

Can Usability Improve Productivity in Banking?

Technology is the crux using which businesses garner forward momentum. It changes business processes and operations. Information Technology (IT) has overhauled the banking business completely. Today, every major bank has invested largely in IT or have lined up plans for major investments. Banks look towards IT to improve productivity, operate in a global scenario and offer services in a wider market, round-the-clock.

The banking industry looks at IT to provide software products best suited to their needs. The organizational aim for higher productivity can be achieved only if end users of a product are able to use it to its full extent; in the manner it is best intended. Users should be able to learn the software and get productive quickly. Most banks understand that a software that handles 'n' number of transactions per hour is more productive than a software that handles only a fraction of that. What is also important is the fact that a software that lets the user create 'n' number of customers per hour increases productivity of the bank by leaps, as compared to a software that lets the user create only a fraction of that. Software that increases productivity also adds to the banks' return on investment (ROI).

Usability:

Usability is a quality attribute that assesses ease-of-use for user interfaces. Usability also refers to methods to improve ease-of-use during the design process. Usability is defined by five quality components:

- **Learnability:** How easy is it for users to accomplish basic tasks the first time they encounter the user interface?
- **Efficiency:** Once users have learned the behavior of the software, how quickly can they perform tasks?
- **Memorability:** When users return to the software after a period of not using it, how easily can they re-establish proficiency?
- **Errors:** How many errors do users make, how severe are these errors, and how easily can they recover from those errors?
- **Satisfaction:** How pleasant is it to use the software package?

(The above definition is taken from www.useit.com, the website of Jakob Nielsen, a renowned usability professional.)

Usability is a structured approach to designing usable software. Keeping end users as the key audience, it builds software around their mental models, to increase productivity and for easy learning. A report published in 1999, identifies User Involvement as the top-most among the ten factors for a successful software project. 'Not surprisingly' it goes on to say, 'the absence of user involvement is a major cause for project failure. Even when delivered on time and on budget, a project can fail if it does not meet users' needs.

A usable product is one that is easy to learn, increases users' efficiency, minimizes user-errors and guides the users to resolve errors when they commit any, and results in subjective satisfaction of its users. Is productivity of its employees not what matters to banks? In the context of business in the current times, I am sure it is. Any bank would prefer to use software that allows its employees to create a greater number of customers per day, handle more transactions per day, is easy to use and enhances productivity.

Usability achieves this by studying the actual application users and the existing user interfaces to which users are already exposed to. This helps to identify typical interaction models that users may already be familiar with. The new software can leverage these behavior patterns which are used by other software, and which the users are already familiar with. Using familiar behavior patterns decreases learning time. For example, it is easier for a user exposed to a computer mouse to start using a joystick than for one who has never used a mouse.

The Design Approach:

The design team begins designing the software flow and user interface from the software requirements published by the requirements experts. The team interacts with the development, infrastructure, software testing, user education and other teams in this process. Input from all of these teams become significant aspects to consider while structuring the flow of the software.

The design input received from various teams and those from the user studies help in developing the initial design of any software. While input from internal teams help decide how various features can be achieved, input from user studies help decide what the end-users need. It would make little sense to provide features in the software that the banks don't need!

The initial designs need to be refined again to ensure that they work well in the context of the software being created and that each feature has optimal usability. It should be possible to complete each task in minimum possible time, with minimum number of clicks and with none or minimal errors. There are more than one ways of doing this. One very effective method is to do Heuristic Evaluation. Yet another is Usability Testing. Both these methods are discussed further in this paper.

Heuristic Evaluation is a process where a group of User Experience Design (UXD) experts (who are not stake holders in the design process) walk through the prototype screens and evaluate the design. Since this group of experts does not have a stake in the design process, it is not likely to be biased. Also, since this is a group and not an individual, it is unlikely to be biased by a single opinion either! This helps to eliminate any design bias that the design team might have.

Usability Testing is an approach where the actual users (bank employees in this context) are required to carry out specific task scenarios on actual software screens – raw prototype or the final ones – and while the users carry out the task scenarios, the usability testing experts observe the test users – with due respect to their sensibilities and concerns – mainly for the time taken for completing the tasks, the number of errors committed, and the number of assists needed. This input is further used to refine the screens so as to minimize the time taken to complete a task, minimize errors that occur while using the software and also to refine the help provided in the software for the users.

Involvement of actual users in such an exercise also allows banks to collaborate in the software development process. Though the involvement of users is only for a short period, it goes a long way in providing an insight to software developers to understand user requirements, their environment, their background and mental models, based on

which the product can be appropriately designed to enhance usability. An appropriate, usable and user-friendly design increases overall productivity of bank employees and bank customers.

Usability testing being an iterative process of refining and re-refining software usability, it is important for design teams to have a usability strategy for their products and keep refining the usability with every new release. By its very nature, the usability testing tends to become a long drawn process and takes time before any software version reaches a user group who can then participate in testing, again. One also needs to ensure that input from such an exercise is incorporated in the ongoing design cycle to bring in that incremental upgrade in usability.

This is essentially a collaborative process where software buyers (banks) need to strategize and invest in collaborative development with software vendors. It is only by understanding the end user that vendors can create usable software. Every usable feature of a software eventually enables users perform their tasks effectively. This leads to increased productivity of the bank.

Many other aspects go into designing software user interfaces – like manufacturability, scalability, branding, packaging, customization and user education. As the Usability team interacts with various cross sectional teams, with usability requirements of the users as a starting reference, the design that evolves is user friendly and carefully handles manufacturability, scalability as well as system issues.

Collaboration with the user education group – if your software team has one – is of prime importance in improving software usability. Creation of usable software revolves around users' mental models and how they can be best leveraged to shorten the users' learning curve. Before users actually start using the software, they know how to use it only in parts – to the extent to which they are exposed to some functionality when using other software. Everything else is new. It is the planned presentation of the new software – the way it unfolds and relates its interaction patterns to those already existing – that helps users learn newer features faster, thus allowing them to get productive quickly. Any drastic change in interaction patterns or introduction of newer interaction patterns can be exclusively taught to

the user community through user education. The potential benefit of interacting with the user education group can be leveraged at the design stage itself. In the later phase, it helps the user education group plan user education appropriately.

Productivity:

What is productivity? Productivity is a measure of the efficiency of production, and is most often expressed as a ratio of output over input. For software, one would measure time as the critical input factor and achieving the goals – e.g. creating customers on the system, doing transactions, etc. – as the measurable output.

Usability as a Productivity Enabler:

Usability leverages on users' prior experience on systems they have been using, and hence when these users start using the banking software, they would have to spend little time in understanding how the software behaves. A good example from real life can be of someone who is familiar with a city and therefore zips past around as compared to someone who is new and therefore slow, and will fumble at many places before he actually reaches his destination.

Besides, software developed with full understanding of the users' needs would go on to minimize the number of clicks, and therefore facilitate faster user inputs. With a minimal system response time and having minimized on hits to the server, it would have optimized on time – which is a valuable feature in this case.

Software features easy to memorize are equally easy to use and therefore increase productivity. The need to look around, or stress memory, or refer the help files every time, asks the user to invest time before he can deliver anything, and this is a drag on productivity.

Guiding error messages, instead of being just informative, can help users quickly resolve errors rather than just knowing that an error has occurred. Besides, for data capture pages, hints provided upfront help users understand how to enter data rather than showing up error messages when the system has thrown errors. Avoiding errors would save time in the first place by not letting errors happen!

A well-designed information architecture (a term used for the organization of on-screen information in software) helps present relevant information when it is needed, and saves user, time when he is looking for it. At the same time, it ensures not too much information is presented that can confuse the user. Any feature that is in the software but not easy to find, is as good as not being there. Good information architecture will ensure all software features are known to the user and available when required. It saves the user a lot of time and lets him concentrate on his tasks rather than look around for the features.

All these features add to software usability, and enable users to do their tasks in minimal time. No errors or minimal errors also eliminates irritants in software usage, develops user confidence and brings in subjective satisfaction. Overall, this definitely increases productivity by leaps. Good usability should amount to good business!

Conclusion:

A good product is a usable product. Usability includes all features that make the software usable for its intended users. It encompasses installation, learning, being able to perform all intended operations of the software in the most effective manner with the quickest turnaround time, thus enhancing productivity. It also means doing things with none or minimum errors and with none or minimal prompts/ assists. If users can accomplish their intended tasks with the software, the software is right on target!

Acknowledgements:

- Shanmugam Rajasekharan, Head, UXD Group, Infosys Technologies Ltd. India
- Sunita Shellar, Technical Documentation Head, Finacle IBU, Infosys Technologies Ltd
- Finacle IBU of Infosys Technologies Ltd. India for its insights into Software Product Development Lifecycle
- www.useit.com by Jakob Nielsen
- Gartner Inc Research Papers:
 - Findings From the 'Emerging Technology Trends' Research Meeting: Usability as a TippingPoint (ID Number: G00130542, Publication Date: 22 August 2005)
 - The Human Impact of Business IT: How to Avoid Diminishing Returns (ID Number: G00125740, Publication Date: 7 January 2005)
 - The Standish Group International, INC.
 - Chaos Report published in 1999

Author

Bipin Gupta

Finacle - Usability Specialist
Infosys Technologies Ltd.



Infosys Technologies Limited, Plot No. 44, Electronics City, Hosur Road, Bangalore - 560100. India • Tel.: +91 80 28520261 • Fax: +91 80 28521747
e-mail: finaclemktg@infosys.com • www.infosys.com/finacle

"COPYRIGHT NOTICE: Copyright ©2009 Infosys Technologies Limited, Bangalore, India. ALL RIGHTS RESERVED." Finacle logo is a registered trademark of Infosys and Infosys acknowledges the proprietary rights of the trademarks and product names of other companies mentioned in this document. Infosys believes the information in this publication is accurate as of its publication date; such information is subject to change without notice.