Situational Application Usage: A Framework for Success in Enterprises

Kevin Mcneill, Sofia Meacham

Department of Computing and Informatics, Bournemouth University, Fern Barrow, Poole, Dorset, BH12 5BB, UK.
1359442@bournemouth.ac.uk, smeacham@bournemouth.ac.uk

Abstract

Situational Application usage is on the increase. Instead of targeting large scale projects which traditional IT developments address, Situational Applications look to remediate the often forgotten long-tail of business requirements. They are often considered as temporary solutions to an immediate demand, which (solutions) fit poorly with an organisation's robust software development and delivery lifecycle. Despite both fulfilling a legitimate business need, and seeing an increase in usage, there are few positive or well publicised success stories within industry and/or academia. In this paper we will investigate the motivational use of Situational Applications in enterprise, as well as the benefits and risks which they bring to both end users, and organisations as a whole. Whilst it is beyond the remit of this paper to answer how situational applications can be implemented successfully, a methodology will be proposed such that further research can be undertaken. The method defined, if implemented, would allow the creation of a template for successfully creating, deploying and maintaining situational applications. Furthermore, it would facilitate the creation of a risk analysis model to determine if situational applications are appropriate for a given organisation as well as document key performance indicators for measuring success. If an enterprise was truly able to successfully create, deploy and maintain Situational Applications consistently, then the possible benefits to an organisation's return on investment would be significant. Such benefits could be enough to challenge the use of traditional software applications for addressing business requirements as well as existing widespread software development lifecycle approaches.

Keywords: Situational applications, Mashups, Enterprise mashups, Web 2.0, Software development lifecycle

1.0 Introduction

The concept of Situational Applications (SA's) is not a new one. They have long existed, going under many different names; situated applications, 'good enough' software, opportunistic software and mashups to name but a few. In this paper, the term that will be used is Situational Applications. Shirky [20] defined SA's as software "designed for use by a specific social group, rather than for a generic set of users". Similarly, Balasubramaniam et al. [3] agree, adding that it is "personalized, localized software that has evolved organically and has been created by the community that uses it". These interpretations differ slightly from that of Jhingren [6] who focusses in on how SA's are those software products "constructed on the fly for some transient [business] need".

Unfortunately SA's remain a poorly defined term. Many have argued that it is easier to define them by what they are not. SA's are generally not created by traditional software developers, they are not hosted on robust architectures, they do not serve a long term purpose and invariably they are not normally scalable [3]. Although differences of opinion still exist, there appears to be some agreement amongst researchers and industry experts pertaining to the key characteristics that SA's usually possess. In their seminal work, Cherbakov et al. [19] presented an in-depth analysis of what factors that identify SA's. They proposed that SA's are:

- Created for a specific user community, often a small subset of the larger organisational user base.
- Required to fulfil an immediate business requirement and thus transitory in nature with a short life span.
- Built by end-users in an iterative fashion, often outside of the remit and control of central IT departments.
- Developed using an iterative development lifecycle leading to a short time-to-market.
- Created to address the 'long-tail' of business requirements that traditional enterprise application development projects would not usually address.

It is widely accepted that SA's have been in existence for a number of decades, going undefined until the term 'Situated Software' was coined by Shirky in his groundbreaking essay of the same name [20]. One of the most well known SA's is Microsoft Excel [7] giving users the ability to create relatively complex and interactive spreadsheets through the utilisation of in built VBA code and macro functionality, along with extensive datasets. In recent years the advancement of Web 2.0 technologies, such as AJAX, as well a push towards more collaborative and communal working, has brought about an increase in SA usage within the enterprise [15].

Much of the available literature on SA's examines the definition, rise in popularity, and nuances of their development lifecycle. Very little research to date focusses on how to create, deploy and maintain SA's in a successful and sustainable manner.

Research in this area is critical to understand if SA's can become a complimentary addition to a company's existing traditional software development stack. This would fill a niche that clearly exists for rapid delivery on requirements that would never normally be addressed. The benefits to an organisation if they were able to utilise such software could be profound.

Much of the available literature on SA's examines themes such as their history, rise in popularity and notable characteristics for identification. Furthermore, many studies even present prototype development environments to aid in their creation. There is a notable lack of coherent and methodical analysis to address vital questions such as:

- Why are users creating SA's?
- What factors are contributing to the rise in popularity of SA's?
- What are the common types of Situation Applications?
- Who usually creates/maintains SA's?
- Do companies encourage or discourage the use of SA's, why?
- What benefits do SA's bring to the Business Users vs. the company as a whole?
- What risks/issues are presented through the use of SA's?
- Should companies embrace the use of SA's, if so in what form?
- Is there a legitimate gap in software delivery within a company that SA's can fulfil? If so, how?
- How can SA's be implemented successfully and how can it be measured?

In this paper we will present a research methodology that, if implemented, aims to elicit the necessary information to specifically focus on the last question highlighted above, i.e. how can SA's be successfully developed, implemented and maintained.

If we are to take the widely accepted belief that SA usage is on the increase [21] then it is clearly apparent that there is an urgent need to address the questions highlighted above. The outcome of this research aims to enable better integration with existing enterprise technology stacks, improving the utilisation of resource skill-sets (in the form of non-technical business users), delivering on a larger proportion of business requirements, reducing overall IT costs and ultimately, increasing profit margins.

It is the authors experience of working with SA's in the IT Sector that has driven this research. This involvement within industry consisted of working on a project to create a new web application platform and development framework which was used to migrate large numbers of standalone SA's. Thus, bringing them into IT control in a scalable and repeatable manner fashion

In this paper we will discuss what research is needed in order to add further value and determine whether it is sensible for enterprises to utilise SA's and if so, how this can be achieved successfully. This paper has been divided into 3 sections. In the first section we examined related research concerning SA's. The second, and primary section, is concerned with outlining a proposed methodology to answer the research questions discussed thus far. The third section discusses the limitations of the proposed research method and potential future work. Finally, the paper will end with a conclusive summary.

2.0 Related Work

2.1 The Rise of SA's

As explained in the introduction, it is clear that much of the available literature concerning SA's focuses upon the recent surge in interest amongst end users. However, interest alone does not always lead to adoption. Yu et al. [10] highlights that there has been significant interest being shown in Mashups, a type of SA, but point to the notable lack of widespread uptake. In their detailed study of Mashups they suggest that this is attributed to a lack of formalised and established development frameworks aimed at SA's vs their traditional software counterparts.

Zou and Pavlovski [11] observe that the lack of accountability with Mashup implementations could also be limiting their take up within enterprise. Others, such as Maraikar et al. [14] go even further by proposing a prototype development platform for Mashups which could improve the adoption rates. However, these studies fail to consider whether focusing on a subset of SA's, Mashups, is representative of the group as a whole. Whilst both studies suggest a framework for how to implement SA's or whether it is appropriate for a particular organisation to do so. Neither defines how to measure success in anyway, and are thus of limited use.

With traditional software it is standard governance practice within an organisation for IT development activities to be owned by a centralised department, calling on staff with specialist skill sets. Invariably the proportion of IT Developers is small in comparison to the overall personnel base. The largest employee segment of most organisations by far is the non-technical users, SA's are developed by such users to address their own requirements rather than involve IT Developers who will likely not even understand the users business domain at any great depth [18].

It is therefore clear that the IT Department itself, with a small resource pool, could easily become a bottleneck for fast and effective development. The evidence from this study suggests that organisations would be remiss to ignore the benefits such users can bring in developing SA's. In the same vein, Cherbakov et al. [19] suggests that by providing appropriate tools and development frameworks to non technical business users they are able to contribute even greater to an organisations development agenda. However, they do not go so far as to say how this should be accomplished in order to lead to a successful outcome.

Addressing niche business requirements is one of the critical defining factors of SA's [1]. Such software is created to address the long tail of business requirements, as opposed to the traditional big ticket items which software projects usually focus on. This alone has the capability of becoming a game changer for those enterprises who are able to successfully deploy and maintain SA's. Anderson [1] studied this effect, known as the 'long-tail'. He identified that considerable opportunity existed in the smaller niche markets and desires of consumers than when compared with the opportunities in the larger high volume 'common' market segments. For example, a large proportion of Amazon's revenue comes from the many varied products that only sell in very small numbers. Similarly, Han et al. [8] notes the same conceptual link between SA's and addressing niche requirements.

Company's are now able to open up new opportunities and increase revenue streams by addressing requirements that, before, would have been unfeasible to target, or addressed too late to see a return on investment [9]. The need for business agility in increasingly competitive market arenas is key is business are to succeed and flourish. This view is supported by Liu et al. [13] who write of the increasingly important need for businesses to respond quickly to changing, often complex, requirements within the 'long tail' in order to succeed.

2.2 Similarities and Differences

Another area where a considerable amount of literature has been published concerns the development, technology and governance aspects of SA's compared with traditional applications. A key study by Cherbakov et al. [19] found that the development lifecycle for SA's differs greatly from traditional software applications. Firstly, time to market is significantly reduced, secondly there is little to no project management or milestone tracking, thirdly functional requirements are not formally recorded and non functional requirements such as scalability are not even considered. Finally, there is no defined testing period, instead opting for continual testing of the application through actual usage.

From a technology perspective SA's are created using tools that are far less mature than traditional applications. For example, they are often created with Microsoft Excel or emerging development platforms such as Google Mashup Editor, Microsoft Popfly and Yahoo Pipes [10]. Further, Cherbakov et al. [19] point to the often transitory nature of SA development platforms. With the exception of Excel, the majority of technology tools referenced in SA literature have been decommissioned or retired. In contrast, major development languages used in traditional software applications often have lifespans of many decades.

It is not simply the development lifecycle and technologies that differ, the way in which enterprises manage them is of significant interest. Balasubramaniam et al. [3] highlight that SA's are more decentralised by their very nature, often initiated and owned by end users as opposed to management executives or sponsors. Control of the applications is also shifted from a centralised IT Departments ownership model to localised groups of users.

When reviewing the available literature it is clear that there is an overriding bias towards highlighting differences, rather than similarities. It is difficult to ascertain whether this is a gap in research or simply highlighting a pertinent fact about the two types of traditional vs situated applications. Further research is needed in this area in order to answer this question appropriately.

2.3 Benefits and Challenges

It is with the benefits and challenges of SA's that we see the true potential of the research being proposed. Cherbakov et al. [19] lists three groups of benefits which SA's can bring to an enterprise: (1) empowering businesses through encouraging innovation, eliminating frustration and improving morale; (2) improve business solutions by developing closer fit solutions, addressing niche requirements, complimenting existing IT development methodologies and traditional software products; (3) improve return on investment through shorter development periods, utilising non-technical business users for development and thereby cuttings costs, addressing requirements that would have unlikely been addressed with traditional software projects. Almost every paper that has been written on SA's includes at least one or more of these benefits. Xie et al. [12], for example, point out that SA's are a useful complimentary tool in an enterprises software offering.

De Vrieze et al. [17] draws our attention to a number of difficulties that can be experienced in adopting SA's, specifically mashups, but these points are still valid and can be applied to the SA group as a whole. They list five significant challenges in enterprise adoption of SA's: (1) they are not a mature technology offering; (2) it is difficult to implement effective change control processes as applications are distributed and not always entirely under IT control; (3) security and privacy concerns must be addressed; (4) IT governance and ownership must be thought through; (5) duplication of work is possible when end users are able to go off and create SA's in isolation.

3.0 Methodology

It is evident that a number of important themes are lacking from the related work discussed is section two: (1) whilst SA usage is on the rise, should enterprises actually be using them? (2) how is risk impact assessed for enterprises wishing to use SA's?; (3) how can SA's be created, implemented and maintained successfully?; (4) how do we define success in a comprehensive and consistent manner?

3.1 Research Question

How can business organisations successfully develop, implement and maintain situational applications alongside traditional software application offerings?

3.2 Aims and Objectives

In order to successfully answer the question above, the research will be designed to meet the following aims and objectives. The authors aim is threefold.

Aims:

- To outline an implementation framework to be used in guiding business management in creating, deploying and maintaining SA's successfully.
- To create a substantiated risk analysis model that can be used to determine if SA's are appropriate for a given organisation or situation.
- To identify and define a suite of key performance indicators to aid in defining what can be classified as a successful SA.

Objectives:

The aims identified above will be accomplished by fulfilling the following research objectives:

- Define what constitutes a situational application, including attributable characteristics.
- Review literature concerning SA usage by enterprise organisations.
- Collect data on the usage of SA's in a real world company.
- Compare the creation, implementation and maintenance activities of SA's with that of traditional software applications.
- Measure any change in development trends of SA's in a real world company.
- Identify who creates SA's in a real world company.
- Compare the IT and Business governance oversight process of SA's with that of traditional software applications.
- Identify and investigate risks associated with enterprise usage of SA's.
- Identify the key metrics to determine how to measure success when utilising SA's with an enterprise.
- Identify how to successfully create, implement and maintain SA's in an enterprise setting.

3.3 Research Design

The first step in any research is always a comprehensive literature review. Based upon the review within this paper it should be clearer to the reader as to where gaps in existing SA research lie, and thus where the author intends to focus, this can be seen in the aims and objectives section above. A systematic literature review was considered but rejected as it is not viable or appropriate due to the small volume of available research material concerning the research questions highlighted e.g. how to create, implement or maintain SA's. Instead, the majority of literature focuses on

defining what SA's are and the benefits they offer. This is not helpful in answering the research questions laid out in this paper.

The aim of this paper is to set out a research methodology that could be implemented in order to solve the questions put forward earlier in this section. In order to do this, it would be necessary to secure real world data and thus a case study approach should be followed, selecting a large scale enterprise organisation which currently uses and develops SA's. This type of ethnographic approach will allow the author to explore, in detail, the themes highlighted from the literature review as well as meet the above aforementioned objectives. A case study approach benefits from encouraging data triangulation through the examination of multiple data sources, e.g. documents, interviews and focus group accounts. Whilst it is acknowledged that more than one case study would be beneficial it is difficult to ensure engagement from multiple companies, instead a focus will be maintained on one company allowing an in-depth analysis. If resources allowed a small scale pilot case study could be performed in order to fine tune the approach before engaging in a more costly (in time and money) full scale case study.

An exploratory sequential mixed methods research approach is proposed. In addition to the benefits of data triangulation, mixed methods leads to methodological triangulation also. The biggest advantage of a mixed methods approach is that the weaknesses of one method will often be countered by the strengths of the others, this enables the overall study to weather result nuances or inconsistencies, leading to a more credible outcome [5]. The problem this paper sets out in section 3.1 is primarily exploratory in nature and thus beginning with a qualitative research approach is logical. It allows an in-depth exploration of managers and end users perceptions of SA critical success factors within their natural organisational setting. Once key themes around success and risk have been explored the second phase of the study will turn to quantitative methods to allow a wider focus through the use of data extraction on usage of SA's in particular lines of business or departments as well as through surveys. The result being it is then more feasible to generalise from the findings through the larger sample size.

3.3.1 Interviews and Focus Groups

The first phase of research focusses on qualitative interviews and focus groups. A minimum of four interviews will be held with participants being selected from a variety of organizational roles, these will be chosen following completion of a pre-selection survey to identify those best suited take part in the Case Study. Examples may include:

- Chief Technology Officer
- Risk Manager
- Operational Analysts

The interview subjects should be chosen from the roles identified through the pre-selection survey and should be very familiar with SA's and their use at the chosen

organisation. It is necessary to begin the research by gleanings much in-depth knowledge as possible - starting with expert users and managers gives the best chance of eliciting the most information downstream in later research phases, e.g. focus groups and surveys. Demographic variables, such as age and gender, are not relevant to the study and therefore no demographic stipulations will be made against the chosen interview population. The selection of interviewees will be made in combination with an approved representative at the chosen organisation, i.e. the gatekeeper, this individual will facilitate all access to materials or people for the researcher. The gatekeeper will advise and suggest a pool of resources who meet the role criteria outlined above for selection by the researcher.

Given the exploratory nature of the SA problem domain a semi structured interview type will be followed. This gives the advantage of providing key questions or subject areas for discussion whilst avoiding the very rigid and inflexible nature of structured interviews which could result is the researcher being unable to follow interesting or emerging themes. The following open discussion questions will be followed in the semi structured interview:

- How are SA's created, implemented and maintained?
- What are the risks with SA's?
- What makes a successful SA?

The intent of this research is to examine multiple themes relating to SA's, such as creation, risk and governance, therefore it is logical to use focus groups as an additional method to garner feedback. Focus groups are beneficial when exploring multiple themes, especially when little is currently known about the research area. The interactive nature of focus groups gives a distinct advantage over stand alone interviews, often they are able to elicit more information by promoting open discussion. Focus Group participants will be drawn from a pool of expert and non-expert users across the four role categories previously highlighted. To get the most out of the focus group a sample size of 8 will be taken; one expert and non-expert from each of the four role categories defined. Furthermore, as the intent of this research is to ultimately create a survey, to allow for more quantitative data collection and analysis, focus groups function well in helping to find areas of significance and in identifying common user language, these points are all useful in developing a successful survey design.

Interviews and Focus Groups will be held on the company premises in a private office in order to meet on natural territory for the participants, as well as an environment that is safe and secure. All participants will be first presented with a ‘Participant Information and Consent’ form that confirms they understand the study, that their participation is voluntary, that they may withdraw at any time and that their anonymised responses will be used in the specified research. Given that the participants will be employee's of the case study organisation, and thus subject to a multitude of privacy and code of conduct regulations, all forms will include a passage that affirms participation has been approved by their employer and will be

signed by the designated company gatekeeper. All interviews and focus groups will be audio recorded, in agreement with the organisation, for later transcription and analysis but the researcher will make notes throughout. It is acknowledged that some participants may be uncomfortable with audio recording and therefore all participants will be asked if they object before commencement.

3.3.2 Dataset Extraction

Moving on from the qualitative interview and focus groups, the research will progress to more quantitative methods such as dataset extraction. Datasets will be collected through the case study organisation to identify all SA's that have been created by scanning networked drives with a specialised tool, such as Prodiance eDiscovery⁻. The resultant datasets will provide a full inventory that is risk assessed against predefined criteria. This information will be helpful in answering the research question identified above as well as shaping the survey design.

3.3.3 Surveys

The second phase of quantitative research, surveys, builds upon the foundations of the qualitative first phase. A single department at the chosen case study organisation will be targeted for the survey recipients. The choice will be made in consultation with the organisation gatekeeper who will be able to assist in identifying a department who are significant users/creators of SA's and therefore would make a good choice of recipient group. The survey script will focus on the areas of investigation found during the interview, focus group and data extract stages. Whilst the data collected from these stages is highly useful in determining themes and answering some of the research questions, it does not allow sufficient generalisations to be made. Thus, the survey as another research tool will be useful here. As with the interviews and focus groups, participants in the survey stage will be first presented with a 'Participant Information and Consent' form.

3.4 Analysing Data

In order to examine the data collected from the methods defined above a variety of data analysis techniques will be utilised. The primary analysis technique for the interviews and focus groups will be to transcribe and code all data, leading to the creation of categories and ultimately themes that will be used to guide the survey creation. The survey results will be analysed using a suit of standard statistical inferential analysis techniques that will be more applicable to the larger sample size the survey returns.

3.5 Ethical Considerations

In order to meet the Bournemouth University Research Ethics Code of Practice [4] the first and foremost consideration for the proposed research is that it should do no harm, either to individual participants, case study organisations or researchers themselves. Privacy and anonymity of the business organisation participating in the case study should be upheld as any information relating to their use of SA's is deemed commercially sensitive. A Non-Disclosure Agreement should be drawn up

and signed by parties involved in the research study, this ensures that commercially sensitive data remains protected and secure. Given that interviews with staff may touch upon commercially sensitive data, any information gleaned as part of the research must be secured and held confidentially, it should not be shared or passed to any other individual or organisation without prior consent. Interview subjects themselves must be informed as to the purpose of the research and what will be done with the data collected, thereby ensuring informed consent. At any point during the data collection phase participants should be able to withdraw from the process without penalty.

During interviews and focus groups the researcher must ensure that they do not cross the line and become intrusive or collect data outside of the remit which has been agreed. For example, it is understandable to record data pertaining to types of SA's, general functionality and feedback etc. But, storing any data held within the applications is likely to be unnecessary and could be deemed intrusive. This could damage relationships between the case study participant, university and researcher, not to mention it would go against the first rule to do no harm [16].

In accordance with Bournemouth University Guidelines an Ethics Checklist would be required to be submitted as per the online template⁻.

4.0 Discussion

In the proposed investigation there are several sources of potential errors limiting the usefulness of results. The main issue is the sample size of the case study, just one organisation. The research method detailed in section 3 has been designed to extract as much data from the case study organisation as possible, through a progressive approach of qualitative research leading to more generalisable quantitative results. Although useful information could be gleaned from the research methods detailed in this paper, it is important to avoid over fitting the results to just the organisation in question and be cautious when extrapolating results for wider application. In further research it is recommended that multiple organisations be included to expand the case study sample size.

As it has been noted already, there is a lack of prior research studies in the field of SA's. This has meant the starting point to answer research questions such as those posed in this paper is wider and more generic than the author would prefer. It is therefore necessary to expend more time and energy in this study documenting the foundational understanding and categorisation of the problem domain. Only then, is it possible to go on to focus on the specific objectives and aims surrounding how we implement successful SA's in a risk adverse manner.

5.0 Conclusion

It may not be possible to investigate all themes highlighted during the research process. Commercially sensitive data may be withdrawn, or access prohibited, by

organisation gatekeepers. The impact could limit the usefulness of results and the ability to generalise enough to create a template for implementing SA's successfully or a risk analysis model. Such limitations drive the need for more case study participants even further.

Finally, given the disclosed interest of this author in the topic and his employment with an organisation who could become a case study participant, there is a risk of bias. It is therefore recommended that any chosen organisations are made from a pool of viable choices with which the researchers are not employed or related.

Much is known about SA's and their use by enterprise organisations, yet still there is no widely accepted or utilised template for consistent success in developing, implementing and maintaining them. Neither is there a commonly held view on the risks they pose or address. The research laid out in this paper aims to address these core questions through a sequential exploratory mixed methods case study approach. It is clear that one case study alone will not yield enough data to generalise and create definitive models for success and risk management, however it is the aim of this author that the research study outlined here is a stepping stone in the right direction.

6.0 References

1. Anderson, C., 2006. Long Tail, The: Why the future of business is selling less of more. United States: Hyperion Books.
2. Anderson, C., 2009. The Long Tail - About Me. [online]. Available from: http://www.longtail.com/about.html [Accessed 18 January 2016].
3. Balasubramaniam, S., Lewis, G. A., Simanta, S. and Smith, D. B., 2008. Situated software: Concepts, motivation, technology, and the future. IEEE Software, 25 (6), 50–55.
4. BU, 2009. Research Ethics Code of Practice. Poole: Bournemouth University [online]. Available from: http://blogs.bournemouth.ac.uk/research/files/2012/09/8B-Research-Ethics-Code-of-Practice_February-20142.pdf [Accessed 09 January 2016].
5. Creswell, J. W., 2011. Educational research: Planning, conducting, and evaluating quantitative and qualitative research. 4th edition. Boston, MA: Addison Wesley.
6. A. Jhingran, “Enterprise Information Mashups: Integrating Information, Simply”, VLDB 2006: 3-4.
7. Wang, G., Yang, S., & Han, Y. (2008, November). A Spreadsheet-like Construct for Streamlining and Reusing Mashups. In *Young Computer Scientists, 2008. ICYCS 2008. The 9th International Conference for* (pp. 880-885). IEEE.
8. Han, Y., Wang, G., Ji, G., & Zhang, P. (2013). Situational data integration with data services and nested table. *Service Oriented Computing and Applications*, *7*(2), 129-150.
9. Hoyer, V., Stanoesvka-Slabeva, K., Janner, T., & Schroth, C. (2008, July). Enterprise mashups: Design principles towards the long tail of user needs. In

Services Computing, 2008. SCC'08. IEEE International Conference on (Vol. 2, pp. 601-602). IEEE.
10. Yu, J., Benatallah, B., Casati, F., & Daniel, F. (2008). Understanding mashup development. *Internet Computing, IEEE*, *12*(5), 44-52.
11. Zou, J., & Pavlovski, C. J. (2007, October). Towards accountable enterprise mashup services. In *e-Business Engineering, 2007. ICEBE 2007. IEEE International Conference on* (pp. 205-212). IEEE.
12. Xie, L., De Vrieze, P., & Xu, L. (2009, November). When social software meets business process management. In *Computer Sciences and Convergence Information Technology, 2009. ICCIT'09. Fourth International Conference on* (pp. 238-243). IEEE.
13. Liu, X., Ma, Y., Huang, G., Zhao, J., Mei, H., & Liu, Y. (2015). Data-Driven Composition for Service-Oriented Situational Web Applications. *Services Computing, IEEE Transactions on*, *8*(1), 2-16.
14. Maraikar, Z., Lazovik, A., & Arbab, F. (2008). Building Mashups for the Enterprise with SABRE. In *Service-Oriented Computing–ICSOC 2008* (pp. 70-83). Springer Berlin Heidelberg.
15. Maximilien, E. M., Ranabahu, A., & Tai, S. (2007, October). Swashup: situational web applications mashups. In *Companion to the 22nd ACM SIGPLAN conference on Object-oriented programming systems and applications companion* (pp. 797-798). ACM.
16. Miller, T., Birch, M., Mauthner, M., & Jessop, J. (Eds.). (2012). *Ethics in qualitative research*. Sage.
17. De Vrieze, P., Xu, L., Bouguettaya, A., Yang, J., & Chen, J. (2009, May). Process-oriented enterprise mashups. In *Grid and Pervasive Computing Conference, 2009. GPC'09. Workshops at the* (pp. 64-71). IEEE.
18. Quinn, K., 2007. Not everyone who drives a car fixes it themselves. 2 October 2007 [online]. Available from: http://www.information-management.com/news/columns/-1041222-1.html [Accessed 11 December 2015].
19. Cherbakov, L., Bravery, A. J., & Pandya, A. (2007). SOA meets situational applications, Part 1: Changing computing in the enterprise. *IBM developerWorks*, *23*.
20. Shirky, C., 30 Mar. 2004. Situated Software in Clay Shirky's Writings about the Internet [online]. Available from: http://www.shirky.com/writings/situated_software.html
21. Simmen, D. E., Altinel, M., Markl, V., Padmanabhan, S., & Singh, A. (2008, June). Damia: data mashups for intranet applications. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data* (pp. 1171-1182). ACM.