Assessing the Effectiveness of Project-Based Learning in The Computer Science Classroom on Student Mastery of Advanced Programming & Coding Concepts

An Action Research Plan

Jarrett L. Lindsey

Lamar University

Assessing Digital Environments - ADL 5315

Dr. Meeuwse

October 3rd, 2021

Assessing the Effectiveness of Project-Based Learning in The Computer Science Classroom on Student Mastery of Advanced Programming & Coding Concepts

Planning Stage

Action Research Topic

The conventional "stand-and-lecture" method of teaching computer science in high school classrooms has failed to inspire motivation and personal responsibility in students. Because what they are studying does not resonate with them on an interpersonal level, learners are not encouraged to see beyond what they are learning. Given the lack of a strong foundation when presented with early computer science components and theory, expanding on broader concepts is even more challenging. Because programming is still the most difficult subject for students in the computing area to grasp (Park 2016), the main principles of computational thinking must be investigated.As such, my action research will address the question of; What impact does project-based learning (PBL) have on student coding and programming application development skills in computer science?

Fundamental Research Question & Purpose of Study

Computational thinking is a term that refers to a combination of creative problem-solving talents, programming, computer hardware, and software. Human capacities solve problems using logical reasoning and code writing skills, resulting in the development of computer applications and hardware, demonstrating the value of computational thinking. (Djambong & Freiman 2016). According to Wing (2008), when humans utilize logical thinking to tackle difficult

problems by breaking them down into little parts and solving them using a mathematical approach, they are engaging in computational thinking, which eventually leads to the creation of computer applications.

This study aims to gauge student performance of learning programming techniques and concepts while engaged in a yearlong project. The project will involve developing a mobile application using the Swift programming language. While the fundamental research question is; Can PBL approach to teaching coding motivate learners to master coding techniques and concepts? PBL's benefits and implementation in teaching new ideas, engaging learners, and creating pathways to student-driven classrooms are highlighted in this study for Computer Science and STEM educators.

Research Design and Methods

The method that this study will utilize will be a hybrid model of quantitative and qualitative methods. This being my first action research attempt, I want to collect enough data to create a holistic vision of my action research, innovation plan, and contribution to the computer science education field as a whole. I will assume the position of instructor and will also collect data from students in my classroom. Objectively, at the end of the academic year, all learners (students and teachers) should complete a mobile application. The completion or incompletion of the project is where the qualitative data will be assessed. Learners will be assessed based on their project’s creativity, completion, programming level, and business acumen. The overall quality of their project will be determined by the project's learning level, the programming language used to construct it, and a rubric that will be established at the start of the project.

Quantitative data will be gathered from several different sources. Information will be collected from the number of projects completed over the school year. This information will aid us in determining the time it takes for learners to grasp their programming language and apply it to their project. Second, information will be gathered from the number of students who did not finish their projects during the academic year. The data will be evaluated using the rubrics as a reference point. The larger question that will be addressed is why the project was not completed and what circumstances contributed to the project's incompletion? This question is crucial to the study since it pertains to the project's most effective utilization of time and resources.

Type of Data Collecting & Measurement Instruments Utilizing

The types of data collected are presented in various formats. Each of them is outlined below with their respective rationales.

- *Rubrics' (Checklists) for Project/Assignments*: At the start of the project, learners will be given a rubric. Using this rubric as a starting point, they will see where they should be at the project's beginning, middle, and conclusion (see Appendix A).
    - *Project Scores*: The rubric will include a score based on the completed items.
    - *Project Completion*: The number of projects completed from the beginning of the year to the year's conclusion will be evidence of this.
- *Personal Reflection/Surveys (Quarterly)*: Every 2.5-3 months, learners will provide an individual assessment of how the project is progressing. Learners will describe current problems, target accomplishments, and their overall attitude toward the project (see Appendix B).

- *Observations*: Observation data will be used to assess student learning from the perspective of the instructor/mentor. The instructor/mentor will use the rubric as an observation tool to better evaluate current student learning and the requirement for any additional skills training or readaptation for the overall project (see Appendix C).
- *Interviews*: Individual interviews with students and mentors will be held in August, December, January, and May to assess the start of the project, goals for start/end of the semester/project, and overall feelings towards success or improvements on the project (see Appendix C).

Literature Review

My literature review's purpose is to address the need for change in the field of computer science education. Students have a misconception of computer science's nature and its greater social significance. Because students are not fully engaged in programming sessions in the early stages of learning, important information is lost. The early studies bore students and they did not feel involved with them because the courses did not link to their inherent drive for pursuing Computer Science. As a result, understanding how people obtain information is critical to closing the literacy and retention gap. Connecting students' goals to their learning goals is another important element that will help them develop their coding abilities. I'll be answering the following questions by using existing literature, which I believe are crucial aspects in implementing PBL in the computer science classroom.

1. Can teaching programming using a PBL (Project Based Learning) method help students master coding techniques and concepts?

2. How was PBL implemented successfully in the computer science classroom, and what areas did not work with PBL implementation in the computer science classroom?
3. What data was pertinent to both success and failure in the implementation of PBL in the computer science classroom?

Through my research, I found that Project-based learning can help students develop systematic approaches to problem-solving and computational thinking, as well as cooperation. However, while project-based learning in computer science benefits students greatly, it necessitates a considerable degree of organization. Students, staff and other interested parties must be on one accord if they are to succeed in implementing PBL. Teaching coding through project-based learning is possible if an implementation strategy is in place, resources are available to support that plan, and there is overall support from any entities, such as school administration and district administration, that can help a generous amount.On many occasions, project ideas offered by students prove to be extremely successful. Most likely as a result of the students' high levels of enthusiasm for the project. Due to the diversion from the traditional curriculum, a project-based learning model may prove to be more complex due to multiple curricular components and state standards.

Plan Implementation

- **Fall**
    - *August*: The learners will be given a rubric that explains the project's aims and expectations for the upcoming year. The focus of student learning will be on mastering ideas aligned with curriculum standards and completing their project design. Learners will also be given a survey to fill out on how they feel at the start of the project and where they expect to be by the end.
    - *September*: Learners will start working on their project concept. For their application development project, they will conduct research and collect data on personal or outside ideas. This information will be gathered with the help of Google or Microsoft Office applications. Learners must store all data to a drive of their choice.
    - *October*: After choosing a project, students will begin putting their learning method into action. During September, this will be produced. Learners will allocate their time wisely to start determining the optimal strategy for developing their projects. They will evaluate themselves by reflecting on each phase of the process and learning their project's programming language.
    - *November*: Learners will continue their learning as we approach the end of the first half of the year.
    - *December*: Learners will examine where they are with their project goals right now. Interviews with students who have shown significant progress will be conducted by teacher mentors. Students will also use Microsoft Office or Google

tools and audio/visual devices to keep track of where they are on their projects. The rubrics will be graded on how much work was achieved in the first half of the year. Finally, for the first half of the project, all data will be merged into a single document, which will be completed in May.

- **Spring**
    - *January*: Learners will re-evaluate their goals in light of their current project progress. Interviews will be performed with students who are currently challenging their projects to see how they can be helped as they progress. Teacher mentors will be given surveys to assess their growth in terms of seeing their learner's progression.
    - *February*: Learners will continue their learning and working as we approach March.
    - *March*: Learners will continue their learning and working as we approach Spring Break. Learners will self-assess their current status on the project as well as provide an update on the estimated completion of their respective projects. Surveys are conducted by both learners and teacher mentors to give a clear outlook as we approach the end of the project.
    - *April*: learners will continue their learning as they approach the conclusion of their project and final implementation of their designs.
    - *May*: all learners will present their final project designs. The rubric will be assessed by teacher mentors as well as professional volunteers who will review the project in its entirety. This is to ensure that teacher bias is not taken into effect

when grading the finished product. Student learners will provide a detailed summarization of their project as well as an essay outlining the challenges as well as successes of completing their project. Student learners may choose to record themselves in a video summarization as well.

Acting Stage

Collect & Analyze the Data (Post-Project)

At the conclusion of the project, student learners and teacher mentors will organize all data collected and created during this time into a single digital folder organized by last name and student ID. The purpose of this is to ensure the validity of the data and its relation to each student learner. The Initial data of project completion and incompletion would have been collected prior to the post-project phase. Any and all video and audio recordings would have been published to the public domain, using YouTube as the medium of publication. All written data should be organized in the student folders for further review; one of the core components that will be reviewed once the data is collected is the completion and incompletion of the project. Additionally, the learner's growth and the completion of the core learning for the academic year will also be assessed. This will determine the answer to whether or not PBL is possible in the computer science classroom.

Developing Stage

Develop the Action Plan

Action Planning Template				
GOAL: This study aims to gauge student performance of learning programming techniques and concepts while engaged in a yearlong project. Can the PBL approach to teaching coding motivate learners to master coding techniques and concepts?				
Action Steps	**Persons Responsible**	**Timeline**	**Resources Needed**	**Evaluation**
Develop Project Rubrics, Assessment tools, Criteria for Recording, Curriculum and TEKS alignment (if applicable), and Timetable	Lead Teacher App Development Team	Summer (June-Aug.)	Meeting Space Microsoft Office Products (Word, PowerPoint, and OneDrive)	Completed Project Rubrics, Assessment tools, Criteria for Recording, Curriculum and TEKS alignment (if applicable), and Timetable
Present Rubric of Project and Survey of Skills to Learners	Lead Teacher	August	Survey Rubric Computer/iPad	Completed Understanding of Rubric Page and Survey.
Begin Project Development Through research	Learners	September	Google or Microsoft Office applications	Completed Outline of Project Proposal and Learning Implementation Plan
Learning Implementation Begins	Learners	October	Google or Microsoft Office applications Learning Management Systems (LMS)	Weekly Summary of Learning Progress Submitted
Learning Continues	Learners	November	Google or Microsoft Office applications	Weekly Summary of Learning

			LMS	Progress Submitted
Learning Continues Recording of Project Update Data Rubric Review	Lead Teacher Learners	December	Google or Microsoft Office applications LMS Media Tools	Weekly Summary of Learning Progress Submitted Progress Interviews The First Half of the Rubric is Grades
Learning Continues	Lead Teacher Learners	January	Google or Microsoft Office applications LMS Media Tools	Updated Goal Sheet for the New Semester Survey of Students and Teachers Weekly Summary of Learning Progress Submitted
Learning Continues	Learners	February	Google or Microsoft Office applications LMS Media Tools	Weekly Summary of Learning Progress Submitted
Project Finalization	Learners	March	Google or Microsoft Office applications LMS Media Tools	Weekly Summary of Learning Progress Submitted Self-Assessment before Spring Break

Project Finalization	Learners	April	Google or Microsoft Office applications LMS Media Tools	Weekly Summary of Learning Progress Submitted Final Self-Assessment as learners approach the end of the project.
Project Finalization	Learners	May	Google or Microsoft Office applications LMS Media Tools	Rubric for Guest Judges Rubric completion for Learners Final Presentation of Projects
Data Review	Lead Teacher App Development Team	Summer (June-July)	Meeting Space Microsoft Office Products (Word, PowerPoint, and OneDrive)	All Data was transcribed into a readable and presentable format.

Reflecting Stage

Share and Communicate Your Results

The findings of my research will be presented in a variety of formats. The first is a written document that organizes all of the information. The second format will be a video/presentation for professional development in which we will demonstrate the effectiveness (or failure) of PBL in the computer science classroom.

Reflect on the Process

Finally, the teacher mentors will reflect on what worked and what did not work over the year as the data is examined. What strategies can be changed? What does not need to be changed? In this area, we will reflect on the improvements that we can make as teacher mentors to improve the quality of projects for our student learners. The final reflection will also include our recommendations for those who plan to implement PBL in their computer science or STEM classroom.

Results

My action research goal is to ascertain a positive relationship between project-based learning, computational thinking skills, and increasing student programming ability and comprehension through my research findings. This study will highlight the benefits of PBL in addition to supporting instructional requirements. In addition, the study's purpose is to use the data and statistics from the first year of implementation to persuade Computer Science, STEM,

CTE, and even core teachers to include programming-based learning environments in their classrooms.

References

Talib, N., Nasir, M. K. M., & Yassin, S. F. M. (2017). *Teaching and Learning Computer Programming Using Gamification and Observation through Action Research*. HRMars. from https://hrmars.com/papers_submitted/3045/Teaching_and_Learning_Computer_Programming_Using_Gamification_and_Observation_through_Action_Research.pdf.

Park, N. 2016. Development of Computer Education Program Using LOGO Programming and Fractals Learning for Enhancing Creativity: Focus on Creative Problem-Solving. International Journal of u- and e- Service, Science and Technology 9(2): 121–126.

Wing, J. 2008. Computational thinking and thinking about computing. Philosophical Transactions of The Royal Society (July): 3717–3725. doi:10.1109/IPDPS.2008.4536091

Appendix A

iOS Mobile Application Development Project - Proposal

Group Name		Computer ID	
Device Name		iOS Platoform	
App Name:			

Project Description:

Our app…

What we propose to do is create an app that will do the following:

- *The system shall …*
- *The system shall …*
- *The system shall …*

We plan to incorporate the following features:

- …
- …
- …

iOS Mobile Application Development Project

Getting Started

You should already have your teams pre-selected at this point

Project Description

Now is the time to let your creativity shine! Using what you have learned and continue learning during this program, create a mobile app of your own purpose and design using iOS!

The domain (i.e. what the app will do) is totally up to you. We find students enjoy projects more when they are invested in what the project does / means for them. So, here's your opportunity to build the app of your dreams!... within reason, of course... as you only have a few months.

And you can't do a Bucket List app or anything that's basically just a list with an info page for each item. You MUST branch out and do something substantively different. Not that you can't have features in this space, but there must be enough to establish that it is a wholly different project from the mini app assignments.

Need ideas? Come talk to Mr. Lindsey during office hours! We can think of something interesting!

Features and Rubric

For the final project, there are a set of required features and points that must be completed by all projects. After that, you may choose any combination of optional features up to the max.

Notes:

- No extra credit given past 100 pts.
- Required Features = 40 pts / Optional Features = 60 pts.
- You cannot do the same optional feature twice for more points.
- Students declare in their docs what they want considered for the optional features.
- We will not grade the "best" set of optional features (i.e. people can't "shotgun" for more points).
- If your feature selection point total is between 60 and 70 points, and removing any features will drop you below 60 points, we will scale your points accordingly during grading (i.e. 65/65 => 60 pts, 63/65 => 58 pts, etc.).
- If your feature selection point total is over 70, you must select features that will not be graded.

Required Features

Points	Description	Ranking
15 pts	At least 3 "screens"/"major features" - We expect your app to be of significant enough complexity that they are not "one screen" apps. "Trivial" screens will not be counted toward this (i.e. splash screen, etc.).	5 per feature – ☐ 5: Good and well-thought out, ☐ 3: Reasonable but lacking, ☐ 1: Poor
5 pts	App can rotate / can be interrupted and works "correctly" given the domain - We expect your app to handle rotation and interruption as a full, professional release would. Games and other required locked-perspective apps do not have to do rotation, but they must make up the points in the optional features.	☐ 5: Works, ☐ 3: Mostly works, ☐ 0: Doesn't work
15 pts	Design and Presentation - We expect your app to be well-designed and polished. This includes everything from following interface guidelines to good layout to good fonts/colors.	☐ 15: Great, ☐ 12: Good, ☐ 9: Reasonable, ☐ 6: Okay, ☐ 3: Poor, ☐ 0: Unusable
5 pts	Documentation - Use the same format as the proposal, with these additional sections added after the wireframe explanation:	☐5: Great ☐ 3: Needs More Information ☐ 1: Poor

		☐ 0: None
Total Points Earned:		
• Platform Justification - What are the benefits to the platform you chose? • Major Features/Screens - Include short descriptions of each (at least 3 of these) • Optional Features - Include specific directions on how to test/demo each feature and declare the exact set that adds up to ~60 pts • Testing Methodologies - What did you do to test the app? • Usage - Include any special info we need to run the app (username/passwords, etc.) • Lessons Learned - What did you learn about mobile development through this process?		

The format can be found here: iOS Mobile Application Development Project - Proposal

Appendix B

iOS Mobile Application Development Project - Update Survey

This survey is to be completed at the end of each quarter of the project.

* Required

1. Evaluation *

   Mark only one oval.

   - ◯ iOS Application
   - ◯ Final Project

2. Your Name *

3. Your Student ID *

4. What areas of learning are working? *

5. What areas of learning are NOT working? *

6. Rate the Progression of Learning Towards Your Project *

Mark only one oval per row.

	Excellent	Could be Better	Work in Progress
Still Learning	◯	◯	◯
I know a lot but not enough to start	◯	◯	◯
Started Programming	◯	◯	◯

7. Personal Reflection of Progress *

8. Overall Rating of Learning Progression *

Mark only one oval.

	0	1	2	3	4	5	
0 - No Effort Given	◯	◯	◯	◯	◯	◯	5 - Excellent - Consistently went above and beyond

iOS Mobile Application Development Project- Partner Evaluation

* Required

1. Evaluation *

Mark only one oval.

- ◯ iOS Application
- ◯ Final Project

2. Your Student ID *

3. Answer the following questions about your partner. *

Mark only one oval per row.

	Never	Rarely	Sometimes	Usually	Always
Did your partner attend group meetings?	◯	◯	◯	◯	◯
Did your partner make a serious effort on the work before/after meetings?	◯	◯	◯	◯	◯
Did your partner communicate with you about their schedule/efforts?	◯	◯	◯	◯	◯
Did your partner contribute to the assignment?	◯	◯	◯	◯	◯
Were your ideas taken seriously by this person?	◯	◯	◯	◯	◯
Did this person contribute ideas to how to complete the assignment?	◯	◯	◯	◯	◯
Did this person put in the appropriate time for this assignment?	◯	◯	◯	◯	◯

4. Assess the technical competency of your partner relative to yourself. *

Mark only one oval.

- ◯ Weaker Than Me
- ◯ About The Same as Me
- ◯ Better Than Me

5. Assess how compatible you and your partner were. *

Mark only one oval.

- ◯ Not At All Compatible
- ◯ Somewhat Compatible
- ◯ Very Compatible

6. Overall Rating *

Mark only one oval.

	0	1	2	3	4	5	6	7	8	
0 - No show	◯	◯	◯	◯	◯	◯	◯	◯	◯	8 - Excellent - Consistently went above and beyond

iOS MAD Project Final Survey

* Required

1. Evaluation *

Mark only one oval.

- ◯ iOS Application
- ◯ Final Project

2. Your Name *

_____

3. Your Student ID *

_____

4. Did you complete your project *

Check all that apply.

- ☐ Yes
- ☐ No
- ☐ Maybe

5. If you answered "No" to the previous question, outline what areas you were not able to complete the project. Identify what limitations affected your ability to complete the project? If "Yes", type N/A. *

_____

_____

_____

_____

_____

6. Personal Reflection of Project *

7. Is there anything you would have liked to have done differently? *

8. What advice can you give to next year's students? *

9. Overall Rating of Your Success in this Project *

Mark only one oval.

	0	1	2	3	4	5	
0 - No Effort Given	◯	◯	◯	◯	◯	◯	5 - Excellent - Consistently went above and beyond

Appendix C

iOS Mobile Application Development Project – Teacher Observation Guidelines

Points	Description	Ranking
15 pts	The learner is making significant progress in project development or project learning.	☐ 15: Great, ☐ 12: Good, ☐ 9: Reasonable, ☐ 6: Okay, ☐ 3: Poor, ☐ 0: Unusable
10 pts	Mini projects have been completed timely and with significant accuracy.	☐ 10: Great ☐ 7: Mostly works, ☐ 5: Needs More Information ☐ 3: Poor ☐ 0: None
15 pts	Design and presentation of mini-projects are to standards.	☐ 15: Great, ☐ 12: Good, ☐ 9: Reasonable, ☐ 6: Okay, ☐ 3: Poor, ☐ 0: Unusable
10 pts	Comments and documentation in code are easily understandable inaccurate.	☐ 10: Great ☐ 7: Mostly works, ☐ 5: Needs More Information ☐ 3: Poor ☐ 0: None
Total Points Earned:		