HyMobWeb: A hybrid adaptation of context-sensitive Web interfaces with multimodality support in mobile devices

Danilo Camargo Bueno, *Universidade Federal de São Carlos, Sorocaba, SP*
Luciana Martinez Zaina, *Universidade Federal de São Carlos, Sorocaba, SP*

***Abstract*—The use of mobile devices to browse the Web has become increasingly popular as a consequence of the easy access to the Internet. However, moving from the desktop development to the mobile platform features requests to the developers an important focus on the interaction elements which fit into the interaction demands. The frameworks front-end is the solution most adopted by Web developers for working with application's adaptation supported by responsive Web design techniques. Nevertheless, this technique has shortcomings that directly impact in the interaction elements and user satisfaction. This article presents a hybrid adaptation approach of context-sensitive web interfaces with multimodality support called HyMobWeb. The proposal treats the context sensibility and multimodality interaction in two phases static, during the development time, and dynamic, in the run-time of the application.From the approach, the developer can modify the Web mobile interfaces with adaptable features which will be executed during the user interaction. The phase static of HyMobWeb was evaluated in the perspective of developers, further demonstrating its viability.**

***Index Terms*—Adaptation of user interface; Web mobile; Context-aware interfaces; Multimodal interfaces; Framework Front-end.**

I. Introduction

In the recent years, Web developers have been facing problems regarding application suitability to small mobile devices interaction, given its capacities and resources to browsing the Web. Frequently, the development of Web application requires endeavor upon modifying and optimizing the application to be used in mobile devices, since the support of native browsers is limited and differs according to different devices [1]. Aiming at minimizing this problem, caused by devices heterogeneity and context of using, interface approaches, processes, tools and frameworks to mobile devices, have been emerging.

Responsive Web Design (RWD) has often been utilized as one solution to overcome the problem of adaptation to mobile Web. RWD is a technique which employs the adaptation of interaction elements automatically according to the screen size and the device's orientation through the changing of CSS styles (*Cascading Style Sheets*). Although RWD is considered a low-cost solution, there are restrictions in its adaptation resources as such as: (i) RWD detects contextual changes only linked to screen resolution and device orientation; and (ii) the supporting to these adaptations can be specified merely by showing/hiding elements and changing some visual attributes [2].

Recognizing the popularity of RWD, many Front-end Frameworks (FeF) - which concentrate on developing the user interaction - came up in order to enhance the construction of mobile device applications. The most popular one of its category is *Bootstrap* [3], [4]. Developers, thus, started to use FeF to maximize the mobile development since its interaction elements may be adapted according to the device's features (screen size and orientation). These solutions have promoted recent RWD trends, adjusting the delivery of content to a large range of different devices. However, the focus of these FeF concerns strictly in handling features inherited from the RWD as the different screen sizes. They do not fulfill the adaptation to different input modalities or even different contexts of use [5]. Based on the restriction of FeF, we can pinpoint that an important question regarding the elements adaptation to mobile devices is not covered by these frameworks. The resources available by RWD techniques are limited and do not allow the developers to mark the elements and inform how they could be adapted according to the context of use, or even shift the interaction mode of this element (voice, movements, etc.).

The objective of this article is to present the HyMobWeb approach, a proposal of hybrid adaptation on Web user interfaces to mobile devices. HyMobWeb enables the developer to handle features of context sensibility and multimodality. The approach extends the FeF enabling multimodality and sensibility control through the addition of adaptation points placed on the interface code. The proposal was designed based on a previous experimental study on the weaknesses of FeF in resources for adapting user interfaces. An evaluation was conducted in order to understand the perceived ease of use and usefulness of the approach in developers' perspective.

II. Backgrounds

The fundamentals of interaction patterns (focus on interface elements), responsive Web design and interfaces adaptation, FeF (definitions about the tool), multimodal interaction and context sensibility was studied. The following subsections shall describe these fundamentals and the related works.

A. Fundamentals

Interaction elements are defined as a set of interface objects responsible for a functionality and that enables the users to interact with the application (ex. buttons, hyperlinks, forms, etc.) [6]. The user interfaces' patterns arise to solve interaction problems frequently found in the interfaces. These patterns aim to depict solution to common difficulties in

the interfaces' construction; moreover, the patterns usually are important references to experienced interface developers, since it enables the development of optimized and organized solutions. There are catalogs of patterns that sort out the patterns in categories. Each category reports the type of user action or objectives such as: getting input, navigation, dealing with data, social, onboarding and miscellaneous [7]. There are several approaches to interface adaptation, usually classified into two general categories: adaptable and adaptive interfaces. The adaptable interfaces allowing the selection of the interface by the final user through specific user profile from a predefined list. The adaptive ones are controlled by the system and adapt the interface automatically for people with different characteristics/needs [8].

Within the line of adaptive interfaces, one of the solutions widely used by Web developers has been the concept of Responsive Web Design (RWD). The RWD technique suggests that the design must respond to the behavior of the user and environment basing itself on the screen size and orientation of the device [5], [9]. Media queries[1] are the basis to employ RWD that permits the definition of different conditions to the element visualization. Ethan Marcotte [9], pioneer of the RWD, quotes three fundamental principles to this kind of development: (i) flexible layouts with relative dimensions based on grids; (ii) flexible images and videos content through dynamic resizing; and (iii) usage of media queries [9], [10].

In order to optimize the construction of Web mobile applications, the FeF emerged. They combine the basic concepts of responsive development to the interface patterns to using the interaction elements in the construction of interfaces. A set of components built in HTML - Hypertext Markup Language, CSS and the JavaScript programming language composes the core of FeF. Among the various FeF, we can highlight Bootstrap, Zurb Foundation, Pure, Materialize KickStart[2], etc. Bootstrap is the most popular of its category, being adopted by a great part of interaction developers and designers. Its popularity is a consequence of the framework maturity (precursor among FeF) and the support to adapting the interfaces [3] [4].

Nevertheless, some limitation in the adaptation resources was found in FeF. One of these is in how it handles with multimodality. Multimodality refers itself to the identification of the most effective interaction modalities combination with the purpose of improving user communication and interaction with the application. Usually, multimodal systems utilize two or more modalities combinations, such as speech, touch, 2D gestures, etc. The diversification of the nature of interaction turns the multimodal interaction a suitable option to be used in adaptive interfaces [2], [11].

Another FeF restriction concerns to context sensibility which is related to an application adaptation according to changes of aspects in the context of use. The context of use can be structured along four main dimensions related to i) the user, ii) the devices iii) the environment and iv) the social relations [2], [12]. As previously mentioned, the main issue in FeF is that the contextual changes are limited to screen resolution and device orientation [2].

Discussed by Manca and Ramos [13], [14], those limitations spotlight a gap on how the Web application approaches are able to treat the development issues of using multimodal interaction and of adapting to different contexts of use. Furthermore, the interactions occurred in the physical, social and virtual context should be taken into account whenever possible [15].

The transition of desktop Web applications development to mobile has demonstrated the need of modifying interaction elements, producing new emerging patterns. This modification has been explored through the endorsement of the development model based on the RWD, sustained by features offered by the FeF. However, there are still aspects of adaptation that cannot be worked on these environments without employing too much effort by developers.

B. Related Works

Although many works discussing the interfaces adaptation topics in literature, there were no works on FeF found. In this subsection, we present works related to interfaces adaptation.

Taking into account the modification of end-user visualization forms, we can spotlight the works Semantic Transformer [16], Small Screen Device (SSD) Browser [17], Tree Adapt [18] and W3Touch [1]. All of them present different ways of manipulating a Web page's content so that it can adapt to the size of the mobile device. However, all of the four works are the focus in the content adaptation and not in the interaction elements adaptation.

Semantic Transformer [16]is an interface adapter which has the main objective of transforming desktop Web application to Web mobile. Based on a recursive algorithm, the Semantic Transformer searches and analyze each element in the Document Object Model (DOM)[3] of the application. Its adaptation algorithm calculates the performance cost to the exhibition of each element in a mobile device. After the analysis, the system classifies the best element to be exhibited and then modifies the original interface structure of the application. With this interface adjustment, the tool can create ramifications of new interfaces in run-time to separate satisfactory contents to fit into the screen specifications. Nevertheless, unfolding in other interfaces may hinder user understanding, often mischaracterizing the interaction pattern that was originally designed.

Small Screen Device (SSD) Browser [17] presents a new desktop-mobile Web adaptation method based on the analysis of the HTML document's structure that is performed in two steps: visual and structural analysis. . The visual analysis utilizes a set of four heuristics in supporting the identification of positioning elements on the interface. The structural analysis is guided by six predefined heuristics checking the elements of the document and their relation. The two analyses - visual and structural - occurred in combination with the intention of identifying linked contents, and possible elements to be adapted. The ten heuristics definition was the result of a study

[1] *media query*, a CSS technique in which, presentations can be adapted according to the features of devices.

[2] Frameworks: getbootstrap.com, foundation.zurb.com, purecss.io, materializecss.com, getkickstart.com

[3] Programming interface which, enables the content, structure and document style manipulation.

involving several popular sites. Besides the adaptation based on the information retrieved from the page regarding structural and virtual analyses, the adapter enables the user to configure some adaptations. So the interface adaptation is achieved by all these analyses and combination of information.

The Tree Adapt [18] provides a consolidated visualization of a Web page adapted through the *toggle* functionality (content expansion/contraction). Its focus is in the exhibition form and not in the resizing and in the elimination of unnecessary content to fit into mobile devices. This visualization is created by transforming of all content of a Web page to the tree of HTML elements. Starting from the structure, a selection of relevant content is performed and new functionalities are added to the elements. Through five stages, the adapter algorithm carries out the creation of the tree, the identification of important blocks, the content organization, the object selection and the content identification. The result is a visualization summarized by blocks' title. Users can visualize the whole content of a block by clicking on the title, which triggers the block's expansion.

The W3Touch [1] is a set of tools that produce analyses of user interaction to support the detection of possible design problems. It utilizes three main approaches: the injection of JavaScript code to track touchscreen events (interaction tracking), the configuration of metrics and adaptation rules corresponding to the device and its visualization conditions (adaptation catalog). In its core, there are the metrics utilized to both, data analysis and definition of adaptation rules. Two of them, error correlation and zoom level, are used as default. Supporting by the data gathered and the rules previously defined, W3Touch can automatically detect and suggest page components which need to be adapted to the specific configurations of the devices.

Manca [13] proposes a Web interface adaptation approach by combining graphic and voice modalities. The applications are built in a specific environment where is detailed the logic of the application as well as rules employed in the adaptation. Subsequently, such information is used as input to the adaptation mechanism for choosing the suitable adaptation to the device. To treat the voice modality, two API (Application Program Interface), Automated Speech Recognition (ASR[4]) and Text-To-Speech (TTS[5]), were used from an extension of the Google Chrome browser. As the proposal demands a browser extension and since the mobile browser does not support extensions, a new one was developed to the mobile environment. In the approach construction step, each interface element is marked as a CSS class and based on this and in adaptation rules, the multimodal interface is automatically generated.

Despite the works propose diverse solutions to problems of interfaces' adaptation, some issues can be pinpointed. The Semantic Transformer, Tree Adapt, SSD Browser, W3Touch approaches indicate straightly the topic of adaptation to the Web environment, but do not concern itself with supporting the multimodality adaptation. The solutions presented by SSD Browser and by Manca [13] introduce barriers to their adoption due to the demanding of installing new mobile browsers. Besides, there were no works found that explore the design issues providing the multimodality adaptation on web applications to attend different contexts of use.

[4]interface to the service of voice recognition.

[5]interface to the service of transformation of text to voice.

III. Exploring the Web mobile adaptation in frameworks front-end

Considering the lacks pinpointed in the related works, we decided to conduct an observation with end-users in practice concerning on the using of interaction elements which are highly adopted in mobile Web applications. The objective of the study was to explore and verify what are FeF limitations in handling with Web interaction adaptation on mobile devices. The Bootstrap was the FeF chosen, given its vast utilization by developers and designers [3]. Additionally, the other FeF usually offer similar elements and adaptation resources to Bootstrap.

Aiming at defining the scope and focus of the study, some interaction elements were selected based upon their popularity, usage importance and category. The selected elements were menu, slideshow and text input of navigation categories, dealing with data and getting input, respectively [7]. Firstly, we conducted an investigation from FeF on the utilization characteristics of each element and its potential to causing interaction problems. Table I shows the findings.

The first study revealed that FeF interaction elements only fulfilled basic features of adaptations, not taking into account different forms of interaction (multimodality) and limited to aspects of the device (size and screen orientation) in relation to the context variables of use (context sensibility).

Taking into account the results, two Web mobile applications, in which the elements previously studied were used (see Table I), were developed aiming to compare adaptations provided by them. An application named as *base* was created employing only FeF Bootstrap resources. The application was an online virtual museum that provides the study of History and Art by the exploring of museum's artworks. The same application was modified by incrementing it manually with adaptation resources, thus generated a second application called *manual*. The main idea was to verify if the increment with adaptation resources would improve the interaction in the perspective of the end user. The functional features of both applications were produced based on a software product line provided by the project "A Software Product Line to Intensify the Development of Virtual Museums with Multimodal Interaction" [24]. The first application was developed by students of the course of Computer Science of the Federal University of São Carlos. The second was modified by the researcher himself.

In the *manual* application, the interaction elements were modified so that they could be adapted to the particularities of the mobile devices. Furthermore, the multimodality characteristic was added to the elements following the recommendations of Adzic et al. [23]. The authors state that this resource brings along more flexibility to the application for treating the end-user needs. The modalities added were 2D gestures - divided in

TABLE I
FEF INTERACTION ELEMENTS - UTILIZATION CHARACTERISTICS AND INTERACTION PROBLEMS

Category: Element	**Relevance**	**Characteristics in Bootstrap**	**Interaction problems**
Navigation: **Menu**	It is the foremost elements of using in the context of mobile devices since it should support users in their navigation tasks. Therefore, its shape should be consistent and suitable to the device's screen size in order to enable the user in guiding himself and interacting with the interface. They are vital to the operation of Web applications; Must be usable independently of how the users interact with the applications [6], [17], [19], [20].	*Bootstrap* uses a pattern known as Toggle Menu. This pattern has an option list that is hidden by default until a user interaction happens and the list is expanded. Frequently, the element is seen as an icon known as a hamburger menu, represented by three horizontal lines and it is placed in the top of the page; the action of pressing it expands and compresses its visualization in vertical mode. In cases of several browsing levels, a Multi-toggle solution is usually adopted. In this solution each item can be expanded which open the browsing of the next level [6].	A number of items may transform the toggle Menu in a harmful option because the menu can be drawn in a large part of the screen and block user's browsing within the main content [6]. We may further highlight that: (a) little is explored on the resources available on mobile devices; (b) the excessive number of items may result in a longer vertical scrolling, and (c) an inadequate item size may difficult the selection of it.
Dealing with data: **Slideshow**	It is one of the most popular ways of presenting photos and also known as *carousel*. Particularly in Web applications, it is often employed to attract user attention; and is considered an essential feature of image managing systems. Besides, it allows to optimizing the use of vertical space [21], [22].	The Bootstrap solution allows the user to interact only through touch screen onto two different points: (1) through an arrow icon in the left and right sidelines of the image and (2) through little circumference symbols, one for each image, shown in the bottom of the image.	Concerning on a mobile navigation, the gallery may present some problems about ways of interaction: i) interaction limited to taps on the screen, not allowing any other type of movement, ii) reduced sizing of available screen elements and iii) difficulties in the visualization of control elements.
Getting input: **Input text**	The text box is the most one type of data input used in Web applications. Its use ranges from the ordinary data gathering to more complex input features such as input to search. The search is only one functionality through data input [17], [23].	In Bootstrap, the text box manages the inputting data only through typing in the virtual keyboard of an smartphone.	In order to test the Bootstrap resource, the data input was used as a search function. The search resource provides the user with a fast and efficient method to find information through keywords. However, the effort to typing the search terms in mobile keyboards may turn the search an inefficient resource [17], [19].

surface gestures (finger sliding movement - swipe) and (touch screen - tap) - and speech. The available adaptations to be performed in each are described henceforth.

In *manual* application, Toggle Menu, the solution proposed by Bootstrap, was replaced by another composed of the Side Menu and Multi-toggle Menu patterns. Using the Side Menu by tapping a button in the top of the page, the users reveal the menu content covering only one side of the page. For Multi-toggle, each menu item is extending and the user can browse to the next level. This new solution can improve the user experience since the menu can withstand a bigger amount of options; besides making the navigation experience similar to native applications [6].

Moreover, the multimodality was added and enabled the swipe movements for two interaction elements. First, allowing to the user to reveal the menu at any point of the application. Second, adapting the slideshow element enabling the user, beyond traditional interactions, to make swipe movements to change the images. Similarly to the first solution, this one is close to the found in native applications.

Finally, in the input text element, came the addition of interaction by using voice modality, which was linked to the search feature of the application.

After the development of the applications base and manual, a controlled experimented was conducted. The experiment aimed to verify the acceptance or non-acceptance of the adaptation aspects inserted into the manual version in the end-users' perspective. Following the guidelines of Wohlin et al [25], the experiments controlled throughout the stages of planning, execution and evaluation of results, subsequently described.

TABLE II
RESEARCH QUESTIONS AND HYPOTHESIS

Research Questions	**Hypotheses**
RQ1: Does the responsive Web design resources of Bootstrap allow to perform adaptations more efficient than the adaptations manually added?	**H01:** The efficiency of manual and base adaptations is not significantly different. **HA1:** The efficiency of manual and base adaptations is significantly different.
RQ2: Does the user's satisfaction with the application manually adapted is greater than with the one which uses the native responsive Web design of Bootstrap?	**H02:** The user's satisfaction on the manual adaptations and traditional ones is not significantly different. **HA2:** The user's satisfaction on the manual adaptations and traditional ones is significantly different.

1) Planning: In the experiment planning, the hypothesis' formulation, variables selection and participants, a project of the experiment and its validity considerations are specified. Initially, in order to direct the study, two research questions (RQ) and four hypothesis were defined (Table II).

A total of three tasks (signed by T), which two treatments, were designed based on the interaction's features of each selected element. Each task should be performed in both

manual and base application. In task 1, **T1 - menu**, the user would search in the menu a specific item among the available ones; **T2 - slideshow** similarly T1, the user would navigate through the photos' gallery to find an image; and **T3 - text box** the users should input in the field a keyword to be searched. A tutorial to guide the users on performing the tasks was provided.

Aiming to answer RQ1 and considering the tasks previously outlined, some metrics were defined: time and number of movements to complete a task. In order to answer RQ2, the Self-Assessment Manikin (SAM) was adopted. SAM is a non-verbal pictorial evaluation technique and aims to verify user's emotion as a reaction on using a system. User's emotion is evaluated in three dimensions: pleasure (positive or negative emotional reactions), arousal (body stimulation from experience) and dominance or feeling of control (sensation of being or not in control of the situation). After an interaction, the user selects a picture whose represents his emotion for each dimension. The pictures are arranged in a scale ranging 1 to 9 [26].

For gathering further information, a pre-questionnaire (user profile questionnaire) were elaborated requesting data of participants' profile, such as age, the level of education, the frequency of accessing the Internet on mobile devices and what were the resources he mostly used in the device.

The following steps were defined in order to conduct the experiment: **Experiment overview** - explanation of the study features to the participants: goal, instructions of conduction, of how results will be evaluated, as well as the presentation of the independent and informed term of consent. **Application of user profile questionnaire** - the participants answer a seven question pre-questionnaire of their preferences and characteristics. **Tasks in action** - each participant performs a set of pre-established tasks. By the end of each task and for each element, the participant informs his opinion on interaction with the element by answering SAM questionnaire. In order to complement data gathering of the user perception and to support a subsequent analysis, the user interaction would be filmed[6]. In addition, the researcher has developed a tool to catch data about the time a user spending in a task and his number of movements that he made. Before starting the experiment, two individuals performed a pilot test to make a previously evaluation of the experimental materials and methods (pre-questionnaire, a list of tasks and applications).

2) Execution: The participants of this experiment were volunteers students at Federal Institute of São Paulo - Campus Itapetininga - who have attended the course of Maintenance and Support of Informatics and have been selected by convenience.

Firstly, the participants accepted and signed the independent and informed term of consent about the use of data and images gathered in the study. The participants were daily users of mobile devices; such users' features avoided a bias in the study which could be caused by the misunderstanding on using the smartphones. In addition, the use of the applications was balanced preventing of another bias. As the participants performed the same tasks in both applications - manual and base - their contact with the first one could influence the use of the second. Hence some participants have had their first experience with manual application and others with the base one.

A total of 18 users has engaged in the experiment. Most of them aged 16 to 18 years old (72.2%), attended high school (77.8%), and accessed Internet on mobile devices frequently (94.4% daily). Fitting to the infrastructure conditions of the environment, the experiment was performed in two parts. In the first one, 8 users completed the tasks T1 and T2; and the second one, 10 participants performed T3. The participants were separated into two groups, given Internet access was mandatory to execute task T3 and it was not available at the time of the first part.

Initially, the participants received the instructions about the conduction of the experiment and its artifacts. The tasks were presented in textual form on the device screen. The participants had no set time to conclude the tasks; besides, all of them used the same device to prevent hassles about versions of the mobile platform [7]. Moreover, a tool - developed by the researcher - was installed to catch time and movements data automatically.

Before performing the tasks, the users answered the user profile questionnaire. After that, the participants performed each task and upon completing a task, they answered the SAM questionnaire.

3) Evaluation: Based on data automatically gathered and the recordings of each participant, the evaluation phase was conducted in order to seek the results of the metrics and other relevant information. The analyses of efficiency (RQ1) focused on evaluating the time spending by the user and number of movements he had done in a task. SAM answers supported the user satisfaction analysis (RQ2).

Table III shows the cross-checking of data: user, task and its respective treatment. For this, we considered two groups (GR) of participants, BF for those individuals who use the base application (adaptation was implemented using Bootstrap) and MF for who used the manual application (adaptation was manually implemented). Moreover, table shows time spending to perform the task on milliseconds (TE), the number of movements (NM), pleasure index (PI) and dominance index (DI). Reinforcing that the users who performed the tasks T1 and T2 are not the same who executed T3. The analyses were conducted by each interaction element aimed at verifying the hypothesis outlined in Table II.

The first analysis compares data of task T1 in the menu element in both applications. Since both samples - manual and base - did not have normal distributions, the paired Wilcoxon test was employed to the metrics resulting in $p = 0.01563$ to time metric and $p = 0.01562$ to movement metric. Taking into account the p <0,10 and the medians of time and movements (see in Figure 1) we can perceive a decrease in the number of movements and time in the manual application. We can conclude that "the efficiency of manual and base adaptations

[6]*AZ Screen Recorder* is installed in the user device and allows recording the sequence of the user interaction.

[7]*AZ Screen Recorder* requires a particular version of some mobile platforms to work.

TABLE III
USERS' OUTCOMES IN THE TASKS.

		$T1_B$ (base - menu)				$T1_M$ (manual - menu)			
ID	*GR*	*TI*	*NM*	*SI*	*IC*	*TI*	*NM*	*PI*	*DI*
1	BF	10645	10	8	8	5347	4	9	9
2	BF	16196	8	8	5	1405	8	7	8
3	BF	12173	8	9	8	4902	4	9	8
4	BF	10012	8	7	5	6023	4	8	8
5	MF	8443	6	7	7	8657	5	6	8
6	MF	11219	7	9	9	5343	3	9	8
7	MF	9942	7	6	6	6934	5	9	8
8	MF	36400	18	9	9	8092	4	9	9
		$T2_B$ (base slideshow)				$T2_M$ (manual slideshow)			
1	BF	13134	9	7	9	10989	9	9	8
2	BF	20582	14	5	5	9122	10	9	9
3	BF	12146	10	8	8	8701	9	8	8
4	BF	20143	12	5	3	13654	9	7	8
5	MF	18581	10	8	8	13580	9	8	8
6	MF	18261	10	6	9	13297	9	9	9
7	MF	16711	13	5	5	13449	10	8	8
8	MF	18675	10	8	9	14957	10	9	9
		$T3_B$ (base input)				$T3_M$ (manual input)			
9	BF	38002	-	9	9	26005	-	9	9
10	BF	15981	-	8	9	5989	-	9	9
11	BF	46011	-	5	5	7999	-	9	9
12	BF	21003	-	8	9	23987	-	8	9
13	BF	29018	-	9	9	13019	-	9	9
14	MF	17009	-	7	8	24013	-	7	7
15	MF	11001	-	7	9	11987	-	9	9
16	MF	22011	-	4	9	12976	-	9	9
17	MF	16015	-	7	8	6000	-	8	9
18	MF	16989	-	1	7	10992	-	8	8

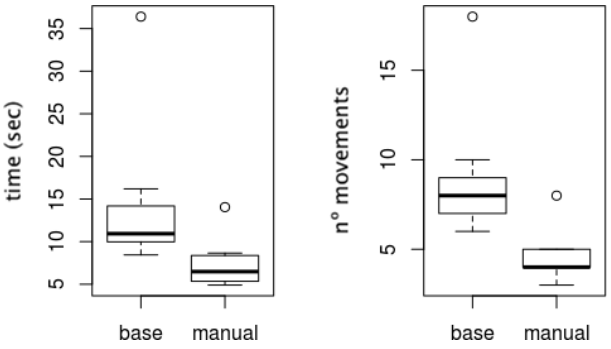

Fig. 1. Boxplots of task T1 to menu comparing the metrics of time and movements.

is significantly different" accepting the alternative hypothesis HA1, and consequently, rejecting the null hypothesis HA0. This evaluation brings up evidence that proposal adaptations (base application) can improve the efficiency of the user interaction when compared to the adaptation provided by FeF Bootstrap.

The outliers on both graphics (see Figure 1) probably appear in consequence to the lack of attention of two users in performing T1. In the user's recordings, we could see that they forgot the task during a time what caused the increase in the time and number of movements to conclude T1.

Regarding user satisfaction, Figure 2 shows that the difference between the medians of the two groups revealing that the most of the users' choices to manual adaptation concentrated in the top level of pleasure dimension; for base adaptation, the selections were distributed. For the dominance dimension,

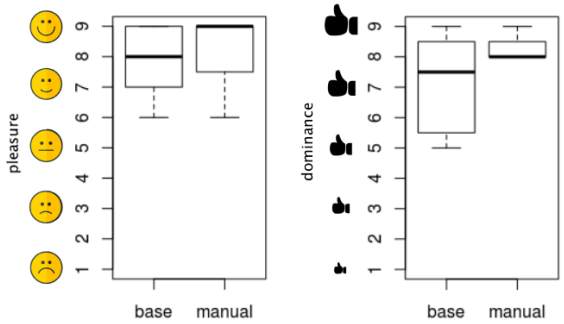

Fig. 2. Boxplots of pleasure and dominance indexes - menu

an even higher concentration of users' choices in the superior scales of SAM to manual adaptation and a higher distribution to the base is found. Those results were achieved based on the manual modification of menu element with the addition of the multimodality; the modification turns the interaction in the application more close to the native interaction giving to the user feeling of greater control. Thus, we accepted the alternative hypothesis HA2 in conclusion of that "the user's satisfaction on the manual adaptations and traditional ones is significantly different"; as consequence, the null hypothesis H02 was rejected.

The analysis of recordings improved our conclusions about the adaptations. Some contributions may be outlined: (a) less vertical scrolling - the user could activate the menu from any place of the application; (b) the experience in the navigation more close to native one; and (c) reducing the number of movements for more complex menus. In general, through the metrics of movements and time, it can be verified an improvement on the efficiency in the tasks' execution; as well as in the user satisfaction revealed by the pleasure and dominance dimensions.

Proceeding in the same directions, eight users performed the task T2 (slideshow). The set of time data followed a normal distribution, hence, the paired t-test was applied ($p = 0.001706$). In another hand, to the number of movements, the paired Wilcoxon was used ($p = 0.03125$) since data did not follow a normal distribution.

The results show a significant time improvement of task execution, reinforcing that the interaction on the manual adaptation is easier on mobile devices. Concerning on the number of movements, however, the number of movements remained the same in both adaptations, manual and base. It was a predictable result because only the interaction mode was changed.

In Figure 3 we can see that the time medians are significantly different, and to the movements, the difference are close to null. The distinguished outlier refers to an excessive number of movements of a user did around the application since he has forgotten of the item he was searching. With these results, the null hypothesis H01 was rejected and the alternative hypothesis HA1 was accepted outlining the significant difference in the efficiency of manual and base adaptations.

Important findings of the T2 execution on base adaptation

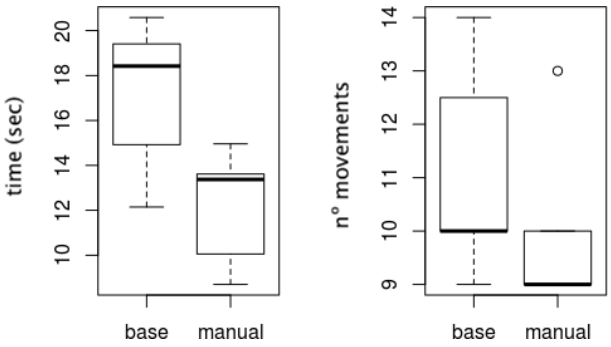

Fig. 3. Boxplots of task T2 to slideshow comparing the metrics of time and movements.

Fig. 4. Slideshow issues.

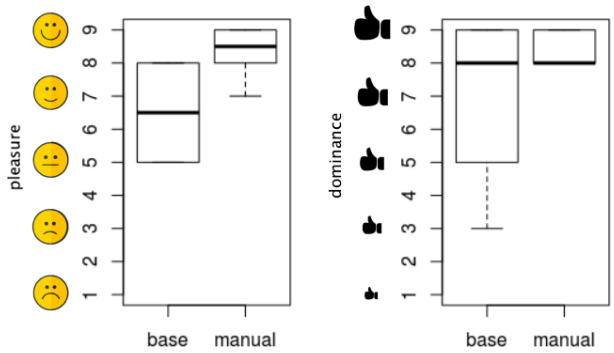

Fig. 5. Boxplots of pleasure and dominance indexes - slideshow.

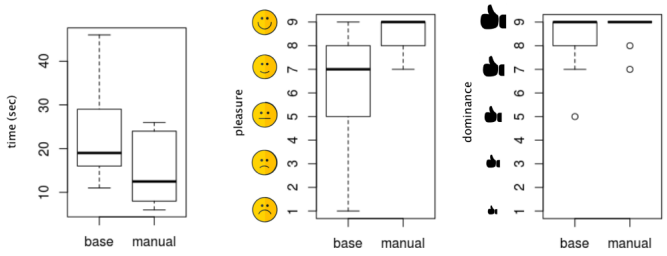

Fig. 6. Boxplots of the time and movements metrics and pleasure and dominance indexes - text box.

are: (i) the users attempted to perform the swipe movement in the gallery even with the instructions explaining the use of the traditional way (through the touch) highlighting the expected users' interaction with the component; (ii) the users presented difficulties to click on the traditional arrows because they are of reduced size resulting in a greater number of movements in the accomplishment of the task; (iii) the users had difficulty finding the arrow icons used for navigation due to their reduced size and the transparency applied to the element (Figure 4 illustrates this scenario); and (iv) Bootstrap uses the pseudo-class "hover" [8] in order to increase the opacity of the navigation arrows of the gallery, however, the use of "hover" has problems on mobile platforms [27].

Figure 5 shows data of SAM for the slideshow. They suggest an increase in user satisfaction (pleasure dimension) over the adapted photo gallery when they used the manual adaptation. While base one divided the users' opinions, manual adaptation concentrated data in the top level of the scale. In the dominance dimensions, the medians remained the same, however, a wide distribution was found in the base adaptation which discloses the reasons of why the participants have difficulties in using the element. Based on data, the null hypothesis H02 was rejected and the alternative hypothesis HA2 was accepted. The development of manual adaptation with features and appearance more close to the native one allows the mitigation of some problems not addressed by the framework. In general, we can achieve an improvement in the time of accomplishment of the tasks and in the satisfaction of the user when the component has been adapted.

A total of ten volunteers participated in the evaluation of the text box (task T3). In this step, only the time of accomplishment of the tasks was considered since it was used the voice mode and no movements become necessary. Because the sample had a normal distribution, the paired t-test was used and resulted in a value of p = 0.04892. The time medians of attainment of the task shown in Figure 6 take to the rejection of the null hypothesis H01 and the accepting of the alternative hypothesis HA1 revealing that the manual and base adaptation has efficiency significantly different.

Figure 6 also shows pleasure and dominance indexes when using data input in the base (textual only) and manual (using voice features). The increasing in pleasure dimension is perceived by the majority of users being placed in the highest value of the scale for manual adaptation. Such increase may be a consequence of using the voice modality which caused a decrease in the number of errors in data inputting. Visiting the recordings, we can notice that 80% of users mistyped the search keyword when using the keyboard of the smartphone; nonetheless, the same users did not make mistakes when speaking the keyword. In another hand, the dominance dimension did not present a perceptive difference, what is expected due to the fact that both modalities - typing and voice - being commonly used in native applications; thus users did feel in control in both applications.

The conclusions of the experimental study revealed gaps in FeF which have opened possibilities to proposing new improvements to the Web mobile interaction. Based on this, we proposed a new approach which is presented in next section.

4) Validity Threats: In order to avoid potential threats to the validity of the experiment, some strategies were adopted

[8]screen point where user rolls the cursor over some element.

considering four levels of treatment: internal (i), external (ii), construction (iii) and completion (iv) [25].

Firstly, the tasks were shuffled so the item requested to be located in the base menu was different of the item to be found in the adapted menu. For the interaction with photo gallery, the same idea was followed. These procedures were done because, in the pilot test, we observed that the interaction of the user was improving according to his knowledge about the element and its location, which could influence the data collection (i).

Aiming at avoiding misunderstandings on using mobile resources, users who daily use smartphones were selected so that they would be able to perform the tasks and they represented a sample of the population of ordinary users (ii).

Users were randomly separated into two groups with the same number of participants so that the order of usage of the adaptations could be modified. Hence, half of the participants used the base adaptations of Bootstrap first and thereupon the manual adaptations; and the other group of participants performed exactly the inverse order. The main focus was to avoid any kind of influence over using one or another adaptation. In this way, the users would be evaluating two different adaptations and judging the one that in their opinion stood out (iii). A tool to catch the user interaction automatically was developed. It assisted in the data triangulation allowing the crossing out of these data with recordings and observations notes; the variety of sources data provided us with a high level of confidence of information (iv).

IV. HyMobWeb approach

HyMobWeb is a hybrid approach of context-sensitive Web interface adaptation with support for multimodality in mobile devices. The hybrid strategy used by the approach is a combination of the two main strategies for interface adaptation: static and dynamic. Static adaptation has specifications of different code to meet diverse interactions characteristics implemented by the developer in coding time. Dynamic strategy, the code modification occurs in the application run-time [12]. In this article, we focused only on the static one.

HyMobWeb approach proposal is an expansion of the adaptation capacity of the application concerning the variables of the user context (**context sensibility**) in combination with the different interaction methods (**multimodality**). These two aspects are outlined by Dumas [11] as essential dimensions to be treated by adaptive systems. The author mentions that the multimodality is an important aspect of improving the usability in mobile devices, and consequently, providing a better interaction with the user. Besides of improving the user experience, the treatment of context-sensitive aspects can enhance user performance on the application [11], [13]. Furthermore, the dealing with such dimensions may add more for flexibility for user interaction in devices [14]. In this direction, the HyMobWeb approach offers to developers new possibilities of working in Web interface adaptation through the exploration of multimodality and context sensitivity resources. Figure 7 illustrates the approach and its functioning.

In order to support developers with new possibilities of implementing static adaptation, HyMobWeb proposes a Domain Specific Language (DSL) aiming at helping the development of context-sensitive mobile Web application with multimodality support. DSL has a set of codes which aids developers to implement in the application the two dimensions of adaptations. The flow following by static adaptation is illustrated in step (A) in Figure 7. Based on the application constructed by the FeF (1a), the developers use the DSL (2.1a) provided by HyMobWeb (2a) to mark the points of the element's code where the adaptations will undergo (3a) in run-time.

Through an adaptation engine (step B), HyMobWeb promotes the dynamic adaptation searching and modifying the chunks of elements' code which were previously marked in the development time (step A). In the dynamic adaptation, the engine checks each marking that was inserted via static adaptation (1b), interprets such codes and delegates the execution to multimodal (2.1b) and context (2.2b) handlers. The multimodal handler deals with the adaptations supported by the interaction modalities of the elements. The context handler is in charge of constantly analyzing changes in the variables of user context and making the required adaptations. Finally,the adaptation engine returns the adaptations code needed to the application (4b).

The handlers are responsible for taking care of an adapting dimension - context or multimodality - delivering new element code to the central adapter. The dimensions of adaptation are described hereinafter.

A. Context Sensibility

Context sensibility concerns application ability to adapt according to changes occurred in the utilization context. According to Paternò [16], these aspects may be divided into four categories related to:

- **user**: preferences, goals and tasks, physical state (such as position), emotional state, etc.;
- **technology**: screen resolution, connectivity, browser, battery, etc.;
- **environment**: location, luminosity, noise level, etc.
- **social**: privacy rules, collaboration, etc.

HyMobWeb context handler detects changes on the variables of the user context related to user (physical state) and technology (luminosity) categories; besides, it provides a DSL to the developer treat these variables. Through it, the developer can write the code of the adaptations based on context conditions. The writing of the code can be done by two distinct methods. In the first one, the developer uses CSS language which provides a higher flexibility to the code, being limited by the features in the language. The following grammar defines this resource offered by the DSL.

```
@context ( <aspect> : <state> ) { ... }
```

The grammar is composed of 4 components: the keyword `@context` which informs the context handler what is the context source that it should be treated; the `<aspect>` which is formed by the variables of the context of use (as previously listed - screen resolution, luminosity, battery level, etc.); the `<state>` in which that variable stands to the specific aspect;

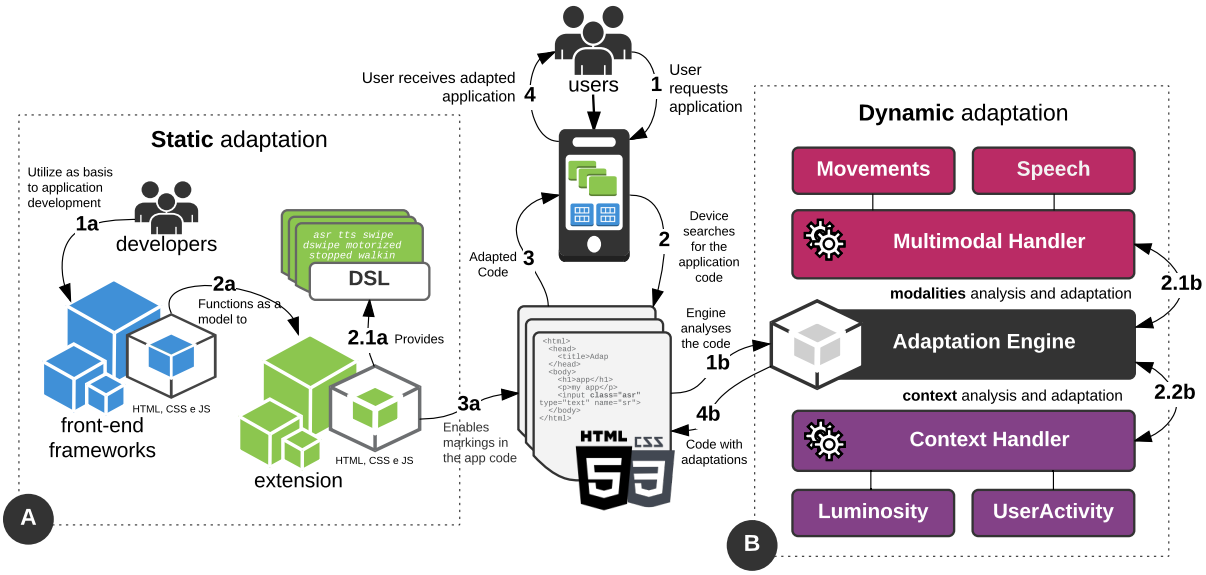

Fig. 7. HyMobWeb approach.

and the CSS code {...} that will be applied to one or more elements to their adaptations.

The adaptations will be executed when the specified *state* of an aspect is equal to the current user state - the value of context variable at the moment of using the application. This general definition to enables the extension of the HyMobWeb approach to diverse context sources. Some examples are *@context (user-activity: walking)*{ ... }, *@context (battery-level: low)*{ ... }, *@context (noise-level: high)*{ ... }. They represent the conditions of the user walking, the device's battery at a low level and a high noise level in the environment, respectively.

The second writing method uses the markup language HTML and supports the developer to alter a specific interaction element. The adaptations, in this case, are preconfigured by the adapter and will affect only the element marked by the developer. Analogous to the first method, a grammar to CSS class definition was proposed as follow.

```
.<object-behavior>-on-<aspect>-<state>
```

The grammar is composed by 3 elements: `<object-behavior>` represents the behavior that the object should take; the `aspect` reports the available context aspects (battery-level, luminosity, user-activity, etc.); the `state` describes what are the values that the aspect can have according to the type of context. Following the same directions of the first approach, the proposal also allows an extension of different states and contexts. The behavior will be executed if the aspect state is equal to the user current state.

Basically, the difference now to the previous method is in the definition of object behavior where the developer coding the element behavior through the CSS language. In this, HyMobWeb provides a list of pre-defined behaviors as such as *visible*, *hidden*, *bigger*, *darken* and etc. For instance, some potential classes can be *.visible-on-user-activity-walking*, *.bigger-on-luminosity-low* and *.hidden-on-noise-level-low*.

Although the proposed grammars are generic to be applied to different aspects, some technological requests of devices and browsers' capacities must be considered to practical implementation and extension of the HyMobWeb approach. There are a great number of aspects and because of this, we selected two aspects to be treated in this article, UserActivity and Luminosity.

The **UserActivity** characterizes a user state when he is using the application and belongs to the category of user aspects. It enables the developers to deal with the physical state of the end-user as if he is stopped, walking or in some kind of motorized vehicle. The **Luminosity** aspect, environment category, allows the gathering of information about the level of luminosity in the current environment which can be classified in low, medium and high. These luminosity values were similar to the range values defined by Kronbauer [28].

The subsequent code shows a usage example. On it, the code between the brackets will be executed when the condition defined by the developer - code inside the parentheses - is true. In the example, the first line with *@context*, represented by three dots inside the brackets {...}, is the code that should be performed in the moment that the environment exhibits a high level of luminosity (*luminosity: high*). In the second line, the code should be executed only when the user is walking (*user-activity: walking*).

```
@context (luminosity: high) { ... }
@context (user-activity: walking) { ... }
```

As a second example of our proposal, but now in HTML, the code illustrates a simple panel component to information exhibition that is commonly used in the FeF. In the example, the addition of the highlighted class changes the behavior of the component where it should shown only when the user is moving on a motorized vehicle.

```
<div class="panel panel-default
    visible-on-user-activity-motorized">
  ...
</div>
```

B. Multimodality

The multimodality concept concerns on the combination of different modalities of interaction, categorized into input and output types, with the purpose of improving the user communication and interaction on the application [2]. Examples of input modalities are speech, touch, multi-touch, pen, gestures, gaze and head and body movements. In another hand, output modalities are visual (texts, graphics, animations), speech translated to text, recorded audio and tactile (vibration) [29]. With the advent of touch-sensitive interaction, the ways of interaction had to be changed. The interactions elements, previously developed to be used through keyboard and mouse (desktop platform) have been adequate to the interaction in a mobile platform.

Aiming at meeting the multimodality issues, HyMobWeb offers new interaction modalities to the development of Web mobile applications through a similar grammar to the context-sensitive one. Three elements describes the grammar: the `<object-behavior>` represents the action to be performed by the object; the `<aspect>` reports the available aspects of interaction modalities (element, screen, movements, device etc.); and the `<state>` describes the conditions that can be used. When the object takes over a state the object performed the behaviour.

Examples of the multimodality treatments are *.listen-on-element-focus* (activate automatic speech recognition when the element receive focus), *.speak-on-user-activity-walking* (activate speech synthesizer when the user is walking), *.hidden-on-movements-swipeleft* (hide the element when the swipe to the left movement is recognized), and *.vibrate-on-luminosity-low* (vibrate the device when the ambient light level is low).

Even the proposal being generic, the technological issues must be taken into account again. Considering this, two modalities were selected as a case study - Speech and Movements.

The **Speech** modality works as data input and output through voice and is based upon two APIs. The first one, Automatic Speech Recognition (ASR), enables the user to input a value by voice channel and thus receive the return by a textual form. The second one, Text to Speech (TTS), works in an opposite way, from a textual input it returns a sound to the user.

Based on the grammar, the classes `listen` and `speak` were defined. Using ASR API, the `listen` class can translate the voice input of the user to the textual form. Hence, the `speak` class, using TTS API, synthesize the text which was inputted in the element delivery in a voice output format to the user. The classes have constraints on the elements in which they can be used due to the fact that some elements can not receive an automatic adaptation changing its feature of interaction. Therefore, the *listen* class is restricted to the input element and the *speak* class to the *label* element.

The default action to activate the modality is when the element receives focus (class `listen-on-element-focus`). However, as this is the default action, the second part of the class can be removed as shown in the following example. Besides the traditional interaction used to be by the virtual keyboard of the smartphone, the approach enables the data input by the automatic speech recognition API. The *speak* class works in the same way than *listen*.

```
<input type="text" class="form-control
    listen" placeholder="Search">
```

The **Movements** modality deals with movements making by the user interaction on the device as such as single swipe, double swipe, pinch, flick, shake, tilt, zoom, etc. The approach works with *swipe*, *double swipe* and *pinch* movements. All of them are treated in a single step since the user usually prefer to make the movements in one step, for instance, making a single swipe rather than multiple steps of gestures [20]. The movements are classified in subcategories according to their characteristics which can be directions (top, right, bottom and left) to *swipe* and *double swipe*, and into and out to *pinch*.

Taking into account the grammar, the classes were defined as *movements-swipe[left, right, top, bottom]*, *movements-dswipe[left, right, top, bottom]* and *movements-pinch[in, out]*. The values in brackets describe the possibilities of combinations for each of the movements. In following example, button will be activated when a swipe to the left is detected.

```
<button class="btn btn-primary
    tapped-on-movements-swipeleft">
    Previous</button>
```

In addition to the main classes for using Speech and Movements modalities, the developer can combine them to work in a specific context. An example is the voice mode being available only when the user is walking: *.listen-on-user-activity-walking*.

C. FeF extension

As mentioning previously, FeF may be classified as a collection of interacting elements built from user interface design patterns through the HTML, CSS and JavaScript technologies. They range simple elements from text box to more complex as modal. The main goal of FeF is to boost the development of Web applications to mobile devices providing the developer with a set of elements.

In the construction of the applications and supported by frameworks, the developer uses the FeF pre-defined structures and can modify the element by coding it to the required behaviour. The code is usually changed through the use of CSS classes which are also predefined by FeF. For example, adding a `hidden-xs` class to the `class` attribute of one component configures the element to be displayed only on large-screen devices; the FeF defines the condition to be considered a large screen device and generally uses as default parameter to the width of the device screen.

Besides the addition/alternation of CSS class, the FeF can use media queries technique to provide more refined adaptations on the interface. Media queries allow the creation of code

that alter the interface elements when they are on determined condition. The technique offers conditions that are based upon screen width and device orientation as is shown in the example bellow. The element with id "search" will undergo changings when the screen is larger than 200px.

```
@media (min-width: 200px) {
  #search { display: none; }
}
```

This development standard based on media queries and CSS classes is common to developers and to many FeF. However it has some restrictions, especially when the objective is to implement different treatments. In this sense HyMobWeb takes advantage of the already consolidated standards of code writing and offers a grammar that helps the developer to reuse proposed solutions (encapsulating some of the implementations) and hence decreasing the curve for learning to use it.

As previously mentioned, FeF works with the dimensions of multimodality and context sensitivity, but with restrictions. The treatment of context sensitivity in FeF is limited to the category of device aspects, more specifically screen size and device orientation. In the case of multimodality, the touch is the exclusive mode treated. As a result, developers have to put a lot of effort into adding context-sensitive and multimodal features to applications developed from FEF. HyMobWeb overcomes these limitations defining standards for the extension of dimensions that allow the addition of new modalities and context sources.

HyMobWeb adds the modalities of Speech and Movements providing to developers ways to work with automatic speech recognition, text-to-speech, swipe movements, double swipe, double tap and pinch. Regarding context sensitivity, the approach supports developers on the analysis of characteristics of user and environment using the contexts UserActivity and Luminosity, considering the user's physical state (stopped, walking or motorized) or the level of luminosity (low, medium or high), respectively.

Although in this article, HyMobWeb shows concrete aspects (Speech, Movements, UserActivity and Luminosity), its DSL is flexible enough to extend these aspects or even creating others. In this way, HyMobWeb can expand building new sources of context and modalities, giving the application different capacities to adapt to the needs of the individuals. The adaptability of the application can reduce the effort on user-application communication process [14].

V. HyMobWeb evaluation in developers' perspective

A case study was carried out aiming at evaluating HyMobWeb approach regarding the use of its features and ease of use from the point of view of developers. The evaluation in this perspective is extremely relevant since the developers are users of the proposal in the step of static adaptation. They will use the aspects defined in the proposed grammar. Following Wohlin et al [25] guidelines for primary studies, the study was structured in planning, execution and evaluation phases as described bellow.

TABLE IV
List of tasks

ID	Aspect	Goal
T1	Movements (swipe)	the user can browse the albums through swipe **movements**.
T2	Movements (double swipe)	the user may turn the application's volume up using double swipe **movements**.
T3	Speech	allows the user to input a search querying by **voice**.
T4	User-Activity	change the application behavior according to **user activity**.
T5	Luminosity	the application behavior alters regarding the specific level of environment **luminosity**.

A. Planning

Research questions, selection criteria of variables and participants, and description steered the study conduction. We initially defined the research questions (RQ) that should be answered in the study:

- RQ1: Do developers consider resources provided by HyMobWeb **useful** for the development of Web mobile applications?
- RQ2: Do developers consider **easy** of using the resources provided by HyMobWeb for the development of Web mobile applications?

This study was focused on evaluating the static adaptations provided by the approach and the RQs address specifically to this point. The evaluation of the dynamic adaptation (adaptation during the interaction) will be conducted in another moment and in the perspective of end users.

As the proposal of HyMobWeb is to improve users' interaction on the mobile Web application, developers with expertise on Web mobile development were invited to participating in the study. The requirement for participating was to having prior knowledge in Web software development so that they could perform all the tasks and further contribute with the realistic evaluation concerning the use of HyMobWeb.

A base application of a music player was developed to function as the initial artifact to the participants. The application had two screens and the features: i) browse among the albums available in the library; ii) visualize the album's songs; iii) play and pause a song; and iv) turn the volume up or down. The application was developed to mobile devices only using the resources of FeF Bootstrap, and therefore, with the limitations concerning context sensibility and multimodality, previously mentioned.

For guiding the evaluation five tasks were defined. The idea was for participants to use the base application and to complement the application with the functionality described in the tasks in Table IV. For this, they would use HTML and CSS code. The tasks T1, T2 and T3 concern on multimodality whilst T4 and T5 to the sensitivity to the context. In addition to the base application, participants will receive documentation on the use of HyMobWeb with the description of the aspects provided by the approach, the rules to be followed, and an example of use.

Two questionnaires were prepared to support the data gathering concerning use and acceptance of the approach. The pre-experiment questionnaire aimed to collect the participants'

TABLE V
TAM QUESTIONNAIRE

Perceived Ease of Use	Perceived Usefulness
F1 - I would find the aspect * easy to use.	U1 - Using the aspect * would enable me to accomplish tasks more quickly.
F2 - Learning to operate the aspect * would be easy for me.	U2 - Using the aspect * would improve my job performance
F3 - I would find it easy to get the aspect * to do what I want it to do.	U3 - Using the aspect * in my job would increase my productivity.
F4 - I would find the aspect * to be flexible to interact with.	U4 - Using the aspect * would enhance my effectiveness on the job.
F5 - My interaction with the aspect * would be clear and understandable.	U5 - Using the aspect * would make it easier to do my job
F6 - It would be easy for me to become skillful at using the aspect *.	U6 - I would find the aspect * useful in my job.
F7 - I find it easy to remember how using the aspect *.	**Observation**: the * represents the evaluated aspects.

profile (age, the level of education, the level of knowledge on software development and Internet access frequency through mobile devices). For the post-experiment questionnaire was based on TAM (Technology Acceptance Model) [30] and intends to catch HyMobWeb acceptance.

TAM is an important instrument used to analyze the acceptance and behavior of using an information technology. It comprises two dimensions, perceived ease of use and usefulness. Perceived Usefulness covers the level in which an individual believes that the use of a specific technology may improve their performance. Ease of use concerns on the level of perceiving of using a technology can be done effortless [30], [31]. Table V shows the questions. The questions were constructed to each of the aspects implemented by the approach: Movements (swipe and double swipe), Speech, UserActivity and Luminosity. The * marking in the question represents each of them. In total, each participant should answer 65 questions, being 13 questions for 5 aspects covered. For each question, the participant should select the degree of agreement represented in the six-point of Likert scale - ranging from "Completely Disagree" to "Completely Agree".

Besides the questionnaires, an analysis guided by the technique of Defect-Based Reading (DBR) was planned. DBR is software inspection technique that focuses on detecting specific classes or types of defects [32]. The objective of using the DBR was to perform an inspection in the code generated by the developers aiming to verify if the HyMobWeb approach was correctly employed.

A training on interface development for Web mobile applications was planned to aim at leveling the participants' knowledge. During 15 hours divided on five meetings of 3 hours the participants would work with HTML, CSS, JavaScript, Responsive Web Design, FeF based development and the Bootstrap framework. The criteria for participating would be the individual has previous knowledge in Web programming.

The case study execution would follow the steps: **Presentation** - explaining of the goal, instructions of the tasks, explanation of support documentation and independent and informed term of consent presentation; **Application of the pre-questionnaire**; and **Tasks execution** (see Table IV).

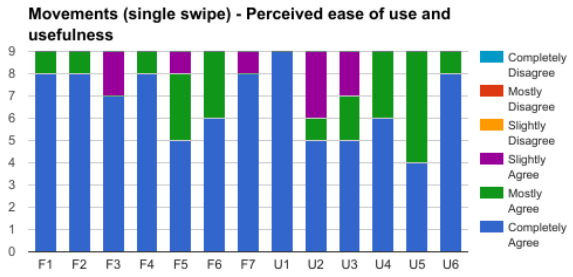

Fig. 8. T1 Movements (swipe) - Perceived ease of use and usefulness

B. Execution

The participants of the study were students volunteer of the course of Computer Science of the Federal University of São Carlos (UFScar) - Sorocaba campus. They were daily users of mobile devices and have prior knowledge in Web mobile development. Salman et al. [33] conducted a study in Software Engineering field which shows evidence that experts and students have small differences in performance in new activities. The study suggests that both profiles performed the tasks in a similar way. In this sense, the students represent a valid sample and can contribute to the confidence of the evaluation outcomes on the use of the proposed approach.

A total of 9 individuals participated in the experiment. Most are between 19 and 24 years old (88.9%) - being 66.7% with deep programming knowledge and 33.3% with good knowledge - and with frequent Internet access through mobile devices (100% accessing it on daily basis). All of them attended the training on mobile Web development technologies.

The conduction followed the planning reported in Subsection V-A. The participants answered the post-questionnaire (TAM questionnaire) for each task he had finished giving out his point of view about the approach.

C. Results Evaluation

Based on the data collected from the questionnaires and the code produced by the participants, the evaluation of the results was performed in order to answer the research questions. Firstly, we conducted an individual analysis by aspects. Figures below show the outcomes regarding the perception of the participants about the resources provided by the approach.

Figure 8 shows the result of perceived ease of use and usefulness of swipe movements. There was no disagreement regarding the perception, only variations in the degree of agreement. The developers were unanimous in reporting that the *use of the aspect Movements enabled them to develop the functionality more quickly (U1)*. In another hand, in using the double-swipe aspect (see Figure 9) we can notice that one participant reported a disagreement regarding the perception of the aspect utility. He justified that *"he considered the touch interaction much sensitive to be used for this specific functionality"*. However, the addition of modality is only an option and does not replace the previous modality available.

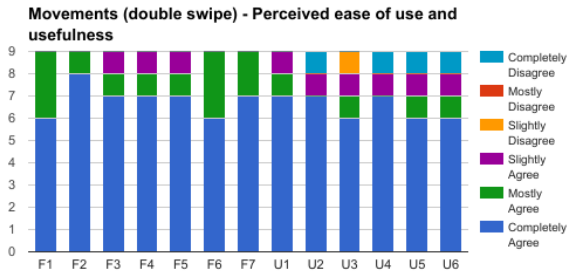

Fig. 9. T2 Movements (double swipe) - Perceived ease of use and usefulness.

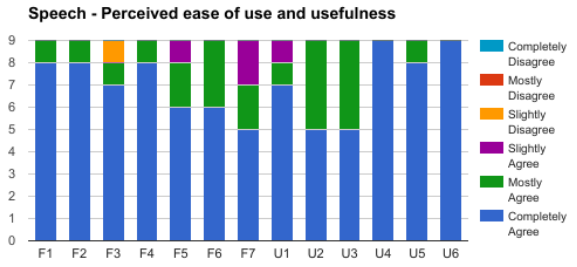

Fig. 10. T3 Speech - Perceived ease of use and usefulness.

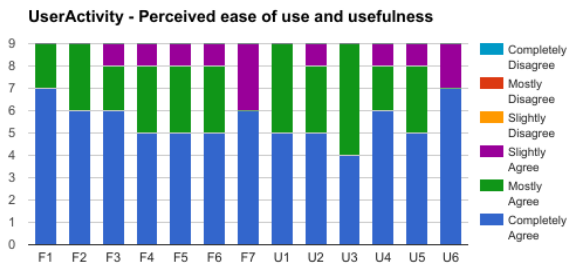

Fig. 11. T4 UserActivity - Perceived ease of use and usefulness.

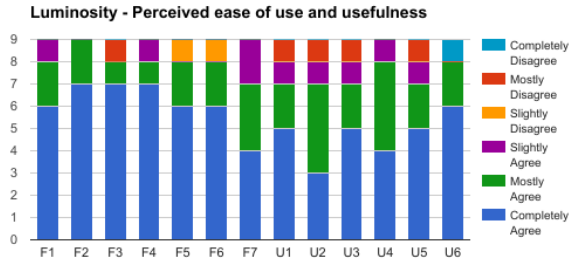

Fig. 12. T5 Luminosity - Perceived ease of use and usefulness.

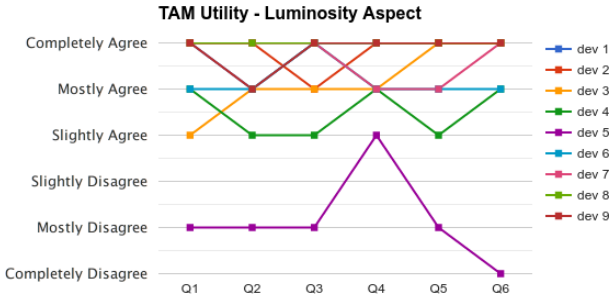

Fig. 13. Individual answers - Luminosity aspect.

For Speech aspect (see Figure 10), the most of the participants selected the agreement in both TAM dimensions. One of the developers commented: *"I consider that the communication by the aspect (Speech) incredibly useful to develop functionalities, and more natural to the user"*. As an exception for F3 (*My interaction with the aspect * would be clear and understandable*), we found a disagreement for a developer.

The lowest indexes in the questions F3, F4 and F5 were presented by the UserActivity aspect (see Figure 11). The questions stated the perception of ease of use of the guidelines and the participant understanding on the aspect. The lower level suggests that the instructions for use were not clear enough to ensure developers' understanding. Probably, this misunderstanding can be accountable for the participants' mistakes. Finally, the results suggest that a review of the documents that drive its use should be taken to improve the developers' understandings.

Considering all the TAM questions, the luminosity was the aspect with lower acceptance concrning the perceived usefulness (see Figure 12). Analyzing the data individually, we could note that one developer answers caused the decline in the index as shown in Figure 13.

Concerning the attention on the data, we could notice that the developer 5 differs from others by disagreeing in most of the questions. Analyzing his justification, we could pinpoint that his answers were based on his preference of use of the resource. This was spotlighted in some reporting as *"I do not consider useful the aspect of luminosity, I do not even like when my cell phone changes the amount of brightness depending on the luminosity. If it's something optional for the user, that's fine "*. His comments were important because it draws attention to the fact that the user should have the control over the use or not of the aspects in question. User control over the adaptations that are proposed automatically is recommended by several works of adaptive interfaces [11], [19]. The user control will be considered in future versions of the approach.

After individual analysis, the data were compiled to find the general acceptance degree of each aspect. In Figure 14 we can see that the perception of ease of use was bigger than perceived usefulness for most of the aspects and with degrees of acceptance higher than 90%. The data suggests that the developers had a positive perception of the easiness of the approach and this result points out to a positive answering to RQ2. Probably the use of development standards based on already consolidated FeF reduced the learning curve what makes the approach feasible to be adopted. An exception was found in the Speech aspect where perceived usefulness was bigger than perceived ease of use. However, the value still suggests a perceived ease of use by the developers.

Regarding the perceived usefulness, Luminosity and Movements (specifically in double swipe case) were the aspects with smaller acceptance indexes, lower than 90%. The rationale of this decline was previously discussed in the individual evaluation. In a general view, the developers perceived a utility

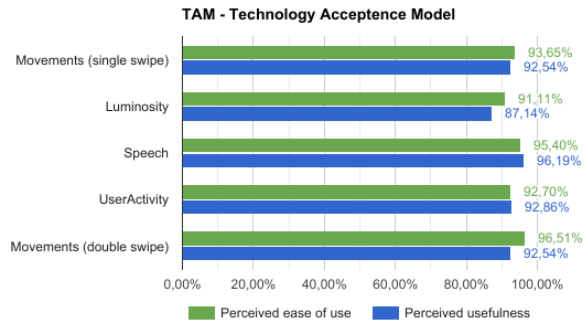

Fig. 14. TAM general acceptance degree.

Fig. 15. Defects x tasks

in the approach's aspects, suggesting indexes to confirm RQ1. However, we can pinpoint issues which must be individually worked further so that they will not compromise the application adaptation affecting negatively the user interaction.

Besides the evaluation of participants' perception, we carried out a Defect-based Reading (DBR) analysis in order to identify and classify defects which could be found in the code produced by the participants. The target is the verification of the outcomes according to code syntax and semantics concerning the proposed DSL in HyMobWeb. The defects were classified into CA (Correctly Applied), when the correct syntax and semantics was applied, PCA (Partially Correctly Applied), when an incorrect syntax *or* semantics were found, and IA (Incorrectly Applied), when both syntax *and* semantics were used in a wrong way.

For the evaluation was used a code baseline previously developed by the researcher. First, we carried out the syntax analysis verifying for each participant's code and for each task whether the code writing followed the HyMobWeb rules, available in the documentation. After that, in the semantics analysis, we compare the task requested to what the code performed. Figure 15 shows the outcomes.

In a total of 45 artifacts produced to 5 tasks only 3 presented defects. Two of them were classified as PCA for having only semantic defects in task T4 (UserActivity). In the first one, although the developer has used the correct class to the aspect, the class was placed in the wrong part of coding. In the second, the developer has a misunderstanding of the task. The task asks to hide the element when the user was in a motorized vehicle (*.hidden-on-user-activity-motorized*). However, instead of that, the developer utilized the class to show the component when the user was still stopped (*.visible-on-user-activity-stopped*), therefore, the element would not be available when the user was walking or in a motorized vehicle. The last one, classified as IA, shows both syntax and semantics defects in task T5 (Luminosity). It is important to observe that the errors happened only in the aspects which, treat context sensibility, a little-explored resource when compared to multimodality.

In general, the outcomes reported a low number of defects reinforcing the findings showed in TAM that the approach had a good acceptance and supporting the answering of questions RQ1 and RQ2, positively. Moreover, some propositions regarding the perceptions of ease of use and utility can be reported from the comments of the developers, listed thereafter:

- *"Having such tool will be practical in the overall development,* ***both in time to develop and its utility****"*
- *"The aspect mentioned* ***highly optimizes the code*** *since we should simply insert a class in replacing of the function's implementation."*
- *"Greatly functional, if we can somewhat support the user, this is good and worthwhile"*
- *"****Highly useful and easy****. It is a trend and interesting to the user, both concerning ergonomic and energy saving issues"*. The participant commented specifically on the Luminosity aspect.
- *"Excellent since the approach used the Bootstrap's "hidden-",* ***exploiting an already established semantics."***

The last comment suggests the idea of the approach as an extension of the FeF, exploiting the code writing standards already consolidated (Section IV-C).

VI. Final Considerations and Future Works

The main goal of this article was to present a hybrid adaptation approach to Web mobile development adding to FeF the treatment of multimodality and context sensibility.

Firstly, a controlled experiment was performed aiming at studying the problems of using the Web application, developed from FeF, on a mobile device in the perspective of end users. The findings revealed that the FeF should be improved to allowing the development of applications more suitable for mobile interaction in regarding of context sensitive and multimodality aspects. Besides of the controlled experiment, an investigation for workings that discusses the using of FeF on mobile devices was carried out and no proposals were found.

Based on the controlled experiment and in an investigation of the gaps existing in the FeF, HyMobWeb approach was designed. The hybrid approach is an expansion of FeF resources by adding the combination of different interaction methods (**multimodality**) and the capacity of adaptation concerning the user context (**context sensibility**). HyMobWeb uses the best practices of code writing patterns consolidated by the FeF with the intent of re-exploiting such solutions and abbreviate the learning curve.

HyMobWeb is divided into two steps of adaptation, the static and dynamic. In this article, we focused on the static one.

HyMobWeb provides the developers with a grammar which can reduce the effort of coding solutions for multimodality and context-sensitive aspects. The grammar is flexible enough to be extend aiming at attending different resources. However, in order to evaluate the approach in concrete ways, some aspects were implemented. A case study was conducted and data collected showed the great acceptance of the proposal in the perspective of developers who used it. The findings suggest that HyMobWeb brings significative contributions to the developers' work.

As future works, we intend to evaluate the dynamic step of HyMobWeb checking whether the static contribution can achieve and improve the end-user interaction on mobile devices.

VII. Acknowledgement

We thank the financial support – grant #2013/25572-7, São Paulo Research Foundation (FAPESP).

References

[1] M. Nebeling, M. Speicher, and M. Norrie, "W3touch: metrics-based web page adaptation for touch," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2013, pp. 2311–2320.

[2] G. Ghiani, M. Manca, F. Paternò, and C. Porta, "Beyond responsive design: context-dependent multimodal augmentation of web applications," in *International Conference on Mobile Web and Information Systems*. Springer, 2014, pp. 71–85.

[3] M. Otto, J. Thornton *et al.*, "Bootstrap· the world's most popular mobile-first and responsive front-end framework.'," *Getbootstrap.com*, 2015.

[4] J. Spurlock, *Bootstrap*. Sebastopol, CA, USA: O'Reilly Media, 2013.

[5] M. Nebeling and M. C. Norrie, "Beyond responsive design: adaptation to touch and multitouch," in *International Conference on Web Engineering*. Springer, 2014, pp. 380–389.

[6] H. L. Antonelli, "Navigation menus in web applications for mobile devices: issues of access and usability," Master's thesis, Universidade de São Paulo, São Carlos, SP, 2015.

[7] A. Toxboe. (2016) Ui-patterns. [Online]. Available: http://ui-patterns.com/

[8] F. Gullà, L. Cavalieri, S. Ceccacci, M. Germani, and R. Bevilacqua, "Method to design adaptable and adaptive user interfaces," in *HCI International 2015-Posters' Extended Abstracts*. Springer, 2015, pp. 19–24.

[9] E. Marcotte, "Responsive web design, a book apart (2011)," 2011.

[10] D. W. Fabri, T. Krempser, and L. V. L. Filgueiras, "Estudo de responsive web design aplicado a um sistema de pesquisa de opinião na Área mÉdica," in *Proceedings of the 12th Brazilian Symposium on Human Factors in Computing Systems*, ser. IHC '13. Porto Alegre, Brazil, Brazil: Brazilian Computer Society, 2013, pp. 264–267. [Online]. Available: http://dl.acm.org/citation.cfm?id=2577101.2577159

[11] B. Dumas, M. Solórzano, and B. Signer, "Design guidelines for adaptive multimodal mobile input solutions," in *Proceedings of the 15th International Conference on Human-computer Interaction with Mobile Devices and Services*, ser. MobileHCI '13. New York, NY, USA: ACM, 2013, pp. 285–294. [Online]. Available: http://doi.acm.org/10.1145/2493190.2493227

[12] C. E. Cirilo, A. F. do Prado, W. L. de Souza, and L. A. Zaina, "A hybrid approach for adapting web graphical user interfaces to multiple devices using information retrieved from context." in *DMS*, 2010, pp. 168–173.

[13] M. Manca, F. Paternò, C. Santoro, and L. D. Spano, "Generation of multi-device adaptive multimodal web applications," in *International Conference on Mobile Web and Information Systems*. Springer, 2013, pp. 218–232.

[14] L. Ramos and R. C. Betini, "Desafios da computação ubíqua por uma visão de ihc," in *Proceedings of the 13th Brazilian Symposium on Human Factors in Computing Systems*. Sociedade Brasileira de Computação, 2014, pp. 429–432.

[15] A. H. Kronbauer, C. A. S. Santos, and V. Vieira, "Um estudo experimental de avaliação da experiência dos usuários de aplicativos móveis a partir da captura automática dos dados contextuais e de interaąão," in *Proceedings of the 11th Brazilian Symposium on Human Factors in Computing Systems*, ser. IHC '12. Porto Alegre, Brazil, Brazil: Brazilian Computer Society, 2012, pp. 305–314. [Online]. Available: http://dl.acm.org/citation.cfm?id=2393536.2393582

[16] F. Paternò and G. Zichittella, "Desktop-to-mobile web adaptation through customizable two-dimensional semantic redesign," in *Proceedings of the Third International Conference on Human-centred Software Engineering*, ser. HCSE'10. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 79–94. [Online]. Available: http://dl.acm.org/citation.cfm?id=1939212.1939221

[17] H. Ahmadi and J. Kong, "User-centric adaptation of web information for small screens," *Journal of Visual Languages & Computing*, vol. 23, no. 1, pp. 13 – 28, 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1045926X11000656

[18] R. Anam, C. K. Ho, and T. Y. Lim, "Tree adapt: Web content adaptation for mobile devices," *International Journal of Information Technology and Computer Science (IJITCS)*, vol. 6, no. 9, p. 1, 2014.

[19] D. Zhang and J. Lai, "Can convenience and effectiveness converge in mobile web? a critique of the state-of-the-art adaptation techniques for web navigation on mobile handheld devices," *International Journal of Human-Computer Interaction*, vol. 27, no. 12, pp. 1133–1160, 2011. [Online]. Available: http://dx.doi.org/10.1080/10447318.2011.559876

[20] S. S. Billinghurst and K.-P. L. Vu, "Touch screen gestures for web browsing tasks," *Computers in Human Behavior*, vol. 53, pp. 71–81, 2015.

[21] T. W. W. W. Consortium, "World wide web consortium (w3c)," 2016, https://www.w3.org.

[22] J.-C. Chen, W.-T. Chu, J.-H. Kuo, C.-Y. Weng, and J.-L. Wu, "Tiling slideshow," in *Proceedings of the 14th ACM International Conference on Multimedia*, ser. MM '06. New York, NY, USA: ACM, 2006, pp. 25–34. [Online]. Available: http://doi.acm.org/10.1145/1180639.1180653

[23] V. Adzic, H. Kalva, and B. Furht, "A survey of multimedia content adaptation for mobile devices," *Multimedia Tools and Applications*, vol. 51, no. 1, pp. 379–396, 2011. [Online]. Available: http://dx.doi.org/10.1007/s11042-010-0669-x

[24] L. A. Zaina, A. D. Leles, A. Duarte, G. Góis, and E. F. Welter, "Uma linha de produto de software para construção de museus virtuais para aprendizagem," in *XXVI Simpósio Brasileiro de Informática na Educação (SBIE 2015) - Congresso Brasileiro de Informática na Educação*. SBC, Oct. 2015, pp. 51–60.

[25] C. Wohlin, P. Runeson, M. Host, M. C. Ohlsson, B. Regnell, and A. Wesslén, "Experimentation in software engineering: an introduction. 2000," 2000.

[26] M. M. Bradley and P. J. Lang, "Measuring emotion: The self-assessment manikin and the semantic differential," *Journal of Behavior Therapy and Experimental Psychiatry*, vol. 25, no. 1, pp. 49 – 59, 1994. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0005791694900639

[27] Mozilla, "Mozilla developer network," 2016, https://developer.mozilla.org.

[28] A. H. Kronbauer, C. A. Santos, and V. Vieira, "Um estudo experimental de avaliação da experiência dos usuários de aplicativos móveis a partir da captura automática dos dados contextuais e de interação," in *Proceedings of the 11th Brazilian Symposium on Human Factors in Computing Systems*. Brazilian Computer Society, 2012, pp. 305–314.

[29] B. Dumas, D. Lalanne, and S. Oviatt, "Multimodal interfaces: A survey of principles, models and frameworks," in *Human machine interaction*. Springer, 2009, pp. 3–26.

[30] F. D. Davis, "Perceived usefulness, perceived ease of use, and user acceptance of information technology," *MIS quarterly*, pp. 319–340, 1989.

[31] G. A. Dias, P. M. da Silva, J. B. Delfino Jr, and J. R. de Almeida, "Technology acceptance model (tam): avaliando a aceitação tecnológica do open journal systems (ojs)," *Informação & Sociedade*, vol. 21, no. 2, 2011.

[32] A. A. Porter, L. G. Votta, and V. R. Basili, "Comparing detection methods for software requirements inspections: a replicated experiment," *IEEE Transactions on Software Engineering*, vol. 21, no. 6, pp. 563–575, Jun 1995.

[33] I. Salman, A. T. Misirli, and N. Juristo, "Are students representatives of professionals in software engineering experiments?" in *Proceedings of the 37th International Conference on Software Engineering-Volume 1*. IEEE Press, 2015, pp. 666–676.