Lab 2 – Tutor Dash Product Specification

Duncan J. Holterhaus

CS 411

Professor T. Kennedy

17 October 2019

Version 1

TOC HERE

1 Introduction

Universities have a need for tutoring on their campuses. A better informed, a more knowledgeable populous, will increase the overall standing on their campus, and invite more people to attend. Universities would like to see higher rates of graduation than other campuses to remain competitive and to receive more funding for their research projects, growth projects, etc. A simple answer to see this change in college campuses would be to reform the way we approach tutoring.

Tutor Dash is aiming to improve the overall market of tutoring on all available college campuses. By using the untapped market of students, Tutor Dash will create a smartphone application that will pull students who have the knowledge and students who need the knowledge together.

1.1 Purpose

The aim of Tutor Dash is to make the whole market of finding a college tutor simpler. By matching the pool of students who have the knowledge with the students who need it, Tutor Dash will attempt to do the following:

1. Improve the tutoring market by increases the number of individuals that can tutor.
2. Improve the overall knowledge base of the student body.
3. Allow tutors to have a source of income from their knowledge.

Tutor Dash is being developed for university students who wish to be tutored, and university students who require tutoring. By putting these two customer bases together, the application will attempt to make a self-sufficient community that can aid each other in improving their knowledge base.

Universities do not have the ability to provide tutoring for all available courses on their course catalog. To provide such tutoring, they would have to invest many resources into expanding their services, which will include tutoring locations and available tutors. The lack of such an investment leads to most courses being cut out, and tutoring hours being limited for the courses that have a tutor available.

Tutor Dash's purpose is to fill the void that universities do not have the capability to fill. The smartphone application will create a centralized platform that will allow all users to navigate to their nearest tutor, so that the wait time to a tutor and the quality of tutor will be significantly improved.

1.2 Scope

The long-term goal of Tutor Dash is to be visible on all campuses across the United States. In order to reach this goal, the team will need to start small. The prototype will only be available for Old Dominion University students because of the customer support required to maintain a viable product. Old Dominion University is a good proving group for any sort of private tutoring application that aims to pull the whole student body together.

Tutor Dash will develop a centralized platform for students to give or receive tutoring services. Universities, specifically Old Dominion University, tend to have multiple websites for their different tutoring services. This can lead to confusion among students due to multiple sources having to be updated or being updated by different individuals. The same can be said for the private tutoring market. There are multiple sources for information among different services, including individuals who are seeking to advertise themselves either on or off campus.

Centralizing the market for tutoring will have various benefits to the student body. With more knowledge, students on average will see higher grade point averages, less course withdrawals, and greater passing rates. If students pass their classes without any trouble, then they won’t have to take out more student loans to retake the courses they failed.

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.

DFWI rates: An abbreviation for Drop/Fail/Withdraw/Incomplete 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.

G Suite: An API created by Google to access all their services.

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

Serverless Architecture: Concerning database interactions over a network server, this type of architecture implies that the server's implementation is invisible (or abstracted) to the team developing the product using the actual server.

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 to 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

"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

Team Gold. (2019, October 7). Lab 1 – Tutor Dash Description. Retrieved October 17, 2019 from https://www.cs.odu.edu/~411gold

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

1.5 Overview

The information detailed in the rest of this document describes the general information about Tutor Dash as a product. The architecture, features, and capabilities will be a focus of discussion and detail.

2 General Description

Tutor Dash will have multiple features pertaining to customer service, real-time scheduling, automatic payment providing, and customer rating. Table 1 displays all functionality that will be present in the both the real world product and the prototype.

[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

2.1 Prototype Architecture Description

The Tutor Dash prototype architecture will be identical to the real-world product. The architecture will have three main components: The Tutor Dash application, the database, and external APIs.

- Tutor Dash application: an application that will contact the other two components based on user need. This can occur when the user requires an API such as the Google Calendar API to make an appoint with another user. The application can also make a call to the database for any user data needed to help find a tutor or a tutee.
- Database: The database will hold all applicable information to the user and will be called upon to either write or read said information. This will include all information involving the user such as user ID, rating, location, etc. It will also hold all information regarding the school, courses, reviews, blacklist, etc.
- The APIs: The APIs will be called on to process different portions of information based on the function calling it. This will include BrainTree for payment processing, Google Suite for all session scheduling, and Google Maps for location-based requirements.

Figure 1 provides all links between these three components and how they interact.

[This space intentionally left blank]

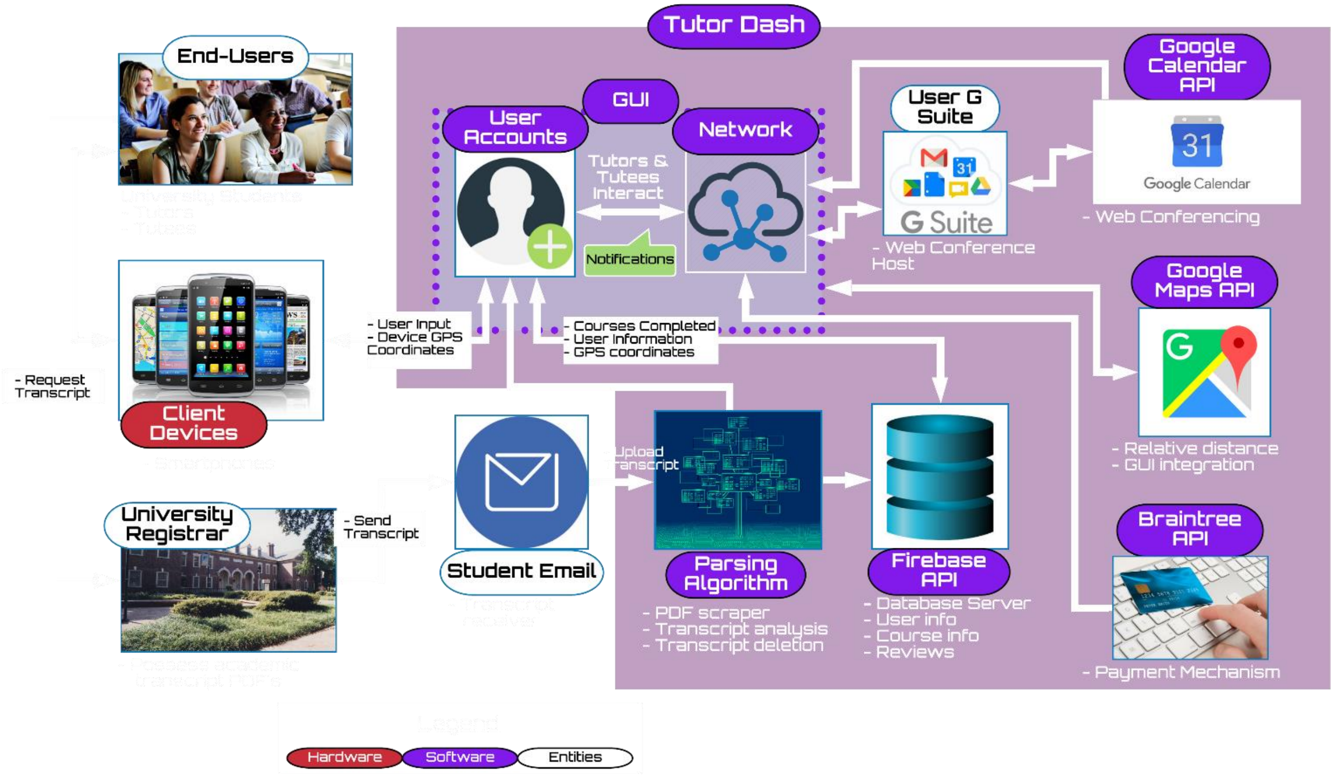

Figure 1 - Major Functional Component Diagram

2.2 Prototype Functional Description

As shown in table 1, Tutor Dash will provide various functions for the user to use. This functionality includes user quality of life functions such as a rating system, real-time scheduling, and real-time payments. The core architecture will be fully functional for the prototype. Most of the functionality that won't be operational for the prototype will be customer facing such as the support for multiple universities, night mode, the possibility of free sessions, etc. Tutor Dash will have partially functioning methods such as automated pay rate calculation, reporting features, refunds, and blacklisting of users.

2.3 External Interfaces

Tutor Dash will maintain three different types of interfaces: Hardware Interface, Software Interface, and User Interface.

2.3.1 Hardware Interface

Tutor Dash will utilize Android smartphones to run the application. The smartphone will allow the user to interact with the application and will display all user interfaces. Tutor Dash is expected to be viable on both Android and Apple devices for the real-world product.

2.3.2 Software Interface

Software being utilized will include a user interface, user accounts, and networks inside the application. Tutor Dash will also have a database run from Firebase and algorithms to gather and produce information from user input. Multiple APIs will be utilized for specific functions such as Google Calendar to set schedules, Google Maps to track and display student locations, and Braintree to process in-application payments.

2.3.3 User Interface

The user will need to download the application from the Google Play store. They will need to utilize the touch screen on the smartphone in order to interact with the application itself. The application will give alerts and notifications based on events that occur on the application.