Applying Industry Time Estimation Techniques on the Computer Science Capstone Project

by

Tristan Lyon

An Honors Capstone

submitted in partial fulfillment of the requirements

for the Honors Certificate

to

The Honors College

of

The University of Alabama in Huntsville

11/8/2023

Honors Capstone Project Director: Mr. Kevin Preston

Tristan Lyon 12/1/2023

Student Date

Kevin Preston 7 Dec 2023

Project Director Date

12/7/23

Department Chair Date

Sean Lane Digitally signed by Sean M Lane
Date: 2023.12.08 10:27:03
-06'00'

Honors College Dean Date

HONORS COLLEGE
THE UNIVERSITY OF ALABAMA IN HUNTSVILLE

Honors College
Frank Franz Hall
+1 (256) 824-6450 (voice)
+1 (256) 824-7339 (fax)
honors@uah.edu

Honors Thesis Copyright Permission

This form must be signed by the student and submitted with the final manuscript.

Tristan Lyon

Student Name (printed)

Tristan Lyon

Student Signature

12/7/2023

Date

Table of Contents

Dedication

To my family and friends who have helped me get this far.

Abstract

The University of Alabama in Huntsville (UAH) Computer Science Senior Design class (CS499) instructors are always evaluating new projects to use in the class. Prior investigations performed comparisons of projects with industry for class projects and time management [1]. Due to lack of student data, this investigation was not able to come up with any clear recommendations with regards to projects. Students need to be able to complete the Senior Design project in a single semester. Course instructors need a method to evaluate a project assignment to ensure it can be completed by teams of 3 or 4 students in a semester.

The purpose of this study is to analyze completed projects, compare estimated effort to actual effort and determine what would be needed to make accurate estimates from proposed projects. One advantage of this study over the previous one is the collection of student records of time spent working on the project.

Introduction

Estimating the amount of time a given project might take for successful completion by a group of individuals is an important issue for the software development industry to solve. This issue extends to the field of teaching software engineering as well. It is important that students are assigned a project that is within scope to be completed in the span of a semester. This poses a unique challenge when compared to the estimation techniques commonly used within industry, as the academic setting often keeps a constant set of projects while swapping out the individuals working on them. As such, data that had been collected from prior semesters of the CS Senior Design project as part of the curriculum was analyzed for use with common industry estimation techniques.

Methodology

Data Analysis

As part of the Senior Design curriculum, students are required to fill and submit team and individual reports. Team reports include information such as meetings, action items, and key decisions. Individual reports contain the amount of hours spent working on the project throughout each given week as well as current and finished assignments. After cleaning the data and normalizing the data format, analysis was performed on the student data. The full results of the analysis are presented as Appendix A. It was found that students would spend on average 5 hours on the project per week, and would generally have 16 action items assigned to them across the span of the semester.

Use Case Points

The use case point method was proposed in 1993 by Gustav Karner [2] as a means of estimating the amount of effort required to see a software project to completion early within the software development cycle. The use case points (UCP) of a project consists of three factors: the unadjusted use case points (UUCP), the technical factor (TCP), and the environmental factor (EF). The UUCP is calculated by taking the sum of each use case and actor within the project after assigning a weight to them based on complexity, as seen in Tables 1 and 2.

Table 1. Use Case Factors

Use Case	Description	W^i
Simple	Use case involves 3 or less possible transactions.	5
Average	Use case involves 3 to 7 possible transactions.	10
Complex	Use case involves more than 7 possible transactions.	15

Table 2. Actor Factors

Actor	Description	W^i
Simple	Actor interacts via an API or other programming interface.	1
Average	Actor interacts via protocol or command line terminal	2
Complex	Actor interacts via a graphical user interface.	3

The TCF is the summation of various factors (Table 3) that add notable complexity to a given project, and can be calculated as such:

$$0.6 + 0.1 * \sum_{i=1}^{13} F^i * W^i$$

where F^i is a number from 0 to 5, where 5 means it is essential and 0 means it is irrelevant. If all factors are 3, the TCF will be ≈ 1.

Table 3. Complexity Factors

F^i	Description	W^i
F^1	Distributed systems	2
F^2	Performance, response or throughput	1
F^3	End user efficiency	1
F^4	Complex internal processing	1
F^5	Reusability	1
F^6	Ease of installation	0.5
F^7	Ease of operation	0.5
F^8	Portability	2
F^9	Changeability	1
F^{10}	Concurrency	1
F^{11}	Security	1
F^{12}	Third Party Access	1
F^{13}	User Training Facilities	1

The EF is calculated in a similar manner to the TCF and instead considers factors that affect individual productivity (Table 4), and is calculated with:

$$1.4 - 0.03 * \sum_{i=1}^{8} F^i * W^i$$

Table 4. Environmental Factors

F^i	Description	W^i
F^1	Familiar with the process	1.5
F^2	Part-time Workers	-1
F^3	Analyst Capabilities	0.5

F^4	Application Experience	0.5
F^5	Object-oriented Programming Experience	1
F^6	Motivation	1
F^7	Difficult Programming Language	-1
F^8	Stable Requirements	2

The final UCP is calculated by multiplying the UUCP, TCF, and EF.

$$UCP = UUCP * TCF * EF$$

The effort a project necessitates can then be estimated by multiplying by the work hours per UCP:

$$Effort = UCP * WH/UCP$$

which Karner estimates to be around 20 hours per UCP [2].

Application

The UUCP and TCP can be calculated as is for each project, but with individual teams swapping every semester, the EF must instead be estimated. From our analysis of student data, the average student works about 5 hours per week directly on the project, and about 7 hours per week on the class as a whole including meetings and class. As such, the factor in which "Part-time workers" is affecting the project was increased positively. Likewise, students are expected to come into the project with prior knowledge of object-oriented programming, and some manner of group experience on a past project. However, there is to be some error within this estimate, the language chosen for the project, the capabilities of the individuals in question, and the motivation of the group cannot be ascertained.

Table 5. Considered Projects

Project Name	Description	Considered Semesters
Drone Telemetry	GUI that displays drone telemetry data as a set of customizable gauges.	Spring 2021, Spring 2022
Robot Vacuum	GUI that simulates a robot vacuum cleaning a house for the purposes of testing pathfinding algorithms.	Fall 2019, Spring 2020, Spring 2021, Spring 2022
Piracy Simulation	GUI that displays a simulation of piracy off of the coast of the Indian Ocean near Somalia.	Fall 2019, Spring 2020, Spring 2021, Spring 2022

Three projects were chosen for analysis with use case points. These projects have all been successfully completed by students, and can serve as a basis to compare an unknown project with. Due to data loss, completed projects were unable to be differentiated from failed projects. To account for this however, only projects that completed over 100 cumulative hours of work on the project were considered. Overall, 10 projects from Mr. Preston's Senior Design courses were considered.

Results

Table 6. Project UCP Application

Project	UUCP	TCF	EF	UCP	Average Hours Worked	Average UCP/work hour
Drone Telemetry	105	0.91	0.92	90.42	312.52	3.46
Robot Vacuum	115	0.88	0.92	95.53	427.83	4.77
Piracy Simulation	110	0.855	0.92	88.89	229.6	2.58

The UUCP and UCP of the three projects were evaluated to be very similar. The work hour/UCP was much lower than Karner's estimate. This indicates that the work hour/UCP may be scaled by another factor, such as the hours worked per week. The variance within work hour/UCP was greater than expected; however, there is an explanation for the increase of the hours worked for Robot Vacuum.

The students of the Spring 2020 semester worked over 600 cumulative hours on the project, noting in their feedback that they had certain 8-hour sessions where all members of the group would work on the project collectively. The increase in hours of the Spring 2021 and Spring 2022 might also be related to their recommendation of this method within their feedback. This shows that work hour/UCP is very susceptible to outliers, as individuals decide to work above what is strictly required. Despite a higher UUCP, the piracy simulation was often completed with notably less effort compared to drone telemetry, implying notable importance of the TCF in the final estimate.

By averaging the work hour/UCP together, it was found that students will complete 1 UCP's worth of work in approximately 3.6 hours. With this, the time required for a successful completion of a given project can be estimated using Karner's equation for effort. To demonstrate this, the UCP of a fourth project, one that tasks students to create a simulation of life in a similar vein to Conway's Game of Life, was calculated. Evaluation of the project showed a UUCP of 95, a TCF of 0.905, and a EF of 0.92, resulting in a UCP of 81.6. Applying Karner's equation, this project should take around 294 work hours to complete. The calculated average for this project is 253.17 hours, indicating an error of about -13.8%. Taking the average work hour/UCP and average hours worked in a week, the average UCP/week can be calculated as approximately 1.3 UCP/week per student. This can then be used to calculate the UCP total an average group of students can complete within a semester. By multiplying the UCP/week by the number of weeks in a semester and number of students in the group, the UCP of a project that can be completed by 4 students in 16 weeks is 83.6.

Conclusion

This study has introduced a method in which prospective projects can be measured and compared against currently used projects in order to estimate the time it would take students to complete. However, there still exists room for improvement. Data loss prevented accurate cleaning of the data to remove failed projects. Only 4 semesters were considered within this analysis, which may not be enough to showcase any patterns within any single project. The use case points method necessitates an expert's judgment, so it may be likely that miscalculations occur. Additionally, there exist variations on the use case points method such as e-UCP and re-UCP [3] that may prove more accurate than UCP. Regardless, the basis that has been created can be used for instructors to evaluate projects to ensure that they can be completed by students within the span of a semester.

References

[1] E. Tucker, "A Comparison Between How Agile Software Development is Practiced in a Corporate Environment Versus an Educational Setting," *Honors Capstone Projects and Theses*, no. 636, Dec. 2018.

[2] G. Karner, "Resource Estimation for Objectory Projects," *Objective Systems SF AB*, vol. 17, no. 1, p. 9, 1993.

[3] M. Manzoor and A. Wahid, "Revised Use Case Point (Re-UCP) Model for Software Effort Estimation," *International Journal of Advanced Computer Science and Applications*, vol. 6, no. 3, 2015, doi: https://doi.org/10.14569/ijacsa.2015.060310.

Appendix A. Student Data Analysis

Table A1. Student Averages

Student averages compiled from the individual reports turned in by students as a part of the Senior Design curriculum. Student Contribution marks hours that are specifically spent on the project, while Student Total includes meetings and class hours.

	Average Student Contribution (hr)	Average Student Total (hr)	Average Assignments Per Student
Fall 2019	3.91	5.96	21.9
Spring 2020	4.79	7.75	25.05
Spring 2021	4.56	6.53	14.21
Spring 2022	5.10	7.04	11.32
Total	4.70	6.76	15.94

Table A2. Team Averages

Team averages compiled from the team reports turned in by students as a part of the Senior Design curriculum.

	Average # of Meetings	Average # of Action Items	Average # of Risks	Average # of Problems
Fall 2019	20	42.33	3	2.67
Spring 2020	26.83	64	3.83	0.67
Spring 2021	24	34.38	4.25	0
Spring 2022	18	19.09	3.18	0.09
Total	21.82	35.57	3.61	0.46

Figure A1. Time Spent Working on Capstone

The average amount of time students spent working on the project each week during the semester.

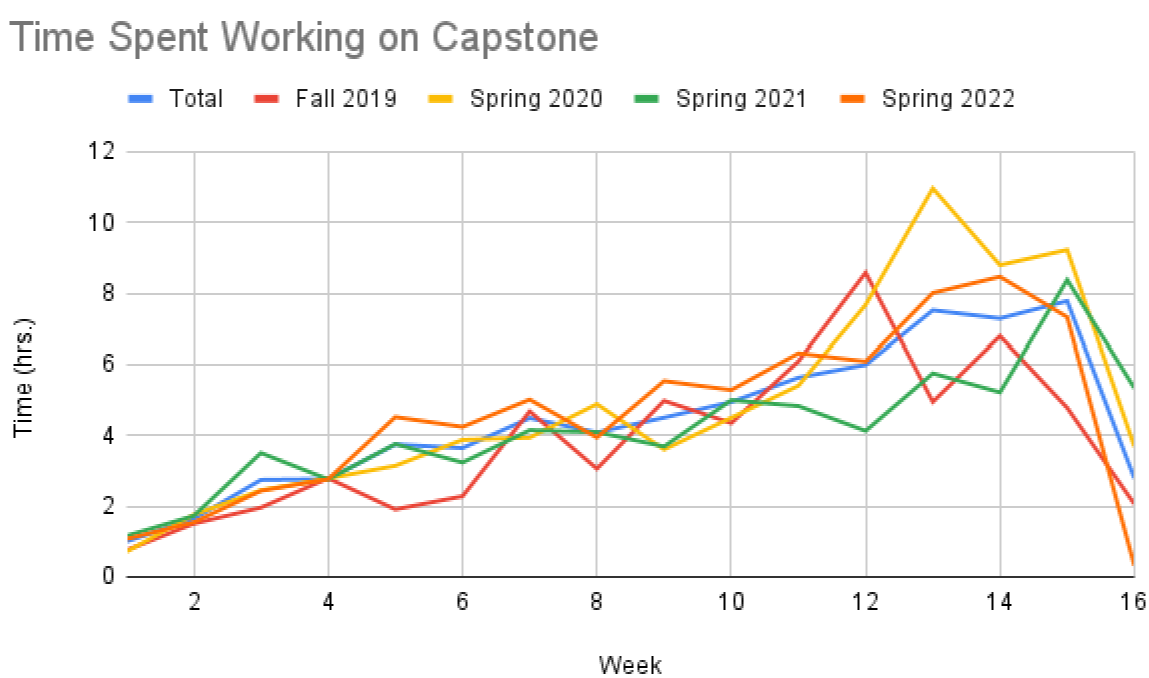

Figure A2. Time Spent Working on Capstone Cumulative

The average cumulative amount of hours students spend on their projects throughout the semester.