**tABLE OF CONTENTS**

|  |  |
| --- | --- |
| **Part A: Context** | 2 |
| Background Information | 3 |
|  |  |
| **Part B: Required Functionality** | 4 |
| Functional Requirements | 5 |
| Non-functional Requirements | 9 |
|  |  |
| **Part C: Desired Functionality** | 11 |
| Knowledge Web | 12 |
| The Wizards Crystal Ball | 13 |
| Defense of Calculot | 15 |
|  |  |
| **Part D: Specifications** | 18 |
| List of Topics | 19 |
| Collection of Sample Questions and Answers | 20 |
|  |  |
| **Appendix A** | 21 |
| Applications that have similar components to the project | 22 |
| How does “Defense of Calculot” do things differently? | 23 |

A medieval-themed collection of interactive tools to help users learn lower-level university mathematics

Calculot

Jean Acebedo, jacebedo

Kenneth Zheng, kza21

Levent Kylych, lkylych

Ryan Peterson, repeters

Created for:

Introduction to Software Engineering – CMPT 276 D100, fall 2017

Instructors & Staff:

Fahira Naz

Xia Hu

developed by:

PART A: CONTEXT

**Background Information**

**Domain:**

Entertainment: Educational Mathematics Game

**Topics:**

Linear Algebra – Vectors, Matrices

Pre-Calculus – Trigonometry, Complex Numbers, Polar coordinates

Calculus – Theorems, Limits, Derivatives, Integrals

Theme – Medieval Fantasy

**Problem:**

Some students have problems grasping an understanding around post-secondary mathematical concepts. This is because of a lack of fundamental structure around pre-calculus and calculus.

**Sub-problems:**

While some applications provide insight on this problem, these applications provide a stale implementation around the topic. This can be detrimental to the learning process. Boredom can lead to procrastination which prevents effective learning.

**Purpose & Audience:**

High school and University students taking Pre-calculus, calculus, or linear algebra can use this app as a helpful tool to test and better understand the fundamental concepts in mathematics while providing an interesting thematic approach to the topic to keep things fresh.

**Storyline:**

User Requirement Definition:

The <app> should provide tools for learning concepts in the three subjects linear algebra, pre-calculus and calculus as well as test them in an enjoyable way. App should distinguish different students profile, allowing multiple users to save learning progress.

**Required Functionality – Functional Requirements**

PART B: REQUIRED FUNCTIONALITY

**Login System**

* **Upon runtime, this system shall display all the registered users in the database.**
  + All usernames must be clickable and will lead the user into the password entry activity, which they will enter the password to continue.

**Password Entry System:**

* T**he system shall log the user into the game upon entering valid user information**
  + The user must be registered into the system.
  + When the user enters information that is not in the database, they will receive an error dialog that prompts them to “register” or to “try again” which brings them to either the registration activity or cancel the dialog message.

**Registration System**

* **The system shall place the user information into the user database, if the information is ‘valid’.**
  + Validity is defined as such:
    - The user has not already been registered into the system.
    - The first name entered only contains alphabetical characters (A-Z, a-z) in their first name. All other characters invalidate the username.
    - The username only contains alphanumeric characters and underscores.
    - The password must include: At least 1 Uppercase letter, 1 Lowercase letter and 1 number. All other characters, or the exclusion of any of the characters listed will invalidate the password.
  + If all the inputs are “valid”, the system will notify the user via a “Toast” message that will tell them that the system has correctly placed their information in the database.
  + If one of the inputs are “invalid,” The system will display a dialog message notifying the user which of the fields have the invalid input, and will prompt the user to either go back to the main menu, or try again (dismiss the prompt).
  + If the username, and first name already exist in the system, but under a different password, a dialog will appear that will ask the user if they want to switch passwords. This will be a yes-or-no dialog and choosing yes will activate a prompt to enter the “old” password and “a new password”
    - If the old password is in the database under the username, the stored password will be changed to the new password, and a dialog will notify the user that their password has been changed.
    - If the old password is incorrect, or the password is invalid, the system will display a different dialog that will notify them that the password change has been unsuccessful. The dialog will prompt them to either try again or go back to the registration menu.

**Learning System**

* **This system shall provide a list of clickable topics which brings up information about the topic as (videos, images, etc.)**
* **The system shall provide a set of examples, in which a few are filled out and the few are blank.**
  + Upon the user filling out all the blanks and switching screens or clicking the ‘solve’ button, the system shall check if the examples entered by the user are correct.
  + If the system deems the solutions to be correct, the topic will indicate change (in the form of either text color, borders, background or icon)
  + If the system deems the solution to be incorrect, no changes will be made.
* **Clicking the android back button from a specific topic page shall bring the user back to the list of topics, which they can choose another topic or return to the menu screen.**

**Practice System – Multiple Choice Infrastructure**

* **Upon clicking the ‘practice’ button,** **the system shall load up a set of questions related to the topic.**
* **After clicking one of these choices, the system shall launch the game.**
* **The game shall display a question, the lives of the user, 4 possible answers, and the experience gained from the current playthrough.**
  + When the user clicks on the correct answer, the user shall gain experience, and a new question will appear.
  + When the user clicks on a wrong answer, the user shall lose a life, and a new question will appear.
  + If the user takes too long to respond, the user shall lose a life, and a new question will appear.
  + Example Question: Differentiate / Answers: , , ,
* **When the user reaches 10 answers, the system shall end, rewarding the user with experience to be stored in their profile.**
* **When the user loses all their lives, the system shall display a dialog which displays how many questions they have answered correctly and how many experience they have achieved from the playthrough of the game.**
  + The user shall only receive a quarter of the experience, due to the loss
  + A dialog shall appear and prompt the user to either: go for another round or go back to the practice menu.
* When the user clicks on the android back button, **a dialog shall appear that asks the user if they want to quit. The system shall prompt the user to either continue the game or save the progress and return to the practice menu.**

**Profile System**

* **The system shall display the player character of the user, the username, and first name entered during the registration page.**
* **The system shall display 3 different fields for experience: Total XP, differentiation XP and Integration XP, Algebra XP, and Trigonometry XP.**
  + Total XP is defined as the sum of all the experience of the topics.
  + There will be a bar that shows the graphical interpretation of the XP bar.

**Achievement System**

* **The system shall display a list of achievements in the profile page, and will display all the ‘locked’ achievements as the grayed-out version of the achievement.** 
  + When an achievement is ‘locked,’ it means that it has not been achieved. After the achievement is unlocked, the achievement will be unlocked.
  + Any achievement that has been unlocked will display a message that notifies the user that they have unlocked an achievement.

**Non-Functional Requirements**

* **Dependability Requirements**
  + The application shall handle errors gratefully. No user input shall cause the system to crash.
* **Efficiency Requirements**
  + Performance: The system shall respond smoothly to user input.
    - It shall not respond longer that 1-2 seconds for the system to respond after receiving the users input
  + Space: The system shall consume a small amount of memory space for installation.
    - The application shall not consume more than 200 MB of memory on the user device.
* **Usability Requirements**
  + Users shall have the ability to use the application whenever they desire.
  + The interface shall provide an easy-to-use interface and high-level instructions.
  + All components shall be simple for understandability reasons.
* **Security Requirements**
  + The system shall first ask the user to register or sign in an already existing user account.
  + The user must be unique in all cases and the user progress must be unique only to that user.
  + The user profile must only be accessible through the credentials that the user has entered.
* **Organizational Requirements**
  + Operational: The system shall:
    - Store user accounts in the database.
    - Allow user to sign into their account in the database
    - Provide the user questions on the selected topic.
    - Collect user answers to the question and verify if it is correct and inform the user if the answer is correct or incorrect.
  + Development Requirements
    - The system shall be developed using the Java Language
    - The system shall be developed using Android Studio exclusively
  + Environmental Requirements
    - The application shall work on all android operating systems running on API 16(Android 4.1 Jellybean)
* **External Requirements**
  + Ethical Requirements: The application shall provide correct information about math topics to not confuse users.
  + The application shall not display, or distribute any user information to external sources.
  + The application shall give credit to all sources used through a reference page.

PART C: DESIRED FUNCTIONALITY

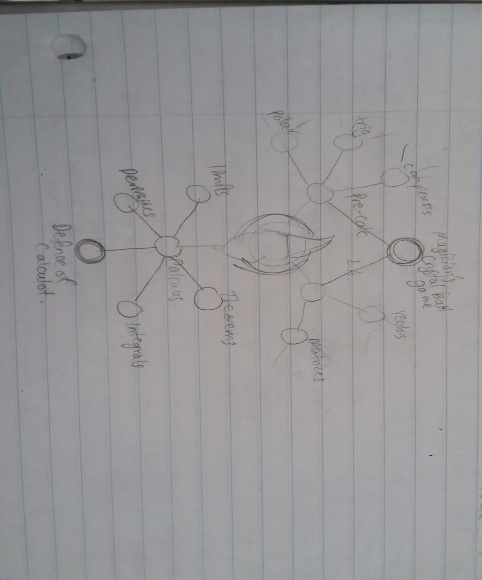
**Desired Functionality**

**Learning Center:**

**The Knowledge Web**

An alternative view of the learning topics will be presented as a web. The center will have a small circular icon showing the user as a wizard and the total XP earned so far. We can consider this the “root” of the web. The web consists of nodes connected by links where each connected node share some commonality of topic.

* The initial three nodes will be Linear Algebra, Calculus, and Pre-Calculus and further branch off into its respective sub-topics. For linear algebra this will be vectors and matrices which may further branch off into “Lesson X”, “Lesson Y”, and so on.
* Each node may have a bar showing the level of completion of that particular topic. The two games “Defence of Calculot” and “The Wizard’s Crystal Ball” may be considered as separate nodes which may appear to connect to multiple other node topics which contain the tested material.
* When a node is clicked, more information is brought up as a dialog, including the title of the topic, a summary of the material covered and two options in the form of buttons to start the particular activity or cancel the dialog.
* Some nodes may appear locked in which case may be greyed out. Prerequisite nodes, which should be closer to the root node, will have to be completed before the particular node is unlocked. “Closer” is defined as requiring the traversal of fewer links in order to reach the root node.



**Games**

**The Wizard’s Crystal Ball**

**Game Purpose:**

The purpose of this game is to test for the understanding of trigonometric functions, complex numbers, vectors, and polar coordinates.

**Levels:**

Levels differentiate between different difficulties and as the levels progress new topics are introduced into the set of potential questions. Levels are broken up into smaller stages.

Level one may test for the components of a vector given the x and y coordinates and level two may introduce the use of “theta” as the reference angle of the vector in the set of potential questions, potentially combining the two concepts into more complex questions as the stages within the level progress.

**Presentation and Crystal Ball Stages:**

Present a plane with two axes. The real/x axis and the Imaginary/y axis. The player begins with a “crystal ball” drawn around the origin of the plane. The crystal ball is initiated at “stage 1” and has a small radius around the origin of the plane. Each stage represents a new layer around the previous ball. A completed level requires the completion of several stages.

The game objective is to gain a “full” crystal ball by answering multiple choice questions correctly presented as a choice of buttons.

As the stages progress and the player is successful, the ball grows in concentric rings. Each stage must be fully drawn with “360” points representing a “full” stage before the next stage may be drawn. A small crystal ball represents minimal knowledge/mastery of the level topic. A large and completed crystal ball means the player has successfully gained mastery of the level topic and is ready to progress to the next level. “Trig XP” experience may be granted after completion of every “question” vector depending on player performance.

**Question Vector:**

Upon level start, a question vector will be presented and limited data about the vector will be presented. This data may include the x or y coordinates, the reference angle, the norm of the vector or any other suitable information. The user must answer a multiple choice question, presented as a selection of buttons, which may include, but is not limited to the components of the vector or the complex number represented by the given vector or the polar coordinates of the given vector.

**Gaining and Losing Points via Clock Vector:**

A “clock vector” is instantiated at the same time and same angle as the “question” vector and represents the “potential gains” or “potential loss” of the current question. A translucent circular layer representing the next stage is drawn over the previous layer such that the previous layer remains visible. The clock vector rotates clockwise, slowly erasing the “potential gains” translucent layer. A counter between 0 and 360, representing the angle from the clock vector to the question vector in the “clockwise direction”, decrements along with the animation of the clock vector. If the player answers the question correctly while the counter is between 0 and 360 then the remaining points are added. Visually, this may appear as the potential gains layer solidifying into the crystal ball as a solid colour.

Stages may be partially filled in which case the potential gains fill the previous stage to 360 first before contributing to the points of the next stage.

If the clock vector undergoes one full revolution, then the translucent potential gains layer will have fully disappeared and a new potential loss layer will appear. This layer may be visually contained within the stages considered to be a part of the crystal ball. The counter now runs from -1 to -360. If the player answers the question correctly while the counter is between the aforementioned range, then the loss will be calculated to the crystal ball. Once the counter reaches beyond -360 then the loss is calculated as the full -360 to the crystal ball. That is, a full stage is lost. A new question vector is then presented.

If the player answers the question incorrectly at any point in time, then the answers are locked for a brief moment, perhaps one to two seconds and the counter decreases by n\*180 where n is the number of failed attempts for the particular given vector.

**Beating/Losing the Game:**

If the crystal ball no longer contains any stages, that is the original stage one contains no more points then the player has failed the level. All potential “Trig XP” gains are halved. If the crystal ball becomes full, that is every stage has reached 360 points with the number of stages varying depending on level, then the player has completed the level. All potential “Trig XP” gains are fully applied to the user profile.

**Defence of Calculot**

**Upon clicking the wizard game button, the system shall display three choices for the user: Differentiation, integration, or both.** This choice will decide the questions that are asked to the user during the game.

**After selecting one of these choices, the system shall launch the wizard game**

**The wizard game shall display**

* Tower
* Monster (moving towards the tower)
* Lives/health remaining
* The current question
* Four possible answers to the question

**Monsters**

* **Idea 1:** Each monster approaching the tower has a different question associated with it. The colour of the monster is defined by the difficulty of the question it carries. More difficult monsters move slower. The user will select a monster to attack, which will prompt the current question. When the question is in play and the monster is selected, the monster will be highlighted so the user knows which monster is currently being attacked.
  + After answering *n* questions, the monsters move faster towards the tower
* **Idea 2:** One monster approaches the tower at a time. The monster represents a question that is asked to the user. As there is only one monster on the screen at a time, the user does not need to select the monster, instead the question will appear automatically when the monster spawns. After *n* monsters, a more difficult monster spawns with a more difficult question. More difficult monsters move slower.
* If a monster reaches the tower before the user has correctly answered the question, the user loses a life/health and is a step closer to losing the game.

**Questions**

* The questions are multiple choice and should not be too difficult. Upon the question being asked the question will appear on screen as well as four boxes at the bottom of the screen, one for each answer to the question.

**Upon selecting an answer to the current question**

* If the answer is correct
  + The monster is defeated. The closest monster will disappear from the screen, and another monster will take its place entering into view from the starting area.
* If the answer is incorrect
  + The monster is not defeated. Instead you must wait a short time interval before trying to answer that question again.
  + **OR**
  + The monster is not defeated. The question is gone and that monster carries a new question

**Beating the game**

* The user will win the game after a certain number of questions are answered (all the monsters are defeated).

**Losing the game**

* If the user loses all their lives/health, the application will display a dialog which shows how many answers they answered correctly and how many they got incorrect.

**Upon game completion**

* The user will gain XP dependant on how many questions were answered correctly. This applies whether or not the user wins or loses the game.
* If the user wins the game by answering all the questions, they will gain additional XP as a bonus for winning.
* The application will ask the user if they would like to play again or go back to the previous menu.

**Pause button**

* At any point the user may pause the game by pressing the pause button in the corner of the screen.
* This will pause all game activity including timers and moving monsters.
* When the game is resumed the current question will disappear and either a new one will appear or the user will get to select a monster again.
  + If the user selects the same monster, the question will be different from what it was before.
* This will ensure the user can not cheat by pausing the game to allow more time to answer the question.
* A dialog will appear on screen saying the game has been paused. There will be a button in this dialog saying “Quit”.
  + Pressing the quit button will prompt the user if they would like to abandon their progress and quit the game.

**Back button**

* Pressing the Android back button will pause the game in the same was as described above.

**Multitask and Home button**

* Pressing the Android home button or Android multitask button will cause the game to pause automatically while it is suspended in the background

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Topic Sections:**

* Each topic selected (e.g. differentiation, integration, etc.) will have another menu for how the user would like to interact with this topic. They shall be as follows
  + Learning
    - This will include reading as well as external links and videos to teach the user about the topic, along with some fill in the blank examples for theorems and questions.
  + Practice
    - Games that will allow the user to practice the skill with fast, easy problems. This will allow the user to build their speed and memory for these topics

**Leveling up**:

* As the player gains XP and levels up, their rank and title will change as well as their picture and sprite. This adds fun to the game and encourages the user to get to the next level.

**Design and GUI**:

* The application shall have a consistent theme and design. The theme of the application may match the theme of the implemented games.

**Games:**

* The core elements of the games are to be implemented as required functionality and finished in Sprint 2.
* Adding game functionality including graphics will be a desired feature and will be added for Sprint 3.

PART D: TOPIC SPECIFICATION

**Topics Covered**

Taken from:

Calculus: Early Transcendentals Textbook, 7E. by James Stewart

Contemporary Linear Algebra 9E by Anton Busby

|  |
| --- |
| **Part A: Trigonometry (Appendix D)** |
| Basic Trigonometric Identities (sine, cosine, tangent, cosecant, cotangent, etc.) |
| Trigonometric Identities (Double Angle, Addition formulas, product formulas) |
| Trigonometric Identities of the Unit Circle |
|  |
| **Part B: Differential Calculus (Chapter 2 and 3)** |
| Limits & the Definition of a Derivative |
| Differentiation Rules of Basic Functions |
| Chain Rule |
|  |
| **Part C: Integral Calculus (Chapters 4 and 5)** |
| Definite Integrals |
| Indefinite Integrals |
| Substitution Rule |
| Integration by Parts |
|  |
| **Part D: Linear Algebra (Chapter 1 of Busby Textbook)** |
| Vectors |
| Vector Operations |
| Norm and distance for a vector |
| Dot product |
| Orthogonality & Orthonormality |
| Matrix Operations |
| Matrix Row/Column Elementary Operations |
| Inner Product of a Matrix |
|  |
| **Part E: Subsections (Various)** |
| Polar Coordinates |
| Complex Numbers |

**List of example questions**

|  |  |  |
| --- | --- | --- |
| Question | Possible Answers (for Multiple Choice) | Correct Answer |
| Which ratio gives the following trigonometric identity: |  |  |
| True or False: | True, false | True |
| Answer the following: | Various |  |
| True or False:  The tangent at point a represents the derivative as the limit of x approaches a. | True, False | True |
| Differentiate: |  |  |
| Integrate (assume there will be a constant): |  |  |
| Evaluate: |  | 11 |
| Compute: |  | 1 |

\*Note: These questions are only for discussion and verification purposes. The final product may or may not have these questions included.

APPENDIX A

Applications that contain similar systems to “Defense of Calculot”

**HKU Calculus –** HKU Calculus provides a similar idea to help students learn, and track their progress. They include text and video resources to aid the student in learning how to approach the topics being discussed.

Along with this, HKU Calculus also provides a Multiple Choice game to allow users to practice the theorems and methods they have learned from the application.

Google Play Store Link:

<https://play.google.com/store/apps/details?id=com.hkuscifac.hkucalculus&hl=en>

**Test Your LIMITS –** Test your limits include a Multiple Choice game around a theme of racing. This application focuses around the same topics as “Defense of Calculot” to help users practice calculus.

Google Play Store Link: <https://play.google.com/store/apps/details?id=com.jcweaver.calcLimits&hl=en>

How does “Defense of Calculot” differ from these applications?

**HKU calculus**

* While there are text and video resources for *HKU Calculus*, “Defense of Calculot” shall apply a more hands-on approach to the learning resources. There will be examples that will have a blank answer in which the user will input what they believe is the right answer.
  + Upon clicking the “finish” button, the android back button, or the implemented back button, the system shall check if all examples are correct.
  + If the examples are correct, the system will place a checkmark in in the topic outline, to signify that the user has “completed” the learning experience.

**HKU Calculus & Test your LIMITS multiple-choice Game**

* *Defense of Calculot* will be a more thematic approach to the multiple-choice game genre. While the core of the *Defense of Calculot* is a multiple-choice game, the implementation will provide a graphical and more interesting approach to the multiple-choice game genre.
  + Example: In “Adventure Mode” There will be monsters that will approach the user’s character. The user must click these monsters and answer the multiple-choice question. This interactivity provides an interesting approach to the genre.