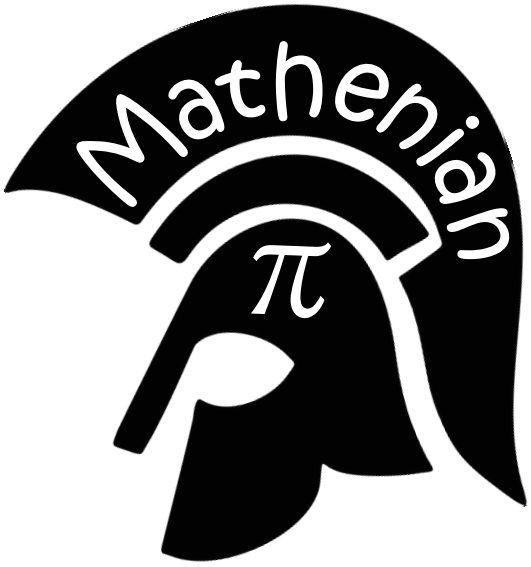
MATHENIAN

Project Proposal

YEE STUDIOS



Submitted By: Junmin Yee
Submitted To: Calvin Caldwell
Date: May 13, 2019
Email: junmin.yee@oit.edu
Version 1.0

Legal Notice

Yee Studios makes no warranties of representation of any kind concerning the accuracy or suitability of the information contained in this document for any purpose. All such information is provided “as is” and with specific disclaimer of any warranties of merchantability, fitness for purpose, title and/or non-infringement. In no event shall Yee Studios, its employees or agents be liable for any direct, indirect of consequential damages resulting from the information provided in this document. This exclusion and limitation only applies to the contrary in any written license or subscription agreement from Yee Studios in respect of this document or references.

Copyright Notice

Revision History

Author	Company	Date	Version	Comments
Junmin Yee	Yee Studios	05/12/2019	v0.1	Initial creation of document
Junmin Yee	Yee Studios	05/13/2019	v1.0	First draft for review
Junmin Yee	Yee Studios	05/21/2019	V2.0	Final draft for submission

Signatory Page

This document is accepted by:

X _____ X _____

Calvin Caldwell Date

This document is submitted by:

X _____ X _____

Junmin Yee Date

Table of Contents

1. Introduction

1.1 Purpose

The purpose of this document is to propose the design and implementation of the project, Mathenian. The sections after are dedicated to defining and describing the format of this document and the intended design plans and scope. If this proposal is accepted, it well serve as a guide for the implementation process.

1.2 Scope

The scope of this document is limited to project management, general system description, and a description of the product requirements which describes in limited detail the intended design features and functionality of the product. Also included is the specific functionality of modules and otherwise discrete functionalities that may later be added to the system.

1.3 Intended Audience

The intended audience of this document includes Calvin Caldwell, Waseem Sheikh, Junmin Yee, and any interested third party.

2. Project Management

2.1 Change Management Procedures

In the event of a change request to the current system, the CAT Team will be notified with a Change Request Form detailing the change request, the reasoning behind the request, and any additional comments. The Change Request Form can be found in Appendix B.

CAT Team – The CAT Team will consist of Junmin Yee and either Calvin Caldwell or Waseem Sheikh depending on availability. The CAT Team will evaluate the impact of proposed changes to the development of the system. Proposed changes may be accepted or rejected by the team with a brief explanation for the decision. Proposed changes will be archived by the author, Junmin Yee.

Medium – All changes must be submitted through the Change Request Form by any member of the CAT Team. Submissions may be physical or electronic. The form must be submitted to each member of the CAT Team for a request to be considered.

Protocol – The Change Request Form will be processed over a maximum of one business week. Responses will be emailed to the addresses detailed on the submitted Change Request Form.

Impact Analysis – The impact of a change to the system must be analyzed before its acceptance. If a proposed change will move the project past the scope of the Senior Project time constraints, then the change will be denied.

Archive – All change requests will be archived for future reference. The archived requests will be appended to this document in Appendix D. Any physical copies will be scanned for archiving purposes.

2.2 Software Delivery, Installation, and Acceptance Criteria

Software delivery will be distributed to all interested parties for evaluation by the end of Winter Term of the 2019-2020 academic year. Installation will be through the Google Play Store for Android users and the Apple Store for iOS users. Systems requirements will be outlined in the download page of the application. Acceptance criteria will be based on the completeness and operability of the software as determined by Calvin Caldwell and/or Waseem Sheikh.

2.3 Documentation and Online Help

Software documentation will only be provided on the installation page of the application. Provided documentation will detail the system requirements, required permissions, and operation of the application. There will be no online help for the project.

2.4 Project Risks

Project risks include, but is not limited to, learning new technologies required for the project, time management, and potential project costs.

2.5 Customer Responsibilities

The customer is responsible for providing their own phone with adequate system requirements as described.

2.6 Status Reporting

Status reporting will be submitted weekly to either Calvin Caldwell and/or Waseem Sheikh. The report will be through the Status Report Form found in Appendix C. The form will be submitted either electronically or physically. The report will include the following:

- Work completed during the current week
- Work to be completed during the following week
- Issues and potential solutions found during the current week

3. System General Description

3.1 Project Summary

Mathenian, a current placeholder name, is a mobile application designed to help users improve upon their mathematical skills through practice questions. The main audience for this app will be the general adult population from 18 and up tailored to anyone who wants to continue improving upon or maintain their skills in mathematics.

The mathematical topics covered in this app will be Arithmetic, Algebra, Geometry, Set Theory, Calculus, Probability, and Statistics. Each topic will have its own lesson except for Calculus which will be broken down into smaller subsets of the topic. The topic of Calculus will be split into Limit Theory, Differential Calculus, and Integral Calculus.

Each topic will feature an introductory lesson which will give a basic outline of the topic and the possible type of questions to expect. After this introductory lesson, the user will then be given a series of 10 problems to solve with no time limit. The lessons will neither include a calculator or a space for scratch work and as such, the difficulty of the questions will be easy enough to be answered without them. The types and values of these problems will be randomly selected from a pool of existing questions. There will be a progress bar to track the number of correct answers in that specific lesson. Each lesson is repeatable to ensure that users can continually practice.

Answering all questions correctly in a lesson will improve mastery of the lesson by 25%. Not answering all questions correctly in a certain lesson will reduce the amount of mastery gained. A user can exit out of a lesson at any time but will have no gain in mastery. Each lesson can be tested out of with a test consisting of 25 questions where a user must answer at least 23 questions correctly. Lessons will be separated into a hierarchal structure (See page 2). There will be different tiers of mastery, starting at Bronze, Silver, Gold, and Diamond. Each subsequent tier will give users more complex questions. A user is only required to master the Bronze level before being allowed to continue onto further topics. If a user finishes all topics, all lessons will still be open for practice thereafter.

Users will be able to log in which allows the app to track their current progress through the various lessons. The app will track the ‘number of questions completed daily’, ‘daily streak’, and ‘total number of questions answered correctly’ metric to show user progress. User progress will be shown in a profile page for the user. The app will also keep track of averages between all users. This will consist of ‘average number of correct answers’ in each specific lesson, and ‘percentile of users who have reached this lesson’. These averages will be shown within each specific lesson for a specific mastery. These averages will not be updated if a user quits the lesson early. The app will not have any nor require any monetary services. The mobile application will work on both Android and iOS compatible systems.

3.2 Relation of System to Existing System

N/A

3.3 Hardware Platform Description

The project requires an Android phone or an iPhone with at least 1GB of RAM, at least 16GB of hard disk space, and must be able to run the required software listed below. The phone will need a maximum of 50MB of available hard disk space for the application.

3.4 Software Platform Description

The project is intended to run on either Android 7.0, iOS 10, or higher.

3.5 Third Party Libraries

Third party libraries for this project will include database libraries and mathematics libraries.

4. Product Requirements

4.1 Functional

1. Users must be able to access the application through a smartphone.
2. Users must be able to select a lesson from a hierarchy of mathematical concepts.
    2.1. Topics covered include:
        2.1.1. Arithmetic
        2.1.2. Algebra
        2.1.3. Geometry
        2.1.4. Set Theory
        2.1.5. Limit Theory
        2.1.6. Differential Calculus
        2.1.7. Integral Calculus
        2.1.8. Probability
        2.1.9. Statistics
    2.2. A lesson has 4 levels of mastery. Mastery tiers will consist of:
        2.2.1. Bronze
        2.2.2. Silver
        2.2.3. Gold
        2.2.4. Diamond
    2.3. A lesson requires 100% Bronze mastery before the next lesson in the hierarchy is unlocked.
        2.3.1. Mastery will be shown by a colored overlay of the topic's tile.

2.3.2. A user can still open a lesson after 100% Diamond mastery.
2.3.3. The hierarchy is found in Appendix A.

3. The application will provide lessons to help users practice mathematical concepts.
   3.1. Each topic will have an introductory lesson that covers the basic concepts of that topic.
   3.2. Each topic will have 10 questions about the topic for the user to answer.
      3.2.1. Numerical values in questions will be randomly generated.
      3.2.2. Question templates will be stored within a database and randomly selected.
      3.2.3. The questions will be easy enough to answer without the use of a calculator.
   3.3. The lesson will display a progress bar to show the user their current completion of a lesson. Lessons are not timed.
      3.3.1. The progress bar will be split into 10 boxes, one for each question.
      3.3.2. A box will be marked green if the user answers the question correctly.
      3.3.3. A box will be marked red if the user answers the question incorrectly.
   3.4. A lesson will NOT provide the following:
      3.4.1. Space for scratch work.
      3.4.2. A built-in calculator.
   3.5. A user can quit a lesson at any point. The lesson will not count towards completion.
      3.5.1. If a user force closes the app, the lesson is not saved.
      3.5.2. If a user switches apps and returns with the app still running, the lesson remains.
      3.5.3. If a user wants to access a different lesson, they must quit the current lesson first.
4. Users must be able to test out of a lesson if wanted. Tests are not timed.
   4.1. The test will consist of 25 questions.
   4.2. A user must answer at least 23 of the 25 questions correctly.
   4.3. The test will display a progress bar to show the user their current completion of a lesson.
      4.3.1. The progress bar will be split into 25 boxes, one for each question.
      4.3.2. A box will be marked green if the user answers the question correctly.
      4.3.3. A box will be marked red if the user answers the question incorrectly.
5. Users must be able to have accounts.
   5.1. A user can create an account. Information required to create an account will consist of:
      5.1.1. First Name
      5.1.2. Last Name
      5.1.3. Username
      5.1.4. Email Address
      5.1.5. Password
   5.2. Information for accounts will be encrypted and stored in a database.
   5.3. A user can login after an account has been created.
      5.3.1. A user can enter in their username on the login screen.
      5.3.2. A user can enter in their password on the login screen.
      5.3.3. The application will verify credentials by comparing values after encryption.
   5.4. A user can recover their account if lost.
      5.4.1. The application will send an email to the user's email address.
      5.4.2. The user can reset their password from the email.
   5.5. A user can logout of the application.

6. The application will track user specific statistics and overall statistics.
    6.1. User specific statistics will consist of:
        6.1.1. ‘Number of questions completed daily’ line graph.
        6.1.2. ‘Daily streak’ counter.
        6.1.3. ‘Total number of questions answered correctly’ counter.
    6.2. User specific statistics will be displayed on the user’s profile page.
        6.2.1. The profile page is accessed from the lesson hierarchy page in a sidebar.
        6.2.2. The profile page includes all information about a user.
    6.3. Overall statistics will consist of:
        6.3.1. ‘Average number of correct answers in a lesson’ counter.
        6.3.2. ‘Percentile of users who have reached this lesson’ counter.
    6.4. Overall statistics will be displayed within each specific lesson.
        6.4.1. Each lesson will have a small tile in the upper right corner that opens an overlay for the statistics for a specific lesson of a specific mastery.
    6.5. Statistics will be updated after each lesson.
    6.6. Statistics will not update if a user quits a lesson.
7. Users must be able to change the theme of the application.
    7.1. The themes provided for the user will consist of:
        7.1.1. Light Theme – For day usage.
        7.1.2. Dark Theme – For night usage.

4.2 Performance

The application will operate with minimal delay. The only loading time will be from database access to generate new questions for lessons and for user information.

4.3 Data Transfer Description

The application will need to transfer user information between a user’s device and the database. The amount of information transferred will not exceed 10kbps.

4.4 Security/Safety/Constraints

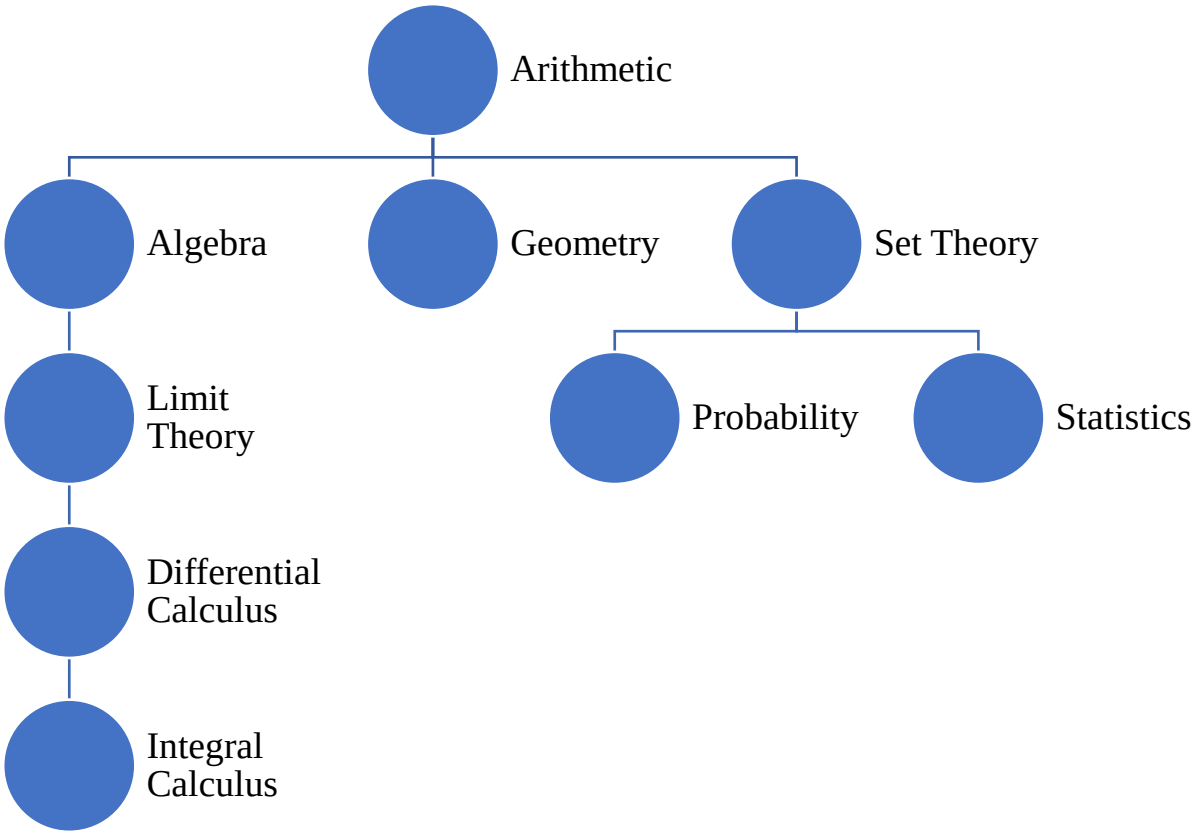
The application will encrypt user information for data transfer between device and database. Yee Studios expects the user to be within a safe environment while operating the application. Yee Studios also expects the user to conform to all civil and federal laws while using Mathenian.

5. User Profiles

The users of this software application will be people interested in maintaining or improving upon their mathematical skillset.

Glossary of Terms

Appendix A – Hierarchy of Mathematical Concepts



Appendix B – Change Request Form

Name	Email Address	Date

Change Request:

Purpose for Change Request:

Additional Comments:

Appendix C – Status Report Form

Name	Date

Work completed during the current week:

Work to be completed during the following week:

Issues and potential solutions found during the current week:

Appendix D – Change Request Archive