Applying user interface guidelines to the development of educational software for equation solving

Daphne Robson
Christchurch Polytechnic Institute of Technology, New Zealand
daphne.robson@cpit.ac.nz

Walt Abell
Lincoln University, New Zealand
walter.abell@lincoln.ac.nz

Therese Boustead
University of Canterbury, New Zealand
therese.boustead@canterbury.ac.nz


Abstract

Creating educational software requires a thorough understanding of several key areas: pedagogy, software development and user interface design. This study, which is part of a larger investigation into the impact on learning of educational software for learning equation solving, focuses on user interface design and its relationship to pedagogical principles. User interface features considered include: nature of feedback, screen layout, metaphors, instructions, buttons, and score. This paper is based on trials with real users of educational software at Christchurch Polytechnic Institute of Technology and provides practical information for computing students designing user interfaces.

Keywords

User interface, interface design, software design, educational software, equation solving

1. Introduction

Creating educational software requires a thorough understanding of several key areas: pedagogy, software development and user interface design. This study, which is part of a larger investigation into the impact on learning of educational software for equation solving, focuses on user interface design and its relationship to pedagogical principles.

User interface design is an integral part of study for most tertiary computing students in New Zealand. For example, at Christchurch Polytechnic Institute of Technology (CPIT), the Co-operative Education Project, BCCE301, requires computer students to
work with clients or colleagues and apply learning from completed courses to design, implement and evaluate a piece of work (CPIT, 2013). Nesbit (2010) found that 51% of these projects involved development of software including web and mobile applications, and a further 20% of projects involved website evaluation. Thus, at least 71% of projects required computer students to have a good knowledge and understanding of user interface design.

The educational software in this study was designed to help polytechnic students learn to solve equations, a skill needed for their planned engineering or science study. Trials were carried out to determine the attitudes of users (i.e. mathematics students) to specific features of the interface in terms of the purpose of the software which is to promote the learning of equation solving. This paper includes an overview of interface design guidelines and pedagogical principles, a description of the user interface in this study, results of the trials, and a summary of the findings.

Note that in this study, the term "computer students" is used for students studying computing to distinguish them from the students who participated in the study and were studying mathematics. The terms "user interface" and "interface" are used interchangeably.

2. Background

When designing a user interface, the needs of the user and the purpose of the interface should be considered (Shneiderman & Plaisant, 2010) and the interface should empower the user to achieve the goal of the software (Nielsen, 2005). In this study, the goal of the software is to help students learn. Therefore the design of the user interface should follow both user interface design principles and sound pedagogical principles.

Eight "golden rules" of interface design are described by Shneiderman (1987). Interfaces should be consistent, suit different types of users, provide informative feedback, indicate completion of actions, prevent user errors, permit actions to be reversed, minimise cognitive load and allow the user to feel in control. According to Norman (1995), enough information about the current state of the system must be provided with possible actions clearly visible so that users can predict the effect of their actions.

In addition to these important and well-established principles for interface design, the nature of visual elements needs to be considered. Watzman (2003) recommends that the overall layout should be balanced and elements should work well together with related elements in close proximity. Typefaces should be easy to read and text should be placed so that it is easy to find. Graphics should be simple, consistent and appropriate to the content. Colours should be chosen taking into account their purpose and effect. Nielsen (1993) recommends that the visibility of objects should relate to the needs of the user.

Metaphors can be used to provide structure to the interface design (Nielsen, 2000). For example, a "home page", with its icon of a house is the main starting point for a website. The metaphor suggests that the user is being welcomed, and that it is a safe place to return to if the user becomes confused by the website. An effective metaphor also allows users to apply knowledge of the metaphor to the activity. It must be easily understood but free of meanings which could mislead the user (Erickson, 1995).

Nielsen (1993) recommends that cognitive load placed on a user by an interface should be minimised. To help with this, only a few rules should be needed to use the interface. Also, as every element on the screen adds to the cognitive load, fewer elements make an interface easier to use. Nielsen suggests putting only the most important information on the main screen with additional information on other screens. For example, he suggests dividing messages into two levels so that a short message is displayed and a longer message available when requested.

Feedback can be presented in different ways but it shouldn’t obscure a user’s actions (Cooper, Reimann & Cronin, 2007). It should be obvious what clicking a button will do, for example, by labelling it with a verb. Messages about errors should be polite
and provide information that allows users to plan their next actions.

In education, feedback is recognised as an important pedagogical principle and aid to learning. It is generally agreed that feedback improves learning if it encourages students to think actively and to take action (e.g. Hattie, 2009, p173-4).

The potential for technology to enhance student engagement with feedback has long been seen and there are a growing number of studies that support this hypothesis (Hepplestone, Holden, Irwin, Parkin & Thorpe, 2011). Technology can be used to design feedback with features that have been shown to improve learning such as: different feedback according to the response of the learner (Narciss & Huth, 2002), feedback that is not available until a student has given a response (Mory, 1996), and the opportunity to try again after receiving feedback about an error (Dihoff, Brosvic, Epstein & Cook, 2005).

Studies which investigated specific features of feedback such as amount, timing and frequency are inconsistent (Mory, 1996; Mason & Bruning, 2001). For example, Mory (1996) observed that some studies found more information helped learners develop their understanding whereas others found this increased the cognitive load or distracted the learner. To balance these conflicting findings, the option for students to request additional feedback is suggested, although once again results are inconsistent (Mason & Bruning, 2001).

In some cases, conflicting recommendations are suited to different situations and conditions (Mory, 2004). For example, immediate feedback is usually better when it is designed to help students analyse their errors and determine an improved course of action. On the other hand, delayed feedback is suited to memorisation tasks (Narciss, 2008).

Thus, recommendations from studies of educational feedback delivered by technology are varied and sometimes conflicting. As the studies were done in a wide variety of contexts, it is relevant to consider a study of feedback for learning equation solving. Nguyen-Xuan, Nicaud and Gelis (1997) tested different types of feedback provided by software and found that it should be short, include consequences of errors, give enough information for students to see why their response was incorrect, but allow them to work out the next step themselves.

Software must also be easy to use as user interfaces that are difficult to use can interfere with the instructional value of the software (Frye & Soloway, 1987). Poor interface design can lead to students taking longer and being less likely to complete lessons (Szabo, 2000) and can affect student motivation as students lose interest when they became confused (Vonderwell & Zachariah, 2005). On the other hand, student motivation can be improved by applying principles of game design software to a user interface, for example, requiring students to make meaningful decisions, and providing clear goals and frequent rewards (Prensky, 2003).

In summary, there are many interface design guidelines and pedagogical principles that need to be taken into account and a software designer must often decide which of two considerations is more important (Nielsen, 1993).

3. Design of the Interface

This section begins with an overview of the design of the user interface of prototype software in this study, Equations2go (Robson, 2004) and is followed by a description of specific features of the user interface. Decisions about the interface design were based on a combination of the interface design guidelines and pedagogical principles described above and the results of usability testing with five students.

3.1 Overview of Interface Design

Equations2go is designed to help students learn to solve simple linear equations. It allows students to choose what to do at each step and the software then carries that out. This principle allows students to focus on the decisions they make at each step and how these decisions combine to form a strategy for solving an equation. To
encourage deep learning, several different strategies for solving the equations are accepted by the software.

Students click the mouse on "hot spots" on the equation and choose options from visual menus that appear. Guidance is provided to students with several types of feedback, and a visual record of student actions is provided by "trails". In Figure 1, a partially solved equation in Equations2go is shown.

![Figure 1. Partially solved equation in Equations2go](image)

Figure 1 shows the result of the first step in which 4 was added to both sides of the equation. The mouse is shown hovering over a hot spot causing a menu of operations to be displayed for the second step.

### 3.2 Metaphors

In Equations2go, metaphors are used to help students see equation solving strategies as a sequence of steps and that different strategies can be used to solve an equation.

The step by step nature of equation solving is represented by a stepping stones metaphor of "one step at a time". The equation to be solved appears on the first stone, and as each step is completed a new stone appears with the simplified equation ready for the next step. In this way, a stone visually identifies each step and allows students to consider each step in turn.

Another interpretation of the stone metaphor is "leave no stone unturned" which reflects the students' search for different strategies. The concept of different strategies leading to the same solution is supported by the metaphor of tracks in a forest in which several routes lead to the same place and signposts provide guidance. In Equations2go, the tree image suggests a forest, trails record the route taken, and feedback flags act as signposts.

Another metaphor is the use of opposite directions to represent inverse operations such as addition and subtraction.

### 3.3 Feedback

The amount of feedback needs to be a balance between short feedback recommended for equation solving and sufficient information for guiding students in their next decision. In Equations2go, the balance was achieved by displaying a quick "tip" inside a flag with a more informative explanation being available at the request of the student. This conforms to the recommendation that screen design should be kept as simple as possible to minimise cognitive load but that more information should be available on request.
The flag, which only appears when student actions are not accepted, contains a short phrase about the type of student action needed. The explanations, which are available for all student actions consist of an equation and one or two sentences of information. They provide students with more information than the flag, but do not require students to read large amounts of text. In this way, short feedback is always provided by the flag, whereas the student controls the display of the more informative explanations. In addition, the equation shows consequences of errors. An example showing these types of feedback is shown in Figure 2 in which the action chosen was to divide both sides by 2 and this was not accepted.

![Figure 2. Feedback for a step not accepted by Equations2go](image)

Visual changes provide additional feedback to verify each correct step. A new stone appears ready for the next step, and a trail appears between the two stones recording the student action. The scoring panel on the tree also changes colour whenever the score increases and a quick animation (a spinning star) signals when the equation is solved.

The purpose of the score is to help motivate students by providing short term goals, and to provide feedback about students’ progress through an equation. A button for turning the score on or off was added as one of the five usability testers was unwilling to explore the software because she didn’t want a low score. An undo button was also added after receiving feedback from usability testers.

In these ways, the visual stepping stones metaphor is supported by colour and visibility changes of stones, menus, trails, scoring panel and the spinning star. See Figure 3.

![Figure 3. Visual feedback for an accepted strategy](image)
3.4 Hot Spots andMenus

To navigate through the steps, students use hot spots located on equation elements and menus which appear around the stone for the current step. Students choose what to do at each step by clicking on a hot spot and then selecting from the menu that becomes visible. Menu previews, which appear when the mouse hovers over a hot spot, were added because usability testers wanted to know what options would be available before they clicked a hot spot. See Figure 4. This was a reminder of the importance of users being able to predict the effect of their actions.

![Figure 4. Hot spots and menus](image)

3.5 Screen Layout

The screen layout was designed to be as simple as possible to minimise the cognitive load associated with learning to use the software. Graphics and typefaces are simple and used consistently. To keep the number of elements on the screen to a minimum, each element is included for a specific reason related to the learning activity. For example, the stones are related to the stone metaphor used to help students visualise the step by step nature of equation solving. The reason for the Show/Hide Explanation button is to allow students to navigate to more informative feedback. It also helps simplify the screen layout as the explanation is not visible until a student requests it.

4. Trials

Trials were conducted with students and although the main focus was to investigate the impact of the software on learning, aspects of the user interface were also explored. Trials were conducted with eight classes of students preparing to study engineering or science at CPIT by studying Algebra at NZQA Level 1 or Level 2. There were 75 students who took part in the trials but the data for 13 students were discarded as invalid because of being incomplete or having taken part in the trials twice. Of the 62 valid participants, 29 were male and 33 were female. Half were aged between 20 and 29, approximately quarter were under 20, and approximately quarter were 30 or over.

Trials took place during class sessions and with ethics approval. After doing a written pre-test on solving equations, students used the software for 20 minutes and during this time student actions were logged electronically by the software. Students then completed a post-test and a post-questionnaire. The post-questionnaire asked students about how easy or hard it was to use the software. It also listed the following features of the user interface: quick tips, explanations, undo button, instructions, and score. For each of these features, students were asked to indicate whether they found them helpful or unhelpful or whether they didn't use that feature. They were then asked to write comments about these features as well as to make general comments. See Appendix.

Finally, students worked in pairs or small groups to discuss and record what they liked about the software and what they found annoying. There were 29 discussion groups and these included some of the students whose data were invalid, but the comments of these students were included because it was not possible to identify and exclude their contribution to the discussion groups.

The data and comments from the post-questionnaires and discussion groups were
examined and categorised. The logged data were summarised and the number of times students used each button were calculated.

5. Results and Discussion

The results include quantitative data about whether specific features of the interface helped students learn and qualitative data in the form of comments. Comments included reasons for their attitudes to specific features of the interface and suggestions for improving Equations2go. The following are considered: ease of use, feedback, graphics, buttons, and score.

5.1 Ease of Use

Ease of use is important because learning can be affected if students find an interface difficult to use (Frye & Soloway, 1987). Most students found Equations2go easy to use with 92% indicating that it was very easy, easy or OK to use. In the group discussions, ease of use was mentioned by 7 groups. These results suggest a successful application of the principle that being able to use software should require minimal cognitive load (Nielsen, 1993; Shneiderman, 1987).

Five students found Equations2go hard to use and they would be expected to show less evidence of learning than the others. However, this was not case. Students who found Equations2go hard to use showed an average increase between their pre-test and post-test performance of 12% compared to 8% for the students who found Equations2go easy to use. Three students showed little change whereas two showed a substantial increase. This suggests that the difficulty encountered in using the software may have interfered with learning for three students but that the other two students overcame this barrier. Thus difficulty with using an interface affected learning for some students as found by Frye and Soloway (1987).

5.2 Software Features

The main pedagogical principle of Equations2go is the emphasis on strategies, and this principle is supported by the design of several features of the interface. In the post-questionnaire, students were asked whether specific features of the interface helped them learn. The responses are summarised in Table 1 and referred to in the following sections.

<table>
<thead>
<tr>
<th></th>
<th>Percentage of responses to questions in post-questionnaire</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Helpful</td>
</tr>
<tr>
<td>Quick tips</td>
<td>81%</td>
</tr>
<tr>
<td>Explanations</td>
<td>84%</td>
</tr>
<tr>
<td>Undo button</td>
<td>58%</td>
</tr>
<tr>
<td>Instructions</td>
<td>73%</td>
</tr>
<tr>
<td>Score</td>
<td>79%</td>
</tr>
</tbody>
</table>

5.3 Quick Tip and Explanations

Most students found both the quick tip in the flag and the more detailed feedback in the explanations helped them learn. Logged data for the use of the explanations were recorded for 57 of the 62 participants and showed that students made good use of the explanations. Students requested them an average of 57 times per student during the 20 minutes of the trial. This is an average of three explanations per minute. These results suggest that following the recommendations for the design of feedback of Nguyen-Xuan, Nicaud and Gels (1997), Nielsen (1993) and Mason and Bruning (2001) helped students learn.
In the last exercise of the trials, students were asked to explore strategies they thought would not work and to look at consequences of their decisions, as shown by the equations that are included with the explanations. This exercise probably caused students to view more explanations than they would otherwise and therefore probably contributed to the reported helpfulness of the explanations. There is potential for this exercise to become an integral part of the design of a future version of Equations2go.

Comments about the explanations were made by nine discussion groups and two individual students (one of whom was in one of the nine discussion groups). Students particularly liked the explanations and several reasons were given:

"It told you when you got it wrong and showed you what your answer would be if you continued on that track."

"Explanations clear and helpful."

"Language was simple to understand."

To keep the main screen as simple as possible (Nielsen, 1993), the explanations were only displayed when requested by a student so that the more informative feedback was only provided when students needed it (Mason & Bruning, 2001). However, students used the explanations very frequently, so in future it may be worth displaying the explanations every time a student makes a step rather than only when requested. This would reduce the student contribution to the interactivity so further investigation is needed to find out if students use explanations when they are displayed without being requested.

5.4 Graphics and Metaphors

Only one student commented on the graphics and it is a limitation of this study that students weren't asked any specific questions about the graphics and the metaphors. However, during the group discussions students were given the opportunity to comment on anything they liked or found annoying about the interface. The only student to comment on the graphics disliked the colour scheme but saw an extra metaphor in the colour change of the tree that was designed to provide positive feedback for a correct decision:

"The tree that glows like a light bulb - genius."

As no other comments were made about the graphics or the metaphors, they probably met the recommendation that they should not be misleading (Erickson, 1995). However, there is no direct evidence that the stone metaphor, the forest metaphor or the opposite directions that represent inverse operations made any contribution to learning. Students may not notice visual metaphors while they are concentrating on solving equations.

5.5 Undo Button

In the post-questionnaire, just over half the students indicated the undo button was helpful whereas 38% indicated they did not use it. Thus, the undo button was helpful to most of the students who used it. The logged data for this button, which was available for 57 of the 62 participants, are consistent with the post-questionnaire data as they show that 35% of students did not use the undo button. The other 65% of students who did use the undo button used it an average of four times.

One discussion group and one individual student in this discussion group liked the undo button:

"... if I made a mistake I could always undo it, keeping the equation neat, whereas on paper you have to scribble on it."

The only other comment about the undo button was made by a group which included the only student who indicated that it was unhelpful in the post-questionnaire:
"Didn't like the undo button because couldn't understand."

5.6 Other Buttons

Although most students found it easy to use Equations2go, three individual students commented that hot spots were not where they expected to find them or that the associated buttons did not do what they expected, for example:

"It confused me knowing which button to push."

It is important that students are able to predict the effect of their actions (Norman, 1995), and although only a few students had difficulty, this problem may increase if Equations2go is extended in future to accept more strategies. The comments from these three students highlight the importance of the principle as students are unlikely to learn if they are confused about the effects of buttons.

5.7 Location of Buttons

When using Equations2go, students' main focus should be the equation on the current stone and this is always positioned near the centre of the screen initially. The undo button and the Show/Hide Explanation button were originally placed near the edge of the screen so that their location was consistent (Shneiderman, 1987). During the software development process, usability testing showed that some students did not notice the Show/Hide Explanation button. The location of this button now varies so that it is close to the current stone, where it follows the recommendation that related elements should be close to each other (Watzman, 2003). This is an example of needing to prioritise conflicting principles as described by Nielsen (1993).

5.8 Score

The score was initially switched off in the trial, and in the second exercise students were asked to switch it on. Many students (79%) liked the score switched on, so there is potential to develop the scoring system further to provide more goals and rewards as recommended by Prensky (2003). Such goals could motivate students to use Equations2go in particular ways in order to achieve high scores: for example, to solve an equation without making errors, to find all available solution strategies, or to explore feedback.

As other students (18%) preferred the score switched off, and early usability testing showed that the score could inhibit student exploration, a balance needs to be found between encouraging students to explore Equations2go in specific ways with a score displayed and encouraging students to explore without the consequences of viewing a poor score.

5.9 Summary of Results

The results of following user interface guidelines when designing Equations2go are summarised in Table 2.

Table 2. Summary of results
### 6. Conclusion

Equations2go is educational software and therefore its design is based on pedagogical principles as well as user interface guidelines. The user interface guidelines (UIGs) supported the pedagogical principles (PPs) and this occurred in the following ways.

In some cases, UIGs supported PPs by being similar. For example, the UIG to put the most important information on the main screen with additional information on other screens (Nielsen, 1993) is similar to the PP of providing additional feedback on request (Mason & Bruning, 2001). Most students found the quick tips helpful with further explanations available on request.

In other cases, UIGs supported PPs by being consistent. For example, the UIG that users should be able to predict the effect of clicking a button (Norman, 1995) was consistent with the PP that students should be prompted to think of the next action themselves (Nguyen-Xuan, Nicaud, & Gels, 1997). The consistency is that being able to predict the effect of buttons provided prompts for the next action. In Equations2go adding menu previews, as a result of user testing, meant that these two consistent guidelines were followed.

Another way that UIGs supported PPs was by facilitating them. For example, the PP that students should be guided to think of the next step themselves was facilitated when two UIGs were prioritised as recommended by Nielsen (1993). Students found it easier to access the explanations that enabled them to think of the next step once the Explanation button was moved from a consistent location (Shneiderman, 1987) to a location closer to the current step (Watzman, 2003). Another example is the score which many students found helpful. The helpfulness of the score is consistent with Prensky’s (2003) assertion that the UIG of engaging users in software games by providing short term goals with a score could facilitate the PP of motivating students to take part in learning activities.

A further contribution of UIGs was to make the software easy to use. The well-established guideline to minimise cognitive load (Shneiderman, 1987; Nielsen, 1993) was followed and most students found the software easy to use. Thus for most students, the cognitive load associated with using the software was kept to a minimum so that they could apply most of their cognitive skills to their learning.

A limitation of this study is that it is part of a larger investigation into the impact on learning of the design of Equations2go. A consequence of this is that students were asked specific questions about only a limited number of features of the interface. In particular, participants should have been asked specific questions about the graphics and metaphors and this needs further investigation. There is also a need to
investigate whether the explanations should be visible at all times rather than on request as the explanations were frequently requested by students.

It was important that user interface guidelines were applied to the design of Equations2go as this led to software that most students found easy to use, allowing them to concentrate on their learning. User interface guidelines also supported students' learning by being similar to, by being consistent with, or by facilitating pedagogical principles. These examples, results and conclusions will be useful to computer students learning to design user interfaces as well as to their tutors.

**Acknowledgement**

Thanks to Dr Keith Unsworth from Lincoln University for his guidance and suggestions during this study.

**References**


Appendix

Post-questionnaire

1. How easy was it to learn to use the computer program?
   - Very hard
   - Hard
   - OK
   - Easy
   - Very easy

2. Did you find the following features helpful?
   - Flag with a quick tip
   - Explanation at bottom of screen
   - Undo button
   - Instructions
   - Score

3. Did you prefer to have the score?
   - On
   - Off

4. Please write any other comments you have about the computer program. Eg, why did you like or dislike any of the above features?

5. Do you have any other general comments?

Copyright © 2013 Robson, D., Abell, W. & Boustead, T.

Journal of Applied Computing and Information Technology (JACIT): ISSN 2230-4398

(I incorporarting the Bulletin of Applied Computing and Information Technology, NACCQ: ISSN 1176-4120 and Journal of Applied Computing and Information Technology, NACCQ: ISSN 1174-0175)

Copyright ©2013 CITRENZ.

The author(s) assign to CITRENZ and educational non-profit institutions a non-exclusive licence to use this document for personal use and in courses of instruction provided that the article is used in full and this copyright statement is reproduced.

The author(s) also grant a non-exclusive licence to CITRENZ to publish this document in full on the World Wide Web (prime sites and mirrors) and in printed...
form within the Journal of Applied Computing and Information Technology. Authors retain their individual intellectual property rights.
Michael Verhaart, Donald Joyce and Nick Wallingford (Eds.).

An Open Access Journal, DOAJ #22304398, ( ✓zotero)