Community College Pre-Assessment and Placement Tools: Development of the PAT

Project Plan

Team Members:
Josh Schmitt
Peter Benzoni
Colin Duffy
Nathan Steussy
Alex Yasa

Client:
East Los Angeles Community College

Advisor:
Diane Rover

Group May15-04

Table of Contents

1 Problem Statement

The goal of the PAT- ELAC group is to create a web based Pre-Assessment Tool (PAT) that will allow students to determine their projected class placement in English and Mathematics by taking one or more pre-tests in each area. The Mathematics test will be adaptive by level, beginning with basic pre-algebra and proceeding through calculus. The English test will be two separate tests designed to test students' aptitudes in different areas of English knowledge.

Based on the result of the tests, the tool must output a placement level and a variety of resources focused on their areas of weakness to assist students in preparation for the final assessment.

For research purposes, and among other information the PAT will collect student information (ID number), overall pre-assessment quiz scores, and scores for individual items on the quiz, to be used to determine PAT question, topic, and test efficacy.

The primary stakeholder in this project is East Los Angeles College, (ELAC) with secondary stakeholders of Jefferson Community and Technical College, (JCTC) Iowa State University, (ISU) Des Moines Area Community College, (DMACC) and the Iowa Board of Regents.

The primary participants in the PAT-ELAC project are we, the five undergraduate software developers; Ruben Arenas, the point of contact for ELAC; and our advisor, Diane Rover.

2 Deliverables

The overall products of our development will consist of four primary deliverables and the required sub-criteria thereof:

- Fully Functional Website
    - Well Documented Code
    - Built in a Model/View/Controller framework
    - Consistent, Appealing, and Simple User Experience
    - Scalable
    - As per-user customizable as reasonably possible
    - Presents vital statistics in a usable analyzable form

    - Logging
    - Clear and understandable Data Models
    - Entirely (or nearly) bug free
- Project Report
    - Reflection
    - Guide to PAT use and, particularly, administration of the PAT
    - Highlights of key features
    - Avenues for expansion
    - Documentation of the design and creation process
- Project Presentation
    - Walkthrough
    - Key Challenges
    - Important outcomes
    - Addresses all of our identified goals

Additional deliverables may arise in the development of the overall product, but we consider these to be the three most important items for us to produce.

3 Specifications

The Pre-Assessment Tool will have three main categories of users: students, instructors, and school administrators. The capabilities of each type of user are described below.

3.1 Student capabilities:

- Register an account by navigating to the registration page and providing a username, email address, and password. After validating their email address, their account will be activated.
- Log in to their account by navigating to the login page and providing their username and password. After authenticating the student, the system will redirect the student to the student homepage.
- Take a math exam by selecting the appropriate button on the student home page. In the math exam, students will answer a series of ten questions at a time from a test created by an Instructor. If a student answers enough questions correctly (exact number determined by the school admin), they will move on to more difficult questions. If a student misses too many questions, the test will end and they will be given their math placement score.
- Take an English exam by selecting the appropriate button on the student home page. In the English exam, students will take a two-part exam consisting of reading comprehension and sentence completion. After completing both sections, students will receive their English placement score.

- View their math and English placement scores by navigating to the student home page.
- View resources, such as YouTube videos and worksheets, that specifically cover the topics that they lost points on during the exam. The resources can be found by navigating to the student home page and selecting “View Resources”.

3.2 Instructor capabilities:

- View exams, topics, and questions that they have already created. They are displayed on the instructor home page.
- Create an exam by clicking on the appropriate button on the instructor home page. They will be prompted to select which topics to include on the exam and give the number of questions over each topic. Then the instructors can enter the different courses that students can test into and indicate how many correct responses are required to test into each course.
- Edit an exam by selecting an existing exam on the instructor home page and clicking the edit button. Instructors can select what part of the exam they wish to edit, such as the number of questions, topics covered, or adjust the placement courses.
- Delete an exam by selecting an existing exam on the instructor home page and clicking the delete button. Instructors will be prompted to confirm the deletion before the exam is deleted. Topics and questions are unaffected by deleting exams.
- Create a new question by clicking on the appropriate button on the instructor home page. Instructors will be prompted for the question, the correct answer, and 1-4 incorrect answers. They will also have the option of labeling the question under an existing topic.
- Edit a question by selecting an existing question on the instructor home page and clicking the edit button. They can edit the question, correct answer, or incorrect answers.
- Delete a question by selecting an existing question on the instructor home page and clicking the delete button. Instructors will be prompted to confirm the deletion before the question is deleted.
- Create a topic by clicking the appropriate button on the instructor home page. Instructors will be prompted to name the topic and then a list of all questions will be displayed. They can select which questions should be added into the newly created topic. They can also upload worksheets or videos to be used as resources by students who struggle with the topic.
- Edit a topic by selecting a topic on the instructor home home page and clicking the edit button. Instructors can edit the name of a topic or add/remove

questions from the topic. They can also add/remove resources associated with the topic.

- Delete a topic by selecting a topic on the instructor home page and clicking the delete button. Instructors will be prompted to confirm the deletion before the topic is deleted.
- View student reports by clicking on the appropriate button on the instructor home page. Instructors view students math and English placement scores as well as more detailed information about their exams. Students can be sorted by name, score, or ID number.

3.3 School Administrator capabilities:

- Add an instructor to the system by navigating to the admin home page and clicking the appropriate button. School administrators will supply an email address and name of the instructor and the instructor will be emailed with instructions on setting up their account.
- Remove an instructor from the system by selecting the instructor on the admin home page and clicking the delete button. School administrators will be prompted to confirm the deletion before the instructor is deleted.
- View aggregate data on students' results by navigating to the admin home page and clicking the appropriate button.
- Use all of the functionality added for instructors.

4 System Requirements

4.1 Functional Requirements

Requirement	Testing
Must allow students to take math and english tests	We will use functional testing on multiple sets of Math and English tests. We will ensure that we test every possible placement.
Allow students to save incomplete tests that can be revisited some other time	We will start many tests and come back to them several days later to finish. We will then review results to ensure proper scoring.

Allow students to review their results	After taking tests as in the above criteria, we will ensure that results are shown. We will also ensure that results can be reviewed from the users profile page.
Allow students to view resources to study	Login as an admin and add resources. Run queries on the database to ensure they were added. Login as a student and ensure these resources load properly.
Give students proper placements	
Allow admins to create tests	Login as an admin and attempt to create tests. Run queries on the database to ensure the test was added.
Allow admins to modify tests	Login as an admin and view list of tests. Select a test to modify. Update topics, description and/or type. Run queries on database to ensure update was made. Attempt test as a student.
Allow admins to review students results	Run queries to view certain student placements. Login as an admin, select same students and compare results.
Allow admins to provide resources for specific topics	Login as an admin. Attempt to add a resource. Query database to ensure resource was added.

Table 1: Functional Requirements

4.2 Non-Functional Requirements

- Scalable to allow many users at any time
- Give students placements in a timely manner
- Interface that is simplistic enough for minimal training
- Students should be able to use the tool with zero training
- Scalable for more test types in the future

4.3 Assessment of Proposed Solution

The proposed solution provides a very scalable design, mostly due to the database which is designed in way to handle any type of test, any topic areas for a test, any number of questions for a test and any number of possible answers for a question.

Drawbacks to this solution are the number of queries required. This will not hurt performance much though, as SQL is extremely fast.

Along with the databse design, ASP.NET MVC framework is being used for development, which provides a modular structure to the source code. Separating the models, views and controllers in a web application allows for easy optimization. The ability to use C# and standard object oriented practices for back end code allows a large amount of flexibility. While the front end is developed using Razor, which blends html and C# to provide simple design with advanced capabilities. Two drawbacks here are that the application must run on a Windows server and the developing team has little experience with the framework.

4.4 Validation and Acceptance Test

Functional testing practices are mentioned alongside each requirement. This practice was chosen opposed to unit testing for simplicity. Developers will do all functional testing until December when the application will be deployed to the server, which will allow our client to start testing as well. Once our client has more visibility, the developers will have more customized feedback which will allow for a

5 System Description

5.1 Process Details

The current structure of the project is modular. The application was built using the ASP .NET MVC framework, so the code is separated into models, views, and controllers. The team has implemented basic login functionality for users, using CSHTML to generate the view and C# for the model and controller. Using the Entity framework, user information is stored in the database. The modular design of the project will allow us to add additional functionality by building on the existing MVC structure. We have started adding additional views for student and admin web pages. Next semester the team will implement the additional models and controllers. The database schema will need to be modified to store information about tests and resources for students.

5.2 Test Plan

In the course of the prototype's development, several tests will be conducted to determine and enhance the viability of the model, with each test changing in sophistication and purpose as the project nears completion.

5.3 Test Coverage

Testing will begin as rudimentary checks to determine if each program part is compiling and committing in a timely manner, with later checks for bugs that can be removed and for successful communication between parts of the program.

As the project nears completion, further testing will take place using various case examples to see how the prototype reacts in each case. Ultimately, a full test of the prototype will be conducted to determine if the prototype is ready for delivery to the client.

An added feature to this process, in the case of the current project, will be the entry of the client in the later stages of testing, both for quality assurance and as an opportunity to test cases that may not have come up in earlier development.

5.4 Test Methods

There are four test methods that will be used for this project. The vast majority of the tests that will be used will be usability tests and regression tests. The usability tests will be used to test components and features and determine if they fit the benchmarks set by the client at the time of handover.

To that end, regression tests will be conducted beforehand during the development process to ensure proper function and coverage, and each developer has a hand in ensuring the success of the parts before further tests.

These tests will consist of a specific set of tests for roles within each part, such as build acceptance (which consists of both the successful compiling of the part as well as successful subversion), a related smoke test (determining if further testing is possible with the current level that the part is at), bug regression (in the simplest

sense, finding errors and troubleshooting them), and critical path testing (assuring that each part of the project is flowing to where it is supposed to end up and focusing on features that will be part of everyday use).

After it is reasonably certain that all major testing problems have been solved, unit testing will take place, allowing each developer to determine if proper functionality and flow has been accomplished in the course of development.

Closer the end of the project, before complete handover to the client, there will be either one or a series of final release tests to determine that the prototype is ready to be used in the field. At this point, any minor and non-threatening bugs will be evaluated and final troubleshooting will take place.

5.5 Testing Tools

In the course of testing the prototype, the team has found that common programming and subversion platforms work best for the needs of development, testing and troubleshooting.

For regression testing, Visual Studio is being used for most evaluation needs, such as bugs and other errors, as well as to measure successful compilations and program execution. As well, TortoiseSVN is being used not just for subversion, but to measure the health of each new submission, determining if a new addition can be brought in without damage to the rest of the project.

For further testing involving end users, the project is being released to the client by way of movement to a Windows server, so that the client can conduct their own tests on their own equipment for how much the project matches their specifications.

When the project reaches the mobile development phase, the regression and usability cycle will restart for testing the new components. Because of its application for Android and other XML development, Eclipse is the current favorite for creation and testing at this late phase.

5.6 Test Responsibilities

The project team will be conducting the rudimentary regression tests (build acceptance, bug regression, critical paths, etc.) during the early phase of the project, as these are relatively simple to check and easy to fix. The client will be introduced when the time comes for unit testing and the final release test. As well, other end users may be introduced for later testing stages if the need arises. These end users would be other students, professors, administrators or programmers, and could be used to evaluate their respective parts or niches within those parts.

Data to be collected from the tests that is not part of the regular course of building the program will include test questions and answers, student scoring and course information, information and acceptance levels for the courses themselves, and (potentially) resources that can be used to preview courses and as reference material for the subject matter in the tests.

6 Work Breakdown Structure

6.1 Project Schedule

The following tables show, by semester, the schedule for the project. This is divided first by the tasks that need to be completed during the given periods, and then into who was responsible for what part of the project (or tasks listed alone for group collaborations).

Fall Semester 2014								
Area	To 9/15	To 10/1	To 10/15	To 11/1	To 11/15	To 12/1	To 12/15	To 1/1
Project Prep	Research project requirements and start using Visual Studio for code development			All further tasks are outlined in the subsequent rows, but is related to the actions done in the project prep.				
Website Framework			Josh: Build website	Josh: Maintain website and add new documents to the site as required.				
Build Database			Colin: Find and secure server needed to store database		Colin: Continue updating database tables Alex, Nathan: Integrate test taking, results and other program parts into the server			
Implement Basic Admin Functions				Alex: Create basic framework for admin page Peter: Flesh out admin page and create spinoff functions for page Colin: Create superadmin and user roles		Nathan: Create student info page for admins All: Debug problem areas within the prototype		
Develop Testing	Colin: Create prototype model for project.				Alex: Create test taking and results pages Peter: Create page for question creation as well as storage for answers		Peter: Edit question creation so that it writes to database Colin: Update test model for database table	
Implement Basic User Functions					Colin: Create and update user login, other user roles Josh: Create basic framework for student page			All: Debug problem areas within the prototype
Implement MVP							(Carries over to late Jan. 2015) Move project over to dedicated servers, open it up to client	
General Project Items	Biweekly meetings with advisor, biweekly communication with client, weekly meetings to review tasks and documentation, weekly status reports							

White boxes are initializations
Gray boxes are maintenance

Table 2: Fall 2014 Schedule and Duties

Spring Semester 2015								
Area	To 1/15	To 2/1	To 2/15	To 3/1	To 3/15	To 4/1	To 4/15	To end of term
Add Extra Features		Evaluate need to add non-essential features (references to course material, videos, etc.)						
Testing/ Troubleshooting	Preliminary viability testing will be done beforehand.		Begin running prototype, fix any bugs that come up, add essential components as necessary					
Mobile Development				Adapt prototype to work on Android (will add xml programming at that time).				
Verification, Documentation, etc.	Most documentation tasks will be done throughout the semester and are listed under "General Project Items" in the Fall 2014 table.						Complete documentation, present at end of semester	

Table 3: Spring 2015 Schedule and Duties

As well, the tables below demonstrate further work to be done on the PAT after the completion of the team's installment. By necessity, the categories on these tables are different from ours, to reflect involvement from outside groups and more general project needs at the relevant times.

Year 1				
Area	**Summer 2015**	**Fall 2015**	**Winter 2016**	**Spring 2016**
Human subjects	Obtain final IRB approval at all sites			
Mathematics and English/Reading PAT questions	Generate list of topics covered by assessment test packages that are universal		Develop PAT questions for topics	Iterative Process of testing the questions
		Develop practice problems tied to topics		
Promotion and Marketing	Design promotions and marketing strategies		Print promotion and marketing materials	Train staff to use materials
Mathematics and English/Reading videos	Choose actors from existing student tutors.	Train actors.	Pre-production/ Production/ Post-production (repeat for each topic)	
	Contract videographer	Inventory existing videography equipment. Purchase as necessary.		
Site Visits		Visit DMACC	Visit JCTC	
Software development	Assessment of prototype PAT strengths/weakness.	Needs assessment for PAT software at 3 test sites	Present software design to PI and co-PI's	
		Initial PAT design	Refine design	Programming of the PAT as a web-based system
Research and Evaluation	Design meetings	Map of the PAT	Design of the data collection system	Create databases (College level data)
Advisory board	Finalize advisory board membership	Initial advisory board meeting		Advisory board meeting
Dissemination		Present prototype PAT at student success conference		Presentation at a state and national conference

Table 4: Plan for Summer 2015 to Spring 2016

Year 2				
Area	**Summer 2016**	**Fall 2016**	**Winter 2017**	**Spring 2017**
Human subjects	Renew IRB documentation			
Mathematics and English/Reading PAT questions	Develop PAT questions for topics			Testing of questions
	Develop practice problems tied to topics			
Promotion and Marketing	Continue development of materials	Web-based promotions		
Data Collection	Collect college level data including levels of specified enrollments (Math and English)			
YouTube	Upload videos to YouTube			
Mathematics and English/Reading videos	Rehire and train actors as necessary			
	Pre-production/ Production/ Post-production (repeat for each topic)			
Software development	Review of design with PI and co-PI's	Presentation of software to test sites	Software tests	Installation of PAT at test sites
	Programming of the PAT as a web-based system		Complete programming of web-based PAT	initial field based testing
	Design of PAT app	Programming of PAT app		Complete programming of PAT app
Site Visits		Visit JCTC		Visit DMACC
Research and Evaluation	Pilot studies at ELAC begin	Descriptive analyses - gender, Pell, ethnicity, etc.		PAT goes live and begins collecting student data
Advisory board		Advisory board meeting		Advisory board meeting
Dissemination		Present on PAT - state level organizations in CA, IA, and KY		State and National conference presentations

Table 5: Plan for Summer 2016 to Spring 2017

Year 3				
Area	**Summer 2017**	**Fall 2017**	**Winter 2018**	**Spring 2018**
Human subjects	Renew human subjects approvals			
Mathematics and English/Reading PAT questions	Refine PAT items based on statistical data			
	Add to practice problem database			
		Field testing of questions		
Data Collection	Collect college level data including levels of specified enrollments (Math and English)			
Promotion and Marketing	Monitor promotion plans and modify as necessary			
Mathematics and English/Reading videos	Reshoot, refine, and edit videos as needed			
Software development	Software tests of PAT and modifications as needed			
YouTube	Testing of YouTube videos			
Research and Evaluation	Tests of Fidelity of the instrument - Analyses- and reports		Analyses - Use of materials (videos and other)	Evaluation of PAT use
Advisory board		Advisory board meeting		Advisory board meeting
Dissemination	Writing articles for publication in national journals		DREAM conference	Presentation at AERA

Table 6: Plan for Summer 2017 to Spring 2018

Year 4				
Area	**Summer 2018**	**Fall 2018**	**Winter 2019**	**Spring 2019**
Human subjects	Renew IRB approvals			
Mathematics and English/Reading PAT questions	Refine PAT items based on statistical data			
	Add to practice problem database			
Data Collection	Collect college level data including levels of specified enrollments (Math and English)			
Mathematics and English/Reading videos	Reshoot and refine videos as needed			
Promotion and Marketing	Monitor promotion plans and modify as necessary			
Site Visits		Visit JCTC		Visit DMACC
Software development	Review functioning of PAT software and modify as needed			
Research and Evaluation	Survival analysis methods & event history analysis			
Advisory board		Advisory board meeting		Advisory board meeting
Dissemination	Writing articles for publication and presentation	Presentations at state level		National conferences

Table 7: Plan for Summer 2018 to Spring 2019

6.2 Risks/Feasibility Assessment

A feasibility assessment, by necessity, must consider five aspects of what makes the PAT possible and what impact those areas have, in both the the short and long term. These areas, technical, economic, operational, and schedule feasibilities, are examined in detail below:

Arguably the primary hurdle in determining the feasibility of the PAT is the technical feasibility of such a product. As students, the team lacks much of the technical expertise , technical resources, and technical tools typically present in the creation of

such a project. However, these three areas of potential pitfall can be mitigated. First, our lack of expertise, both in the framework and overall, is made up for by the vast technical knowledge base available online, especially for the C#/ASP.NET MVC Razor framework, one of the most used web frameworks. Any potential problems should be able to be addressed by the large number of tutorials and resolved problems online. In regards to tools/resources, Iowa State University has provided much of both, by providing a server and hosting, as well as bandwidth, all adequate to its current needs. Moreover, its software license have been essential in getting access to proprietary software. Where Iowa State falls through, however, are easily made up for in open source software and libraries. In the long term, the coalition of schools coming together on the PAT have world class technical resources and support, so it shouldn't become an issue.

Many of the economic considerations are covered in the cost analysis section, so this section will be general and brief. The primary economic benefit here is primarily to the student, who can by using the PAT prepare for tests that cost between $25-75 per student. At universities where this bill is footed by the institution, this savings can be transferred to the institution. Moreover, much of the costs associated with the site are to be offset by a series of grants.

Operational feasibility, too, is of concern. The PAT must not only solve the problem presented, but solve it well. Namely, at its minimum, it must:

- Predict a community college applicant's placement in both English and Mathematics regardless of the required placement exam (ACCUPLACER, COMPASS, or ALEKS)

- Be easily calibrated to college-specific cut-scores aligned with course and remediation levels

- Recommend instructional videos and other materials specifically aligned with the applicant's deficit areas (in both English and Mathematics)

- Be assessable and repeatable by students without cost

(Arenas, Et Al 4)

As the proposed solution and specifications are fundamentally centered around these assumptions and how to meet said needs, this is perhaps the area it is most suited for. Finally, the project team has put particular emphasis on reliability, usability, and maintainability and have therefore integrated them into aspect of the project.

The problem of schedule is primarily addressed above in the schedule system. The PAT team is part of a larger implementation plan and has broken its section down into smaller, manageable parts. Given the level of technical experience and total period for development, it is believed that should be easilbly achievable

The financial feasibility of the Pre-Assessment Tool is expanded in considerable detail in the Cost Considerations section below. The gist of the conclusion, though is simple: low costs in the initial period of the site, though these will rise quickly with time

Table 8 represents the potential risks that the development team faces.

Risks	Probability	Criticality	Risk Factor	Mitigation Strategy
Not being able to effectively communicate with client due to distance	0.9	High	Low	Email makes it easy to discuss things with our client. Also, our meeting schedule is fairly flexible so we're able to be efficient with our time.
Unfamiliarity with C# and MVC.net framework	0.8	Medium	Low	Use online tutorials and resources to ensure that all members of the team are able to help develop the project effectively.
Creating adaptive testing structure	0.95	High	High	Collaborate as a team to help best design the adaptive testing algorithm.
Client decides to use different language or framework	0.05	Very High	Very Low	Develop our code in a way that we can reuse it on a different platform easily, if need be.
Client decides to change project requirements	0.25	Very High	Medium	Keep track of all designs and plans to make any changes easier to deal with.
Not having enough time to finish due to busy schedules	0.1	Very Low	Low	Plan our time effectively to help mitigate any project halts due to schedule

				constraints.
Lack of user base for field testing	0.95	Medium	Low	Keep our client updated with all progress, and test it with others as we go to ensure that they are satisfied with the project.
Website optimization may be difficult	0.8	Low	Medium	Continuously test and improve optimization of the project.

Table 8: Risks & Mitigation Strategies

6.3 Cost Considerations

Costs, while not of paramount concern in the early stages of the project, must still be considered, especially as this project is expected to scale to the national level within the next 5 years, and could experience relatively high levels of use as it is adopted among universities across the nation, as envisaged in the 45 page proposal narrative provided by the author. Such a vision can be executed in one of two ways - single source hosting (Likely Iowa State University) or virtual web hosting. (Amazon AWS) Both possibilities will addressed herein, along with initial costs (those created during development) and an evaluation of costs of ongoing support upon completion of the Senior Design Project phase of this project.

Initial design costs are extremely low. As this is a senior design projects, the developers are unpaid, and due to the hosting being kept in house, the two primary costs normally associated with web development, then, have been effectively mitigated entirely. What costs that remain are simply small incidentals, such as software licenses and pizza.

Looking into the slightly longer term, upon completion of the Senior Design, the university consortium will likely have to hire someone to maintain and update the product. This will cost >$50000 for a single developer. It is suggested that the university consortium fold this into one of their web development departments, such as the eminently qualified Iowa State WebDev team.

With this approach, then, it also makes sense to host in house (at Iowa State) with its privileged bandwidth and large number servers. Even scaled, the University should be able to manage the relatively high number of requests that would come through (particularly in the late summer) This would allow them to keep annual costs in a range affordable to the entire consortium of schools, ~$100,000 based on the cost of server maintenance and bandwidth

Though, in terms of just in time scalabilities, Amazon AWS reigns, so it remains to be seen which would be the most affordable in the long run.

6.4 Market/Literature Survey

Arguably the most essential part of a feasibility assessment, this marketing and literature analysis is split into its two logical parts: Marketing, which will consider the potential demand for the services the PAT provides and includes a long term plan for adoption. It will then consider literature - that is, existing alternatives as well as research on adaptive testing and practice tests

Data collection will be a very important aspect of the PAT. A mechanism that will record student information as it's imputed into the system. As students log in, and they begin taking and finishing exams, their data will be collected automatically.

Table 9 (provided by ELAC) below represents the data that the system will collect as users are logged in and accessing the system.

PAT	Transcript data	Demographics (College Data)
Student ID	Student ID	Gender
PAT Scores - Mathematics and English	Actual Placement - Mathematics and English	Race
Predicted Placement	Enrollments in Mathematics and/or English	Age
# minutes observing resources	English/Mathematics grades and course withdrawal information	Pell eligibility
# times Repeat PAT	Retention and Completion	
	Semester and Cumulative GPA	

Table 9: User Data Collected

Through this information, ELAC will be able to determine specifics as to who is using their system, and what changes may be necessary. In terms of marketing, this allows for flexibility in the system, and invaluable instantaneous feedback.

The literature survey of the market shall be conducted by ELAC. This very comprehensive, extensive study will analyze the ways that commercial test systems, such as ACCUPLACER,

COMPASS, and ALEKS function in terms of accuracy placement. The PAT system will require such analysis to be done in order to ensure that the system is placing students in the correct spot. Individual research may be done by individual members of the senior design team to help gain an understanding of other projects on the market before ELAC has time to finish their study. This will aid in both the design and the development of the product, and also the ways that it will function in terms of accuracy and overall system placement. The main goal here is to provide ELAC with a system that will be easily expanded upon in the following years.

7 Conclusion

The Pre-Assessment Test system is being developed for East Los Angeles Community college to aid incoming students in class placement, and to assist in the collection of data that will allow the college to effectively tune the system to their needs. It will ultimately become a product that will be marketed to other colleges and universities. As the project moves forward, the feedback provided by the client will provide details that will help in nailing down specifics.

The project plan will be followed through May to ensure that all necessary tasks are completed on time, and that the client is able to obtain all information that they need in order to guarantee the product they receive is exactly what they want.