

Postprint Version	1.0
Journal website	
Pubmed link	
DOI	

This is a NIVEL certified Post Print, more info at <http://www.nivel.eu>

# Early User Involvement and Participation in Employee Self-service Application Deployment. Theory and Evidence from Four Dutch Governmental Cases

GERWIN KOOPMAN RONALD BATENBURG

## ABSTRACT.

This paper theoretically and empirically addresses the notion that user participation and involvement is one of the important factors for IS success. Different models and studies are reviewed to define and classify types of early end-user involvement and participation. Next, five case studies are presented of Dutch governmental organizations (Ministries) that have recently deployed an employee self-service application. Based on interviews with developers, project managers and users it can be showed that the deployment success of such systems is positively related to the extend of early user involvement and participation. In addition, it was found that expectancy management is important to keep users informed about certain deployment decisions. In this way, employees can truly use the self-service applications without much support from the HR-departments.

## INTRODUCTION

In 2007, the Dutch House of Representatives asked the Dutch Government questions about their ICT-expenditures. Concerns were raised about how much money was wasted by governmental ICT projects that resulted in failures. The Dutch Court of Audit was instructed to come up with a report on governmental ICT projects and the possible reasons for failures. When the report (Dutch Court Audit, 2007) was finished, it named several difficulties that can be faced when executing ICT projects for governmental organisations. Among this list is the impact of the changes caused by the implementation of the IT-system. Users' current way of working may be completely changed by changing work processes when the system is introduced. Users therefore need to be informed and trained to completely benefit from the system. Another cause for problems is the need for clear goals and demands. If the software developer does not receive clear demands and wishes, the actual end-product might not be what the government thought it would receive.

In both of the mentioned problems users play an important role in making the system a success. There is already a lot of agreement on the fact that users should be involved to produce usable software programs. It is recommended in ISO standard 13407 to get better insights in the requirements for a software application. Most attention to user involvement is still on the usability testing of systems, which happens on a later stage in the development process. However, the sooner the end-user is involved, the more efficient it is (Noyes et al, 1996; Chatzoglou & Macaulay, 1996; Blackburn et al, 2000).

One of the challenges in involving users in IT developments is the time factor that plays a very important role in governmental IT projects. Most of the decisions to implement or develop new Information Technology have a political background. This means the project will have to be delivered at the end of the current cabinet's term. This introduces a certain pressure for the project to be delivered as soon as possible. This conflicts with the idea that user involvement will take a serious amount of extra time needed in the

development of a new system (Grudin, 1991). The systems that have the specific attention of this research on first sight also seem to conflict with this additional time needed in IT projects when involving users. Main reasons for implementing E-HRM systems and Shared Service Centres (SSC) are increasing efficiency and productivity (Verheijen, 2007; Janssen & Joha, 2006). For the current Dutch cabinet this is very important because it wants to decrease the number of civil servants with a number of 12,800 to achieve a cost cutback of 630 million Euros in four years (Ministerie van Binnenlandse Zaken en Koninkrijksrelaties, 2007).

Another difficulty in the involvement of users is the selection of the right groups of employees to have participating in the project (Grudin, 1991). This is especially true for most governmental organisations, because they employ a large amount of civil servants. As the applications that are the subject of the research are mainly aimed at self-service these are all potential end-users. This is a very diverse group and it can be considered a challenge to make the right selection of users from this total population.

In this paper we address the question which methods are currently used within the Dutch governmental institutions to involve end-users when deploying employee self-service applications. We also investigate the relationship between end-user participation and involvement and the success of such e-HRM applications. Four Dutch governmental organizations (Ministries) are investigated that have implementing employee self-service applications (i.e. e-HRM), which offers the possibility to compare the different development approaches that are followed. Semi-structured and topic interviews were held with stakeholders within the Ministries to explore which methods are already used to involve users in the process of deployment. Their experiences are described and reflected upon at the closing section of the paper.

### **Some Theory: A Review on the Role of User Participation**

DeLone & McLean (2003) evaluated empirical testing and validation of their original D&M IS Success Model (DeLone & McLean, 1992) by other researchers and developed an updated IS Success Model visible in Figure 1. It is based on the three different levels of communication that were already defined by Shannon & Weaver (1949). The *technical* level is concerned with the physical information output of a system, and evaluated on accuracy and efficiency. The *semantic* level deals with the meaning of the output of a system, and specifically with how well the output conveys the intended meaning. The *effectiveness* level or *influence* level (Mason, 1978) concerns the effect of the information on the receiver.

#### **[FIGURE 1]**

On the left-hand side of the model there are three different quality dimensions. The upper two were already present in the original model. Terms to measure *information quality* were accuracy, timeliness, completeness, relevance and consistency. This concept and the terms thus measure success on the semantic level. For *system quality* these terms were ease-of-use, functionality, reliability, flexibility, data quality, portability, integration and performance. These measures are related to the technical success level. The concept of service quality was added, because of the changing role of IS organisations, they more and more deliver services instead of only products.

The *use* concept in the model measures success on the effectiveness level. It is considered an important indicator for system success, especially when it is “informed and effective” (DeLone & McLean, 2003). There are however some difficulties in the interpretation of the concept *use*. It has a lot of different aspects, for instance voluntariness and effectiveness. It therefore received critique in research of the original model. One example is that use is not a success variable in the case of mandatory systems. DeLone & McLean (2003) argue however, that no use is “totally mandatory” and that mandatory systems will be discontinued if they do not deliver the expected results. The differences are thus on the different levels: mandatory to use for employees, but voluntarily to terminate for management. Nevertheless, to overcome some of the difficulties, in the updated model a distinction is made between *intention to use* (attitude) and *use* (behaviour).

*User satisfaction* is considered a useful measure in evaluating the success of information systems (Ives, Olson, & Baroudi, 1983). In most cases this subjective judgment by users is considered to be more user practical, because of the difficulties in measuring the success of IS objectively (Saarinen, 1996; Lin & Shao, 2000). The *net benefits* concept is regarded as a set of IS impact measures, for instance work group impacts and organisational impacts. More specific measures are for instance quality of work, job performance and quality of work environment. To reduce the complexity of the model these are all grouped

together, and which impact will be chosen will depend on the information system that is evaluated. These two concepts (user satisfaction and net benefits) measure success on the effectiveness level.

Lin & Shao (2000) investigated the relationship between user participation and system success. They found a significant relationship between both concepts, but warn that the context should be taken into account. Both user participation and system success can be directly and indirectly influenced by other factors. Based on the outcomes of their data analysis they also suggest that “getting users involved in the development process may improve their attitudes toward the system and enhance the importance and relevance users perceive about the system”. Other findings were the positive influence of user attitudes on user involvement and the fact that users are asked more to participate when the complexity of systems increases.

In a survey of 200 production managers Baroudi et al, (1986) found positive correlations between user involvement and user information satisfaction and system usage. User involvement in this case was conceptualised as activities during the development that enabled the users to influence the development. Although, this is more towards user participation, it does not completely distinct the behaviour and psychological concepts. Interviewing users and developers from 151 projects, McKeen & Guimaraes (1997) found a positive and significant relationship between user participation and user satisfaction. They also noted that projects concerning systems or tasks with a high complexity called for more user participation.

To positively influence the success of new software applications, developers often turn to user involvement. According to Kujala (2003) user involvement can be seen as a general term and refers to Damodaran (1996) who suggests a continuum from informative to consultative to participative. However, some researchers suggest a difference between user involvement and user participation. User participation will then be the “assignments, activities, and behaviors that users or their representatives perform during the systems development process” (Barki & Hartwick, 1989). User involvement can be regarded as “a subjective psychological state reflecting the importance and personal relevance that a user attaches to a given system” (Barki & Hartwick, 1989). McKeen et al, (1994) in their research on contingency factors also describe the development of the division of these two concepts. The refined model is even further extended, in that user the relationship between user participation and system success is influenced by other moderating variables like the *degree of influence, attitude, communication* and *type of involvement*. In this study therefore the distinction between the two concepts *user participation* and *user involvement* will also be used.

Several researchers do not only recognize the link between user participation and system success, but even stress the importance of involving end-users in the development of software. Kensing & Blomberg (1998) state the participation of end-users is “seen as one of the preconditions for good design” (see also Shapiro, 2005). The workers have the information about the working environment and organisation, which designers logically do not always possess. Combining the domain knowledge of the workers and the technical knowledge of the designers is considered a foundation for the development of a useful application (Kensing & Blomberg, 1998).

Reviewing several studies on user involvement Damodaran (1996) identified a number of benefits of participating end-users:

- *More accurate user requirements*

Numerous problems or defects in software applications can be traced back to poorly capturing requirements at the beginning of the development process (Borland, 2006). Pekkola et al. (2006) also argue one of the reasons for information system development projects to fail are incomplete requirements. In their studies they found user participation useful in gathering “credible, trustworthy and realistic descriptions of requirements”. In turn these accurate user requirements result in an improved system quality (Kujala, 2003).

- *Avoidance of unnecessary or unusable costly system features*

Two of the usability guidelines given by Nielsen (1993) are “Designers are not users” and “Less is more”. Designers might think of certain features to incorporate in the application without consulting end-users. Functionalities that are completely logical for developers might be completely incomprehensible for users. This might result in users having to spend too much time on learning how to use these functionalities or

even not using them. Designers might also have the tendency of incorporating too many options to satisfy 'every' end-user. Besides the fact that users might never know of these options and use them, they can also work contra productive by overwhelming to users. A lot of time and effort for developing these features can be saved by participating end-users.

- *Improved levels of system acceptance*

The levels of system acceptance can be positively influenced by user involvement in several ways. Among the list Ives & Olsen (1984) have found in other literature is for instance the development of realistic expectations about the information system (Gibson, 1977). Also decreasing user resistance against the new system and actually creating commitment for it are other results of user participation (Lucas, 1974). Cherry & Macredie (1999) state participatory design as a means to overcome the acceptability problems systems might encounter without the participation of users.

- *Better understanding of the system by the user*

Logically during the participation users will learn about the system by experiencing the development (Lucas, 1974). This familiarisation also leads to the increase in chances users will come up with suggestions during the development, because they will feel more confident (Robey & Farrow, 1982). In the end this greater understanding should lead to a more effective use of the application.

- *Increased participation in decision-making within the organisation*

Clement & Van den Besselaar (1993) point to the fact that participation is not only restricted to the design of an IT-system. The application will probably change the way tasks are executed, thus affecting the entire organisation and by participating employees have the possibility to influence this (Robey & Farrow, 1982). It might thus not be restricted to the design of the application, but also to other decision-making processes within the organisation.

Although involving users is considered to be useful, it also introduces a number of difficulties. Firstly, a large amount of a user's knowledge about the process or task the software application will have to support has become tacit (Wood, 1997). It might therefore be hard to get information from these users about the way they work. An example of this was also visible at the Ministry of the Interior. Developers built a certain functionality based on the description of how a task was executed by employees without an application. After implementation however, it became clear in the former way of working an extra file was created by users to keep track of the status of the tasks at hand. Since this was not formally part of the process, they forgot to mention this to the developers. It was thus not incorporated in the new application, while this would have been relatively easy to realise. To overcome this kind of problems it is possible to perform field studies, which have the advantage that users do not have to articulate their needs (Kujala, 2003). Other researchers also suggest the use of (paper) prototypes to counter the difficulties users might have in articulating their needs (Pekkol et al, 2006; Nielsen, 1993).

Users can also be reluctant to have developers observing them while they work (Butler, 1996). They might express concerns about justifying the time they would have to spend with the design team or disturbing their co-workers. Solutions to this problem are getting commitment from management (Grudin, 1991) and having sessions in separate rooms so no colleagues would have to feel bothered. Besides these problems Butler (1996) mentions the fact that these sessions are considered to consume a lot of time, as well in planning them, as in executing them. Several researchers also point out the fact that involving users most of the time delivers a large amount of raw data that is difficult to analyse and to use in decision making (Brown, 1996; Rowley, 1996). This will make projects where users participate more time-consuming and thus something development teams want to cut back on. Grudin (1991) also noted the judgement of developers that user involvement would take too much time. However, as already stated, allocating more time upfront will result in a faster cycle time for software developers (Blackburn et al, 2000).

Some members of the design team might simply not have the abilities needed to communicate efficiently with users (Grudin, 1991). They might find it difficult to understand the work situations of users or miss the empathy needed when communicating with users that do not possess the computer skills they have themselves. As a solution to the problematic communication between users and developers, mediators could be brought into action (Pekkola, Niina, & Pasi, 2006). They can act as a bridge between both groups,

translating the different concepts from one group to the other. Mock-ups and prototypes from the design team are for instance discussed with users, while user input and feedback is given to the design team. Developers can then focus on the design and implementation of the application instead of having to spend time and effort on user participation methods.

A challenge that occurs even before all of these mentioned is the selection of user representatives and obtaining access to them (Grudin, 1991). Even when an application is developed specifically for one organisation, developers might fear the risk of missing a certain user(group) in their selection. A possible solution is to define a few personas based on intended users. A persona is defined as “an archetype of a user that is given a name and a face, and it is carefully described in terms of needs, goals and tasks” (Blomquist & Arvola, 2002). This can be useful in organisations that have large groups of users, which makes it tricky to randomly take a small selection out of the total group. Subsequently getting hold of the ‘selected’ end-users might also pose some difficulties. There might be several barriers like information managers acting as user representatives, but who do not resemble the actual end-user. Also the physical distance between developers and users might create problems. One of the solutions is to, if possible, have the development team working on location of the customer. This way easy access to users is possible (planned or ad hoc).

### **Early Involvement and Participation of Users**

The reasons of having user participation are clearly visible, but when should end-users engaged in the development process? Several researchers suggest that users should be involved early in the process. For instance, if users are used as sources in the requirements capturing process, the number of iterations is less than if they are not (Chatzoglou & Macaulay, 1996). Also capturing usability problems early in the process is very rewarding. Mantei & Teorey (1988) estimate that correcting problems early in the development process cost three times less than correcting them later on. Nielsen (1993) also supports the involvement of users just after the start of the design phase. Regular meetings between users and designers could for instance prevent a mismatch the users’ actual task and the developers’ model of the task.

In comparing software development firms Blackburn et al. (2000) found that the ones that were considered to have a faster cycle time, were the ones that spent more time on for instance getting customer requirements at the early stages of the project. In their follow-up interviews of their quantitative data analysis, managers mentioned that much time in projects is consumed by rework. To reduce this time it is important to capture the needs of the users early in the development, so before the actual programming has started. In the end this will actually improve the speed and productivity of the software developer.

Damodaran (1996) underlines the justification of early user involvement by pointing to one of the principles of a number of social design approaches. That is, organisations will just postpone the detection of problems if there is no effective user involvement. Again, problems that have to be solved later on in the development, or even after implementation, will result in higher costs.

User participation can take on a number of forms in the development of a software product. Kujala (2003) suggests four main approaches are detectable, which are *user-centred design*, *participatory design*, *ethnography* and *contextual design*. Since involving end-users from the beginning of the project is considered very beneficiary, the focus will be on those approaches and methods that take place early in the development process.

Gould & Lewis (1985) in their research on *user-centred design* recommend the early focus on users and direct contact between development team and end-users. This implies doing interviews and discussions with end-users, even before any design has been made. Also people should be observed when performing tasks, as well in the present situation as with prototypes that are developed during the project. Also the design should be iterative, this could for instance be realised by using prototypes that can be reviewed by users.

*Participatory design* is considered to be a design philosophy instead of a methodology (Cherry & Macredie, 1999). It is not prescriptive and therefore the set of techniques that could be used should be considered open-ended. The approach does have some identifiable principles however, firstly it aims at the production of information systems that improve the work environment. Secondly, users should be actively involved at each stage of the development and finally the development should be under constant review (iterative design). Cherry & Macredie (1999) also mention four important techniques, cooperative prototyping being the main technique. The other techniques are brainstorming, workshops and organisational gaming.

*Ethnography* consists of observing and describing the activities of a group, in an attempt to understand these activities (Littlejohn, 2002). In the design of information systems it is defined as developing “a thorough understanding of current work practices as a basis for the design of computer support” (Simonsen & Kensing, 1997). The reason for this is the occurrence of differences in what users say they do, and what they actually do (Nielsen, 1993). The approach is descriptive of nature, is from a member’s point-of-view, takes place in natural settings and behaviours should be explained from their context (Blomberg et al, 1993). A typical method of ethnography is observing end-users while they perform their daily work. This can be done following them in their work, so designers being present at the office, or recording the tasks on video and then analysing this footage later on.

Similar to ethnography is *contextual design*. Its goal is to help a cross-functional team to agree on what users need and design a system for them (Beyer & Holtzblatt, 1999). The approach focuses on the improvement of the current way of working within an organisation. It thus is not only limited to the design of a system, but also incorporates redesigning the work processes. Users are the main source for data to support decisions on what developments should take place. Specific methods to obtain information from users are (paper) prototyping and contextual inquiry. The latter method is a combination of observing users and interviewing them at the same moment (Beyer & Holtzblatt, 1999).

Co-development, ethnographic methods and contextual inquiry are participatory methods that are located early in the development cycle (Muller, 2001). Most of the approaches actually span the entire development. Table 1 summarises this section and lists the techniques that could be used in the early stages of the development.

#### [TABLE 1]

#### THE CASE: EMPLOYEE SELF-SERVICE APPLICATIONS

As stated in the previous section the type of system and the contextual environment are important factors to keep in mind when measuring IS success. In this paper we focus on Employee Self-Service (ESS) systems that represent one of the fast developing trends in the domain of e-HRM (Strohmeier, 2007; Ruël et al, 2004). This type of systems is specifically relevant for this study as it directly relates to the issue of user participation, as it aims to empower employees within organizations.

ESS is defined by Konradt et al. (2006) as “corporate web portal that enables managers and employees to view, create and maintain relevant personnel information”. Konradt et al. also identify four different basis channel functions the ESS can support:

- informing employees about rules and regulations
- providing interaction the access to personal information
- supporting transactions, like applications for leave
- delivering for instance payslips or training videos

All of the above tasks are normally done by the organisations’ HR departments. Fister Gale (2003), in her study on three successful ESS implementations, describes reducing the workload of these personnel departments is a major reason for implementation. For instance, changing personal information of employees in often several databases normally had to be done by HR employees. This can now be done by employees themselves by filling in web based forms, resulting in (real-time) updates of the databases of the HR systems. The web based nature of the ESS also offer the possibility to significantly decrease the paperwork that needs to be handled. However, the benefits are not only on the organisations’ side, employees also profit from the implementation of ESS. They have instant access to information and the effort needed for certain transactions, like expense claims, is reduced. Managers also benefit from the up-to-date information and easy access to for instance reports, resulting in a better overview over their resources.

#### ESS and User Satisfaction

Konradt et al. (2006) used the well-known Technology Acceptance Model (TAM; Davis, 1989) to describe the influences of a systems’ usefulness and ease of use on user satisfaction and system use. Ease of use related positively to user satisfaction, as well as to usefulness. Usefulness in turn positively influenced both system use and user satisfaction. A final relationship was described between user satisfaction and system use.

A number of implications were drawn from these findings to ensure the success of an ESS implementation. The suggestion that system acceptance is mainly determined by the usefulness of the system and its ease of use, implies that enough attention should be paid to these factors. Informing and involving employees during the development is advised to influence the ease of use and usefulness of the application. It should be clear to employees why it is beneficiary to them to use the ESS, to ensure system acceptance. If users do not accept the system, the workload reduction for HR department will not be realised. Instead of the normal workload, HR employees will be flooded with help requests by users who do not understand the system or even are reluctant to work with it.

### **Data and methods**

To determine in what ways users are involved or enabled to participate in the development of software applications, interviews were held at four Dutch ministries. The cases are described below.

#### *Emplaza at the Ministry of the Interior and Kingdom Relationships*

The person interviewed representing the Ministry of the Interior and Kingdom Relationships is the project leader Self-Service / Emplaza. The application is called Emplaza, a combination of the words *Employability* and *Plaza*.

This self-service human resources application is used by approximately 5,500 civil servants within the Ministry of the Interior and Kingdom Relationships. The application is also used by the Ministry of Agriculture, Nature and Food Quality and the Ministry of Economic Affairs, resulting in a total number of about 17,000 users. The software supports up to twenty HR-processes, for instance applying for leave or filing an appraisal conversation. The application is actually a sort of web application and functions as layer over the actual administrative IT-system. It is built and managed by an external party.

At the time of the interview a new release of the application (version 4.3) was under development. This will be the base for this case description. Since the application is not entirely new some of the reactions of the users can be expected based on experiences from the previous releases. These experiences also influenced the way in which new releases or features are developed. This time however, the release has taken more than a year to develop because of some important differences with previous situations. First of all the builders were new to the project and therefore the advantage of having worked together (as with previous releases) was lost. Second, release 4.3 can be considered larger and more extended in words of number of functionalities. As a result testing the application a considerable amount of extra time was needed to test this version. Finally, the change in organizational structure with the introduction of P-Direkt (see next section) also took some time to get used to. P-Direkt for instance now takes care of communication between the external builder and the user group of the Ministry.

For the development of the new releases key-users or super-users were selected to participate. These civil servants have a lot of knowledge about the process the application is supposed to support. By interviewing them they current way the process is executed is determined. A next step was to establish which forms should be available to support task within the process. After that the next task was to find out how the forms and workflow should look like in Emplaza. When agreement was found on these issues the Functional Designs were created by the software developer. Before the actual programming started, a number of applications that supported similar HR-processes were investigated. Findings from this analysis formed the starting point of how this should be realized in the Emplaza application.

The key-users are thus very involved in the business rules that need to be implemented in the system. Other aspects they are asked to judge, are the look-and-feel of the user interface and the performance of the application. To do this they have to use test-scripts that will force them through every step and part of the new functionality so they will be able to comment on all the new developments. Members of the HR self-service project team also test the application by looking at it from the viewpoint of a 'new' user. They specifically pay attention to the help texts that are created for the end-users to guide them through certain tasks.

A number of criteria were used in selecting employees to participate in the development of the new release. Participants had to have a lot of knowledge and experience in the field concerning the process at hand. Furthermore, they had to be available to cooperate, i.e. they had to be freed from their normal tasks. Finally, they also had to be able to think constructively about the new functionality. Most of the time it had become clear in earlier sessions whether or not people met this latter criterion. For testing the application managers are asked to cooperate, they are selected on their position within the organization and thus all

have a different role in the HR-process that is going to be implemented in the system. It is tried to have two 'camps': those who are sceptical of ICT and those who feel positive about ICT.

End-users are actually involved just when the new release has gone 'live' i.e. has gone into production. Complaints and issues that come up during the use of the application are gathered and reviewed. These form the foundation for the change proposals that are discussed on an interdepartmental level. During these discussions decisions are taken on which changes really need to be implemented. If end-users are involved earlier after all, most of them come from the central apparatus of the Ministry. The reason for this is that they are located close to the test location. They are chosen as randomly as possible, so no real criteria are used to select participants. This way the development group hopes to get 'fresh' insights about the application.

#### *P-Direkt to be used throughout several Dutch Ministries*

In July 2003 the Dutch cabinet chose to start the establishment of a Shared Service Centre (SSC) called P-Direkt. This should be a Human Resource Management SSC for Personnel registration and Salary administration. Although the project had some major problems, it is now still in progress with the same main goal. It should lead to a more efficient HR-column of the government (Ministry of Internal Affairs, 2006). Two identified conditions to reach this goal are joining administrative HR-tasks and the implementation of digital self-service. The latter of these recognised conditions makes P-Direkt an interesting subject to examine and see how user involvement or participation are applied in this project.

The respondent for this interview is the Test manager at P-Direkt and is responsible for the Functional Acceptation Tests and User Acceptation Tests of the HR-Portal that is currently developed. This self-service HR-portal should eventually be used throughout the entire government. In contrast to the Emplaza 4.3 release this is an entirely new application. It is built using mostly standard functionalities of SAP but, if necessary, customization is also applied.

In the process of developing the self-service application users are involved in different ways and at different stages. Right from the start several workgroups are formed. These consist of civil servants from several ministries that in the end will use the application. The members of these groups could be considered end-users, as they will eventually use the application in their normal work. However, they have a lot of knowledge about the HR-processes the application should support. These workgroups are full-time dedicated in the development of the application for a longer period of time. One workgroup for instance has been involved from the start in simplifying and standardizing the HR-processes within the Dutch Government. After twenty-four processes had been defined they formed the basis to build the technical system that should support them. The workgroups were then involved in incorporating the right business rules within this system. An example of such a rule is calculating the maximum compensation that should be granted in different situations.

At the final part of building the application a number of end-users are asked to test the application. This group of end-users do have knowledge about the processes that should be supported, however they were not earlier involved in the development of the application. The involved departments are asked to send one or two employees to take part in the User Acceptation Tests. Per session seven to ten participants are asked to complete the scenarios that are designed to guide them through a certain task. These tasks, for instance filing an expense claim, are subdivided in different steps. This way users cannot only comment on the application in general but also notate findings about specific steps in the process. In this way the scenarios also contribute to being able to easily group comments about certain steps in the process or specific parts from all the different test users. These grouped and summarized comments and findings are then discussed by P-Direkt and the builder of the application. During these discussions it is decided which findings need fixes and those are then built within two to three days. After that a new test session is held to examine whether or not the problems were sufficiently solved.

#### *P-Loket at the Ministry Of Health, Welfare and Sport*

The Ministry of Health, Welfare and Sport uses an application which is very similar to Emplaza. It is called *P-Loket* and was also developed for the Ministry of Social Affairs & Employment and the General Intelligence & Security Service, because they use the same payroll application. *P-Loket* is a web application, and functions as a layer on top of this payroll application (PersonnelView, or P-View). This situation is thus very comparable to the one at the Ministry of the Interior.



*P-Loket* is a totally new developed application, which can be used by employees to support them in (personnel) tasks like for instance the filing of a request for leave. In June 2007 around 12 different forms are supported by the application, which can be used by approximately 2,250 civil servants. These numbers should grow to about 18 forms and 5,000 employees by January 2008. The forms and processes that should be supported were chosen based on the outcomes of the standardisation workgroup of the P-Direkt project. The P-Direkt project is also the reason that after the 18 forms are finished no further developments will be done. The P-Direkt application will eventually substitute *P-Loket*.

From the start of the development of the application (in 2006) it was already clear P-Direkt would be the governmental HR self-service application. However, for reasons of more rapidly realisable efficiency benefits and to get used to self-service applications, it was decided to still start the development of *P-Loket*. The first quarter of 2006 was used as preparation and to come up with a plan of how to approach the project. The second quarter of the year was used to prepare for the building, make a process design and setting up authorisations. By the end of June the actual creation of the application could start.

Building the application was done by an external software company that also created the P-View application. *P-Loket* was also a totally new developed application for them. However, P-View and P-Loket are quite similar web applications, which had some advantages. The links for instance that had to be available, were already more or less present and thus had not to be created completely from scratch. They had one developer working full-time on the project.

Employees of the Ministry were involved in several ways during the development. The project group that was formed at the start consisted of hr-employees, members of the audit service and two employees of the Information & Communication department. The latter two were experts on web (applications) and usability. Both these experts had the task to look at the application from a user perspective. The usability expert for instance discussed a number of prototypes (on screen) with the builder. By asking questions like “what will happen if a user clicks this button?” issues could already be addressed before anything was programmed. Apart from the experts the project group members did not attend to the User Interface of the application. Their focal point was on the business rules that should be implemented.

Next to the fact that users were represented in the project group, other civil servants were asked to cooperate in an usability test. This test was carried out by a third party and the main reason was to resolve usability issues the software builder and the usability expert from the project could not agree on. The test was carried out with one test person and a guide in one room, observers were in another room to take notes and film the session with a camera. In selecting employees to take part in this test, the project group tried to have a balance in computer skills, male/female ratio and office/field staff ratio. To find eight participants, contact persons were asked if they knew employees that fitted the necessary characteristics.

Another way to involve end-users was to have sessions with managers to discuss the functionality that supports performance review conversations with them. Per session the application was demonstrated to three to twelve managers. It took roughly five weeks to complete the sessions with two hundred and fifty managers. Managers could immediately deliver feedback in the form of questions or remarks during the demonstration. Although this way of involving end-users took considerable time and effort, it was considered to be very useful and contributing to the acceptance of the application. One of the strengths of having different sessions was that certain issues came up in numerous occasions. This made it easier to establish the importance of a problem or request. The issues from the different sessions were combined and for each issue the urgency was determined. Subsequently the impact of solutions for these issues was discussed with software developer.

Also the fact that the project group was located in the same offices as end-users that were not in the project group offered the possibility to ask these colleagues for their opinions in an informal ad hoc way. The project group gratefully made use of this opportunity during the development of P-Loket.

#### *PeopleSoft/HR at the Ministry Of Defence*

The Ministry of Defence started implementing self-service on HR-processes in 2004, but without involving end-users. As a result the users started having wrong interpretations about the application. Therefore the Ministry started with improving the self-service parts of the application in 2006. The application is based on the PeopleSoft HR-system and the first processes to be supported were looking into personal data, filing requests for leave and filing requests for foreign official tours. Approximately 80,000 users make use of the software, of which about 65,000 are permanent staff of the Ministry. Besides this

large group of users another point of consideration is the sometimes disrupted relationship with the formal superior. This is due to frequent shifts within the organisation, for instance staff being posted abroad for military operations. As a result the application should offer the possibility to delegate certain tasks to other superiors, planners and/or secretaries.

It took about one and a half years from the beginning of the project until the improved application went live. As a start it was determined which people and processes should be supported. Subsequently the possibilities of the (then) current application were investigated. There were three important points of departure:

- outcomes of using should be visible to the user
- employees use the application in good faith (“the user does nothing wrong”)
- no training should be necessary to use the application.

The main idea behind this is that the development should not only be seen as supporting a process by an application, but also supporting users in their actions when using the application. Usability research was done by someone from outside the Ministry who had no knowledge of HR-processes or PeopleSoft. This person asked the civil servants how the current application was used. Some consultancy was done by external parties, but it felt most of the work was done by the internal organisation to come up with advises and reports.

Besides the usability research, users were also involved in other ways. Employees with reasonable knowledge and skills about IT were asked to name functional gaps in how the support of certain processes. Next to that case studies were done by randomly asking people in the organisation to perform tasks with the application. They only got a short introduction of the task and the reassurance they could do nothing wrong. So nothing about how the application worked was explained. After this users were observed completing the tasks, while they were invited to think-aloud. The moments when users hesitated or were in doubt, were explained as moments in the process the application should offer help. The outcomes of this test were thus:

- Information on how the application was used
- Whether or not concepts and descriptions were interpreted as intended
- Functional problems
- Insights in perceptions on what has happened by performing this task (“what will be the next steps in the organisation?”)

These outcomes findings were incorporated in the improved version of the software. It would be valuable to perform such a test again now the build is complete, however at the moment there is no time available to do this.

Demands for support and help options are not the same for every user, for instance because of the mentioned differences in IT-skills. One of the tools for help within the PeopleSoft application is the “See, Try, Know, Do” principle. Users can first look at a demonstration (see) before trying it themselves in a simulation mode (try). A next step is then to take a test to check if they understand everything (know), before finally actually performing the task with the application (do). Users can use one or more of these functions to support them in the use of the software.

In choosing people for the tests information managers were asked if they could point out employees that met certain criteria. One criterion for instance was whether or not they were very skilled in using IT. Although there were some criteria, no standard profiles were used to categorise users in groups. To confront these users with the improved application, during the development, prototypes were used.

### **Cross-case Comparison Analysis**

The four case studies presented in the preceding sections have several things in common. First of all, they are of course all developments for a governmental organisation. Secondly, they clearly all support self-service on Human Resource processes. Thirdly, they all serve a large number of users, the HR self-service applications have a range from 5,000 to 80,000. The fourth parallel is that in all cases external organisations were hired to assist in the development, although in the case of the Ministry of Defence this was mostly limited to advisory reports.

**THE PREVIOUS SECTIONS ALSO SHOW THAT USERS ARE INVOLVED OR ARE ABLE TO PARTICIPATE IN THE DEVELOPMENT OF IT-SYSTEMS IN THE DUTCH GOVERNMENT. [TABLE 2]**

depicts the different ways users participated in the different cases that were discussed.

**[TABLE 2]**

It is clearly visible that users get most attention during the test phase of the development. In all of the discussed applications end-users have participated in one or more tests during the development. Logically, this testing found place in later phases, however in three cases prototypes were used to be able to show users (parts of) the application earlier in the process of development. Non-expert end-users that participated from the start of the projects were visible in two cases (P-Loket at the Ministry Of Health Welfare and Sport and PeopleSoft/HR at the Ministry Of Defence). Most of the users that were engaged from the start had expert knowledge on the processes that were computerised by the implementation of the application.

Of particular interest is how the participation of end-users might have influenced the success of the applications in question. Figure 2 has the different applications from the interviews depicted in a diagram that scores them on success (y-axis) and user participation (x-axis). Their position on both axes is based on the interviews, but is of course subjectively determined. It is not meant to imply that any of the applications is 'better' than the others. The concept of 'application success' is seen from the viewpoint of each organisation and is based on:

- Time and effort needed for development (relative to the amount of features)
- Number of problems encountered during tests and implementation
- Satisfaction with the end-product
- Contribution to increase in efficiency of the supported tasks

As a final step to complete the four case studies among the Dutch Ministries, additional information about the (perceived) success of the ESS-applications was collected. This was conducted through one short personal e-mail sent to the interviewees, containing six system quality criteria for which an answer on a 5point scale was requested:

1. The number of problems reported by users in the user (acceptation) tests (ranging from 1 being "very few" to 5 being "seriously many")
2. The amount of rework needed after testing (ranging from 1 being "very little" to 5 being "very much")
3. Current satisfaction with the application by end-users (ranging from 1 being "very dissatisfied" to 5 being "very satisfied")
4. The amount of questions that reached the helpdesk shortly after implementation (ranging from 1 being "very few" to 5 being "seriously many")
5. Contribution of the application to the increase in satisfaction (ranging from 1 being "very little" to 5 being "very much")
6. The overall success of the application (ranging from 1 being "very low" to 5 being "very high")

Table 2 shows the answers of the respondents that were queried for four different Ministries/cases.

**[TABLE 2]**

Based on these additional data, the actual relation between early user involvement, satisfaction and application success can be estimated for the four cases under investigation. Before we do so, some remarks beforehand.

*First*, it should be noted that for Emplaza both the latest version (4.3) and earlier versions were investigated (E1). During the interviews it became clear that in earlier versions a lot more user participation was applied. The short communication lines resulting from the programming team working on the same location as users, made sure users could be frequently consulted. At the deployment of Emplaza 4.3 end-users were involved, but not at all before a working product was developed. A number of problems arose in the development and testing of the 4.3 version. Some functionalities could for instance not be implemented on time, because tests by key users revealed too many hick-ups. User satisfaction with the functionalities that could be implemented on time however, was considered to be reasonable high. Also, it would not be fair to let the low score on participation of end-users to be the only reason for the low score on application success. There were numerous other problems mentioned that contributed to the difficult development of Emplaza 4.3, as described in section 3.2.1. Before these problems are solved however, earlier versions can be considered relatively more successful than the latest one.

*Secondly*, it needs to be considered that the PeopleSoft/HR application also has earlier versions as ESS within the Ministry of Defence. In the use of the initial version users encountered too many problems. So, the second version was developed to be an improvement of the first version. A lot more attention for usability went hand in hand with the increasing possibilities for users to participate in the development. A lot of problems were therefore found, resulting in quite a lot of rework. Although not all difficulties for users could be solved the second version was considered to be superior to its predecessor. This thus shows in participation, so the second version is placed more to the right of the graph. However, the success positioning is not as high as might be expected with regard to the amount of user participation. This mainly has to do with the low scores on contribution to efficiency and the overall success rating of the application by the respondent.

*Thirdly*, it appeared that the P-Direkt application to be used by several Ministries is hard to compare with regard to its success as Table 3 indicates. For questions 1 and 2, the scores by the respondent were given 2 and 1 respectively. The current application is still only partly implemented and used by two departments, while the goal is to use it at all government departments. Therefore questions 3 to 6 are not applicable in this case.

*Fourth and final*, we need to clarify how we quantified user participation in order to score the four cases on this dimension and plot it against their application successes. We judged that the Ministry of Health, Welfare and Sports demonstrated relatively the most time and effort spent on user participation. For instance all managers were approached by demo sessions and invited to comment on the application. Although the different approaches used are less than with the Ministry of Defence, the relative amount of time spent is considered to be more, so the P-Loket application is considered to score higher on user participation than the second ESS-version at the Ministry of Defence. Besides the delay in the start of the project, not a lot of problems arose during the development of the application. Also most of the problems users experienced with the application were caught in the different tests during the development. The amount of rework to be done was therefore considered 'much', but it could be done early in the development. Since the application introduced self-service, it contributed a lot to the efficiency of the organisation. From the interviews it became clear that users participated less in the cases of Emplaza 4.3 and the first version of the Ministry of Defence, therefore they are ordered at the lower end of this dimension.

Given these remarks, the user participation and application scores for the different cases are plotted in Figure 2, recognizing that for two cases actually two measurements in time are included.

**[FIGURE 2]**

Without claiming to have precise measurements and quantifications, Figure 2 clearly confirms that user participation is positively related to (perceived) application success. This is supported by both cross-

sectional comparing the four cases, as well as comparing the two Emplaza and PeopleSoft/HR cases over time. The implications of this hypothesized and convincing result are discussed in next closing section.

## CONCLUSIONS AND DISCUSSION

This paper departed from an analysis of current literature on information system success, user satisfaction and user involvement. A number of researches were found that described what factors influenced the success of information systems. From the DeLone & McLean IS Success Model the concepts of system quality, (intention to) use and user satisfaction were found to be important influential factors. Other findings mentioned the influence of perceived usefulness and perceived ease of use on user satisfaction and intention to use. Subsequently these concepts could be influenced by the involvement of users in the development process of new software applications. Next to this a distinction between the concepts of involvement and participation was suggested. Several findings of positive relationships between user participation, involvement and system success were presented. In the end the literature study was combined into a conceptual model. This model visualised the mentioned links between a number of concepts of the DeLone & McLean's IS Success Model, the Technology Acceptance Model and User Involvement & Participation.

The case studies portrayed are based on interviews with civil servants employed at different governmental organisations. One of the outcomes was a list of currently used user participation methods. In line with the findings from the literature study respondents also argued that users should be involved early, however, not too early because it would delay the development process too much. The challenges faced with involving users also did not deviate from the literature section.

Investigating the cases points also to the positive effect of user participation on the success of an application. Projects that have users participating in the development seem to be more successful than the ones that show less user participation. The only clearly visible exception is the case of the Ministry of Defence. It might be hard to compare the success of the different applications, however the differences between versions of applications are obvious (Emplaza and the Ministry of Defence).

Besides this confirmation of the positive results of user participation, also a number of lessons can be learned when studying these cases. A number of hints and points of attention were even explicitly mentioned by the respondents with regard to user participation. A number of important hints are listed below:

- *User participation requires time and a good schedule*

It is important to think about the consequences of participating end-users. Input from users will need gathered and put in order, this takes time. Subsequently the results need to be analysed to for instance decide which requirements should be incorporated in the design or which findings should be solved. Using MOSCOW-lists, it is possible to rank requirements and suggestions in "must have", "should have", "could have" and "would have" items. To ensure the project stays on schedule, it is necessary to set deadlines when decisions need to be made, otherwise endless discussions might arise and requirements will keep changing. Also concerning scheduling is to make sure end-users are brought in after some basic ideas are already thought of by the development team.

- *Try to find motivated end-users and have something to show them*

In choosing users to have participating in the project, try to find the ones that will be motivated to constructively think about the application. It might not be an easy task to do this in such large organisations, but the network of the project group or managers could be asked to produce a list of possible participators. To enable this set of end-users to come up with useful suggestions, it is wise to visualise parts of the application already. People will find it difficult to supply ideas without something they can see, even a simple mock-up will be fine to start a discussion.

- *Keep in mind the overall process that needs to be supported or automated*

The development of the self-service application itself is not the main goal of the project. There is a process or task that needs to be automated or supported. When designing, developing and testing always keep in

mind this process or task. For instance observe users when executing a task to find out what other processes might be linked to this task. Or, when testing, ask the test person about his or her perceptions on what has happened and what the next step in the process will be.

- *Development team on location*

Having the development team close to end-users, for instance on location, shortens the communication lines. This enables more frequent consultation between end-users and programmers, concerning for instance uncertainties about requirements or just asking user's opinions on what has been developed thus far. Being able to follow the progress more easily will also positively influence the involvement of end-users.

- *Expectancy management*

Make sure to tell participating users what will be done with their input and why. Not all of their suggestions and problems might be implemented or solved. In ensuring they maintain willing to cooperate it is important to communicate why certain decisions have been made and why some of their input is not visible in the developed application.

A number of additional lessons were mentioned by the respondents. It was mentioned to keep in mind employees should be able to use the self-service applications without too much support from the HR-departments. Otherwise it would only be a shift of the workload for the HR-department from HR-tasks to supporting users with the use of the application. In that case the organisations would not show the targeted improvements in efficiency. A third important aspect to take into account is the distinct decision process organisations like the Ministry have. This was a confirmation of one of the points in the report of The Netherlands Court of Audit. The decision process includes fairly a lot of people, takes considerable time and can be politically oriented.

## REFERENCES

- Barki, H., & Hartwick, J. (1989). Rethinking the Concept of User Involvement. *Rethinking the Concept of User Involvement* , 13 (1), 53-63.
- Baroudi, J. J., Olson, M. H., & Ives, B. (1986). An Empirical Study of the. *Communications of the ACM* , 29 (3), 232-238.
- Beyer, H., & Holtzblatt, K. (1999). Contextual design. *Interactions* , 6 (1), 32-42.
- Blackburn, J., Scudder, G., & Van Wassenhove, L. N. (2000). Concurrent Software Development. *Communications of the ACM* , 43 (11), 200-214.
- Blomberg, J., Giacomi, J., Mosher, A., & Swenton-Hall, P. (1993). Ethnographic field methods and their relation to design. In D. Schuler, & A. Namioka (Red.), *Participatory Design: Principles and Practices* (pp. 123-155). Hillsdale: Lawrence Erlbaum.
- Blomquist, Å., & Arvola, M. (2002). Personas in action: ethnography in an interaction design team. *Proceedings of the second Nordic conference on Human-computer interaction* (pp. 197 - 200). ACM.
- Borland. (2006). Opgeroepen op April 25, 2007, van [http://www.borland.com/resources/en/pdf/solutions/rdm\\_whitepaper.pdf](http://www.borland.com/resources/en/pdf/solutions/rdm_whitepaper.pdf)
- Brown, D. (1996). The challenges of user-based design in a medical equipment market. In D. Wixon, & J. Ramey (Red.), *Field Methods Casebook for Software Design* (pp. 157-176). New York: Wiley.
- Butler, M. B. (1996). Getting to know your users: usability roundtables at Lotus Development. *Interaction* , 3 (1), 23-30.
- Chatzoglou, P. D., & Macaulay, L. A. (1996). Requirements Capture and Analysis : A Survey of Current Practice. *Requirements Engineering* (1), 75-87.
- Chatzoglou, P. D., & Macaulay, L. A. (1996). Requirements Capture and Analysis: A Survey of Current Practice. *Requirements Engineering*, 1 (2), 75-87.
- Cherry, C., & Macredie, R. D. (1999). The importance of Context in Information System Design: An Assesment of Participatory Design. *Requirements Engineering* , 4 (2), 103-114.
- Clement, A., & Van den Besselaar, P. (1993). A retrospective look at PD projects. *Communications of the ACM*, 36 (4), 29-37.
- Damodaran, L. (1996). User involvement in the systems design process-a practical guide for users. *Behaviour & Information Technology*, 15 (6), 363 - 377.
- Davis, F. D. (1989). Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. *MIS Quarterly*, 13 (3), 319-340.

- DeLone, W. H., & McLean, E. R. (1992). Information system success: The quest for the independent variable. *Information Systems Research* , 3 (1), 60-95.
- DeLone, W. H., & McLean, E. R. (2003). The DeLone and McLean Model of Information Success: A Ten-Year Update. *Journal of Management Information Systems* , 19 (4), 9-30.
- Dutch Court of Audit. (2007). Lessons from IT-projects at the government [Lessen uit ICT-projecten bij de overheid] Retrieved at [http://www.rekenkamer.nl/9282000/d/p425\\_rapport1.pdf](http://www.rekenkamer.nl/9282000/d/p425_rapport1.pdf) [in Dutch]
- Fister Gale, S. (2003). Three Stories of Self-Service Success. *Workforce* , 82 (1), 60-63.
- Gibson, H. (1977). Determining User Involvement. *Journal of System Management* , 20-22.
- Gould, J. D., & Lewis, C. (1985). Designing for Usability: Key Principles and What Designers Think. *Communications of the ACM* , 28 (3), 300-311.
- Grudin, J. (1991). Systematic sources of suboptimal interface design in large product development organization. *Human Computer Interaction* , 6 (2), 147-196.
- Ives, B., & Olson, M. H. (1984). User Involvement and MIS Success: A Review of Research. *Management Science* , 30 (5), 586-603.
- Ives, B., Olson, M. H., & Baroudi, J. J. (1983). The measurement of user information satisfaction. *Communications of the ACM* , 26 (10), 785-793.
- Janssen, M., & Joha, A. (2006). Motives for establishing shared service centers in public administrations. *International Journal of Information Management* , 26 (2), 102-115.
- Kensing, F., & Blomberg, J. (1998). Participatory Design: Issues and Concerns. *Computer Supported Cooperative Work* , 7 (3-4), 167-185.
- Konradt, U., Christophersen, T., & Schaeffer-Kuelz, U. (2006). Predicting user satisfaction, strain and system usage of employee self-services. *International Journal Human Computer studies* , 64 (11), 1141-1153.
- Kujala, S. (2003). User involvement: a review of the benefits and challenges. *Behaviour & Information Technology* , 22 (1), 1-16.
- Lin, W. T., & Shao, B. B. (2000). The relationship between user participation and system success: a simultaneous contingency approach. *Information & Management* , 37 (6), 283-295.
- Littlejohn, S. W. (2002). *Theories of Human Communication*. Belmont: Wadsworth/Thomson Learning.
- Lucas, H. J. (1974). Systems Quality, User Reactions, and the Use of Information Systems. *Management Informatics* , 3 (4), 207-212.
- Mantei, M. M., & Teorey, T. J. (1988). Cost/Benefit Analysis for Incorporating Human Factors in the Software Lifecycle. *Communications of the ACM* , 31 (4), 428-439.
- Mason, R. O. (1978). Measuring information output: A communication systems approach. *Information & Management* , 1 (4), 219-234.
- McKeen, J. D., & Guimaraes, T. (1997). Successful strategies for user participation in systems development. *Journal of Management Information Systems* , 14 (2), 133-150.
- McKeen, J. D., Guimaraes, T., & Wetherbe, J. C. (1994). The Relationship between User Participation and User Satisfaction: An Investigation of Four Contingency Factors. *MIS Quarterly* , 18 (4), 427-451.
- Ministry of Internal Affairs (2006). Press statement Retrieved July 21, 2007, from P-Direkt: [http://www.p-direkt.nl/index.cfm?action=dsp\\_actueelitem&itemid=QKNGJL8E](http://www.p-direkt.nl/index.cfm?action=dsp_actueelitem&itemid=QKNGJL8E).
- Ministerie van Binnenlandse Zaken en Koninkrijksrelaties. (2007). Nota Vernieuwing Rijksdienst. Opgeroepen op Februari 08, 2008, van <http://www.minbzk.nl/asp/download.aspx?file=/contents/pages/89897/notavernieuwingrijksdienst.pdf>
- Muller, M. (2001). A participatory poster of participatory methods. Conference on Human Factors in Computing Systems, CHI '01 extended abstracts on Human factors in computing systems, (pp. 99 - 100).
- Nielsen, J. (1993). *Usability Engineering*. San Diego: Academic Press.
- Noyes, P. M., Starr, A. F., & Frankish, C. R. (1996). User involvement in the early stages of an aircraft warning system. *Behaviour & Information Technology* , 15 (2), 67-75.
- Pekkola, S., Niina, K., & Pasi, P. (2006). Towards Formalised End-User Participation in Information Systems Development Process: Bridging the Gap between Participatory Design and ISD Methodologies. *Proceedings of the ninth Participatory Design Conference 2006* , 21-30.
- Robey, D., & Farrow, D. (1982). User Involvement in Information System Development: A Conflict Model and Empirical Test. *Management Science* , 28 (1), 73-85.
- Rowley, D. E. (1996). Organizational considerations in field-oriented product development: Experiences of a cross-functional team. In D. Wixon, & J. Ramey (Red.), *Methods Casebook for Software Design* (pp. 125 - 144). New York: Wiley.
- Ruël, H., Bondarouk, T., & Looise, J. K. (2004). E-HRM: Innovation or Irritation. An Explorative Empirical Study in Five Large Companies on Web-based HRM. *Management Revue* , 15 (3), 364-380.
- Saarinen, T. (1996). SOS An expanded instrument for evaluating information system success. *Information & Management* , 31 (2), 103-118.

- Shannon, C. E., & Weaver, W. (1949). The mathematical theory of communication . Urbana: University of Illinois Press.
- Shapiro, D. (2005). Participatory design: the will to succeed. Proceedings of the 4th decennial conference on Critical computing: between sense and sensibility, (pp. 29-38). Aarhus, Denmark.
- Simonsen, J., & Kensing, F. (1997). Using ethnography in contextural design. Communications of the ACM , 40 (7), 82-88.
- Strohmeier, S. (2007). Research in e-HRM: Review and implications. Human Resource Management Review , 17 (1), 19-37.
- Verheijen, T. (2007). Gestrikt: E-HRM komt er uiteindelijk toch. Personeelsbeleid , 43 (11), 20-23.
- Wood, L. E. (1997). Semi-structured interviewing for user-centered design. Interactions , 4 (2), 48-61.



**FIGURES AND TABLES**

Figure 1: D&M (Updated) IS Success model (DeLone & McLean, 2003)

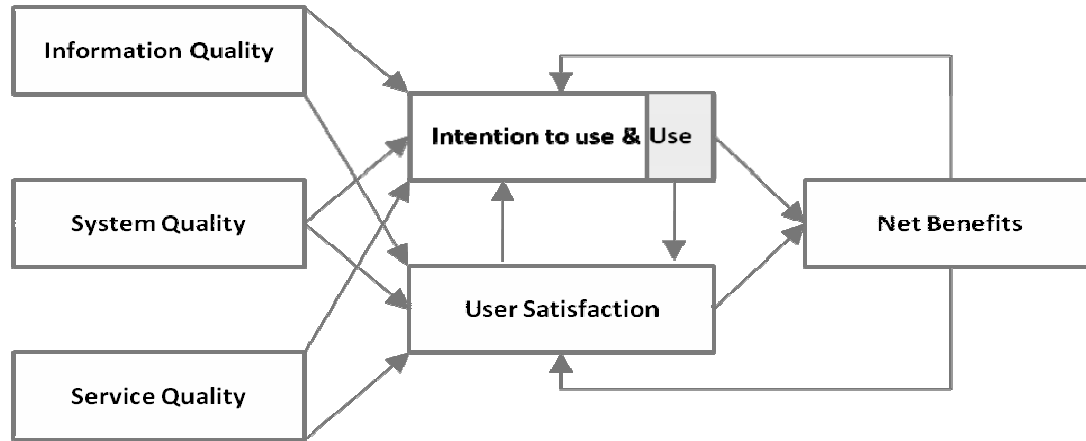


Table 1: Potential early participation methods

Method	Approach (Kujala, 2003)
Observation	User-centred Design / Ethnography
Interviews	User-centred Design
Discussion	User-centred Design
Prototyping	User-centred Design / Participatory Design / Contextual Design
Brainstorming	Participatory Design
Workshops	Participatory Design
Organisational gaming	Participatory Design
Video analysis	Ethnography
Contextual Inquiry	Contextual Design

Table 1: Used methods to involve end-users

	Prototypes	Testing	Use-research
Emplaza at the Ministry of the Interior And Kingdom Relationships		x	
P-Direkt to be used throughout several Dutch Ministries		x	
P-Loket at the Ministry Of Health, Welfare and Sport	X	x	
PeopleSoft/HR at the Ministry Of Defence	X	x	X

Table 2: Scores by respondents to extra questions

	<b>Emplaza at the Ministry of the Interior And Kingdom Relationships</b>	<b>PeopleSoft/HR at the Ministry Of Defence</b>	<b>P-Loket at the Ministry Of Health, Welfare and Sport</b>	<b>P-Direkt to be used throughout several Dutch Ministries</b>
<b>1 Reported problems after test</b>	5	3	2	2
<b>2 Amount of rework needed</b>	5	4	4	1
<b>3 Current satisfaction level</b>	4	4	4	n/a
<b>4 Questions at helpdesk</b>	2	3	4	n/a
<b>5 Contribution to efficiency</b>	4	2	4	n/a
<b>6 Overall success</b>	3	2	4	n/a

Figure 2: User participation and application success

