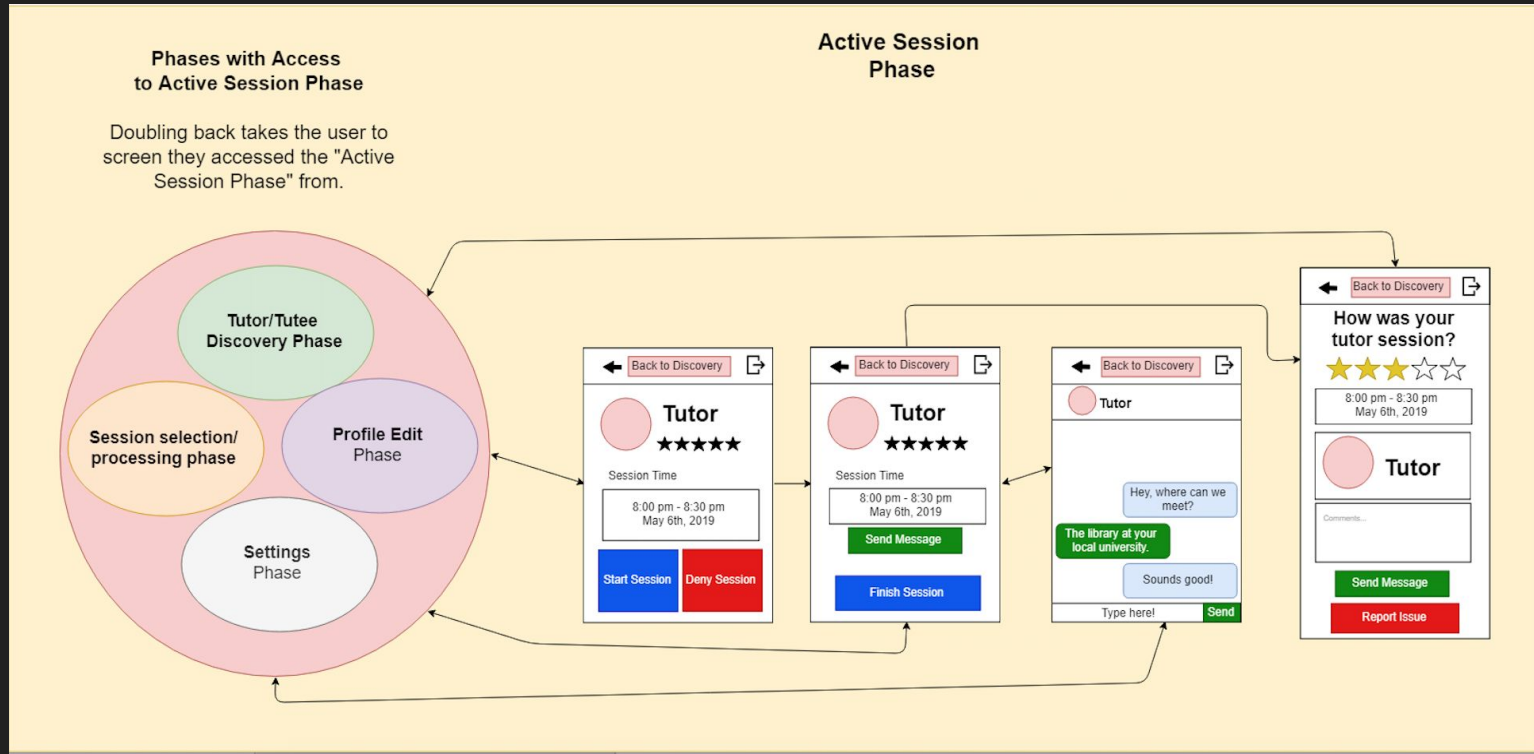
The Active Session Phase gives the user the ability to do as follows:

- Begin scheduled tutoring session.

- Communicate through in-app messaging system to coordinate session details with tutor.

- Dictate end of tutoring session upon completion.

- Rate tutor out of 5 stars and review them for the. recently-finished session

- Report issue with current session and/or tutor.
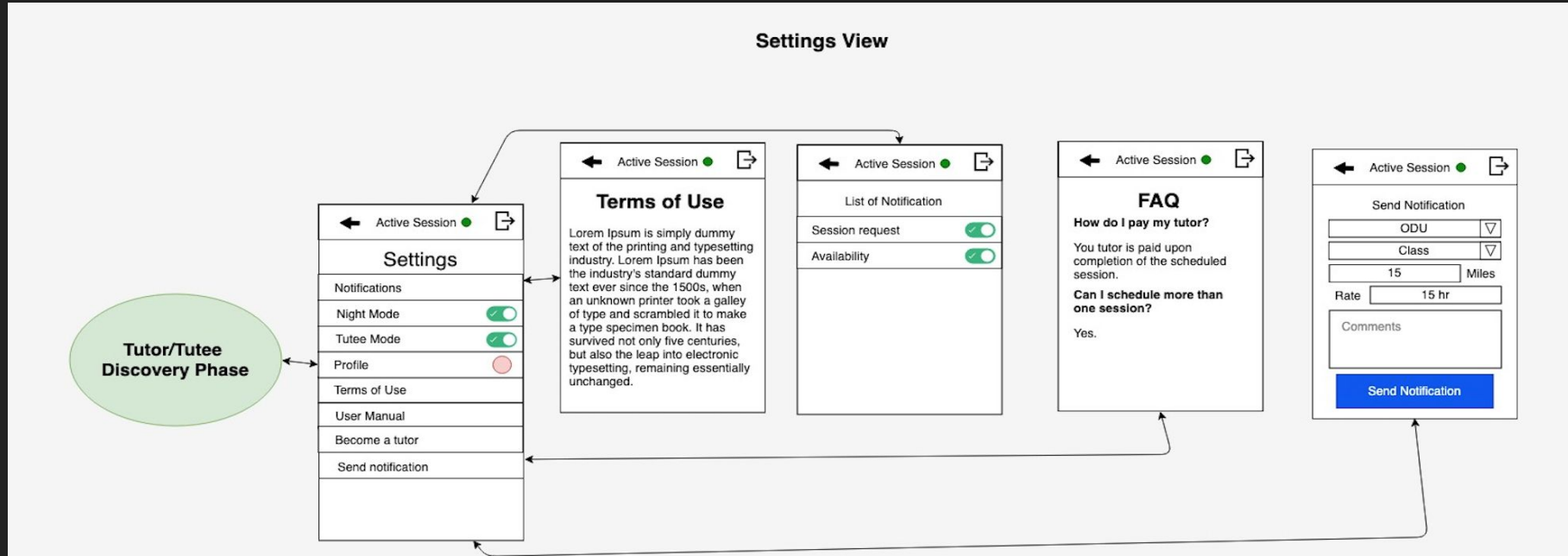
Phase 4 - Active Session UI

View 1 - Options & Settings

The Options and Settings View gives the user the ability to do as follows:

- Modify Notifications
- Toggle Night Mode
- View Calendar for future sessions
- View their profile
- View the 'Terms of Use' policies
- View user's manual
- Become a tutor
- Send out local notifications

View 1 - Options & Settings UI

Settings View

View 2 - Profile Edit
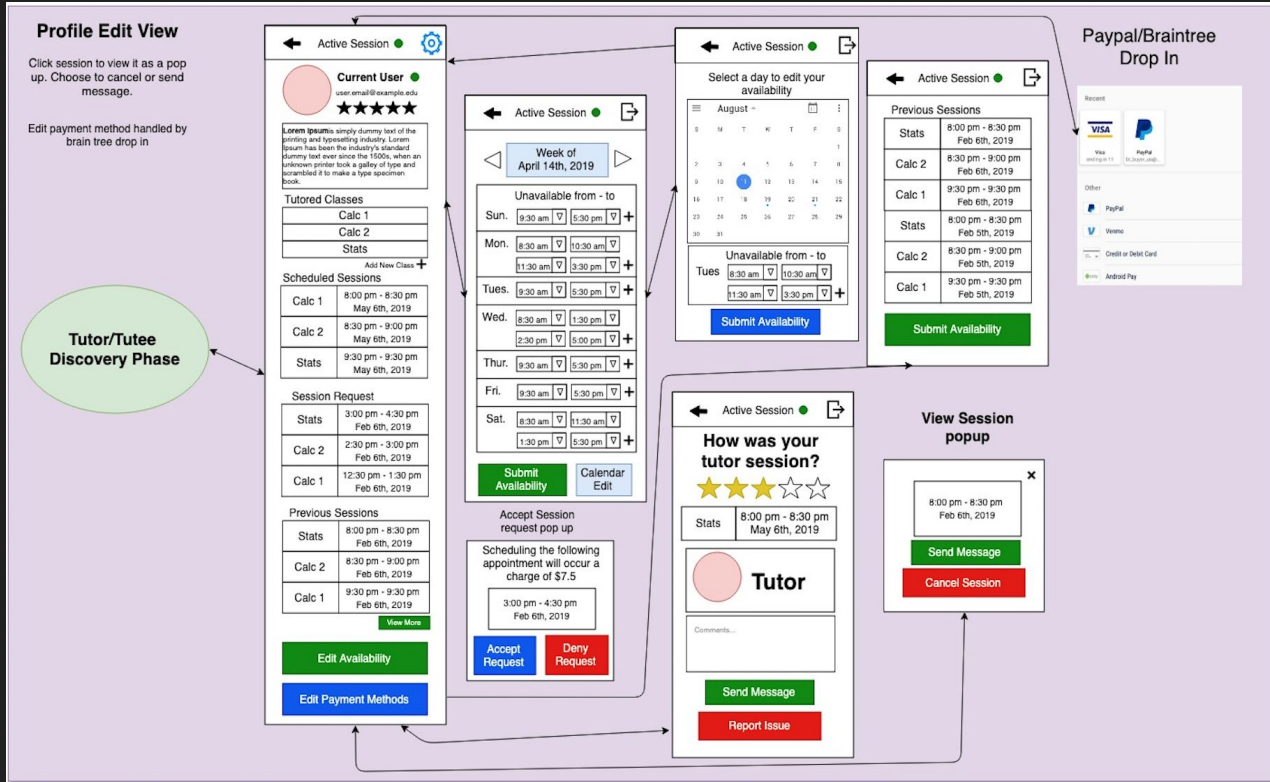
The Profile Edit View gives the user the ability to do as follows:

- View their account information:
    - Username
    - Email
    - Rating
    - Biography
    - Tutored classes
    - Scheduled session(s)
    - Session request(s)
    - Previous session(s)
    - Availability

- Modify any of the aforementioned elements other than personal rating

View 2 - Profile Edit UI

Database

Firebase

- Firebase is a mobile and web application development platform.

- Tightly integrated with Google Cloud Platform

- Tutor Dash will use the following Firebase products to handle backend services:
  - *Cloud Firestore* - Database
  - *Firebase Auth* - User Authentication
  - *Cloud Storage* - File Storage

Used By:
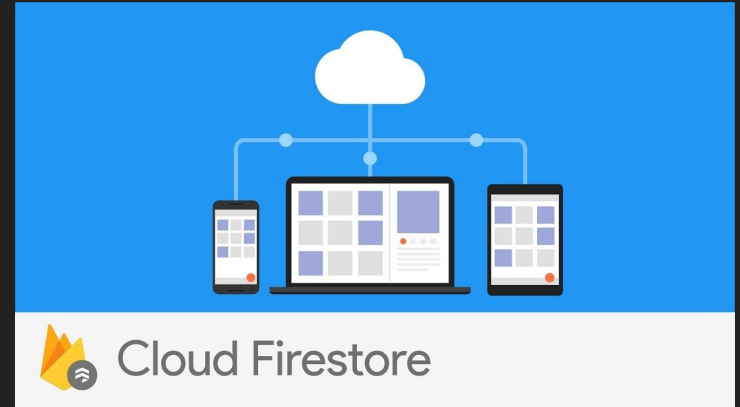
Cloud Firestore

- NoSQL document-oriented database

- Keeps data synced across all client devices in real time

- Designed for automatic scalability

- Database will store user information, school information, schedules, messaging, and reviews.

Database Schema

- Blacklisted users' emails are added to the 'Blacklist' table once they get banned. Their account will subsequently be deleted.

- Payments options are not stored in the DB, but payment history is stored.

User
- *UID*
- *uName*
- *fName*
- *lName*
- *email*
- *picURL*
- *schoolID*
- *isTutor*
- *isAvail*
- *coursesOffered*
- *coursesEligible*
- *coursesPayRate*
- *tutorRating*
- *tuteeRating*
- *inPerson*
- *webConf*
- *location*
- *bio*
- *timesSinceRequest*
- *courseHours*
  - *courseID*

School
- *schoolID*
- *schoolName*
- *schoolSuffix*
- *courses*

Courses
- *courseID*
- *courseName*
- *meanPayRate*
- *stdDev*

Chat
- *UID1_UID2*
- *senderName*
- *sendeeName*
- *message*
- *timestamp*

Blacklist
- *email*
- *schoolID*

Reviews
- *UID*
- *reviewerUID*
- *rating*
- *comment*
- *timestamp*

Payments
- *UID*
- *receiverUID*
- *dateTime*
- *amount*

Schedule
- *schoolID_UID*
- *date*
  - *eventID*
    - *eventName*
    - *startTime*
    - *stopTime*

User Authentication & Cloud Storage

- **User Authentication**
  - Authentication allows users to create an account using email and password.
  - Capable of sending email confirmation to activate account.
  - Integrates with Cloud Firestore for secure DB access.

- **Cloud Storage**
  - Capable of storing user generated content such as photos and videos.
  - High scalability. Same infrastructure that powers Spotify and Google Photos.

Braintree - Payments

- Collection of cross-platform tool sets that allow developers to include multiple payment options safely within their applications

- Include drop-in UIs

- Managed by PayPal

- Used by several well-known companies such as Uber and Poshmark

- *Required toolsets*
  - Braintree Java SDK
  - Braintree Android v3 SDK

Paypal/Braintree Drop In
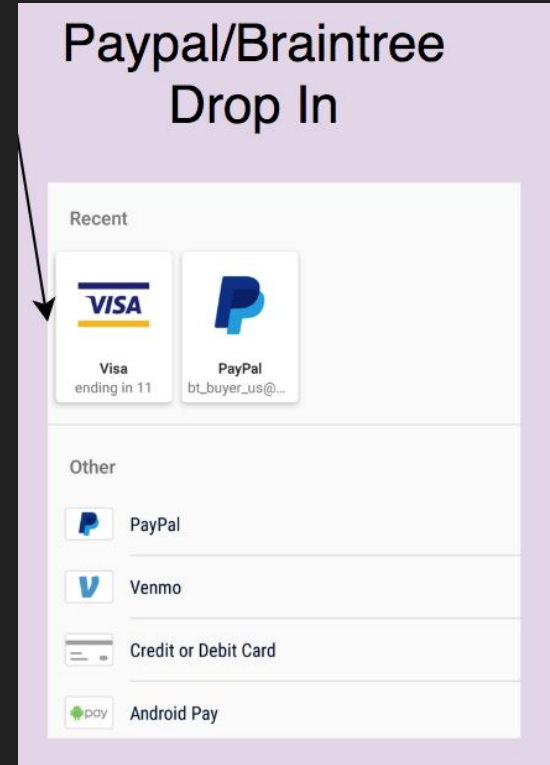
Algorithms

PDF Transcript Parser

- **Purpose**
  - Determine which classes a user is qualified to tutor

  - Add new courses being tutored to the database

- **Tools**
  - PDFBox Java Library

- **Parameters**
  - University name
  - Transcript (PDF)
  - Minimum qualifying grade
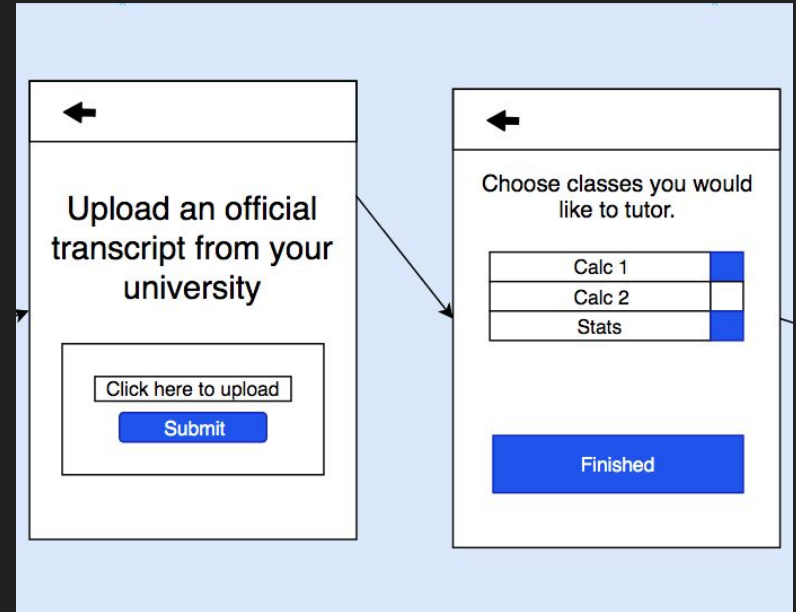
PDF Transcript Parser - Logic Flow

User creates Tutor Dash account

User uploads transcript PDF

Parser

Read in entire PDF as a string

Search for offical university watermark

Locate course id's and grades using regular expressons

Store course id's in a map<course, grade>

if grade < minimum requirement, remove entry from map

Display map to user as list

User Account

User selects desired courses from list

Add any new selected courses to database

Delete transcript

Template transcript for university is provided

Database

Legend

Prerequisite | Algorithm | Entity

Pay-Rate Calculator

- **Purpose**
  - Keep pay-rates competitive by providing an upper bound based on various factors
  - Alleviate the possibility of tutors not getting hired often enough

- **Parameters**
  - Tutor rating (course-specific)
  - Tutor rating (overall)
  - Course demand/popularity
  - Tutor's Experience
  - Time of day
  - Mean & standard deviation of rates for courses
  - Time since tutor's last request in that course

Pay-Rate Calculator - Logic Flow

Time

Database

User Account

Rerun x times per week
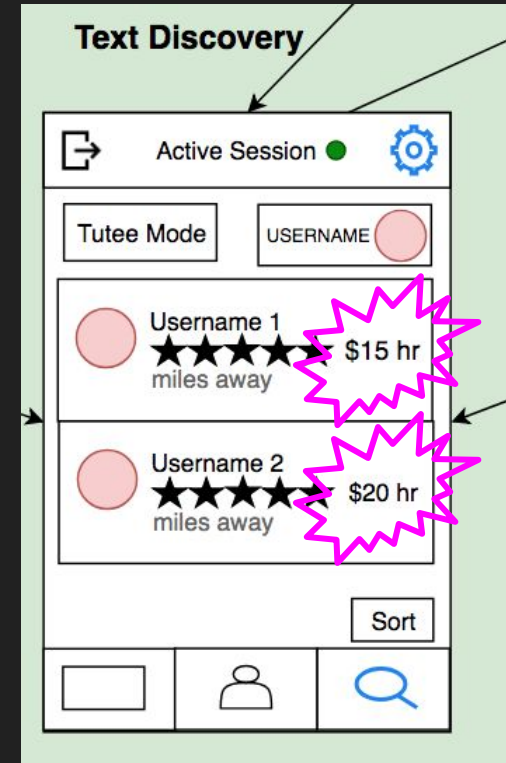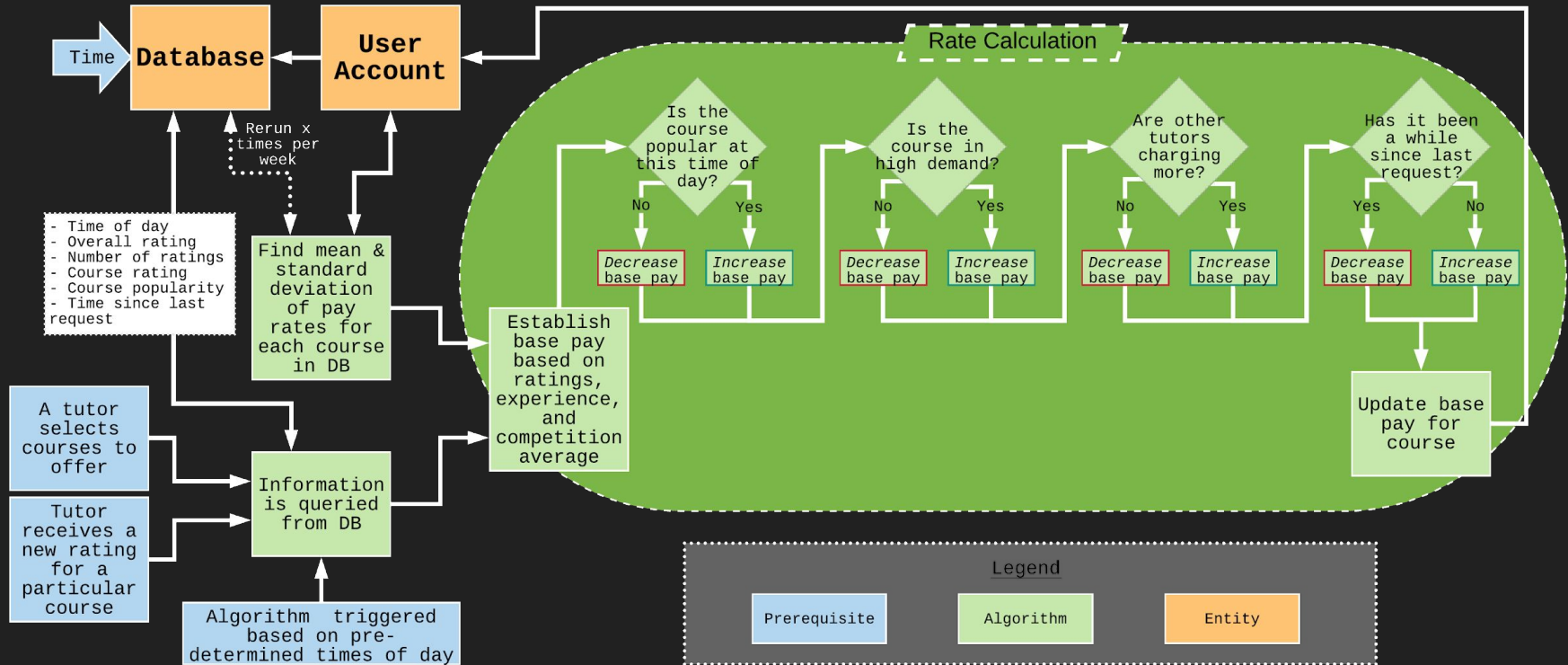
Rate Calculation

Is the course popular at this time of day?
No   Yes

Is the course in high demand?
No   Yes

Are other tutors charging more?
No   Yes

Has it been a while since last request?
Yes   No

- Time of day
- Overall rating
- Number of ratings
- Course rating
- Course popularity
- Time since last request

Find mean & standard deviation of pay rates for each course in DB

Decrease base pay   *Increase* base pay

Decrease base pay   *Increase* base pay

Decrease base pay   *Increase* base pay

Decrease base pay   *Increase* base pay

Establish base pay based on ratings, experience, and competition average

Update base pay for course

A tutor selects courses to offer

Tutor receives a new rating for a particular course

Information is queried from DB

Algorithm triggered based on pre-determined times of day

Legend

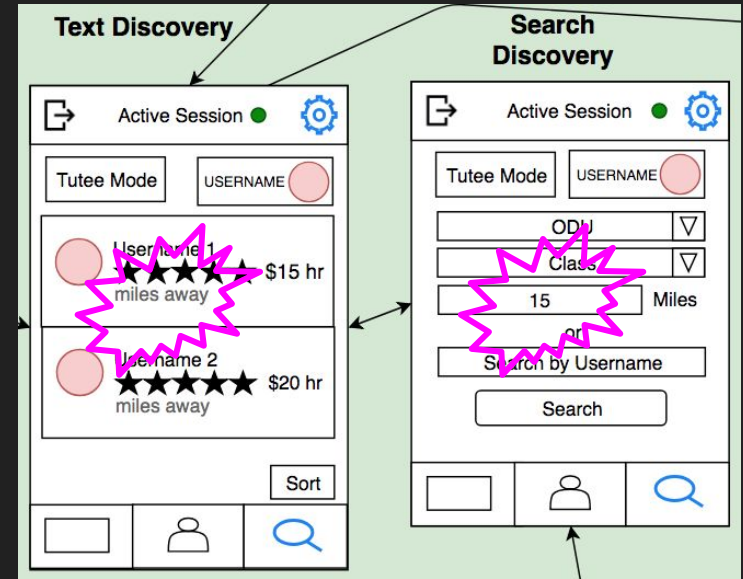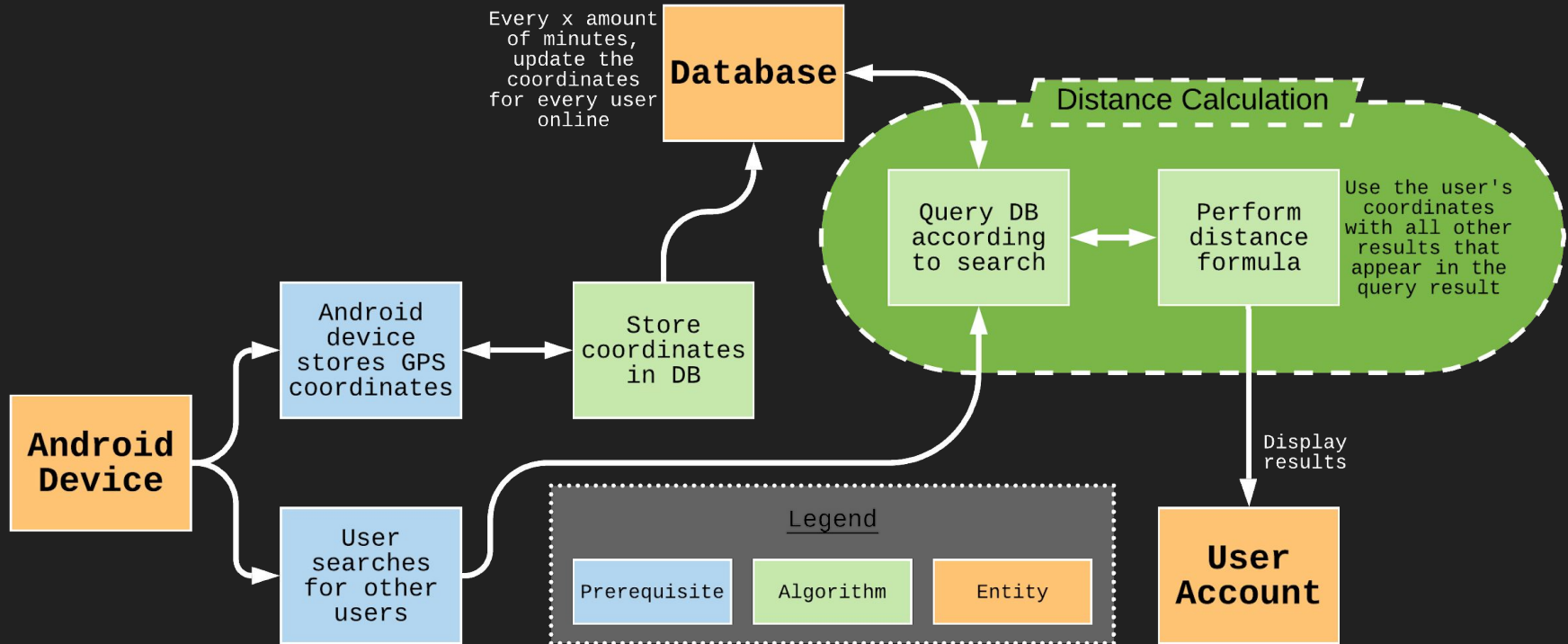Prerequisite   Algorithm   Entity

Relative Distance Estimator

- **Purpose**
  - Display how far a set of users B is from user A based on user A's search results.

  - Keep the distance information updated as often as possible

- **Parameters**
  - Time
  - Time interval for updates
  - Android device's GPS coordinates
    - User A's longitude/latitude
    - Every user in B's longitude/latitude

Relative Distance Estimator - Logic Flow

Every x amount of minutes, update the coordinates for every user online

Database

Distance Calculation

Query DB according to search ↔ Perform distance formula

Use the user's coordinates with all other results that appear in the query result

Android device stores GPS coordinates

Store coordinates in DB

Android Device

User searches for other users

Display results

User Account

Legend

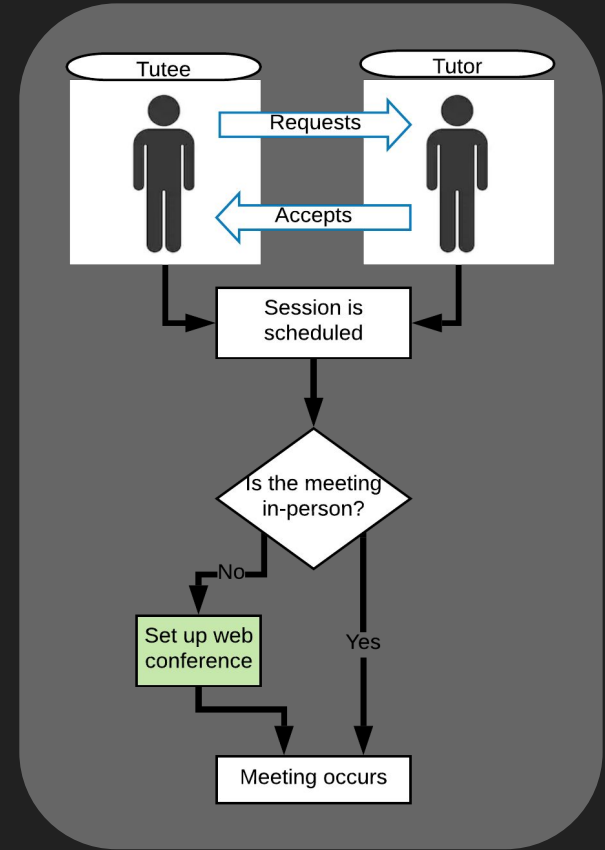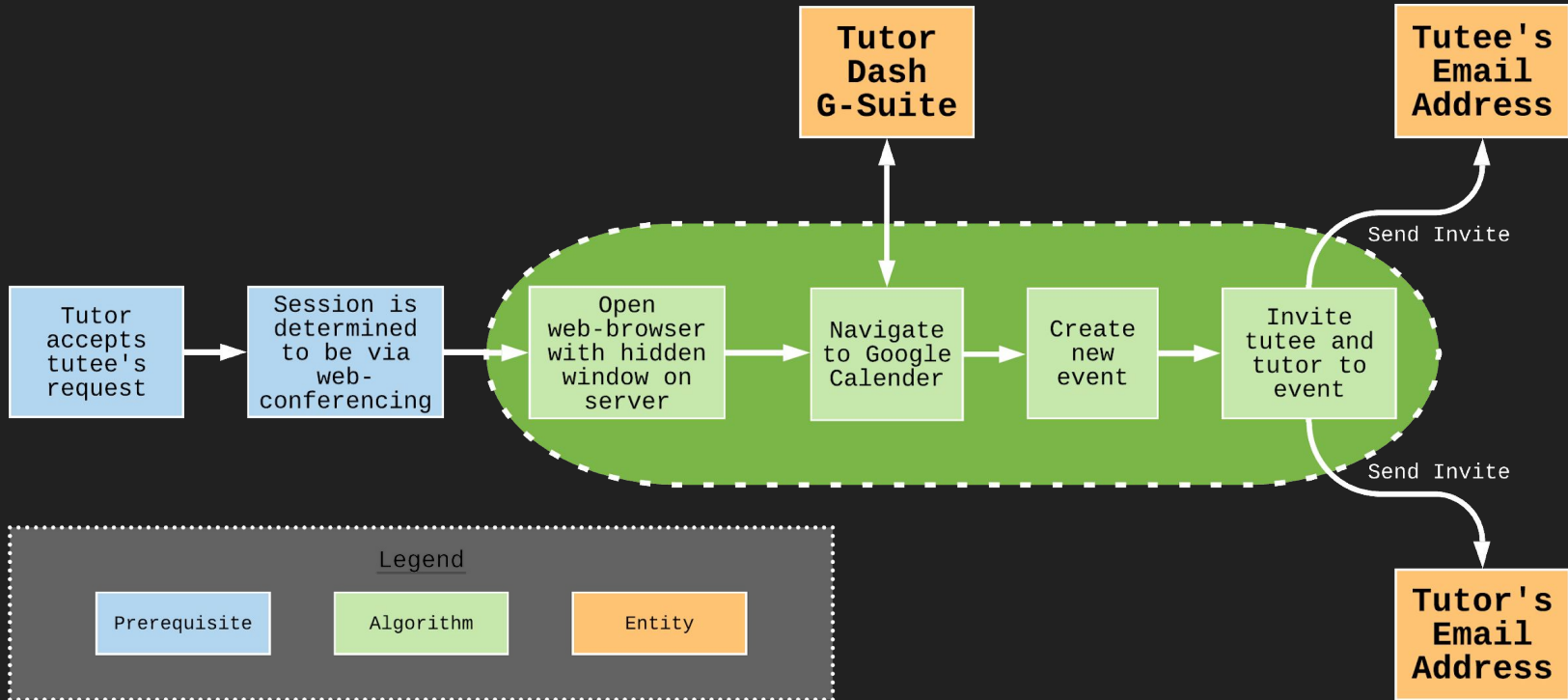| Prerequisite	Algorithm	Entity

Web Conference Creator

- **Purpose**
  - Create a google hangouts meeting for two user's if the tutoring session in question is via web-conferencing

- **Tools**
  - Selenium Webdriver (Java)

- **Parameters**
  - Start time of session
  - End time of session
  - User email addresses
    - Tutor
    - Tutee

Web Conference Creator - Logic Flow

Tutor Dash G-Suite

Tutee's Email Address

Tutor accepts tutee's request → Session is determined to be via web-conferencing → Open web-browser with hidden window on server → Navigate to Google Calender → Create new event → Invite tutee and tutor to event

Send Invite

Send Invite

Tutor's Email Address

Legend

Prerequisite   Algorithm   Entity

Hardware System Requirements

- Android API 16 (JELLY_BEAN) is required for both Firebase and Braintree

- Devices must have GPS capabilities

Risk Matrix

- "C" → Customer Risks
- "T" → Technical Risks
- "L" → Legal Risks

Probability

	Very Low	Low	Moderate	High	Very High
Very High	T3, T4, L1	T6, L2		C3, C4	
High	T1, C7	C6, T9			
Moderate	C5	T8	C1, C8, T7		
Low	C2, T5		C10, T2		
Very Low		C9			

Impact

Risks - Customer Risks (Part 1)

ID	Risk	Mitigation(s)
C1	Student finds tutors to be unhelpful	• Rating system • Payment refunds
C2	Prospective tutors faking their qualificaitons	• Require official transcript from university registrar • Make tutors only eligible to tutor classes he/she has received a B or higher in
C3	Shortage of tutors	• Give small bonuses to tutors for a limited time (similar to Uber's business model)
C4	Shortage of tutees	• Give free sessions to new users • Give loyaltee-free sessions for a certain number of usages
C5	Tutor/tutee leaves a false negative review	• Allow users to challenge reviews (requires manual investigation) • Withhold ratings/reviews until both users agree on justification

Risks - Customer Risks (Part 2)

ID	Risk	Mitigation(s)
C6	Users abuse application; use app maliciously	• Require users to agree to the terms of use agreement • Blacklist (ban) users who violate terms of use agreement
C7	Identity theft; non-users impersonate users and/or users impersonate other users	• Re-authentication when navigating to app from outside window • "Handshake" agreement between users when sessions begin
C8	Participating tutor/tutees don't show up to their scheduled meetings	• Preallocate payments • Require deposits • Threat of poor ratings
C9	Users try to book overlapping sessions	• Only allow users to make appointments for times they don't currently have a scheduled session • Applies to both tutors and tutees
C10	Tutors are not adequately prepared to engage with tutees via web conferencing	• Alert users of the minimum requirements for web conference meetings upon selecting 'web conferencing' as a tutoring preference.

Risks - Technical Risks

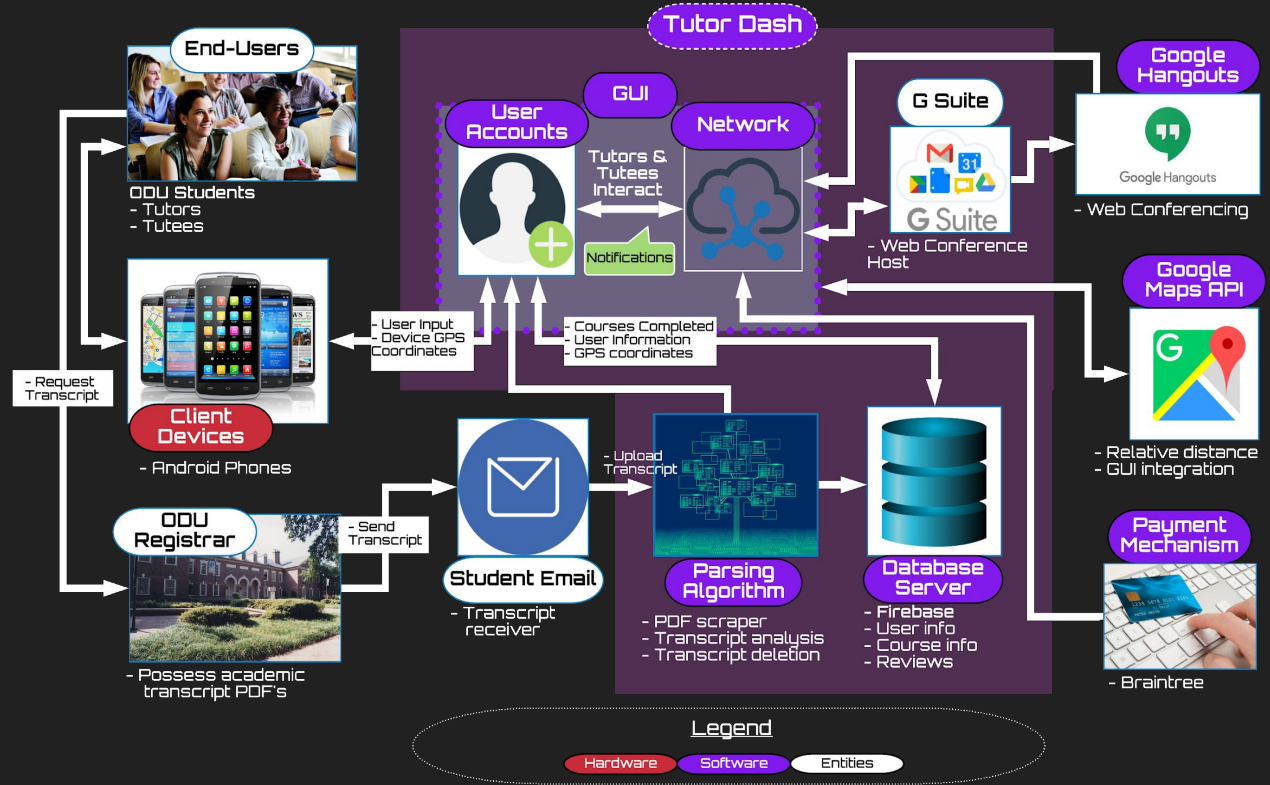| ID	Risk	Mitigation(s)
T1	Payment is not received	• Integrate usage of a 3rd party API designed to handle e-transactions • Braintree
T2	Difficulty automating the process of reading a submitted transcript	• Define reusable code for general case • Optimize as more information is discovered
T3	Database server failure	• Use reliable servers maintained by large corporations • Firebase
T4	Security breach	• Use 3rd party APIs which are already secure
T5	Application is not compatible on all android devices	• Define minimum SDK for weaker hardware phones • Define normal SDK for normal hardware phones
T6	Network server failure	• Server redundancy
T7	Pay-rate algorithm doesn't calculate competitive rates	• Determine a base pay that will increase/decrease due to various factors • Compare pay-rates of similarly rated tutors who tutor the same courses
T8	Web-conferencing session is not set up properly	• Use Google Hangouts • Use one or more G Suite hosts operated by Tutor Dash
T9	Unexpected interruption prohibits online sessions from occurring	• Refund payments in this case as long as both parties arrived to the meeting. • Use a Google Hangouts, a commonly used web conference tool maintained by a large corporation.

Risks - Legal Risks

ID	Risk	Mitigation(s)
L1	Violating The Family Education Rights and Privacy Act (FERPA)	• No portal access • Transcripts are analyzed then thrown out • Users agree to grade disclosure in terms of use agreement
L2	Users use application for illegal activities	• Terms of use agreement • Reporting features

Real World Product vs. Prototype

Functional Element	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	Fully-Functional - Necessary since Tutor Dash targets both day and night crowds
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	Eliminated - Unnecessary for prototype
Support of multiple universities	Fully-Functional	Eliminated
Cross-platform support	Eliminated	Eliminated
Firebase console linked to test suite(s) with mockups	Eliminated	Fully-Functional

Prototype MFCD

- All major functional components remain

- User-base constrained to ODU students

- Hardware constrained to Android smartphones

End-Users

ODU Students
- Tutors
- Tutees

Client Devices
- Android Phones

ODU Registrar
- Possess academic transcript PDF's

- Request Transcript

- Send Transcript

- User Input
- Device GPS Coordinates

Student Email
- Transcript receiver

Tutor Dash

GUI

User Accounts

Tutors & Tutees Interact

Notifications

Network

G Suite
- Web Conference Host

- Courses Completed
- User Information
- GPS coordinates

- Upload Transcript

Parsing Algorithm
- PDF scraper
- Transcript analysis
- Transcript deletion

Database Server
- Firebase
- User info
- Course info
- Reviews

Google Hangouts
- Web Conferencing

Google Maps API
- Relative distance
- GUI integration

Payment Mechanism
- Braintree

Legend

| Hardware | Software | Entities |

Testing - Core Components (Unit Testing)

Approach: Test each core component independently first.

- UI/UX

    - Espresso - Built into Android Studio, this serves as the primary tool that android developers use to automate the testing of UI's.

- Database

    - Firebase Test Lab - A cloud-based app-testing infrastructure that allows developers to view results including logs, videos, and screenshots in the Firebase console.

    - JUnit - Will be used to query the database for expected output.

- Algorithms

    - JUnit - A popular testing framework for Java that allows developers to automate tests and integrate them into the build when necessary.

Testing - Integration Testing

Approach: Merge core components one by one to ensure compatibility.

- Most likely will be automated using a combination of the unit testing tools.

- Since the major functionality of Tutor Dash is the database, Firebase Test Lab will be used to query information which our team will mockup to test the behaviors of the queries.

- Eventually, once all core components are integrated into the build, these tests evolve into system tests.

Testing - System Testing

Approach: Test the system as a whole using virtual & physical devices.

- Firebase Test Lab

  - Tutor Dash's networking functionality is almost entirely dependent on the database. Because of this, the database must be tested extensively to ensure that it will run on all target devices.

- Android Emulator

  - Android Emulator is a part of Android Studio that allows developers to choose from a variety of emulators. This will ensure that Tutor Dash fits our minimum SDK requirements for usage.

Development Model - Agile

- Work completed in brief time iterations (sprints)

- Testing implemented throughout development

- Focus on observable behavior
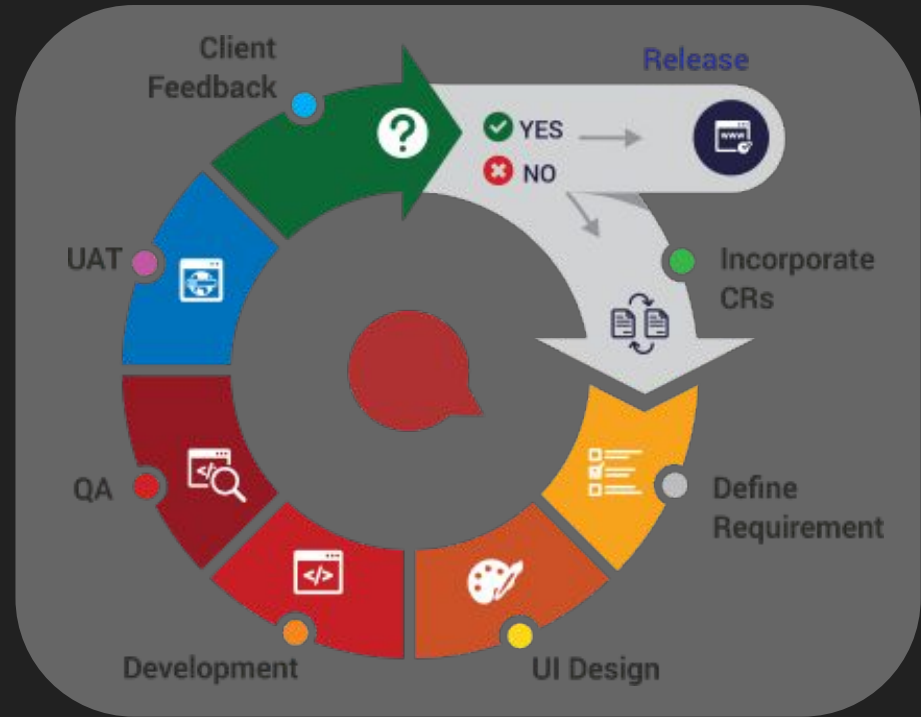
Source: *Intelegrain Technologies*

Agile Sprints

The Tutor Dash prototype will be divided up into 8 sprints over 11-16 weeks, each with their own focus:

1. Environment Setup: Set up DB and make sure we can query information.

2. UI/UX: A substantial part of the UI will exist. Incorporate that into backend.

3. Link Front/Back-Ends: Payments/User Auth. will be nearing completion. Significant progress on algorithms. Link UI with DB and test.

4. Algorithms: Keep making progress on algorithms. Continue testing algorithms and DB.

5. UI/UX & Testing: All UI/UX phases/views will exist in one form or another. Keep testing, and incorporate into DB.

6. Database Finalization: Finalize the DB server and finish its testing. Begin system tests.

7. UI/UX Completion: Link all backend with every UI/UX frame. Focus on integration testing.

8. Testing Finalization: Testing needs to be finalized. Unit tests should be passing, and system tests will reflect functional reqts. and hardware specifications.

Development Schedule

#	Time Frame: weeks	Focus	A. DB Server	B. User Auth.	C. Brain-tree	D. UI Phase 1	E. UI Phase 2	F. UI Phase 3	G. UI Phase 4	H. UI View 1	I. UI View 2	J. Algor. 1: PDF Parser	K. Algor. 2: Pay-calc.	L. Algor. 3: Dist. est.	M. Algor. 4: Web conf.	N. Test DB	O. Test UI	P. Test Algor.	Q. Test Sys.
1	1	A, D, J	S			S						S				S	S	S	
2	1-2	B, D, E, O, P	C	S		C	S					C	S			C	C	C	
3	1-2	B, C, D, J	C	F	S	F	C	S				F	C	S		C		C	
4	1-2	K, N, O	C		C		C	C	S		S		C	C		C	C	C	
5	2-3	C, E, K, L, O	C		F		F	F	C	S	C		F	F	S	C	C	C	
6	2	A, N, O, P, Q	F			R			C	C	C				C	F	C	C	S
7	2-3	G, H, I, M, O, Q	R	R			R	R	F	F	F	R	R	R	F		F	C	C
8	1	P, Q	R		R	R	R	R	R	R	R	R			R	R	R	F	F

S = "Start"   C = "Continue"   F = "Finish"   R = "Refine"

Alex | Brandon | John | Duncan | Jamauni | Dwight | Not Assigned

Benefits to Customer/User Base

- Expands the scope of any university's course-specific tutoring program(s).

- Decreases availability conflicts.

- Tutors aren't waiting around for responses.

- Reduces frequency of cancellations by providing real-time scheduling.

- As tutoring increases, DFWI (Drop/Fail/Withdraw) rates decrease. Students save GPA and money.

- Students won't have to go out of their way in search of tutoring services since they are contained conveniently in one place.

- Alleviates frustration in regards to contradictory information. Improves information consistency.

Long-Term Goals

- Various university support

- Active community 24/7

- Every class offered at every university in the system has at least one eligible tutor

- Repeat/long-term customers

In Summary

- *Our solution aims to engage university students and help them find what they need in an easier and more convenient format than what currently exists.*

- Some students need tutors

- Some students want to tutor

- Tutor Dash makes the connection obvious

References - University Tutoring Resources

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

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

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

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

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

References - Student Behaviors

6. 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.

7. 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=Ema ...

8. 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.

9. 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.

References - Student Behaviors cont.

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

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

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

13. Pierce, Dennis. "Supporting Students Beyond Financial Aid", 2016 URL:
    http://eds.b.ebscohost.com.proxy.lib.odu.edu/ehost/detail/detail?vid=0&sid=d93df6c4-3729-4b6
    2-8d58-95e25c309878%40sessionmgr102&bdata=JnNpdGU9ZWhvc3QtbGl2ZSZzY29wZT1zaXRl
    #AN=114789419&db=ehh.

References - Competition

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

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

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

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

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

19. "Tutor.com - The Princeton Review." *Tutor.com*, The Princeton Review, March 2019. URL: www.tutor.com.

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