CHAPTER 8

COLLABORATIVE GAME-BASED LEARNING WITH iPADS AND EXTERNAL KEYBOARDS IN A WEB DEVELOPMENT CLASS

AEKATERINI MAVRI
FERNANDO LOIZIDES
NICOS SOULELES

Abstract

Employing the iPad’s seductive powers to facilitate active learning through stimulating intrinsic student interest and providing richer interactive experiences, while enjoying easy mobility and flexibility of use in multiple teaching settings, has been a popular pedagogical practice over the past few years. Previous work investigating the use of iPads for active learning in the field of web design and development in Higher Education (HE) infers that student experience and perceived learning performance was hindered by the absence of a direct point and manipulation device (i.e., a mouse), as well as cumbersome text-input activity on the touch interface.

As various peripherals—aiming to transform mobile phones and tablets into production devices—are constantly being released, this study re-examines the use of iPads in the same field, by pairing external keyboards to the devices. In doing so, it seeks to elicit results about the user experience as well as the perceived student learning process and outcomes.

1 Aekaterini Mavri, Cyprus Interaction Lab, Department of Multimedia and Graphic Arts, Cyprus University of Technology
Fernando Loizides, Emerging Interactive Technologies Lab, Department of Maths and Computer Science, Faculty of Science and Engineering, University of Wolverhampton, UK
Nicos Souleles, Art + Design: eLearning lab, Department of Multimedia and Graphic Arts, Cyprus University of Technology
in performing practical coding exercises. Apart from the external keyboards, two additional parameters are now present: (a) a larger class size and (b) game-based learning. Results indicate a significant perceived improvement in coding performance and clear student preference in favor of the external versus the native keyboard. Issues derived from small key sizes, non-standardized layout, and unconventional key combinations intended for critical functions, were recorded. The friendly competitive-collaborative approach was found to enhance the learning process. Learners showed a strong preference towards laptops, since they support multi-tasking and direct manipulation capabilities, over the iPad’s notable qualities such as portability and convenience.

**Keywords:** Collaboration, mobile-enhanced learning, code development

**Acknowledgments**

The authors acknowledge travel funding from the European Union's Horizon 2020 Framework Programme through NOTRE project (H2020-TWINN-2015, Grant Agreement Number: 692058).

This research was also supported by resources and equipment from the ‘Art + Design: elearning lab’ (www.elearningartdesign.org), at the Department of Multimedia and Graphic Arts, Cyprus University of Technology.
Introduction

This study aims to contribute to existing work in the area of active collaborative learning with mobile devices, as an instructional method in an undergraduate Web Design and Development class.

Active collaborative learning, through practical exercises and related activities, iterated with short taught sessions in the classroom, is known to motivate students and assist them in learning, while forming constructive social interactions and relationships (Burguillo, 2010; Johnson, Johnson, & Smith, 1991). Previous work in the field of web development agrees that this pedagogical method is known to both stimulate student interest and participation as well as maximize the perceived learning outcomes (Mavri, Loizides, & Souleles, 2014). Issues associated with poor usability in using iPads for code development, however, seem to act at the cost of this instructional approach and overall user experience is consequently compromised. Amongst others, one of the most serious concerns is the absence of the point-to-click capability in targeting specific on-screen items, as well as the problematic affordances of the touch screen for text input (Mahmoud & Popowicz, 2010). Web development or coding is significantly dependent on typing efficiency; programmers consider themselves typists first and programmers second; a competency that allows them to type in response to their running cognitive processes. In order to address such issues while utilizing portability, different types of small-sized, lightweight physical keyboards have been released in the market, for pairing with a variety of mobile devices (Chaparro & Phan, 2014).

In light of the above, in this work, we repeat the former experiment, by introducing physical foldable keyboards attached to the iPads, to address text-entry as well as direct manipulation (i.e., point and select) issues, during coding. As programming learners are in favor of more in-class practical exercises (Hill, Ray, Blair, & Carver, 2003), which are also known to reinforce the education objectives, the active learning protocol is carried forward in the recent study. In introducing game-based learning, both through collaboration and competition, with a larger group of students, we aim to elicit information in regards to the user experience as well as the perceived impact on students’ learning outcomes. This research is guided by the following questions:

1. What is the contribution (if any) of the external keyboard addition, during in-class coding and browsing activities, compared to the native touch keyboard on the iPad?
2. How does in-class small-team collaboration within a friendly competitive setting affect the student experience and performance in learning web development?

3. What is the impact (if any) of a larger group size on classroom-based active learning methods?

The next section looks at existing work in the area and related fields of active, collaborative learning as well as competition, with mobile devices and peripherals in the classroom. Following that, the study design is explained. The analysis section presents both quantitative and qualitative outcomes, examined by order of importance in three subsections: the use of external keyboards, iPad as a device (hardware and software), and the active learning methods employed. Finally, key findings are summarized and analyzed in the discussion and conclusion sections.

**Literature Overview**

Collaboration in education is employed as an effective method in order to enhance the learning and assimilation process in students who share common interests and goals. Educational psychology embraces social interdependence theory as a base model for evaluating interactions within a group (Coleman, 2011; Johnson & Johnson, 2009). Interdependence—within a collaborative setting—can be positive (collaboration), negative (competition), or absent (individualistic). Team members with common goals have positive interdependence and through this, they obtain higher levels of achievement, psychological satisfaction, and form better quality relationships. These are realized through effective actions, positive cathexis (positive psychological energy for peers), substitutability (substituting for one another’s actions), inducibility (allowing for being influenced by and influencing others), and promotive interaction (encouraging and facilitating each other’s efforts). Collaboration is effective, especially in small groups, by encouraging sufficient individual responsibility, involvement, and accountability in order to achieve up to a preset criterion (Shindler, 2010). It ensures that team members work together to improve not only their own learning process, but that of their peers too since individual success leads to team success (Burguillo, 2010; Cantador & Bellogín, 2012). This practice is particularly common in programming education and practice (Hill et al., 2003). Working in groups of twos, or ‘pair programming’ is known to achieve higher code quality, in the same period of time needed by “solo programmers” (Stotts, Williams, & Nagappan, 2003, p. 2).
The positive impact of collaboration through pair programming is also evident in studies conducted in education (Mahmoud & Popowicz, 2010; Nagappan et al., 2003; Salleh, Mendes, & Grundy, 2011). However, as the serious struggles of beginner programmers are known to hinder their overall learning experience, it becomes inherently crucial to seek for other, best practices to revert such outcomes. Jenkins (2002) argues that the reason behind the difficulties in learning to program is the blend of required learning styles: surface learning for remembering elements such as syntax and order of precedence, and deep learning in the understanding of concepts and development of true competence. Thus, programming cannot only be through reading books and theory, but through hours of practice; it is after all a problem solving activity. Taking a step further, it also seems that programming students are reluctant to invest in practice time, unless they are obliged to (Hill et al., 2003). Classroom-based hands-on experience is therefore considered to be a highly effective constructivist approach (Jerinic, 2015; Whittington, 2004) since it provides the chance to timely reinforce concepts in greater depths than one-way lecturing does.

Student motivation and engagement in the learning process can be further promoted, when the hands-on methods include elements of ‘game-playing’ as part of a friendly competitive setting (Burguillo, 2010; Verhoeff, 1997). In this context, known guidelines for healthy game competitions include (Hill et al., 2003): (a) game objectives adequately aligned with learning objectives; b) simple, quick in-class implementations; (c) clear and straightforward game rules; and (d) play-testing prior to class, to ensure that learning is not overwhelmed by the operating ‘mechanics’ of the game. It is worth noting here that intergroup competition is still considered to be a collaborative condition (Johnson, Johnson, & Stanne, 2000). Additionally introducing mild competitions, with a symbolic prize, minus the negative aftermath of losing (Shindler, 2010) can encourage even weak students to take part into the process and persistently contribute to the group effort (Cantador, 2015). The time constraint factor that typically comes with class competitions is also worth looking at, for procrastination and lack of effective time management are major issues affecting (mostly) undergraduate students. Negative procrastination (i.e., postponing actions, in spite of resultant negative consequences), is often associated with learning disabilities and lower levels of self-time-regulation and self-efficacy (Edwards, Martin, & Shaffer, 2015; Hen & Goroshit, 2014). This condition is particularly prominent in Computer Science/programming education where task aversion is common due to assignment complexity. It is one of the main factors contributing to poor performance and course failures. Collective
time management through enforced deadlines, as a classroom intervention, can therefore help alleviate this problem.

The use of mobile devices can facilitate as well as expedite the collaborative/competitive learning model through support of real-time student activities (i.e., the real time projection of the scaffolding state in solving tasks; Herreid & Schiller, 2013). As well as offering an exciting platform to work from (Shudong & Higgins, 2006), mobile devices are important in web design education as development is increasingly focusing on the ‘mobile-first’ approach (Gardner, 2011); they provide instant gratification of testing out solutions on the intended medium (Tillman et al., 2012). Aside from their ‘seductive powers’, mobile devices in education have received mixed outcomes in existing research.

In regards to iPads in particular, technical issues relating to small screen size, limited memory, lack of multi-tasking, and awkward text input may appear overwhelming for learners (Budiu & Nielsen, 2011). The known read-to-tap asymmetry, lack of fine navigation, unintended touch activity, and slow typing performance are the cause of serious problems in learning programming (Chaparro & Phan, 2014; Mavri et al., 2014; Tillman et al., 2012). Lacking tactile feedback, touchscreen keyboards require the user’s visual attention during typing (Findlater & Wobbrock, 2012). Since dominant trends inherently require mobile devices to undertake a more production-oriented role, research is being carried out to make these more text-friendly, through multi-gestural, adaptive touch keyboards. On the other hand, producer companies address these issues by building peripherals to pair with the devices: Chapparo and Phan (2014) mention that the ‘markets are flooding with ‘petite’, lightweight and unobtrusive physical keyboards to match mobile devices. By comparing typing performance and user satisfaction between pressure-sensitive versus mechanical keyboards, the same study found that the latter scored higher in both parameters, indicating a lower user mental load during typing. However, there is little evidence on the use and impact of pairing peripherals to mobile devices in educational contexts.

**Study Design**

This study builds on previous work conducted in an undergraduate (3rd year) Web Design and Development module (Mavri et al., 2014). The general outcomes pointed towards a positive attitude towards the active learning model, yet, indicated a series of device-related issues, the two most prominent being the problematic (touch) text entry as well as the inability to point-and-click on specific elements.
In this study, we look at the use of iPads with an added external mini keyboard (Bluetooth Genius LuxePad i9010)—a mechanical QWERTY type keyboard—that can be clipped and folded onto the device. The keyboard is primarily designed for the iPad mini and not fully compatible with the iPad 3 model used in this study. Students were advised to use the device’s single magnetic hinge side for the cover case, and fold the cover case into an upright positioning support base, then place the mini keyboard at the front (Figure 8.1) of the device to create a comfortable working setup.

![Figure 8.1. iPad (upright position) and external keyboard setup.](image)

Students were also prompted to login to the Facebook lesson page, where exercises were to be successively posted, following short, taught sessions. Additionally, they were asked to agree to a consent form as well as complete an online survey, in regards to previous experience and general bias towards the device, before the study.

Following this stage, the instructor explained the lesson structure and study outline: iterated cycles of briefly taught sessions followed by practical tasks that required students to (a) solve exercises on recently taught material and (b) search for online resources for support and assistance. Information on key combinations for accessing specific characters was provided, and students were prompted to practice typing on the external keyboard for a few minutes before the study.

In order to stimulate interest and motivation, exercises of varying degrees of difficulty were introduced, as part of a scaffolding approach: the first-in-line were less demanding in terms of coding complexity; they also contained more initial information in place. The following exercises utilized previous knowledge joined with newly acquired information. A third iteration also required locating missing clues, which could only be found through online search. The lecture notes were not made available
during the lesson, in order to prevent students from outrunning the process and looking for answers from following slides. Again, this necessitated further online browsing, in search of correct terminology and syntax—a deliberate attempt to cultivate student-initiated research versus an available resource pool.

This study takes place with a twice as large group—20 instead of 10 students. Following a pilot test, the research team decided against using the single-projected exercises model conducted during the previous study, as it was deemed unsuitable for reasons explained in the ‘Pilot Tests’ section.

Thus, the students were divided into two major teams, A and B. Each team was then split into five groups of two students and each pair was given an iPad. In order to avoid interpersonal tension which could hinder the learning process, the students were asked to form groups with the people they sat next to, usually their classmates. Expectedly, this also resulted in a student-preferred seating plan in the classes to follow. Nevertheless, the purpose of the experiment was not to force team partners to work together; the goal was to stimulate active learning while maintaining a comfortable, easy-going setting.

In order to create a challenging activity, the teams were given a total of 10 minutes per coding exercise. The role of the instructor was to facilitate this process by providing subtle hints to guide the pairs. The first pair to finish would earn 10 points per solved exercise for its mother team (A or B). At the end of the two-week-long period, the team with the highest score would earn its 10 members, an additional 4% value on their overall semester grade. Winning pairs would receive an additional 1% (a total of 5%). Although a minor prize, this was expected to act as an adequate incentive for student engagement in the process.

**Technical Features**

The experiment was conducted using 3rd generation iPads running iOS 8 operating system. Unlike the previous study, the unsolved exercises were not already pre-loaded on the iPads, due to instructor workload for the increased number of devices. As mentioned, the exercise briefs were timely made available via the Facebook lesson page. Students used JS Bin, an IDE (Integrated Development Environment software) that fulfilled the following features:

- Simple code editing interface
- Simultaneous preview/output pane
• Ability for independent pane management (scroll, zoom, move)
• Support of HTML, JS and CSS code syntax
• Version control

Pilot Tests

The new study was conducted with a larger group than previously, as the course was made mandatory as opposed to an elective. Out of the two types of hands-on exercises previously used, ‘single (whiteboard-projected)’ and in pairs (Mavri et al., 2014), we maintained only the latter. Drawing on past results, we hypothesized that the risk of exposing possible personal weaknesses (e.g., insufficient knowledge levels, language/writing limitations) would be exacerbated in larger groups. Pilot tests confirmed this assumption; apart from the above limitations, further issues surfaced:

1. Room layout, student seating, and technical specifications: The oblong rectangular shape of the room (Figure 8.2) compromised visibility for the people at the back. As the amount of coding increased and became more complex, the active working student had to zoom out, for better code overview. Decreased font sizes on a relatively small whiteboard, compromised visibility for students positioned the furthest from it, who complained that it was difficult to follow and comprehend what was going on in the exercises.

2. Lack of positive social interdependence, which suggests that the “actions of individuals promote the achievement of joined goals” (Johnson & Johnson, 2009, p. 366) in a group.
In the previous study, students reported feeling a sense of unity and equality in learning together as a group. This was not evident in a larger group setting. Whether this was due to lack of responsibility and engagement from group members, or the convenient ‘hiding’ backseat spots for shy or weak students, active collaboration was not found to be an effective approach in this case.

**Analysis**

Post-study data were collected via an online survey and a focus group, both on week 4 of the course—following 1 week of pilot tests (week 2) and 2 weeks of active experimentation (weeks 3 and 4). The surveys consisted of both close and open-ended questions; the latter prompted participants to explain and elaborate on their preferences, thoughts, and feelings. The close-ended question data from the pre- and post-study surveys were evaluated in Excel while nVivo 10, a qualitative data analysis (QDA) computer software, was used for qualitative analysis. The open-ended responses were imported in nVivo in the form of datasets, by identifying the codable text fields. The focus group recordings were also transcribed in nVivo. An initial review produced a total of 131 and 52 codes for the survey and focus group respectively. Following several rounds of analysis, these were merged and reduced to 101 and 41 respectively and were both classified into seven major thematic categories.

*Figure 8.2. Laboratory plan view—physical layout compromised visibility for the back seats, in ‘single-projected’ exercises.*
A subsample (10%) of the data was analyzed and coded by two researchers independently and an inter-rater agreement value of 0.87 (based on Cohen's Kappa coefficient statistical measure) was recorded.

As expected, survey and focus group results were similar topic-wise. Outcomes related primarily to the overall experience and usability of the device combined with external keyboards, the game-based learning approach as well as general feelings and attitudes towards the process.

Table 8.1

Reference Occurrence per Thematic Category

<table>
<thead>
<tr>
<th>Thematic Categories</th>
<th>Description</th>
<th>Survey</th>
<th>Focus Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Exercise-related</td>
<td>Codes related to collaborative, competitive, or individualistic learning, practical exercises, time restriction, and bonus marks</td>
<td>31.6%</td>
<td>9%</td>
</tr>
<tr>
<td>2 External keyboard</td>
<td>Codes related to the external keyboard (+keys) in terms of size, layout, and combination of keys in comparison to the virtual keyboard</td>
<td>14.2%</td>
<td>21%</td>
</tr>
<tr>
<td>3 The iPad device</td>
<td>Codes related to the device both as hardware and software, usability and comparison/preference to other technologies (mobile/touch devices, laptop etc.)</td>
<td>13.8%</td>
<td>50%</td>
</tr>
<tr>
<td>4 Familiarity/learnability</td>
<td>Codes related to previous experience with the devices and speed of learnability</td>
<td>13.5%</td>
<td>13%</td>
</tr>
<tr>
<td>5 Active learning</td>
<td>Codes related to the learning-by-doing method, the perceived user experience, and learning outcomes</td>
<td>11.9%</td>
<td>-</td>
</tr>
<tr>
<td>6 Purpose of use</td>
<td>Codes related to the iPad used either to write code or for online research</td>
<td>8.4%</td>
<td>-</td>
</tr>
<tr>
<td>7 Emotional aspects</td>
<td>Codes related to feelings, thoughts, and attitudes</td>
<td>3.8%</td>
<td>7%</td>
</tr>
<tr>
<td>8 Other</td>
<td>Unrelated codes</td>
<td>2.7%</td>
<td>-</td>
</tr>
</tbody>
</table>

As the key variable in this study, we will first begin with results related to the use of external keyboards. As these are inevitably interrelated, we will then look at the iPad user experience in active learning. Game-based factors such as collaboration and competition are then examined.
External Keyboard

Quantitative and qualitative analysis of collected data indicates a general positive consensus towards the addition of external keyboards. The preference of physical over the virtual (native) keyboard was significant ($M=3.3$, $SD=1.15$, $Mode=4$)—based on close-ended responses, from a 1 to 5 scale with 1 = Totally Disagree and 5 = Totally Agree. It was generally found to enhance and facilitate the code development process during practical assignments.

As students had to manage their settings, connect to the wireless network, and access the Facebook lesson page—prior to attaching the keyboards—they also made use of the touch keyboard. As a result, comparison was inevitable; they remarked that the physical keyboard improved their activities: “writing code became so much easier after installing the keyboard, compared to that on the screen” and “it came handy” while confirming that typing was extremely “difficult” on touch devices. Two of the most recurring advantages of the external keyboard were: (a) the arrow keys and (b) the larger available screen area.

The four directional arrow keys, located at the bottom-right corner of the external keyboard (Figure 8.3) were found important for accessing specific lines or characters in the code. The majority of the students agreed that it ‘significantly helped’ to use the arrows to navigate through the code. Previous studies (Budiu & Nielsen, 2011; Mavri et al., 2014; Tillman et al., 2012) have also indicated that the inability to hit specific targets in text and perform functions such as select, copy, and paste, were major issues on touch devices. The fact that more on-screen space—typically occupied by the virtual keyboard—was now available was considered an essential advantage; it allowed for larger code and preview areas. Reportedly, developers will always opt for bigger screen sizes (Godse & Godse, 2008) as programming objects appear larger, detail is illuminated and mistakes are better prevented.

The overall positive attitude came, however, with a considerable set of drawbacks:

1. Small keyboard and key size
2. Layout and key combinations
3. Absence of keyboard shortcuts

**Small keyboard and key size.** The small keyboard, designed for optimum compatibility with the iPad mini, had a negative effect on the participants’ coding experience. Students argued that it interfered with the typing flow, due to many incorrect key substitutions. In fact, some
students made illustrative remarks during the focus group: “you go a bit like this (careful and slow typing)... like an old lady... in order to type”, “the keys were so small!” . The ‘tap-to-screen’ asymmetry issue (Budiu & Nielsen, 2011) widely found to hinder touch-typing, was also experienced on a physical level. A student commented: “I didn't know how to write fast... I am used to typing on my laptop where they keys are much bigger...”. As a result, two key issues are affirmed:

1. Constant comparison with a normal keyboard (i.e., laptop) was inevitable, as the two share similar physical characteristics.
2. Phrases such as “fast, slow, time-consuming, wasted time” place time and speed as crucial factors: coders need to timely externalize and evaluate the results of current cognitive activities. Likewise, responses suggest that it was clearly down to “a matter of time”, that students would decline using the iPad for coding, even with the add-on keyboards.

**Keyboard layout and key combinations.** Apart from size, the non-standard positions of keys for basic HTML, CSS, and Javascript required characters ( {} [] “” ’ ) lead to incorrect typing performances (Figure 8.3). These could only be accessed by holding down the Fn—instead of the Shift key—unlike traditional QWERTY keyboards. Moreover, the essential cmd (command) button (equivalent to Ctrl in Windows), was located exactly on the left of the space bar—like in a standard Mac keyboard—rather than three keys to the left of the space bar, like in a Windows-based or other larger external keyboards for tablet devices (Figure 8.4). This imposed a considerable memory load on top of the current mental processes: actively ‘thinking’ about the right key combination for accurate typing, took away from the focus on the logic. The keyboard layout posed obstacles for users in this case.

**Absence of keyboard shortcuts.** The absence of the desired copy-paste shortcut, was an additional factor that impeded perceived performance. This function ensures that identical segments of code are reused in multiple instances, a standard programming requisite. In reality, the keyboard did facilitate a copy-paste shortcut; students however, used the wrong key combinations: Ctrl+C and Ctrl+V instead of cmd+C and cmd+V. The faulty behavior resulted from the habitual use of a standard PC keyboard.

Thus, although preference was admittedly in favor of the physical versus the virtual keyboard, this issue helped create a negative bias, as it was still found to be “difficult” and “complicated”.
Figure 8.3. The Genius LuxePad i9010 bluetooth keyboard for iPad Mini size: 20 x 14.7 x 1cm

Figure 8.4. Genius LuxePad 9100 bluetooth keyboard for Windows, Android or iOS based devices – Size: 26.4 x 12.8 x 1.75cm
iPad: The Device

Point to click. Most negative responses related to the point-and-click limitation and in effect the inability to efficiently select, copy/cut, and paste, during coding on the iPad. Although the keyboard’s arrow keys helped, this issue hindered the user experience. Hitting text targets requires fine navigation and precision, clearly not facilitated on touch devices (Tillman et al., 2012) and, as inferred, not well supported through additional props. This was consequently a setback for select, copy and paste actions, used to avoid logic repetition and save time; time emerged once again as critical; participants could not perform at a satisfactory pace: “when you want to copy paste… by the time you select something… time is wasted…”. Coinciding with previous findings (Mavri et al., 2014), students resorted to re-typing entire code segments from scratch and consequently spent more time in doing so.

Multi-tasking. Multi-tasking issues emerged as prominent during this study, as students had to navigate through several windows in order to:

- Preview results in a browser
- Access initial exercises on Facebook
- Search for—or browse through—online resources

In specific, the need for simultaneous side-by-side window displays and direct manipulation was vital in the study in order to access multiple sources, compare, and reference information for coding (Budiu, 2015). Split-screen viewing is typical and well exploited on larger monitors; yet mobile devices are largely constrained to a single window and offer only serial access between windows instead (Stotts et al., 2003). This is further compromised as the number of windows increases. Apart from being time-consuming, this activity can overload working memory (i.e., needing to remember more items) while working on a task. In the study, students reported that they could not work efficiently: “we just needed to have multiple windows there and we couldn’t!”. Quantitative analysis of the responses in regards to web search and referencing presented a negative outcome ($M=2.3$, $SD=1$, $Mode=2$) based on a 1 to 5 scale—with 1 = Totally Disagree and 5 = Totally Agree.

Comparison. Result in response to device preference, both in a laboratory or an auditorium setting, revealed a strong tendency towards laptops as opposed to iPads. In addition to aforesaid usability issues, the aspect of familiarity and habitual use of laptops emerged as prominent. Participants mentioned that using their laptops was more intuitive, faster, and efficient: “working on the iPad was undoubtedly a time-consuming procedure compared to the laptop” and “the first thing to come to mind is
to turn the laptop on and get the job done”. Some participants, however, regarded their lack of expertise with tablets as accountable for the issues encountered during the study. Thus, sufficient time for practice and familiarization with the device is recommended prior to any similar future experiments.

Even though existing work argues that tablets can provide the “functionality and connectivity of a laptop, with the mobility of a smartphone” (Melhuish & Falloon, 2010, p. 5), outcomes from this subject-centric study indicate the opposite.

The suggested addition of a mouse, combined with an external keyboard, prompted the majority of participants to agree that it would defeat the point, regardless of the light-weight and mobility trade-offs. “Why would you bother bringing the sum of all this stuff with you then?” They added that the release of new generation 12 and 10-inch laptops provides such trade-offs. In addition, observed connectivity problems with these ethernet-less devices, are too big to ignore, when Wi-Fi infrastructures fail, under demanding usage conditions.

Positive responses in favor of the iPad compared to a laptop, presented arguments in regards to the absence of cables, longer battery life, and better portability in auditorium-like environments. Students also showed excitement and anticipation to work with the iPads, prior to the experiment as they thought it would be “cool to work with”. They also believed they were familiar with the device, based on previous experience with the Apple equipment provided in the department’s labs.

Collaboration. Quantitative results from the study indicate that class-based collaboration improved perceived comprehension and learning, versus individual work ($M=3.2$, $SD=1.4$, $Mode=3$) based on scaled responses, on a 1 to 5 scale—with 1 = Totally Disagree and 5 = Totally Agree. Qualitative analysis indicated an even stronger preference with twice the amount of positive (than negative) responses, largely due to perceived higher achievement level, constructive social interactions, and better time-management.

Positive interdependence: substitutability and promotive interaction. Collaboration appeared as an effective factor in students’ perceived learning process, even within a larger group. Participants readily admitted that although they could “probably solve the problems on their own, it was much better to do so collectively”. Most of the responses attributed this to substitutability and promotive interactivity, as demonstrated by the social interdependence model (Johnson & Johnson, 2009). In other words, group members recognized each other’s limitations and contributed their own knowledge and skills to help reach a common goal (Villanueva, 2003).
Comments such as “…we filled in for each other’s gaps” and “two brains were better than one” are indicative of concrete positive interdependence in the pairs. A student also remarked that learning outcomes could be further reinforced with individual follow-up training, through reflection of the practical experience in class.

Good interpersonal communication, an elevated sense of trust, and overall team bonding was also observed in paired teams, who encouraged and mentally supported each other (promotive interaction) through dialogue: “It was easier when we could talk about it in class”, “helping one another was better”, “yes, its better to work together and talk about it…”. Evidently, having the chance to go about solving a task through discussion enhanced the understanding of the required components and facilitated the solution process more efficiently rather than doing so alone.

**Time management: procrastination.** The perceived advantages of time regulation and management through collaboration in a friendly competitive setting were prominent in this study as two key issues have emerged: (a) procrastination and (b) prolonged task completion times (analysed in the ‘Competition’ section).

Actively collaborating to practice programming tasks in class was found to help alleviate procrastination, which as explained, can pose serious problems in academic progress (Edwards et al., 2015; Hen & Goroshit, 2014).

Participants deemed the co-located collaboration and timed practice as an important feature of the active learning method. The fact that students were “forced to do so… in the good sense…” collaborating in the allocated time period, rather than leaving the task for homework, was perceived as an advantage: “it was faster to solve this collectively” and “we found out what our mistakes were in 10 minutes…”.

Procrastination is particularly evident in such challenging modules as programming (Edwards & Snyder, 2009), since concepts and principles are complex to comprehend and apply. In addition, buggy coding and other technical issues may present a deadlock for students who often put off working on assignments, until very close to the submission date. Task averseness and subsequent delays are factors that contribute to stress, confusion, poor work quality, and in effect, bad learning outcomes (Hen & Goroshit, 2014). Understandably, students appreciated the benefits of being obliged to practice their assignments straightaway. Additionally, the instructor noted that the ‘no-submit’ incident numbers decreased from past years.

**Competition.** A lot of controversy is associated with competitions in the classroom; as a result, its impact on learning is still not clear
Outcomes from this study, suggest a positive incline towards a friendly-competitive active learning setting. Game-based learning between groups was considered beneficial ($M=3.2$, $SD=0.6$, $Mode=3$)—based on the close-ended responses from a $1$ to $5$ scale with $1 = \text{Very Negative Impact}$ and $5 = \text{Very Positive Impact}$. Qualitative analysis indicated that positive responses were interestingly equal to those of a neutral character. The majority of the students were either happy or indifferent about the friendly competition setup, as opposed to those who objected to it. Those who favored the element of competition did so for reasons relating to increased motivation, improved quality and quantity of work, better time management, and enjoyment.

Competition amongst pairs and teams, reportedly encouraged more intensive effort on behalf of the students to produce more and better results: “Competition makes you work better”, “You try for the best” and “…its better because we work more…” are a few of the comments that led to this assertion. Perseverance from all pairs in support of each other as well as the ‘mother’ team (A or B) was evident, as observed by the instructor, during the practical sessions, regardless of individual knowledge and capabilities.

**Time management: extended completion times.** All pairs tried to finish first or to complete the exercises within the deadline. The competitive, time-constrained exercises were also perceived to enhance time management by avoiding slacking; instead, they evidently urged for more determined attempts to solve the problem at hand, within the given time limit. A lot of interesting comments were collected: “we wanted to finish first, we were working fast and did not let the exercise take half an hour”, “we were pressurized by time… in the good sense” and “…we did not leave it down to luck…” This reveals students’ awareness of their own limitation to often control their times in order to expedite assignment completion. This is slightly different to procrastination; it means increased task completion times, not task postponement.

Existing literature refers to human effort that expands in order to fill the allotted time for a given task, even if not actually needed; a phenomenon dubbed Parkinson’s Law (Latham & Locke, 1975). Likewise, outcomes from this study suggest that students are aware of taking longer to complete tasks, if excessive time is provided (i.e., in the case of weekly deliverables). All the same, time and pace of work is readjusted to fit a shorter, compulsory time frame. Increased levels of satisfaction and better student engagement were reported by students—provided that realistic time limits were, of course, enforced. Finally, positive comments referred
to the time-constrained competition as stimulating, challenging, novel, game-like, and fun.

**The no ‘punishment’ approach.** A large sum of neutral responses emerged alongside the positive ones. Typical remarks such as “there was simply no impact” were collected and some students explained it was because it would not, after all, affect their grades. The competitive method employed was carefully planned to present a prize for winners and no punishment for losers (Burguillo, 2010): “If I don’t finish it on time, I have the chance to see it finished on the board” and “If you are capable of doing it you will solve it anyway”.

According to existing literature, there is no “significant difference between working to achieve a reward and working to avoid a loss” (Johnson & Johnson, 2009, p. 367). Both attempt to gain a positive outcome, but not avert punishment, which is essentially different to the loss of a reward. This indicates that in both cases, students put effort in the competitive tasks, not because they felt threatened, in case they failed to ‘win’. Interestingly a few participants commented on the fact that they “were not an antagonistic type of group”, implying the bad, undesired aspects of competition. As inferred, the students did not suffer any harsh competitive effects.

**The ‘shielded’ approach.** The researchers do not attribute this only to the insignificant marking value but also to the multi-layered group structure. The choice of the two larger groups (A and B) was intended to ‘shield’ the pairs and individual players, who were in direct competition with other internal teams (with a minor incentive of 1%) and in secondary competition with groups from the opponent team (with an increased incentive of 4%); this negligible marking value increased proportionally to the competitive distance. In this way, negative, face-to-face tension was minimized for the actual players and allowed them to focus on the learning process rather than the prize to be won (Shindler, 2010). The concept of a mild, friendly competition is indeed proven to enhance learning by providing students with an adequate amount of intrinsic curiosity and motivation to engage in the process (Kumar, 2000) minus any serious consequences.

About one fourth of the collected responses opposed competitions and time-restricted exercises that referred to these as a ‘pressure and stress-inducing’ factor. They explained that there was not enough time to learn about the principles, reflect on them, and solve the exercise. In addition, a participant remarked that “they should not be playing games” while learning. Expectedly, the same participants also stated that they preferred to study and practice on their own, rather than collectively in class.
Discussion

Active, collaborative learning has increasingly been used as an instructional method over the past few decades. It is argued that it offers an engaging and motivating process, it reinforces learning objectives through enhanced reflection and assimilation, while establishing better relationships and enabling students to develop a higher self-esteem of themselves within a social setting (Fulu, 2007; Shindler, 2010). This study aimed to examine previous findings on such active learning methods, through the use of iPads and physical keyboards, within a larger group setting.

The decision to add mobile devices to the active, game-based learning approach aimed to enhance and ‘spice up’ the student experience, using the appeal of mobile technologies (Mehdipour & Zerehkafi, 2013), while promoting portability and easy handling. Even though, earlier years saw the shift from physical—towards—soft keyboards, which appeared only whenever needed and allowed for infinite layout customizations (i.e. on PDAs, digital cameras), recent developments indicate a back-to-the-basics approach. Apart from issues concerned with obscured screen space and lack of tactile feedback, working with a touch interface requires learning and memorizing a series of gestural movements for navigation and data input (Hoggan, Brewster, & Johnston, 2008; Norman & Nielsen, 2010); this can have a serious impact on the cognitive load of novice learners, especially during code development in the class.

Pairing the iPad with a physical keyboard was indeed found to resolve previously encountered problems, mainly associated with text entry. Student response was obviously influenced by a strong comparison between physical and touch-based typing as they were reminded of the latter, in performing activities prior to pairing the keyboards. Coinciding with the researchers’ expectations, the keyboards were not only found to enhance typing performance but partially resolve direct manipulation (i.e., pointing) issues too, through the use of arrow keys. The physical keyboards were nevertheless received with a set of problems. In evaluating these, we infer that they emerged due to a single denominator: size. Size was the cause of incorrect key substitutions, increased mental overload, slow typing, and key omissions. Other than the evident key-to-finger asymmetry, standard keys were also re-arranged due to space limitations on the mini keyboard; hence, the users’ repetitive ‘hit and miss’ attempts for various key combinations. It is a fact that the smaller the mobile devices get, the smaller their peripherals become. Despite their intention to
overcome mobile and touch-derived issues, these are however inherited into the physical context. Size needs to be evaluated and balanced against efficiency of use. Proper user-evaluation should be performed prior to causing confusion for unsuspected users, especially where education is concerned (Norman & Nielsen, 2010).

In this study, the external keyboard was chosen based on its compatibility with both versions of available faculty equipment—3rd generation iPads and iPad minis—as originally experiments were planned to run with the latter too. In fact, the keyboards’ lightweight attributes were originally perceived as an advantage compared to other, cumbersome options in the market. Based on the results of this study, future work could investigate the outcomes of using alternatives to facilitate active learning for collaborative web development. Such alternatives could be for example, Apple’s Smart Keyboard for the iPad, which was recently introduced.

Other than the problematic text-entry activity, this study suggests impaired user performance due to the lack of concurrent, versus serial, multi-tasking on the iPad. It is of no surprise that split-view multitasking is now introduced into iOS 9 (made only available on the iPad Air 2, iPad Pro and iPad mini 4 for the time being). Together with constant enrichment of technical capabilities, this is an example of the conversion leap from consuming to more powerful, production-capable devices. These are certainly promising and worth testing, in forthcoming studies which involve active learning—especially in web development—which seemingly revolves around a mobile-first approach (Mehdipour & Zerehkafi, 2013). Educators need nevertheless be careful when incorporating them into the learning process. Indubitably, an evident outcome from this study suggests that students should be thoroughly familiar with the devices before using them as part of the lesson. Lack of familiarity, combined with the aforementioned usability issues, can impose unnecessary load, lead to a disappointing user experience, and as a result, hinder the general learning performance of novice learners.

**Conclusion**

There have been quite a few studies on the impact of technology-enhanced learning using mobile devices. A few of these are subject-centric; even fewer are dedicated to learning programming, and there are almost none concerned with web design and development. Based on previous results, this study examined the user experience and perceived learning outcomes, through the use of mobile devices paired with external
keyboards, as an in-class active learning platform. Through this, two new parameters were examined: layered collaborative-competitive setup, within a larger group setting. The results indicate a favorable response towards the addition of a physical keyboard, as it is perceived to enhance the coding performance, versus the native touch keyboard. The external keyboard was found to (a) enable faster typing, (b) allow for larger on-screen area for coding and preview, and c) help locate specific on-screen information with the arrow keys. The small key size, however, was responsible for incorrect key substitutions and omissions. The non-standardized keyboard layout generated additional cognitive load and confusion for users. In regards to active learning, pair collaboration was favored, as it was perceived to improve learning outcomes. In addition, students deemed the friendly ‘layered’ competitive setting as beneficial due to better time management and instant feedback. The use of iPads was found to distract rather than support the code development process, mainly due to the lack of split-view multi-tasking, as well as missing direct manipulation (i.e., point-and-select) capabilities. With the advent of new generation, smaller, and lighter laptops, with touch screen capabilities and improved battery life, student preference appears to lean towards these, versus the iPad, ignoring the device’s obvious portability and convenience qualities. However, a series of upgrades and additions to recent iPad models, as well as new line of peripherals, appear to rectify such issues and this invites future work in this area.

This study is limited in terms of exclusive evaluation of the perceived user experience and performance. There were no quantitative measures of actual time records, error incidence, and final marks. Using older generation iPads paired with keyboards designed for the iPad mini was also a constraint that needs to be taken into account in future studies. Finally, the lack of appropriate iOS web development editors (code + preview), providing utilities such as auto-completion and auto-correction, compared to richer development applications such as Adobe Dreamweaver, may have influenced the users’ experience and attitude against mobile devices for code development. Lastly, this study did not investigate the contribution of the iPad per se towards active learning, and the latter was solely used as a context for this investigation.
References


