

Identifying Cross-Platform and Cross-Modality Interaction Problems in e-Learning Environments

André Constantino da Silva
Institute of Computing (PG)
UNICAMP, IFSP

Campinas, Brazil, Hortolândia, Brazil
acsilva@ic.unicamp.br

Fernanda Maria Pereira Freire
NIED
UNICAMP

Campinas, Brazil
ffreire@unicamp.br

Heloísa Vieira da Rocha
Institute of Computing, NIED
UNICAMP

Campinas, Brazil
heloisa@ic.unicamp.br

Abstract—Web applications and sites are designed to use keyboard and mouse as input devices and a medium resolution screen as output device. Mobile devices, such as smartphones and tablets, have enough computation power to render Web pages, allowing browsing the Internet. But, their main interaction style is touching style that was not usually considered in the Web applications design. Changing the platform or interaction style can lead to interaction problems. To study these problems, we investigated the use of TelEduc, an e-Learning environment designed to Internet and to be used with keyboard and mouse, in two touchscreen devices, a smartphone and a tablet. Some problems are usability problems and do not have relation with the platform or modality, but other problems are related to the platform or modality changing.

Keywords—Mobile devices and services; Interfaces, interactions and systems for distance education; Interface evaluation; Usability testing and evaluation;

I. INTRODUCTION

Devices, such as smartphones and tablets, are becoming increasingly popular; most of them have touch screen displays, access to the Internet and enough computing power to process Web pages. So, Web sites and Web applications, initially developed to be used with keyboard, mouse and a medium size display, are been accessed by small touch screen devices.

One kind of Web applications is e-Learning environments, as Moodle [1], SAKAI [2] and TelEduc [3], which are applications with tools to support teaching and learning activities though the Web. These tools allow users to create content, communicate with other users and manage the virtual space.

These environments take advantages of the Web to offer content with text, images, audios and videos in a hypertext document. Tools like chat, forums, portfolios, repositories are widely used, and tools that explore the audio and video resource to user communication, such as instant messenger and video-conferences, are becoming common among the environments.

Since smartphones and tablets are easy to carry, have autonomy for hours and Internet access, the e-Learning environments' development teams are building solutions to provide access on mobile devices. Three kind of solution are

emerging: specific device application, Web site specific for mobile devices, and improve the Web site for mobile and desktop access.

Browsing Web site using another device (last two solutions), the user can deal with some problems related with the platform changing (e.g., from desktop's browser to smartphone's browser), but some problems happen due the interaction style changing (e.g., keyboard, mouse and medium screen to a small touchscreen). We call these problems as cross-platform interaction problem and cross-modality interaction problems, respectively.

So, which cross-platform and cross-modal interaction problems users deal? How to categorize a problem? Thinking about this question in the e-Learning context, we developed this work. Though a user interface analyze by a specialist using a smartphone, a tablet and a desktop, we found some problems to browsing TelEduc using two touchscreen devices: a smartphone and a tablet. These problems were classified into categories: cross-platform problem, cross-modality problem, and platform and modality- independent problem.

The next Section describes related works. Section III presents the TelEduc Project with a brief historical view, the tools and features of TelEduc e-Learning environment. Section IV shows the material and method adopted. Section V describes some identified interaction problems, and, in the Section VI, these problems are classified into the three described categories. Section VII presents conclusion and future works.

II. RELATED WORK

The e-Learning environments' development teams are building solutions to provide access on mobile devices. Three kind of solution are emerging: specific device application; Web site specific for mobile devices; and improve the Web site for mobile and desktop access.

Building specific device application allows designing a suitable user interface for the device and taking advantages of smartphone's features, such as touchscreen and camera, but needs develop an application for each mobile platform, so to be developed needs specific knowledge programming team and increases the code lines number to maintain. Moodle community offers the Moodle App [4] and Moodbile

[5], two native mobile applications with versions for the most popular smartphone’s platforms.

Moodle, since version 2.1, offers a Web site specific to mobile devices, an example for the second type of solutions for access e-Learning environments in mobile devices. Building a specific Web site to mobile device allows designing a suitable user interface for mobile devices taking account some common characteristics, such small touchscreen, but depends of the browser to access some platform features, such GPS, and increases the code lines number to maintain too.

The latter solution considers that smartphones and tablets have enough computational power to render Web pages and to do some adaptation if it is necessary, and offer the same user interface for any device. To design this kind of user interface it is necessary to do some usability studies to found barriers or user interaction problems. Disadvantages of this solution are to depend of browsers to use the mobile features and the difficult of consider many interaction styles in the same user interface.

This solution can start from a user interface design model for desktop and be improved to consider mobile devices. So in the initial design was designed thinking the user will interact by keyboard, mouse and medium size display and, allowing users accessed these applications on mobile devices, there is an increasing of interaction styles number, such touchscreen. With the interaction hardware changing the user deals with new interaction problems. Shrestha [6] points out some problems when the users try to use mobile devices to do specific tasks into Web sites designed to desktop, so the mobile Web browsing experience needs to be improved to a more mobile friendly Web site and some mobile browser improvement (here we consider browser as one platform characteristic).

Shrestha [6] considered mobile devices equipped with joystick and a small screen. Maurer *et al.* [7] did some usability studies using touchscreen mobile devices and desktop for browsing in Web sites, shows that “more and more people prefer using original content instead of the mobile version, especially for users of new generation mobile devices like iPhone and Android phones”. Another result of this work was the users prefer to use the standard Web site instead of tailored mobile versions of Web site. But Schmiedl, Seidl and Temper [8] have a different opinion; in their research they conclude most of the users still prefer tailored versions. Kaikkonen [9] shows that the standard and the tailored Web sites are both used but for slightly reasons.

Considering only the e-Learning environment context, we agree to Maurer *et al.* [7] when they argue the user prefer to use the standard version instead of mobile version of Web site, and this is one of the motivations of our work to study the third kind of solution to access e-Learning environment using mobile devices.

Here, we studied the use of touchscreen devices into e-Learning environments and distinguish the problems into categories. Shrestha [6] studied the use of joystick and a small screen to browsing into some Web sites, while Maurer *et al.* [7] considers touchscreen devices. All these researches point out some interaction problems, but do not classify them

if they happen due the platform changing or due the interaction changing.

III. TELEDUC E-LEARNING ENVIRONMENT

The TelEduc is a teaching and learning environment developed by the Nucleus of Applied Informatics in Education (NIED) and the Institute of Computing (IC), State University of Campinas (UNICAMP), and adopted in several public and private institutions, like UNICAMP through Ensino Aberto project [10].

The TelEduc environment was conceived in the end of 90, born with the Cerceau’s Master dissertation (1998), with professor Heloisa Vieira da Rocha as advisor, applying constructivism theory [10][12] in situated learning [13] or in contextualized learning [14] for teacher’s continuance formation. In 2001 February, the first free version was released over GNU General Public License (GPL), an unprecedented fact in the Brazilian Educational Software scenario. Many public and private institutions adopted the TelEduc as platform, increasing the TelEduc user’s community, and consequently, the development demand. This fact culminated in the release of TelEduc version 3.0 in March 2002. The version 3.0 was completely redesigned and optimized, reason for TelEduc project was awarded by ABED (Brazilian Association for Distance Education) in the “Research and Development about Distance Learning” category. In August 2011, TelEduc version 4.3 was released, with its user interface redesigned to improve user tasks and be more similar than popular Web sites.

TelEduc is a system that aggregate administration, management and communication tools designed to support teaching and learning activities. Some tools allow users to create content, other ones allow synchronous or asynchronous communication among users, and manage participants and courses. The course page of TelEduc 4.3 is structured in two parts: the left one (Fig. 1a) has a list of all tools available and in the right one (Fig. 1b) the content of the selected tool.

In the course showed at Fig. 1a, the teacher dispose the Course Dynamic, Agenda, Readings, Support Material, Activities, Chat, Mail, Discussion Forum, Frequently Asked Questions, Portfolio, Groups and other tools available. Fig. 1b shows the user interface to visualize a Support Material item, where the teacher can change the title, content, attach or remove files or links and see and write comments, and the student can see the item, download the attached files and

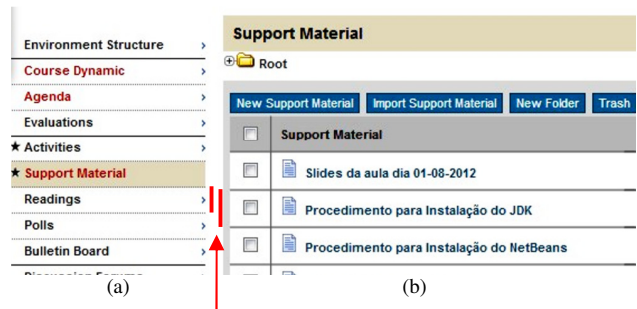


Figure 1. Height difference between menu items (a) and Support Material items (b).

visit the posted links.

Agenda is the ongoing course home page and shows the course's program for a given period (daily, weekly, etc.). Agenda is an important tool because organize the activities that must be done in a specific period, similar teachers do in the beginning of a presence class.

The Support Material is a tool that provides an area for file storage and sharing among course participants, named Support Material Area. To store an item in this area, the user needs to be a coordinator or an instructor. When the user stores an item in support material area, she can specify the sharing type: i) not shared; ii) shared only with users who have instructor role; or iii) shared with all participants. Users with student role can access the stored items published with all participant sharing type, read their content, visit their links and download their attached files.

The Readings, Activities and Frequently Asked Questions tools have similar features and user interface with Support Material tool, but different purposes. Readings tool is used to publish relevant documents, like books, magazines, news and articles. The Activities tool is an area to publish activities to the accomplished during the course, like home work descriptions. The Frequently Asked Questions tool contains a list of the most frequently questions done by the participants during the course and their respective answers.

Tools like Discussion Forums and Mail are used to participant communication, supporting text message exchange in asynchronous mode. To synchronous communication, there is the Chat tool, its features is similar to Web chat sites.

The Portfolio is a communication tool that aims to promote the collaboration among participants through the sharing of "items" (documents, presentations, programs, links, etc.). So the Portfolio tool provides an area to item storage and sharing for each participant (user or group of users) within a course.

The Bulletin Board tool is a dedicated space where all the participants can post information considered relevant to the course content.

The Agenda, Activities, Support Material, Readings, Bulletin Board, Discussion Forums, Mail and Portfolio allow users to create text content using a text editor, the CKEditor [15]. CKEditor is a third-party WYSIWYG text editor to be used inside Web pages, bringing to the Web application common editing features found on desktop editing text applications, but CKEditor version 3.3 does not work on mobile devices. In [16], we studied some problems in use of third-party software on TelEduc to create and to visualize documents, describing that mobile compatibility needs to be considered to not prejudice mobile users.

The Agenda, Activities, Support Materials, Readings, Mail and Portfolio tools allow attaching files into them items in similar way: the user clicks in the "Attach File" button, select the file to be attached using a dialog and, after the click on the save button, the file is uploaded.

To provide content, TelEduc uses the Web infrastructure, more specifically, hypertext with images, links, audios and

videos. All these media can be published as content in tools like Agenda, Support Material and Readings.

Since the e-Learning environments need to be easy to use for users with different levels of Web experience, the usability is an important nonfunctional requirement. TelEduc was designed in an iterative design-evaluation process to have good usability and the user interface does not impair teaching and learning activities, so many usability evaluations were done. The accessibility is another nonfunctional requirement desired for TelEduc, to allow impaired people to use the environment without meet barriers or obstacles.

TelEduc was designed to use a mouse and keyboard as input devices, and a medium screen size as output device. TelEduc is better visualized into 1024x200 pixels screen resolution. Visualize it into a lower screen resolution cause some user interface problems like dispose interface components in wrong position.

Since the e-Learning environments are available on the Internet, this software can be accessed by smartphones and tablets nowadays and the developers need to study how to allow all features into these devices. Access the environment in anywhere and anytime is one of the biggest attractions, but research is necessary to have a user interfaces with high usability and good user experience. To reach it, it is necessary identify cross-modality problems when TelEduc is used in touchscreen devices.

IV. MATERIALS AND METHOD

To investigate which problems happen when the users use mobile devices to browsing a Web application, we adopted the following method: a human-computer interaction (IHC) specialist, using a touchscreen device, accesses the Web application and collect interaction problems. To each collected problem, the specialist checks which ones from the three devices the problem happens and so classified into the categories: platform and modality-independent problem, cross-platform problem and cross-modality problem.

We used a Motorola Milestone smartphone [17] and a tablet PC [18]. The Motorola Milestone has a 3.7 inches multitouch display with 133MB internal storage memory expansive up to 32 GB with a memory card, 600 MHz Cortex-A8 processor and 256 MB RAM, a 5MP camera, GPS and wireless connection by Wi-Fi 802.11 b/g and bluetooth. Android 4.0.3 [19] was used as operation system (OS). Motorola Milestone has a proximity sensor, an ambient light sensor, a 3-axis accelerometer and a geomagnetism sensor to provide orientation with respect to Earth's magnetic field. To browse in the Web application, the specialist uses the Android stock Web browser. The proximity sensor and the accelerometer can be used to interact with applications, but the stock Web browser does not use these features as an input device. Only the touchscreen was used as input device.

The Tablet PC is a computing device designed to "imitate" a notebook, allowing the user interact with a pen. Resuming, the Tablet PC has the following hardware characteristics: (i) Pen sensitive screen; (ii) Screen that

allows different positions; (iii) Wireless network access by WLAN and bluetooth technology; (iv) Microphones and embedded loudspeakers; (v) Keyboard (some models the keyboard are detachable); (vi) Batteries.

In this exploratory study, we used a HP TouchSmart TX2-1040br, a 2.2 GHz dual-core processor computer with 3 GB RAM and a 12" touchscreen. The installed operation system was Windows Vista with Portuguese manuscript recognizer. This model has the design similar to HP laptops but it is equipped with the described hardware for Tablet PC. To the study case, the specialist only used touch in the interaction; she did not use the Tablet PC's keyboard or pen. Chrome browser version 22.0.1229.94 m [20] was used to navigate through Web application.

To classify the problem, the interactions in the three devices were compared (Fig. 2). If the problem appears in all devices, the probably of the problem be a platform and modality- independent problem is high. If the problem only happens in touchscreen devices, the probably of the problem be a cross-modality problem is high. But if the problem only happens in smartphone or only in tablet, maybe it is a cross-platform problem. So, to distinguish cross-platform problem, cross-modality problem and platform and modality-independent problem we needed to use these three devices. The Tablet PC and Desktop have the same platform (Windows Vista and Chrome). Tablet PC and the smartphone are both touchscreen devices and different platform (smartphone uses Android).

V. INTERACTION PROBLEMS

The specialist, using both touchscreen devices, found some interaction problems. An overview of the main problems is presented in this section.

A. Problem 1: Fat Finger Problem

At TelEduc, the tools available to be used in a course are listed in a menu disposed in the left side, and each tool is an option in this menu (Fig. 1a). TelEduc uses vertical lists not only for the tools menu, it uses vertical list to shows items to choose, e.g., in Support Material tool, each item is an option in a vertical list (Fig. 1b). An option in items lists (36 px) is higher than an option into tools menu list (23 px). This is a small difference, but the specialist points out problem to select a tool into the menu. Due to the small option menu height the specialist had problem to select a tool, triggered

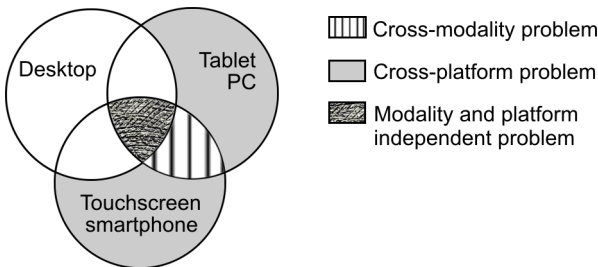


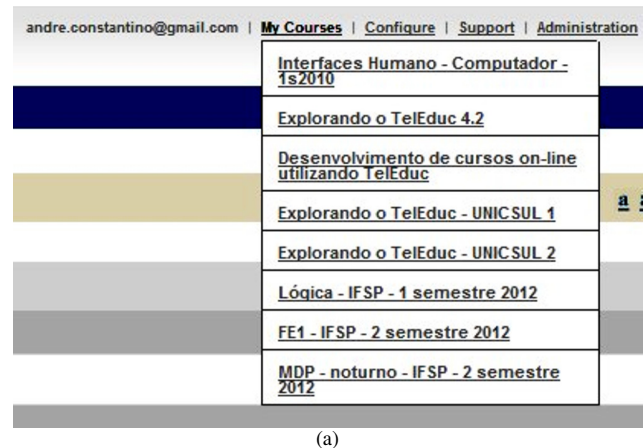
Figure 2. Categories of identified interaction problems on used devices.

the upper or down one, but this problem happen because the touched area is bigger than the clicked area when the user uses a mouse. This problem is called fat finger problem [21], when a user tries to acquire a target, the center of the contact area tends to be located a couple of millimeters off the target location—typically “below” the target [22].

The observed fat finger problem happens in both touchscreen devices, but more into smartphone probably due to page adaptation to show all content in the screen. The specialist observed the problem happens more into menu selection, because the space among menu items are not enough too large. The specialist observed the problem did not happen when selecting an item in the Portfolio or Support Material tool, probably because the distance among the items.

B. Problem 2: Mouseover functionality

TelEduc have My Course link to see the enrollment courses the user are and go to the Courses page. In desktop, the user can see the enrollment courses putting the mouse pointer over My Courses option, localized in screen right-top (Fig. 3a, the courses name is in Portuguese). If the user clicks in My Course link, she goes to Courses page. Touch screen devices does not show this menu since there is no mouseover action (Fig. 3b, the Agenda content is in Portuguese), and when the user try to see the enrollment courses, she triggers



(a)



(b)

Figure 3. Position of the My Courses menu (a) on Desktop and (b) on smartphone, the user cannot trigger the menu.

the My Course link and goes to the Courses pages.

This problem happens due the JavaScript *mouseover* function. It is common in Web applications the use of JavaScript to improve their user interfaces. But, some features of the JavaScript can cause interaction problems, like the *mouseover* function. The Android stock Web browser on smartphones makes a map between user’s action to browsing event, but there is no valid mapping to activate the *mouseover* function, since the used devices do not identify finger proximity, so it is not possible to have a feature similar to *mouseover* to do with the finger. The same problem happened on tablet.

Some of features in Table I are triggered by gesture. Using gesture into touchscreen devices the user can go forward, backward, scroll up and scroll down the page.

C. Problem 3: Gestures

Motorola Milestone and HP Tablet PC allow user interaction using gestures. To browse using these devices, the user can use one or two fingers and make gestures. Default gestures are to zooming; scroll up and scroll down a page; forward and backward pages; and select, copy and paste text. For novice users, it is not clear how to make the gesture. This is a discoverability and visibility problem as Nielsen and Norman related [23].

But, the user does not add specific gesture to use with a Web application, and the Web application cannot add specific gestures to user browser. This limitation does not allow TelEduc have gestures to create a new portfolio item, to select some Mail message or to delete a Support Material item. Since gestures have the promise to brings a powerful interaction [24], the Web application does not get all the promised potentiality, and the users gesture are limited to scroll up or scroll down a page; forward or backward pages; zooming; and select, copy and paste text.

TABLE I. MAPPING BETWEEN USER ACTIONS AND BROWSING EVENT IN THE ANDROID STOCK WEB BROWSER

| Browsing Events | Using mouse | Using finger |
|-----------------|--|---|
| Link activation | Left button click | Touch with one finger |
| Menu drop down | Right click button | Touch with one finger and hold |
| Scrolling text | Mouse click over scroll component interface | Touch with one finger and drag |
| Zooming | Not possible (needs change browser configuration) | Touch with two fingers and spread/pinch or double tap |
| Select text | Left mouse click over the text beginning and drag until the text end | Touch with one finger over the text and hold, release and drag the selection text component |
| Copy text | Select the text, click on right mouse button and choose copy | Select the text, click the option button and choose copy |
| Paste text | Right mouse button and choose paste | Touch with one finger and hold, choose paste option |

D. Problem 4: Device features

Android platform specification defines four physical buttons: Back, Menu, Home and Search. Android platform allows developers customize the reaction of these standard buttons, like use search to find in application data or show the application menu instead of the default menu. This is one of the differences between Web applications and Android applications. Web applications do not have this possibility, and the action buttons are defined by the Web browser. So the search button, instead of searching into Web application data, opens the URL field (search a page).

Smartphones have a lot of features and the Web application cannot use. The specialist tried to post a photo and a video in her Portfolio, but the browser does not upload them and does not show a message error.

E. Problem 5: Third-party text editor

TelEduc uses CKEditor to allow users write rich text instead of simple plain text (Fig. 4a). But CKEditor does not work in Android devices (Fig. 4b), but works in the tablet.

F. Problem 6: menu activation

TelEduc menu item is only activated when the user click in the menu item text, a little different from the computers menu interaction. This difference is easy to understand when browsing in the menu, since the mouse pointer does not change when the pointer is over the menu item. In the case of smartphone and tablet, the user can try many times touching the space in front of the text to understand there she must tap the text.

Table II summarizes the identified problems and the platform where they happen. The next section discusses and classifies these problems.

VI. CROSS-PLATFORM AND CROSS-MODALITY PROBLEMS

The fat finger problem (Problem 1), *mouseover* functionality (Problem 2), gesture integration (Problem 3) and menu activation (Problem 6) happened in both smartphone and tablet. Gesture integration can be classified

TABLE II. IDENTIFIED PROBLEMS AND THEIR CLASSIFICATION

| | Problem | Touchscreen smartphone | Tablet PC | Desktop | Classification |
|---|-------------------------|------------------------|-----------|---------|-----------------------------------|
| 1 | Fat finger problem | yes | yes | no | cross-modality |
| 2 | Mouseover functionality | yes | yes | no | cross-modality |
| 3 | Gesture integration | yes | yes | no | cross-platform |
| 4 | Device features gap | yes | no | no | cross-platform |
| 5 | Third-party text editor | yes | no | no | cross-platform |
| 6 | Menu activation | yes | yes | yes | platform and modality-independent |

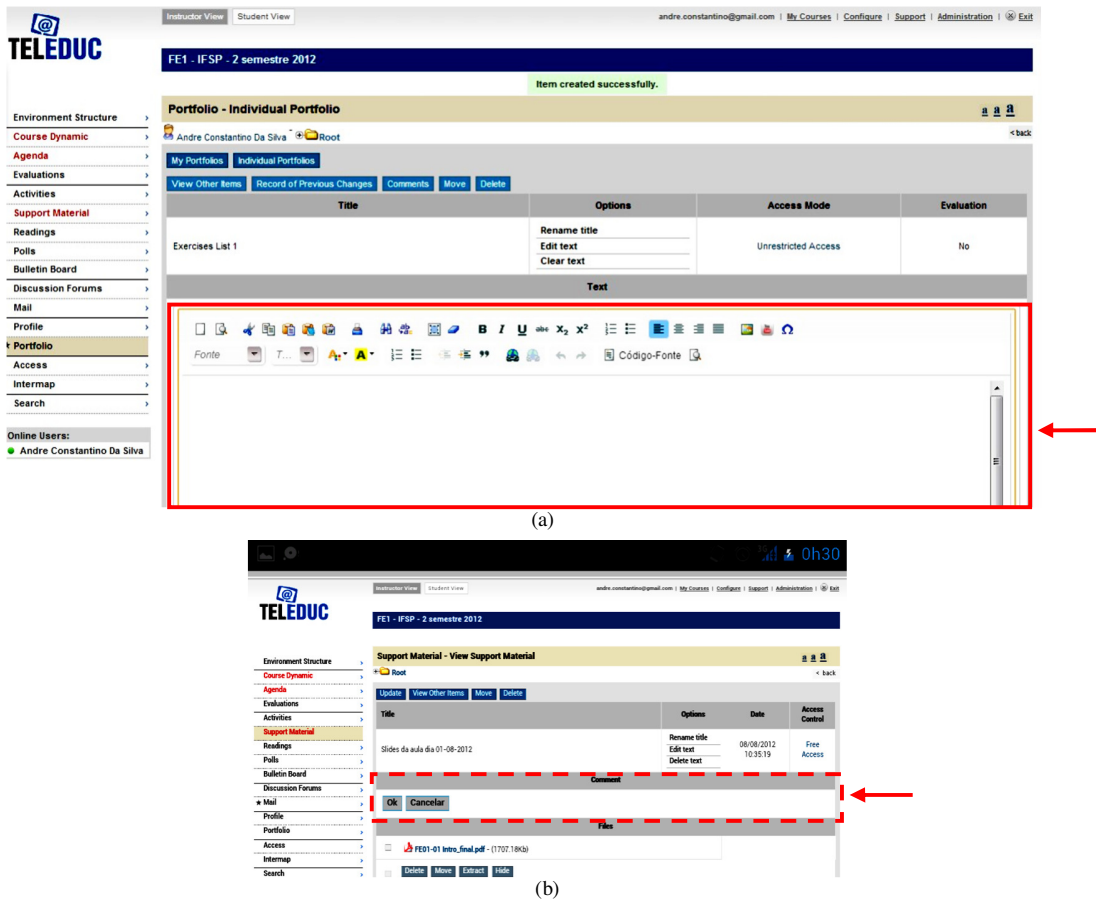


Figure 4. Position of the rich text editor in Teleduc a) the editor is displayed on Desktop and tablet PC and b) it is not displayed on Android-based devices.

as platform problem, though it happens on both devices, this problem is not directly related with touch; this is a lack of functionality. Fat finger problem and the use of *mouseover* functionality were considered as cross-modality problems. They happened when the user interface, designed to be used with an interaction style, is accessed by different interaction styles. If the application was designed to be used with touchscreens, the designer would choose the suitable interface components size and the spacing among them, decreasing the occurrence of the fat finger problem. In the case of *mouseover* functionality, the designer would choose a better way to show the menu, e.g., when the My Course link is activated, the menu is showed instead of go to Courses page.

Menu activation (Problem 6) was considered as platform and modality-independent problem, since the problem happened in desktop computers too, but when the user is using a smartphone the problem turns more severe and be easier to find.

Device features gap (Problem 4) only happens in smartphone due the difference between the Android platform and Desktop Operating Systems. This problem was classified as cross-platform problem.

The problem with no display the CKEditor (Problem 5) was classified as cross-platform problem. CKEditor works

fine in the tablet, though the specialist has difficulty using a text editor.

VII. CONCLUSION AND FUTURE WORK

Almost Web sites and applications are developed thinking to be used with mouse and keyboard as input device, and medium size screen as output devices. But smart phones with touchscreen have enough computational power to access Internet, so these devices allow users browse into Web applications by touching. When a smartphone is used to access Web applications we can have modality and/or platform changing.

Changing the interaction style or platform brings interaction problems, impacting on usability. Which problems, if we consider the software have good usability in desktop computers, appears when we used a non-specified interaction style to browse a Web application? And which problems are not related with the platform changing, just only with the modality changing? We call cross-modality interaction problems. We developed this work trying to identify cross-platform and cross-modality interaction problems using Teleduc, an e-Learning environment developed to be used with mouse and keyboard as input devices, in touchscreen devices.

To identify problems, a HCI specialist analyzed the interaction and observed problems, classified them into platform and modality-independent problems, cross-platform problems and cross-modality problems. Through this work we prove software designed to be used with some interaction styles may have problems to be used with other interaction style. In this paper, we presented 6 problems to distinguish cross-platform and cross-modality problems, and show interaction problems that are more severe when the interaction style is changed.

It is important highlight we used devices with limitation, e.g., the algorithm who decides which user interface component the user touched. The algorithm accuracy may prejudice the interaction problem identification. Disregarding these limitations, it is clear the need to a better integration between platform and Web applications to increase the user experience, gestures need be more explored when user browsing.

Another important result is the perception of the changing interaction styles allows highlight existing usability problems.

As future works, we planned to study solution for these problems, the relationship between the problem and the code and study cross-modality problems for other input devices, like pen.

ACKNOWLEDGMENT

The authors thank the CAPES and CNPq for financial support and for FAPESP through TIDIA-Ae Project (n. 05/60572-1), which provided the equipment.

REFERENCES

- [1] Moodle Trust. Moodle.org: open-source community-based tools for learning. Available at <<http://moodle.org>>. Accessed in January 2013.
- [2] SAKAI Environment. Sakai Project | collaboration and learning - for educators by educators. Available at <<http://sakaiproject.org>>. Accessed in January 2013.
- [3] TelEduc Environment. TelEduc Ensino à Distância. Available at <<http://www.teleduc.org.br>>. Accessed in October 2012.
- [4] Moodle Trust. Moodle App. Available at: <http://docs.moodle.org/dev/Mobile_app>. Accessed in January 2013.
- [5] M. J. Casany, M. Alier, E. Mayol, J. Piguillem, N. Galanis, F. J. García-Peñalvo, and M. Á. Conde. "Extending Moodle Services to Mobile Devices: The Moodbile Project". Proc. of the 6th International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2012), IARIA, 2012, pp. 24-28.
- [6] S. Shrestha. "Mobile Web Browsing: Usability Study". Proc. of the 4th International Conference on Mobility Technology, Applications and Systems (Mobility 2012), ACM Press, 2007, pp. 187-194, doi: 10.1145/1378063.1378094.
- [7] M.E. Maurer, D. Hausen, A. De Luca, and H. Hussman. "Mobile or Desktop Website? Website Usage on Multitouch Devices". Proc. of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries (NordiCHI 2010), ACM Press, 2007, pp. 739-742, doi: 10.1145/1868914.1869018.
- [8] G. Schmiedl, M. Seidl, and K. Temper. "Mobile Phone Web Browsing –A Study on Usage and Usability of the Mobile Web". Proc. of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI'09), ACM Press, 2009, doi: 10.1145/1613858.1613942.
- [9] A. Kaikkonen. "Full or tailored mobile web – where and how do people browse on their mobile?". Proc. of the International Conference on Mobile Technology, Applications and Systems (Mobility 2008), ACM Press, 2008, pp. 111-111, doi: 10.1145/1506270.1506307.
- [10] M. A. Franco, L. M. Cordeiro, and R. A. F. del Castillo. "O ambiente virtual de aprendizagem e sua incorporação na Unicamp", Educação e Pesquisa, vol. 29, issue 2, Jun-Jul 2003, pp. 341-353.
- [11] S. Papert, "Constructionism: a new opportunity for elementary science education". Proposal to The National Science Foundation. Massachusetts: Cambridge, 1986.
- [12] J. A. Valente, "Computadores e conhecimento: repensando a educação". Campinas, Brazil: Gráfica da UNICAMP., 1993.
- [13] J. Lave and E. Wenger, "Situated Learning – Legitimate peripheral participation". Cambridge: Cambridge University Press, 1991.
- [14] J. A. Valente, "Formação de Professores: Diferentes Abordagens Pedagógicas". In O Computador na Sociedade do Conhecimento, J. A. Valente, Ed. Campinas, Brazil: UNICAMP/NIED, 1999, pp. 131-156.
- [15] CKSource - Frederico Knabben. CKEditor - WYSIWYG Text and HTML Editor for the Web. Available at <<http://www.ckeditor.com>>. Accessed in January 2013.
- [16] A. C. da Silva, H. V. da Rocha. "Improving e-Learning Environment With GPL Software - The visualization content case for computers and mobile devices". Proc. of the 6th International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM 2012), IARIA, 2012, pp. 222-227.
- [17] Motorola Mobility Inc., 'Milestone Android – Phone Without Compromise - Tech Specs - Motorola Mobility India', Available at: <<http://www.motorola.com/Consumers/IN-EN/Consumer-Products-and-Services/MobilePhones/ci.Motorola-MI-LESTONE-IN-EN.alt>>. Accessed in January 2013.
- [18] HP TouchSmart tx2-1040br Notebook PC (NL266LA) specification. 2009. Available at: <http://h10025.www1.hp.com/ewfrf/wc/product?cc=us&lc=en&dlc=en&tmp_geoLoc=true&product=3937930>. Accessed in January 2013.
- [19] Google. 'Android'. Available at: <<http://www.android.com/>> Accessed in January 2013.
- [20] Google Inc. Chrome Browser. Available at: <<http://www.google.com/chrome>>. Accessed in January 2013.
- [21] D. Vogel, and P. Baudisch. "Shift: A Technique for Operating Pen-Based Interfaces Using Touch". Proc. of the 25th International Conference on Human Factors in Computing Systems (CHI'07), ACM Press, 2007, pp. 657–666, doi: 10.1145/1240624.1240727.
- [22] C. Holz and P. Baudisch. "The Generalized Perceived Input Point Model and How to Double Touch Accuracy by Extracting Fingerprints". Proc. of the 28th International Conference on Human Factors in Computing Systems (CHI '10), ACM Press, Apr. 2010, pp. 581–590, doi: 10.1145/1753326.1753413.
- [23] D. A. Norman and J. Nielsen. "Gestural Interfaces: A Step Backwards In Usability", Interactions, vol. 17, issue 5, Sep.-Oct. 2010, pp. 46-49, doi: 10.1145/1836216.18362287.
- [24] D. A. Norman. "Natural User Interfaces Are Not Natural", Interactions, vol. 17, issue 3, May-Jun. 2010, pp. 6-10, doi: 10.1145/1744161.1744163.