Multipurpose Public Displays:

How Shortcut Menus Affect Usage

Vassilis Kostakos, Hannu Kukka, and Jorge Goncalves ▪ *University of Oulu*

Nikolaos Tselios ▪ *University of Patras*

Timo Ojala ▪ *University of Oulu*

This study of an iterative, longitudinal deployment of a multipurpose public display examines two mechanisms that help users find the available applications: a quick-launch menu and a browsable application directory. Using the measures of relative and absolute utility, the study reveals these mechanisms' complex effects on application usage.

Public-interactive-display research and technology have come a long way, and interest in this topic is increasing substantially. A particular research challenge is managing the transition from single-purpose to multipurpose public interactive displays (or hotspots), which could have dozens of applications. This transition raises a number of issues in such systems' design. For instance, given these systems' public nature, how should the applications be presented and organized, and how should users browse them? Unlike smartphones, tablet computers, and laptops, public displays can't be customized for each potential user: one size must fit all.

Here, we identify how menu structure affects the use of public interactive displays. Specifically, in displays with dozens of applications, optional shortcut menus (*wizards*) can help users avoid searching and exploring the whole application directory for a specific interest. But how else do these shortcuts affect the display's use?

Measuring Utility

Wizards let users launch a subset of available applications with one touch. The alternative is to launch an application from the application directory, which contains all the applications. For our study's purposes, we assume that the applications in the wizard are also in the directory.

We're interested in more than simply measuring which mechanism is used more. We assume that people use displays because they offer some value to them, such as meeting entertainment or information needs. So, to quantify an application's value (popularity) within the ecosystem of applications on a public display and how the application-launching mechanism affects that value, we use *relative utility* and *absolute utility*.

Relative Utility

The relative utility of application A is the portion of the total application launches on the display, registered during a day, that are attributed to A:

$$Utility_{\text{relative}}(A) = \frac{Launches_A}{Launches_{\text{total}}}.$$

The relative utility quantifies how likely a person using the display is to launch a particular application. It depends mainly on the application's purpose and functionality and the likelihood that a user will discover that application's existence. Someone interacting with the display is more likely to launch an application if it's useful. Furthermore, if an application's functionality remains constant over time, we can probably attribute substantial changes in its relative utility to its placement in the wizard.

Absolute Utility

We couldn't directly measure the number of people near a display. So, to estimate this number, we relied on a proxy variable: the number of unique Bluetooth devices detected in the environment.

 Published by the IEEE Computer Society

Other studies have shown that this variable has a linear and monotonic relationship to the number of people in the environment.[1]

We define the absolute utility of A as the portion of people in the environment who launched A:

$$Utility_{\text{absolute}}(A) = \frac{Launches_A}{Bluetooth\ devices_{\text{total}}}.$$

Unlike relative utility, which reflects competition between applications to attract users' attention and launches, absolute utility quantifies the extent to which an application itself attracts users to the display. Here, we expect that people find applications with higher absolute utility useful or interesting enough to lead them to interact with the display. Such applications are *attractors* for the display. Furthermore, if an application's functionality remains constant over time, we can probably attribute substantial changes in its absolute utility to its placement in the wizard.

The Study

We formulated three questions:

1. Is an application more likely to be launched from the wizard than from the application directory?
2. Does placing an application in the wizard increase its relative utility?
3. Does placing an application in the wizard increase its absolute utility?

Question 1 assesses which mechanism users prefer. Question 2 assesses how the launch mechanism affects the utility tradeoffs a user must make when choosing one application over another. Question 3 assesses how the launch mechanism affects the utility tradeoffs a user must make when choosing one application over other attractors in the physical environment.

The Hotspot

We looked at hotspots in Oulu, Finland—networked nodes that can serve many applications through their 57-inch touchscreen interface.[2] We focused on the hotspot in the lobby of a large sports facility that was available to all visitors during the facility's operating hours (see Figure 1). The facility also serves as a space for concerts, youth fairs, exhibitions, and so on. We focused on a single display to account for any environmental effects on use—keeping the environment as constant as possible makes the rest of the analysis more tractable. The hotspot has housed nearly 50 different applications

Figure 1. A hotspot at a sporting facility in Oulu, Finland. This public display employs a 57-inch touchscreen interface to provide access to many applications.

for varying periods of time. Some of these applications have been active from the beginning; some have been deployed for briefer periods for testing.

When nobody is actively using the hotspot, it functions in idle mode, in which it broadcasts commercial and noncommercial information. When the embedded Web camera, which runs face-detection software, detects someone looking at the hotspot, the hotspot enters a subtle interaction mode. In this mode, the broadcast information remains visible, but the display shows a page ear (an animation) in the upper right to entice people to approach and use the display. When someone touches the screen, the hotspot switches to fully interactive mode, in which all applications and services are available through the touchscreen. After a period of inactivity, the hotspot switches back to idle mode.

In fully interactive mode (see Figure 2), the screen is divided into the broadcast channel (the upper-left quarter), the wizard (the lower-left quarter), and the main application area (the right half). The screen's bottom part has a control bar housing the start button, which displays the application directory. Users can launch applications either directly from the wizard or by opening the application directory and touching an application icon.

Data

We looked at usage for a set of applications that were available through both the wizard and application directory. The system automatically logged all interaction events for 180 days. We divided this collected data into three 60-day periods, each having different applications available through the wizard. (These three periods weren't continuous.) As a control, we used three applications that had never been in the wizard previously. We also conducted frequent manual observations of all hotspots throughout the city during the study.

Figure 2. The hotspot interface in fully interactive mode: (a) displaying the application directory and (b) launching a map application. The screen is divided into the broadcast channel (the upper-left quarter), the wizard (the lower-left quarter), and the main application area (the right half). The screen's bottom part has a control bar housing the start button, which displays the application directory.

Table 1 summarizes the applications and their placement in the wizard and directory in each period. Each application was either in both the wizard and directory or just in the directory.

Results

Table 2 shows the results from all three periods. We hypothesize that the discrepancies in sample size were due to seasonal variations because period 2 was in the summer—when many people leave the city for vacations. On average, there were 65.4 application launches and 52.3 unique Bluetooth devices detected per day; Figure 3 illustrates the entire 180 days. The correlation of the daily application launches and Bluetooth devices detected is $r = 0.2$ ($p < 0.01$).

Choice of Launch Mechanism

Here, we focus on applications that were both in the wizard and directory and consider all three periods. On average, each application was launched 2.02 times per day from the wizard and only 0.02 times per day from the directory. A paired-samples t-test with repeated measures shows that applications featured in the wizard had significantly more launches per day from the wizard than from the application directory: $t(719) = -16.739$, $p < 0.000$.

Relative Utility

We daily sampled each application's relative utility. The relative utility increased by up to 0.12 when applications were in the wizard; at times, the increase was more than tenfold.

We ran a repeated-samples analysis of variance to test how wizard status (the application being in the wizard or not) and the application instance affected relative utility (each of the 14 applications was a separate category). An application's relative utility significantly increased when it was in the wizard ($F(1, 1{,}613) = 54.353$, $p < 0.000$). The relative utility significantly differed between applications ($F(13, 1{,}613) = 15.543$, $p < 0.000$). No significant interaction existed between an application instance and the application's presence in the wizard ($F(6, 1{,}613) = 1.203$, $p = 0.302$).

Absolute Utility

We also daily sampled each application's absolute utility. The absolute utility increased by up to 0.13 when applications were in the wizard; at times, the increase was more than 15-fold.

We also ran analysis of variance again. An application's absolute utility significantly depended on whether it was in the wizard ($F(1,1751) = 64.589$, $p < 0.000$). Absolute utility significantly differed between applications ($F(13,1751) = 51.025$, $p < 0.000$). Significant interaction existed between an application instance and the application's presence in the wizard ($F(6,1751) = 4.861$, $p < 0.000$).

Observations and Interviews

We conducted ethnographic fieldwork to capture users' interactions and experiences with hotspots and their applications. This included passive and participatory observations during public events. We conducted the observations at both the sports facility and identical hotspots (in terms of functionality) at other indoor and outdoor locations across the city.

Table 1. The applications and their placement during the study's three 60-day periods.*

Application name	Category	Placement		
		Period 1	Period 2	Period 3
Blood donation	Commercial	Directory	Directory	Wizard
BlueInfo	General information	Wizard	Directory	Directory
Clio	Digital memories	—	Wizard	Directory
Digifieds	Classified ads	—	Wizard	Directory
Etuovi.com	Commercial	Directory	Directory	Wizard
Finnkino	Commercial	—	Directory	Wizard
FunSquare	Game	—	Wizard	Directory
Hangman	Game	Directory	Directory	Directory
Fast food	Map-based	Directory	Directory	Directory
Map	Map-based	Wizard	Directory	Directory
Oulu University	Commercial	—	Directory	Wizard
RunWithUs	Exercise	—	Wizard	Directory
Street Gallery	Art	Directory	Directory	Directory
Today	News	Wizard	Directory	Directory

**Applications in the wizard were also in the directory.*

We also conducted open-ended interviews with the participants. These in situ interviews provided insights into people's attitudes toward using the wizard and application directory. Convenience appeared to play an important role in the decision to use the wizard: "It is just there—much more convenient and faster." Furthermore, the wizard's icons seemed to entice people to approach and touch the shortcut: "I saw the blood drop from over there and I wanted to know more about it, so I came over here." Finally, the wizard seemed to garner more attention if the person just wanted to kill time or check out the display: "Well, when I approach a public display but do not want to do anything specific there, I just check out the stuff that is on the shortcuts." This feedback suggests that the wizard was an important attractor. At the same time, respondents suggested that it's also a place where they look for new applications to try.

Wizards and Shortcuts

Independently of the functionality and purpose, an application in the wizard was much more likely to be launched from the wizard than from the directory. This finding is far from exciting because the wizard is meant to be a convenient shortcut for launching applications. However, wizards and one-touch shortcuts in the context of multipurpose public interactive displays raise an important question. If an application is in the wizard, how much attention does it attract away from other applications and onto itself?

Our findings indeed show that applications in the wizard were significantly more likely to attract attention away from other applications owing to their increased relative utility. This reflects prior findings in the context of online browsing,[3] in which how pages are linked affects how they're visited. Landing pages (the equivalent of our wizard) will be much more popular than pages that are one or more clicks away (the equivalent of our application directory). This is also in line with participants' comments suggesting that users trying to kill time are more likely to try the applications in the wizard.

Table 2. Usage statistics across the study.

Activity	Period		
	1	2	3
Wizard application launches	1,804	120	676
Directory application launches	2,413	232	660
Total application launches	3,443	344	1,332
Total Bluetooth devices detected	3,934	1,679	3,797

We also found that some applications were significantly more popular—that is, of higher relative utility—than others, regardless of whether they were in the wizard. Such applications are, in a sense, immune to losing launches to applications in the wizard menu. One such application was the traditional word-guessing game hangman, which consistently had a relatively high number of launches without being in the wizard.

However, applications in the wizard still had a relative advantage. This suggests that applications in the wizard are prone to cannibalize and hurt other applications, especially unpopular ones. Contrasting this finding to previous research,[3] we believe that owing to our public display's shallow menu (all applications could be launched within

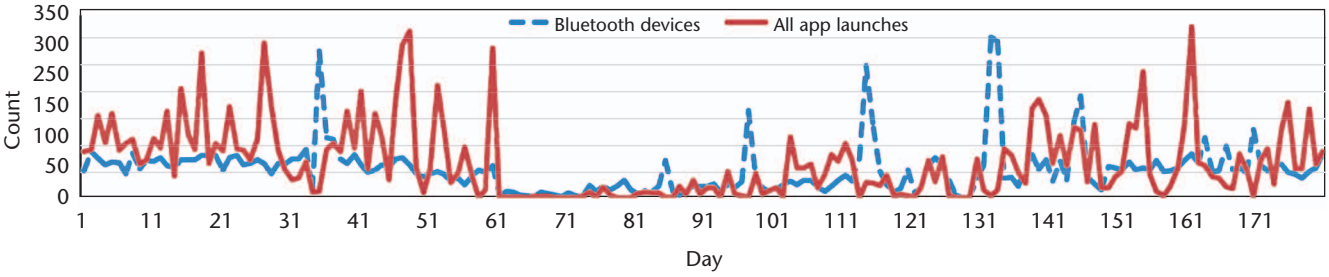

Figure 3. The number of application launches (red line) and unique Bluetooth devices sensed by the displays (blue line), per day. The correlation of the daily application launches and Bluetooth devices detected is $r = 0.2$ ($p < 0.01$).

three touches), the relative utility of applications in the directory didn't decline sharply. If directory applications were, say, five touches away, we would expect a decline of almost 90 percent in their number of launches.[3]

Next, we investigated whether wizard applications were more likely to attract people's attention—that is, to increase the applications' absolute utility. When applications were in the wizard, they indeed attracted more launches in proportion to the number of people near the display. As we mentioned before, significant interaction occurred between being in the wizard and the application instance. So, adding an application to the wizard was likely to attract more people to the application (and therefore the display), but this effect was moderated by the nature of the application itself. In other words, when some applications were added to the wizard, they became really successful at attracting attention; others didn't. Participants' comments support this interpretation, and we suspect that curiosity was an important driver owing to participants claiming that some icons in the wizard simply caught their attention.

Utility and Information Foraging on Public Displays

Because our analysis draws on the concept of utility, we can use information-foraging theory[4] to explain our findings regarding the increased use of applications in the wizard compared to those in the directory. When humans search for information, they use inherent cognitive foraging mechanisms analogous to those our early ancestors used to find food. Users of a public screen will continually decide on

- what type of information to look for (both in the display and beyond the display),
- which information path to follow,
- whether to continue seeking a suitable application at a specific information patch (such as in the wizard) or move to another information patch (such as the directory), and
- when to stop seeking.

These decisions are triggered by inherent cost-benefit analysis—the user examines the information gain against the effort to obtain it. This analysis mechanism suggests that users will leave a patch when the information gain in it drops below the gain they expect to achieve by migrating to a new patch.

Our wizard had only four applications and therefore, in principle, little foraging utility. However, because these applications were directly available to the user and therefore required near-zero effort to focus on them, the wizard's foraging value was quite high. In contrast, the directory's large number of items led to increased effort required to explore and focus on it, resulting in a relative decrease in its foraging value. So, we argue that our findings are consistent with information-foraging theory and highlight the wizard's increased foraging value.

The Effects of Physical Context

During our study, particularly sharp peaks occurred in the number of Bluetooth devices detected near the display (see Figure 3). We correlated these peaks with major events at the sports facility. For instance, the first peak in Figure 3 was on day 35, when the facility housed a high school dance. Another peak was on day 114, when the facility hosted a car show.

Although the number of devices rose sharply on those dates, use of the display plummeted. For instance, in the first peak in Figure 3, the number of detected unique devices rose to 272, but the total application launches dropped to 11. Our observations and analysis of the space suggest that this counterintuitive effect has to do with the events' nature and the display's placement. In particular, we found that people went to the facility on those days with the specific purpose of joining the event, and they were most likely in groups. People would be less likely to break away from their group and use the display while the others in their group went ahead to attend the event.

The large numbers of people during the event also showed that the display's placement wasn't appropriate for dense situations. This is particularly

Related Work in Public Displays

There's a surprising lack of empirical evidence regarding static icon shortcuts' effects on users' task efficiency, even in traditional desktop environments. Current evidence suggests that shortcuts can reduce the time to complete frequently performed tasks but that tasks for which no shortcuts exist take longer.[1]

Recent research has focused on establishing mechanisms for generating dynamic, adaptable shortcuts based on users' behavior. For example, John Tang and his colleagues studied shortcuts to recently accessed folders in a desktop environment or to recently accessed items in a shared workspace. Such shortcuts improved usage, efficiency, and satisfaction.[2] Similar research on shortcuts to frequent actions on mobile phones suggests that shortcuts that change frequently defeat their own purpose; rather, they should remain somewhat persistent.[3] Most work on such adaptive customization, however, requires monitoring and analyzing users with known identities. This requirement isn't feasible for shared public displays.

Other research has also considered shortcuts' multimodal aspects. For example, researchers have shown the benefits of providing stroke-based shortcuts, physical gestures,[4] and speech shortcuts.[5] Although this approach provides stable shortcuts that don't change over time, multimodal gestures can be challenging for infrequent users to master. This is particularly the case for walk-up-and-use shared public displays. In this case, users don't get ample opportunity to train on the system and therefore could find multimodal gestures challenging.

References

1. A. Howes, S.J. Payne, and A. Woodward, "The Trouble with Shortcuts," *CHI '00 Extended Abstracts on Human Factors in Computing Systems*, ACM, 2000, pp. 267–268.
2. J.C. Tang et al., "Recent Shortcuts: Using Recent Interactions to Support Shared Activities," *Proc. SIGCHI Conf. Human Factors in Computing Systems* (CHI 07), ACM, 2007, pp. 1263–1272.
3. R. Bridle and E. McCreath, "Inducing Shortcuts on a Mobile Phone Interface," *Proc. 11th Conf. Intelligent User Interfaces*, ACM, 2006, pp. 327–329.
4. T. Guerreiro, R. Gamboa, and J. Jorge, "Mnemonical Body Shortcuts: Improving Mobile Interaction," *Proc. 15th European Conf. Cognitive Ergonomics*, ACM, 2008, article 11.
5. T. Nakano, "Flexible Shortcuts: Designing a New Speech User Interface for Command Execution," *CHI '08 Extended Abstracts on Human Factors in Computing Systems*, ACM, 2008, pp. 2621–2624.

because people standing in front of the display to use it were likely to block the flow of people and cause discomfort to others. Moreover, research on retail has shown that flows of people can reduce the likelihood of people stopping to look at products.[5] So, we expect that this also decreased the likelihood of people using the display.

Finally, as we mentioned before, our observations suggest that some people use the displays to kill time. For example, in this location on a typical day, people would be waiting for their friends to finish their shower or for a ride home. However, on days when special events occurred, people seemed less likely to have time to kill, so they weren't likely to stop and interact with the display. It's also likely that the themed nature of the events meant that their visitors, such as avid car enthusiasts, had little interest in the display's content.

Toward Multipurpose Public Interactive Displays

Recent advances in public-display technology have enabled the increasing deployment of displays in public locations. These deployments have successfully transitioned from static broadcast displays to interactive ones. This transition, which has allowed members of the public to control the displays, has opened a range of research challenges while broadening the design space for public displays. On broadcast displays, the primary challenge is designing for effectively sharing information with the public. In contrast, interactive displays' main design requirement is interaction—the public's ability to browse, navigate, and identify information that the display makes available.

Our experience suggests that the move from single-application to multipurpose public interactive displays presents a new range of problems and a new design space. It's not clear where we can draw the distinction between single-purpose and multipurpose displays. However, one way to approach this distinction is to consider whether the display's richness is in terms of information or functionality. For instance, a display with multiple types of information about a city is a single-purpose display in that it provides multiple types of information through one interface. On the other hand, a multipurpose display provides multiple functionalities, such as information browsing, games, galleries, and polls.

Most previous research only considered supporting multiple simultaneous users on a single display. It completely overlooked the issues raised when users interact with multiple applications on a single public display. (For more on previous research in public displays, see the sidebar.)

Table 3. Research challenges for multipurpose public displays.

User interface aspect	Topic	Research agenda
Lexical (visual elements)	Presenting multiple applications	Visual organization of applications
		Application representation (icons, text, and so on)
		Notification of new or updated applications
	Interacting with multiple applications	Switching between applications
		Managing applications' state with multiple users
		Allocating screen real estate to multiple applications
Syntactic (conceptual organization)	Organizing applications	Thematic organization of applications
		Menu-based versus search-based navigation
	Promoting applications	Choosing which applications to promote on a particular display
		Identifying how application promotion affects a particular display's popularity

Tradeoffs in Multipurpose-Display Ecosystems

Inevitably, in multipurpose public interactive displays, certain applications will be more popular than others. This could be due to a number of factors, such as the quality of their design, the engagement and fun they offer, or the value they offer. This inequality leads to interesting tradeoffs that public display managers should consider when choosing which applications to promote.

As we mentioned before, our study showed that for relative utility, no significant interaction existed between the application instance and the application's presence in the wizard. This means that when unpopular applications were in the wizard, they became more popular at the expense of other applications. The same was true for already popular applications. However, for absolute utility, significant interaction occurred between the application instance and the application's presence in the wizard. This meant that unpopular applications in the wizard weren't likely to attract more people to the display. On the other hand, popular applications in the wizard were likely to attract people.

So, display managers must decide whether the wizard should feature popular or unpopular applications. Our results suggest that this decision comes down to a particular tradeoff. If an unpopular application is in the wizard, it will likely become more popular in that more users will be likely to launch it. However, fewer people will become engaged with the display overall. Promoting an unpopular application this way could mean sacrificing the display's general attractiveness. On the other hand, if the display manager promotes an already popular application by placing it in the wizard, it will likely even further hurt unpopular applications on the display, but more people will likely be attracted to the display.

In addition, previous research suggests that shortcuts shouldn't change frequently lest this confuses users.[6] This makes the decision about which applications to include in the wizard even more crucial. We believe that you could adapt existing models of human information-seeking behavior[7] and develop techniques such as automated card sorting to identify the ideal applications to include in the wizard.

A few challenges remain concerning multipurpose public displays (see Table 3). One major challenge lies in how to present a set of multiple applications to users. Because public displays generally offer very different browsing affordances compared to PCs' more traditional point-and-click interfaces, traditional multilevel menu structures are difficult to realize. We could model designs based on the world of modern mobile devices—for example, a grid of icons spread over several screens that become available through a swipe gesture. However, mobile devices are personal artifacts—users can customize the number and placement of application icons on home screens, which are thus familiar to the user. Public displays won't likely support such customization. Although researchers have proposed public-display app stores,[8] major hurdles exist to users installing their own apps, including authentication, ownership, and payment and billing.

If users can't customize a display's application grid, they might need to learn several competing layout schemes. This in turn could lead to disorientation and an overall sense of being lost. A suitable metaphor for a public-display interface might be a department store. A store window displays some products (applications) to draw customers in (as in our wizard). After people enter the store (begin interacting with the display), all the products are available in different departments (our application directory).

Application categorization, then, becomes another challenge. People employ diverse mental models to categorize objects and navigate, for example, hypertext structures.[9] Can we assume that developers as experts can build the best categorization scheme, or should, for instance, the categories be created using automated tools and textual descriptions of applications? We're comparing these two methods; our preliminary findings point toward the latter option—we believe automated categorization can work reasonably well. However, more research is required before we can draw concrete design guidelines.

Another important decision concerns the number of applications a display should have.[8] Given the challenges in presentation and categorization, can we as designers identify an optimal number of applications in terms of clarity and usability? Should the number of applications remain constant throughout various displays in the same geographic area (for example, at a city center) or vary across different contexts?

Our study has also shown an interesting competition among various applications for the users' attention. Further study on the phenomenon is required, however, before we can really understand how to foster and exploit this competition to better promote applications and entice further interaction on public displays.

References

1. E. O'Neill et al., "Instrumenting the City: Developing Methods for Observing and Understanding the Digital Cityscape," *Proc. 8th Int'l Conf. Ubiquitous Computing*, LNCS 4206, Springer, 2006, pp. 315–332.
2. T. Ojala et al., "Multipurpose Interactive Public Displays in the Wild: Three Years Later," *Computer*, vol. 45, no. 5, 2012, pp. 42–49.
3. B.A. Huberman et al., "Strong Regularities in World Wide Web Surfing," *Science*, vol. 280, no. 5360, 2003, pp. 95–97.
4. P. Pirolli and S. Card, "Information Foraging," *Psychological Rev.*, vol. 106, no. 4, 1999, pp. 643–675.
5. B. Soars, "Driving Sales through Shoppers' Sense of Sound, Sight, Smell and Touch," *Int'l J. Retail and Distribution Management*, vol. 37, no. 3, 2009, pp. 286–298.
6. R. Bridle and E. McCreath, "Inducing Shortcuts on a Mobile Phone Interface," *Proc. 11th Conf. Intelligent User Interfaces*, ACM, 2006, pp. 327–329.
7. H. Kukka et al., "This Is Not Classified: Everyday Information Seeking and Encountering in Smart Urban Spaces," *Personal and Ubiquitous Computing*, Oct. 2011; doi:10.1007/s00779-011-0474-1.
8. S. Clinch, "Designing Application Stores for Public Display Networks," *Proc. Int'l Symp. Pervasive Displays*, ACM, 2012, article 10.
9. M. Otter and H. Johnson, "Lost in Hyperspace: Metrics and Mental Models," *Interacting with Computers*, vol. 13, no. 1, 2000, pp. 1–40.

Vassilis Kostakos *is a professor of computer engineering in the University of Oulu's Department of Computer Science and Engineering. His research interests include ubiquitous computing, human-computer interaction, and social systems. Kostakos received a PhD in computer science from the University of Bath. He's a member of ACM. Contact him at vassilis@ee.oulu.fi.*

Hannu Kukka *is a postdoctoral researcher in the University of Oulu's Department of Computer Science and Engineering. His research interests include ubiquitous computing and urban informatics. Kukka received an MSc in computer science from the University of Oulu. Contact him at hannu.kukka@ee.oulu.fi.*

Jorge Goncalves *is a doctoral candidate in the University of Oulu's Department of Computer Science and Engineering. His research interests include social and ubiquitous computing, as well as the use of technology to motivate participation. Goncalves received an MSc in computer science from the University of Madeira. Contact him at jgoncalv@ee.oulu.fi.*

Nikolaos Tselios *is an assistant professor of computer science in education in the Department of Educational Sciences and Early Childhood Education at the University of Patras. His research interests include educational technology, human-computer interaction, and Web science. Tselios received a PhD in electrical and computer engineering from the University of Patras. He's a member of ACM, the Technical Chamber of Greece, and the Hellenic Society for the Application of ICT in Greek Education. Contact him at nitse@ece.upatras.gr.*

Timo Ojala *is a professor of computer science and engineering in the University of Oulu's Department of Computer Science and Engineering. His research interests include ubiquitous computing and urban informatics. Ojala received a DrTech in electrical engineering from the University of Oulu. He's a member of IEEE, ACM, and the Pattern Recognition Society of Finland. Contact him at timo.ojala@ee.oulu.fi.*