Evaluating Active Interventions to Reduce Student Procrastination

Joshua D. Martin

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Computer Science and Applications

Stephen H. Edwards, Chair
Manuel A. Pérez-Quiñones
Clifford A. Shaffer

April 30, 2015
Blacksburg, Virginia

Keywords: procrastination, self-regulation, self-efficacy, scheduling, schedule sheets, reflective writing, active learning, e-mail alerts

Evaluating Active Interventions to Reduce Student Procrastination

Joshua D. Martin

(ABSTRACT)

Procrastination is a pervasive problem in education. In computer science, procrastination and lack of necessary time management skills to complete programming projects are viewed as primary causes of student attrition. The most effective techniques known to reduce procrastination are resource-intensive and do not scale well to large classrooms.

In this thesis, we examine three course interventions designed to both reduce procrastination and be scalable for large classrooms. *Reflective writing assignments* require students to reflect on their time management choices and how these choices impact their classroom performance. *Schedule sheets* force students to plan out their work on an assignment. *E-mail alerts* inform students of their current progress as they work on their projects, and provide ideas on improving their work behavior if their progress is found to be unsatisfactory. We implemented these interventions in a junior-level course on data structures. The study was conducted over two semesters and 330 students agreed to participate in the study. Data collected from these students formed the basis of our analysis of the interventions.

We found a statistically significant relationship between the time a project was completed and the quality of that work, with late work being of lower quality. We also found that the e-mail alert intervention had a statistically significant effect on reducing the number of late submissions. This result occurred despite students responded negatively to the treatment.

Acknowledgments

This work is supported in part by the National Science Foundation under grant DUE-1245334. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

Contents

List of Figures

Chapter 1

Introduction

Procrastination is a ubiquitous problem in education. Many students find it difficult to complete assignments on time. This can cause students to receive poor grades in the classroom, negatively impacting their academic performance. In particular, many computer science educators describe procrastination as a common reason students do not succeed in class [7]. Additionally, the concept of late night debugging sessions and last minute project completion seems to be ingrained in the culture of computer science. Students give many reasons for procrastinating, and many potential causes of procrastination have been identified [22][13]. Researchers have also examined different ways to combat procrastination [25][19]. These techniques have varying degrees of applicability and success. In this paper, we discuss previous research regarding procrastination and introduce our own active interventions to combat procrastination in CS courses. We then analyze the impact of these treatments on student procrastination and other aspects of submitted project solutions.

1.1 Procrastination Impact and Causes

Procrastination can be defined multiple ways, but we define it here as "voluntarily delaying an intended course of action despite expecting to be worse off for the delay" [22]. Even worse than traditional procrastination is the behavior of negative procrastinators. Negative procrastinators are defined as those who procrastinate to the extent they experience negative consequences from their behavior. This class of procrastinator is more likely to turn in work late, receive lower scores, report greater stress and health concerns, and visit health-care professionals more often [23].

The negative impact of procrastination on student performance is directly shown in academic literature. In a study examining the performance of 1,101 CS students over a five year period, a statistically significant correlation was found between when students started working on a project and their work quality. Students who started earlier on assignments were significantly more likely to earn a good grade, in this case an A or B letter grade, compared to assignments where these same students started working later [9].

Despite procrastination having such a profound impact on students, there is still a lack of understanding about the behavior [13]. Research has determined that procrastination should be considered a personality trait [22], and multiple tools have been developed to measure an individual's procrastination tendency [14][24]. A number of potential causes have also been examined, including self-efficacy, self-esteem, self-regulation, fear of failure, task aversion, task rewards, neuroticism, and impulsiveness [22].

1.2 Difficulties with Current Techniques to Fight Procrastination

Previous attempts at combating procrastination were targeted at smaller groups. Tuckman, who utilized a "study skills" course to teach students various theories about procrastination, found the course on average increased a student's GPA compared to a control group of students [26]. His work clearly shows a degree of success, but it is unfeasible to assume that every student who struggles with procrastination will have time to take a course. Indeed, given the large number of undergraduate students who procrastinate on coursework, any solution to reduce procrastination will need the ability to reach a large number of students.

Because the most effective strategies to deal with procrastination involve supplementary course or workshop material on motivation and time management strategies, there remains a distinct gap of scalable techniques to fight this issue.

1.3 Research Problem

In this thesis, *we design and evaluate three different scalable techniques to combat procrastination among students*. These interventions are reflective writing assignments, schedule sheets, and e-mail situational awareness alerts.

The reflective writing assignments were inspired by active learning techniques. Students completed these writing assignments outside of class before a programming assignment was completed. The schedule sheet intervention utilized an online system where students estimated the amount of time required to complete a project. The e-mail alert intervention consisted of an automated notification system which sent students information about their current work progress at specified intervals.

To assist with the implementation of these interventions, we utilized the Web-CAT automated grading platform to integrate data collection about each assignment. Web-CAT provides a number of assignment analysis and data collection tools that proved useful in gathering information about the impact of each intervention. Furthermore, both the schedule sheet and situational awareness interventions were implemented as plugins to the Web-CAT system, allowing both interventions to take advantage of the information collected when providing feedback to students.

By integrating our interventions into Web-CAT, we were able to collect a rich dataset on student submissions. The data include many different aspects of each student solution, such as information on student submission behavior, project grades, final grades, and other characteristics. We examined potential correlations between the results from each of the tested intervention groups, and ran statistical tests to see which of the interventions proved to be the best at changing individual behaviors. Additionally, we examined specific information related to each intervention to understand the individual impacts. This includes feedback collected from a survey given to students. The survey asked students in each of the treatment groups multiple questions regarding their own personal feelings about the interventions. The nature of the treatments themselves also generated extra data, such as problems students had creating appropriate schedules or the common themes students discusses in their reflective writing assignments. We discuss this specific information in the following chapters.

We previously published a cursory analysis of these interventions in the 2015 Conference on Innovation & Technology in Computer Science Education. This initial examination of the data collected found a positive correlation between the e-mail situational awareness alert treatment and the on-time status of a submission [7]. This paper examines the impact of these interventions more extensively and in-depth, building off of the results found in our earlier work.

1.4 Organization of this Document

In Chapter 2, we discuss the literature that influenced our research and provide examples of previous techniques used to combat procrastination in the classroom. This includes courses developed to teach students study skills and workshops developed to provide students with information about procrastination. In Chapter 3, we begin our discussion of the active interventions we explored with an overview of the reflective writing intervention. In Chapter 4, we discuss our creation of a schedule sheet system implemented as a plugin to the Web-CAT grading system. In Chapter 5, we describe the email alert intervention setup, also implemented as a plugin to the Web-CAT system. Chapter 6 evaluates each of our four different intervention groups (the three active alert groups described above and a fourth control group). This evaluation includes examination of multiple time-based factors, such as when a student started submitting work to Web-CAT and how early or late their final submissions were. Chapter 6 also examines the impact of lateness on various project quality indicators. Finally, Chapter 7 summarizes the research performed and discusses potential future research that can be conducted on this subject.

Chapter 2

Literature Review

Procrastination impacts almost everyone, but its effect can be most acutely felt among students [13]. There has been a significant amount of research done on procrastination. More important for this paper, the procrastination research includes studies on methods to combat procrastination, particularly among students in computer science. This prior research was crucial in framing our approach to combat procrastination. But despite all of this work, procrastination is still a relatively poorly understood problem.

2.1 Procrastination Definition

When discussing procrastination the best place to begin is to define the phenomenon. Steel describes two connotations of the term procrastination [22]. One less commonly used description of procrastination describes the behavior as a positive force. By delaying a decision, one can gain additional information to make a more informed decision [3].

However, we focus on the typical definition of procrastination. This is summarized by Steel as "to voluntarily delay an intended course of action despite expecting to be worse off for the delay". This definition ensures procrastination is treated differently from positive decision avoidance, as one is expecting the delay will ultimately harm their outcome, rather than enhance it [22]. Whenever we refer to procrastination in this paper, we mean this definition.

2.2 Historical References to Procrastination

Procrastination has long been identified as a problem in human behavior. In Steel's meta analysis of procrastination, he finds one of the earliest references to procrastination in the writings of Hesiod. Hesiod was one of the first recorded Greek poets, assumed to have been active between 750 B.C. and 650 B.C. His words directly warn against putting off one's work until the next day, indicating procrastination was a problem even in ancient Greece. Steel also cites the famous Roman orator Cicero, who discussed procrastination while speaking against his political rival Marcus Antonius. A slightly more recent reference to procrastination comes from Samuel Johnson, who in 1751 described procrastination as "one of the general weaknesses, which, in spite of the instruction of moralists...prevail to a greater or lesser degree in every mind" [22].

Milgram wrote the first historical analysis on procrastination. He argued that procrastination rose from the numerous commitments and deadlines required by technologically advanced societies. Agrarian societies, being undeveloped, did not suffer so much from procrastination [16]. Ferrari et al. made a similar argument, stating that procrastination was more acute in societies that were industrialized, but that it existed throughout history [10].

2.3 Modern Procrastination Research

While procrastination has been studied since ancient times, it is still relatively poorly understood. Several studies researched various aspects of procrastination, including causes, effects, and ways to prevent it. We summarize these potential causes, theories, and studies in this section.

2.3.1 Causes

Steel describes numerous possible causes for procrastination [22]. One possible cause is related to individual task characteristics. A study by Briody found about 50% of people stating a task characteristic caused their procrastination [6]. Researchers have examined two specific task characteristics and their effect on procrastination: timing of rewards/punishments and aversiveness. Studies have shown that an event is less likely to impact one's decision making the longer away it is in terms of time. As a result, time is often considered a prime cause for procrastination. In addition, the aversiveness of a task also impacts the desire to complete a task. Intuitively, more aversive tasks are less likely to be completed and more likely to be put off.

In addition to individual characteristics of a specific task, individual personality traits also seem to influence procrastination. There has been enough research to determine that tendency to procrastinate is an individual personality trait. Multiple tests have been developed to deduce the level of this trait. One such test, developed by Tuckman, was used in our feedback surveys to examine student procrastination tendency [24]. Additionally, Ellis and Knaus studied low frustration tolerance and its impact on procrastination. Their research suggested that this behavior is a main source of procrastination [18]. Irrational thinking also seems to play a role in procrastination, as research has shown strong correlation between irrational beliefs and procrastination tendency [5].

2.3.2 Theories

The various causes examined to explain procrastination have led to researchers formulating many different theories to explain the behavior. Procrastinating on decisions has been theorized as a way for people to avoid confrontations [11]. One study examined the impact of parental procrastination on student behavior, and found that students tended to procrastinate more than their parents. No concrete reason was given, but a few possibilities such as parents being compelled to present themselves in a more favorable light to their children [15]. A study by Rothblum et al. examined academic procrastination and found that high procrastinators had significantly lower behavioral measures of self control compared with low procrastinators [17].

Perhaps one of the most complete theories of procrastination was proposed by Steel and König. They proposed a comprehensive theory of procrastination known as Temporal Motivation Theory (TMT). TMT attempts to incorporate motivational formulations with time. Specifically, TMT incorporates four different factors to describe one's *utility*, or how desirable a task is for an individual. TMT describes utility as a result of four factors: expectancy (E), value (V), delay (D), and sensitivity to delay (Γ). Taken together, E and V indicate how desirable a task is. Both D and Γ capture how time influences the desire to work on a task. The greater the value of delay, the more likely one is to put off a task, therefore utility shrinks. This is multiplied by one's own individual sensitivity to delay. When combined together, the following formula was defined:

$$Utility = \frac{E \text{ X } V}{\Gamma D}$$

This formula captures how students prefer tasks that are enjoyable and can be completed, while tasks resulting in a far-off benefit are less preferred [22]. Steel demonstrates the validity of TMT by showing each of TMT's four components strongly correlating with procrastination. Our own research is based on this theory of procrastination.

2.4 Previous Efforts to Study Procrastination in the Classroom

There have been several previous studies to examine procrastination's impact in the classroom. These studies inspired some of our mechanisms to measure the reach of procrastination. While many of the studies showed techniques that had some degree of success, they were often infeasible to implement on a large scale.

2.4.1 Strategies for Achievement

Strategies for Achievement is elective course offered by the Ohio State University. The course is designed to work as a behavioral intervention to assist students in developing and implementing study skills and good time management. The course met for 4.5 hours per week for 10 weeks. The course was not a traditional course, but instead was a hybrid mix of on-line and in-class content based on the Active Discovery and Participation through Technology instructional model (ADAPT). Most of the students taking the course were struggling academically. One-third of the students had a GPA under 2.2 (on a 4 point scale) [26].

The results for the course were positive. The improvements in student ability were measured in three different ways against a control group of other students at the university who did not take the course. Three factors were measured: mean GPA of the same term, mean GPA of the same term without the study skills course grade, and GPA of the following term. In each of these terms, there was an increase in GPA compared to the control group [26].

While this technique shows clear success, it is obviously resource intensive. Instructors are needed to run the course and students need extra time in their schedule to take the course. It would seem infeasible to make all students who struggle with procrastination take a course like this, especially when so many students procrastinate.

2.4.2 Task Management Groups

Task management groups are another technique used to reduce procrastination. The technique is targeted at students who are struggling with study tasks like preparing for an exam, and is commonly used at Dutch universities.

Typically, students meet in small groups (12 students) weekly for 1.5 hour long sessions. Groups are supervised by a student counselor. Students in the group split up long term tasks into short term goals which must be completed each week. This training helps students develop the necessary skills for managing long term goals [26].

Student feedback to these study groups is largely positive. Students joined the groups because they were procrastinating (with an average score of 4.3 on a 1 to 5 procrastination scale with 5 being the highest level of procrastination). Most participants left their groups because they felt more confident in their own ability to not procrastinate. However, studies have shown the groups did not eliminate procrastination behavior entirely. The groups seem to offer a way to keep procrastination at bay instead of a more comprehensive way of changing student behavior [26].

2.4.3 Extra Credit Opportunities

Another technique used to reduce student procrastination is to offer extra credit to students for completing an assignment early. For example, computer science courses at Virginia Tech have sometimes given additional points to students who finish their work ahead of the deadline. This is a scalable incentive, making it an ideal way to potentially combat procrastination.

However, a study analyzing the impact of giving extra credit for earlier work showed no significant difference in the time spent on a project between students who were offered extra credit and those who weren't. In fact, a within subjects comparison of students showed students who were not offered extra credit actually submitted their work significantly earlier compared to when they were offered extra credit [1]. This indicates extra credit is not the idea solution we are looking for.

Furthermore, there are ethical concerns regarding the use of extra credit. There is a concern that any extra credit awarded will be disproportionately earned by students who do not need it. This can enlarge the gap between strong or high performing students and struggling students. If this effect occurs, the bonus gained from extra credit can add a systemic bias that further disadvantages struggling students [1]. This bias provides further evidence that offering extra credit is not an ideal solution when attempting to reduce procrastination behavior.

2.4.4 Web-CAT

A concept that influenced our own previous work is systematic data collection. This involves collecting information from students as they develop a project, in order to better analyze their performance and check for signs of procrastination before it becomes a larger problem. In order to implement this collection, we use Web-CAT. Web-CAT is an automatic grading tool that provides a number of assignment analysis and data collection capabilities based on the Eclipse BIRT project [2] [12]. Web-CAT provides students with automatic feedback each time they submit a potential solution to an assignment, which encourages students to submit early and often. Furthermore, Web-CAT enables instructors to grade students based on how they test their code. This enables instructors to allow unlimited submissions for a particular assignment, while still teaching students the importance of code testing. Web-CAT also enables instructors to collect information about student progress throughout the life of an assignment, as each submission for an assignment records information about the time of submission, adherence to required coding standards, presence of required documentation, number of student tests written, number of student tests passed, statement coverage achieved, branch coverage achieved, and the size of the submission [8].

In an earlier experiment, Web-CAT was used to examine student performance. Specifically, Web-CAT data collected from 10 previous semesters was used to analyze student submission behavior. This dataset was split into two groups, program submissions scoring in the A/B range and submissions scoring in the C/D/F range. All students who performed exclusively in the A/B or C/D/F range were removed from the set to isolate average performing students from consistently high or low performers. Using the remaining students, the difference between the first submission time for each group was examined. The A/B group had a significantly earlier average first submission compared to the C/D/F group, with the A/B group submitting 37.7 hours earlier than the other group [9].

This research demonstrates that students who start earlier on an assignment receive earlier grades than when they start later on an assignment. Such a result supports the importance of starting work early, which further emphasizes the impact of student procrastination. If students who start earlier get higher grades than when they start later on assignments, it is safe to say procrastination can directly lead to lower grades on project assignments.

Chapter 3

Reflective Writing

The first intervention that we examined involved reflective writing. The specific intervention is inspired by active learning techniques. Reflective writing provides students with active reflection on their work while engaging students about their own time management behavior and how it effects their performance. Students were given each writing assignment when a programming assignment was issued. Each writing assignment contained four prompts with questions regarding the time management and testing strategies the students would implement. The writing assignments were submitted online using a one page form.

In this chapter, we provide specific information about the reflective writing intervention. This includes a discussion of active learning, how the reflective writing assignments were implemented, experimental details, and specific feedback that we received from students about this intervention. We also examine common themes gathered from each reflective writing assignment response.

3.1 Background

The reflective writing intervention was inspired by the active learning technique called a "minute paper" [4][21]. The one-minute paper (OMP) is a short assignment typically assigned at the end of a class by an instructor. The OMP traditionally asks a student two questions: "What was the most important thing you learned today" and"What question was not answered?" [21]. Students are suppose to finish these papers in a few minutes and instructors collect and review the papers to address concerns in the next class.

OMPs can be modified in a number of ways, such as asking different questions, not grading submissions, or making submissions anonymous. OMPs can also be performed in groups or assigned at different time frames during the class.

Students benefit from an OMP by being given an opportunity to voice potential concerns with what they have learned. If a student feels they missed out on a particular concept, the instructor will quickly learn of this by simply reviewing the OMP responses. For instructors, using OMPs offers an opportunity to examine how effective they were able to get concepts across to students. An OMP is an example of active learning, a technique in which students engage in activities to absorb classroom content. The assignment allows students who are daunted by a large lecture to ask questions directly to the instructor. The instructor also gets feedback on what students feel was the most important lesson from that day's class. But perhaps most critically, an OMP enables students to reflect on what they learned. This is important, as research has shown reflection to be an important component of adult learning.

There is significant evidence that the OMP has a positive impact on student work in the classroom. Surveys of teachers indicate an OMP was viewed as a valuable mechanism to obtain timely feedback from students regarding their educational experience. Studies of student groups who completed an OMP versus those who didn't showed that regular completion of OMPs was associated with a statistically significant gain on essay quiz performance [21].

3.2 Implementation

Instead of issuing a paper with a single prompt, the prompt was expanded to four questions based on feedback from the instructors. Instead of giving the writing assignment in class, students were instructed to complete their reflective writing assignments on-line, via a Qualtrics survey. The surveys were collected shortly after each project's due date had passed. The questions on each survey asked students to reflect on the time strategies implemented in their previous programming assignments and strategies they would implement for their next assignment. In the case of the first assignment, where students had yet to work on an assignment for the course, the response was focused on the last significant programming project completed in a previous semester. This led to the intervention being inspired by the original use of an OMP, but it limited the "think/pair/share" concept that underpins the traditional OMP implementation, as students were not necessarily in close proximity when the responses were written. However, this did allow the reflective writing responses to be easily input by the student and collected by the instructor, as opposed to traditional, written responses.

The four prompts listed for all of the reflective writing assignments can be found in Appendix B.

3.3 Experimental Design

To analyze the responses that we collected from the Qualtrics survey, we performed qualitative analysis on the data. This entailed reading each of the survey responses and finding common themes in the responses. These themes were used as codes to determine the most common response content, and inform us about the typical impact of the interventions on students.

3.4 Student Response

3.4.1 Student Feedback

To assess student feedback on this treatment, we issued a survey to participants at the end of the semester. The survey questions were on a five point Likert scale, with 1 indicating the student “Strongly Disagreed” with the question, a 3 indicated the student had a “Neutral” opinion to the question, while a 5 indicated the student “Strongly Agreed” with the question. For some questions, a higher response score indicated the students viewed the treatment positively. For other questions, a higher score meant the student viewed the treatment negatively. We therefore used a normalized scale factoring in responses to all questions. In this scale, a number closer to 1 indicates a student was mostly pleased with the treatment, while a number closer to 0 indicates a student was mostly displeased with the treatment. The same survey was issued to all treatment groups after the semester finished.

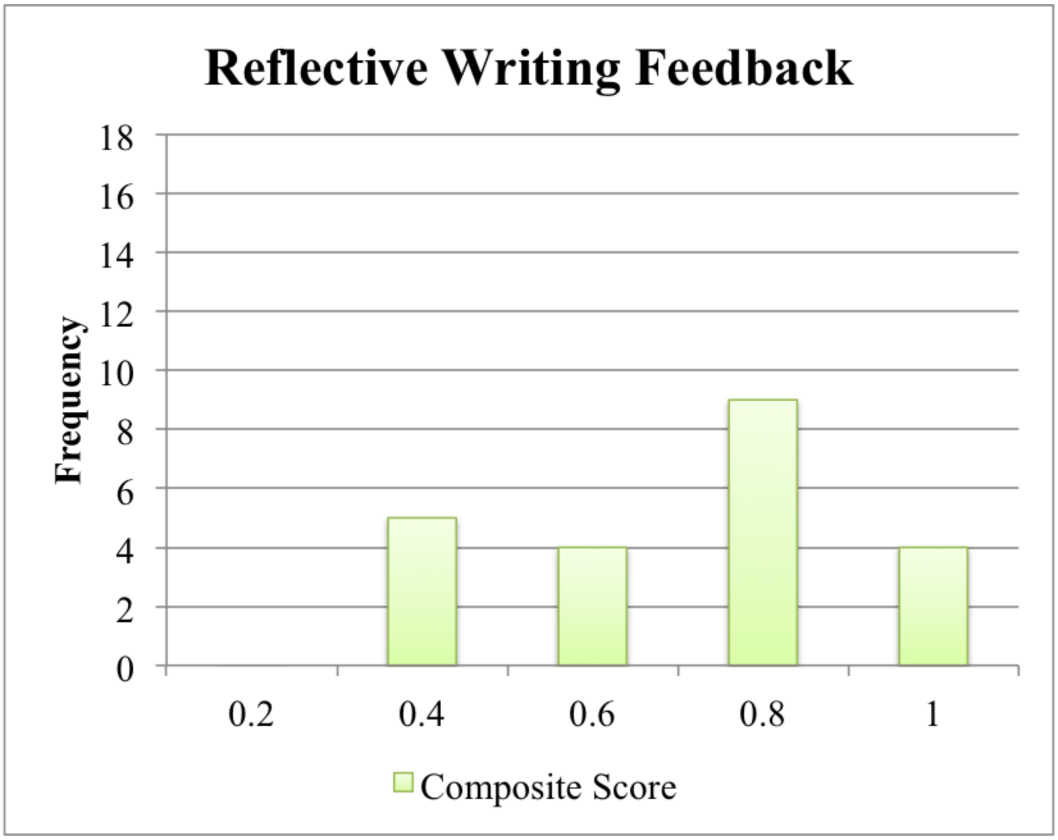


Figure 3.1: The composite score responses to the feedback survey for the reflective writing treatment. Values closer to 1 indicate greater appreciation for the treatment, while values closer to 0 indicate the opposite.

Figure 3.1 shows the composite score for each survey responder in the reflective writing treatment group. The mean of these scores is 0.6, indicating that students in this group had a neutral opinion of the treatment. Analyzing the responses, students seem to agree that the treatment helped them manage their own projects better, with around 59% of respondents agreeing or strongly agreeing with that question. Students also seemed to disagree with the idea that the intervention was a waste of time, with 59% of students disagreeing or strongly disagreeing with that question. Finally, when students were asked if the intervention caused them to start at least one assignment sooner, 69% of students either agreed or strongly agreed. This seems to indicate that students valued the impact of the reflection on their work.

3.4.2 Reflective Response Themes

Analyzing the responses each student submitted for the reflective writing responses, we find several common themes in each of the questions. We list each theme for the questions below.

Note that each response could contain multiple themes.

The themes for question one and question three were nearly identical, as each theme related to time management strategies used by the students. We list these themes below:

- *Early Start* - Responses with this theme indicated a key element of strategy was to start early on an assignment. The actual time varied from student to student, but each response focused on the importance of more time to complete an assignment. (64%)
- *Teamwork* - Responses with this theme emphasized working with teammates to complete a project. This includes good teamwork communication, managing code using version control, pair-programming, and other techniques. (18%)
- *Design/Plan/Schedule* - Responses with this theme described the importance of planning one's work ahead of time. This included the use of schedules to set goals, or designing a solution before beginning to code. (18%)
- *Visit TA or Office Hours* - Responses with this theme emphasized the importance of asking the course TA questions and visiting office hours to talk with the course instructor. (16%)
- *No Plan* - Responses with this theme did not use a time management plan. Students frequently complained of rushing at the end of a project and suffering from poor performance due to procrastination, although a few students indicated the lack of time management did not matter. (9%)
- *Code and Test* - Responses with this theme reported writing a small amount of code and then immediately testing it. Some responses reported using test driven development instead. (8%)
- *Split up work* - Responses with this theme described splitting up work into smaller chunks to make working on a large assignment easier. (7%)

Question two asked students how effective their testing strategy was for the previous project. We found the most common themes in the responses to this question and list them below:

- *Effective Strategy* - These responses indicated students felt their time management strategy was effective in producing a solution. (60%)
- *Good Quality* - Responses in this category indicated their strategy yielded a solution with good code quality. (50%)
- *In-effective Strategy* - These responses indicated students felt their time management strategy was in-effective in producing a solution. (37%)

- *Poor Quality* - These responses indicated the time management strategy used yielded solutions with poor code quality. (27%)
- *Lack of Stress* - These responses indicated the time management strategy used led to a less stressful project experience. (19%)
- *Hardship* - These responses indicated the time management strategy used led students to face significant hardship or stress while working on a project. (13%)

Students responded to question four by stating what testing strategy they planned to use on the upcoming project. We found the most common themes in these responses and list them below:

- *Iterative Testing* - Responses with this theme emphasized a "test as you code" strategy. (54%)
- *Test Earlier and More* - Responses with this theme emphasized testing earlier in the development cycle and writing more tests overall. (27%)
- *Unit Testing* - Responses with this theme explicitly indicated the importance of unit testing in their testing strategy. (12%)
- *No Plan* - Responses with this theme emphasized a lack of test plan. Some responses indicated brute forcing a test by calling functions with different inputs manually. Others discussed writing tests after finishing the project code. (6%)
- *Teamwork* - Responses with this theme place emphasis on testing with team members. This included a variety of strategies such as separating development and testing, or having each team member write separate tests for different units of code. (6%)

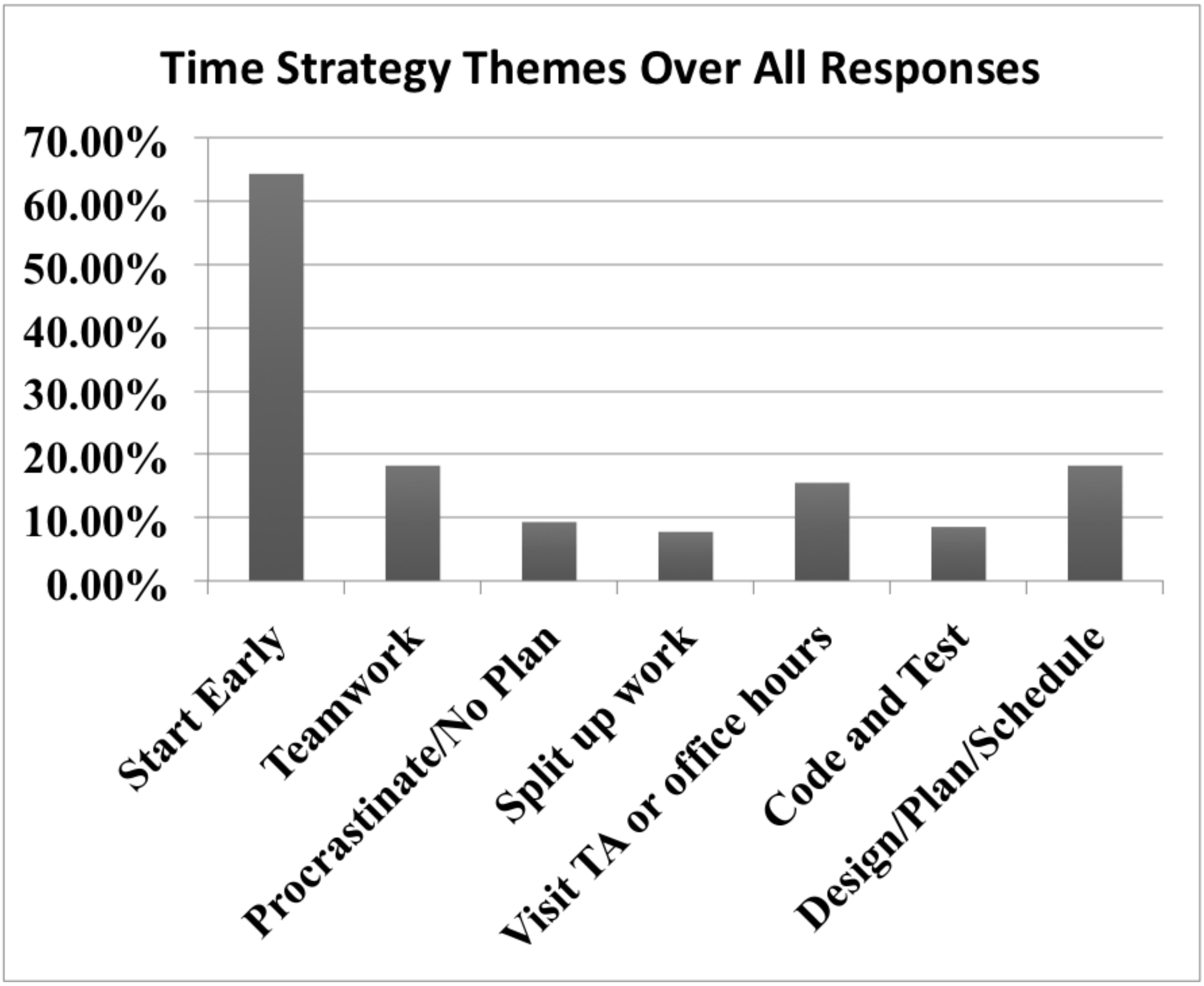


Figure 3.2: The breakdown of responses related to time management strategy combined for question 1 and 3.

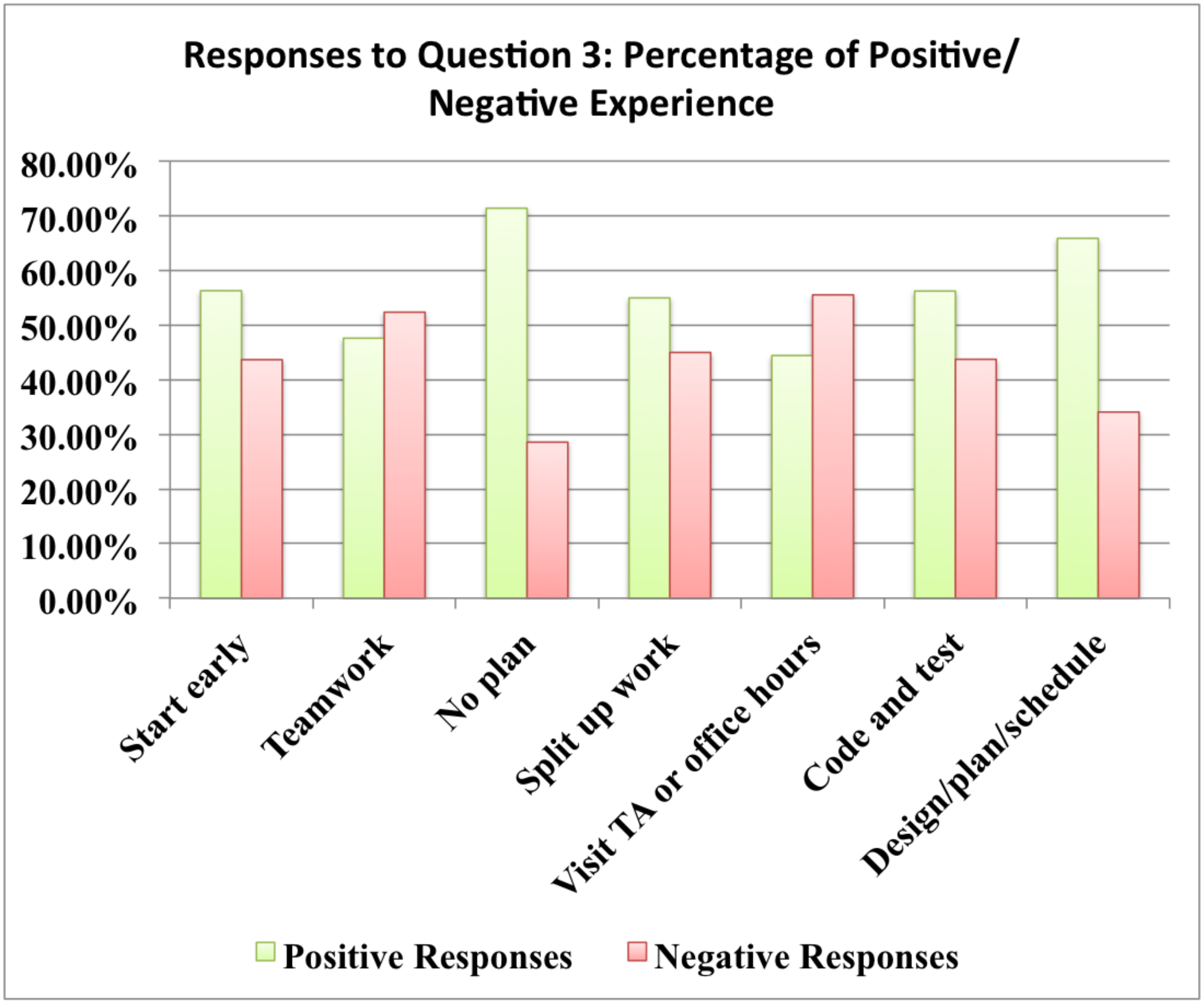


Figure 3.3: The breakdown of responses to question 3.

Examining these numbers, we seem to find that students understand good time management techniques, but typically fail to incorporate these skills when working on programming projects. For example, many students reported understanding the value of starting their work early, with 64% of all responses across both question 1 and 3 mentioning this theme. However, analyzing just question 3 (asking students their strategy for the next assignment), we find a large number of students mentioning they plan to start early because they had a negative experience while working on their last assignment. Furthermore, in 27% of responses for question 1 students indicated they procrastinated on their previous assignment. This contrasts with the 2% of responses for question 3 that indicated students planned to start late on their work for the next assignment.

Figure 3.3 shows the breakdown of responses related to what time management strategy students mentioned for question 3. This graph shows what percentage of responses mentioned a strategy because a student had a positive experience with their previous project, or which students are planing to change to a new strategy because their previous strategy failed. "Positive" responses indicate the student is planning to use the time management strategy for the next project because it worked well form the on the previous one. "Negative" responses indicate students are planning to use the strategy because they had a negative experience on the last project and are trying to change their behavior

Analysis of the concrete impact of this intervention (based on when students began or finished their work) can be found in the evaluation chapter.

Chapter 4

Schedule Sheets

Our second active intervention uses schedule sheets. The use of schedule sheets in computer science classrooms is not new. Virginia Tech's own computer science course instructors have previously used schedule sheets in our advanced data structures course. The schedules were based on Spolsky's "painless schedule" sheets [20]. Survey data collected to measure student responses to the schedules indicate feelings toward the technique are mixed. The purpose of these sheets is to assist in the development of time management skills, forcing students to consider how they will break a large project up into smaller pieces. It also encourages students to self-monitor their own progress and track their pace throughout a semester.

4.1 Background

The concept behind using schedule sheets in a course is to break down each programming assignment into different tasks, with each task representing some functionality required to make the project work. In a previous experiment using schedules, students were given leeway with the of creation of the tasks and sub-tasks that made up the entire schedule. Students were required to create the schedules in mind with their partner, as pair programming was also necessitated for the course. Therefore, other columns on the sheets were dedicated to indicating which student in the pair would be responsible for what task. Additional columns included the self-imposed deadline for the task, the expected amount of time the task would take, the time spent so far on the task, and the expected time remaining on the task [19].

This experiment examined multiple factors examining how the use of schedule sheets impacted the time management of students. The research found students who turned in all of their schedules achieved a significantly higher score compared to students who omitted their schedules. Furthermore, the students who scored below the class mean put in less than 50% of their initial work before the final week of the project, while the vast majority of students who scored above the class mean put in more than 37% of the initial work before the final week. While these results seemed to indicate schedules assisted in making students manage their time better, the sheets did not have any significant impact on overall class performance [19].

4.2 Implementation

Our own implementation for schedule sheets was designed to be easily scalable across any size classroom. In order to effectively administer the schedules and accomplish this task, we designed and implemented an electronic schedule sheet entry system for students to submit their schedules online. There are a number of advantages to integrating the schedules into an electronic system. The first is convenience for the student. Submitting electronically means students can always view their own schedules without the need to make a physical copy. Students are also able to easily adjust their schedules to accommodate plan changes. Another advantage of an electronic schedule sheet system is immediate feedback. Schedules submitted online can provide students with feedback on the feasibility of their schedules. This helps ensure that students create realistic schedules and do not work on the basis of a flawed time plan.

In order to provide students with accurate schedule feedback, we implemented our schedule sheet system as a Web-CAT plugin. Because Web-CAT stores information about each assignment and project submission, our schedule sheet plugin can view deadlines, examine expected amounts of work versus a student's scheduled amount of work, and other factors.

The schedule sheet plugin works similarly to the previous paper schedule sheets. Students list the components or features that they expect their solution will contain. Each component or feature has three separate activities associated with its implementation: design, code, test. Students list the estimated time to complete these activities as well as their anticipated deadlines. After filling in all requisite information, the students can check their work by pressing a button. This begins an analysis by the system, which performs sanity checks on the student's proposed schedule. All of the checks are designed to inform the student of the feasibility of their schedule. This includes determining if a student has scheduled more work that is reasonable between the current date and their proposed deadline for each component. Additional checks examine if a schedule has either too few or too many components and if the schedule has all expected components completed before the project deadline. Once a student is finished modifying the schedule, they submit it through the system and a final validation check is performed. The entire instructions for creating and modifying these schedules can be found in Appendix C.

On subsequent schedule sheets, students are asked about the actual time spent on each activity for the components and if the activity is finished. They then input new estimates for time spent on each activity and new personal deadlines. The sanity checks performed on the first schedule sheet submission are then repeated on the modified schedule. When the student finally turns in a project solution, they subsequently turn in a final schedule representing the actual amount of time worked on a given project.

The major benefit to this new implementation of the schedule sheet intervention is the additional data generated by each schedule sheet submission. Whenever the sanity checks are run on a potential schedule, information about that schedule is saved, including any errors or warnings generated by the checks. This provides valuable data to instructors regarding the effectiveness of the intervention. By providing immediate feedback on schedules, students can modify their work habits immediately instead of after receiving their project grade.

4.3 Experimental Design

In our experiment with schedule sheets, we assigned one section of CS3114 at Virginia Tech to serve as the experimental section for this intervention during Fall 2014. Each project in this course lasted for approximately four weeks from the initial project handout to the final submission deadline. The students were required to fill out an initial schedule sheet shortly after each project was first assigned. Typically this schedule sheet was due within a week of the project being assigned. Students also turned in an intermediate schedule sheet, typically a week before the project was due. As discussed in the previous chapter, this schedule sheet assignment allows students to update their estimates and pieces of work. A final schedule sheet is required after the project has been turned in to list the actual hours worked and final components finished.

The intent with these schedule sheets is to assist students with managing deadlines. By utilizing schedule sheets, students will form, express, manage, and track smaller-scale deadlines themselves. Based on the TMT theory of procrastination, tasks that have a deadline due later will make that task less attractive for one to put effort in. This helps to eliminate the far off deadline research has shown to negatively impact student will to complete a task [22]. The schedule sheet system will, ideally, provide students with a visualized task breakdown to see a project as a set of individual tasks rather than a monolithic unit.

In order to examine if this intervention worked, we need to gather data on how students used the sheets and their work behavior over the semester. Collecting data from these sheets was relatively simple due to the design of the system. When students pressed the "check my work" button to receive immediate feedback, the data collection saved off information regarding the warnings or errors students received. This information was invaluable because it enabled analysis of the specific intervention in addition to the overall impact on lateness and performance. Gathering information on errors and warnings related to the schedule gives us the ability to see how students were trying to use schedules, not just the lateness of individual tasks.

The design of the schedule sheet system meant that schedule accuracy was limited to information that the students volunteered. The instructors noticed early on that some students tried to abuse the schedules by inputting incorrect schedule information or schedules that obviously did not reflect reality. In an attempt to mitigate this, we updated our sanity checks to include additional verification regarding schedules. Ultimately, students were required to self-report the time spent on each component of their project. This information may or may not be accurate, and there is little that an instructor can do to verify it.

4.4 Student Response

4.4.1 Student Feedback

For assessment of student feedback, we used the exact same survey given to the reflective writing treatment group to gain insight into the student opinion of the treatment.

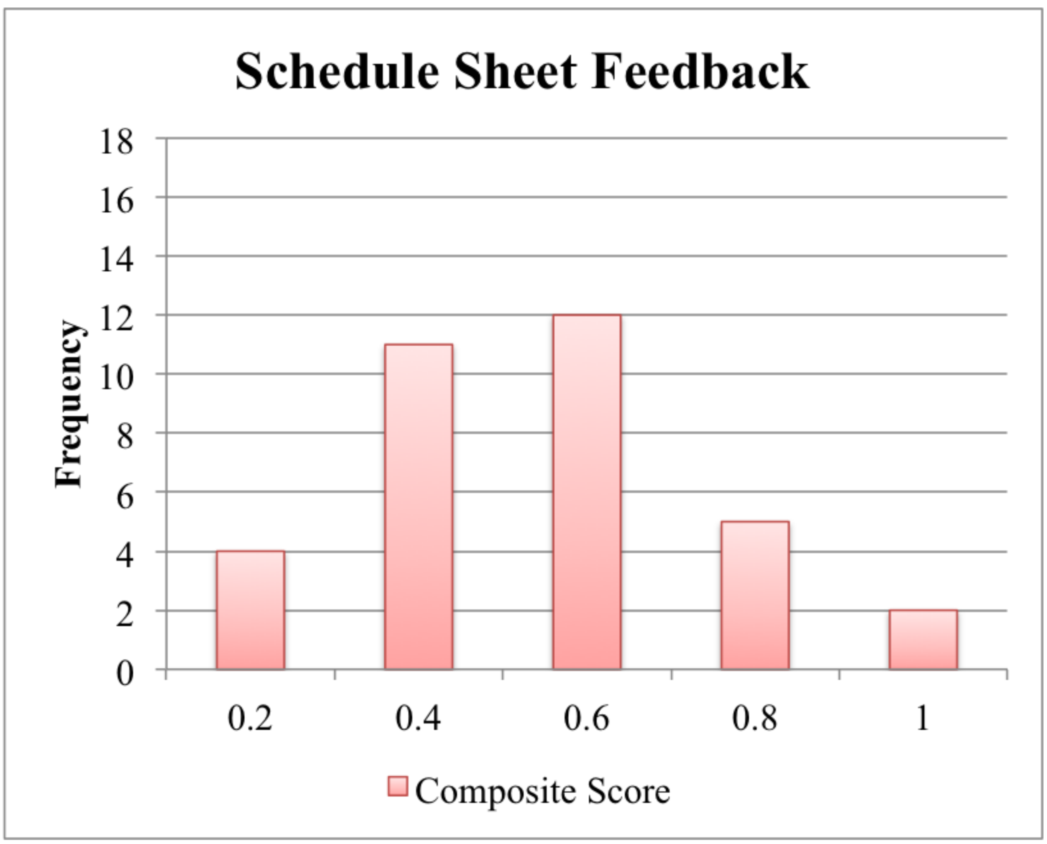


Figure 4.1: The composite score responses to the feedback survey for the schedule sheet treatment. Values closer to 1 indicate greater appreciation for the treatment, while values closer to 0 indicate the opposite.

This figure shows the composite score for the survey responses in the schedule sheet treatment group. The mean composite score for respondents was 0.45, indicating a negative response to the treatment by students.

Around 55% of respondents disagreed or strongly disagreed that the intervention helped them manage their own projects better. Additionally, 55% of respondents agreed or strongly agreed that the intervention was a waste of time. Finally, 48% of respondents disagreed or strongly disagreed that the treatment caused them to start their next assignment earlier. The survey data indicates that, while some students received benefit from the intervention, others felt no assistance from it.

4.4.2 Schedule Accuracy

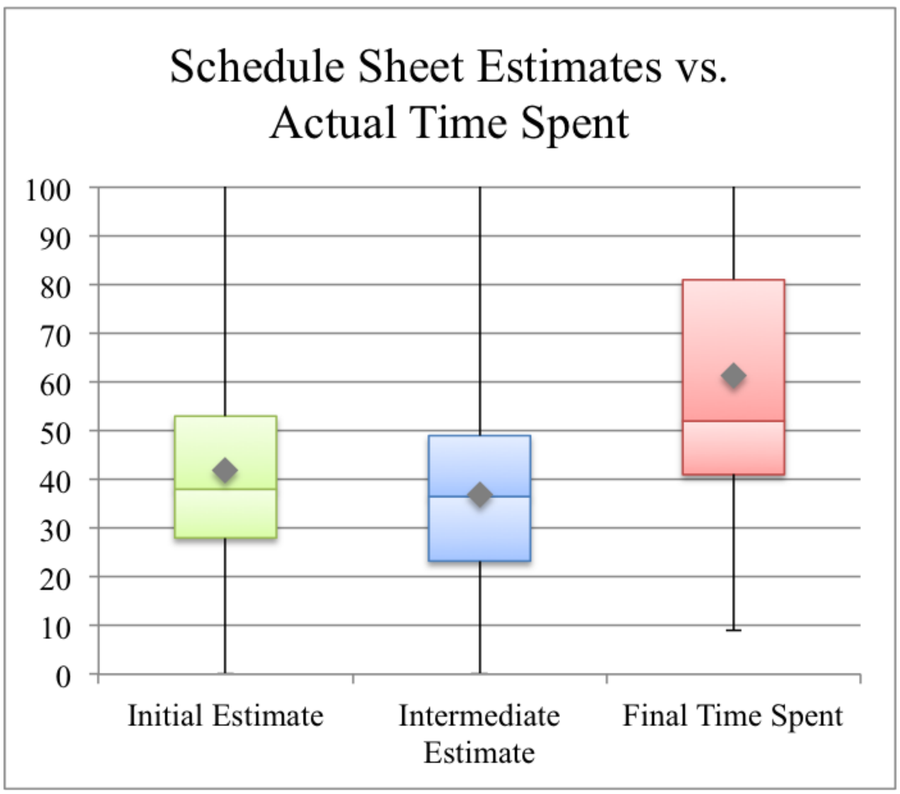


Figure 4.2: The scheduled time across all projects (in hours) compared to the final time actually recorded.

While each individual project differed in complexity, the average initial estimate for the amount of work required to complete a project ranged from 35 hours to 54 hours. On average, students underestimated the amount of work required for a project, with 72% of students facing this problem. Oddly, instead of improving schedule accuracy with the intermediate schedule, students typically increased their underestimation. This meant final time reported was, on average, much greater than students anticipated. These results are visualized in Figure 4.2.

Examining the impact of the accuracy of the final submitted sheet on the time students submitted their projects to Web-CAT, we find a significant relationship between these factors (F ratio = 17.79, P <0.0001). Students who underestimated their work typically submitted their solutions later, while students who overestimated their work typically submitted their solutions earlier.

Ideally, students would get better at making schedules as they progressed throughout the semester. This would indicate students learned how to create more accurate schedule by learning from their mistakes on earlier schedules. We did find the accuracy of a schedule was significantly related to the individual assignment the schedule was created for (F ratio = 10.41, P <0.0001). Students typically had the largest underestimation of work for the first project, followed by a slightly smaller underestimation for the second project. The rest of the projects had smaller underestimations. While this would initially seem to support the idea students created better schedules as the semester went on, this analysis did not account for the differences in project difficulty. One would expect certain projects would be more difficult for students to accurately estimate the amount of work required compared to others.

4.4.3 Schedule Errors

Despite extensive error checking logic, only two types of errors were recorded for the schedules students submitted. The first error notified students that an updated personal deadline was needed for an incomplete activity. The other notified students a component or feature was too large, indicating the student had not appropriately split up their work. The distribution of the number errors was extremely uneven. The median number of errors for the first error recorded was 0 (compared with a mean of 2), and the number required to reach the 90% mark was 6. For the second error type, the median number of errors was 2 (with the mean being 3). The mean number of errors for the sheets saw a significant drop off for the final submission, with mean number dropping to less than 1 for the first error type and the mean equaling 1 for the second error type.

Chapter 5

E-mail Alerts

Our third intervention used e-mail alerts to automatically send an alert to students at predetermined times before a project is due. Each alert is custom generated for the intended student (or student pair if two students are working on a project jointly), and contains information regarding what progress had been made on the project. The alert also contains generic feedback based on the assessed progress, including if a student is at risk of poor performance on the project due to procrastination.

5.1 Background

Previous research has shown that instructors admonishing their students to start programming projects early and spread their workload throughout the project time frame has little impact on student behavior. The limitations of instructor advice helped to inspire our unique approach to project feedback. Our goal with the active intervention system was to provide better feedback directly to students through the life cycle of a project. By giving students accurate and up-to-date information about the shortcomings of their effort at a given time, we hoped to overcome the inherent limits an instructor has when advising students to complete work early. It is infeasible for an instructor in a class of 70 - 90 students to give personalized advice for all students.

As discussed previously, procrastination is ultimately a failure of self-regulation. By providing students with up-to-date information regarding their efforts on a particular programming project, we hoped that student attention towards self-regulation would increase. Students receiving these alerts would immediately understand the impact of their choices on their project grades, as each alert would specify effort, or lack there of, on particular elements of solutions submitted so far.

Furthermore, we hope that the alerts would enable students to contextualize their own progress on a project with the progress expected from them. It can be difficult for computer science students to know exactly how much effort is required to make good progress towards a project solution. Providing alerts that explain to students the effort required to make such progress will enable students to see where their performance fits into the larger goal of project completion. It is reasonable to assume this new-found contextualization of project progress will also increase a student's expectation of successfully completing a task. Steel's TMT suggests this approach will reduce the tendency of a student to procrastinate on a task.

5.2 Implementation

In order to implement the e-mail alert intervention effectively, we needed accurate and up-to-date information about a student's latest submission for a project. By using Web-CAT, we were able to collect this information with relative ease. Each submission to Web-CAT provides an instructor with a wealth of information. This includes lines of code written, unit test coverage, style adherence, and other facts. Therefore we integrated the intervention into Web-CAT to take advantage of this data when evaluating the system.

The first step to generate an email alert comes from a static analysis of the Web-CAT submission itself. Based upon this submission, we analyzed four different factors of student progress. These factors were the amount of code written (relative to a specific target size), the proportion of instructor written reference tests passed, the degree of testing performed, and the number of static analysis checks failed.

Based upon these values, a student's code was ranked according to a five point internal scale. A "Good" rating indicated that the student was making sufficient progress towards a viable project solution. A "Neutral" rating indicated the student was performing to the baseline expectations for the amount of time remaining. A "Bad" rating indicated the student was not making sufficient progress towards a working solution given the amount of time remaining on the project. A "Poor" rating indicated a student was making almost no progress, perhaps making only a few submissions to Web-CAT before the alert was sent. An "Undefined" rating indicated a student had yet to make any project submissions.

These ratings were only used for internal logic to generate a custom email alert. There were two options for the e-mail subject header, depending on the students progress. If student progress was deemed adequate overall, the subject line would be phrased as: "CS 3114: Your progress on Project 2". If progress was not deemed satisfactory, the subject line would instead be phrased as: "CS 3114: You may be at risk on Project 2". Finally, a student who did not make satisfactory progress and was very close to the deadline would receive the following message: "CS 3114: You are at risk on Project 2". These subject lines were designed to strike a balance between drawing students to read and notice the alerts while simultaneously avoiding needlessly scaring them.

The body of each message was tailored based on the ratings assigned during the analysis phase. Examples of alerts received by students are included in Appendix A.

5.3 Experimental Design

Collecting data from the email alerts was a simple process because of the alert integration into Web-CAT. Each time an alert was sent, it was tied to a particular Web-CAT submission from the student or student pair working on the project. If a student (or student pair) did not submit any work at the time the alert was sent, the submission object would be null.

Each submission object contains a wide variety of information related to the work performed on a project. This information was not only used to generate each alert, but it was also used as data to discover if this intervention positively impacts student procrastination. Data we collected from the submissions showed when students made their first submission, how much code was contained in that submission, and if students submitted earlier as the semester went along. We examine such data in the next chapter.

5.4 Student Response

5.4.1 Student Feedback

To assess student feedback for the treatment, we issued the same survey given to the other two treatment groups at the end of the semester. Students were asked to answer each question on a five point Likert scale, with one corresponding to "Strongly Disagree" and five corresponding to "Strongly Agree". To make statistical analysis of the survey data easier, we used a normalized scale to indicate how positive or negative a student viewed the treatment. A score closer 1.0 indicates a student viewed the treatment positively, while a score closer to 0 indicates a student viewed the treatment negatively.

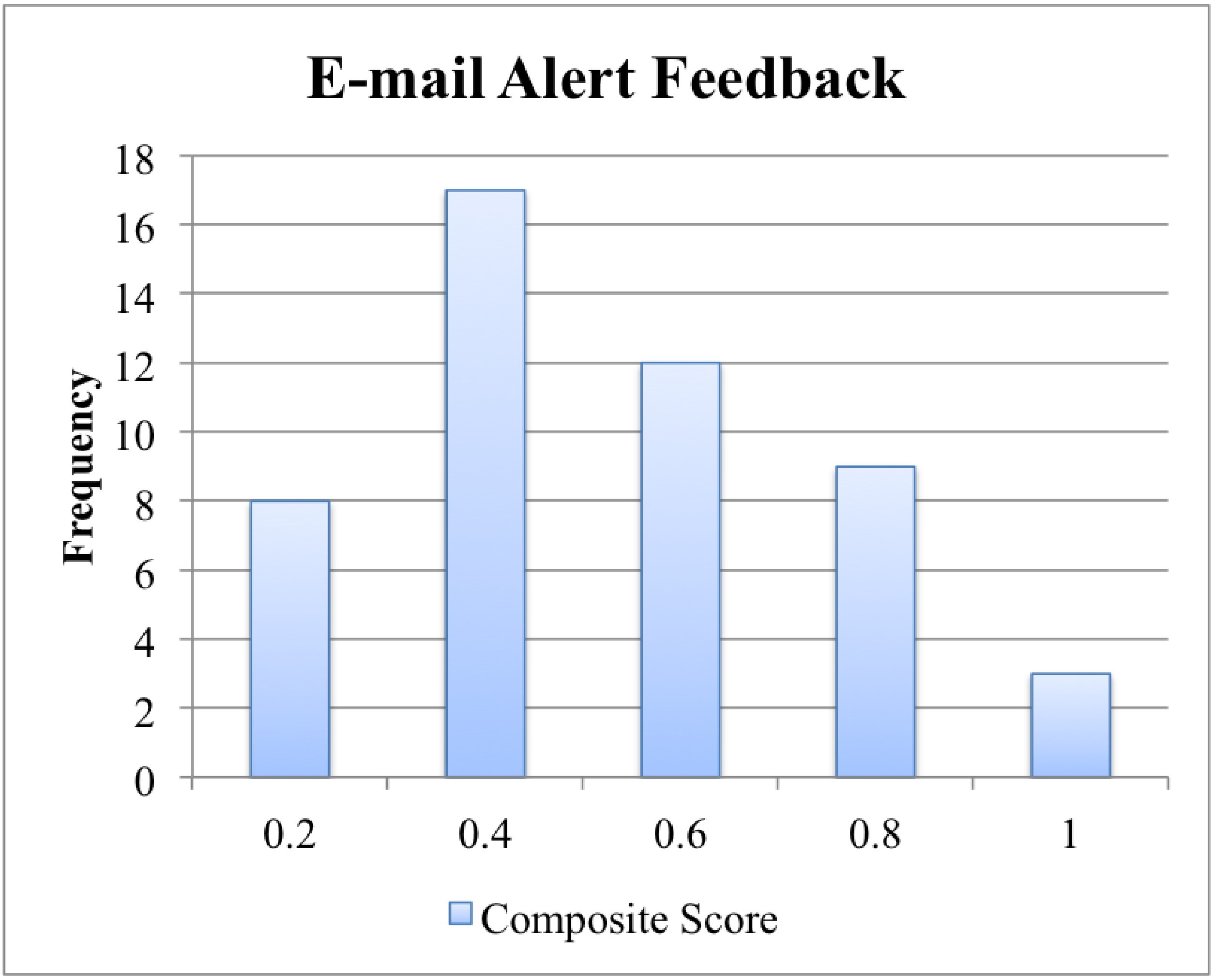


Figure 5.1: The composite score responses to the feedback survey for the e-mail alert treatment. Values closer to 1 indicate greater appreciation for the treatment, while values closer to 0 indicate the opposite.

Figure 5.1 shows the composite score for each survey response in the e-mail alert treatment. The mean composite score was 0.41, indicating a negative response by students for this treatment.

Overall, student responses to the e-mail alert treatment were negative. 56% of respondents indicated they disagreed or strongly disagreed that the intervention helped them manage their own projects better. Additionally, 62% of respondents agreed or strongly agreed the e-mail alert intervention was a waste of time. Finally, 44% of respondents agreed or strongly agreed that the intervention caused them to start their next assignment earlier.

5.4.2 Alert Analysis

Students who received the e-mail alerts typically received a set of three alerts: one seven days before the assignment due date, one four days before the due date, and one two days before the due date. Two of the assignments had an additional alert sent out ten days before the project was due.

The majority of the e-mails sent out contained the "form letter" response for an undefined submission because the students had yet to submit their work to Web-CAT. In fact, only 6–18% of students had made a Web-CAT submission one week before the project deadline. Four days before the deadline, 17–33% of students made a submission. That number grew to almost half of students by two days before the deadline.

We found that students who had made at least one Web-CAT submission when there were four days before the deadline had a significantly higher project grade than students who did not have a Web-CAT submission by that point (F ratio = 12.9, P = 0.0004). This grade comparison did not include any extra credit incentives for those who submitted early.

Chapter 6

Evaluation

To analyze the data that we collected, we used a variety of statistical tests and comparisons to examine multiple potential intervention impacts. Our primary purpose was to examine if any of the interventions positively impacted the timelines of student work. Our hypothesis was students in the treatment groups would show less late submissions and more earlier submissions compared to the control group.

Additionally, we examined the impact of lateness on different dimensions of project submissions. Prior research indicates late work often receives lower scores across these dimensions [9], and we wanted to confirm this link existed in our own study.

To define lateness, we split each final submission into three categories reflecting the on-time status of that project submission. Projects submitted more than 24 hours before the deadline were marked as “early”, projects submitted within 24 hours of the deadline were marked as “on-time”, while projects submitted after the deadline were marked as “late”.

Our dataset consisted of 330 students across four sections of a junior-level data structures course. These sections were split between Fall 2013 and Fall 2014, with the control and reflective writing groups falling in the former while the schedule sheet and e-mail alert groups fell into the later. The students consented to us using their data in the study. Each treatment group completed four different programming projects. The projects had varying degrees of difficulty, with projects 1 and 3 being somewhat less complex, while projects 2 and 4 were somewhat more complex. All of the treatment groups were given the exact same assignments within the same semester, but between semesters the assignments were different. However, they were still designed to be of comparable difficulty. The same instructor taught all of the treatment groups.

For all the projects offered in each section, students were offered a 10% extra credit bonus for completing their assignments at least 24 hours before the project deadline. There was no corresponding penalty for late work, but students were given a “time bank” of late days to self grant a 1 day extension on an assignment deadline. Students received a small, fixed amount of these “bank days”, but if a student used all of these days, late work would be no longer accepted.

In order to more accurately analyze the dataset, our main analysis only examined students who completed all four project and did not drop or withdraw from the course. The idea behind this was to eliminate the impact of any student who did not place any effort into a project, which would skew our results. The number of students who fell into this category amounted to 82, which represented 25% of the population. The amount of students who dropped or withdrew was much larger in the 2014 semester, with 78% of the students removed from our analysis belonging to that semester. In terms of the number of projects, the total dataset represented 1142 project submissions. Our reduction decreased the number of projects to 992, roughly a 13% reduction in total project amount. After the reductions, the size of each treatment group remained roughly equal (control N = 60, reflective writing N = 64, schedule sheets N = 59, e-mail alerts N = 65).

In an attempt to limit the impact of external factors on our analysis, we also split the data into quartiles based on the final numeric semester score assigned to each student. The goal of this split was to perform analysis between students of similar capability. We chose the final semester score students received because we felt it best represented a student’s entire capability, as the final score of the course took into account project scores and exam scores. In each section of our analysis, we first performed analysis on the reduced dataset and then subsequently performed analysis within quartiles of this group. Analyzing within each quartile is a within subjects comparison, and reduces the bias the “good student effect” might have on our results.

Some of the analysis comparisons we ran indicated the semester had a significant bias on the factors we were analyzing. When we saw this influence occur, we performed another, separate analysis to confirm that fact and see if there were significant differences within each semester. We also discuss possible reasons for this semester bias.

6.1 Intervention Analysis

In this section, we summarize the analysis that we performed on the impact of each intervention. Specifically, we focus on measuring intervention treatments with respect to time and submission attributes such as project size or reference test pass rate. The submission attributes were gathered by examining statistics collected by Web-CAT for each submission.

6.1.1 Interventions Impact on On-time Status

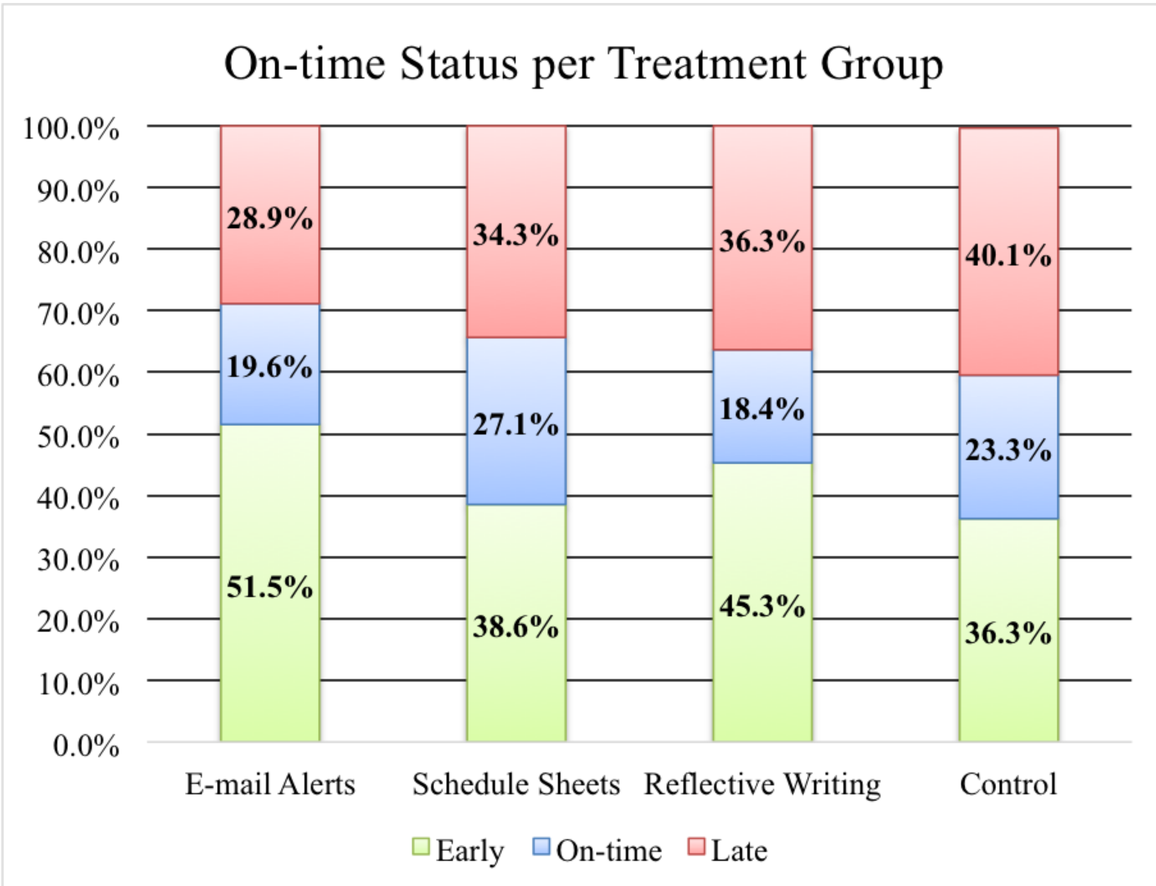


Figure 6.1: The distribution of on-time status against each treatment group.

We analyzed the impact of each intervention on the on-time status across the submissions in our dataset. As mentioned before, the on-time status of a submission records if that submission was submitted early, on-time, or late compared to that project's deadline. Analysis of these variables indicated a significant relationship between treatment group and the on-time status of submissions ($\chi^2 = 12.69$, P <0.005). Examination within each treatment parameter using a Tukey honest significant difference (HSD) test shows the e-mail alert treatment group had a significantly larger early percentage than the control treatment, while the schedule sheet and the reflective writing treatments did not have an observable impact.

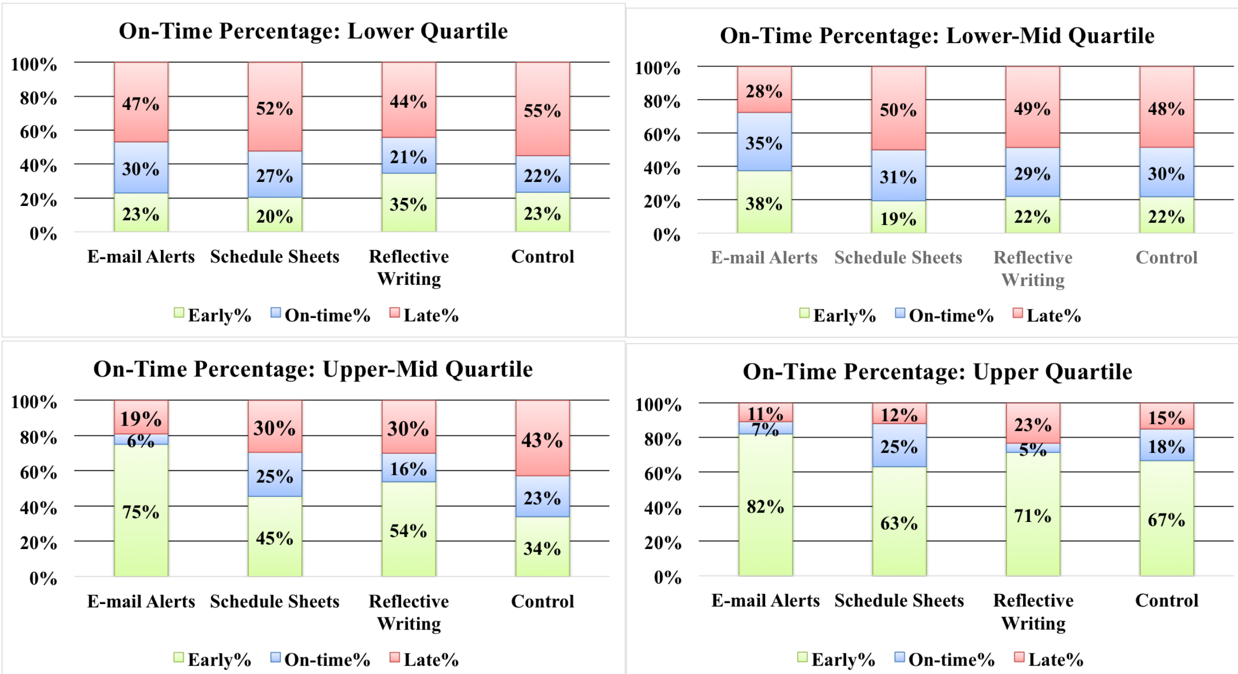


Figure 6.2: On-time status within all quartiles for each treatment group.

Analyzing the impact of each intervention on on-time status within specific quartiles, we find a significant relationship within certain quartiles. Specifically, we find a significant relationship between intervention and on-time status in the upper-middle quartile ($\chi^2 = 25.41$, P <0.0004) and the upper quartile ($\chi^2 = 17.34$, P <0.009). We do not find any observable intervention impact on on-time status for the lower quartile and the lower-middle quartile.

These results seem to show the e-mail alert positively impacting when student submit their work. But the lack of an observable impact on the lower and lower-middle quartile could indicate that all of the interventions are not effective in lower-performing students.

6.1.2 Intervention Impact on First Submission Time

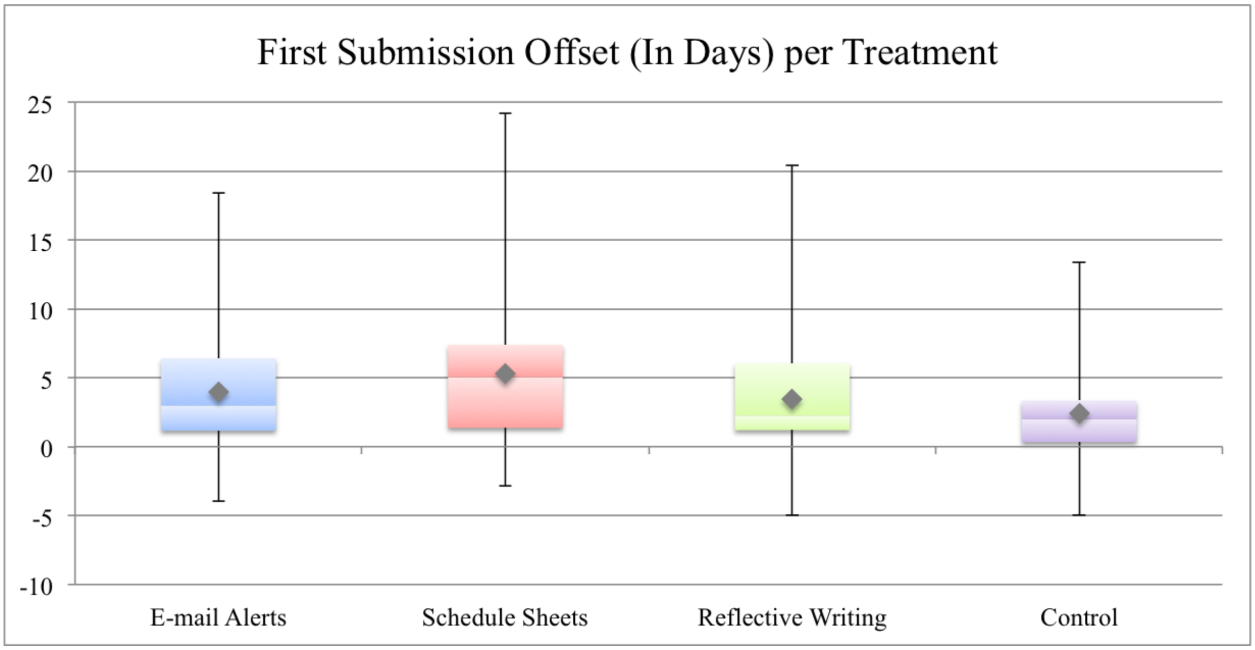


Figure 6.3: The mean first submission offset (calculated by subtracting the time of the first submission from the due date). A larger value indicates an earlier first submission.

We analyzed the impact of each intervention on when students made their first submission to Web-CAT. This was calculated by subtracting the due date time from the first submission time, set to days. Overall, we find a significant relationship between treatment group and first submission time (F ratio = 24.51, P <.0001). A Tukey HSD test shows the schedule sheet intervention had a significantly earlier first submission time than all other treatment groups, while both the e-mail alert intervention and reflective writing intervention had significantly earlier submissions compared to the control group. We should note that students in the section assigned the schedule sheet intervention were required to make a submission to Web-CAT early in the project window, which explains the significance relationship found in that treatment.

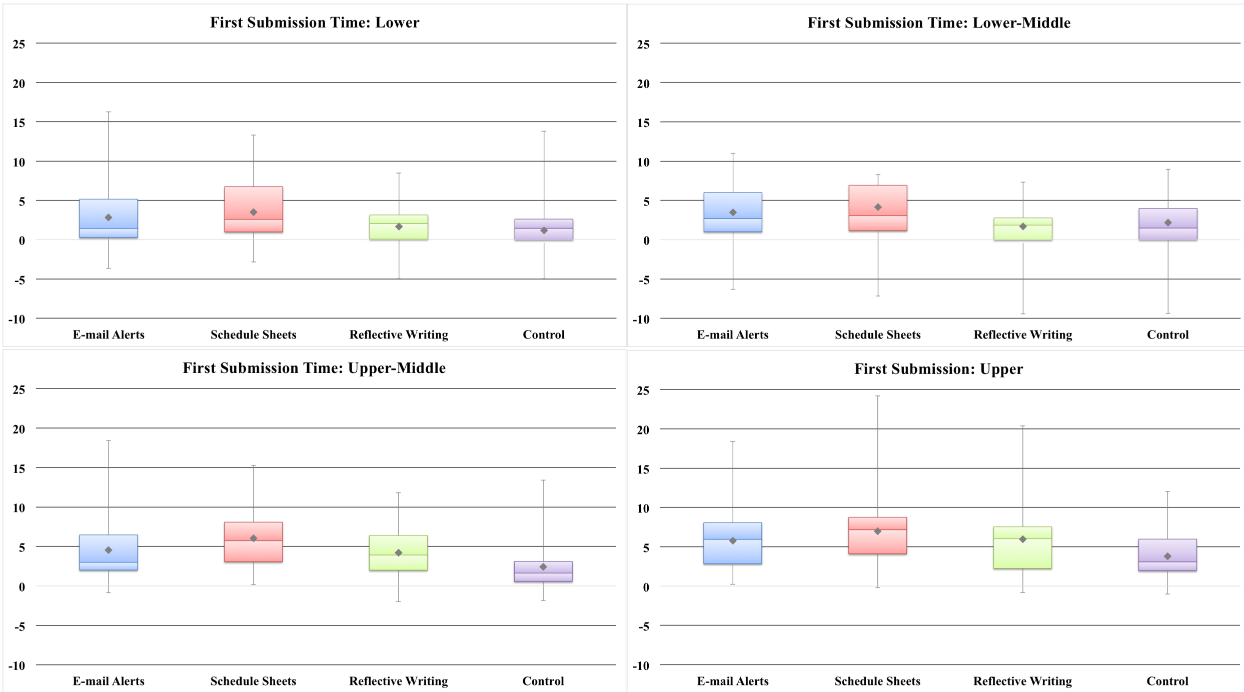


Figure 6.4: First submission time within all quartiles for each treatment group.

Analyzing the impact of intervention on the first submission time within quartiles, we find a significant relationship in each quartile. Specifically, we found this in the lower quartile (F ratio = 5.44, P <.001); lower-middle quartile (F ratio = 7.75, P <.0001); upper-middle quartile (F ratio = 8.98, P <.0001), and upper quartile (F ratio = 7.36, P <.0001).

The result we find here shows a positive impact for all of the interventions. Having all three interventions show a significantly earlier submission time compared to the control treatment is an indication these interventions are influencing students to start their work earlier. This is very definition of procrastination reduction. However, the results are skewed by the requirement of the schedule sheet treatment group to make a submission to Web-CAT soon after the projects were assigned.

6.1.3 Intervention Impact on Final Submission Time

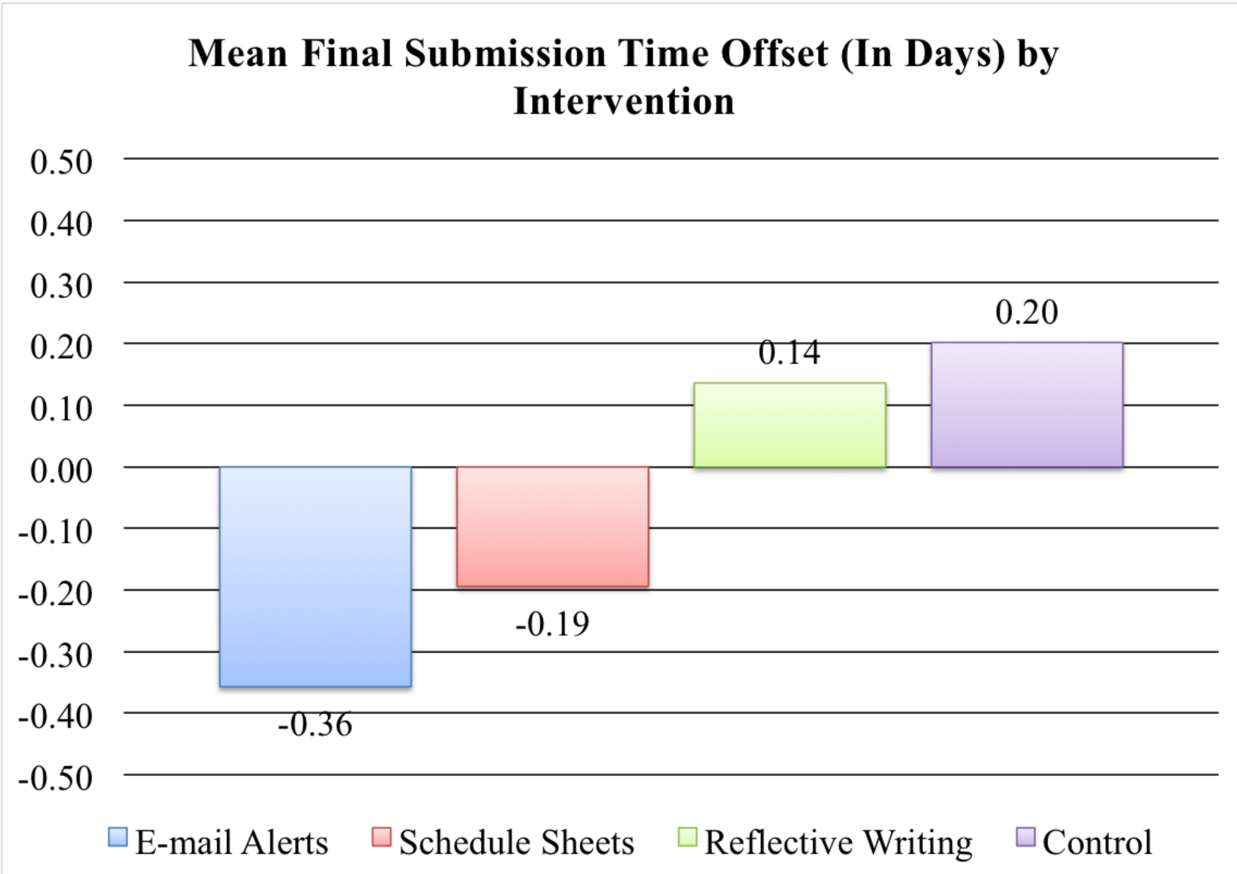


Figure 6.5: The final submission values (in days) per treatment group. A negative value indicates that the submission time was before the deadline, and a positive value indicates that the submission time was after the deadline.

We now analyze intervention impact on submission time by examining the impact of each intervention on the final submission time of a project. This serves as a continuous value for on-time status. Instead of putting project submissions into three discrete bins as we do with on-time status, we measure the final submission time as the raw value a student actually submitted. Analysis yields a relationship that is significant (F ratio = 3.67, $P < 0.012$). A Tukey HSD test to examine each separate treatment group shows that the e-mail alert intervention had a significantly earlier final submission time when compared to the control treatment. There was no observable impact on final submission time for any other treatment group.

Analyzing the impact of each intervention on the final submission time within each quartile, we find almost no observable impact. The only quartile that did show a significant relationship was the upper-middle treatment (F ratio = 2.90, P <0.035). A Tukey HSD test within this quartile revealed no observable difference between the intervention groups.

These results are positive for the e-mail alert intervention. The significant relationships were expected due to the significant relationship found between on-time status and intervention already. But the lack of a significant relationship between final submission time and intervention within any quartile shows the impact of treatment might be influenced by inherent student quality, not just the intervention.

6.1.4 Intervention Impact on Solution Correctness

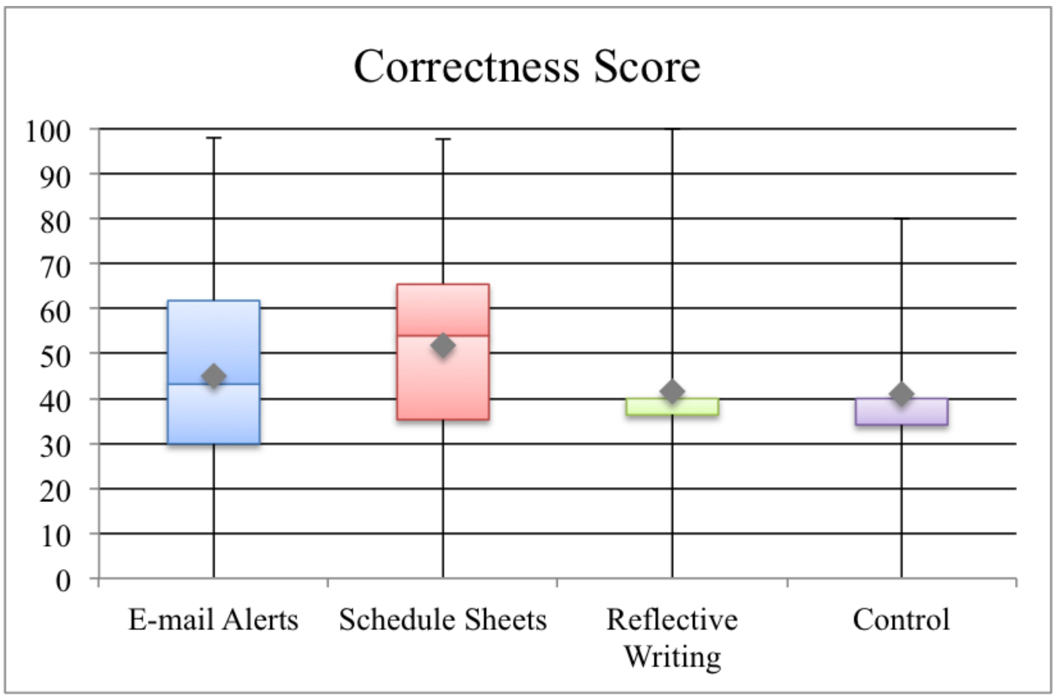


Figure 6.6: Correctness score for each treatment group.

Analyzing the impact of intervention on the correctness score earned by a submission, we find a significant relationship (F ratio = 13.87, P <0.0001). Using a Tukey HSD test to compare the effects of the different treatments, we find the schedule sheet treatment group had a significantly higher correctness score compared with all other treatment groups.

Expanding our analysis to examine the correctness score within each quartile, we find a significant relationship in the lower-middle (F ratio = 3.86, P <0.01), the upper-middle (F ratio = 6.73, P <0.0002), and the upper quartiles (F ratio = 5.29, P <0.002). Using a

Tukey HSD test to compare treatments within each quartile, we found the schedule sheet treatment had a significantly higher correctness score compared to the control group. This indicates that the impact of intervention on correctness was significant despite inherent student capability.

Examining the overall graph of solution correctness, it seems the semester has a significant impact on the correctness score range. In particular, the treatment groups for Fall 2013 (the reflective writing and control groups) had a very small third quartile range of correctness score value. To check this, we split our analysis by semester to examine the treatment groups in their own semesters. We found a significant relationship between treatment and correctness in Fall 2014 (F ratio = 10.08, P <.002) but we did not find a significant relationship in Fall 2013 (F ratio = 0.21, P <0.65). This confirms that semester did indeed have an influence on these results. This is because the solution correctness score is influenced by the reference test score, which also shows a semester influence. We described this influence when we analyze the impact of treatment on the reference test pass rate.

Overall, these results are positive for the schedule sheet treatment. Based on the initial analysis and the quartile analysis, it seems the schedule sheet treatment is positively impacting the quality of project solutions. But our results are certainly being influenced by which semester the treatments were issued in, which must be accounted for.

6.1.5 Intervention Impact on Reference Tests Passed

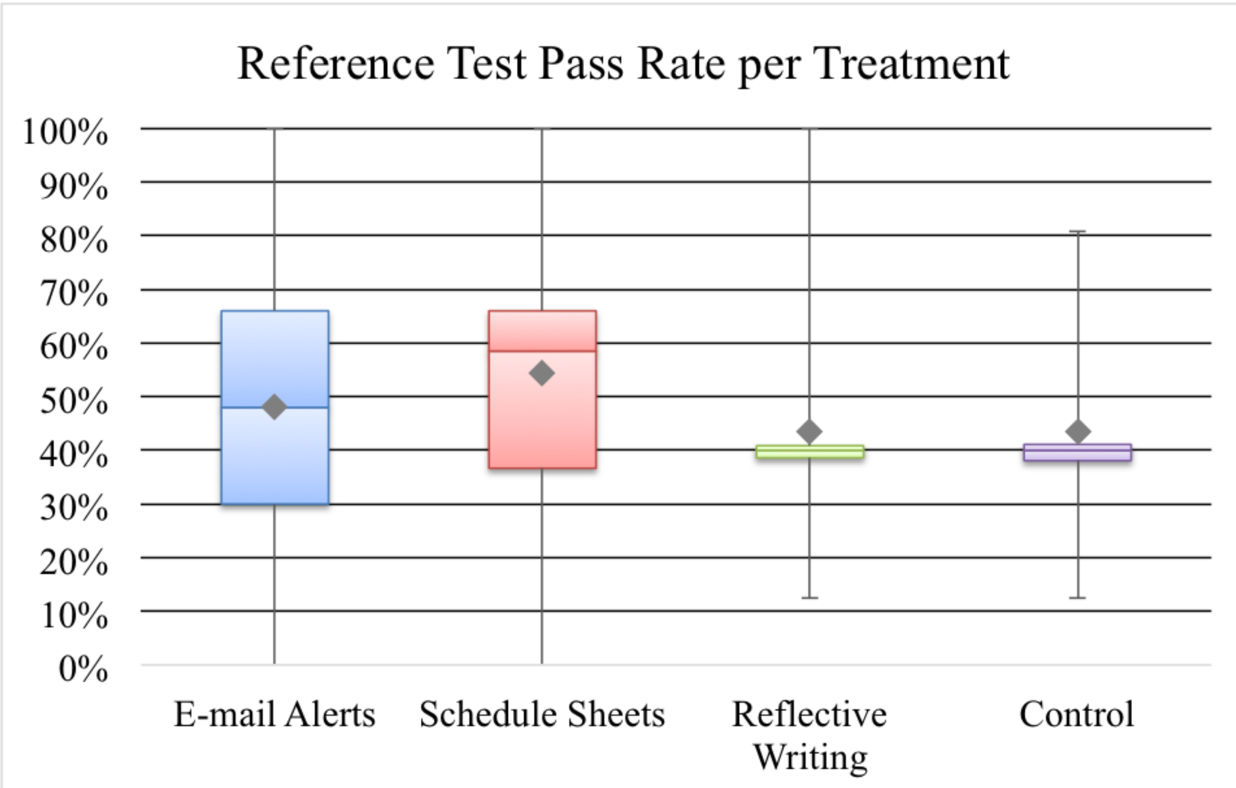


Figure 6.7: Percentage of reference tests passed for each treatment group.

Web-CAT allows for the automatic testing of code by running instructor written reference tests. The number of tests passed is a solid measure of the quality of a submission. Analyzing the impact of treatment on the percentage of reference tests passed in the final submission to Web-CAT, we find a significant relationship (F ratio = 15.03, P <.0001). Performing a Tukey HSD test, we find that the schedule sheet treatment had a significantly higher reference test pass rate compared with all other treatments. No other treatment had a significant relationship.

Figure 6.8: The percentage of reference tests passed within all quartiles for each treatment group.

Analyzing the impact of treatment on the number of reference tests passed in each quartile, we find significant relationships in three of the four quartiles. These quartiles are the lower-middle (F ratio = 4.70, P <.003), the upper-middle (F ratio = 8.96, P <.0001) and the upper (F ratio = 7.27, P <.0001). Running a Tukey HSD test test on these quartiles, we find that the upper-middle and upper quartiles share the same significant treatment parings, with both the e-mail alert and schedule sheet treatments showing significance over the control and reflective writing treatments. The lower-middle quartile has only the schedule sheet treatment showing significance over both the reflective writing and control treatments.

Reference test pass rate probably represents one of the best ways to measure submission quality. As these tests measure the expected behavior of a particular program, a higher pass rate indicates students completed more of the expected behavior the assignment asked for.

This analysis further cements the schedule sheet intervention as providing a positive influence on code quality. Furthermore, the quartile analysis still yields significant relationships for three of the four quartiles, indicating there is still an intervention impact even with students who have similar aptitude.

However, it is clear from the graphs that the data show a semester bias. There is a large gap between the range of reference tests passed from the Fall 2013 semester and the Fall 2014 semester. This difference is mostly likely due to the experience of the GTA who assisted with the course. In the 2013 semester, the TA was new to the Web-CAT system and developed inferior reference tests for the course. By 2014, the TA was more experienced writing at writing reference tests. As students in the 2013 semester suffered from inferior reference tests, it is likely they had a high instance of passing some easy simple tests while failing a few complicated tests. The students in 2014 had higher quality reference tests that examined a wider variety of project features, thus leading to the wider range of pass rates.

Splitting the treatment groups by semester, we still find that the schedule sheets group had a significantly higher reference test pass rate compared to the e-mail alerts group (F ratio $= 8.15$, $P < 0.005$). Both of these groups were in the Fall 2014 semester, and thus had the benefit of better reference tests.

6.1.6 Intervention Impact on Coverage Achieved

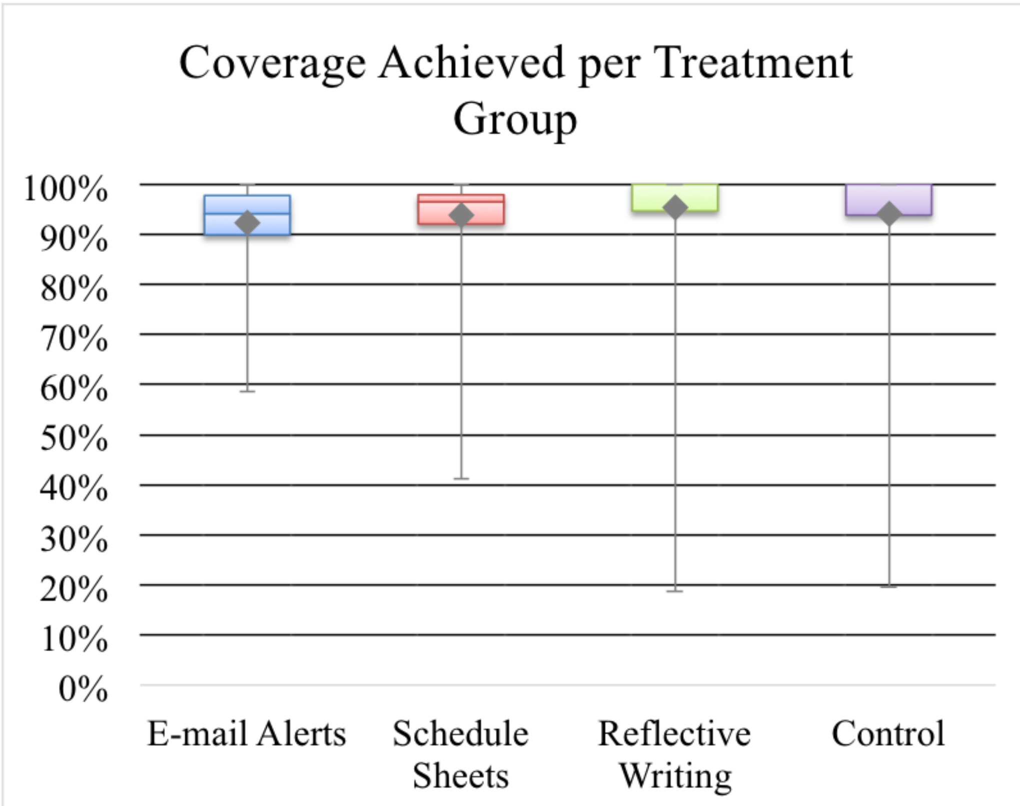


Figure 6.9: Percentage of test coverage achieved for each treatment group.

When analyzing the impact of each intervention on the coverage achieved over tests, we find a significant relationship (F ratio = 3.79, P <.01). Performing a Tukey HSD test, we find the reflective writing intervention has significantly higher coverage compared to the e-mail alert intervention. No other treatment pairing showed any significant difference.

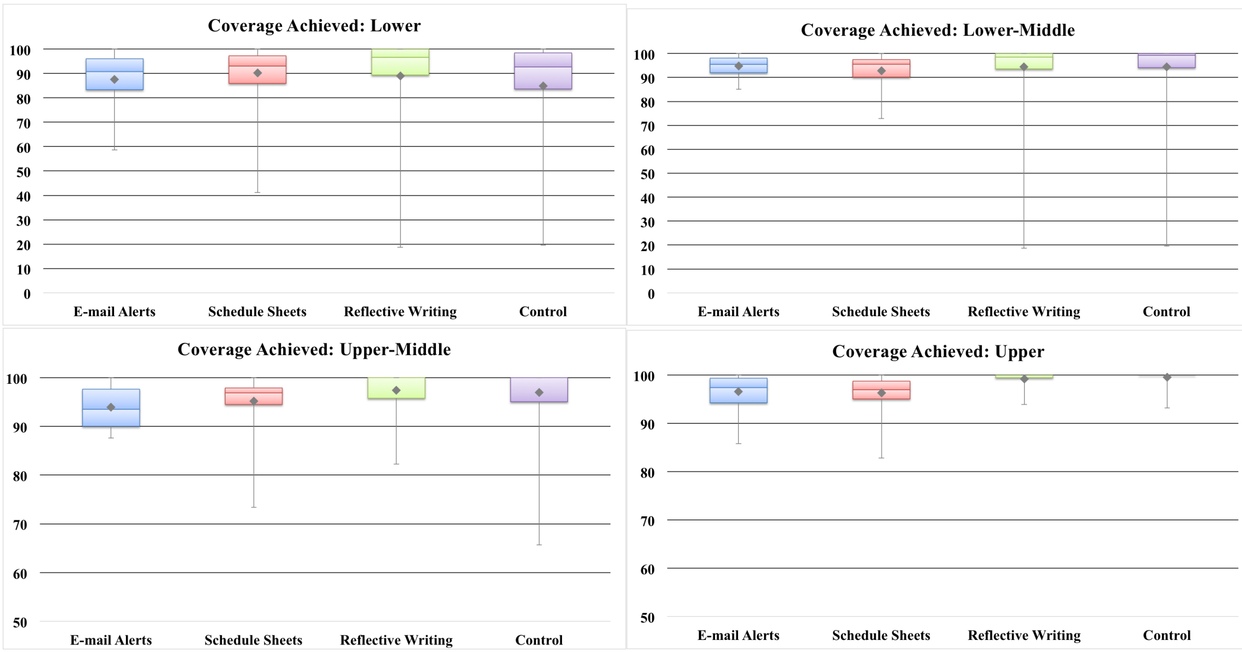


Figure 6.10: Percentage of code covered by tests within all quartiles for each treatment group.

Analysis of intervention impact on coverage achieved by quartile, we only find a significant relationship between the top two quartiles, upper-middle and upper. For the upper quartile, the control and reflective writing group were significant over e-mail alerts and schedule sheets. For the upper-middle quartile, the reflective writing and e-mail treatments are significantly greater than the control group only.

Coverage achieved by student tests also provide a good metric for code quality, as ideal testing should cover all possible execution paths. Based on this analysis, it does not seem that there was a strong pattern between treatment group and code coverage. We should note that code coverage is a metric students can optimize for. Because they are the writers of their own unit tests, they can simply add extra test code and re-submit to Web-CAT. As a result, the code coverage rate across all interventions was fairly high (mean above 90 for all treatments) and it would be difficult to consistently find any single intervention having an impact.

6.1.7 Intervention Impact on Submission Number

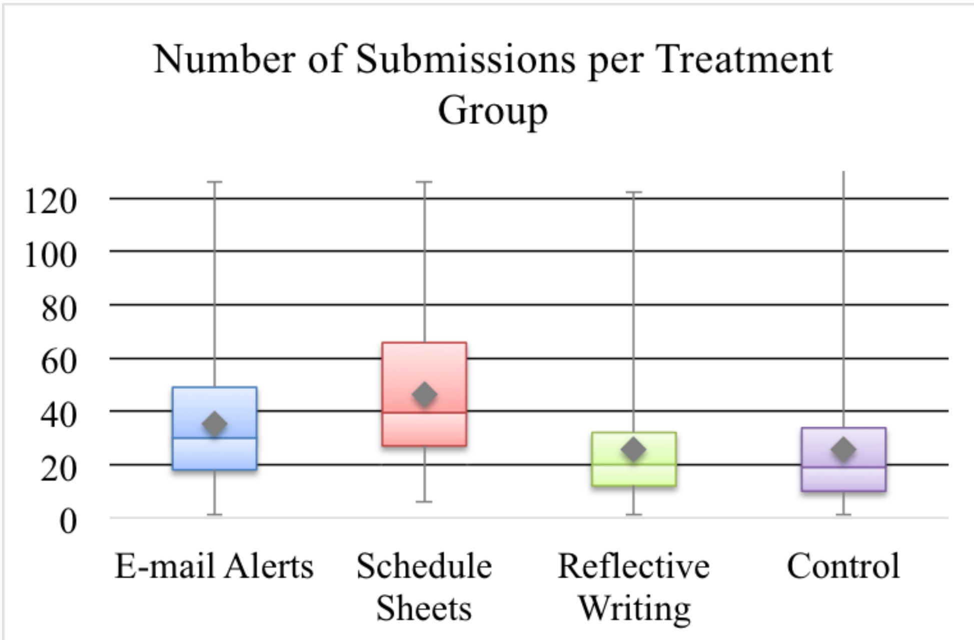


Figure 6.11: The number of project submissions for each treatment group.

Analyzing the impact of the interventions on the number of submissions that students made to Web-CAT, we find a significant relationship (F ratio = 42.46, P <.0001) between the factors. Using a Tukey HSD test to compare the individual interventions, we find that the schedule sheet intervention had a significantly higher mean number of submissions compared to all other treatments. This was followed by the e-mail alert intervention, which had a significantly higher mean number of submissions than both the control and reflective writing treatment groups. There was no observable effect in any other treatment group.

Figure 6.12: The number of submissions within all quartiles for each treatment group.

Analyzing the impact of intervention on the number of Web-CAT submissions per project within each quartile, we find a significant relationship between the factors in every quartile (Lower F ratio = 6.77, P <.0002; Lower-Middle F ratio = 16.71, P <.0001; Upper-Middle F ratio = 11.61, P <.0001; Upper F ratio = 9.91, P <.0001). A Tukey HSD test reveals different significant intervention pairings for each quartile. For the lower quartile, the schedule sheet treatment group has a significantly larger number of submissions compared to both the reflective writing and control groups, while the e-mail alert intervention has a significantly larger number of submissions over the control group only. For the lower-middle quartile, the schedule sheet group shows a significantly greater number of submissions over both the control and reflective writing treatment, while the e-mail alert treatment had significantly greater average number of submissions over the reflective writing treatment. For the upper-middle quartile, the schedule sheet treatment was found to be significant over all other treatments, while the e-mail alert and control treatments were significant over the reflective writing treatment. Finally, in the upper quartile, the schedule sheet treatment was found to be significant over reflective writing and control treatments.

Our initial hypothesis for this analysis is that more Web-CAT submissions correlate to higher quality on a student's final submission. This would make any potential impact treatment group had on the overall number of Web-CAT submissions very important. While we did find a significant relationship between the number of Web-CAT submissions and reference tests passed for all students, reducing our dataset to remove those students who dropped or withdrew from the course removed this finding, potentially making our findings moot.

6.1.8 Intervention Impact on Solution Size

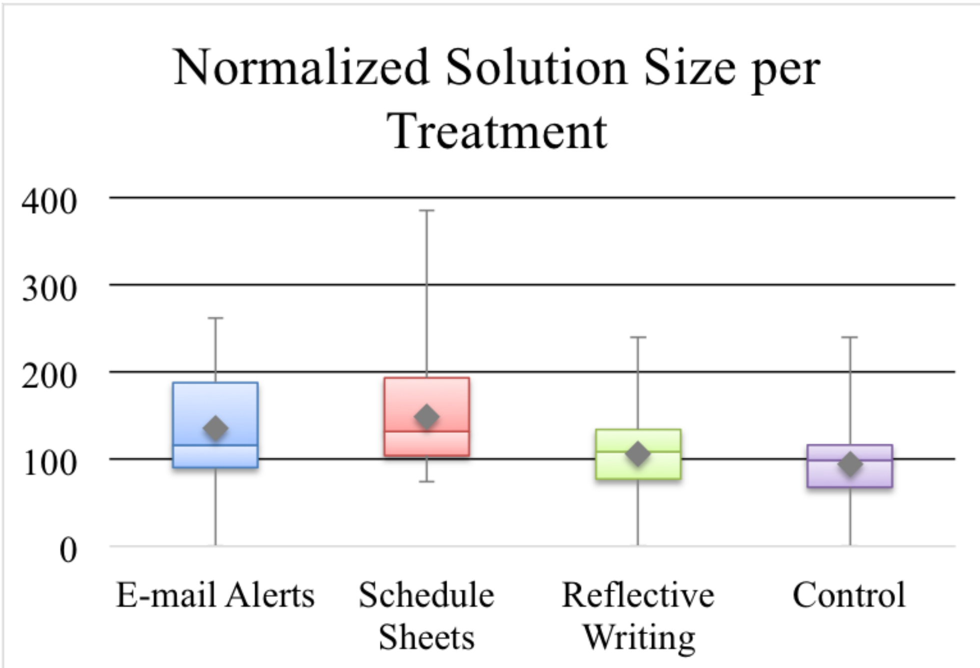


Figure 6.13: The solution size for each treatment group. The values are taken from the final Web-CAT submission.

In order to analyze the impact of intervention on solution size, we normalized the non-commented lines of code value for each project submission compared to the mean of this value for each project in similar semesters. This normalized projects within the semester they were issued, thus normalizing the same projects together. This allowed us to perform a more accurate comparison of project solution size across different projects.

When analyzing the impact of each intervention on the solution size value, we find a significant relationship between the factors (F ratio = 65.34, $P < .0001$). Using a Tukey HSD test to analyze each individual intervention, we find all treatment groups are significantly different from each other. The schedule sheet intervention produced the largest normalized solution size, being significantly greater than all other treatment groups. The e-mail alert treatment was next, having a significantly larger solution size compared to every intervention except the schedule sheet group. The reflective writing intervention only had a significantly higher solution compared to the control group.

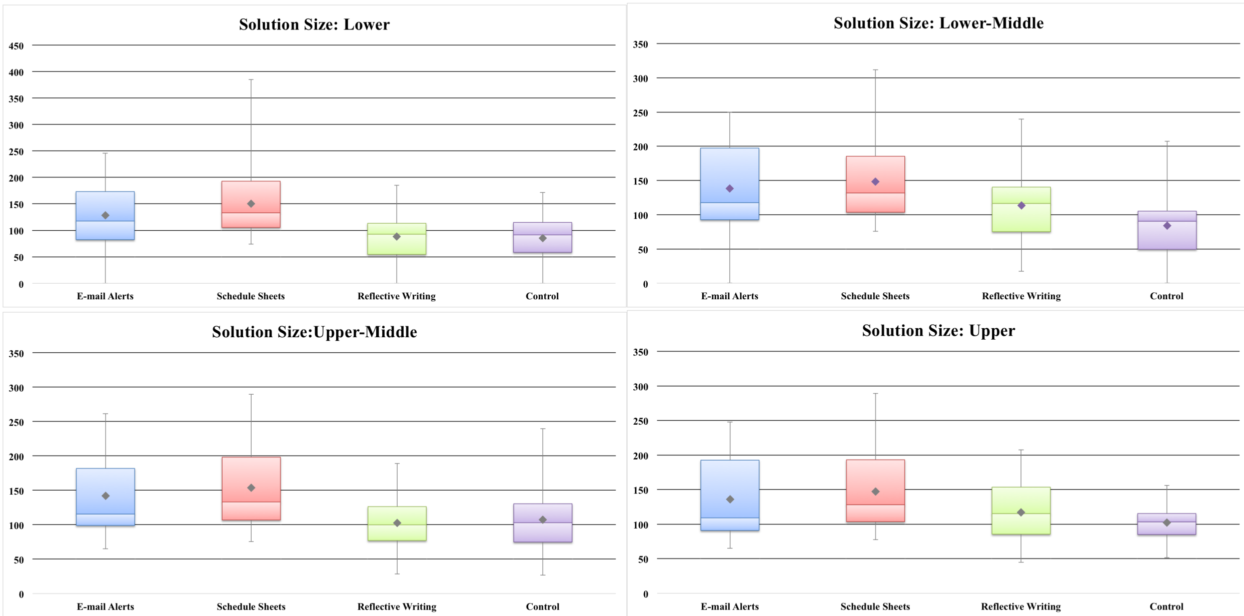


Figure 6.14: The normalized solution size within all quartiles for each treatment group.

Analyzing the solution size within quartiles, we find a significant relationship between treatments and solution size in all four quartiles (Lower F ratio = 20.15, P <.0001, Lower-Middle F ratio = 20.83, P <.0001, Upper-Middle F ratio = 16.39, P <.0001, Upper F ratio = 12.93, P <.0001). This indicates that the significant relationship found between intervention and solution size holds even when comparing students of similar aptitude.

Examining solutions size alone does not confirm that a submission is of higher quality. However, when combined with reference test pass rate, code coverage, and other clear indicators of quality, a larger solution size can indicate give evidence for submissions of higher quality. Therefore, the relationship found between intervention and a larger solution size is a positive sign for all of the interventions.

6.1.9 Intervention Impact on Code Size (Test)

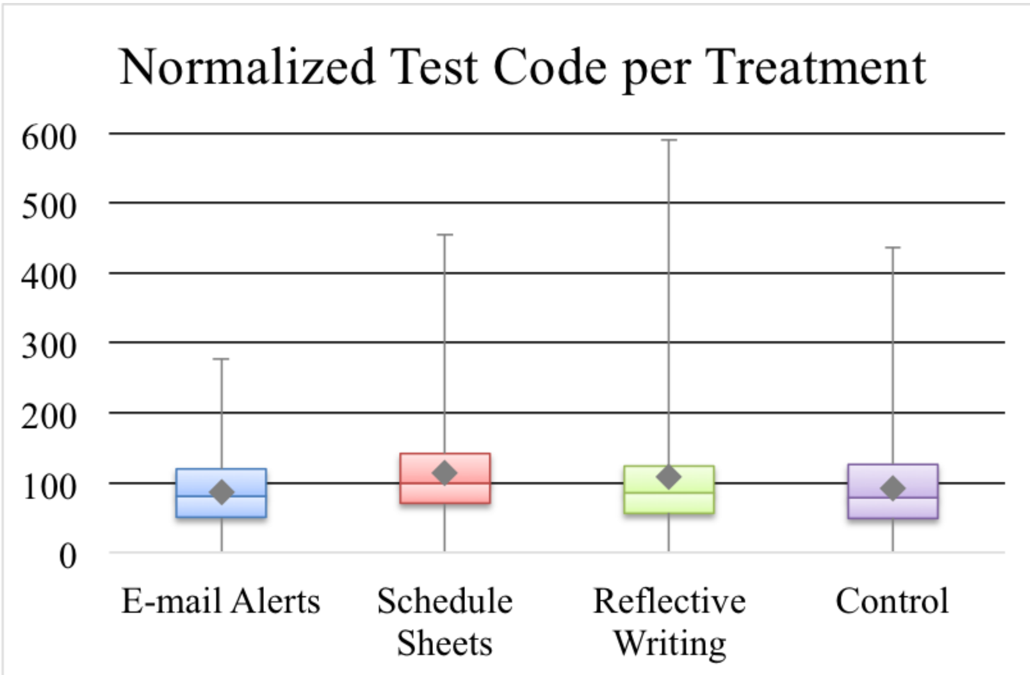


Figure 6.15: The test code size for each treatment group. The values are taken from the final submission to Web-CAT.

When analyzing the intervention impact on test code size, we find a significant relationship (F ratio = 8.16, P <.0001). Running a Tukey HSD test to find relationships between each separate intervention, we find the schedule sheet intervention had significantly higher test code compared to both the control and e-mail alert groups. The reflective writing treatment also had significantly higher test code size compared to the e-mail alert treatment group.

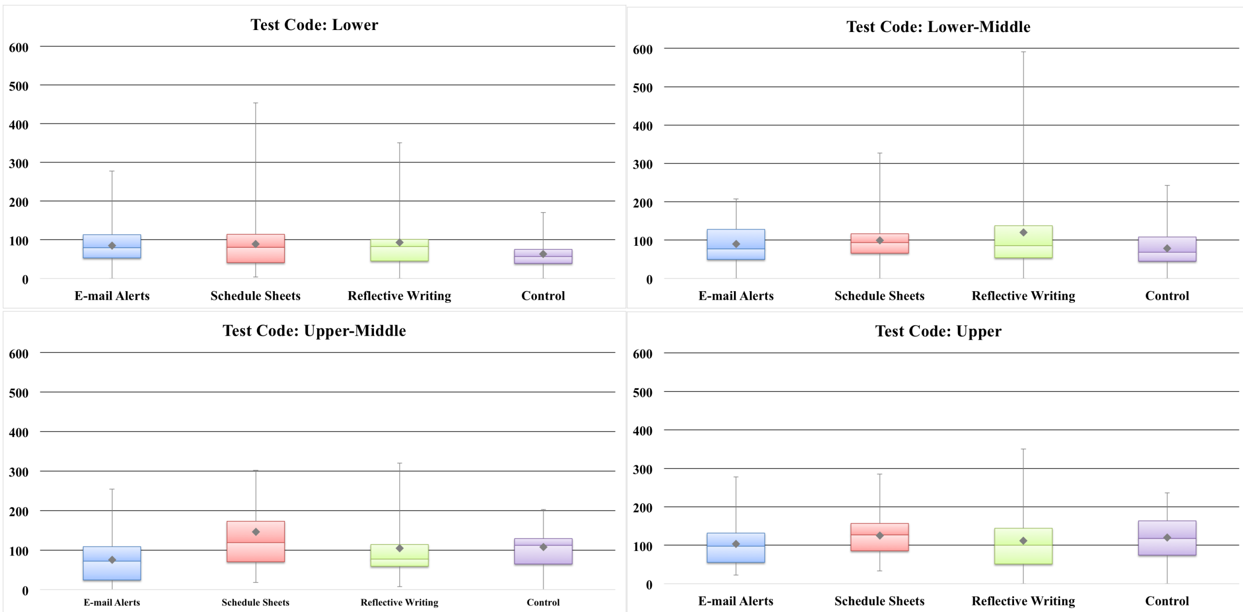


Figure 6.16: The normalized test code size within all quartiles for each treatment group.

Analyzing the impact of intervention on test size within each quartile, we find intervention has a significant impact on test size in three quartiles (Lower F ratio = 2.81, P <.039; Lower-Middle F Ratio = 3.72, P <.012; Upper-Middle F ratio = 6.88, P <.0002). In the upper quartile, there was no observable impact of treatment on test size.

Again, we find similar significant results for test size as we did for solution size. Students must write a larger number of tests for a larger solution in order to accurately test that solution. The quartile comparisons show no sign difference, thus we can say this relationship is most likely due to the treatments themselves and not other factors. While test size cannot be used as a direct measure of submission quality, combining the metric with other quality indicators provides a more complete picture of each submission.

6.1.10 Procrastination Score of Each Treatment Group

At the beginning of each section included in our experiment, we tested students using the procrastination scale measurement quiz developed by Tuckman. This scale ranks students based on their tendency to procrastinate, with a lower score (closer to 0) indicating a student is less likely to procrastinate and a higher score (closer to 1.0) indicating the student is more likely to procrastinate. This scale has been previously shown to correlate well with a students procrastination habits.

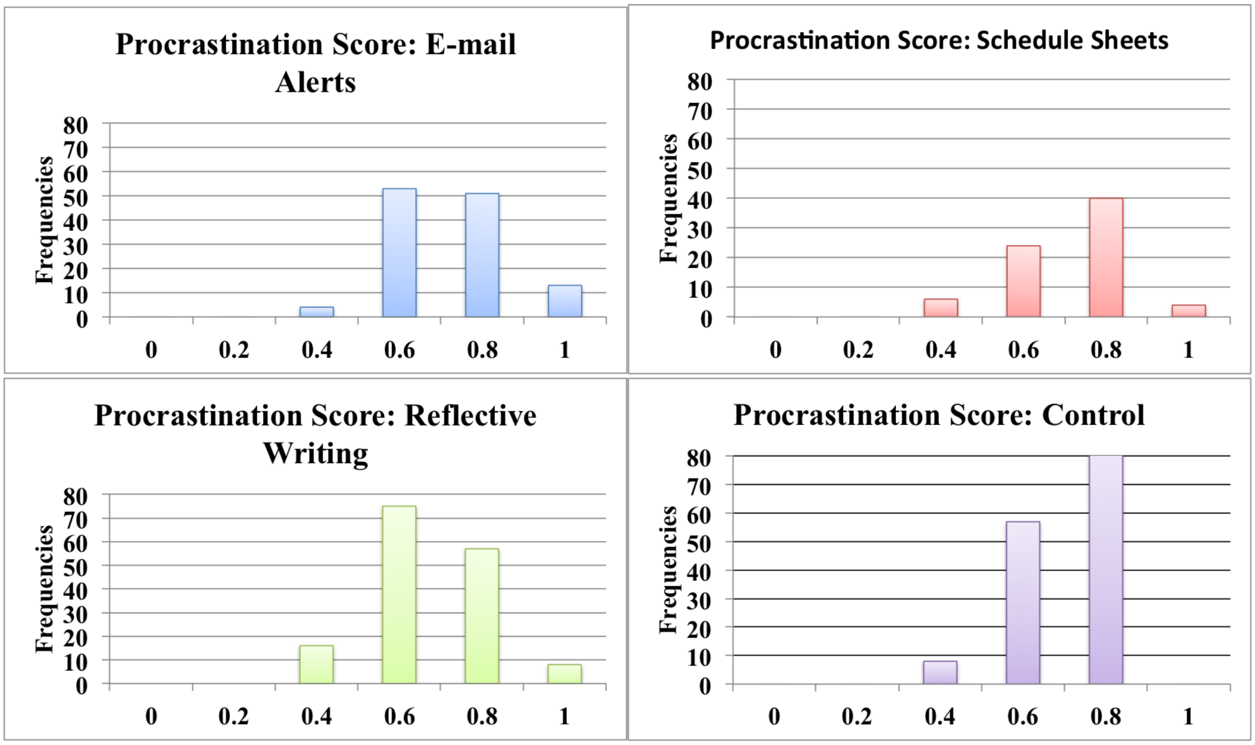


Figure 6.17: The procrastination survey score for each treatment group.

To examine if any single treatment group had a population that was significantly more likely to procrastinate, we analyzed the procrastination scale scores received by the treatment groups. These results showed only the e-mail alert treatment group had a significantly higher procrastination score compared to the reflective writing treatment (e-mail alert mean = .62, reflective writing mean = .57). All other treatment groups had no significantly different procrastination scores, indicating the analysis we performed was mostly on students with a similar tendency to procrastinate.

6.2 On-time Submission Analysis

In this section, we examine the impact of on-time status on various elements of submission quality. As explained earlier in this thesis, on-time status records if a project submission was submitted more than 1 day before the deadline, within 24 hours of the deadline, or after the deadline. We use this as a basic measure of lateness, and by examining different aspects of project quality, we hope to confirm that late work has significantly lower quality compared to earlier work.

6.2.1 On-time Impact on Reference Test Pass Rate

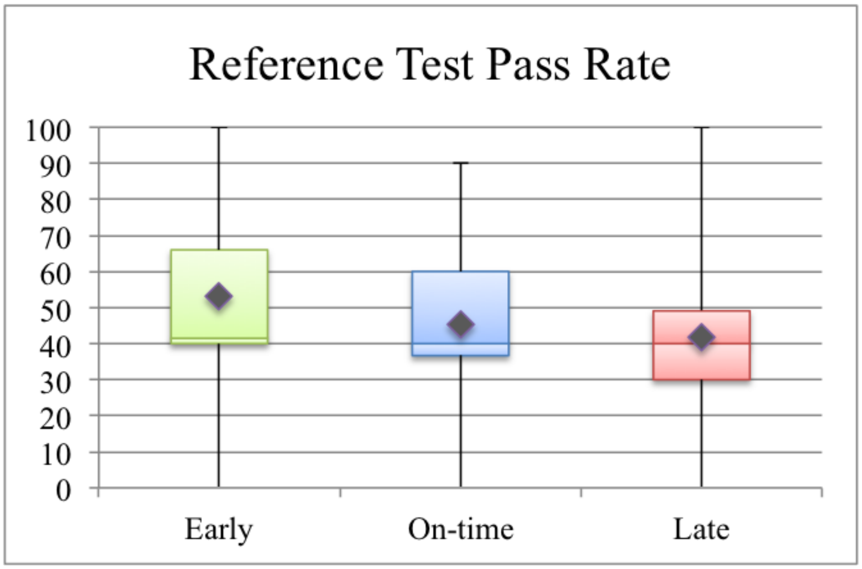


Figure 6.18: The percentage of reference tests passed for each on-time status category.

Analyzing the impact of on-time status on reference test pass rate, we find a significant relationship (F ratio = 30.90, P <0.0001). Projects that were submitted early had a significantly greater reference test pass rate than projects submitted late, which seems to confirm our initial hypothesis.

One potential limitation to this analysis is that "good students" simply do better work and finish earlier, and therefore receive higher scores than weaker students, who are unable to finish their work earlier due to the struggles they encounter when trying to complete an assignment. To help remove this potential impact, we removed students who only submitted their work early or late. With the remaining students, we performed a within-subjects comparison of the reference test pass rate that compared the pass rate achieved for both early and late work for the same student. With this comparison, we still found the on-time status of a submission to be a significant factor in the percentage of reference tests passed (F ratio = 13.7, P <0.0001).

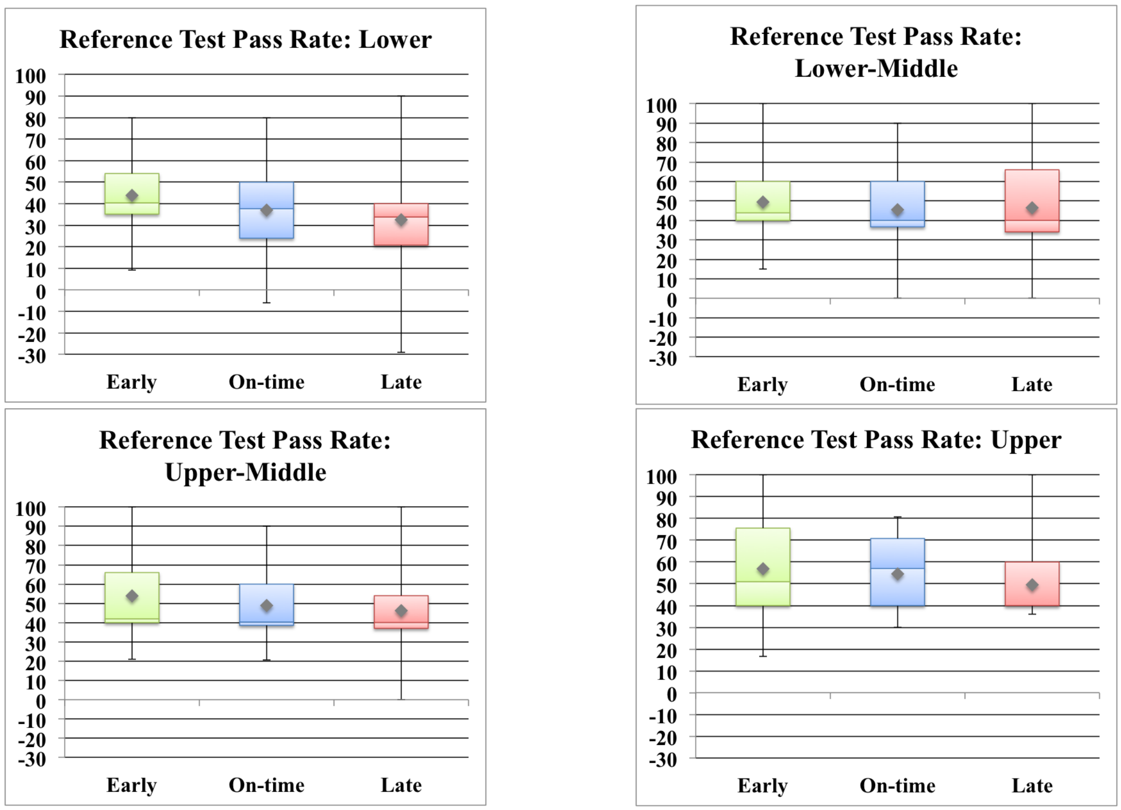


Figure 6.19: The percentage of reference tests passed within all quartiles for each on-time status category.

Examining the relationship between on-time status and reference test pass rate, we find a significant relationship in the lower quartile (F ratio = 6.67, P <0.0016) and the upper-middle quartile (F ratio = 3.98, P <0.02). Within the lower-middle quartile and the upper quartile, no significant relationship was found. Performing a Tukey HSD test within the lower and upper-middle quartile, we find that in both quartiles early submissions had significantly higher reference test pass rates compared to late submissions. This further supports the idea that students who submit their work early experience a significant gain in quality compared to students who submit late.

As mentioned earlier in this thesis, the quality of the reference tests themselves changed between the Fall 2013 semester and the Fall 2014 semester. Therefore, we split our analysis based on semester to see if the significant relationship found between on-time status and reference test pass rate held. We found a significant relationship between these two factors for both Fall 2013 (F ratio = 8.63, P <0.0002) and Fall 2014 (F ratio = 21.26, P <0.0001). A Tukey HSD confirmed that earlier submissions had significantly higher reference test pass rates compared to late submissions for both semesters.

This analysis validates our hypothesis that submitting late impacts the quality of that submission. The correlation between on-time status and reference test pass rate was found both in a within subjects comparison as well as within two quartiles. As reference test pass rate is a strong indications of submission quality, this analysis is a strong indication of the impact of submitting early on submission quality.

6.2.2 On-time Impact on Code Coverage

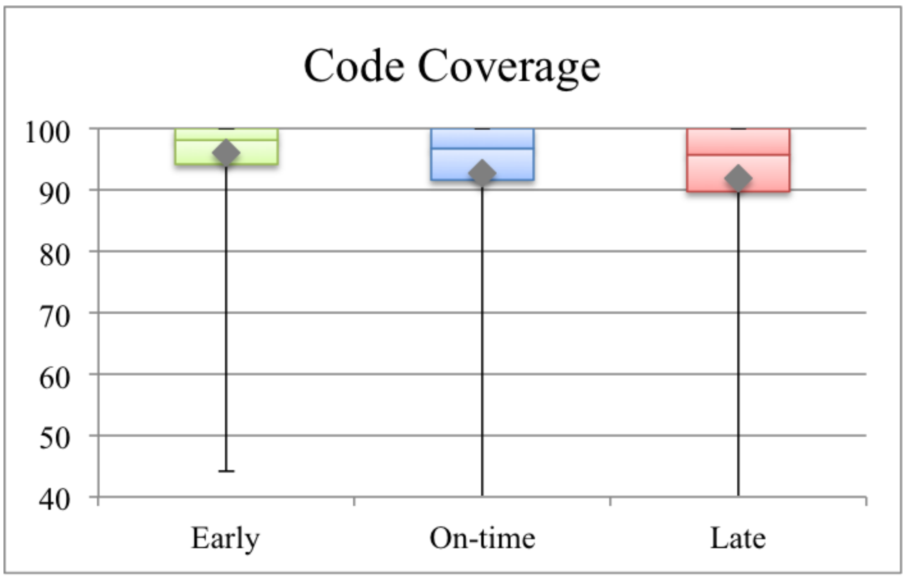


Figure 6.20: The percentage of code covered by tests for each on-time status category.

When submitting to Web-CAT, students were required to write unit tests for their solutions. Students were graded on how much of their solution the tests covered. Analyzing the impact of on-time status on code coverage, we find a significant relationship (F ratio = 17.87, $P < 0.0001$). A Tukey HSD test reveals early submissions had a significantly higher code coverage percentage compared to on-time or late submissions.

The fact that early submissions showed a significantly higher code coverage score compared to on-time or late submissions further validates our hypothesis that work submitted earlier has significantly higher quality compared to work submitted late. We should note, however, that code coverage was fairly high regardless of the on-time status of the submission. Since students wrote their own unit tests, they could modify the tests themselves if they were missing code coverage. Students who completed their own work earlier most likely had the time to expand their unit tests, while students submitting late did not have that additional time.

6.2.3 On-time Impact on Correctness Score

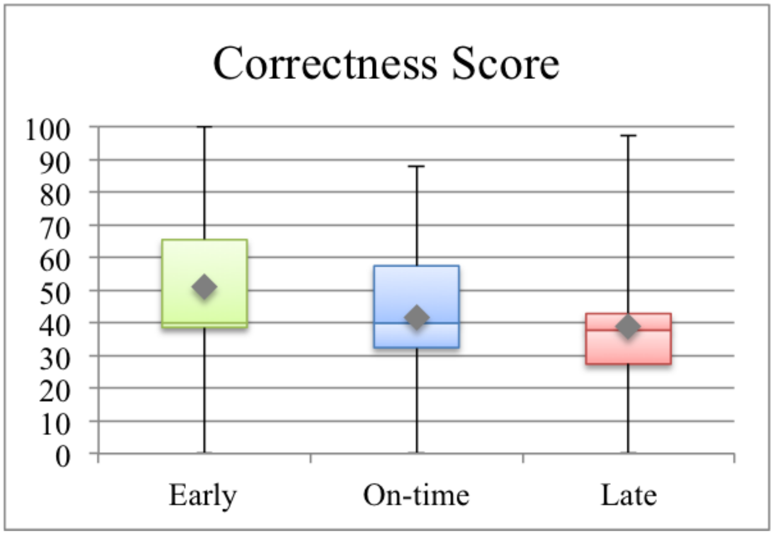


Figure 6.21: The correctness score for each on-time status category.

The *correctness* score of a submission is the score of that submission before the TA score, late penalties, early bonuses, or other score modifiers are included. Analyzing the impact on on-time status on this value, we find a significant relationship (F ratio = 35.77, P <0.0001). Performing a Tukey HSD test between the values of on-time status, we find that early submissions had a significantly higher correctness score compared to on-time or late submissions. This indicates that projects submitted earlier received a higher score compared to projects submitted late, further supporting our hypothesis that early work is of a significantly greater quality compared to late work. This result should not be surprising, however, as the correctness score of a solution is greatly influenced by the code coverage and reference test passed. Since we found a significant relationship between on-time status and those values, it follows that we would find such a relationship here.

6.2.4 On-time Impact on TA Score

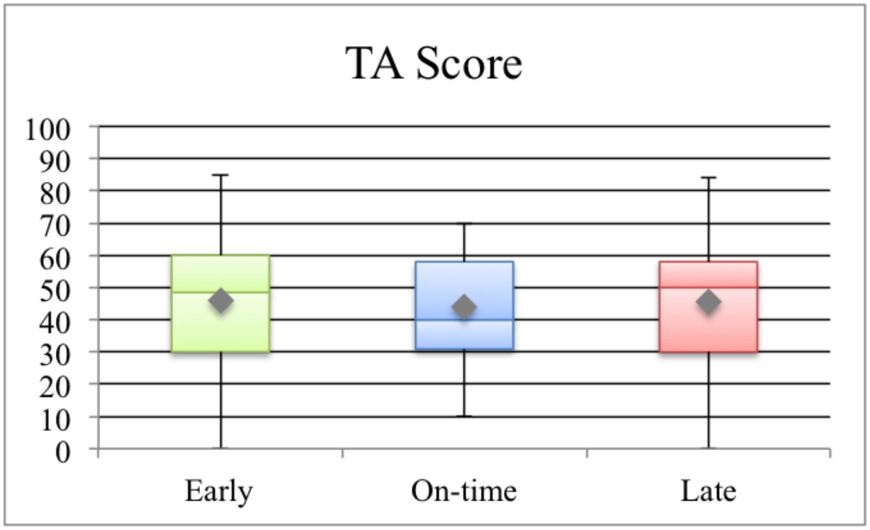


Figure 6.22: The value of the score given by the course TA for each on-time status category.

When analyzing the impact of on-time status on the score given by the teaching assistants, we find no significant relationship (F ratio 1.06, P <0.63). This indicates that the overall time a project was submitted had little to no impact on the score assigned by the TA for the course. The mean scores for all types of assignments were virtually identical, with 45.96 for early submissions, 44.11 for on-time submissions, and 45.77 for late assignments.

We should note that the requirements for the TA grades varied between semesters. For Fall 2013, the TAs graded each submission based on style adherence as well as design choices, while in Fall 2014 the TAs only graded the design of a submission. To account for this, we split the TA grades by semesters and analyzed the impact of on-time status within each semester. For Fall 2013, we found early submissions had a significantly higher TA score than on-time submissions. We found no significant relationship between early submissions and late submissions. In Fall 2014, we found no significant relationship between on-time status and TA score. Overall, it seems the course TAs assigned similar grades to submissions regardless of it they were on-time.

6.2.5 On-time Impact on Solution Size

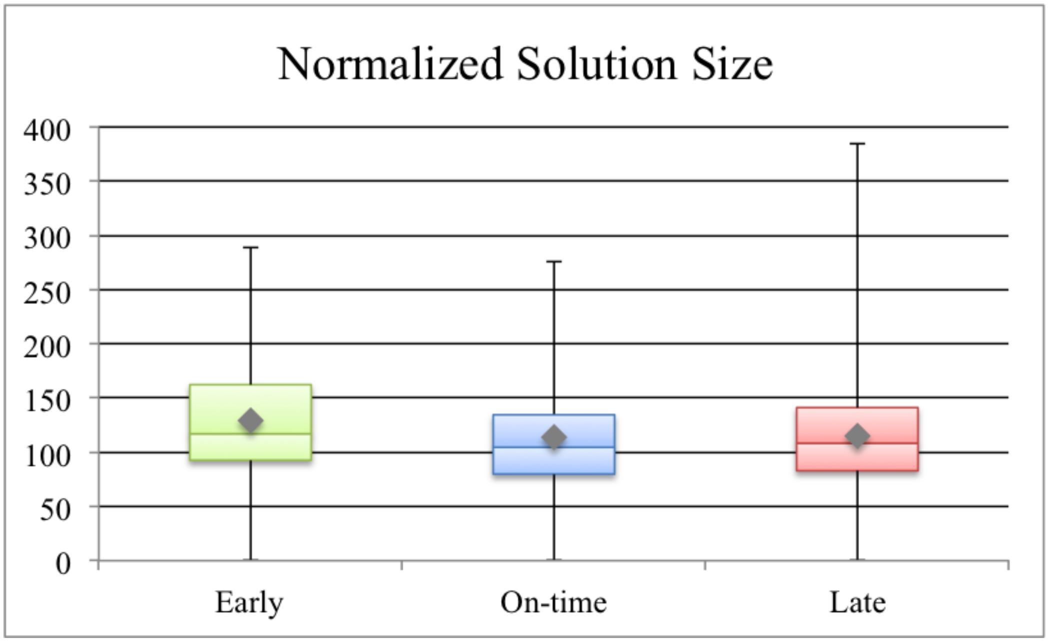


Figure 6.23: The normalized solution size for each on-time status category.

We examined the impact of on-time status on normalized solution size to examine the quality of a project submission. While these values are not indicative of quality by themselves, they can provide a picture of submission quality when combined with other quality measures.

To accurately account for the different projects across semesters, we normalized each code size value. This process is described earlier in the thesis, specifically when we analyzed the impact of intervention on solution size.

When analyzing the impact of on-time status against that project's solution size, we find a significant relationship (F ratio = 8.67, P <.0002). A Tukey HSD test reveals that early submissions had a significantly larger average solution size compared to on-time or late submissions.

This analysis meets our expectations. While solution size alone does not indicate a higher quality submission, this analysis combined with the reference test examination further supports our hypothesis that students who submit early generally have higher quality compared to students who submit on-time or late.

6.2.6 On-time Impact on Test Size

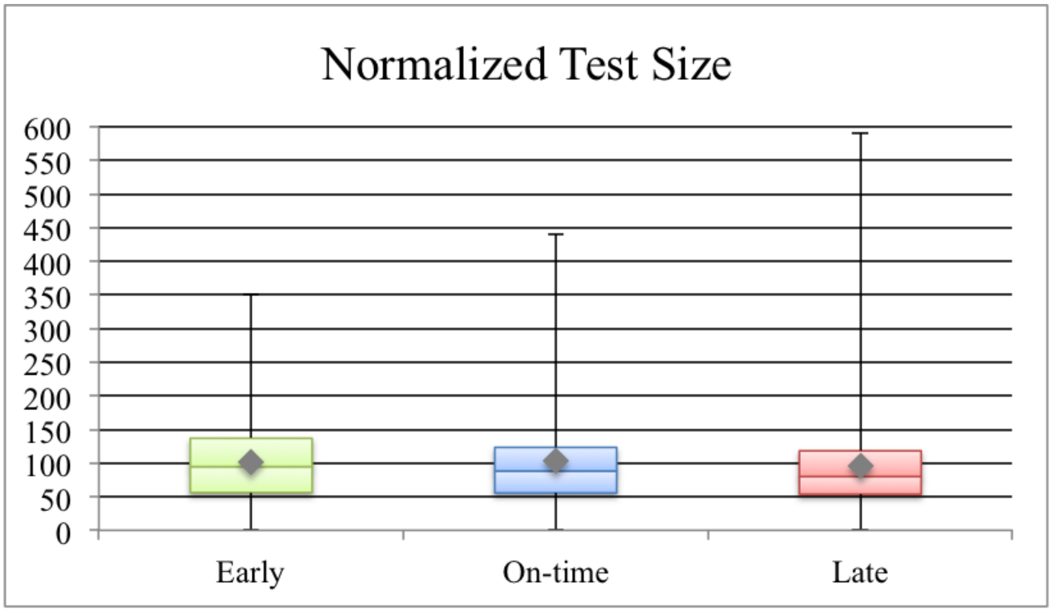


Figure 6.24: The normalized test code size for each on-time status category.

Analyzing the impact of on-time status on the test size of a submission, we find no significant relationship (F ratio 0.76, P <.47). This indicates that students who submitted both early and late typically wrote similar amounts of code to test their solutions.

This analysis is surprising, as we found a significant relationship between early submissions and solution size. Since test size and solution size are strongly correlated, we reasoned this value would also be significantly greater for early submissions. A lack of a significant relationship could simply indicate students tested there code roughly the same, regardless of when they submitted.

6.2.7 Late Impact on Submission Number

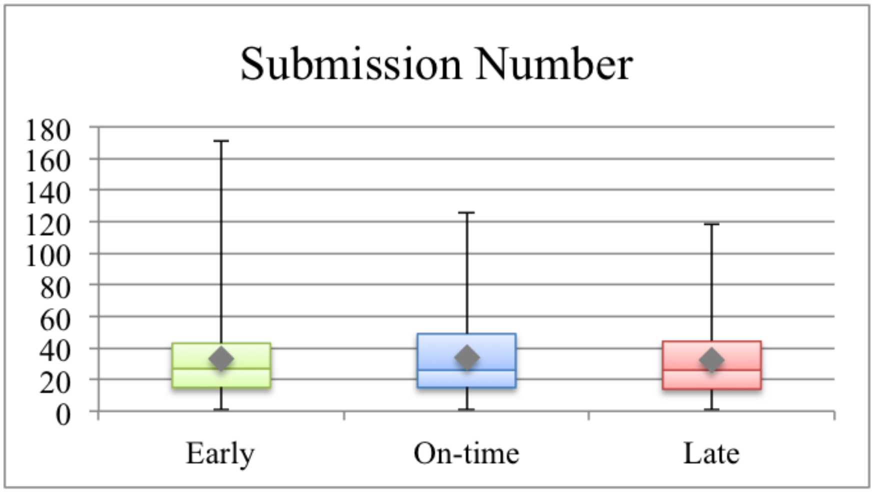


Figure 6.25: The mean number of submissions for each on-time status category.

When analyzing the impact of on-time status on the number of submissions to Web-CAT, we find no significant relationship (F ratio 0.54, P <0.58). Expanding our analysis to examine any potential relationship within each quartile, we still find no significant relationship.

As discussed we discussed previously when we examined the impact of treatments on the number of submissions to Web-CAT, the number of submissions can be viewed as a measure of quality. When we examined the entire dataset, including students who had dropped or withdrew, we found the number of Web-CAT submissions to correlate with the a higher reference test pass rate. However, running the analysis on our own dataset where the students who dropped or withdrew were removed from the data set, we found no relationship between the number of Web-CAT submissions and any solid indicator of quality. Because we did not find that earlier submissions showed a significantly higher number of Web-CAT submissions, but we have shown that earlier submissions are of higher quality, it can be said the raw number of submissions to Web-CAT is not an indicator of overall quality.

6.2.8 Procrastination Tendency Impact on Lateness

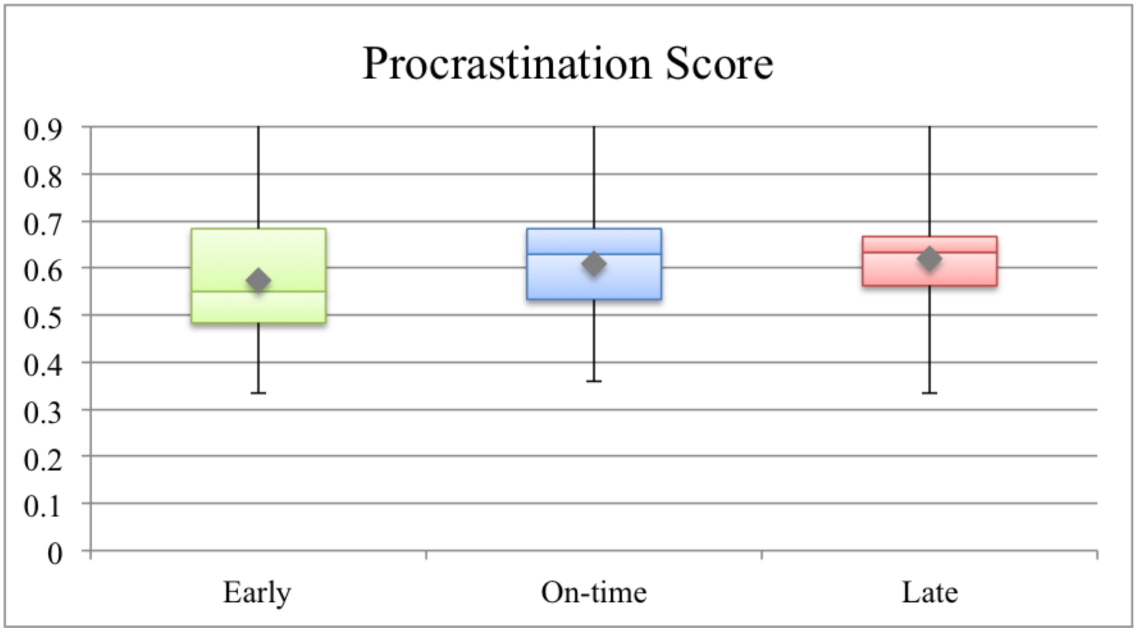


Figure 6.26: The score of a procrastination survey for each on-time status category of the project. The higher the score, the more likely a student was to procrastinate.

Before each of the semesters, we issued a survey to measure a student's tendency to procrastinate. Analyzing the aggregate response to the survey questions, we gain a key metric to measure the impact of lateness. The value is higher if students are considered more likely to procrastinate, while it is lower if students are considered less likely to procrastinate. Analysis does reveal a significant relationship between the procrastination score and the likelihood a project will be submitted late ($\chi^2 = 7.67$, P <.001). The analysis found that the higher the procrastination tendency score, the more likely a submission was submitted late.

This analysis confirms our assumptions about this procrastination scale. It indicates that this survey examination is a good tool for generally measuring how students will procrastinate in a course.

6.2.9 Combined Start Time and Work Rate Examination

One of the potential problems with our analysis is the lack of collected data as students develop their solution. Using Web-CAT, we are only able to collect information as students make submissions to the system. In theory, a student could work independently until the deadline neared and make one submission to Web-CAT that satisfied the requirements of the assignment. Under our examination of start time, students who were working would instead

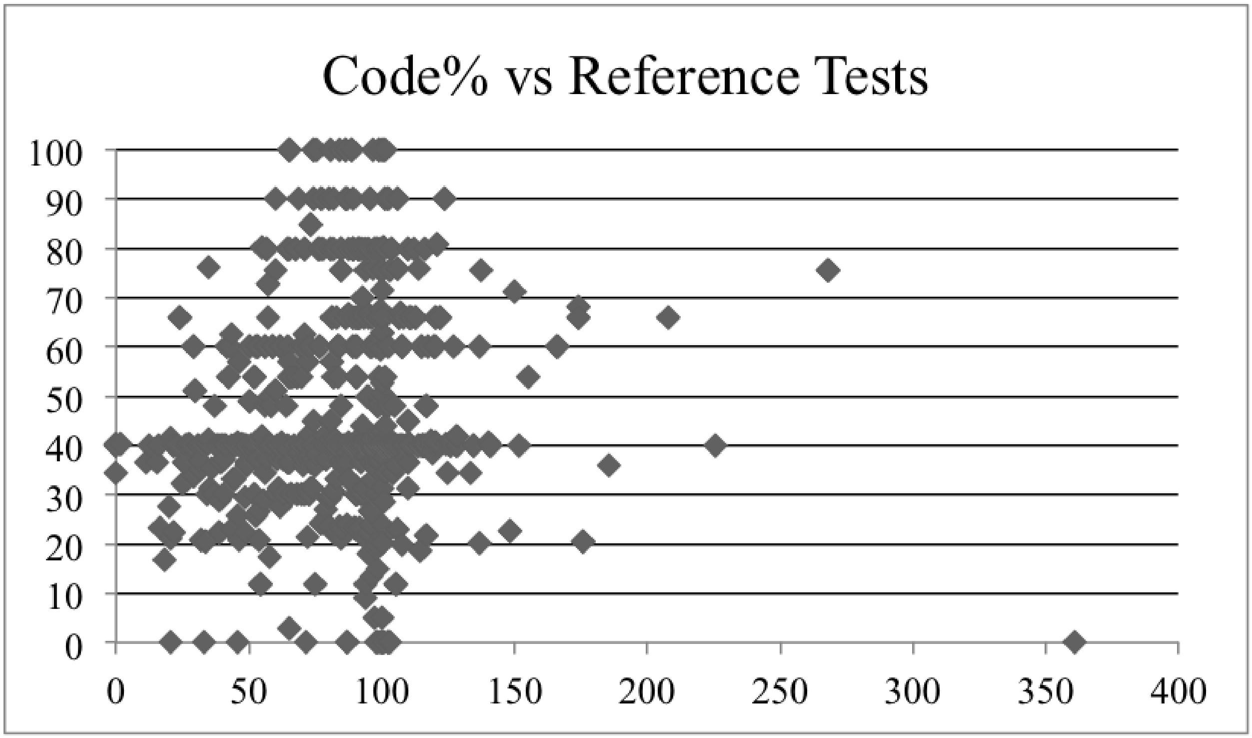


Figure 6.27: First submission solution size ratio versus the percentage of reference tests passed.

be considered procrastinators. In an attempt to measure how much work students completed before they began submitted to Web-CAT, we created a ratio of the solution size recorded by the first Web-CAT solution and the solution size recorded for the final submission. This value gives us a rough estimate of how much of a solution a student has completed when they make their first Web-CAT submission. Combining this with the time a first submission was made gives us a metric that is higher for students who submit their work earlier and lower for students who submit their work later.

We analyzed the impact of this metric against the reference test pass rate to see if this was a good indicator of overall submission quality. We also examined solution size and first submission time individually against reference test pass rate. Overall we found that none of these factors had a solidly significant relationship with the pass rate of reference tests. Eliminating the Fall 2013 semester group, to reduce the influence any poorly written reference tests might have, we still found no significant relationship. As our results our inconclusive, it seems attempting to measure the amount of student work in addition to when they begin submitting to Web-CAT does not yield a viable method of examining how great or little a student is procrastinating.

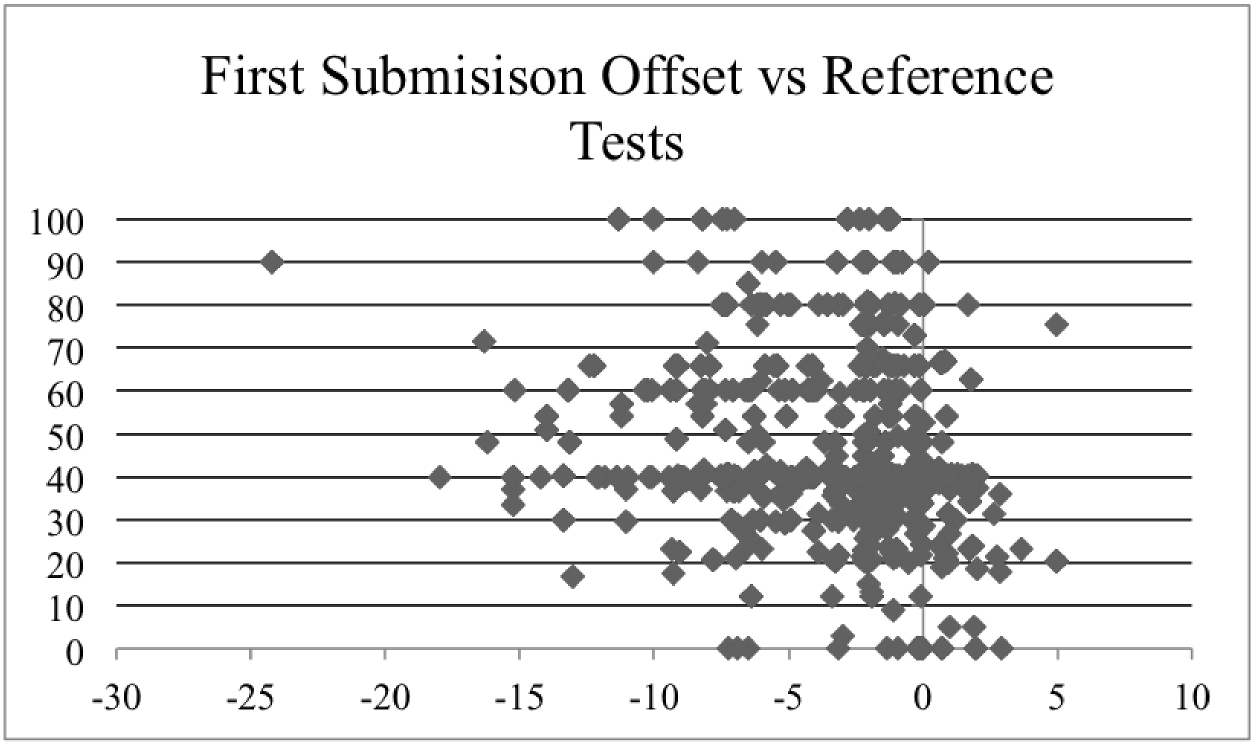


Figure 6.28: First submission time versus the percentage of reference tests passed. A negative value indicates an earlier submission, while a positive value indicates a later submission.

6.3 Analysis Limitations

6.3.1 Semester Influence

Our analysis was limited by a variety of factors we were unable to control for. One such limitation was the potential semester influence. While the sections in each of the different semesters had many common elements, such as the same instructor, lead GTA, and the same overall number of projects, the instructor did feel the projects for the 2014 semester were more difficult compared to the 2013 semester. Furthermore, the projects had a different complexity component between semesters. While the projects in 2014 were designed to be of similar complexity to those in 2013, the instructor did notice that students had a greater challenge in 2014 completing the projects compared to students in 2013. This is evidenced by the larger number of students who dropped or withdrew from the course, which was discussed at the beginning this chapter. Finally, as discussed when analyzing the impact of treatments on the reference test pass rate, the quality of reference tests were different between the semesters. This was due to the experience of the GTA who was writing the tests, and probably negatively influenced the pass rates of students in the 2013 section.

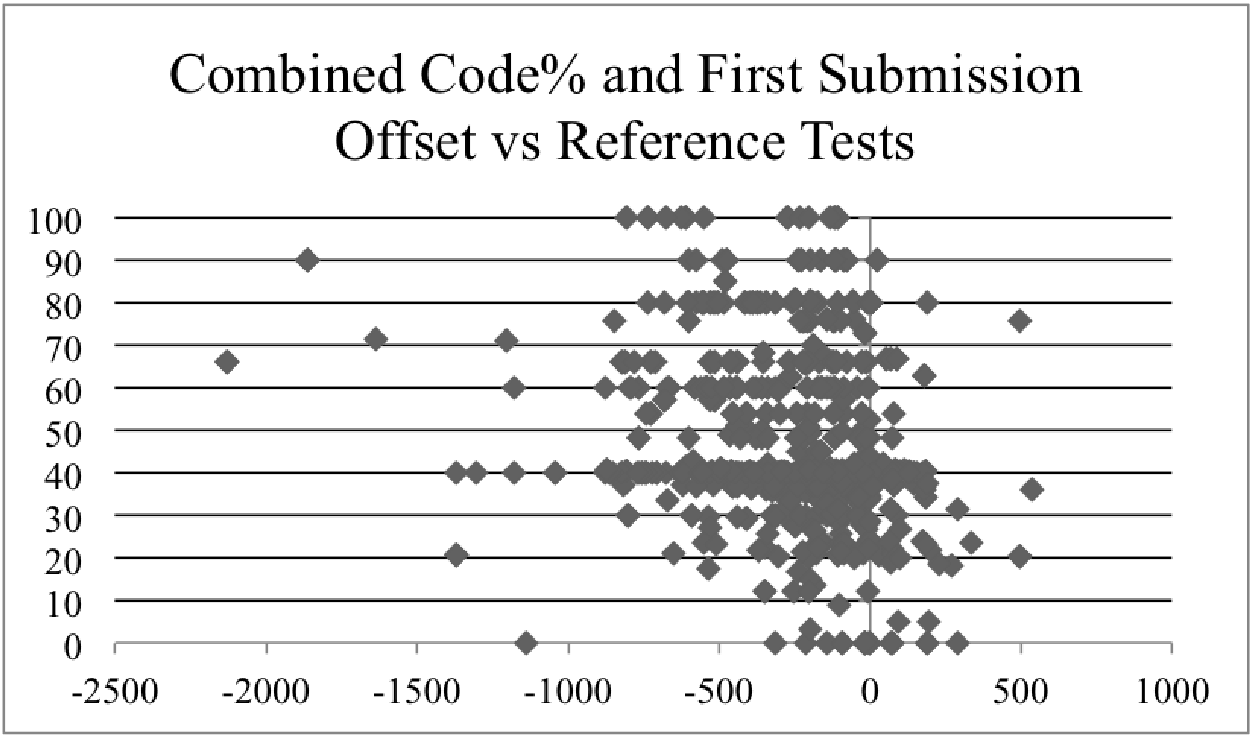


Figure 6.29: Combined first submission time and first submission solution size versus the percentage of reference tests passed.

6.3.2 Assignment Influence

Overall, the projects for the course we used to test our interventions are designed to be easily completed so long as a student understands the core concepts behind them. These projects are used as teaching tools, not as an example of a complex real-world problem which might require a larger effort from students. The lack of large challenge with these projects might skew the overall influence of our interventions, as students might be more able to not manage their time appropriately but still succeed in the course. This would limit our observation of any significant effect these interventions might have on time management. While we cannot eliminate this potential influence from our analysis, it is an important factor to consider when examining our results.

6.3.3 Inherent Student Quality

Our analysis was designed to avoid the influence of the "good student effect". The idea behind this concept is that some students will do inherently well regardless of almost any external factor. These "good students" could throw off any analysis by simply outperforming another group of students, despite their performance not being related to any of the interventions we studied. In order to limit this effect, we performed a within subjects comparison when analyzing the impact of late submissions by reducing our analysis to students who submitted assignments both early and late. While that analysis accounted for students of different aptitude, it did not account for differences between the assignments themselves. We also had this problem with our quartile analysis. The quartiles did account for assignment differences, as the quartiles contained results from all assignments. However, it could be that the final semester grade received by a student is not the best indication of their overall quality. This may hinder the accuracy of our analysis.

6.3.4 Web-CAT Data Limitations

One of the biggest limitations of our analysis came from the inherent limitations of the Web-CAT grading system. Web-CAT is a powerful tool that collects a large amount of data each time a student submits work for a project. However, Web-CAT is limited to collecting data when students submit to the system. Theoretically, students can work completely independently of Web-CAT and submit their work right before a project is due. This would not allow us to accurately predict the level of efforts students are placing into each project. The only way to overcome this limitation is to collect information as students develop, ideally directly from their IDE. We discuss this potential in our chapter on future work.

Chapter 7

Conclusions

In this paper, we have described three different interventions designed to reduce student procrastination in computer science classrooms. These interventions are designed to be scalable, and therefore can be easily implemented in other programming based courses.

7.1 Experimental Results

Based on our analysis of the impact of these interventions, it seems the e-mail alert intervention positively impacted when students submitted their final work to Web-CAT. Furthermore, the e-mail alert intervention also had an impact on when students began submitting to Web-CAT, which is an indicator of when these students began working on their projects. Students starting their work earlier indicates a reduction in procrastination behavior, which can be seen as a success for the e-mail alert intervention. However, students did subjectively report they did not appreciate the treatment and felt it was a waste of time. This indicates there may be additional work required to modify the e-mail alerts so students feel more is gained from them.

Trying to analyze the impact of these interventions on project quality was a harder task. Overall, we did find submissions in some treatment groups, particularly the schedule sheet treatment group, to be of higher quality compared to the others. However, the influence of poorly written reference tests on the Fall 2013 semester makes this analysis suspect. Trying to eliminate that source of bias might allow us to confirm these results.

We were able to confirm that work submitted later was of lower quality, especially when compared to earlier submitted work. This is an important confirmation as it underscores the importance of students starting early, as well as the threat procrastination poses to student success.

7.2 Contributions

The primary contribution of this work is the creation and analysis of three scalable techniques to reduce student procrastination. Previous work shown to reduce procrastination has not been designed to be effectively implemented in large classrooms. With large numbers of students procrastinating, techniques to reduce procrastination will be more effective if they can be easily targeted to more students. This research shows a way to make this happen. Furthermore, our success with e-mail alerts introduces a novel intervention to reduce procrastination that can be built upon in future research.

7.3 Future Work

Overall, our positive findings for these treatments indicate that they can be used as a valuable tool to reduce student procrastination, although most likely in a modified form. Using active collection to measure precisely how successful students program while unsuccessful students fail could vastly improve the e-mail alert intervention by improving the messages students receive. Instead of most students receiving a default message indicating their failure to submit to Web-CAT, they could instead receive nuanced messages containing relevant statistical information on their current progress relative to ideal progress. Instructors would also benefit from this information, as they could quickly identify struggling students and intervene before the impact of procrastination became irreversible.

One way to collect this information would be to monitor student programming behavior within their IDE. Currently, we only examine student submissions to Web-CAT. While this provides valuable information, it does not record certain fine grained details that would be captured within an IDE. With this capability in addition to the Web-CAT data, we could accurately model a successful student throughout the semester. With this model, more accurate e-mail alerts could be sent to students, perhaps increasing their own appreciation of the intervention as well as its effectiveness.

Bibliography

[1] Allevato, A., and Edwards, S. H. The effects of extra credit opportunities on student procrastination. In *Frontiers in Education Conference, 2013 IEEE* (2013), IEEE, pp. 1831–1836.

[2] Allevato, A., Thornton, M., Edwards, S. H., and Pérez-Quiñones, M. A. Mining data from an automated grading and testing system by adding rich reporting capabilities. *EDM 8* (2008), 167–176.

[3] Bernstein, P. L., and Bernstein Peter, L. *Against the gods: The remarkable story of risk*. Wiley New York, 1996.

[4] Bonwell, C. C., and Eison, J. A. *Active Learning: Creating Excitement in the Classroom. 1991 ASHE-ERIC Higher Education Reports.* ERIC, 1991.

[5] Bridges, K. R., and Roig, M. Academic procrastination and irrational thinking: A re-examination with context controlled. *Personality and Individual Differences 22*, 6 (1997), 941–944.

[6] Briody, R. *An exploratory study of procrastination.* PhD thesis, ProQuest Information & Learning, 1980.

[7] Edwards, S., Martin, J., and Shaffer, C. A. Examining classroom interventions to reduce procrastination. In *In Proceedings of the 2015 Conference on Innovation & Technology in Computer Science Education (ITiCSE '15)* (NY, NY, USA, 2015), ACM. to appear.

[8] Edwards, S. H., and Perez-Quinones, M. A. Web-cat: automatically grading programming assignments. In *ACM SIGCSE Bulletin* (2008), vol. 40, ACM, pp. 328–328.

[9] Edwards, S. H., Snyder, J., Pérez-Quiñones, M. A., Allevato, A., Kim, D., and Tretola, B. Comparing effective and ineffective behaviors of student programmers. In *Proceedings of the fifth international workshop on Computing education research workshop* (2009), ACM, pp. 3–14.

[10] Ferrari, J., Johnson, J., and McCown, W. *Procrastination and Task Avoidance: Theory, Research, and Treatment.* Issues in Clinical Child Psychology. Springer, 1995.

[11] Ferrari, J. R. Compulsive procrastination: Some self-reported characteristics. *Psychological reports 68*, 2 (1991), 455–458.

[12] Foundation, T. E. *BIRT Project: Business Intelligence and Reporting Tools*, 2008.

[13] Klassen, R. M., Krawchuk, L. L., and Rajani, S. Academic procrastination of undergraduates: Low self-efficacy to self-regulate predicts higher levels of procrastination. *Contemporary Educational Psychology 33*, 4 (2008), 915–931.

[14] Lay, C. H. At last, my research article on procrastination. *Journal of research in personality 20*, 4 (1986), 474–495.

[15] Milgram, N. N., Mey-Tal, G., and Levison, Y. Procrastination, generalized or specific, in college students and their parents. *Personality and Individual Differences 25*, 2 (1998), 297–316.

[16] Milgram, S., Sabini, J. E., and Silver, M. E. *The individual in a social world: Essays and experiments* . Mcgraw-Hill Book Company, 1992.

[17] Rothblum, E. D., Solomon, L. J., and Murakami, J. Affective, cognitive, and behavioral differences between high and low procrastinators. *Journal of counseling psychology 33*, 4 (1986), 387.

[18] Schouwenburg, H. C., and Lay, C. H. Trait procrastination and the big-five factors of personality. *Personality and Individual Differences 18*, 4 (1995), 481–490.

[19] Shaffer, C. A., and Edwards, S. H. Scheduling, pair programming and student programming assignment performance. *IEEE Transactions on Education 1*, 11 (2007), 1–6.

[20] Spolsky, J. *Joel on Software.* Apress, 2004.

[21] Stead, D. R. A review of the one-minute paper. *Active Learning in Higher Education 6*, 2 (2005), 118–131.

[22] Steel, P. The nature of procrastination: a meta-analytic and theoretical review of quintessential self-regulatory failure. *Psychological bulletin 133*, 1 (2007), 65.

[23] Tice, D. M., and Baumeister, R. F. Longitudinal study of procrastination, performance, stress, and health: The costs and benefits of dawdling. *Psychological Science* (1997), 454–458.

[24] Tuckman, B. W. The development and concurrent validity of the procrastination scale. *Educational and psychological measurement 51*, 2 (1991), 473–480.

[25] Tuckman, B. W. Relations of academic procrastination, rationalizations, and performance in a web course with deadlines 1. *Psychological Reports 96*, 3c (2005), 1015–1021.

[26] Tuckman, B. W., and Schouwenburg, H. C. Behavioral interventions for reducing procrastination among university students.

Appendix A

Sample E-mail Alerts

The following appendix section contains samples of e-mail alerts sent to students. The alerts were part of the e-mail alert intervention. The description of how the alerts were generated and context for their meaning can be found in chapter 5, section 2.

A.1 E-mail Alert Example: Student with No Web-CAT Submissions

Subject: CS 3114: You may be at risk on Project 1

This notification is to increase your awareness of your current progress on Project 1 compared to the rest of the class. Project 1 is due in 7 days.

You have not submitted any work to Web-CAT for analysis on Project 1. As a result, there is no way to assess your progress relative to the rest of the class, indicating you may be at risk for performing poorly on the assignment.

Historical data from many programming classes like this one indicate that when a student starts earlier, he or she earns higher scores on average than when the same student starts later.

If you have not begun work on the project, it is important to do so as soon as possible. If you have already started work, we encourage you to submit to Web-CAT and self-check your work early and often. Remember to incrementally test as you write code as well, since leaving testing for later also increases the risk of performing poorly.

We wish you the best of luck as you work to complete this assignment.

– Web-CAT Situational Awareness Service

A.2 E-mail Alert Example: Student Making Insufficient Progress

Subject: CS 3114: You are at risk on Project 1

This notification is to increase your awareness of your current progress on Project 1 compared to the rest of the class. Project 1 is due in 4 days.

Based on the work you have submitted to Web-CAT, it looks like you are behind in your progress on this assignment. This increases your risk of performing poorly. If you are stuck, you should visit a member of the course staff for assistance in getting back on track.

Based on the tests you have submitted, it appears that you may be waiting until later to test your work, instead of testing it incrementally as you develop it. Typical students earn higher scores when they write tests incrementally with their code compared to when the same student waits to write tests until the code is substantially complete. This increases your risk of performing poorly on the assignment.

We wish you the best of luck as you work to complete this assignment.

– Web-CAT Situational Awareness Service

A.3 E-mail Alert Example: Student Making Good Progress

Section: CS 3114: Your progress on Project 1

This notification is to increase your awareness of your current progress on Project 1 compared to the rest of the class. Project 1 is due in 4 days.

Based on the work you have submitted to Web-CAT, it looks like you are making good progress towards a working solution for this assignment. Starting early is associated with a statistically significant increase is scores earned, compared to when the same student starts later. This increases your chances of success on the assignment.

Based on the tests you have submitted, it appears that you may be waiting until later to test your work, instead of testing it incrementally as you develop it. Typical students earn higher scores when they write tests incrementally with their code compared to when the same student waits to write tests until the code is substantially complete. This increases your risk of performing poorly on the assignment.

We wish you the best of luck as you work to complete this assignment.

– Web-CAT Situational Awareness Service

Appendix B

Reflective Writing Assignment

B.1 Reflective Writing Prompts

This assignment involves writing one-paragraph responses to four questions regarding the last programming assignment you completed for this class, and should only take you about 15-20 minutes to complete.

Consider the following questions and answer each one in a separate paragraph.

1. On the last software project you worked on for a class, consider when you began working on it, the way that you spent your time, and how you managed yourself as you work to complete the project. Then describe the key elements of the plan or strategy you used to manage your time on that project.

2. Again considering the most recent software project you completed for a class, describe how your strategy for managing your time on the project affected the quality of your work or your ability to achieve your goals on the project.

3. Reflecting on your experience with past projects, describe the strategy or plan you intend to use for managing your time on the next software project that you will do for this class.

4. Again considering the most recent software project you completed for a class, a) describe your development strategy that you used to implement and test the program, b) describe how your development strategy affected the quality of your work or your ability to achieve your goals on the project, and c) describe any different strategy to implement and test your program that you will use for the next software project that you will do for this class.

Appendix C

Schedule Sheet Assignment

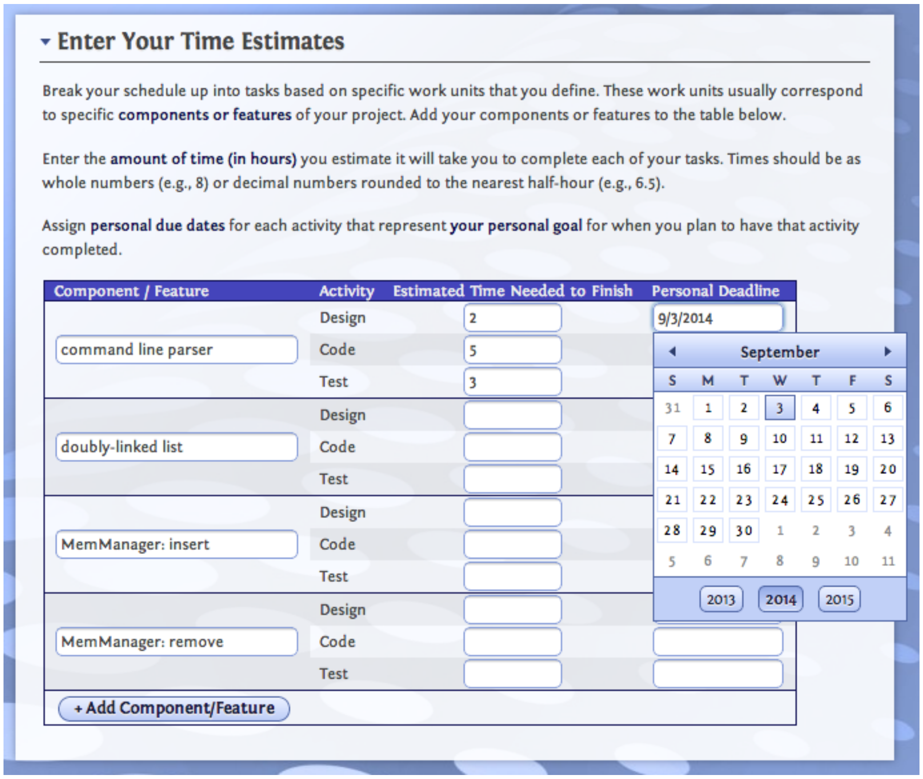
C.1 Schedule Sheet Entry Instructions

For all projects this semester, you will be required to create and update a schedule for the project. For each project, there will be three phases.

1. Toward the end of the first week of the project life cycle, you will create an initial schedule.
2. About a week before the project is due, you will create a revised schedule.
3. When you make your the final submission for your project, you will submit the final sheet to document your time spent.

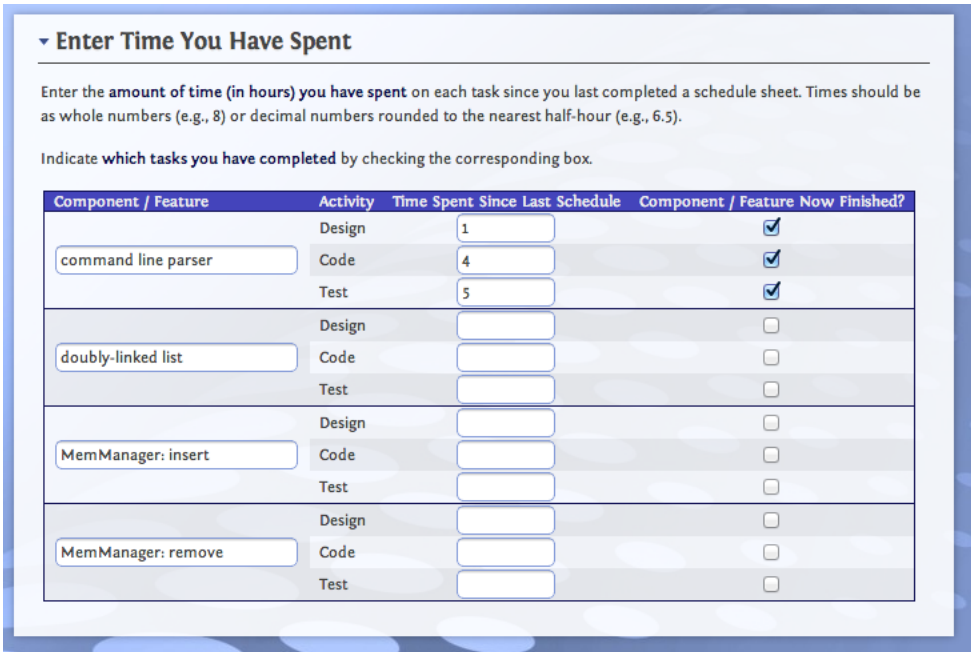
The schedule sheets themselves are online forms created at Web-CAT.

In the first phase, you will break your project down in to as many identifiable "pieces" as you can. The more pieces, the better. The first schedule looks like this:

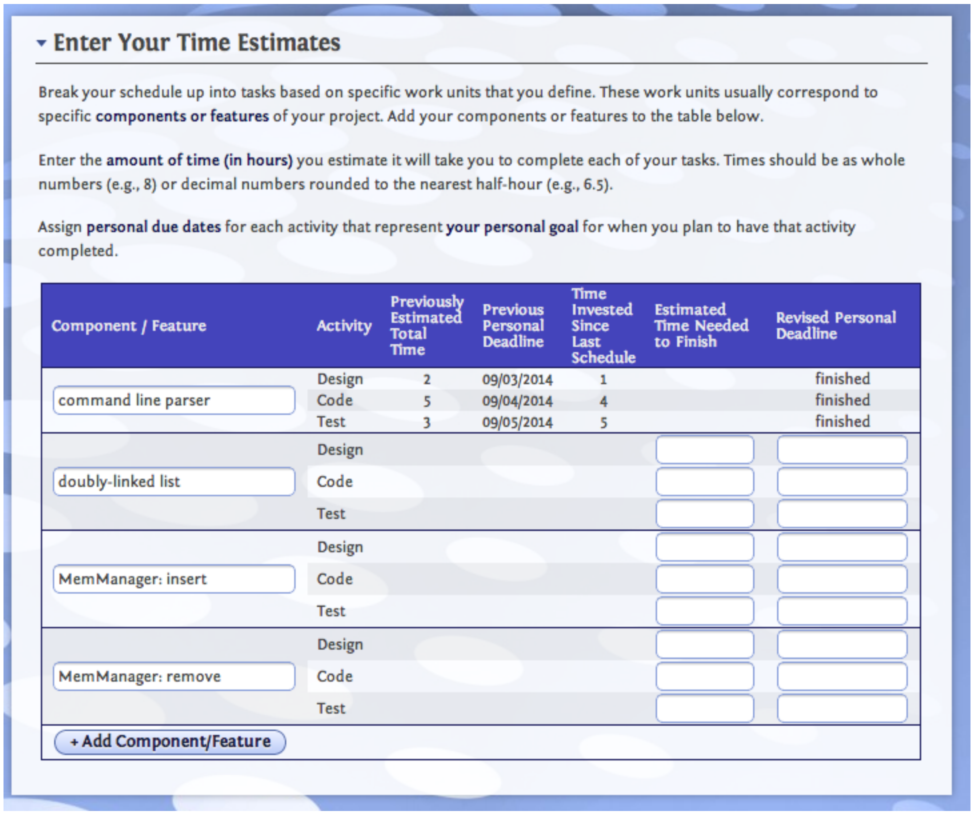


Explicitly separating "phases" (design/code/test) for each task helps you to explicitly consider the necessary effort for each of these separate activities. Feel free to add other phases as appropriate, perhaps such as "debug" separate from "test".

On the later schedules, you will fill out two forms. The first looks like this and is only used for reporting effort expended so far:



The final column in that sheet just has a check box to mark the feature as complete or not. After entering the effort expended, and what is complete, you get a second page to enter new estimates:



Enter Your Time Estimates

Break your schedule up into tasks based on specific work units that you define. These work units usually correspond to specific **components or features** of your project. Add your components or features to the table below.

Enter the **amount of time (in hours)** you estimate it will take you to complete each of your tasks. Times should be as whole numbers (e.g., 8) or decimal numbers rounded to the nearest half-hour (e.g., 6.5).

Assign **personal due dates** for each activity that represent **your personal goal** for when you plan to have that activity completed.

Component / Feature	Activity	Previously Estimated Total Time	Previous Personal Deadline	Time Invested Since Last Schedule	Estimated Time Needed to Finish	Revised Personal Deadline
command line parser	Design	2	09/03/2014	1		finished
	Code	5	09/04/2014	4		finished
	Test	3	09/05/2014	5		finished
doubly-linked list	Design					
	Code					
	Test					
MemManager: insert	Design					
	Code					
	Test					
MemManager: remove	Design					
	Code					
	Test					

+ Add Component/Feature

In this table, all but the rightmost two columns will already be filled in, and you will only enter the last two columns (and only for unfinished tasks–finished tasks wouldn't require any info).

The second page will contain appropriate highlighting/etc. to point out missed personal deadlines, places where time invested exceeds the previous estimate, etc.

When you enter your "final" schedule after finishing the assignment, you will fill out only the first table (time spent since last schedule) rather than both tables.