OWL: Offline While Learning

Information Analysis & Design
CS-670-A
Team #1
Created for:
Professor Moshe Cohen

Sacred Heart University
5151 Park Avenue
Fairfield, CT 06825

Spring 2020t
Created 1/14/2020

Names	**E-mail**	**Contact Number**
Scott Yanoff	scott.a.yanoff@gmail.com	203-451-8644
Julia Katz	jj7jj7@gmail.com	203-273-8807
Michael Lawrence	michaellawrence193@gmail.com	804-517-0998
Vathsav Tipirneni	vathsavtipirneni97@gmail.com	475-231-9054
Meghan Polis	meghan.polis@gmail.com	631-278-3117

Applications used:
Microsoft Word 2019
Microsoft Excel 2019

Table of Contents

6.0 Design Phase

7.0 Construction Phase

8.0 Implementation Phase

1.1 Abstract for the Offline While Learning

This use case describes the event of a client (Parent, Educator, or other people with educational interests) getting their children to learn mathematics through the use of our computer based solution. Our solution combines a dynamic math problem database with an easy to use application to give children a great educational experience without the need for a classroom. Once the child has the application installed on their computer they can customize their experience by creating unique user profiles, so that they can tailor the difficulty and type of content to their needs. If a child is unsure of what difficulty level they should be at our application will have a built in optional pre test that will help the user identify their strengths and weaknesses and suggest a starting point. Once the user has either taken the pre test or self-identified their ideal difficulty level it is time to present the children/students/users with math problems taken from our database which is filled with a large variety of problems using all mathematical operators. Once the child answers the questions they will receive a progress report at the end of their session. Users/Children will be able to use the progress report to see if they are improving or not based on how may they get correct and incorrect. On completion of our objective the student will improve in mathematics.

2. Preliminary Investigation Phase

2.1 Summary of Problems, Opportunities and/or Directives.

One of the global challenges facing children in the US today is the decline in educational standards over the years. This decline has been most seen in the STEM disciplines (Science, Technology, Engineering, and Math) which have fallen from amongst the top in the world to the middle of the pack. The areas in the US that have been hit the hardest in recent years have been more rural areas that do not have access to the same educational resources that children in more urban areas have. This lack of educational resources has led to a large disparity in the quality of education received.

Our solution to close this gap involves an offline application that can be used to supplement their studies in school. Since this is a new venture for us we will be limiting the subject matter of our application to just mathematics.

Our primary reason for developing this application is that currently there are no applications that do what we are trying to do and operate offline. By allowing our application to be used offline it allows access for all customers regardless of the quality of their internet. Having quality internet is a large problem, especially in rural Americans here the telecommunications infrastructure is not as developed as other urban areas. It is also in these rural areas where the quality of education is the poorest, so it is important that we develop a solution that is targeted at this population who are generally ignored.

The Offline While Learning (OWL) application is a solution that will compliment the teachings in the classroom. We want to make sure that the customer knows that this is not meant to replace traditional classroom learning, and is meant to be used in tangent. The subject matter that will be included in this application will be basic mathematics (additional, subtraction, multiplication, and division) which are common in most elementary school math curriculums. We will also include a rotating set of problems from each of the common math operands and direct feedback system that will show the user how well they are doing. It was important to us that we give the user a way to track their progress while using the application as seeing improvement can be a strong motivation to keep using the application and help the user become more engaged with their learning in school as well. Another feature that we wanted to include for our customers is rotating problem sets. The one caveat of having this feature is that it will require the user to access the internet to download the new problem sets. We do believe that having this feature will keep students engaged by having them solve new problems instead of becoming complacent solving the same problems over and over. These problem set updates will not be updated on a bi-annual basis, so we will make sure to include a large amount of problems in each set.

2.2. Statement of Preliminary Scope.

2.2.1. Description of Data used by System Study

Our System Mimics the process of Manual Flashcards with an emphasis on tracking.

- *Database of Math Problems* filled with problem sets by 4 main operands: (+, -, *, /) which our application will calculate the correct answer and compare the user input from the problem set store(database or storage area) a success or failure into the user profile success tracker.
- *Challenging or Advance and Current Operand Setting Data* will help to increase mastery of the subject by making the problems more difficult. Complexity will be available in three categories - single, double and triple digit problems. This will tailor what problem sets show up for the user to solve.
- *User Profile Data* - describes the Profile based on successful problem set completion. The user profile data will be used to track and communicate graphically the rate of success through the application.
    - A table or graph showing success by operand and level.(scores with numbers instead)
    - Trend graph problem sets completed correctly.

2.2.2. Business Application Processes

Application Process:

1. Randomly Generate or load Simple Operand Data
2. Set Display Problem
3. Request calculation
4. Display Evaluation of Answer
5. Store success or failure
6. Display successes or failures graphically

2.2.3. System interface with Users, Locations, and Other Systems

Interface with Users:

Interfaces with users through individual logins and customized user experiences.

Interface with Locations:

Non applicable due to location not playing a factor in our application.

Interface with Other Systems:

It does not interface with other systems. It is an offline app and therefore locked. The scores generated after each problem set can be incorporated into other systems, but not through our application.

2.3. Assess Project Worth in Terms of Cost vs. Value

Following chart lists development cost and thereafter annual operating and maintenance costs which are projected to increase at a rate of two percent each year for the next five years. Annual revenue is projected to increase at a rate of eight percent for five years. Cost and revenue projections are made for five years to allow a more accurate analysis of financial benefits that can be derived from the system. If these figures pan out, a positive return is projected to be generated during the second year of operation.

Cost Value Analysis		
Expected Revenue For Years 1 to 5		**$181,644**
Total Personnel Cost	**$66,000**	
Total Hardware Cost	**$3,500**	
Total Development Cost		**$69,500**
Expected Operating Cost for Years 1 to 5		**$26,000**
Total Profit		**$86,144**
***In depth analysis in 5.2.2** ***All formulas derived from 5.2.2** ***Preliminary estimates do not consider time value of money**		

2.4. Preliminary Project Plan

2.4.1. Master Schedule for Entire Project

Start: 10/1/2019 Finish: 11/26/19

ID	Task Name	Duration	Start	Finish	Resource Names	% Complete
1	**Introduction**	**1 Day**				
2	Cover or Title page	1 day	10/01/19	10/2/19	GROUP	2%
3	**Preliminary Investigation Phase**	**6 Days**				
4	Summary of problems, opportunities, and directives	2 days	10/17/19	10/19/19	SCOTT	3%
5	Statement of preliminary scope	2 days	10/18/19	10/20/19	JULIA	5%

6	Assess project worth in terms of Cost vs. Value	1 day	1022/19	10/21/19	MIKE	6%
7	Preliminary Project Plan	1 day	10/22/19	10/21/19	VATHSAV	7%
8	**Problem Analysis Phase**	**14 Days**				
9	Study The problem domain	6 days	10/22/19	10/28/19	JULIA / MIKE / VATHSAV	10%
10	Analyze problems and opportunities	3 days	10/24/19	10/27/19	SCOTT	11%
11	Establish System Improvement Objectives	3 days	10/24/19	10/27/19	SCOTT	14%
12	Re-evaluate and update project scope	3 days	10/24/19	10/27/19	SCOTT	17%
13	**Requirements Analysis Phase**	**8 Days**				
14	Identify Requirements	4 days	10/29/19	11/2/19	VATHSAV / MIKE	23%
15	Analyze functional requirements using system modeling approach	2 days	11/3/19	11/5/19	SCOTT	30%
16	Master list of all requirements	1 days	11/4/19	11/5/19	JULIA	35%
17	Re-evaluate and update project scope	1 days	11/4/19	11/5/19	JULIA	37%
18	**Decision Analysis Phase**	**12 Days**				
19	Identify candidate solutions	1 day	11/9/19	11/10/19	SCOTT	44%
20	Analyze candidate solutions	4 days	11/7/19	11/11/19	MIKE	51%
21	Compare candidate solutions	2 days	11/9/19	11/11/19	VATHSAV	57%
22	Recommend a final "best" solution	5 days	11/6/19	11/11/19	JULIA	58%
23	**Design Phase**	**22 Days**				
24	Design the application architecture	5 days	11/13/19	11/18/19	JULIA	66%

25	Construct detailed models	5 days	11/12/19	11/17/19	SCOTT	73%
26	Design the system database	3 days	11/14/19	11/17/19	VATHSAV	78%
27	Design the system interface for each model	5 days	11/13/19	11/18/19	MIKE	86%
28	Design the system database	2 days	11/19/19	11/21/19	VATHSAV	93%
29	Design the system interface for each model	2 days	11/19/19	11/21/19	MIKE	100%

2.4.2. Resource Assignment

Personnel

- 1 System Analyst
- 1 Visual Studio Developer
- 1 Network Specialist

Software

- Visual Studio
- SQL Server

Hardware

- Laptops
- SQL Database Server

3. Problem Analysis Phase

3.1. Study the Problem Domain

The problems in the teaching domain are within managing the actual quality of teaching, unbiased tracking, and scalability. There are a number of different ways that the current system fails;

1. The shortcoming of the students learning in a classroom environment is that the amount a student learns varies depending on the quality of the teacher.
2. The second major failure would be in eliminating completely the possibility of a biased/flawed tracking system.
3. Current testing in schools can be biased due to the difference in teaching styles and learning styles. Doesn't tailor education based on the diversity of the children and the teaching of the mentor.

   In recent years, according to Sztajn, many children do not have the opportunity to learn the same quality mathematics as their peers. In the past, teachers had to tailor their teaching styles when children were coming from "different socioeconomic backgrounds". In today's world, students "from upper socioeconomic backgrounds (are more likely to have) experience with solving math problems" than those from lower socioeconomic backgrounds (Sztajn, P. 2003).
4. The third largest failure is the system's inability to scale the solution due to the labor and time it costs to create, manage and support the system.

3.1.1 Data Collected by Current System

In the current system, there are educators who teach students in a classroom setting and give students homework to complete at home. There are 4 types of equation operators that math teachers (educators) mainly teach at school. At the end of each problem assigned as homework there will be a track record, that is manually used to keep a log of the correct answer for each problem which is held by the teacher.

3.1.1.1. Methods

- Using math sheets.
- Answer the math single digit question and put down the answer.
- Using an excel spreadsheet to track the correct answer of each problem.

3.1.1.2. Storage

- The library catalog of 25 homework sheets per math skill which in total is 100 problems on the math sheets.

- Track record would be manually kept in a chart or excel format.

3.1.1.3. Personnel Involved

- Parents would be looking at the track record
- Child would be doing the math sheets
- Teacher would be looking at the track record

3.1.1.4. Time Involved

When completing homework, it may take a substantial time. For example, each student would be given their own 100 math sheets in order to replicate the options available in the proposed solution. The total time would be 15 minutes to work on each math sheet, and 5 minutes per day to manually write down the tracking.

3.1.1.5. Sample Data Model

- Library of catalogs of homework sheets
- Written track record of success and failures kept by teacher

3.1.2. For Each Report Reproduced by the System

3.1.2.1. List Name and Format

- Catalog: Tells you which problems were correctly answered.
- Tracking: In a chart format, which can manually expand into graphs.

3.1.2.2. List Inputs and Outputs

- Catalog:
  - Inputs for the catalog: are the problem and the correct answer.
  - Output: if the answer is correct or not.
- Tracking report:
  - Input: number of correct answers.
  - Output: a trend chart showing either improvement or decline.

3.1.2.3. List Responsible Personnel

- Learner
- Mentor
- Parent(legal guardian)

3.1.3 Processes Currently Implemented

3.1.3.1 Processes

The following are the processes that are currently in place.

- *SCHOOL*: Student walks to the professor about their doubts.
- *HOME WORK*: Student takes care of his work and make sure it’s done if she/he is good with math.
- *TUTORIALS*: Professor gets to know the student standards by the way she/he performed their test.
- *EXTRA HELP*: If the student has less standards then he gets some extra help from his professor.

3.1.3.2 Hardware and Software Used

Hardware

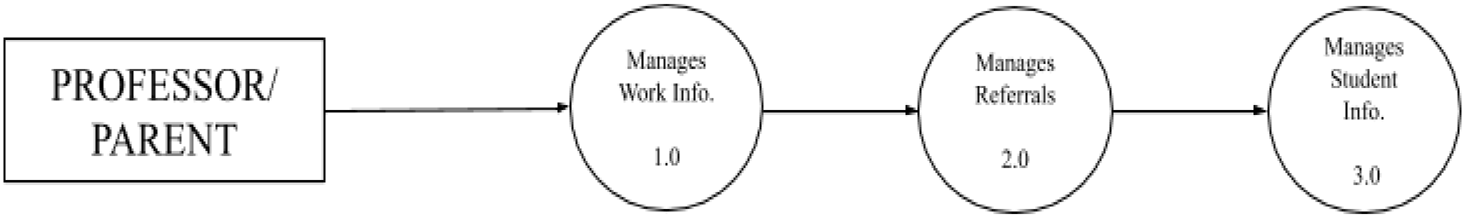
- Laptops/Desktops

Software – Custom off the shelf software

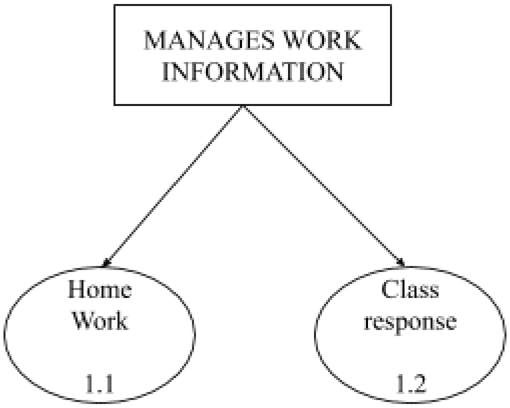
- Microsoft access
- VB.net
-

3.1.3.3 Functional Decomposition Diagram for Current System

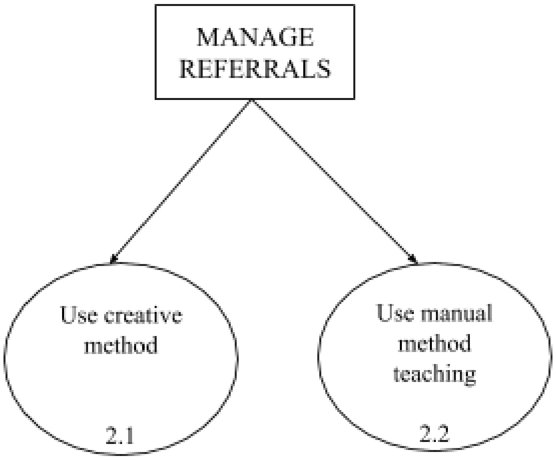
Level-0 Diagram for Current System



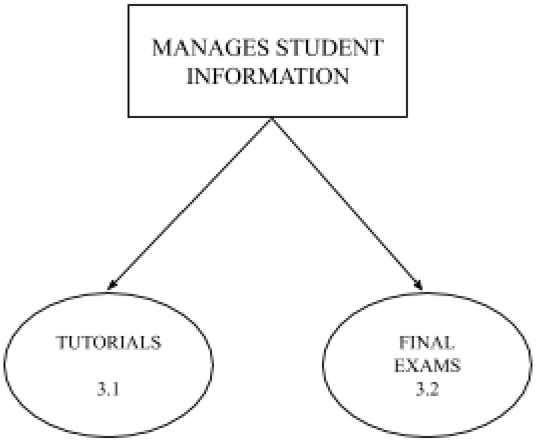
Level-1 Diagram for Current System



Level-2 Diagram for Current System



Level-3 Diagram for Current System



3.1.4. System Interfaces

3.1.4.1 Location Served by the System

The system is available for purchase at any electronic device distributor (Best Buy, Target, Walmart).

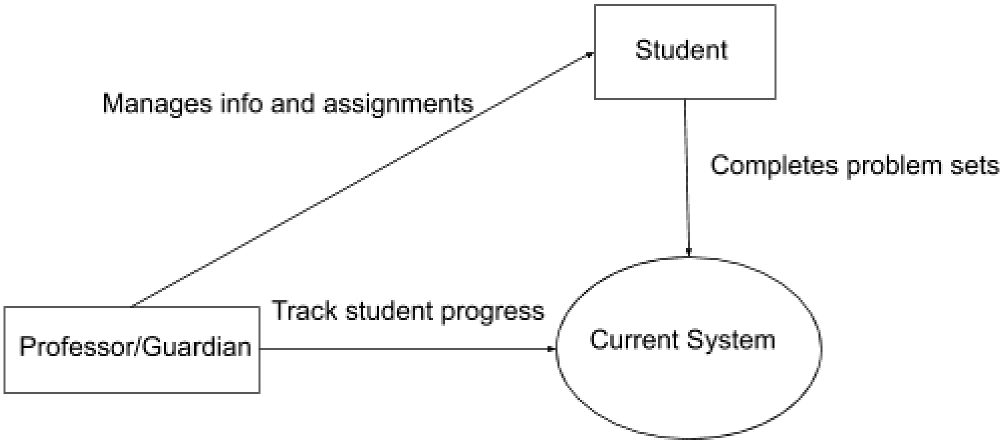
3.1.4.2 Users Served by the System

The system serves youths seeking basic mathematical understanding. The system also serves the youths’ parents and professor(s).

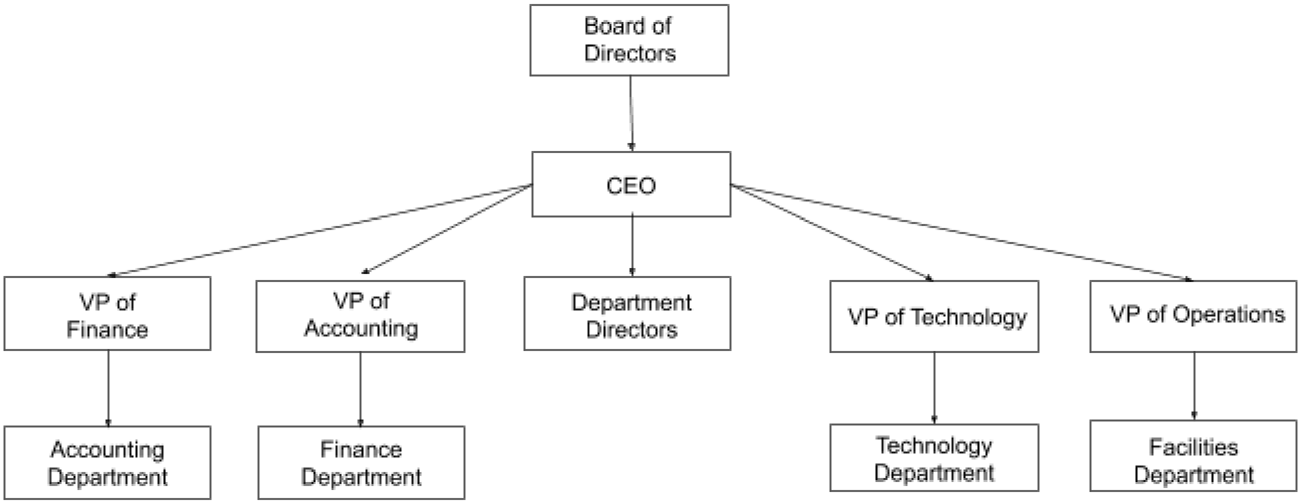
3.1.4.3 Other Systems Interaction

The system cannot interact with other systems of its kind due to a lack of network connectivity.

3.1.4.4 Context Diagram of Current System (WIP)



3.1.5 Current Business Structure Hierarchy Chart



3.2 Analyze Problems and Opportunities

3.2.1. Define Cause and Effect for Each Problem

The following chart summarizes the Cause and Effect Analysis, as well as, System Improvement Objectives. Since our system operates in a way that is unique, we compared it to similar products that we would consider competitors and the functionalities they provide.

Cause-and-Effect Analysis	
Problem or Opportunity	**Cause and Effects**
1. Current systems cannot be used offline	● Customers without access to the internet cannot access the application. ● A new system must be created to address these needs.
2. Method of accessing material is not user friendly	● Customers may get frustrated with using the system, and may just not use it all together.
3. Customer is not able to track their results	● Customers must be able to track the results of each practice session so that they know if they are improving or not.
4. Current system does not have a way to tailor learning sessions for students of different skill levels.	● Customers may not be receiving the appropriate lesson material and may not learn as much.
5. Current system does not have a problem set database.	● System must remember the problems that were given so that the customer can track if they ● All these sources do not come in a standard form so doctors must dig through unimportant information before relevant information appears.
6. Build an application that can run on all computers, and gives the user another way to access problems other than downloading them from the internet.	● When an application can run offline everyone can use it ● Creating a universally accessible application is good for the company image and maximizes the customer base.

3.3 Establish New System Improvement Objectives

3.3.1 New System Objectives

The system will be improved using the following objectives. These objectives are based on the problems and opportunities previously identified in the cause and effect analysis that was performed in 3.2.

Problem or Opportunity	System Objective
1. Current systems are not accessible offline	● Develop a program based application that aids in mathematics learning ● Must contain lessons for each of the primary four math operators ● Each session will have results for the user
2. Method of accessing the material is user friendly	● Develop a friendly and easy to use user interface. ● All input fields must be within a form. ● Math problems must be presented in a non intimidating fashion
3. Customer is able to track their results	● System must develop reports that are easy to read, yet comprehensive enough to cover all data. ● Each report must be associated with a score that the user can understand
4. Current system does have a way to tailor learning sessions for students of different skill levels.	● Develop an extremely simple way to switch between user accounts ● The new login system will help tailor the learning sessions to what the user needs to work on.
5. Current system does have a problem set database	● System can add and modify problem sets easily, so there is no stagnation of problems. ● All problems will be saved to a database.
6. Build an application that can run on computers without always needing to be online	● System will always be accessible to the user
7. Applications may run across all platforms.	● Applications must run on different platforms regardless of operating system. ● Mac OS, Android, Linux, windows

3.3.2 New System Constraints

3.3.2.1 Schedule

The OWL application will be analyzed and designed by 11/26/2019. This gives roughly three months to ensure the new system meets all requirements from a design perspective. After the design the system will be developed in about two to three months. The system will be evaluated bi-weekly to ensure that everything up to the most current milestone is working correctly. Timelines may change due to unforeseen circumstances, which may require different reporting intervals.

3.3.2.2 Cost

The development cost of the OWL system will require roughly (Reference Cost analysis report) in finance. In addition, annual operating costs are set to increase at a rate of 2% per year, whereas, revenue is projected to increase at 4%. At the current projection, the project is set to generate a positive return within the third year of operations.

3.3.2.3 Technology

The new system will not be exposed to any external network as the entire system will be accessible offline. All data will be stored in a database on a private server located onsite, database will be changed on a bi-annual basis. Data from the database will be retrieved via the application. The application can be deployed using a laptop or desktop computer that is updated with the appropriate version of either Mac OS or Windows.

3.3.2.4 Policy

A certification key will be required for granting permission to use the system. It is the consumer's responsibility to ensure that this key is not stolen or used by a third party to access the system. It is the responsibility of the user to ensure that the application is used in the appropriate manner.

3.4 Re-Evaluate and Update Project Scope

Project scope does not need to be re-evaluated.

4. Requirements Analysis Phase

4.1 Identify requirements (for objectives stated in 3.3.1)

4.1.1 Functional requirements in terms of inputs, outputs, processes, storage and control.

Functional Requirements

Requirement	Inputs	Outputs	Processes	Storage	Control
User-Friendly Program Based Application	The system shall accept students login credentials.	The system shall display every student's reports in graphical format.	The system shall create new students profile.	The system shall store profile to students database.	The system shall grant access to provider through validated login credentials.
All Input Fields Must be in a Form	The system shall allow new profiles for both referring professors and students to be created and later modified.	The system shall generate line graph of individual students' grades.	The system shall create new referring professor's profile.	The system shall store professor profile according to their grades.	The system shall allow preferred professors access to students personal information
Easy to Read Comprehensive Reports	Entries of all his grades	The system shall output a graph of all active students over time upon request.	The system shall search and retrieve existing student information.	The system shall save updated student information.	The system shall allow students access to the referring professor information

Integrated Referral Management System	Provider input text notes each visit and summary.	System shall display list of active students follow-up information.	The system shall search and retrieve existing referrer information.	The system shall save updated referring professor information.	The system shall allow his referrer access to his grades
Integrated Follow-Up Management System	Provider inputs dates	System shall display referred professors information.	The system will search and retrieve student information scheduled for follow-up.	System shall save edited dates	System shall allow professor access to follow-up students data.
Application Runs on Different Platforms Accessible with Laptops and Tablets	System authenticates provider's login credentials.	System shall display course information upon request.	The system shall run on multiple operating systems.	System shall maintain functionality across various platforms.	System shall access databases from remote devices off-line.

4.1.2. List and defend non-functional requirements

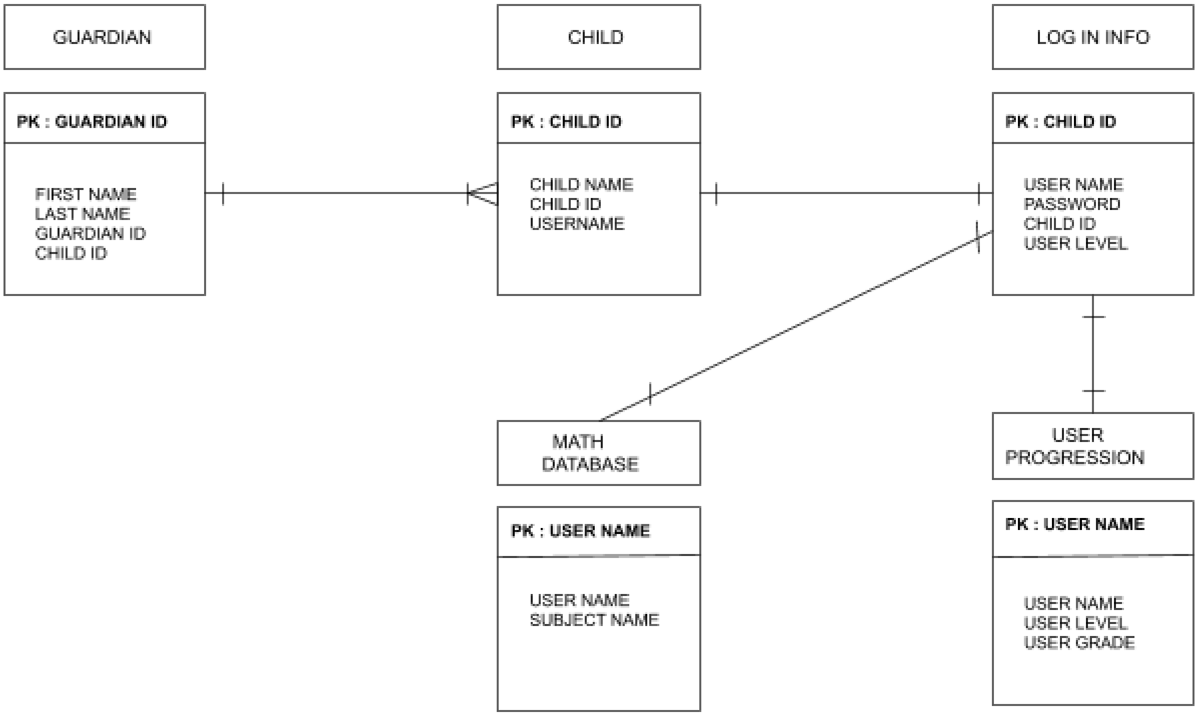
Non-Functional Requirements

Requirements	**Performance**	**Ease of use**	**Cost savings**	**Timelines and deadlines**	**Training**	**Quality Management**	**Security and Audits**
Colorful images	Response time < 360 milliseconds	N/A	10% IT services	Two weeks	N/A	Mean system errors < 5 per day	N/A
Follow-Up Requests	Not affected	Automatic biannual prompt	10% IT services	Two weeks	Follow instructions	Mean system errors < 5 per day	Secure data transmittal and storage

Assessment Test	Response time < 300 milliseconds	User selects “Assessment Test” in menu	30% IT services	One month	Follow instructions	Mean system errors < 10 per day Backup and restore time	Secure data transmittal and storage
Score report	Response time < 300 milliseconds	User selects “Score Report” in menu	20% IT services	One month	N/A	Mean system errors < 10 per day	Secure data transmittal and storage
Accessibility options	Added sounds--response time < 200 milliseconds	Auditory speech option, volume setting	20% of IT services	One month	N/A	Mean system errors < 5 per day	Options only apply to user logged in

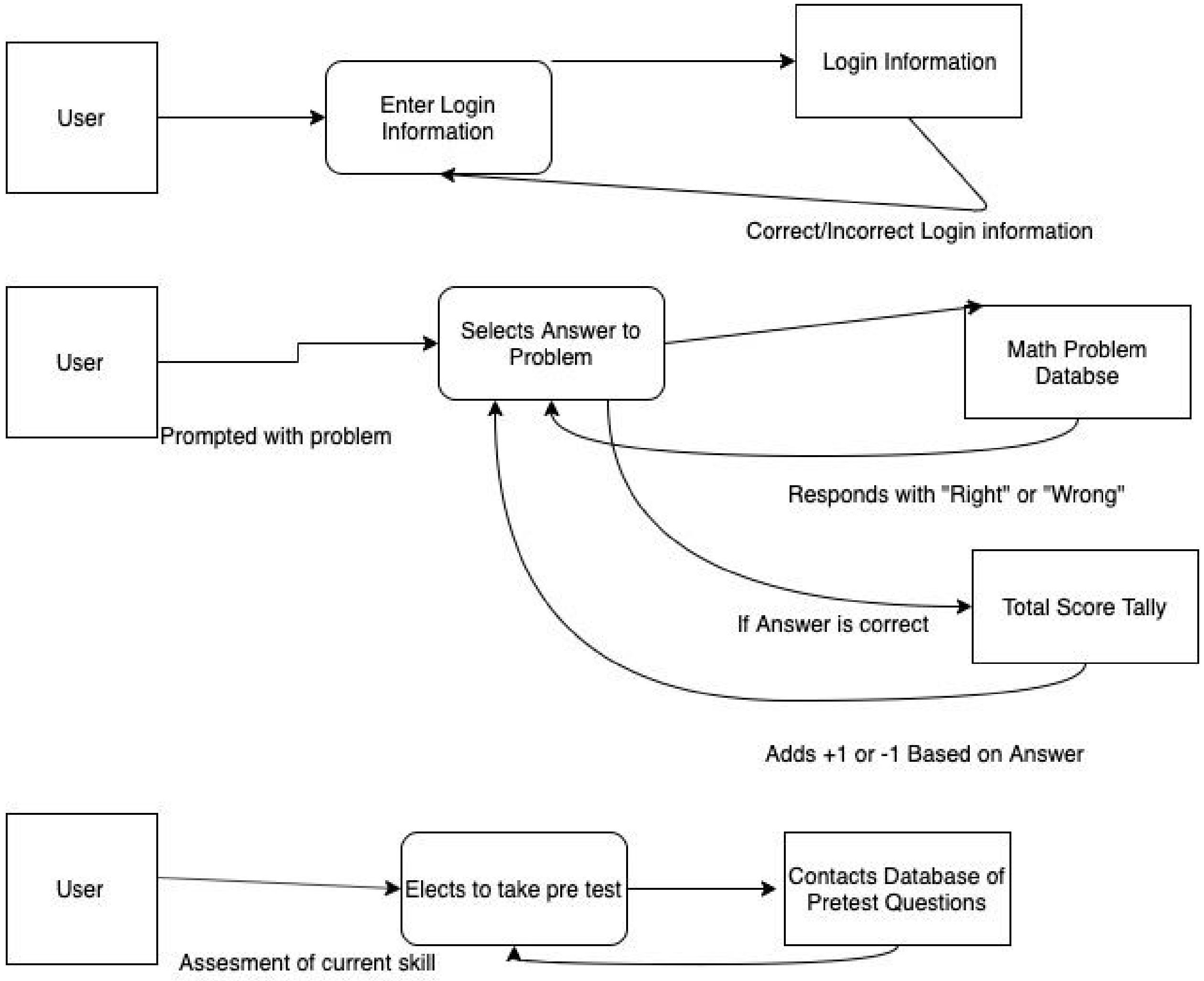
4.2. Analyze functional requirements for new system using system modeling approach

4.2.1. Construct preliminary data model - Entity Relationship (ER) diagram

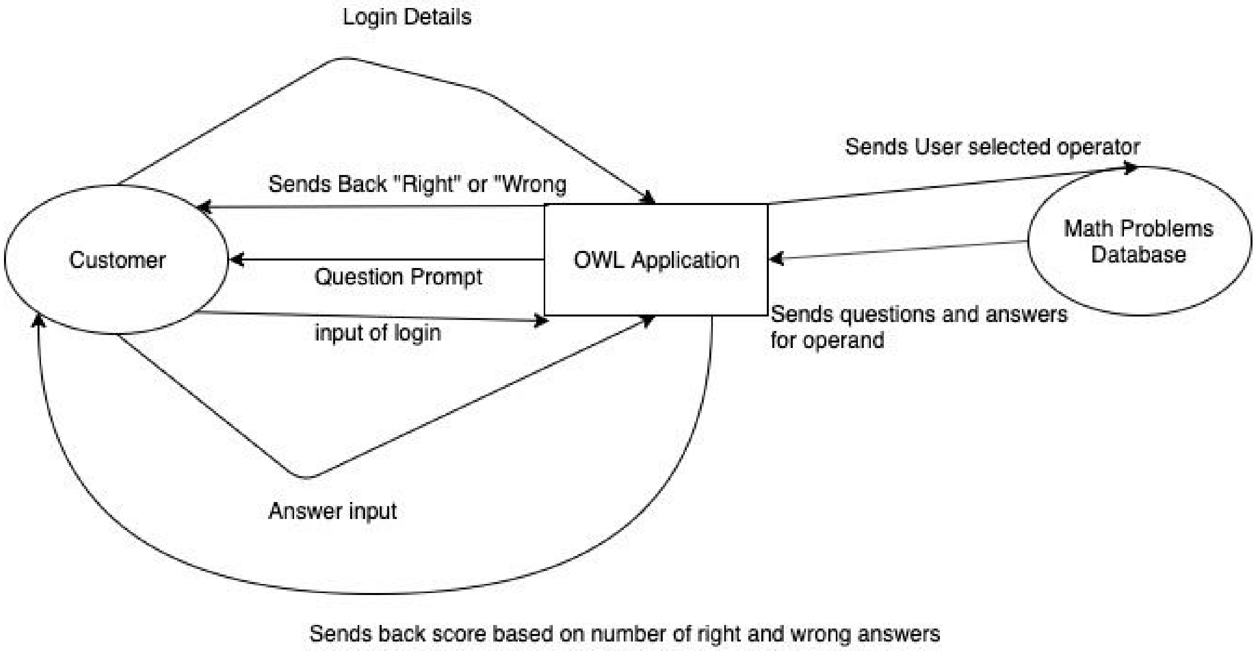


4.2.2. Construct preliminary process model - Data Flow diagram (DFD)

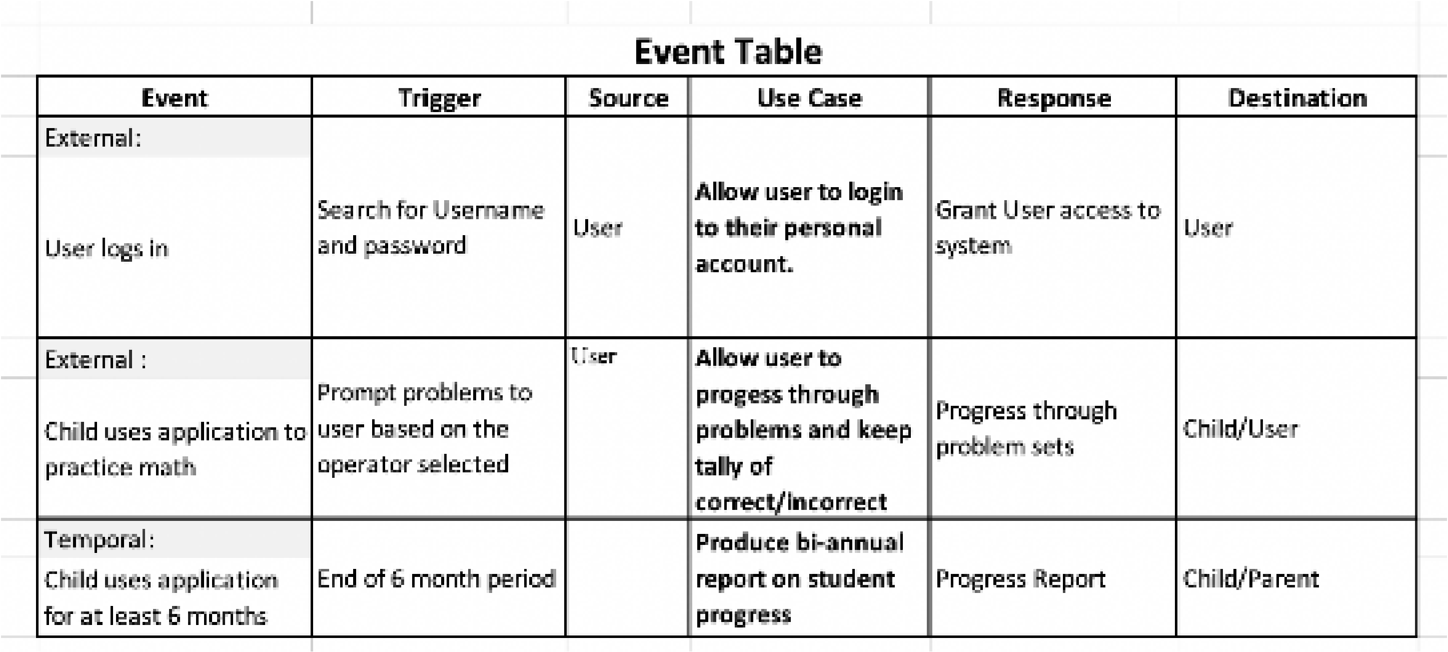
Preliminary DFD Diagram


User
Enter Login Information
Login Information
Correct/Incorrect Login information
User
Selects Answer to Problem
Math Problem Databse
Prompted with problem
Responds with "Right" or "Wrong"
Total Score Tally
If Answer is correct
Adds +1 or -1 Based on Answer
User
Elects to take pre test
Contacts Database of Pretest Questions
Assesment of current skill

4.2.3. Construct preliminary Interface model - Context diagram


Login Details
Sends User selected operator
Sends Back "Right" or "Wrong
Math Problems Database
Customer
OWL Application
Question Prompt
input of login
Sends questions and answers for operand
Answer input
Sends back score based on number of right and wrong answers

Event Table



Event Table

Event	Trigger	Source	Use Case	Response	Destination
External: User logs in	Search for Username and password	User	**Allow user to login to their personal account.**	Grant User access to system	User
External : Child uses application to practice math	Prompt problems to user based on the operator selected	User	**Allow user to progess through problems and keep tally of correct/incorrect**	Progress through problem sets	Child/User
Temporal: Child uses application for at least 6 months	End of 6 month period		**Produce bi-annual report on student progress**	Progress Report	Child/Parent

4.3. Master List of all Requirements in Terms of:

4.3.1 Priority

Functional Requirements

Requirements	Priority
User-Friendly Program Based Application	High
All Input Fields Must be in a Form	High
Easy to Read Comprehensive Score Reports	High
Application Runs on Different Platforms Accessible with Laptops and Tablets	Medium

Non-Functional Requirements

Requirements	Priority
Colorful images	High
Follow-Ups	High
Assessment Test	High
Score report	Medium
Accessibility options	Medium

4.3.2 Deadline

Functional Requirements

Requirements	Timelines and Deadlines
User-Friendly Program Based Application	One month to test and code.
All Input Fields Must be in a Form	One month to test and code.
Application Runs on Different Platforms Accessible with Laptops and Tablets	Three months to test and code.

Non-Functional Requirements

Requirements	Timelines and Deadlines
Colorful images	Two weeks to test and code.

Assessment Test	One month to test and code.
Score report	One month to test and code.
Accessibility options	One month to test and code.

4.3.3 Supporting Requirements

In the database, an important requirement to have would be, database maintenance to reassure and incorporate that the data is functioning properly through the updates and patches. This would be maintained bi-annually with it only being accessed to the internet when available. Another supporting requirement, would be a Customer Service Representative (CSR), which would assist the customer's inquiries and problems with the program. It would also provide customer feedback surveys to help the company improve.

A supporting requirement would also entail a development team to maintain the application and make sure it is running properly and fix any issues, when a problem occurs.

The most important supporting requirement for our application would be security. Security would be provided to ensure nothing is being tampered with and that our application is safe, secure, even though everything will be offline while being used. The database will be secure as well as the payment will be stored in a backup file on the company's system computer

4.4. Re-evaluation and Update Project Scope

Project scope does not need to be re-evaluated.

5. Decision Analysis Phase

5.1. Identify Candidate Solutions

As previously mentioned, the current alternative to our learning application is classroom learning which is simply not cutting it in today's day and age. Today's students need a more engaging education delivery method that keeps the students attention while also maintaining consistency. Engagement and consistency are key factors in a child's education and are lacking in many school curriculums. The following feasibility analysis goes further into detail about the general feasibility associated with traditional school education versus the proposed OWL system.

5.2. Analyze Candidate Solutions.

5.2.1. Feasibility Analysis

	Operational Feasibility
Elementary Classroom	A child can certainly learn basic math in the classroom, but it is going to be rather difficult to teach them. The classroom can provide many distractions for an elementary student to focus on learning math. It is also a challenge to ensure that every single student in a teacher's class is at the same level when it comes to learning math.
OWL	The OWL system provides a personalized approach to learning math without the distractions of a classroom. Children will be able to focus on what they need to learn rather than focus on what they already know. It provides a fun way to learn as well, as many students will think of the system as a game.

	Technical Feasibility
Elementary Classroom	While it certainly is practical to learn math in a classroom, some students do struggle to learn in this setting. Teachers are capable of helping them, but it can be difficult to focus on just one student out of the 20+ others.
OWL	The solution is technically practical because of how simple it is to create. No expert staff is needed to build and/or operate the system.

	Economic Feasibility
Elementary Classroom	Going to school is definitely cost-effective, as children in elementary schools learn more than just mathematics. However, because of how many different topics are taught in the classroom, parents might consider the OWL an economically sound approach for if their child

	is struggling in the math department.
OWL	The OWL solution is very cost-effective because of how cheap it is to make. It is also available for purchase at cheap prices.

	Schedule Feasibility
Elementary Classroom	Children attend elementary school during certain times of the day, as well as certain times of the year.
OWL	OWL solutions can be designed and implemented within a year. The user of the system can then use OWL to learn math year round.

5.2.2. Cost-Benefit Analysis

5.2.1. Chart Cost Analysis of Design

Development Costs					
Personnel Costs					
Position	**Hourly Rate**	**Hours**	**# of Employees**	**Cost**	**Formulas**
System Analyst	$60	200	1	$12,000	B3 x C3
Visual Basic Programmers	$40	300	2	$24,000	B4 x C4 x D4
System Architect	$60	100	1	$6,000	B5 x C5
Database Specialist	$40	300	2	$24,000	B6 x C6 x D6
Total Personnel Cost				**$66,000**	**Sum(E3:E6)**

5.2.2.2 Chart Cost Analysis of Hardware, Software, Miscellanies Purchases

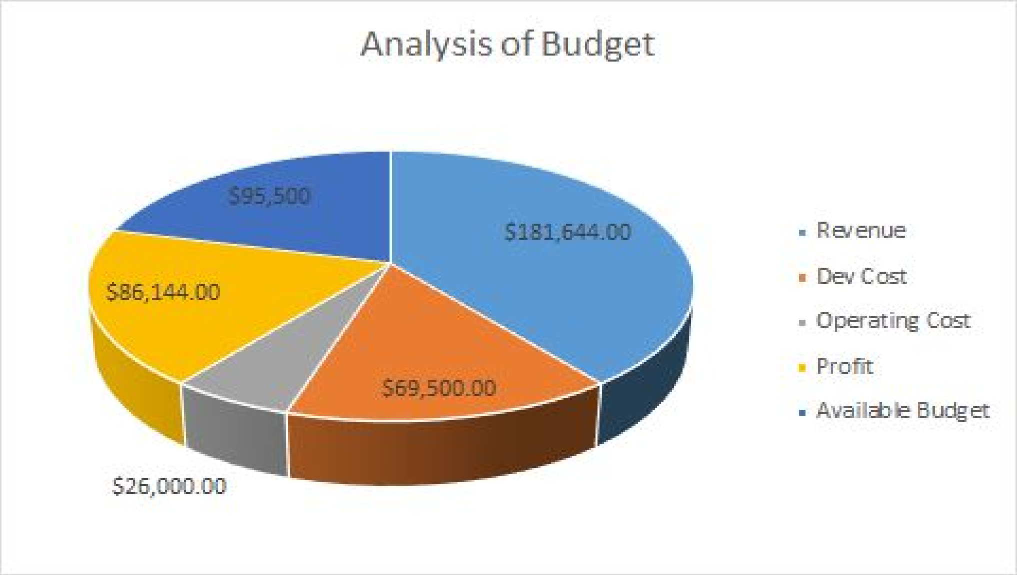
Hardware Costs				
Equipment	**Quantity**	**Unit Cost**	**Cost**	**Formulas**
Tablets	5	$500	$2,500	B3 x C3
Server	1	$1,000	$1,000	B4 x C4
Total Equipment Cost			**$3,500**	**D3 + D4**

Projected Annual Revenue Analysis						
Revenue Category	**Year 1**	**Year 2**	**Year 3**	**Year 4**	**Year 5**	**Total**
System purchase revenue	$23,496	$25,376	$27,406	$29,598	$31,966	$137,842
Private Donors	$4,000	$4,320	$4,666	$5,039	$5,442	$23,467
Fundraiser	$3,467	$3,744	$4,043	$4,366	$4,715	$20,335
Total Revenue (Sums of each year's different forms of revenue, followed by grand total)	**$30,963**	**$33,440**	**$36,115**	**$39,003**	**$42,123**	**$181,644**
Formulas	**Year 1**	**Year 2**	**Year 3**	**Year 4**	**Year 5**	**Totals**
	$23,496	B3 * 1.08	C3 * 1.08	D3 * 1.08	E3 * 1.08	Sum(B3:F3)
	$4,000	B4 * 1.08	C4 * 1.08	D4 * 1.08	E4 * 1.08	Sum(B4:F4)
	$3,467	B4 * 1.08	C5 * 1.08	D5 * 1.08	E5 * 1.08	Sum(B5:F5)
	Sum(B3:B5)	**Sum(C3:C5)**	**Sum(D3:D5)**	**Sum(E3:E5)**	**Sum(F3:F5)**	**Sum(B6:F6)**

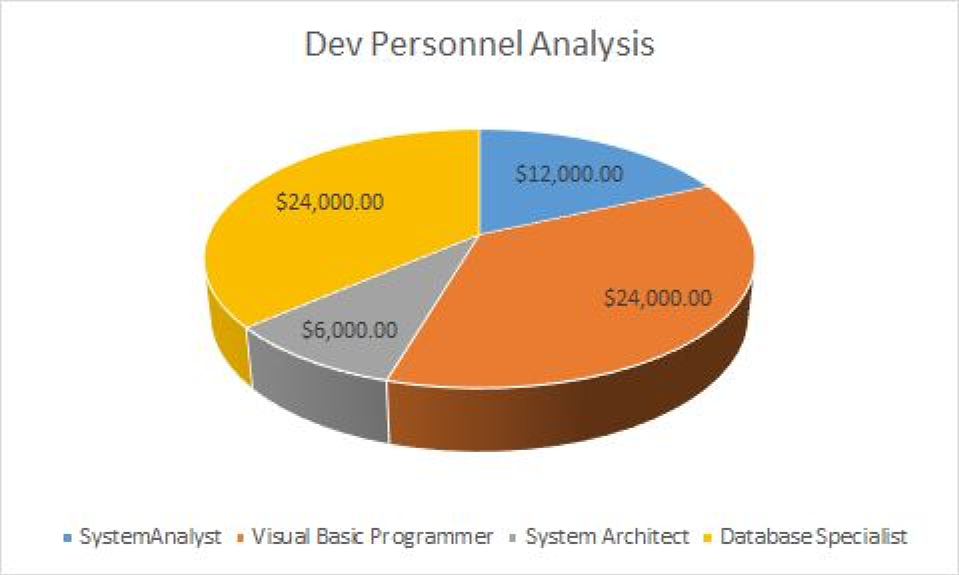
5.2.2.3 Chart Cost Analysis of System Operation, and Maintenance

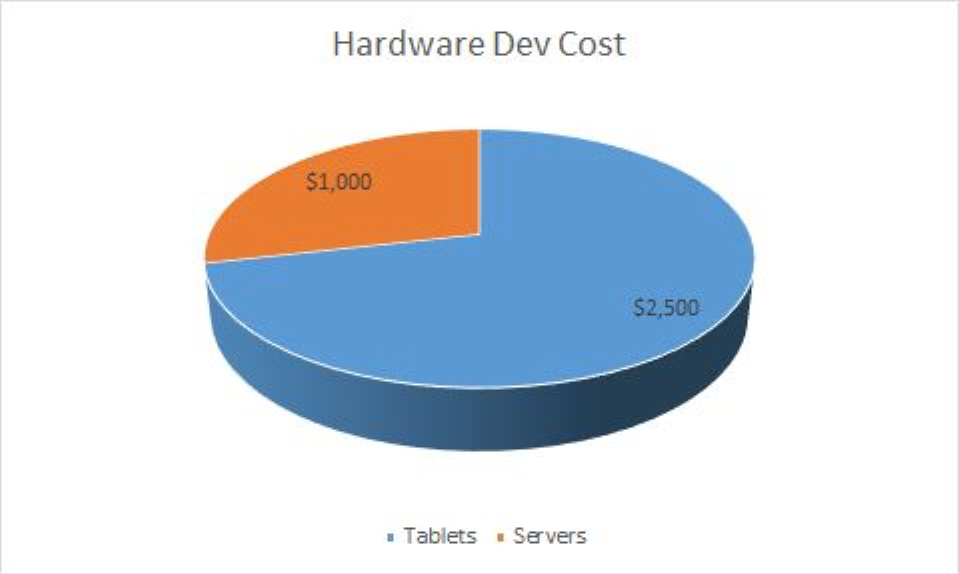
Projected Annual Operating Costs							
Position	**# of Employees**	**Year 1**	**Year 2**	**Year 3**	**Year 4**	**Year 5**	**Total**
System Maintenance	1	$5,000	$5,100	$5,200	$5,300	$5,400	$26,000
Total Annual Cost		$5,000	$5,100	$5,200	$5,300	$5,400	$26,000
Formulas			C3 * 1.02	D3 * 1.02	E3 * 1.02	F3 * 1.02	Sum(C3:F3)

5.2.2.4. Chart Cost Analysis of Budget

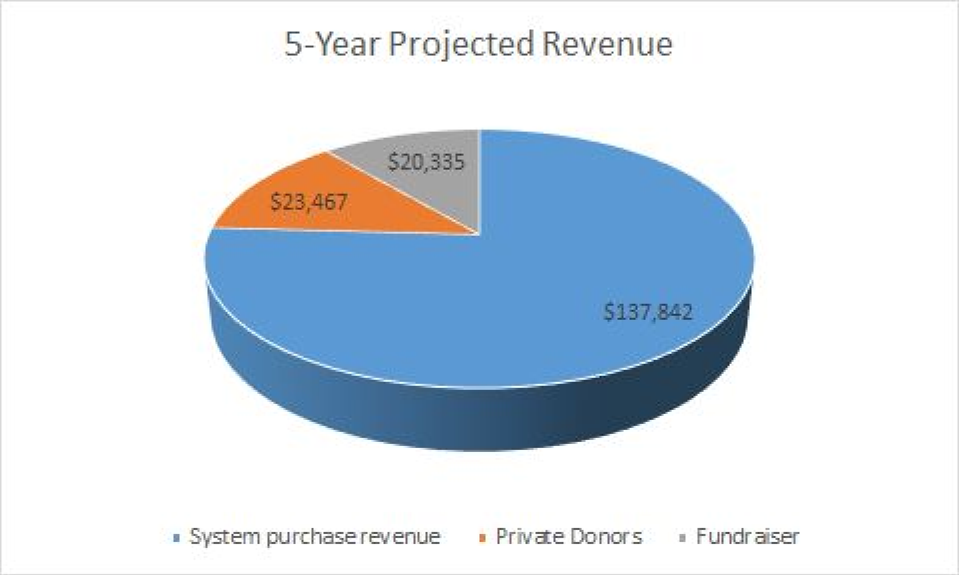


5.2.2.4. Chart Cost Analysis of Development





5.2.2.5. Chart Cost Analysis of Revenue



5.3. Compare Candidate Solutions

Characteristic	Offline While Learning (OWL)	Math Worksheet / Classroom Learning
Operating Environment or Constraints	● Windows 10, Windows 8+, Windows 7+ ● Mac OS 10.11+ (El Capitan), 10.10+ (Yosemite) and 10.9+ (Mavericks)	● Microsoft Windows XP or later ● Windows Compatible Web Browser with current updates/patches
Hardware New and Existing	Windows ● 2+ GHz processor. ● 8+ GB RAM. ● 500 MB of available hard-disk space ● SQL Database Server Mac ● 2.1+ GHz Intel™ processor. ● 8 GB RAM. ● 500 MB of available hard-disk space.	● N/A

Software New and Existing	Windows ● Microsoft® Internet Explorer 9.0 or higher ● Microsoft .NET Framework 4.0 ● Visual Studio 2013 ● SQL Server Mac ● QuickTime 5.0 and web browser for multimedia tutorials. ● Safari. Adobe ● Adobe Acrobat Reader Version 10	● N/A
Staffing New and Existing	● Database Specialist	● Teacher
Training New and Existing	● Tutorials ● Intuitive GUIs ● Embedded Help function	● Teaching Seminars
Installation Requirements	● Microsoft Windows Installer 3.0	● N/A
Performance Requirements	● Data storage backup and recovery time < 30 minutes ● 98% of all response time < 2 seconds	● In class assess ● (OCC)
Development Requirements	● Client-Server model ● SQL Database	● None
Reports to be Delivered	● Metadata view to enable analysis and reporting across disparate sources.	● Timely Assessment Reports
Security Requirements	● Access intranet information through the Internet ● SSL secure authorization	● Secure login
Auditing Requirements	● Audit trails from access logs ● Code review	● Teaching audits

5.4. Recommended Final "Best" Solution

5.4.1. Operating Environment or Constraints (see chart below)

5.4.2. Hardware New and Existing (see chart below)

5.4.3. Software New and Existing (see chart below)

5.4.4. Staffing New and Existing (see chart below)

5.4.5. Training New and Existing (see chart below)

5.4.6. Installation Requirements (see chart below)

5.4.7. Performance Requirements (see chart below)

5.4.8. Development Requirements (see chart below)

5.4.9. Reports to be Delivered (see chart below)

5.4.10. Security Requirements (see chart below)

5.4.11. Auditing Requirement (see chart below)

Criteria	**New (Recommended)**	**Existing (Current)**
Operating Environment or Constraints	● Data connection is NOT necessary for interactive chart reporting. ● Off-line solution available for viewing stored database data.	● Data connectivity required for all functionality.
Hardware New and Existing	Windows ● 2+ GHz processor. ● 8+ GB RAM. ● 500 MB of available hard-disk space ● SQL Database Server Mac ● 2.3+ GHz Intel™ processor. ● 8 GB RAM. ● 500 MB of available hard-disk space.	● Math Sheets ● Pen / Pencil / Marker ● Number chart ● Number line
Software New and Existing	● In-House development ● Control over software improvements ● All of the required features implemented ● Sustains core competencies and will adhere to a higher level quality service	● Excel spreadsheet for tracking to track the correct answer of each problem
Staffing New and Existing	● IT personnel minimal ● Developers	● Learner ● Mentor ● Parent
Training New and Existing	● 1 hour tutorial to introduce the functions and capabilities	● No training ● Based on school knowledge

		● Repetition to learn
Installation Requirements	● Microsoft Windows Installer 3.0	● No installation
Performance Requirements	● 98% Pages loaded within 2 seconds of request	● 25 math sheets per math skill
Development Requirements	● SQL Database	● Hand written
Reports to be Delivered	● Customer is able to track their results	● Tracking would be in a chart form which can manually expand into graphs
Security Requirements	● Individual logins ● SSL secure authorization	● No security
Auditing Requirements	● Login and access log trails	● Verbally saying the problems as it is being written out.

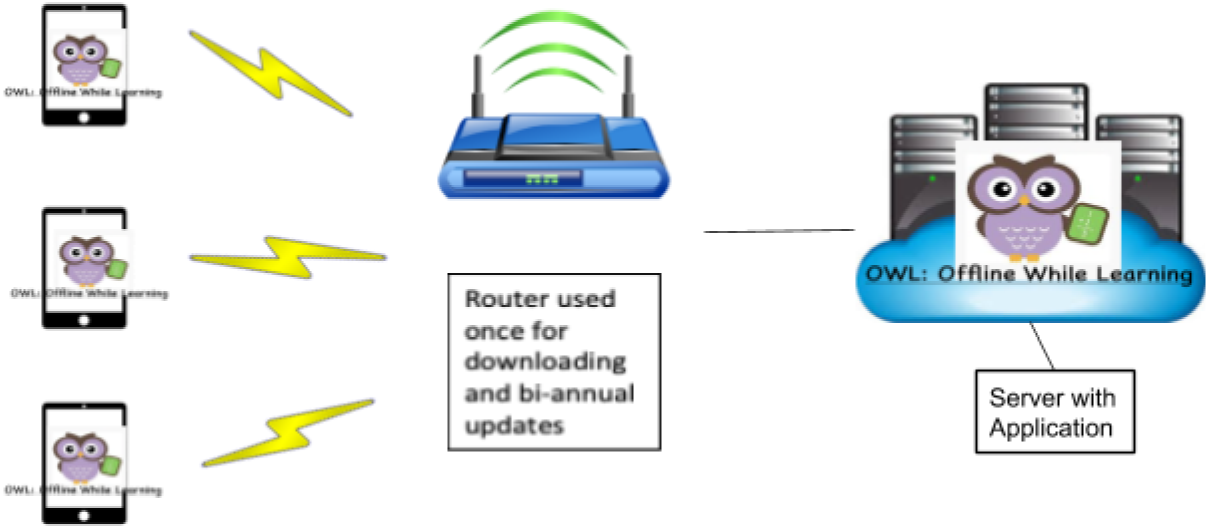
6. Design Phase

6.1 Design the Application Architecture

The application architecture is very simple and the interface is easy to follow, especially for children over the age of 4 and those who are technology savvy. The main reason is that the application does not require external user access, so internet functionality is not needed. A router will only be used for the initial application download, as well as, the bi-annual updates and security patches. The second reason is to optimize the applications ability to help children who need extra help and assistance while learning math. By having an application designed to be free of internet it will force the user to practice math instead of going to other sites or following ads that they might use while using online products. By minimizing the potential for distractions, students will have more effective study sessions and retain more information.

The application will be available on tablet or computer-based devices which will be used by the end user (i.e. student) to access the application on the device with a locally stored database. The server will be used to download the application install and update files.

The tablets will be used by the end user (i.e. child / student) to access the local application and save data in the local database.


OWL: Offline While Learning
OWL: Offline While Learning
OWL: Offline While Learning
Router used once for downloading and bi-annual updates
OWL: Offline While Learning
Server with Application

6.1.1 Networks – Intranet

- The user will use tablets or computers to access the local application via local database (no external internet connection).
- The server will be used for initial application download and bi-annual updates only.

6.1.2 Database Distribution

- Individual tablets or computers will house locally stored databases.
- The primary server will be located at the company and have a redundant system (one mirrored back-up server).
- Servers will be accessed using the router when needed for initial installs and bi-annual updates.

6.1.3 "Off the Shelf" Software

A Microsoft Access Database is the supporting database software included internally with the OWL application install.

6.1.4 User Interface Technology – With Other Users

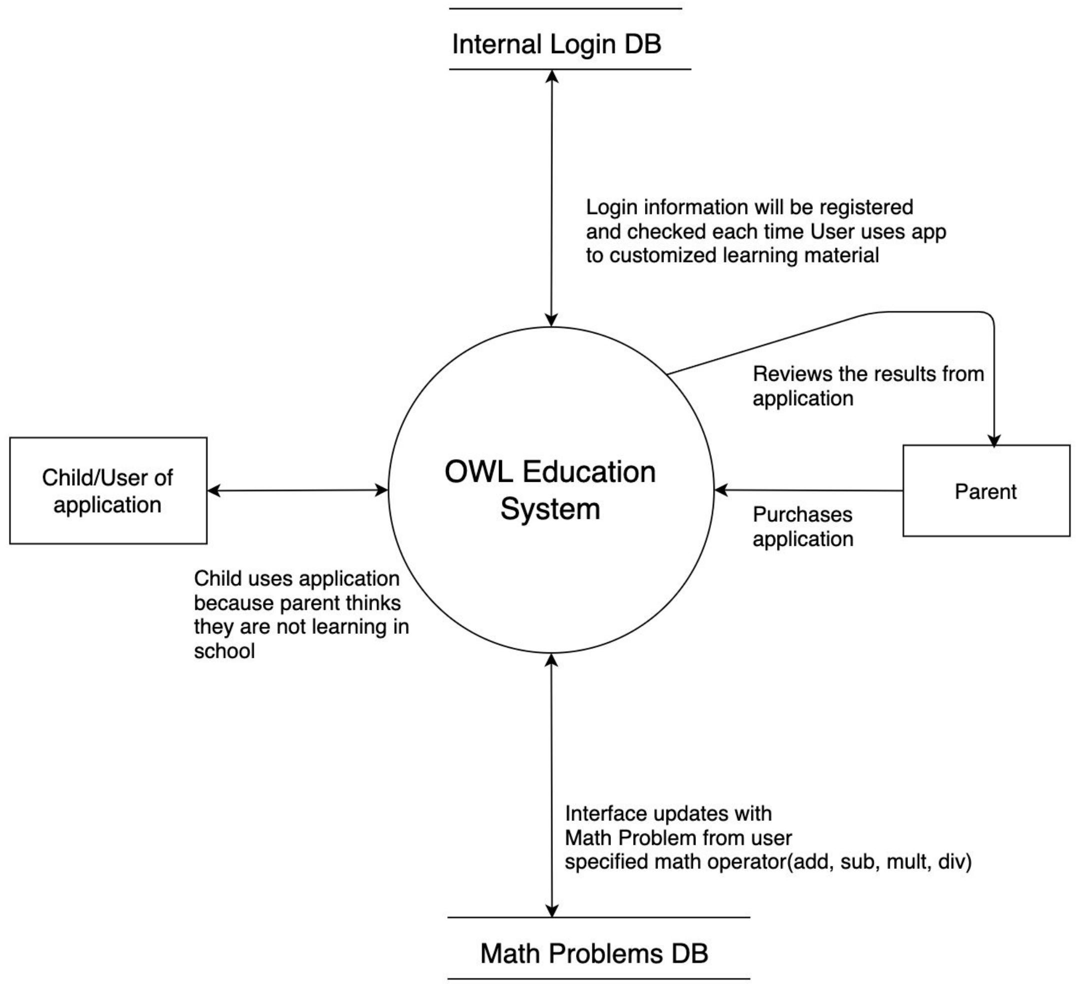
No other users besides the provider will interface with the technology.

6.1.5 System Interface Technology – With other Systems

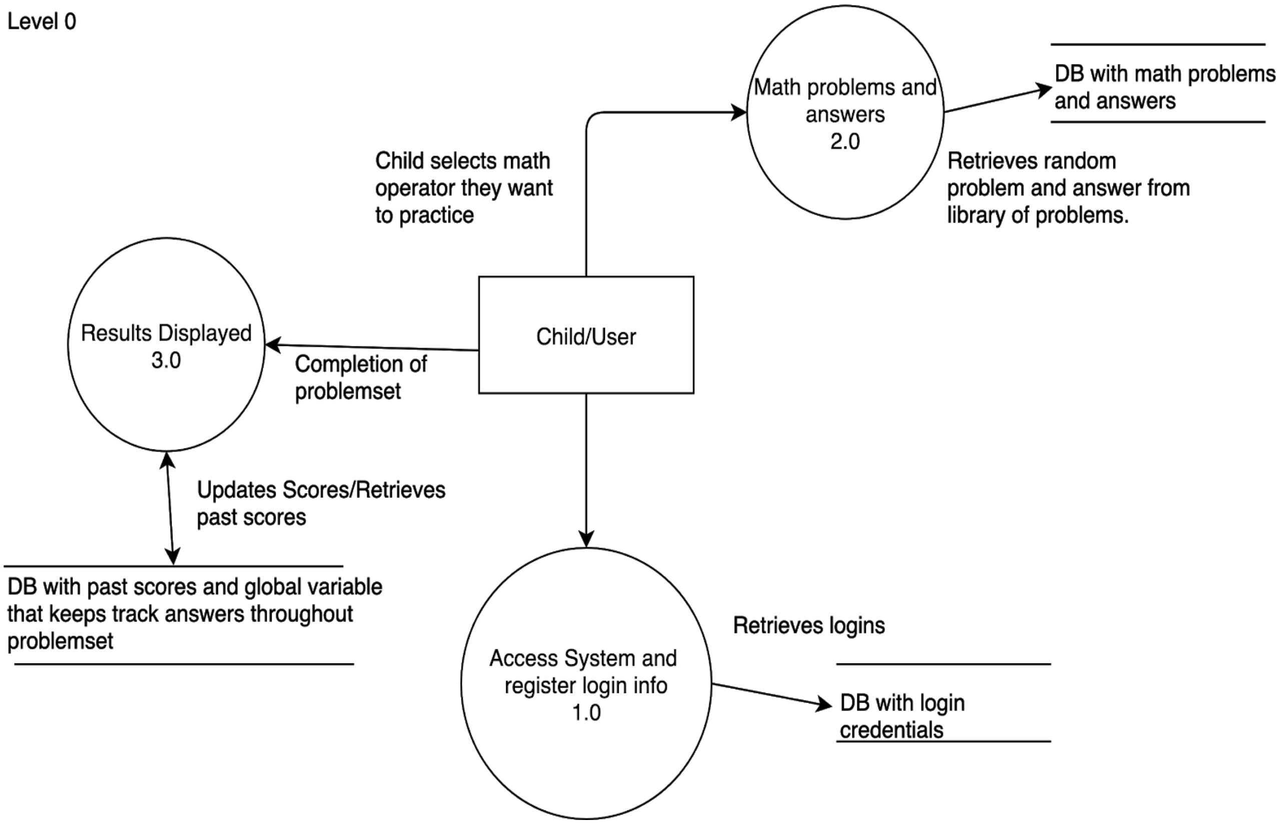
The system will not interface with any other system.

6.2. Construct Detailed Models

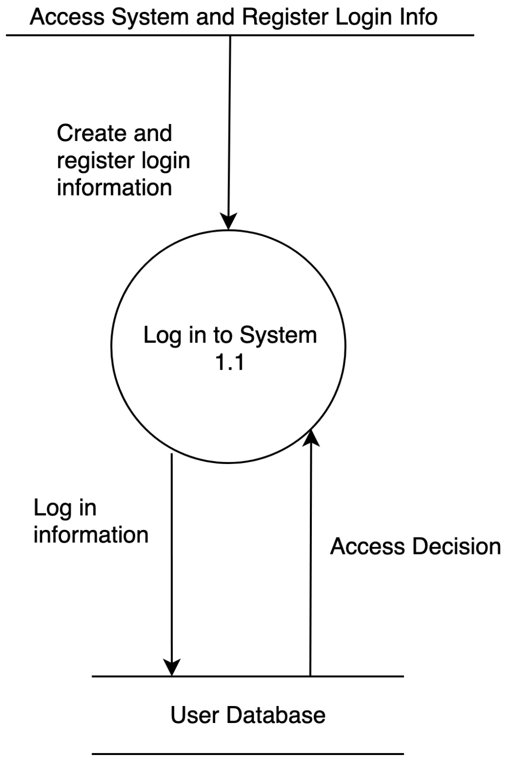
6.2.1. Context Model


Internal Login DB
Login information will be registered and checked each time User uses app to customized learning material
Reviews the results from application
Child/User of application
OWL Education System
Parent
Purchases application
Child uses application because parent thinks they are not learning in school
Interface updates with Math Problem from user specified math operator(add, sub, mult, div)
Math Problems DB

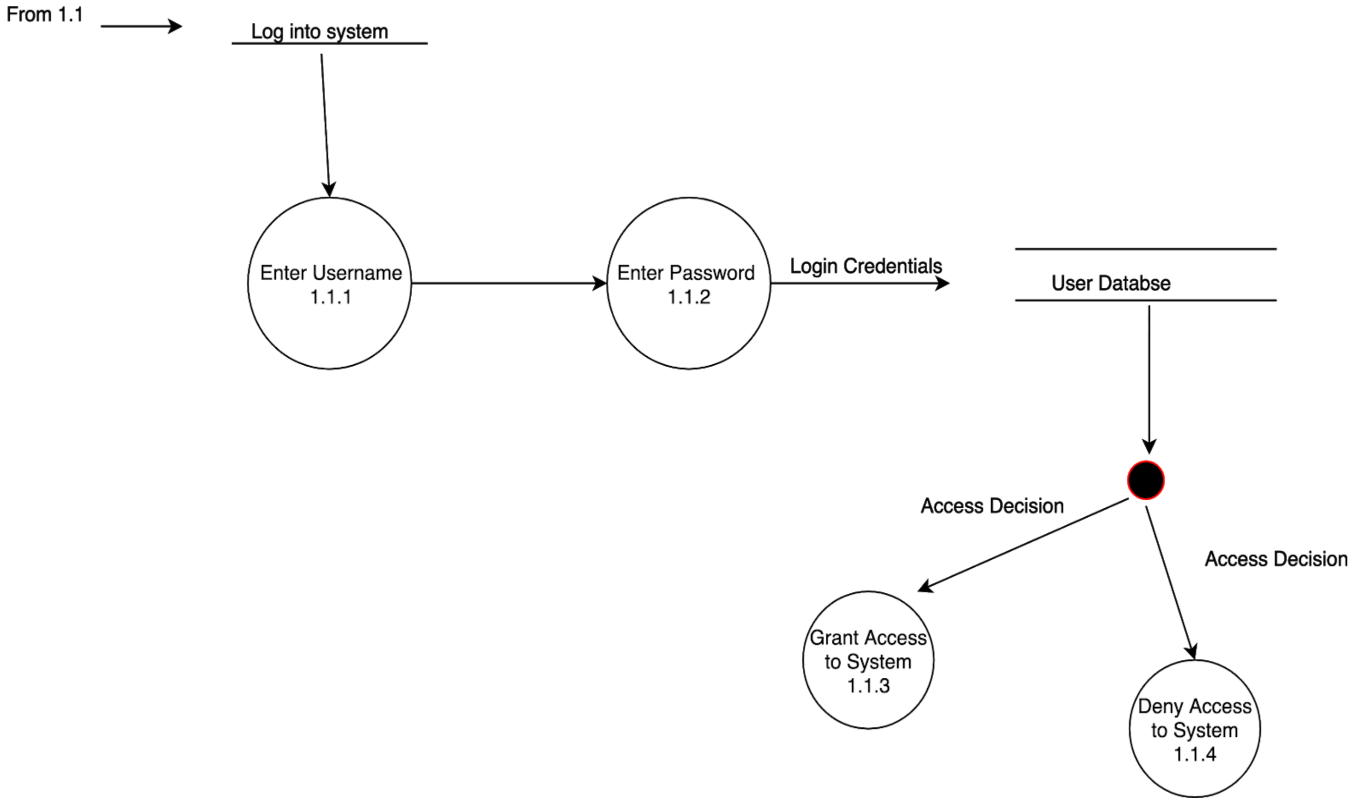
6.2.2. Data Flow Diagram (DFD) Decomposition to System Modules, and Tasks


Level 0
Math problems and answers
2.0
DB with math problems and answers
Child selects math operator they want to practice
Retrieves random problem and answer from library of problems.
Results Displayed
3.0
Child/User
Completion of problemset
Updates Scores/Retrieves past scores
DB with past scores and global variable that keeps track answers throughout problemset
Access System and register login info
1.0
Retrieves logins
DB with login credentials

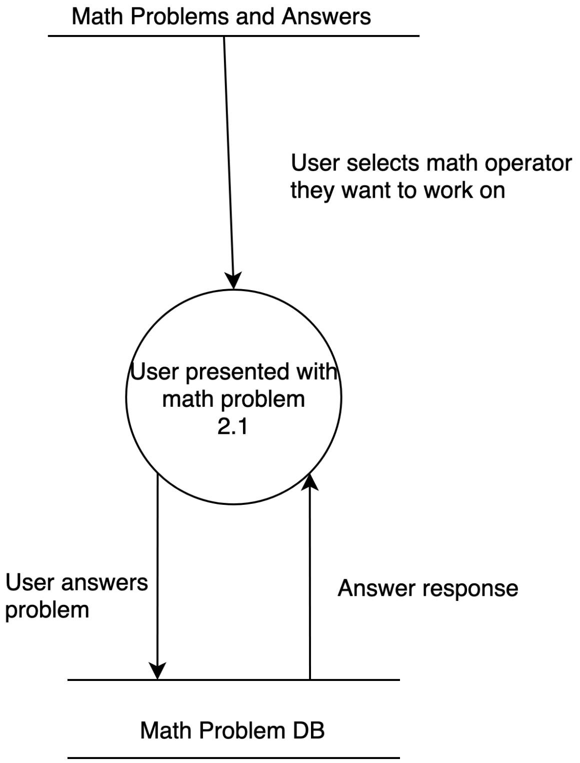
Level 1


Access System and Register Login Info
Create and register login information
Log in to System
1.1
Log in information
Access Decision
User Database

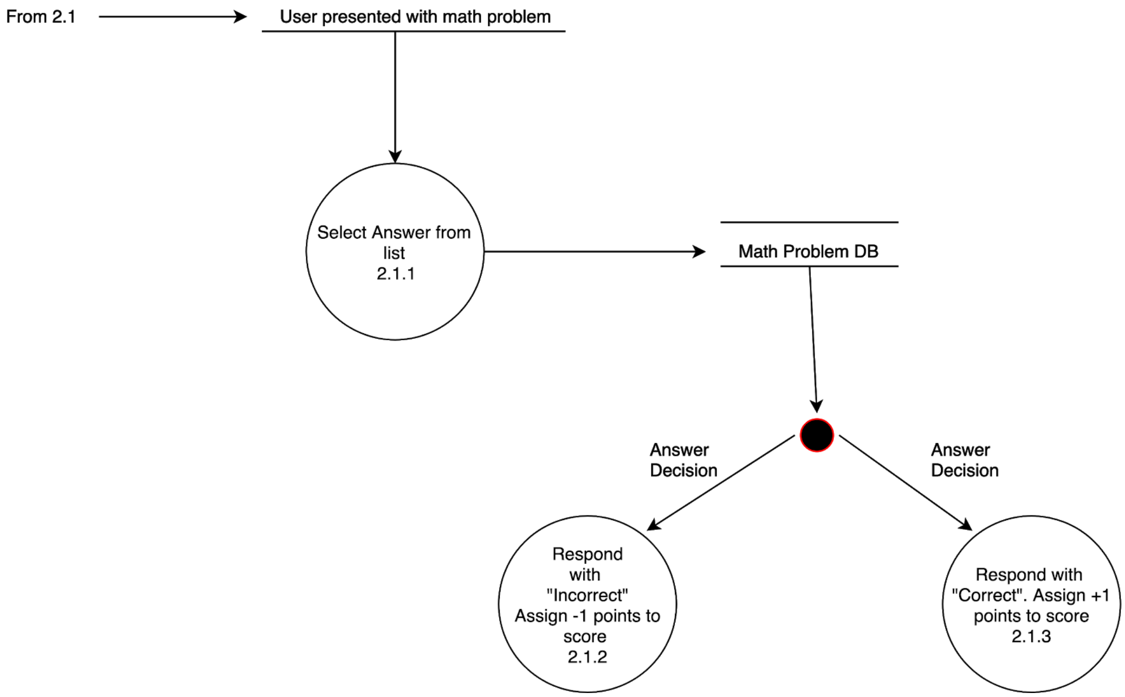
Level 1.1



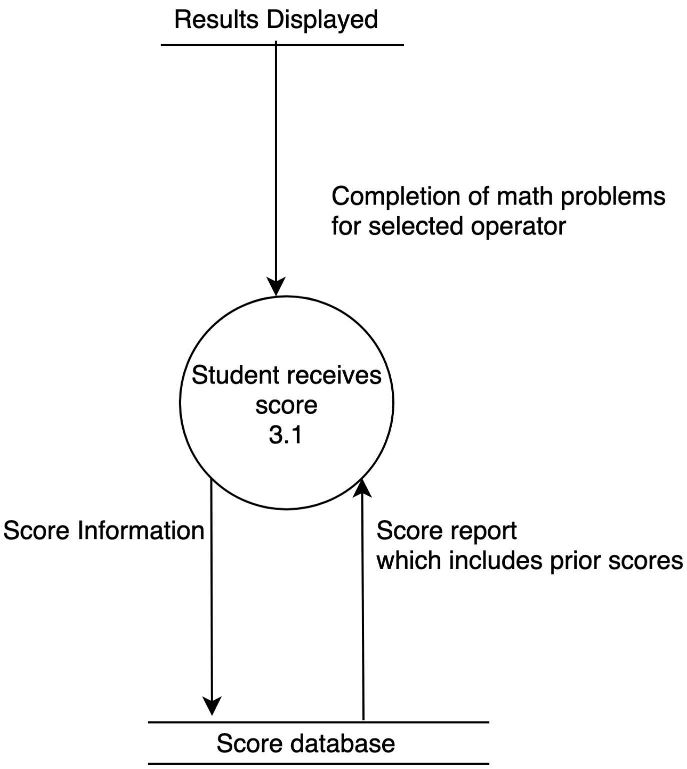
Level 2



Level 2.1

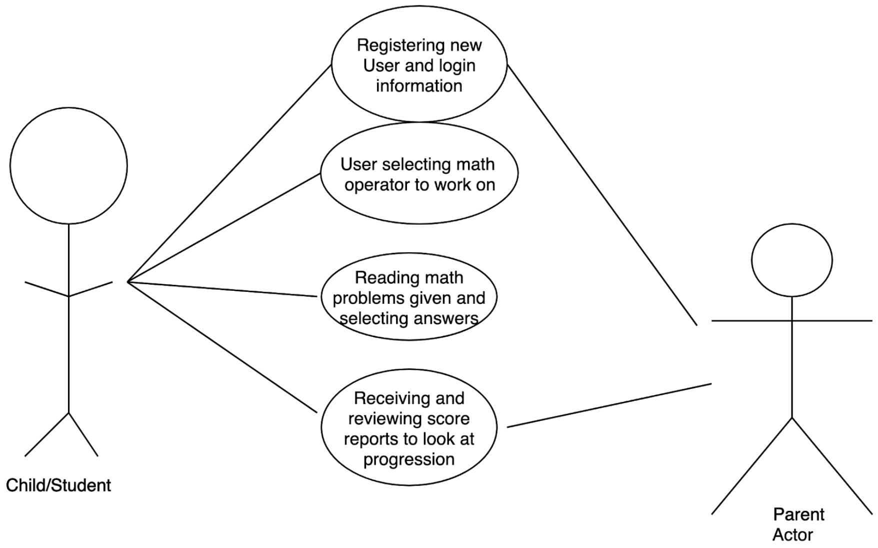

From 2.1
User presented with math problem
Select Answer from list
2.1.1
Math Problem DB
Answer Decision
Answer Decision
Respond with "Incorrect" Assign -1 points to score
2.1.2
Respond with "Correct". Assign +1 points to score
2.1.3

Level 3

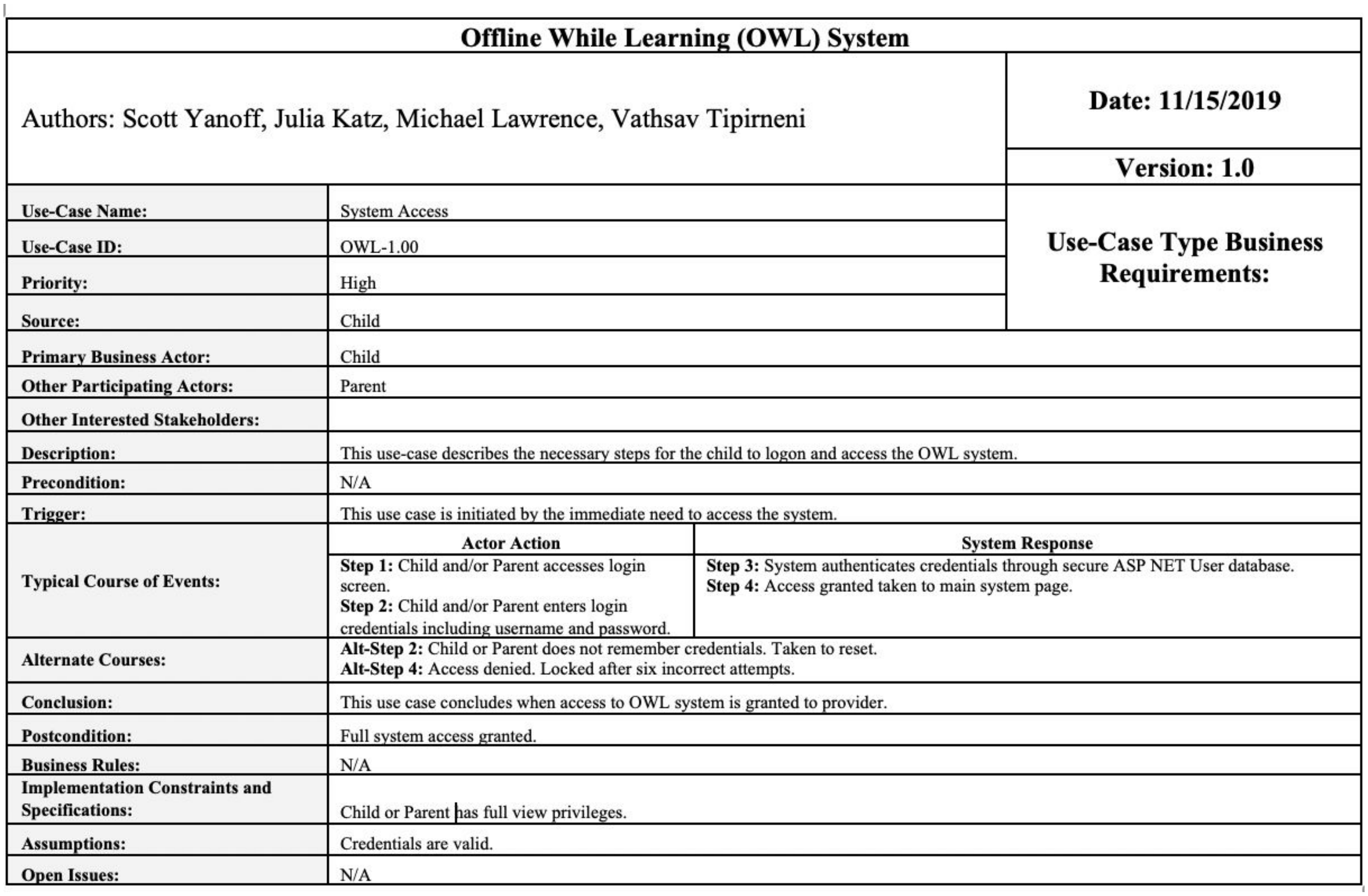

Results Displayed
Completion of math problems for selected operator
Student receives score
3.1
Score Information
Score report which includes prior scores
Score database

6.2.3. Use-Case model diagram with Use-Case Narratives

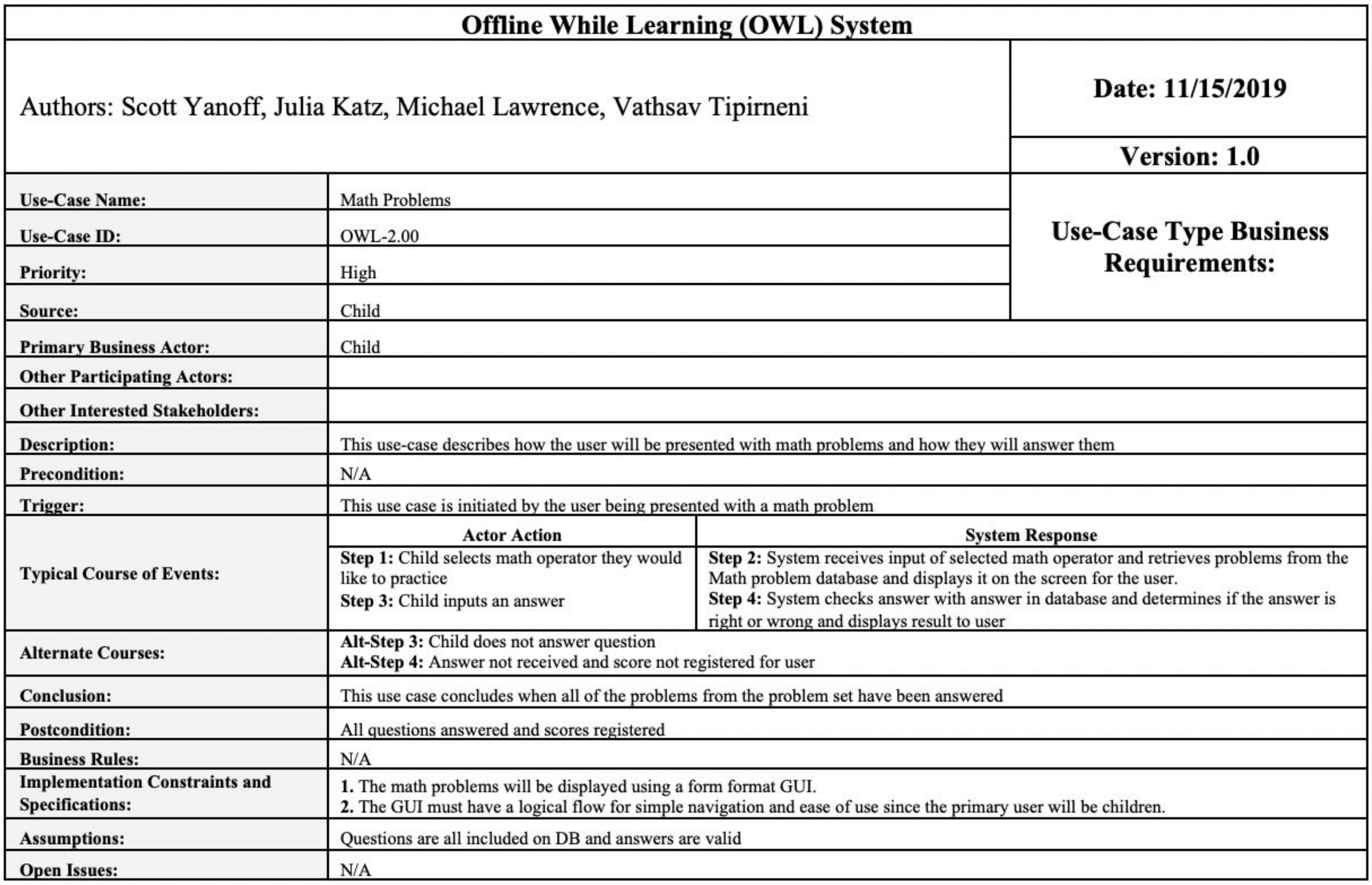
Use Case Diagram Model


Registering new User and login information
User selecting math operator to work on
Reading math problems given and selecting answers
Receiving and reviewing score reports to look at progression
Child/Student
Parent Actor

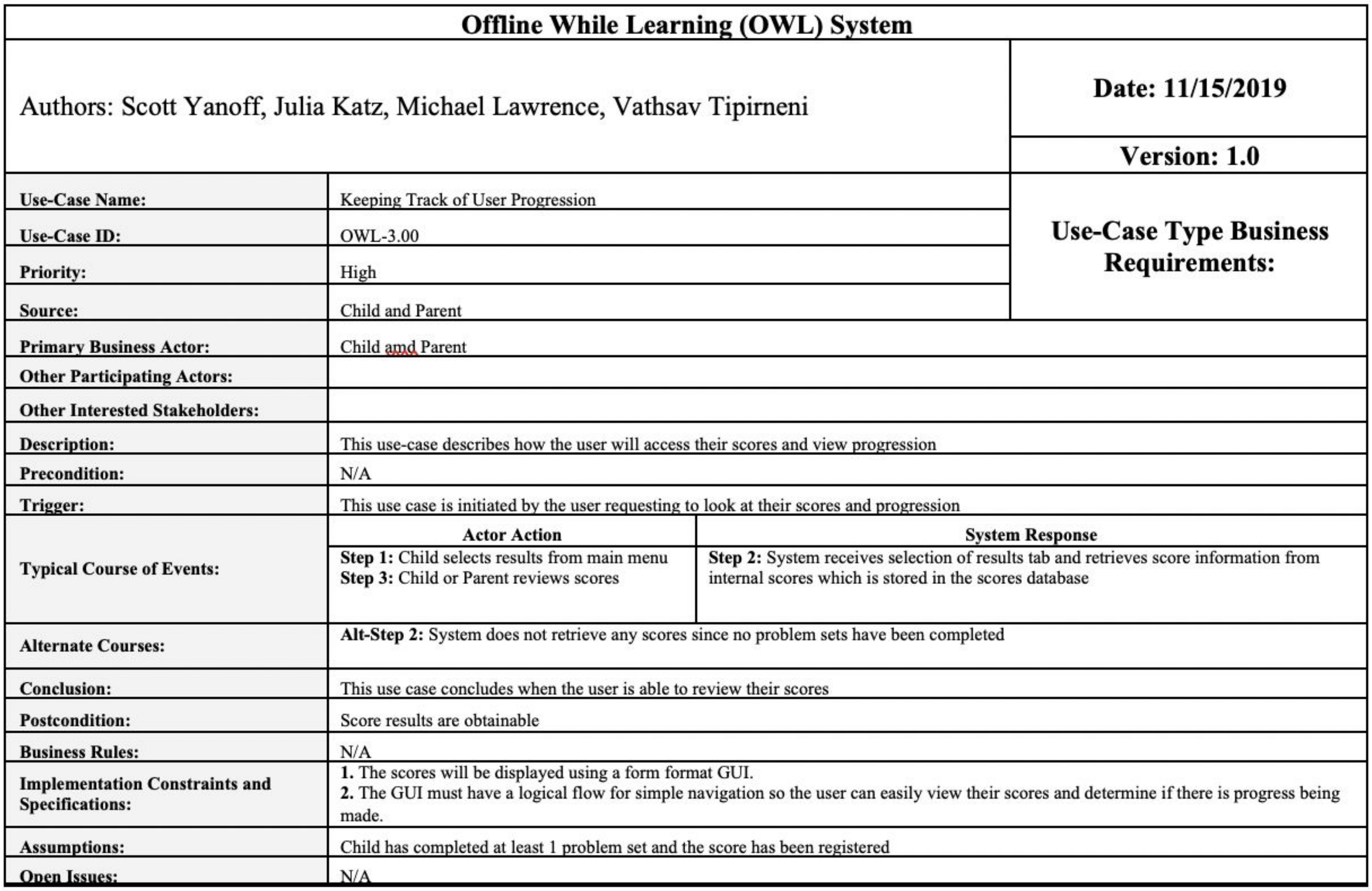
Use Case Narratives



Offline While Learning (OWL) System		
Authors: Scott Yanoff, Julia Katz, Michael Lawrence, Vathsav Tipirneni		**Date: 11/15/2019**
		Version: 1.0
Use-Case Name:	System Access	**Use-Case Type Business Requirements:**
Use-Case ID:	OWL-1.00	
Priority:	High	
Source:	Child	
Primary Business Actor:	Child	
Other Participating Actors:	Parent	
Other Interested Stakeholders:		
Description:	This use-case describes the necessary steps for the child to logon and access the OWL system.	
Precondition:	N/A	
Trigger:	This use case is initiated by the immediate need to access the system.	
	Actor Action	**System Response**
Typical Course of Events:	**Step 1:** Child and/or Parent accesses login screen. **Step 2:** Child and/or Parent enters login credentials including username and password.	**Step 3:** System authenticates credentials through secure ASP NET User database. **Step 4:** Access granted taken to main system page.
Alternate Courses:	**Alt-Step 2:** Child or Parent does not remember credentials. Taken to reset. **Alt-Step 4:** Access denied. Locked after six incorrect attempts.	
Conclusion:	This use case concludes when access to OWL system is granted to provider.	
Postcondition:	Full system access granted.	
Business Rules:	N/A	
Implementation Constraints and Specifications:	Child or Parent has full view privileges.	
Assumptions:	Credentials are valid.	
Open Issues:	N/A	



Offline While Learning (OWL) System		
Authors: Scott Yanoff, Julia Katz, Michael Lawrence, Vathsav Tipirneni		**Date: 11/15/2019** **Version: 1.0**
Use-Case Name:	Math Problems	**Use-Case Type Business Requirements:**
Use-Case ID:	OWL-2.00	
Priority:	High	
Source:	Child	
Primary Business Actor:	Child	
Other Participating Actors:		
Other Interested Stakeholders:		
Description:	This use-case describes how the user will be presented with math problems and how they will answer them	
Precondition:	N/A	
Trigger:	This use case is initiated by the user being presented with a math problem	
Typical Course of Events:	**Actor Action**	**System Response**
	Step 1: Child selects math operator they would like to practice **Step 3:** Child inputs an answer	**Step 2:** System receives input of selected math operator and retrieves problems from the Math problem database and displays it on the screen for the user. **Step 4:** System checks answer with answer in database and determines if the answer is right or wrong and displays result to user
Alternate Courses:	**Alt-Step 3:** Child does not answer question **Alt-Step 4:** Answer not received and score not registered for user	
Conclusion:	This use case concludes when all of the problems from the problem set have been answered	
Postcondition:	All questions answered and scores registered	
Business Rules:	N/A	
Implementation Constraints and Specifications:	**1.** The math problems will be displayed using a form format GUI. **2.** The GUI must have a logical flow for simple navigation and ease of use since the primary user will be children.	
Assumptions:	Questions are all included on DB and answers are valid	
Open Issues:	N/A	



Offline While Learning (OWL) System		
Authors: Scott Yanoff, Julia Katz, Michael Lawrence, Vathsav Tipirneni		**Date: 11/15/2019**
		Version: 1.0
Use-Case Name:	Keeping Track of User Progression	**Use-Case Type Business Requirements:**
Use-Case ID:	OWL-3.00	
Priority:	High	
Source:	Child and Parent	
Primary Business Actor:	Child amd Parent	
Other Participating Actors:		
Other Interested Stakeholders:		
Description:	This use-case describes how the user will access their scores and view progression	
Precondition:	N/A	
Trigger:	This use case is initiated by the user requesting to look at their scores and progression	
Typical Course of Events:	**Actor Action**	**System Response**
	Step 1: Child selects results from main menu **Step 3:** Child or Parent reviews scores	**Step 2:** System receives selection of results tab and retrieves score information from internal scores which is stored in the scores database
Alternate Courses:	**Alt-Step 2:** System does not retrieve any scores since no problem sets have been completed	
Conclusion:	This use case concludes when the user is able to review their scores	
Postcondition:	Score results are obtainable	
Business Rules:	N/A	
Implementation Constraints and Specifications:	**1.** The scores will be displayed using a form format GUI. **2.** The GUI must have a logical flow for simple navigation so the user can easily view their scores and determine if there is progress being made.	
Assumptions:	Child has completed at least 1 problem set and the score has been registered	
Open Issues:	N/A	

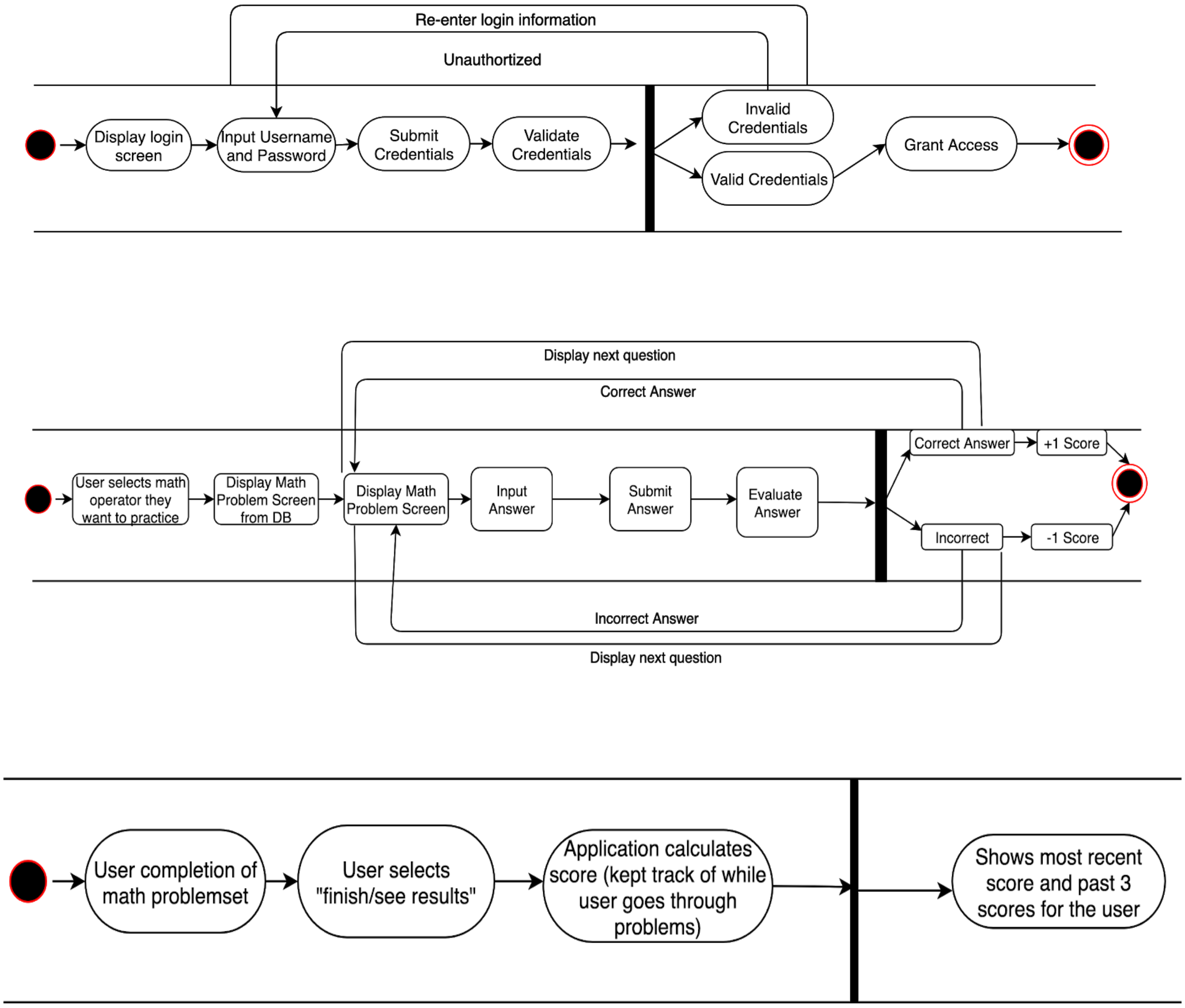
6.2.4. Event Table (Event Diagrams)

Look at the previous event table diagram from Section 5:

Actor/External Agent	Event(or Use Case)	Trigger	Responses
Parent/Child	Login/Registration	Creation of new login	Generate: New Profile and Create: New Login
Child	Going through problem set	Selection of math operand they wish to practice	Generate: Random problem from DB
Child/Parent	Reviewing Results	Completion of any problem set	Generate: Score Create: new score in profile of user

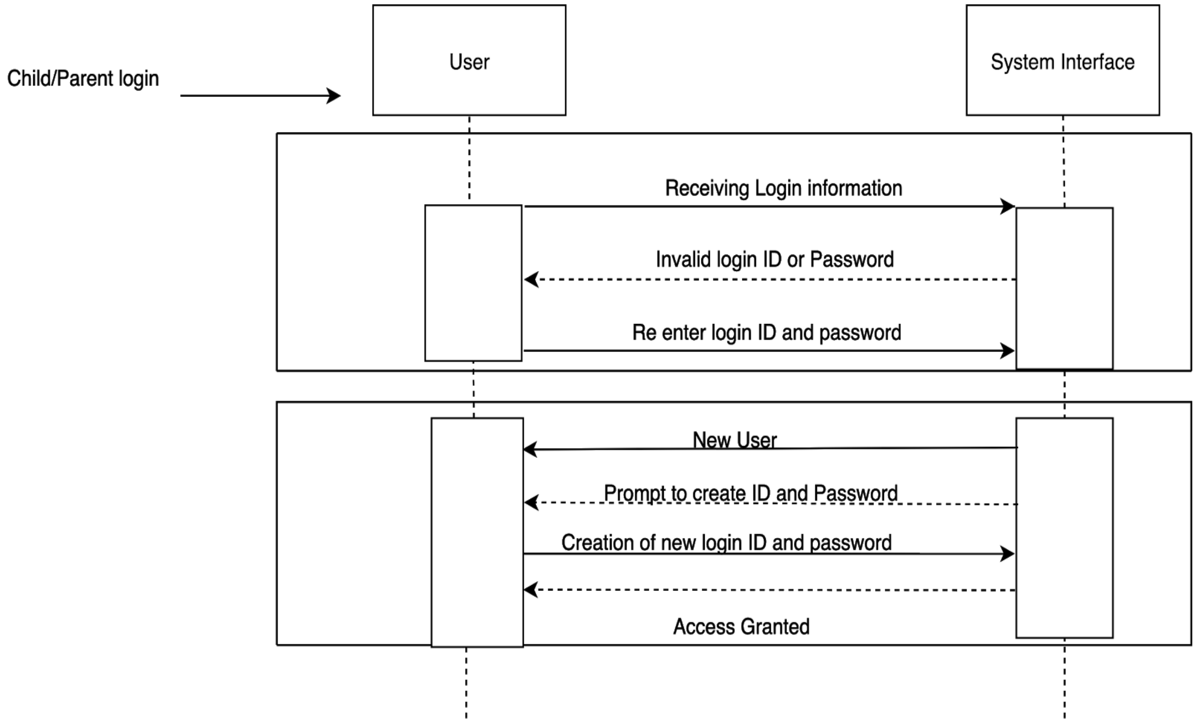
6.2.5. Activity Diagram

Flowchart of Events

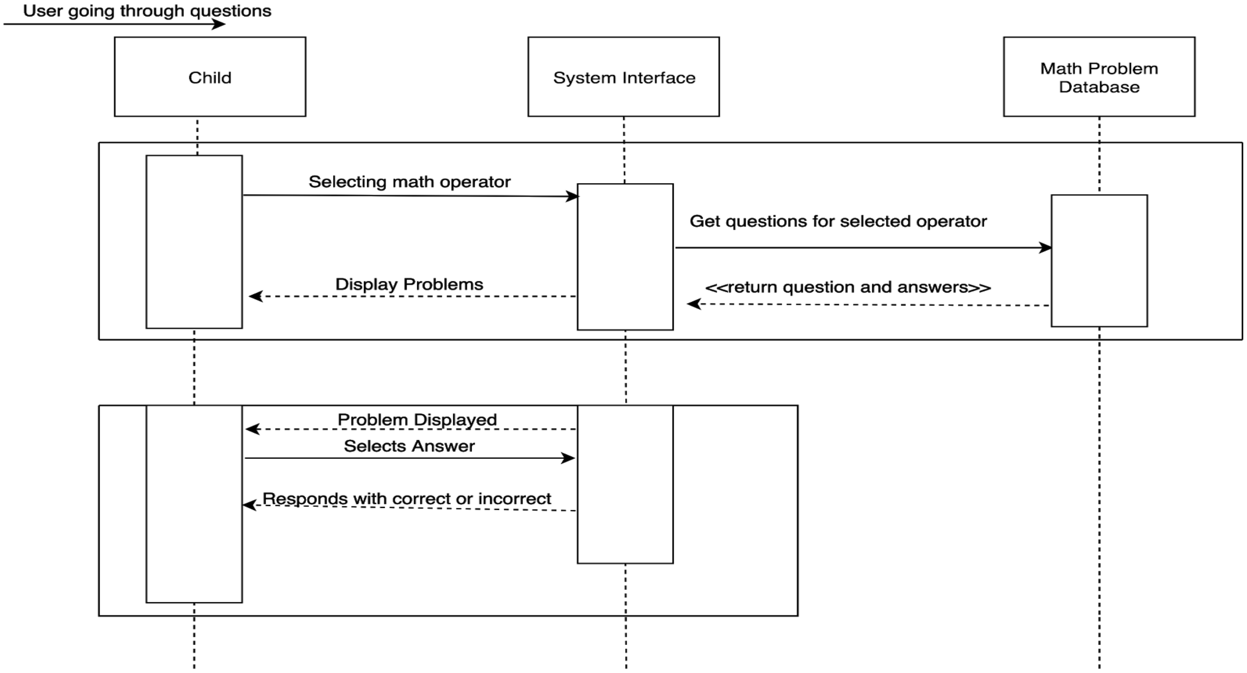

Re-enter login information
Unauthortized
Display login screen
Input Username and Password
Submit Credentials
Validate Credentials
Invalid Credentials
Valid Credentials
Grant Access
Display next question
Correct Answer
User selects math operator they want to practice
Display Math Problem Screen from DB
Display Math Problem Screen
Input Answer
Submit Answer
Evaluate Answer
Correct Answer
+1 Score
Incorrect
-1 Score
Incorrect Answer
Display next question
User completion of math problemset
User selects "finish/see results"
Application calculates score (kept track of while user goes through problems)
Shows most recent score and past 3 scores for the user

6.2.6. Systems Sequence Diagrams for OWL

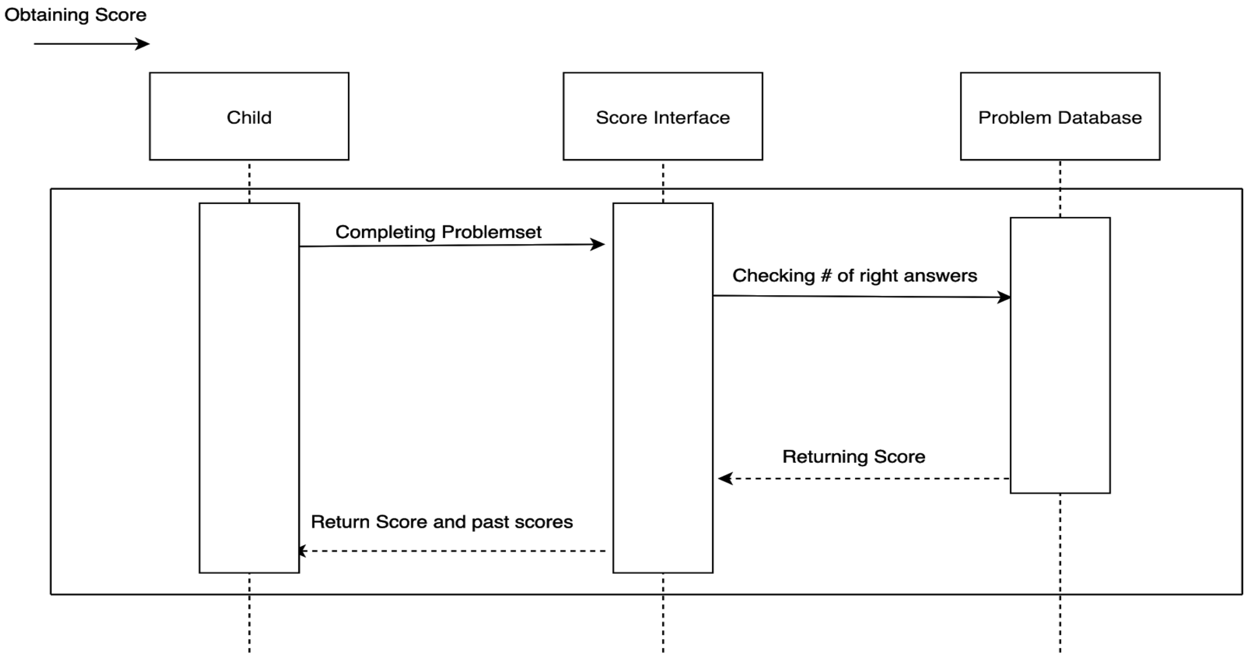
OWL Creating Login Accessing System


Child/Parent login
User
System Interface
Receiving Login information
Invalid login ID or Password
Re enter login ID and password
New User
Prompt to create ID and Password
Creation of new login ID and password
Access Granted

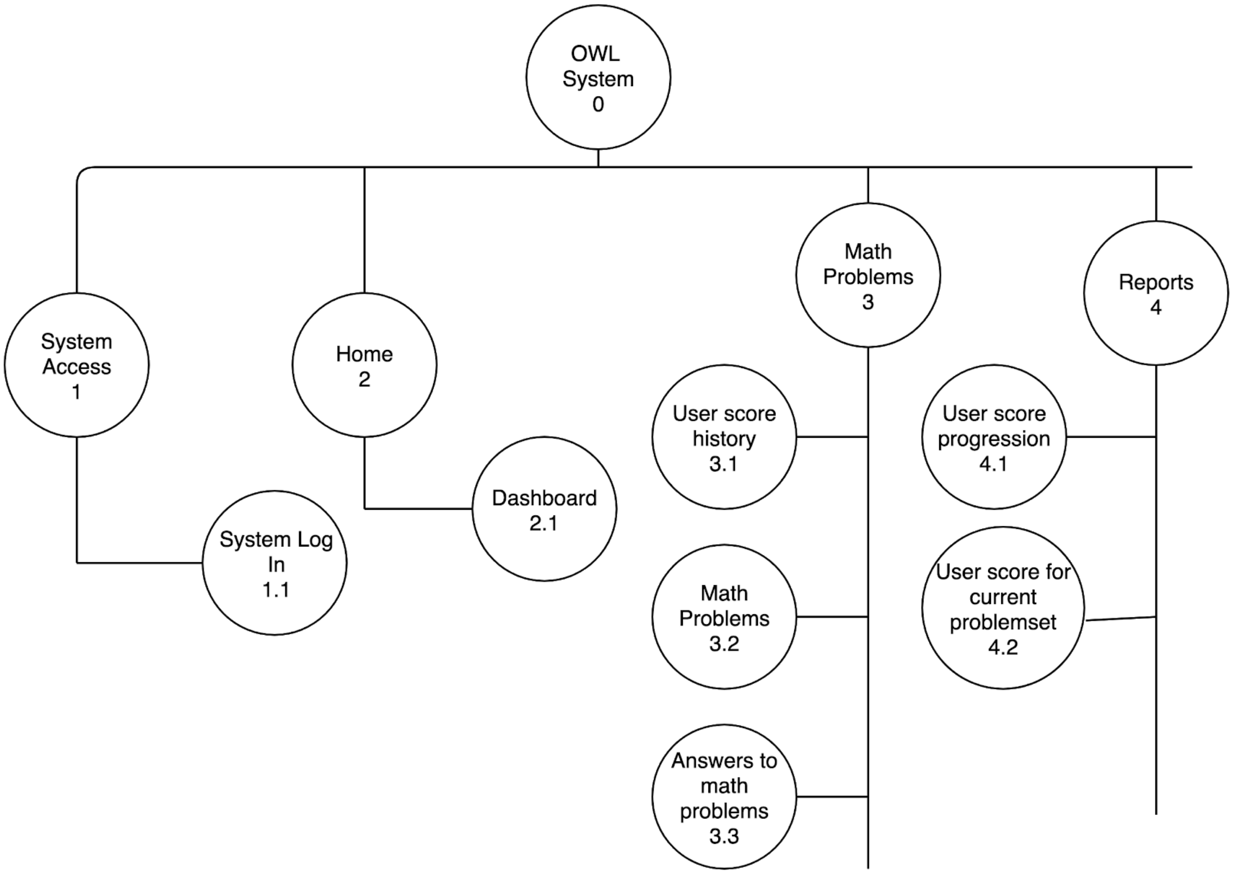
Child Going Through Math Problems


User going through questions
Child
System Interface
Math Problem Database
Selecting math operator
Get questions for selected operator
Display Problems
<<return question and answers>>
Problem Displayed
Selects Answer
Responds with correct or incorrect

Obtaining Score From Application

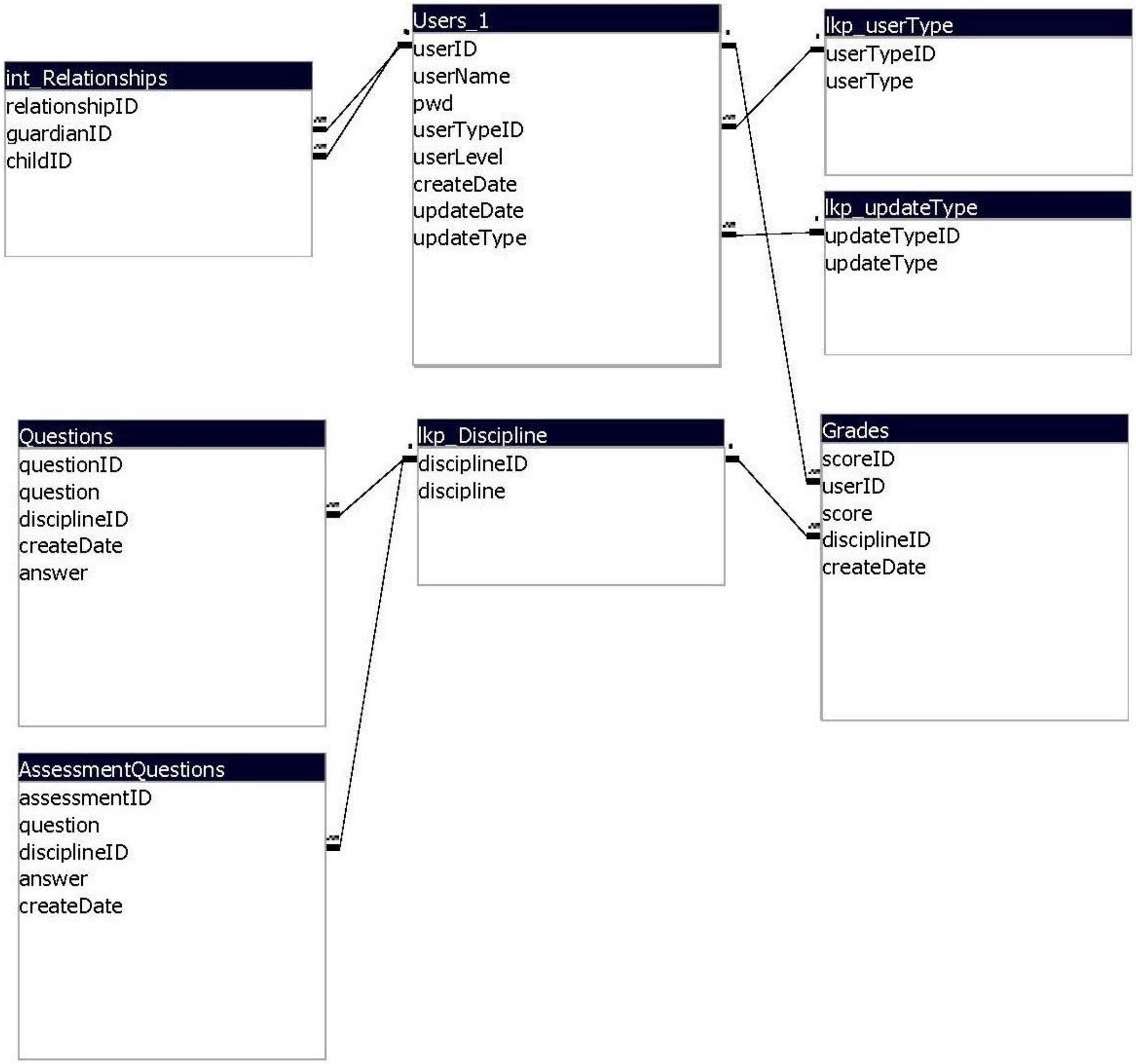


6.2.7. Software Structure Chart For Module Hierarchy

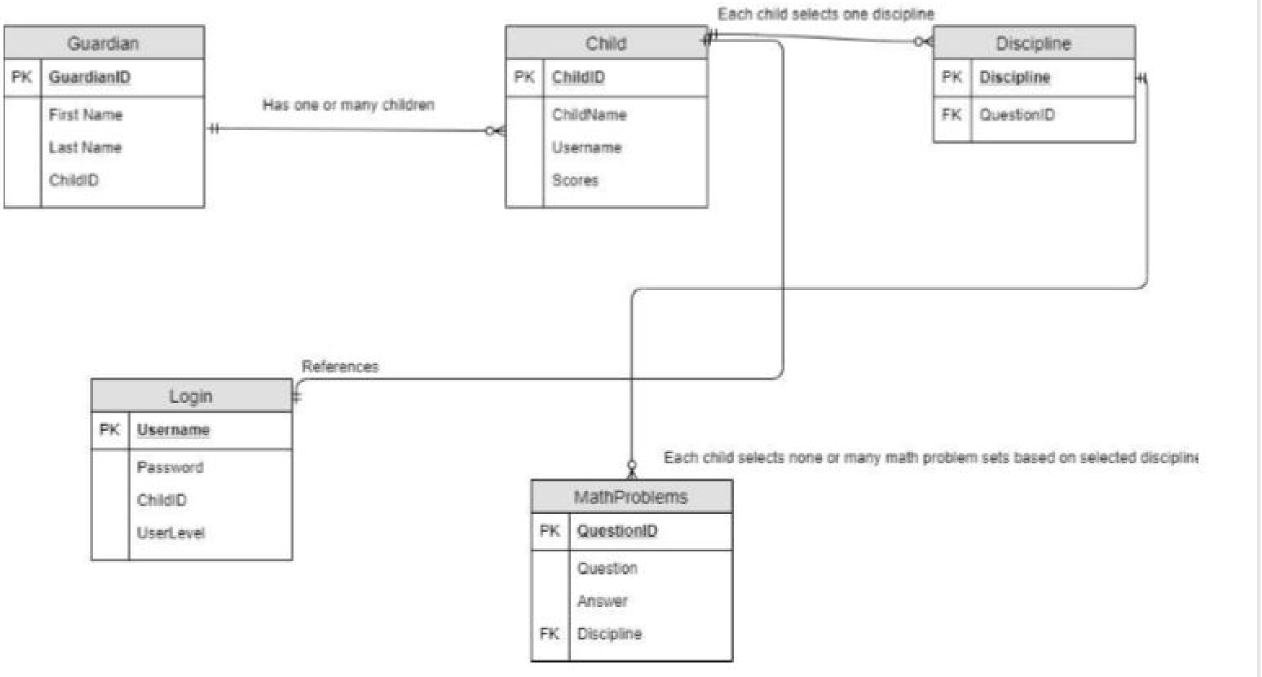


6.3. Design the System Database

6.3.1 Construct the Detailed Entity Relationship (ER) Diagram



6.3.2 Perform the 3 Normalization Forms, Listing Dependencies



6.3.3 Chart Entities, Attributes, Primary Keys and Foreign Keys

Chart Entities
USERS lkp_userType lkp_updateType int_Relationships QUESTIONS ASSESSMENTQUESTIONS lkp_Discipline GRADES

Attributes	
USERS:	userID, userName, pwd, userTypeID, userLevel, createDate, updateDate, updateType
lkp_userType:	userTypeID, userType
lkp_updateType:	updateTypeID, updateType
=int_Relationships:	relationshipID, guardianID, childID
QUESTIONS:	questionID, question, disciplineID, createDate, answer
ASSESSMENTQUESTIONS:	assessmentID, question, disciplineID, createDate, answer
lkp_Discipline:	disciplineID, discipline
GRADES:	scoreID, userID, score, disciplineID, createDate

Primary Keys	
USERS:	userID
lkp_userType:	userTypeID
lkp_updateType:	updateTypeID
int_Relationships:	relationshipID
QUESTIONS:	questionID
ASSESSMENTQUESTIONS:	assessmentID
lkp_Discipline:	disciplineID
GRADES:	scoreID

Foreign Keys	
USERS:	userTypeID, updateType
lkp_userType:	NULL
lkp_updateType:	NULL
int_Relationships:	guardianID, childID
QUESTIONS:	disciplineID
ASSESSMENTQUESTIONS:	disciplineID
lkp_Discipline:	NULL
GRADES:	userID, disciplineID

6.3.4 Data Dictionary for all the Attributes for OWL

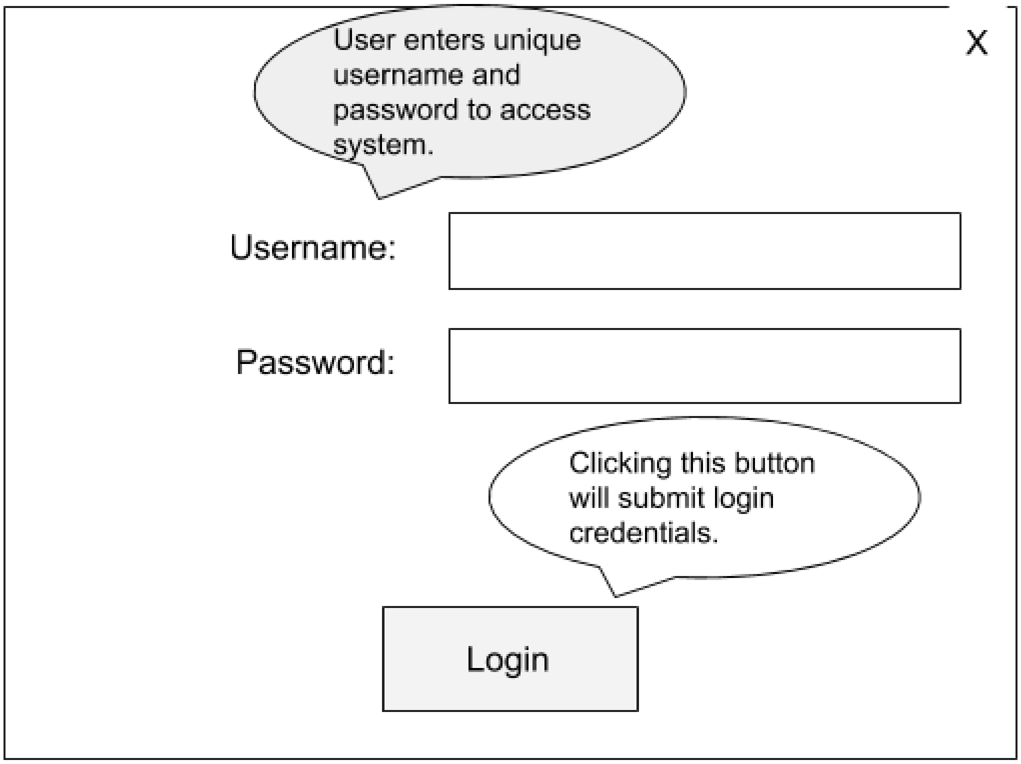
Table	Attributes	Data Type	Primary Key	Allow Null	Description
USERS	userID	int()	Yes	No	Unique user ID
	userName	varchar(10)	No	No	User's username
	pwd	varchar(10)	No	No	User's password
	userTypeID	int( )	No	No	User's type descriptor
	userLevel	int()	No	No	User's level
	createDate	Date	No	No	Date record was created
	updateDate	Date	No	No	Date record was updated
	updateType	varchar(10)	No	No	Record update type descriptor
lkp_userType	userTypeID	int( )	Yes	No	Unique usertype ID
	userType	varchar(10)	No	No	User type descriptor
lkp_updateType	updateTypeID	int( )	Yes	No	Unique update type ID
	updateType	varchar(10)	No	No	Update type descriptor
int_Relationships	relationshipID	int( )	Yes	No	Unique relationship ID
	guardianID	int( )	No	No	Unique guardian ID
	childID	int( )	No	No	Unique child ID
QUESTIONS	questionID	int( )	Yes	No	Unique question ID
	question	varchar(10)	No	No	Math questions
	disciplineID	int( )	No	No	Unique discipline ID/descriptor
	createDate		No	No	Date record was created
	answer	varchar(10)	No	No	Math answers
ASSESSMENT QUESTIONS	assessmentID	int( )	Yes	No	Unique assessment ID
	question	varchar(10)	No	No	Math questions

	disciplineID	int( )	No	No	Unique discipline ID/descriptor
	answer	varchar(10)	No	No	Math answers
	createDate	Date	No	No	Date record was created
lkp_Discipline	disciplineID	int( )	Yes	No	Unique discipline ID
	discipline	varchar(10)	No	No	Unique discipline ID/descriptor
GRADES	scoreID	int( )	Yes	No	Unique score ID
	userID	int( )	No	No	Unique user ID
	score	int( )	No	No	Final score for the exam
	disciplineID	int( )	No	No	Unique discipline ID/descriptor
	createDate	Date	No	No	Date record was created

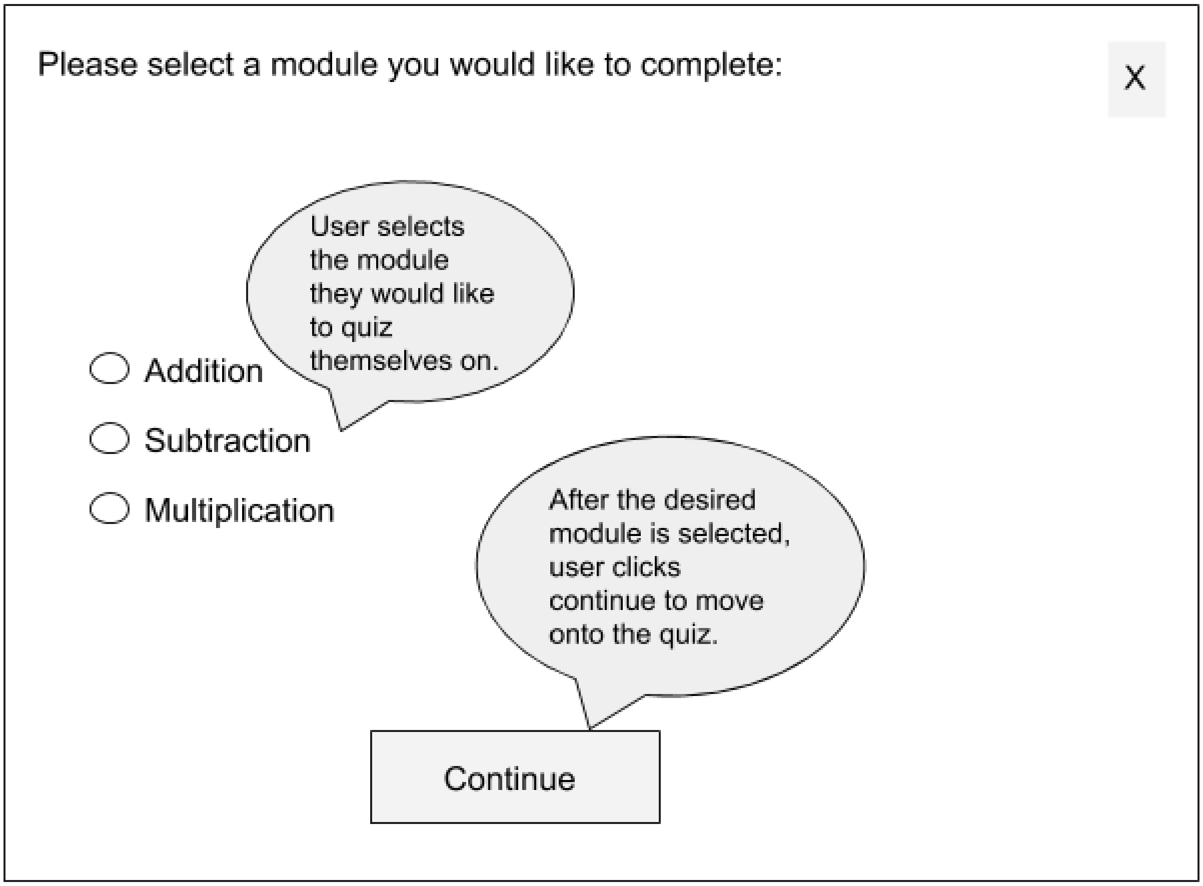
6.4. System Interface Design

6.4.1 - 6.4.11

Module Name	**Login**
Parameters Passed	Username & password, both of which are unique for each user
Description of Module	Used to gain access to OWL system
Input	Click “Login”
Output	Access to OWL gained if credentials are valid. Access denied if invalid credentials.
Called Module	Home
Story	Once credentials are submitted, user will or will not gain access to OWL

 User enters unique username and password to access system. Clicking this button will submit login credentials. 	
Error	If invalid credentials, access denied

Module Name	**Home**
Parameters Passed	Math Module Selection
Description of Module	User can select which math module they wish to complete (addition, subtraction, multiplication)
Input	User checks which module they will do, then clicks “Continue”
Output	Corresponding module is called
Called Module	User dependent (addition, subtraction, multiplication)
Story	From this page, users will be able to select which math module they wish to be quizzed on. Once a module has been selected, clicking the “Continue” button will bring the user to the module that they had selected.



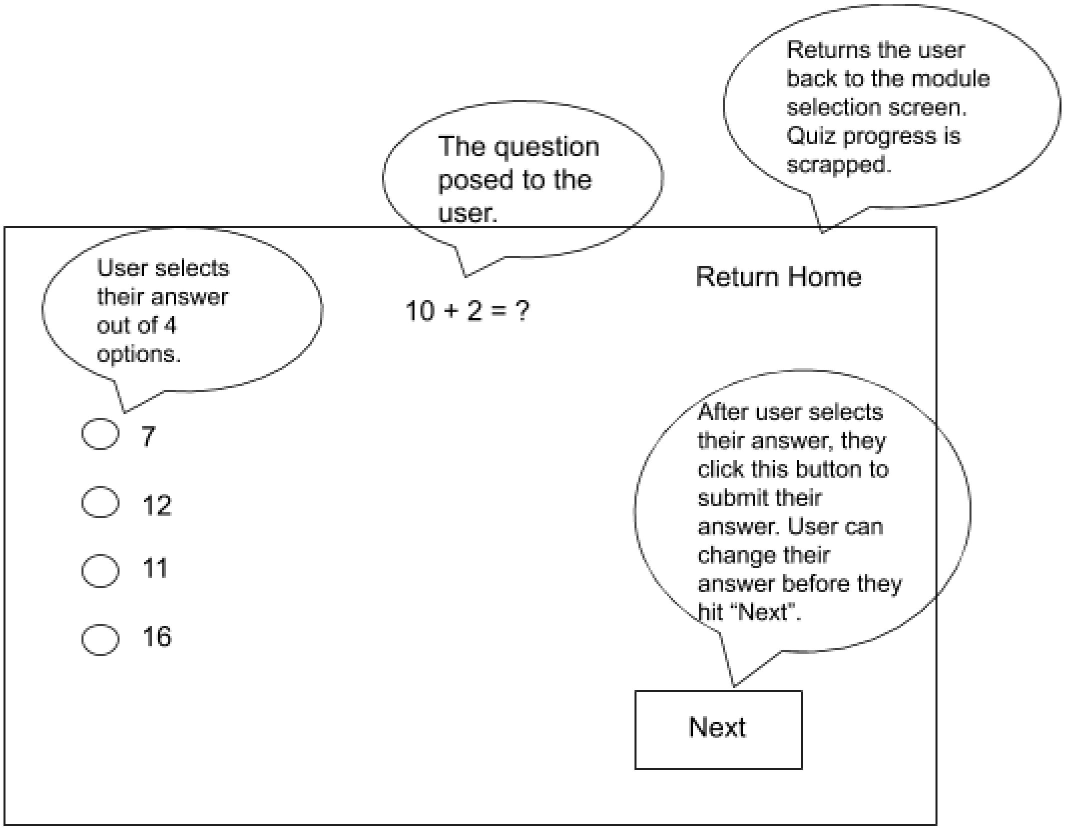
Please select a module you would like to complete:

User selects the module they would like to quiz themselves on.

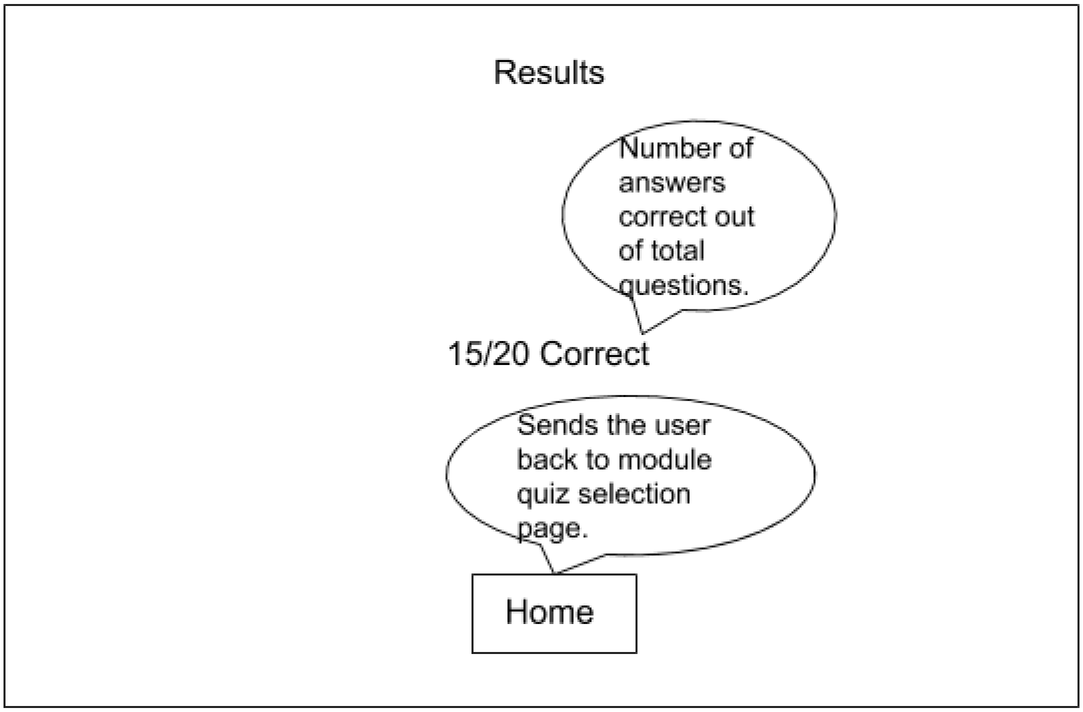
After the desired module is selected, user clicks continue to move onto the quiz.

Error	If no module has been selected and the user clicks continue: “Error: no module selected”.

Module Name	**Quiz (Addition, Subtraction, and Multiplication quizzes all share the same interface)**
Parameters Passed	Score
Description of Module	Quiz that user will take
Input	User selects answer from the four multiple choice answers provided, then clicks “Next” User can also click “Return Home” to go back to the front page.
Output	Points earned if correct answer selected; no points gained if wrong
Called Module	Results/Home if “Return Home” clicked

Story	User is quizzed on different math problems depending on what module the user selected. The Addition quiz will only have addition problems, Subtraction only subtraction problems, etc. The quizzes are all multiple choice, with the user having to select an answer, then click “Next” to proceed to the next math problem. The correct answer gives the user a point, while the wrong answer gives zero points. At the end of the quiz, the user will be taken to a results screen. The user can also click “Return Home” to go back to the homepage. Quiz will not be scored.
 Returns the user back to the module selection screen. Quiz progress is scrapped. The question posed to the user. User selects their answer out of 4 options. Return Home 10 + 2 = ? ○ 7 ○ 12 ○ 11 ○ 16 After user selects their answer, they click this button to submit their answer. User can change their answer before they hit “Next”. Next	
Error	If the user does not select an answer to a problem and clicks “Next”, the quiz will count that as a wrong answer.

Module Name	**Results**
Parameters Passed	None
Description of Module	User’s results of completed quiz will be shown here

Input	User clicks “Home” to go back to homepage
Output	None
Called Module	Home
Story	Users will be taken to this module upon completion of the quiz’s final question. Here the user can look at their score before clicking on the Home button to return to the homepage
 Results Number of answers correct out of total questions. 15/20 Correct Sends the user back to module quiz selection page. Home	
Error	None

7. Construction Phase

7.1. Build and Test Networks (intranet and/or internet), if necessary.

- No networks are necessary (neither intranet nor internet) other than users already established a home or work internet network. They only need the network to visit a website for the initial software downloads and subsequent security/update files.

7.2. Build and Test Databases

- The database was built in Microsoft Access and includes tables that are required to support the application. The database is fully incorporated into the application software and can be written to, read from and data updated within the local file.

7.2.1. Complete the Database Alpha Testing

- The Alpha Testing of the OWL Database, was successful within the project. The questions are able to be pulled from the database itself into the Visual Basic Studios program.

7.3. Install and Test New Software Packages

- The final project solution file was exported as an installation package and installed on test computers. The software works as expected.

7.4. Write and Test New Programs

7.4.1. Complete the Alpha Testing

- Alpha testing was completed and all of the acceptance criteria has been met.

7.5. Schedule - Use the Start and End Dates of this Academic Year

Class Date	Objective to complete
01/14	**Project Documentation and Design final approval** • The team will **submit & present** the project documentation and design from CS620. • General description of the project (where main ideas and steps of the solution are clearly presented) • Software and Hardware requirements
01/21	**Development of unit tests and pseudo code for modules and classes**
01/28	**Development of usability testing** • Develop code
02/04	**Code and Test**

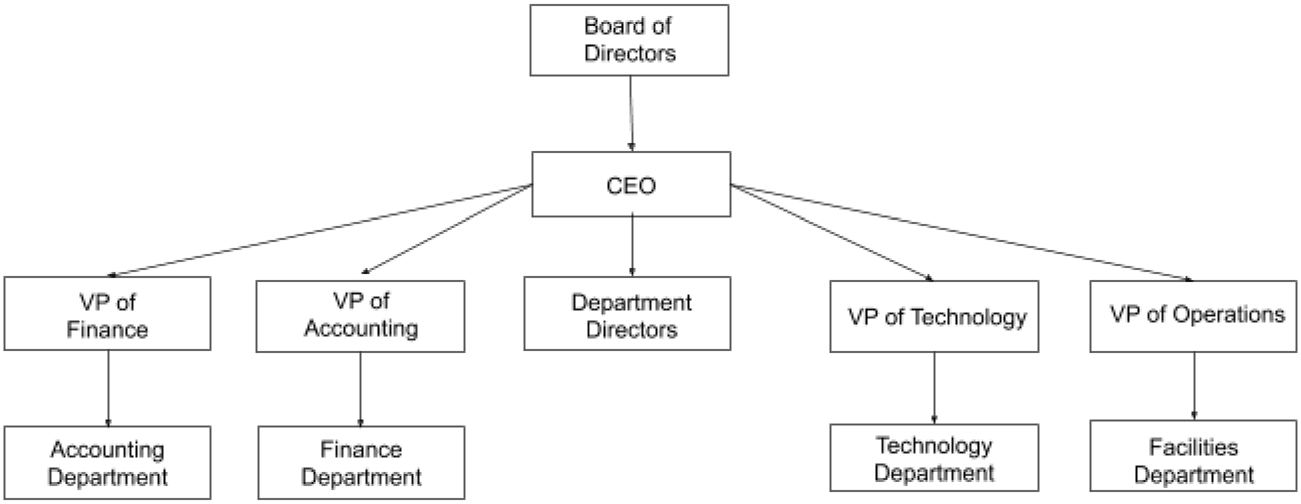
02/11	**Code and Test.** • Results and Evaluations (Make sure to analyze test results) • Usability Testing and Analysis. • Project Code: submitted on a USB
02/18	**Code and Test**
02/25	**Code and Test**
03/03	**Code and Test**
03/10	**Design Documents Final Approval** • You will present documents showing your project development for approval. These documents might include page wireframes (interfaces), code (commented & un-commented), reports layouts, etc. again depending on the needs of each project. • To be posted on blackboard a minimum of three (3) business days prior to your final project presentation.
03/17	**Observations and Conclusions** Review of revised documentation • Data dictionary • Alphabetical Index • Bibliography / References • Glossary • Contacts
03/24	**Final Project Presentation.** Your presentation should be given as if you were presenting your project to an investor or an employer. Make it professional and interesting.

7.6. Staffing - Show Owners Responsibility

The owners of our OWL inc. are all experienced professionals hailing from both the technology and educational industries. Because this is a new company founded within the last 3 months, all of the founders are owners. As owners each of us are responsible for ensuring the quality and functionality of the product.

7.6.1. Staffing Hierarchy Chart

For the purposes of this project, the hierarchy chart will consist of the 5 members of our group, each operating as a co-owner of the company. Ideally, once we get our operations off the ground and have the product ready for launch, we will have a hierarchy chart that looks similar to the following hierarchy chart.



7.7. Acceptance Testing

7.7.1. Plan

For our systems acceptance test, we will go through 3 stages; verification testing, validation testing, and lastly, audit testing. The specifics of each of these tests will be explained in further detail in the following sections.

7.7.2. Schedule

Our plan for the Systems Acceptance test is as follows:

By 2/4/2020, we will begin the verification testing, using a short amount of questions for each math discipline. This will be done to ensure that the program is on the right development track.

By 2/24/2020, we will begin the validation testing, in a live environ, using all of the questions that we plan on using, for the finalized product. During this testing we will look for a variety of things:

- Systems Performance: We will check the throughput and response time of the program to make sure that the program works well, and that it does not overload the computer and has an efficient response time after each event.
- Human Trial: We are going to be checking that the program is easy to use, and that the user understands what they are doing at every step.
- Methods and Procedures Test: This is to check that all of the methods employed in the program, work effectively and there are no logical errors in our code. The methods in question for our project, will be the method used for cycling through

the questions in the database and then calculating the final scores after each quiz and assessment.

By 3/10/2020, we will begin the audit testing portion of the systems acceptance test, where we will have our finalized program and will check to make sure that any errors previously identified, in the other testing stages, were addressed.

7.7.3. Acceptance Criteria

- Login form allows users to log in with their personalized usernames and passwords that they created during registration.
- Users can select a math discipline, and the application will pull the problems, for that discipline, from the database that we created to store the problems.
- Once the user enters an answer for the question, the program takes the answer and compares it against the answer stored in the database.
- Displays via text to the user, whether or not the answer entered by the user, was correct or incorrect.

7.7.3.1. Complete the Beta Testing

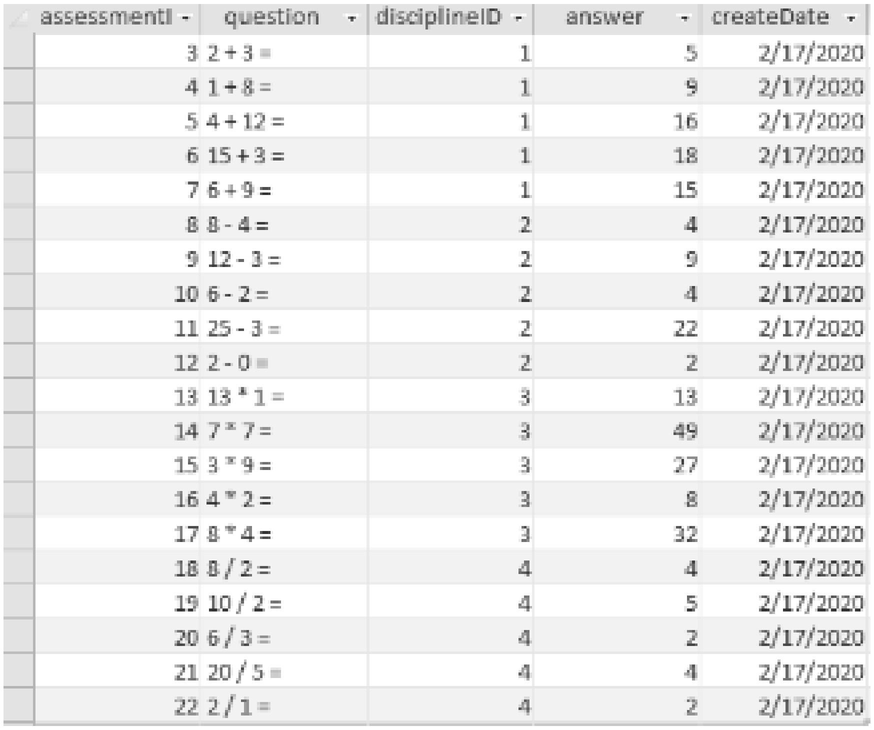
- Beta testing was completed and met all requirements as outlined in our usability analysis document (Appendix B).

7.8. Tools to be Used in the Proposed System

7.8.1. Database

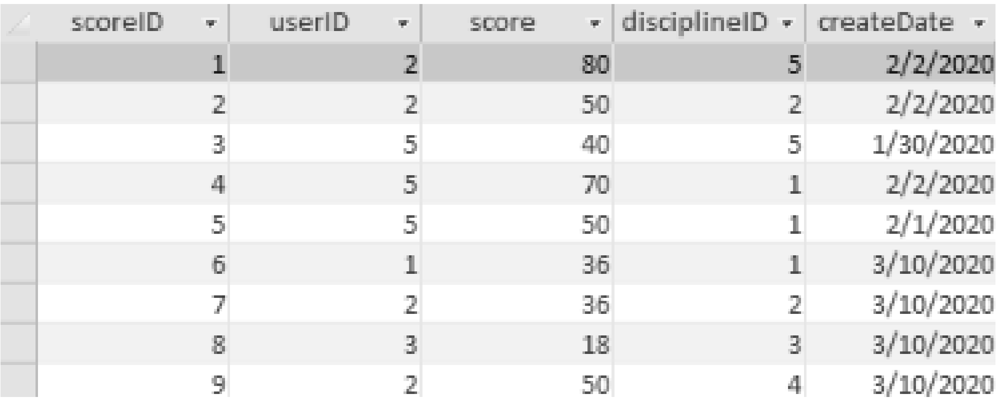
Screenshots of our Access database that will store all of the information for our application:

Assessment Questions Table



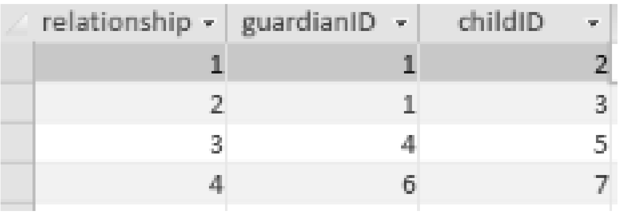
assessmentI	question	disciplineID	answer	createDate
3	2 + 3 =	1	5	2/17/2020
4	1 + 8 =	1	9	2/17/2020
5	4 + 12 =	1	16	2/17/2020
6	15 + 3 =	1	18	2/17/2020
7	6 + 9 =	1	15	2/17/2020
8	8 - 4 =	2	4	2/17/2020
9	12 - 3 =	2	9	2/17/2020
10	6 - 2 =	2	4	2/17/2020
11	25 - 3 =	2	22	2/17/2020
12	2 - 0 =	2	2	2/17/2020
13	13 * 1 =	3	13	2/17/2020
14	7 * 7 =	3	49	2/17/2020
15	3 * 9 =	3	27	2/17/2020
16	4 * 2 =	3	8	2/17/2020
17	8 * 4 =	3	32	2/17/2020
18	8 / 2 =	4	4	2/17/2020
19	10 / 2 =	4	5	2/17/2020
20	6 / 3 =	4	2	2/17/2020
21	20 / 5 =	4	4	2/17/2020
22	2 / 1 =	4	2	2/17/2020

Grades Table



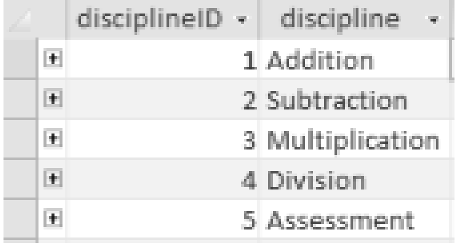
scoreID	userID	score	disciplineID	createDate
1	2	80	5	2/2/2020
2	2	50	2	2/2/2020
3	5	40	5	1/30/2020
4	5	70	1	2/2/2020
5	5	50	1	2/1/2020
6	1	36	1	3/10/2020
7	2	36	2	3/10/2020
8	3	18	3	3/10/2020
9	2	50	4	3/10/2020

Int_Relationships Table



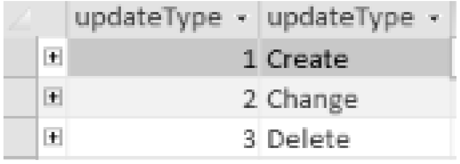
relationship	guardianID	childID
1	1	2
2	1	3
3	4	5
4	6	7

lkp_Discipline Table



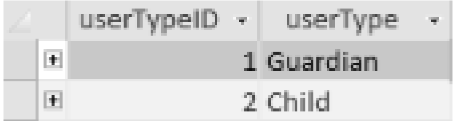
disciplineID	discipline
1	Addition
2	Subtraction
3	Multiplication
4	Division
5	Assessment

lkp_updateType Table



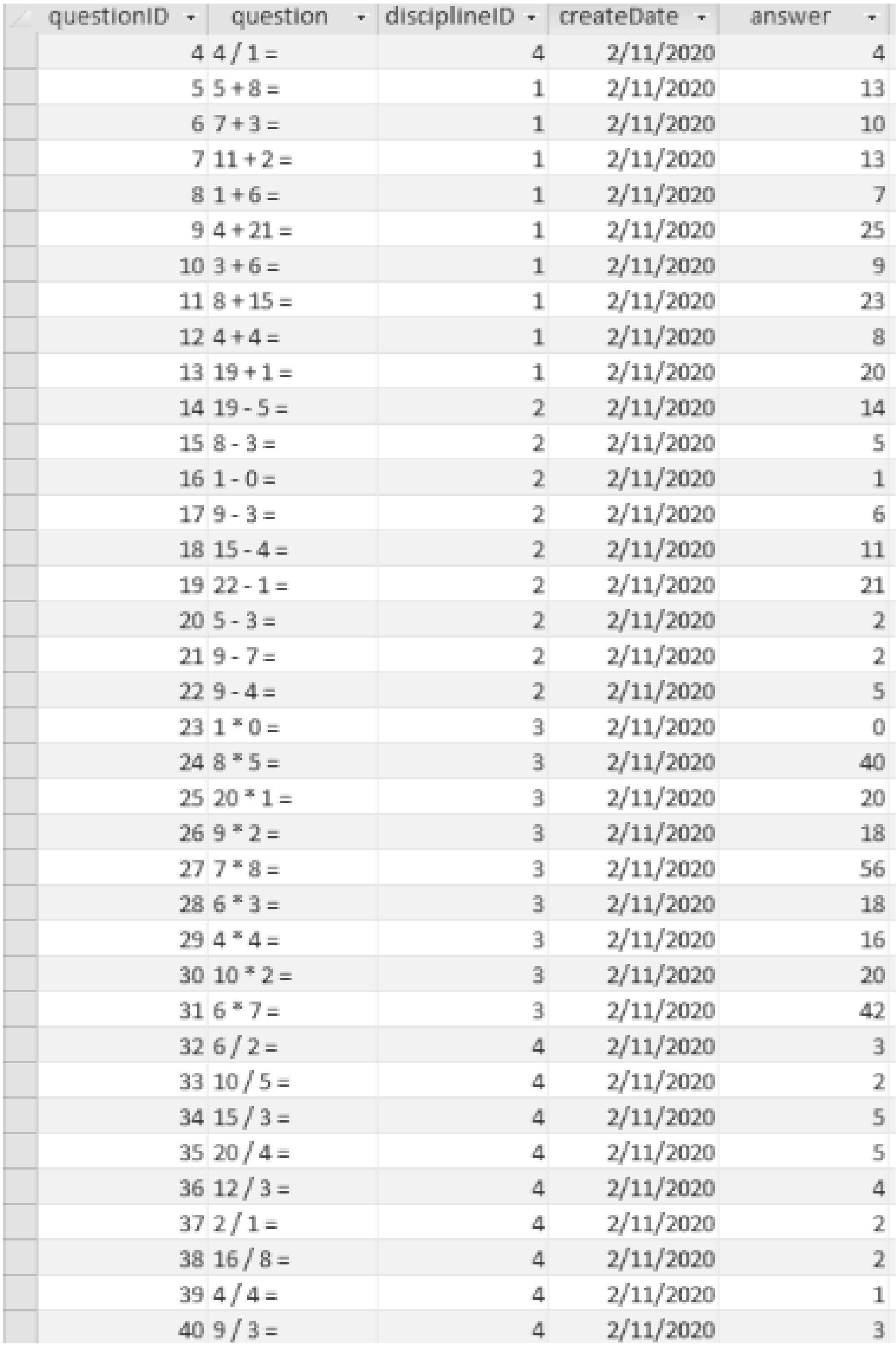
updateType	updateType
1	Create
2	Change
3	Delete

lkp_userType Table



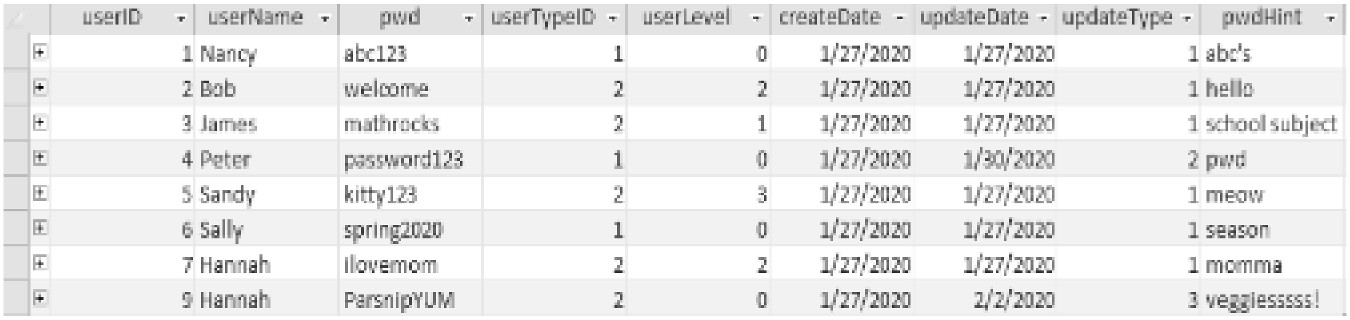
userTypeID	userType
1	Guardian
2	Child

Questions Table



questionID	question	disciplineID	createDate	answer
4	4 / 1 =	4	2/11/2020	4
5	5 + 8 =	1	2/11/2020	13
6	7 + 3 =	1	2/11/2020	10
7	11 + 2 =	1	2/11/2020	13
8	1 + 6 =	1	2/11/2020	7
9	4 + 21 =	1	2/11/2020	25
10	3 + 6 =	1	2/11/2020	9
11	8 + 15 =	1	2/11/2020	23
12	4 + 4 =	1	2/11/2020	8
13	19 + 1 =	1	2/11/2020	20
14	19 - 5 =	2	2/11/2020	14
15	8 - 3 =	2	2/11/2020	5
16	1 - 0 =	2	2/11/2020	1
17	9 - 3 =	2	2/11/2020	6
18	15 - 4 =	2	2/11/2020	11
19	22 - 1 =	2	2/11/2020	21
20	5 - 3 =	2	2/11/2020	2
21	9 - 7 =	2	2/11/2020	2
22	9 - 4 =	2	2/11/2020	5
23	1 * 0 =	3	2/11/2020	0
24	8 * 5 =	3	2/11/2020	40
25	20 * 1 =	3	2/11/2020	20
26	9 * 2 =	3	2/11/2020	18
27	7 * 8 =	3	2/11/2020	56
28	6 * 3 =	3	2/11/2020	18
29	4 * 4 =	3	2/11/2020	16
30	10 * 2 =	3	2/11/2020	20
31	6 * 7 =	3	2/11/2020	42
32	6 / 2 =	4	2/11/2020	3
33	10 / 5 =	4	2/11/2020	2
34	15 / 3 =	4	2/11/2020	5
35	20 / 4 =	4	2/11/2020	5
36	12 / 3 =	4	2/11/2020	4
37	2 / 1 =	4	2/11/2020	2
38	16 / 8 =	4	2/11/2020	2
39	4 / 4 =	4	2/11/2020	1
40	9 / 3 =	4	2/11/2020	3

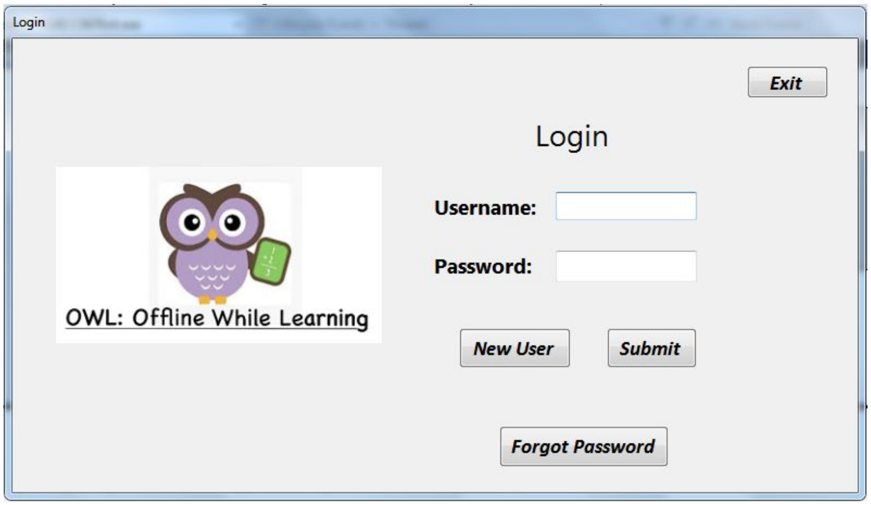
Users Table



userID	userName	pwd	userTypeID	userLevel	createDate	updateDate	updateType	pwdHint
1	Nancy	abc123	1	0	1/27/2020	1/27/2020	1	abc's
2	Bob	welcome	2	2	1/27/2020	1/27/2020	1	hello
3	James	mathrocks	2	1	1/27/2020	1/27/2020	1	school subject
4	Peter	password123	1	0	1/27/2020	1/30/2020	2	pwd
5	Sandy	kitty123	2	3	1/27/2020	1/27/2020	1	meow
6	Sally	spring2020	1	0	1/27/2020	1/27/2020	1	season
7	Hannah	ilovemom	2	2	1/27/2020	1/27/2020	1	momma
9	Hannah	ParsnipYUM	2	0	1/27/2020	2/2/2020	3	veggiesssss!

7.8.2. Screen Generators (Screen shots, save for once we have finalized project)
Screenshots from actual project:

Login Form

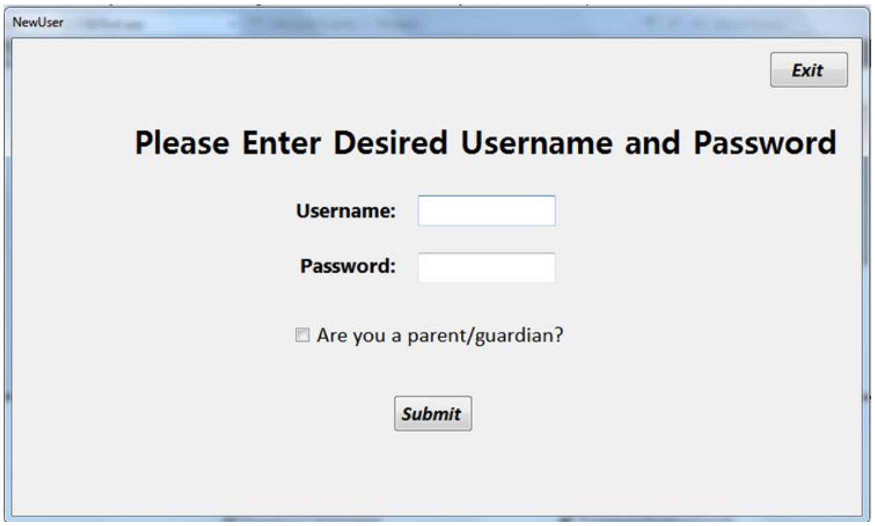


Login

Username:

Password:

New User Sign Up Form



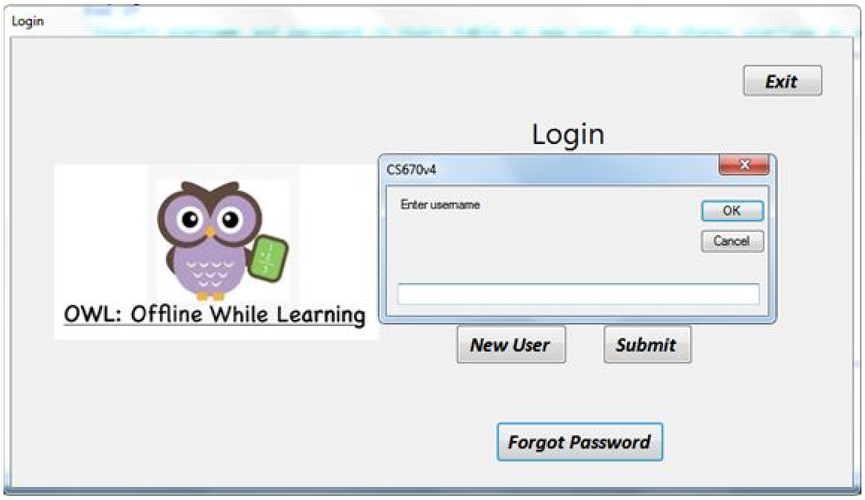
Please Enter Desired Username and Password

Username:

Password:

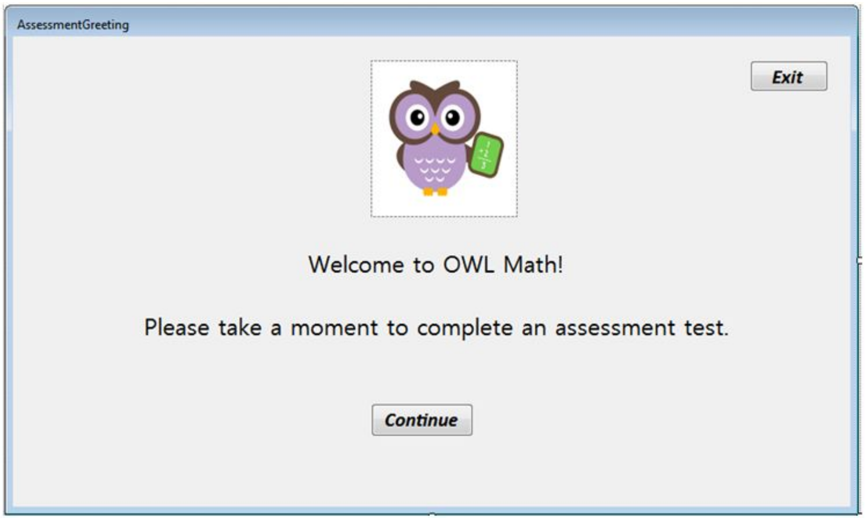
□ Are you a parent/guardian?

Forgotten Password



Enter username

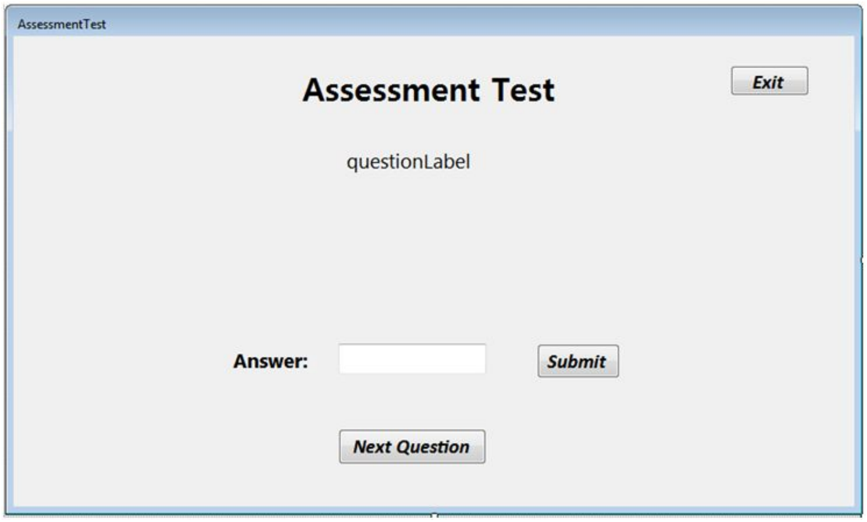
Assessment Greeting Form



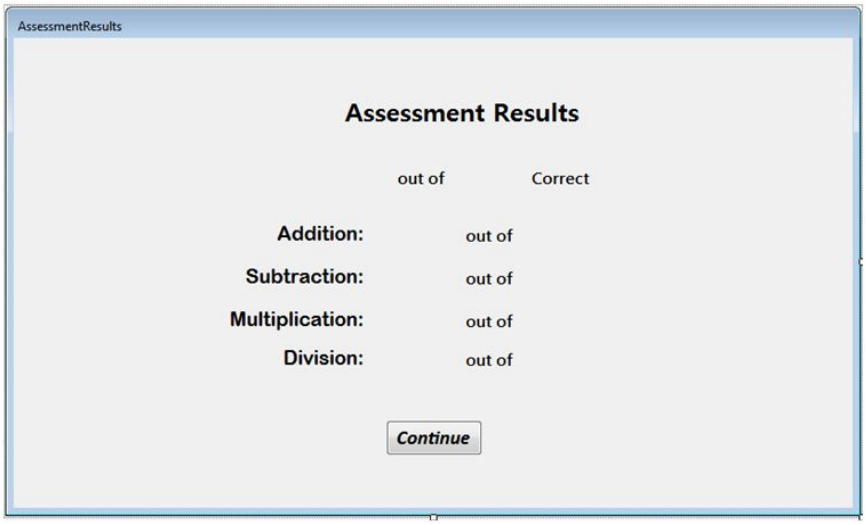
Welcome to OWL Math!

Please take a moment to complete an assessment test.

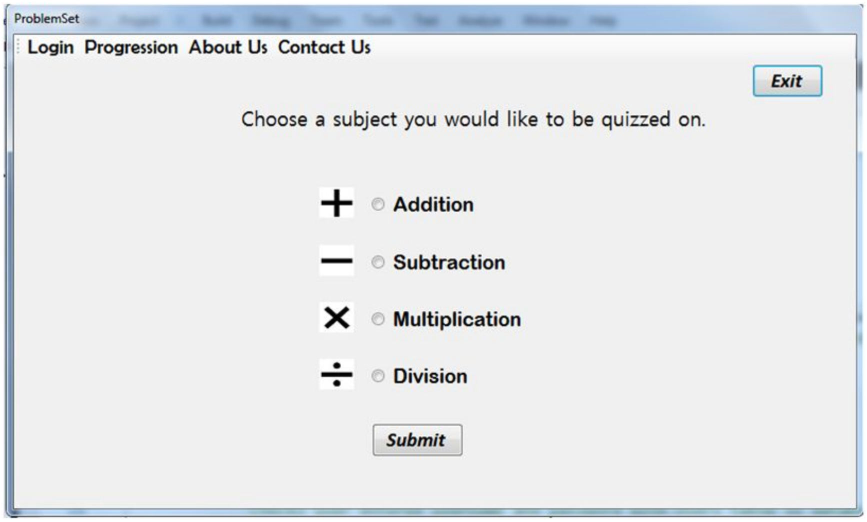
Assessment Test Question Form



Assessment Test Results Form

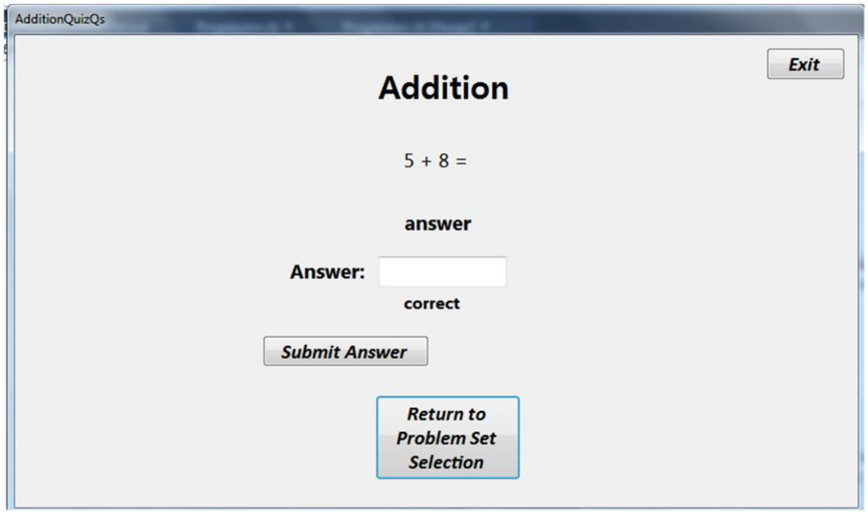


Problem Set Selection Form

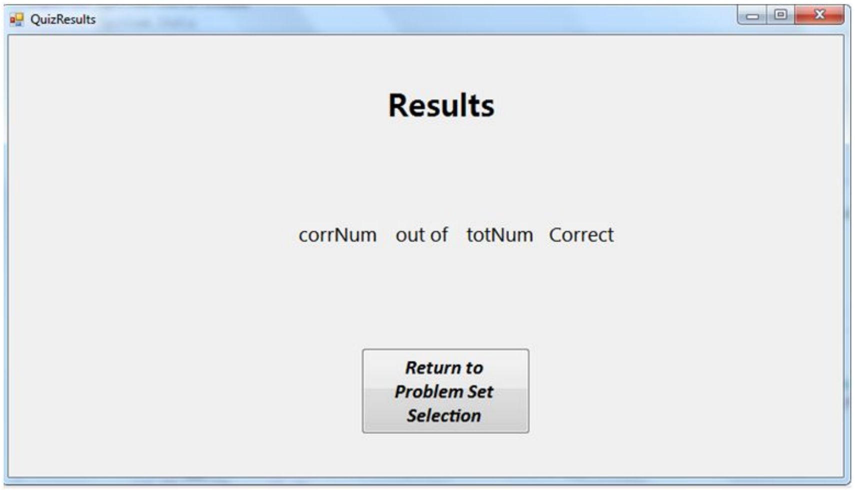


ProblemSet
Login Progression About Us Contact Us
Exit
Choose a subject you would like to be quizzed on.
+ ○ Addition
− ○ Subtraction
× ○ Multiplication
÷ ○ Division
Submit

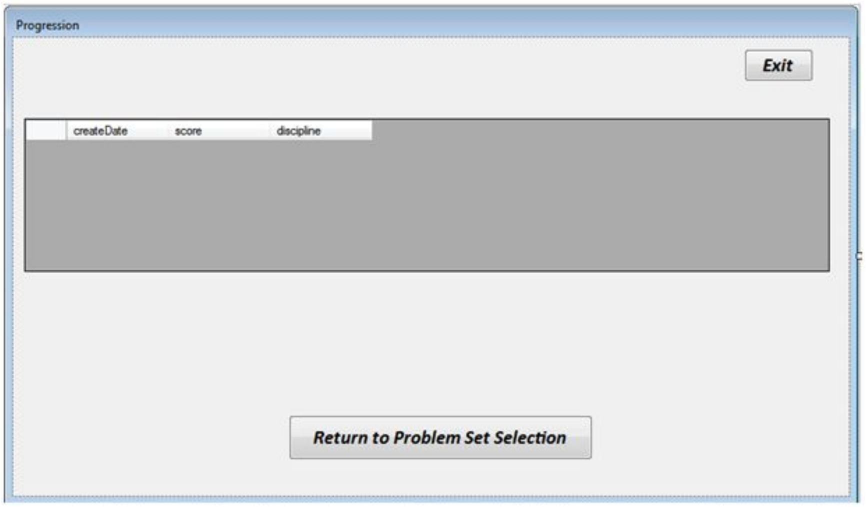
Quiz Question Form



Quiz Results Form


QuizResults
Results
corrNum out of totNum Correct
Return to Problem Set Selection

Progression Form



7.8.3. Hardware - servers, etc.

- The only hardware required to both develop and maintain this application is a working computer. While an internet connection is convenient to have, it is not necessary.

7.9. Maintenance Documentation

7.9.1. List of Modules

- Binder of materials presented to show how the project is supposed to work
- Comments in the actual project itself

7.9.2. Auditing Procedures

- There will not be much auditing to do for our company and project, but we will have an internal log in audit, to see when development team members are working on the program.

- There will also be password audits to make sure that every team member has secure passwords.

7.9.3. Backup and Restore Procedures

- Database and program files will be encrypted and backed up onto an external hard drive, in case there is an emergency and all of the information is lost.
- As for restoration, we will simply pull the files from the hard drive and upload them to our development team's computers.

7.9.4. List of Maintenance Contacts With Email and Phone Numbers

All maintenance will be done by the 5 owners of the company.

Names	E-mail	Contact Number
Scott Yanoff	scott.a.yanoff@gmail.com	203-451-8644
Julia Katz	jj7jj7@gmail.com	203-273-8807
Michael Lawrence	michaellawrence193@gmail.com	804-517-0998
Vathsav Tipirneni	vathsavtipirneni97@gmail.com	475-231-9054
Meghan Polis	meghan.polis@gmail.com	631-278-3117

7.9.5. List of Vendors, Consultants and External Contacts

There are no outside vendors, consultants, or external contacts working on this project. The only people that we have working on this project are the 5 owners who are also the co-founders.

8. Implementation Phase

8.1. Conduct System Test

We conducted the systems test and everything worked as intended.

8.1.1. Plan

Our plan for systems testing will include 3 phases:

Phase 1- In this phase, testing the forms flow from one interface to another seamlessly and with no long loading times. Once this has been established, we will insert sample problems into the application to make sure that the questions and answers are registered correctly.

Phase 2 - During this phase, we will have the database fully built out and implemented into our Visual Basic application. The application will be able to read all of the information from the database and display the information on the forms in Visual Basic.

Phase 3- The application will have a completed question and answer loop that reads problems from the database and checks the user's answer against the correct answer stored in the database. This phase will be used to check the final function of the project which will be the login form. This form will have users create login IDs and passwords that will be stored in a separate database for future use. These logins will be tested, by creating new login credentials, and trying to login multiple times by using those credentials.

8.1.2. Schedule

Testing will take place from 2/17/2020 to 3/2/2020. In that time we will determine if all the acceptance criteria has been met.

8.1.3. Acceptance Criteria

Acceptance criteria for the system test, will be that all of the main functions of the application (question loop, login, and results page) all work properly. If all of these work properly and without any major bugs, we will be ready to move on with the development of our project.

8.2. Prepare Conversion Plan

8.2.1. Plan

We will not have a conversion plan for our project because this is an initial software development project for our application. There is no originating system. This application is being built from the ground up.

8.2.2. Schedule

N/A

8.2.3. Systems Acceptance Test

N/A

8.3. Install Databases

Our database is embedded within our program. Aside from the development of this database there is no need to install any other databases.

8.3.1. Populate the New System's Databases

Data has been populated within the system database and may be referenced via the file included in the application.

A good sample username/password to log in as is:

Username: Bob
Password: welcome

8.3.2. Hardware - servers, etc.

The hardware that will be needed for our application, will be the user's device. Other than the computer no other hardware will be needed to run the application.

8.4. Training

8.4.1. Plan How To

Training for our program will be relatively simple, as the program is designed to be easy to learn. The following will be the stages for our training:

- Begin the program by running it on your device.
- If you are a new user, follow the on-screen instructions to create a new username and password.
- If you are an existing user and have a pre-existing username and password, please use those credentials to log in.
- Once you are logged in, follow the on-screen instructions to either take an assessment test from available math disciplines.
- Once you complete an assessment test, you will be shown your results. If you wish to complete another assessment, please select the option to return to the discipline selection form where you will be able to select another discipline.
- If at any moment you wish to exit from the application, the user will only need to select the exit buttons from any of the forms.
- If the user would like to go back to the login form for any reason, a login button will be available on some selected forms.

8.4.2. Schedule

Since training is not that large of an undertaking for our product, we felt that a detailed training schedule is not required.

8.4.3. Training Manuals, User Guides

As far as training goes for the program, our goal is to have the finalized project be as easy to use as possible for our users. The hope is that there is no need for any additional training in order to use the program. We are going to include a simple step based list to show the order of procedures for this system:

1. Open the program. If you are a new user, you may create a new username and password by registering. If you are a returning user, you can login.
2. If this is your first time using the program it is recommended that you take the assessment test to see what areas you are strong in and what areas you need to work on.
3. Once the assessment test is complete and the results are given, the quiz selection will appear.
4. On the quiz selection form, the options will be given to choose a math operation that you want to work on.
5. Once you have chosen a math operation that you want to work on, you'll be brought to a quiz featuring that operation. You will then take and complete the quiz. Following that, your results will be displayed.

8.5 Convert to New System

8.5.1. Stakeholders Discuss Experiences

Scott - This experience working on this project was pretty smooth. By having our plans and diagrams laid out previously in our CS 620 class it made the construction of the application a lot easier.

Julia - Since the CS 620 class, having the diagrams and interfaces from section 6, made the process of this project in CS 670 very easy to lay out the plan for this class and what our group had to do for the interfaces. Knowing that the entire group knew how to use Visual Basics also helped make the process easier to collaborate and code.

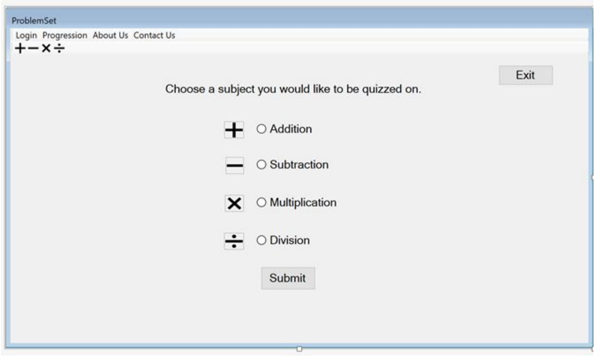
Meghan - Coming into the project halfway through, I was able to add value by sharing my real world experiences and asking a few more in depth questions, helping to hone the project requirements from an outside perspective. Additionally, seeing the CS620 documentation helped me to jump onboard without missing a beat. Having confidence in my teams capabilities and seeing them at work has left me with a great experience and confidence in our application.

Mike - Working on this project was a very smooth and stress-free experience. Laying out the grounds in CS620 definitely played a huge role in how easy the coding process was. Each teammate did his/her fair share of the work and contributed to discussions in a productive manner.

Vathsav - Working throughout this project was really interesting and helped me learn about the software development process. This project also taught me a lot about teamwork and the benefits of working together to achieve a common goal.

8.5.2. Agree on Future Enhancements – Clearly Documented

- Save file implementation.
- Visualization of user’s scores in graphs and other appropriate models.
- Addition of other subjects outside of math.
- Creation of rotating problem sets using all current math disciplines and additional ones as well.
- Port programs to different formats so it can be used on mobile devices and other operating systems outside of just windows.
- Add visual aids for students to help with knowing the wording of the math operation terms. Here’s a sample of what to expect:



Choose a subject you would like to be quizzed on.

8.5.3. What did we learn – Openly documented

We learned about the information systems development lifecycle. This includes the necessary documentation and testing. We also had some difficulty coding the program itself, which required help from outside sources such as other team members, internet resources, and academia. It was important that we learned this lesson because a programmer in a corporate environment, may or may not have all the knowledge necessary to complete a project on their own, and it is important to recognize that there are many sources that one can go to for help.

9.0 Alphabetical Glossary & Index

9.1 Glossary

Activity Diagram - A flowchart to represent the flow from one activity to another activity.
Assessment - The evaluation of the ability of someone.
Client - A person or organization using the services of a professional person or company.
Context Diagram - Identifies the flows of information between the system and external entities.
Database Specialist - Develops databases for businesses and organizations.
Intranet - A local or restricted communications network.
Math Problem Database - A structured set of data containing math problems and answers that is held in the system.
Module - A self-contained section that covers one subject.
Operator - A symbol or function denoting an operation.
Progress Report - A measurement of how much of something has been finished.
Sequence Diagram - Depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario.
User - A person who uses or operates a computer or other machine.

9.2 Index

10.0 Citations

10.1. Books

Whitten, J. L., & Bentley, L. D. (2007). *Systems Analysis and Design Methods*. McGraw-Hill/Irwin.

10.2. Articles

Sztajn, P. (2003). Adapting Reform Ideas in Different Mathematics Classrooms: Beliefs Beyond Mathematics. Retrieved November 18, 2019, from https://link.springer.com/article/10.1023/A:1022171531285.

11.0 Contacts

Names	**Contact Information**	**Area of Expertise**
Scott Yanoff	203-451-8644 scott.a.yanoff@gmail.com	Documentation Writer
Julia Katz	203-273-8807 jj7jj7@gmail.com	Software Development, Framework Designer, Documentation Writer
Michael Lawrence	804-517-0998 michaellawrence193@gmail.com	Software Development Experience, Documentation Writer
Vathsav Tipirneni	475-231-9054 vathsavtipirneni97@gmail.com	Documentation Writer
Meghan Polis	631-278-3117 meghan.polis@gmail.com	Database Design, Software Development Experience, Documentation Writer

12.0 Appendices

12.1 Appendix A

Executive Summary

Our Mission

OWL inc. offers superior technology solutions for the education industry, providing the attention to detail and care that every student deserves. Our team will ensure that both parents and students can be satisfied with high quality learning products. Our program will help students learn mathematics through a game-like environment that will keep them engaged.

The Company and Management

OWL inc. is headquartered In Fairfield, Connecticut and incorporated in Wilmington, Delaware. The company is owned by partners Scott Yanoff, Julia Katz, Meghan Polis, Michael Lawrence, and Vathsav Tipirneni . These partners all have extensive experience in education and information technology and each of them has attained a master's degree in computer science.

Our Services

Our clients are elementary school children and their parents who feel that they need a supplementary learning resource that they can use outside of school. (Insert company name here) offers a variety of technology solutions, all in the comfort of the child's home or wherever they have access to a computer. Some of the services offered include:

- Math Exercises
- Language Skills
- Reading and Writing Comprehension

The Market

Across the United States the education industry has been relatively stagnant over the past 10 years or so. Most of the stagnation has taken place in the STEM (Science, Technology, Engineering, and Math) disciplines. Our market research has shown that across the world the United States does not even rank in the top 10 for standardized STEM test scores.

Our Competitive Advantages

While there are currently other educational services companies offering similar services in the US, not many of them offer the ease of use that our product does. OWL inc's. marketing strategy is to emphasize that our product is easy to use and offers a comprehensive solution for the user. Another advantage that we have over our competitors is that our product was designed by people that have experience working with elementary school children, so we know how to engage the target audience.

Financial Projections

Based on the size of our market and our defined market area, our sales projections for the first year are $340,000. We project a growth rate of 10% per year for the first three years.

The salary for each of the co-owners will be $40,000. In startups, we will have 5 staff members to provide support and development services and expect to hire four more this year once financing is secured. To begin with, co-owner Julia Katz will be scheduling appointments and coordinating services, but we plan to hire a full-time receptionist this year as well.

Already we have service commitments from over 40 clients and plan to aggressively build our client base through newspapers, websites, social media, and direct mail advertising. The comprehensive innovative educational support that (Insert Company Name here) will provide is sure to appeal to both students and parents alike across the United States.

Start-up Financing Requirements

We are seeking an operating line of $150,000 to finance our first-year growth. Together, the co-owners have invested $62,000 to meet working capital requirements.

This concludes the executive summary example based on the fictional company (OWL Company)

12.2 Appendix B

USABILITY ANALYSIS for OWL: Offline While Learning		
PRODUCT UNDER TEST	What's being tested?	The functionality of the OWL: Offline While Learning application will be tested.
	What are the business and experience goals of the product?	To determine design inconsistencies and usability problems with the user interface and content areas. To utilize the application under a controlled test environment that is representative of the users. To baseline user performance and overall experiences to gauge user experience and determine future improvements for application growth and expansion.
TEST OBJECTIVES	What are the goals of the usability test?	The goals of usability testing include establishing a baseline of user performance, validating that project features fulfill user requirements and to identify potential design concerns that may need to be addressed.
	What specific questions will be answered?	Is this application easy to use? Can you successfully log in? ***Parents/Guardians:*** Can you create a profile for your child/student? Can you review your child/students results? ***Children/Students:*** Can you take an assessment? Can you pick a quiz topic?

	What assumptions will be tested?	Technology can help supplement classroom learning.
TEST TASKS	What are the test tasks?	1. The user will enter login info. The screen to provide login info and will identify if it is correct or incorrect login information. 2. The user will be prompted with a math problem and will select an answer to the problem, which the answer from the math problem database will respond with a right or wrong. If the answer is correct, the results will add one point to the total score. If the answer is incorrect, the results will remove one point from the total score. 3. The user will be assessed of current skill and will elect to take a pre test which then contacts the database of test questions.
RESPONSIBILITIES	Who is involved in the test?	The participants' responsibilities will be to follow a representative scenario based on their role type. The role types will be: ‘Parents/Guardians’ or ‘Children/Students’.
RESPONSIBILITIES	What are their responsibilities?	The ***‘Parents/Guardians’*** will be required to perform a number of tasks, to include: creating his/her user profile, logging in, creating children profiles, and viewing assessment results. These users are expected to be performing in a real-life role of parent or guardian to at least one child.

		The ***'Children/Students'*** will be required to perform a number of tasks, to include: logging in, taking the initial assessment, selecting a specific quiz type, and taking a few quizzes. These users are expected to be performing in a real-life role of a student between the first and fourth grades, with a basic knowledge of adding, subtracting, multiplication and division. All participants will be directed to provide honest opinions and feed-back based on using the application in a post-session questionnaire.
BUSINESS CASE	Why are we doing this test?	To make money To make sure that the application works and does in fact improve the mathematics learning experience for the user.
	What are the benefits?	Improved learning experience for users.
	What are the risks of not testing?	Bugs, unexpected outcomes, and bad user experience.
EQUIPMENT	What equipment is required?	Computers, networking equipment, and server racks.
	How will you record the data?	In a custom database as well as internal documents.
LOCATION and DATES	Where and when will the test take place?	Testing will take place between February 25, 2020 and March 3, 2020 at Sacred Heart University.
	When and how will the results be shared?	A pdf file to interested parties by March 10, 2020.

12.3 Appendix C

Project Risk Evaluation		
Project: Offline While Learning (OWL)	Completed by: Scott Yanoff	Date: 1/20/2020
Factors affecting project risk	Rating*	Comments
1. Characteristics of the organization a. Has stable, well-defined objectives?	+1	We understand what the end result should look like and have a clear understanding of how to reach that result.
b. Is guided by an information system plan?	+1	Yes, our project will be guided by the documentation in the “OWL: Online While Learning Project Documentation”.
c. Proposed system fits plan and addresses organizational objectives?	+1	Our project will address most if not all of the issues we identified in our section 3.3 of the “OWL: Online While Learning Project Documentation”.
2. Characteristics of the information system a. Model available/clear requirements?	+1	The information system is simple to work with, but robust in functionality, with detailed requirements as per the “OWL: Online While Learning Project Documentation”.
b. Automates routine, structured procedures?	+1	The finished project will be able to run with the coded procedures based on specific user inputs.
c. Affects only one business area? (No cross-functional or inter-organizational links)	+1	Our main business that we are affecting with this project is the educational industry.
d. Can be completed in less than one year?	+1	Yes, with our current plan we are hoping to have everything for this project finalized in 3-4 months.
e. Uses stable, proven technology?	+1	Yes, all technology used is stable and has been proven over a number of years.
3. Characteristics of the developers a. Are experienced in chosen development methodology?	+1	Our developers are all experienced with different expertise. For example some of our developers are highly experienced with databases while others are very experienced with visual basics.

b. Are skilled at determining functional requirements?	+1	Yes, through our various planning methods we have become very skilled at identifying functional requirements.
c. Are familiar with technology and info architecture?	+1	Yes.
4. Characteristics of the users a. Have business area experience?	+1	Users will ideally be younger children who need extra help with mathematics.
b. Have development experience?	+1	Most of our group has experience working with all of the technology, but not in a professional setting.
c. Are committed to the project?	+1	We are all 100% committed to this project.
Total points	14	
* +1 = yes 0 = maybe -1 = no		

12.4 Appendix D

Unit Testing:

Test #	Input	Expected Output	Test Condition
1 Log in	xyz@gmail.com	Enter the registered login or create new account	Entered into the app
2 Correct	2+2	4	Test correct
3 Incorrect	2+2	5	Test incorrect
4 Discipline	Addition, Subtraction, Multiplication, Division	Enter into the selected page	Test done