



Mobile usability testing Problems and solutions

The report at the conference
"Quality Assurance: Management & Technologies"
November 1-2, 2007
BugHuntress QA Lab
www.bughuntress.com

Mobile usability testing

Problems and solutions

The report at the conference
"Quality Assurance: Management & Technologies" (QAMT Ukraine 2007), November 1-2, 2007

Nowadays the software market is quite saturated. A certain task can be solved by dozens of competing programs with almost identical functionality. Ultimately, the choice for one or another application is increasingly based on non-functional features. And a user interface which provides easy and effective work with a software product can play one of the crucial roles when making such a choice.

BugHuntress QA Lab Company was founded in 2001. Our 6-year experience in independent software testing shows that high-quality user interfaces occur in about 70% of desktop applications. At the same time a similar number for mobile applications is merely about 40%. In other

words, software developers concentrate on valid execution of functionality while user interfaces are elaborated last.

Therefore, the issue of usability is quite acute.

This report covers the following items:

- Usability, what it means and provides
- Usability standards
- Typical mobile peculiarities
- Mobile usability problems
- Successful solutions
- Usability testing automation
- Independent usability testing and its role

What usability is

So what is usability?

ISO standard defines usability as "*the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use*".

The key words here are the last three - **effectiveness, efficiency and satisfaction** while working with a program.

What is usability composed of and what does it give to a user? There are several interpretations. We will not go into theoretical details here. Let's emphasize the points which are crucial for product success in the market.

So an interface is "usable" if it has:

- Aesthetic appeal and gives satisfaction when using the product.

If the product is not considered aesthetically appealing, up-to-date and fancy (of course, it depends on a *specified context of use* and is not the same for games and office suite), the

chances that a user will buy it are poor. If a software product didn't pass such face-control and was not bought, all the rest including functionality simply loses meaning.

- Simplicity and intuitive intelligibility of the interface (respectively, quick learning capability). Nobody would like the idea of sitting for several hours in order to learn and memorize one and only operation. If there are other programs that give the same result and can be mastered on the first attempt, one would prefer this more simple and user-friendly system.
- Ease and forethought of control elements, menu, dialog windows order, hot keys and so on which provide high speed of work with the program.

Due to this the program can be used with minimum time and effort consumption. For instance, in business applications this means higher labor efficiency of user.

- Taking into account scientific recommendations will provide little user fatigability when working with the program. Again, this means high labor efficiency of the user within long time. Such advantage of the system is fundamental for many corporate customers.

At last (but not least),

- Friendliness and user support (and, consequently, minor user errors).

This point is tightly related to the previous one. It is vital in such cases as medical systems, systems of production and transport control, accounting and analytical business applications, as well as many others, where the price of correcting a mistake can be too expensive.



Basic standards of the user interface

How are qualitative user interfaces created?

There exists a range of guides and standards providing a set of recommendations on the development and testing of user interfaces.

Here are some of the usability standards and guidelines:

- ISO 9241-11:1998 Ergonomic requirements for office work with visual display terminals (VDTs);
- Section 508 standard - Web-based Intranet and Internet Information and Applications (Rehabilitation Act, § 1194.22);

- Microsoft* Developer Network (MSDN) recommendations.

ISO basic standard defines the term "usability" and describes definitions related to it. It expands the general principles of usability and also contains particular recommendations.

For Internet-applications there exists 508 Standard. It describes methods of building static and dynamic web-pages. In MSDN one can get acquainted with the rules of building user interfaces according to Microsoft ideology (for both Microsoft Windows and Pocket PC / Windows Mobile platforms).



Mobile standards

Currently mobile technologies are developing very quickly. Handhelds have their peculiarities crucial to ensure usability. This is the reason why special guides and recommendations on usability for mobile platforms have been created. Here are some of them:

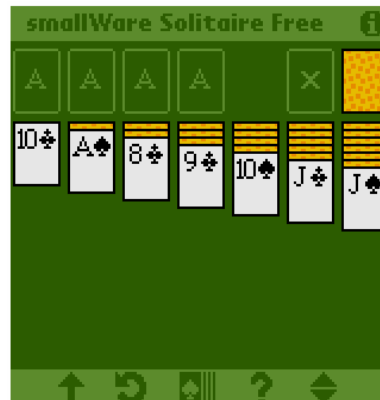
- Symbian S60 Platform Visualization and Graphic Design Guideline;
- UIQ Style Guide;
- Palm OS User Interface Guidelines.

To illustrate "usable" handhelds distinctions let's consider the following examples.

Left-handed mode

A somewhat large share of users (and respectively market share) belongs to left-handed people. For usual desktop applications left hand setting happens on the level of peripherals. For mobile applications it is already a question of interface setting.

Indeed, a lot of mobile applications (e.g., games) imply interaction by means of stylus. Mobile devices have small screens, and it is necessary to design the graphic interface for both right-handed and left-handed people. If it is done, the screen is not covered by the player's hand holding stylus while playing.



Speed or satisfaction

Small sizes of handheld screens lead to the fact that mobile software developers have to search for the compromise between ease of use and subjective satisfaction (in particular, reasonable density of buttons, icons and other controls on the screen), on the one hand, and perform-

ance and speed of work, on the other hand.

It is worthy to note that a lot of optimal solutions of this kind of problems are found at the stage of testing.

Landscape mode

One more example of mobile specificity is a Landscape mode. Desktop computers do not need this because their monitors are rarely used by rotating them by 90

degrees. But for handhelds this is done rather widely. And in this case the application should support both portrait and landscape screen formats.

General problems of usability

Here are some of issues which influence the quality of usability in the mobile software industry:

- No established mobile software usability culture;
- Lack of trained specialists;
- Direct usage of PC-approaches;
- Direct porting between different mobile and embedded platforms;
- Fast-paced software market environment.

1. First, many programmers who currently work on creation of the mobile programs were trained to develop PC software. During decades of PC software development practice there emerged programs and interface libraries that became a standard de facto. Even programmers who are not quite experienced in what concerns usability can borrow such standard interfaces, "automatically" creating fairly usable programs.

In mobile industry the situation differs a lot. Usability culture of this kind does not exist here yet. Conventional programs and libraries are few so far. Moreover, for each platform there is its own standard, the platforms are technologically diverse, and they look as well as operate differently.

2. For this reason, approaches accepted for PC are applied to mobile software. Developers directly transfer an ideology of PC development to the development of programs for handhelds. As a result, there emerge programs for cell phones and PDA with interfaces rather satisfactory if run on PC but never on a handheld. On a handheld they are inconvenient, bulky and sophisticated.
3. Lack of trained specialists. This problem in Ukraine has its specificity. A subject about user interface development is absent in the curriculum of

many specialized colleges and universities. Some of them propose a course "Ergonomics of Web-site". But this is extremely insufficient for development of successful interfaces for desktop, mobile or embedded applications.

4. The next problem is "direct" porting between different mobile platforms. In the struggle for the market share, software developers aim at making their programs support diverse mobile platforms and diverse models of handhelds. However, since the platforms are technologically different the programs are to be re-developed for almost each of them. Accordingly, their interface must take into account architecture and phone model, too. Thus, to be successful, a programmer has to acquire a wide range of knowledge. Such professionals are few, and very often programs migrate among platforms without changes in user interface which affects usability dramatically.
5. Currently software development projects are generally time-restricted. "To make faster" is the motto of our time. However, "faster" and "thought-out" do not always combine in one project. In the long run one has to sacrifice something. High-quality interface design is often the very thing that is sacrificed to a tight schedule.

Successful solutions

Nevertheless, the problems we talked above are solvable. Taptu.com search engine is a bright example of this. This innovative system is built on the basis of Web 2.0 and mobile technologies. It can be used on both desktops and handhelds. At present a beta-version of the system is released. However, it takes into account differences of diverse handhelds and suc-

cessfully adapts the content of the pages displayed to sizes and other peculiarities of mobile devices and cell phones.

Recommendations of testers, working in our company, also contributed to reaching this result.



Automation of usability testing

At present automation tools are actively used for software testing. Usability testing has its peculiarities here.

In fact, the concept of usability is tightly associated with the subjective evaluation of a user. As a result, it defies algorithmization and respectively automation.

For this reason the tools for testing automation are scarce and cover only separate activities, but not the testing process as a whole. Among them we can mention BrowserCam.com service. It helps to gen-

erate Web application screenshots on different systems with various system settings (resolution, brightness, etc.). BrowserCam.com enables to significantly reduce the testing time.

DeviceAnywhere.com service is an example of automation tools intended for mobile applications testing.

In a similar way some other products can be used. They don't perform usability testing in full but allow making it easier and faster.

Independent usability testing

What can be recommended to the companies which want to create qualitative software, also in terms of usability?

In many cases an acceptable and effective solution can be cooperation with an independent testing company.

Indeed, it immediately resolves the problem of training specialists' scarcity. Secondly, the personnel of such companies are able to not only detect problems, but

also give professional recommendations as to interface improvement.

The factor of testing independence itself is becoming of greater importance. In particular, independent testing allows obtaining an unbiased attitude to the software quality. In our practice we had the cases when an internal testing team was pushed to rather conceal bugs than detect them. In such situation independent testing is the best way to avoid problems when releasing the product to the market.



Conclusion

The users have always been and will be interested in the fact that the program is convenient and effective in use, as well as satisfactory to work with. On the other hand, high-quality usability is a weighty competitive advantage of software solutions in the present-day oversaturated software market.

Testing companies play a significant role in usability improvement. It is they who guard interests of the end user. Also, they are ready to propose their experience and resources to software producers to create programs which would be highly appealing for a user and successful in the market.



BugHuntress QA Lab is a Ukrainian testing company founded in 2001. BugHuntress QA Lab offers outsourcing software testing & QA services, dedicated testing teams, and independent verification and validation (IV&V) services.

The core expertise of BugHuntress QA Lab is testing of software for mobile and wireless devices; system software; embedded systems, multimedia hardware-software complexes; WAP, Web 2.0, e-Business applications. BugHuntress QA Lab performs automated and manual testing of functionality, usability, load, performance, localization, and other types of testing.

For more information contact info@bughuntress.com.

All rights to trademarks, product names, and brands mentioned in this document belong to their owners.