R4: Towards A Holistic Framework for Designing and Developing Social Web Applications

Sungwon Peter Choe, Junehwa Song

Computer Science Dept.

Korea Advanced Institute of Technology

Daejeon, Korea

{sungwon, junesong}@nclab.kaist.ac.kr

Abstract— Social Web Application (SWA) design and development is a complex process that requires the understanding and coordination of several domains of knowledge. Yet there remain few if any holistic frameworks to manage this process. In this paper we present such a framework, R4. It coordinates and relates the different perspectives that inform the entire SWA design and development process. We base R4 on SWA design and development framework requirements derived from the literature. We formulate and relate four basic perspectives, the User, Culture, Technology and Business spheres and define stages, (Re)Design, Realization and Reformation (RRR), of the entire SWA design and development lifecycle. We then suggest basic SWA design and development guidelines derived from our R4 framework and develop an example methodology based on that framework.

Keywords-social web applications; online communities; social media; design; development; frameworks

I. INTRODUCTION

Social Web Applications (SWAs) have become ubiquitous in the cultural consciousness and daily lives of those of us privileged to be living in the digitalized world. SWAs connect users in online communities [27] who collaborate in creating or sharing social media (e.g. photos in flickr [36] or status updates and urls in twitter [38]). Such applications are now evolving beyond the desktop computing space to platforms like mobile phones and handheld devices (e.g. facebook [35] and twitter [38] have functionality allowing for updates from mobile phones). Soon SWAs will become even more prevalent as they are deployed widely across pervasive computing environments [11].

SWAs are characterized by continuously evolving cycles of interaction [25] and context-informed [10][19] design and development. An initial SWA design [20][27] is first implemented in software [23]. The subsequent behaviors of users of the application then inform modifications to the design [5][25]. The developers then reflect these changes in the software implementation [1][23]. Many outside contextual influences also impact this co-evolutionary process, including cultural trends [10] and business conditions [2][3][19]. Thus, the dynamic characteristics of SWAs require not only the essential domains of software development methodologies [16] and business project management [4], but also an understanding of other pertinent issues such as the user experience, the cultural context, community management and market positioning.

In this paper, we present R4: a framework that manages the complexity of coordinating the different areas of Social Web Application design and development. Most existing research, however, focus on one or two main domains of knowledge only [1][18][25][27]. Existing works also do not explore many lifecycle issues beyond the initial SWA launch [27]. We suggest requirements for a holistic framework that relates the fundamental domains of knowledge over the entire lifecycle of a SWA. R4 represents our vision of what such a framework would look like, based on these requirements. The R4 framework and the seed SWA design and development guidelines facilitate the development of detailed design guidelines specific to individual methodologies and projects.

The complexity of SWA design and development has been evidenced in part by the failure of many early Web applications and businesses during the dotcom era of the late 1990s and early 2000s [8][9][14][17]. Some startups faced scalability or other technology issues (e.g. friendster [8]), some lacked well-defined business plans or were never able to become profitable (e.g. govWorks.com [17] and kozmo.com [9]), others failed to understand the cultural context and reach of the Internet or failed to understand how and why users would make use of their businesses or services (e.g. flooz.com and pets.com [14]).

SWA design and development has since matured significantly, with many important contributions from academia and industry in technological [1][23], business [2][3][19], cultural [27][15] and user-centric [25] fields. Real world SWA design development is informed by and requires the coordination of all these areas that may have conflicting interests or cross-area dependencies. Yet a holistic framework that relates these different perspectives is still lacking. Thus, while many specific focused areas of SWA development have been enlightened, the design and development process as a whole, particularly the coordination of its parts, remains an ad-hoc process.

R4 consists of two main components to relate and coordinate the SWA design and development process: 4Sphere and RRR. 4Sphere presents four perspectives, or spheres, that must be addressed in the design and development process: the User, Culture, Technology and Business Spheres. These spheres provide a conceptual map for categorizing and relating

different SWA issues. The stages of the design and development process itself are outlined in the RRR lifecycle model; the major stages being (Re)Design, Realization and Reformation. Whereas most related work focuses on issues relevant to development pre-launch [20][26][27], RRR aims to explore the challenges and dynamics of later stages of the SWA's lifecycle beyond just the launch of the application. R4 can thus be used to guide the design and development process through each sphere's perspective over the entire lifecycle of the SWA. R4 as presented here is a powerful conceptual tool in articulating and organizing design and development processes. This paper presents the first step in a long-term research that will flesh out the details of the stages and relationships of the entire design and development space mapped out herein.

This paper is outlined as follows. In section II, we review the related work. In section III, we discuss the R4 framework in detail. Section IV proposes basic design and development guidelines for social web applications and demonstrates how a methodology could be constructed for pervasive computing environments. Section V presents future work and discussion. Section VI concludes this paper.

II. RELATED WORK

Web application design and development in general is a well-researched and documented academic subject. More recently, social web applications in particular have become a prevalent research topic in diverse academic fields with the rise of a variety of social networking applications [8] like MySpace [37] and Facebook [35] and other popular SWAs such as Youtube [39] and Amazon [34]. While there is a wealth of academic work done in these areas, most approaches take a focused look at the process from a specific academic discipline's perspective. We outline and categorize some of the notable works in different areas and compare our work to other frameworks and holistic approaches.

A. Focused Approaches

1) Culture

There have been many works on social web application development that look at the process from a community-based (i.e. inner Culture, see section III) perspective [13][20][26][27]. One of the most widely cited and representative of these is Preece's Online Communities: Designing Usability and Supporting Sociability [27]. Preece provides many design guidelines for building online communities within a conceptual framework called Community-Centered Development (CCD). CCD largely discusses inner Culture and User issues. There is thus little or no discussion of how Business or Technology issues affect the design process. There is also little discussion of issues impacting the SWA past the initial launch.

Definitions of community and cultural issues surrounding SWA have been discussed in [8][29][32]. A number of works have explored the overlap of new understandings of socio-cultural context [6][15] with business implications, such as [2][3][19][30].

2) User

Porter [25], taking a User-centric perspective, examines the usage lifecycle of SWAs to discuss design guidelines to handle each stage of that lifecycle. Porter explores many key user interface and usability design issues pertaining specifically to social web applications, making this an essential reference work for social web application designers. However, there is no discussion of the stages of the overall SWA design and development process itself. Bakx [5] takes a user-centric approach to building a SWA, making use of user studies at every stage of the development process up until launch.

Many academic papers have explored specific fine-grained problems within the User sphere, such as different methods of motivating and engaging users in an online community [21][28][22].

3) Technology

There have been numerous Technology-based approaches, from both academia [16][23] and industry [1][24]. The field of Web Engineering [23] looks at the process from a Technology and Business management perspective. The web application developer 37signals offers the excellent Getting Real [1] handbook for the rapid development of start-up SWAs, also focusing mainly on Technology and Business issues.

4) Business

Countless works exploring Business perspective issues have been written [4][18]. Kawasaki [18] describes some basic guidelines for forming a start-up company. Such works are not specific to web application development and thus focus their discussion mostly on Business sphere issues. Other works [2][3][19][30] explore new business opportunities and cultural implications presented by SWAs and other new technologies, providing some Business sphere perspective for design considerations.

Note that the R4 framework provides a means of categorizing and relating focused approaches such as those discussed above. Such approaches can then be plugged into the framework to develop a holistic methodology (discussed in section IV).

B. Lifecycle Related Work

Most of the works cited above do not investigate the lifecycle stages of the overall design and development of the web application. Many lifecycle models for specific spheres or overlapping spheres, however, have been presented. Porter [25], for example, describes the (outer User) usage lifecycle. Gartner's Hype Cycle [12] lies in the overlap of Business and Technology, describing the adoption and business application of emerging technologies.

There has been much work done on the initial design development (up to launch) of web applications [20][26][27]. However, it is now widely recognized that web application development differs significantly from traditional software development in that web applications continue evolving throughout the lifecycle past launch [1][23][24]. In particular, SWAs characteristically evolve much faster with very short release cycles [1]. Many SWAs have daily release schedules, some even every few hours [24]. SWAs thus essentially remain

in development for their entire lifecycle, not only just before the initial launch.

While the rapid develop-and-deploy nature of SWAs is now widely recognized, the form such development takes and the different design and development stages and challenges that arise as the application matures beyond its initial launch are as yet not well defined. In addition, the dynamics of SWAs require a consideration of the cultural context and the user base (e.g. what kind of users and how many are to be managed or courted at different stages of development) as well as competition and partnerships with related applications and services. The RRR Lifecycle model's Realization and Reformation Stages are an early attempt to define these areas.

C. Frameworks and Holistic Approaches

While holistic approaches to SWA design and development are rare, such a multi-perspective understanding is not as novel in other knowledge domains. The influential sociologist Duncan discussed cultural, behavioral and ecological perspectives to social organization in [10]. In arguing for an ecological approach, he makes use of the "referential concepts" population, environment, technology and organization, which are reflected in R4's User, outer Culture and outer Business, Technology, and inner Culture spheres, respectively.

R4's 4Sphere conceptual model itself was inspired in part by Integral Theory's AQAL (all quadrants, all levels) Model [33]. AQAL's lines and levels of development for each quadrant are also conceptually similar to lifecycle models in each of the four spheres. The categorizing relationships of AQAL inspired R4's categorization of inter-sphere relationships as well.

Bourgeois et al. [7] apply Information System Design Theory (ISDT) to the design and development of a SWA. ISDT focuses on two main aspects: design process and design product, which are similar concepts to our characterization of spheres as *design* (User and Culture) and *development* (Technology and Business) spheres, respectively. However, there is no explicit discussion of different spheres of influence as in R4. The authors use Preece's CCD as their base methodology within their ISDT approach; noting, as we have, that CCD "…was only effective at developing the initial design" and that "…a more complete [methodology] would include the concept of designing for redesign and the understanding that the system would be changing substantially over time." The continuously evolving nature of SWAs and the concept of redesign are exactly what we aim to reflect in our RRR lifecycle model. Bourgeois, et al. use the CCD methodology to inform development within their framework. In R4 we similarly adopt sphere-specific methodologies. However, we use various sphere-specific methodologies to construct a holistic methodology within the framework (as described in section IV) while Bourgeois et al. select only one domain-restricted methodology.

Drawing from Information Systems literature, WISDM [31] recognizes multiple perspectives (Technical, Organizational and Personal, which correspond respectively to R4's Technology, Business and User spheres) that inform the development process, similar to R4. However, there is no prescriptive description of how this is to be done. There are no guidelines for developers to incorporate these perspectives during the development process. The WISDM development process itself consists mostly of inner Technology and inner Business issues, as well as HCI design (incorporating outer User). WISDM does not incorporate as wide a scope of issues as R4 (i.e. outer Technology, outer Business, inner User and the whole Culture sphere issues are not considered) in the development process. This is in part a result of the different application domain assumptions of the two approaches. WISDM offers a general framework for web application development, assuming large organizations that have clearly defined and straightforward web application design objectives (e.g. the case study for WISDM was a website providing beverage consumption data for marketing purposes). The R4 framework specializes in social web applications that characteristically embody a larger area of concern due to the inherent unpredictability and dynamicity of a web application centered around its user's interactions and community. We consider social web applications to be continuously evolving with a feedback cycle between the design and development spheres. Also in WISDM, there again is no explicit discussion of the lifecycle stages of the design and development process itself, nor is there any expression of the relationships between different perspectives.

III. THE R4 FRAMEWORK

As discussed, there are different domains of interest in the design and development of SWAs. These domains reflect numerous overlapping and conflicting concerns to manage and relate. These concerns might include: choosing a culturally relevant purpose, identifying a user base, growing the user base, conducting user studies, formulating a business plan, selecting hardware and software platforms, managing the software development process, creating an appealing graphic design, designing a satisfying user experience, seeking funding, finding the right collaborators, understanding the competition, managing the community, growing the company or organization at the right pace and so on. Our survey of the related work suggests certain requirements for a holistic design and development framework that would be able to categorize and relate these issues. We propose that any such holistic framework fulfill at least these following requirements:

- It must **simplify the complexity** of design and development by presenting simple and intuitive conceptual tools with which to categorize and manage individual issues.

  While there are an overwhelming number of SWA tasks and issues to manage, they fall into general categories. These categories should be fine-grained enough to make meaningful categorizing distinctions, yet course-grained enough to be a simple and intuitive tool to use for practical design and development.

- It must be **comprehensive**.

  A holistic framework must have a means of representing or categorizing every issue pertinent to the design and development processes.

- *It must be able to **express relationships** among overlapping and conflicting issues and provide means of coordinating them.*

  While individual domains of interest are individually well developed and understood, the relationships among them are not. A framework that expresses these relationships would be of practical value to designers trying to juggle overlapping or conflicting concerns.

- *It must **reflect the co-evolving and dynamic nature** of social web applications.*

  Different stages in the lifecycle of a social web application present different challenges and opportunities. A comprehensive framework must be able to articulate these different stages.

The R4 framework consists of two main components that were designed with these requirements in mind: The 4Sphere conceptual map and the RRR lifecycle process.
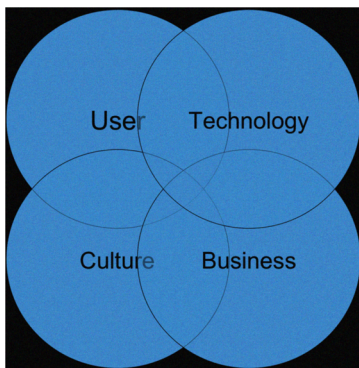

Figure 1.   The four spheres of 4Sphere: Culture, User, Technology and Business

A.   4Sphere Definitions

4Sphere describes four essential perspectives or spheres that must be considered in the SWA development process (figure 1):

- Culture
- User
- Technology
- Business

Intuitively, the Culture sphere *defines* the social web application and its context. The User *experiences* the application. The Technology *implements* it and the Business *supports* it.

Each sphere has an inner and outer dimension that helps inform the social web application development process and focuses the corresponding domain knowledge into specific areas relevant to real world development (figure 2):

- The **inner Culture** sphere (community) describes the purpose and goals of the community and the emergent properties of the community of users.

- The **outer Culture** sphere (socio-cultural context) describes the socio-cultural context of the web application, i.e. how it fits into the culture and into user's lives.

- The **inner User** (user experience) sphere describes the psychological motivation and experience of the user.

- The **outer User** (user activity) sphere describes the user's behavior and activity within the application.

- The **inner Technology** (software) sphere describes the software functionality and implementation.

- The **outer Technology** (system) sphere describes the device platforms, available infrastructure and system metrics and analysis.

- The **inner Business** (organization) sphere describes the management of the project and the organization.

- The **outer Business** (market positioning) sphere describes marketing and the competitive environment.
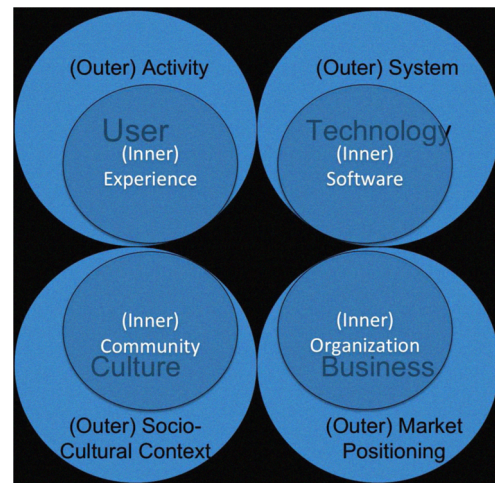

Figure 2.   The inner and outer dimensions of 4Sphere

B.   4Sphere Relationships

In 4Sphere, we use the following characterizing distinctions to describe relationships among the spheres. The left-right and upper-lower characterizing features describe dependency relationships (figure 4), while overlaps describe synergies or potential conflicts (figure 3).

The User and Culture are design-centric spheres (we design the User experience within a Cultural context) while the Technology and Business spheres are development-centric spheres (the design of the left-hand spheres is realized by the implementation and support of the Technology and Business spheres, respectively).

The lower spheres (Culture and Business) define the context of the upper spheres (User and Technology), that is, the kind of activities and interactions that Users perform given the Technology implementation.

These dependency and overlapping sphere relationships suggest general guidelines for relating various issues. Following these guidelines, we can define more fine-grained relationships among specific tasks when constructing a methodology.
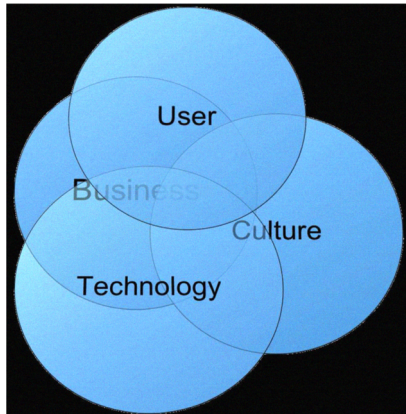

Figure 3. Overlapping Relationships: every combination of overlapping relationships is possible

1) Overlapping Relationships

Overlapping relationships among spheres result for issues that are fundamentally governed by more than one sphere (figure 3). An example of an overlapping relationship would be the design and development of the user interface. The UI must be designed with the technological implementation in mind and the implementation must reflect the design of the intended user experience. In addition, the graphic design of the UI must be consistent with the overall intended user experience as well as the specific software implementation. If there are different individuals or teams responsible for the design and implementation, an overlapping area of concern such as this requires that they work in close collaboration. If a single team or individual is responsible for both, they must consider the implications of both perspectives during the UI design and implementation process. These general guidelines are extended to any issue that lies in an overlapping area. Thus, for such a given issue, each overlapping perspective must be considered during the relevant process.

2) Dependency Relationships

The lower spheres describe a contextual dependency relationship for the upper spheres' interaction (figure 4). That is, they define the space in which the main interactions between users and implemented application itself take place. This is in part due to the fact that the outer Culture and outer Business spheres are extrospective: they relate the social web application to the outside world. The outer Culture sphere relates the application to the larger cultural context (e.g. How is the

application understood in terms of the cultural consciousness at large? Does it enhance existing activities or displace them?). The outer Business sphere looks at the market conditions (e.g. What are competitors or partners doing in this area?) and provides means of changing the relationship the application has with the outside world (e.g. through marketing and partnerships). The core application interactions between users (User) and the application (Technology) thus take place within the context of this Business and Culture sphere-mediated environment (in other words, interactions can not take place out of the space defined by the Business and Culture).
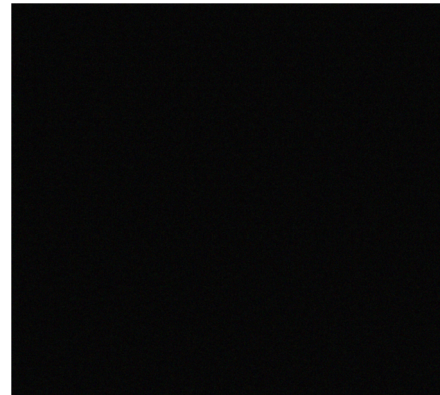

Figure 4. Dependency Relationships: lower spheres are contextual, upper spheres describe interactions. Left-hand spheres are design-centric, Right-hand spheres are development-centric

To give a simple example, a user cannot pay her parking ticket on a photo-sharing site, that is an interaction that takes place outside of the space defined by the application's Business and Culture spheres. In general, the interactions in the User and Technology spheres should "fit" the relevant well-defined Business and Culture context, neither lacking any key, nor providing any extraneous, interactions.
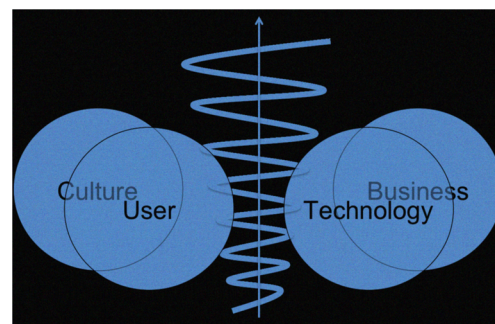

Figure 5. The design/development spiraling feedback loop

The dependency relationship between design (User, Culture) spheres and development (Business, Technology) spheres across the R4 vertical axis is characterized by a

feedback loop (figure 5). For example, an initial design for a social web application may be hatched given a specific purpose or goal (inner Culture) and refined by identifying the users and their relevant activities (User) and the cultural context (outer Culture). The development process then takes the initial design and implements a prototype implementation (inner Technology) using the available resources and personnel (inner Business) and considering the platforms (outer Technology) and market (outer Business). These circumstances in turn must be reflected back in the design. In addition, the design should also be updated using observations (metrics or user studies) of how users actually employ the resulting application and what kind of community (Culture) results. These design changes must then be realized in development, and so on up the spiraling feedback loop.

C. RRR Lifecycle Process

Each of the spheres goes through its own lifecycles. Porter, for example, describes the (outer) User lifecycle in [25]. Many works in software engineering describes the inner Technology development lifecyle [16]. There are also lifecycles that map the overlaps of different spheres. Gartner's Hype Cycle [12], for example, lies in the overlap of Business and Technology, describing the adoption and business application of emerging technologies. The progressive feedback cycle between design and development spheres described in the previous section also suggests the skeleton of a lifecycle model, which is what we base our RRR Lifecycle process on (figure 6). The design and development cycle of social web applications goes through the following progressive stages, each with their own characteristics.
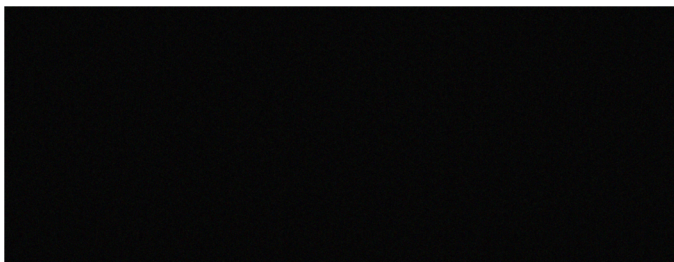

Figure 6.   The RRR Lifecycle Stages

The (Re)Design stage consists of Conceptualization, Prototyping and Development sub-stages. The (Re)Design stage describes the initial design and development of SWA up until launch.

- In **Conceptualization**, the developers formulate and refine their ideas for their application by looking at it from each of the four perspectives.

- In **Prototyping**, the developers test and refine their ideas from each perspective.

- In **Development**, they ready their application and business for launch.

The Realization stage consists of Launch, Courtship and Redefinition. The designers realize their SWA vision as refined by the real-world use of their niche users.

- In the **Launch** sub-stage, the developers launch their application and business and work out the initial kinks.

- In the **Courtship** phase, they actively recruit niche groups of users for their application.

- In the **Redefinition** phase, they make design changes to the application to respond to how users actually use it.

The Reformation stage consists of Expansion, Adaptation and Reformation. In the Reformation stage, the SWA has gone mainstream.

- In the **Expansion** stage, the application courts it's general target audience beyond the initial niche user-base to reach critical mass.

- In **Adaptation**, the developers make design changes to adapt to the increased size and diversity of their user-base. They also take advantage of their size to leverage new business strategies such as partnerships or even branching out into related SWA services.

- Finally, in **Reformation**, the social web application is completely redefined to stay relevant to changing contextual (Culture and Business) conditions.

The lifecycle process is not a strictly linear progression. The feedback cycle relationship between design and development exists to some extent among and within both the major stages and their sub-stages (hence the first major stage is named (Re)Design rather than simply Design). Thus, for example, the purpose of the Prototyping stage is to test out the Conceptualization stage design to further refine and rethink the design decisions. Indeed, the Redefinition and Adaptation and Reformation sub-stages in the subsequent major stages refer back to the original (Re)design stage and processes.

The time-scale of the major stages also increases with each stage. (Re)Design may be a matter of only a few months. Realization may be a few months to a couple years. And Reformation may be several years, if reached at all.

The RRR Lifecycle model simply gives an outline of what must be done at each stage for each sphere. We discuss specific strategies and tasks within each stage in section IV.

IV.   DESIGN GUIDELINES AND METHODOLOGY

As discussed in section II, the R4 framework itself suggests seed design guidelines for a holistic approach to social web application design and development that are summarized below. A more detailed set of guidelines can be derived from these seed guidelines according to the needs of individual methodologies and projects.

1. *Ensure the design and development feedback cycle is a co-evolutionary one.*

   Changes in one sphere may have repercussions in any of the other spheres. Ensure that the effects of changes in one sphere are appropriately propagated to all the other spheres. In addition, ensure that the development spheres accurately implement (Technology) and support (Business) the design spheres.

2. *Ensure that interactions "fill" and fit" the context.*

   Interactions as described in the User and Technology spheres should be relevant and appropriate to the goals and context of the Culture and Business spheres. In other words, there should be enough functionality to fill the bounds of the context, but no extraneous functionality that goes beyond that context.

3. *Coordinate all perspectives in overlapping areas.*

   Consider the most relevant spheres for any given issue. Overlapping areas require careful coordination of the involved perspectives.

A. The Lean & Mean Methdology

The R4 Framework provides a basic structure and guidelines for social web application development. The actual methodology used within that structure depends on the relevant goals and contexts of the individual project as well as the values and priorities of the developers. Here we present an example methodology called Lean & Mean designed for small start-ups with limited resources who want to release and grow their application quickly and efficiently. We construct our Lean & Mean methodology by using R4 to organize and guide our intuition and experience, creating relevant tasks for each combination of RRR stage and 4Sphere inner and outer sphere. We also use existing methods and techniques from each of the four spheres and attempt to resolve potential conflicts between different methods.

Presently, the construction of a methodology involves selecting and adapting the most appropriate source techniques for each sphere (although this is true for the (Re)Design stage, significant work remains in fleshing out the later stages). However, as social web applications continue to move more and more from the desktop to pervasive computing environments, we expect significant new challenges to arise in the social web application design and development process. R4 can be a useful tool in helping to adapt existing techniques and in helping to identify and suggest solutions to new challenges in this imminent social web application space. Our methodology looks towards pervasive computing environments by, for example, considering the networking infrastructure and device platforms on which it will be deployed.

We employ core techniques from the literature to form the basis of our example methodology. For the Culture sphere, our primary source was Preece [27] adapted for our Lean & Mean values of rapid and efficient development (i.e. we rely more on prototyping rather than detailed design documents as artifacts for each stage). For the User sphere, we drew extensively from Porter [25], for Business Kawasaki [18]. For the Technology sphere we used 37 signals [1], which was well-aligned with and an inspiration for the values and priorities of our Lean & Mean methodology.

Listed below are a few example tasks from the methodology in the first sub-stage of the (Re)Design stage, Conceptualization. This list is not complete, but illustrative of how R4 can be used to create a holistic methodology. The tasks in italics were derived using R4 by considering the present RRR stage (Conceptualization) along with the specific 4Sphere perspective in question. The starred task designates an overlap. Following our design guidelines, we begin Conceptualization with the design spheres (guideline 1). The Culture sphere tasks help provide the context for the User tasks to "fill" and "fit" (guideline 2).

- Culture (defining)
  - o Inner (community):
    - Define the purpose of the SWA community [27]
  - o Outer (socio-cultural context):
    - *Define potential cultural concerns with the SWA (e.g. privacy, potential disturbances resulting from the SWA intruding into the deployed environment)*
- User (experiencing)
  - o Inner (user experience)
    - Identify users [27] (i.e. user types and target userbase)
  - o Outer (activity)
    - Activities, Objects, Features (AOF) method [25] (i.e. identifying user tasks)

With some initial design ideas sketched out, we continue the Conceptualization stage with the development sphere tasks.

- Business (supporting)
  - o Inner (management)
    - Refine purpose into defining mantra [18]
    - *Define business values*
  - o Outer (market positioning)
    - Develop initial business model [18]
    - *Differentiate SWA from the competition*
    - *Choose networking infrastructure* (overlap, see outer Technology sphere)*
- Technology (implementing)
  - o Inner (software)
    - Reduce feature-set to essentials [1]

- *Sketch and experiment with paper prototypes (or other rapid prototyping methods)*

  o Outer (system)

  - *Choose networking infrastructure* (overlap, see outer Business sphere)*

  - Choose device platforms [27]

The overlapping task (denoted by a *) requires the investigation of available infrastructure from both Business and Technology sphere perspectives (as per guideline 3). Pervasive technologies require some form of wireless connectivity. Different forms of connectivity may be provided by many different telecoms or ISPs. Thus, the choice of business partners (telecoms or ISPs) is codependent with the choice of networking technologies and protocols (Internet, cellular network, Bluetooth, etc.). Such overlapping relationships are naturally revealed when determining the correct spheres for specific tasks within R4.

Dependency relationships between design and development spheres are also illustrated above. For example, the outer Business task of developing an initial business model and both of the inner Technology tasks of choosing essential features and paper prototyping are clearly dependent on the purpose of the community in the Culture sphere and the identification of users and AOFs [25] in the User sphere.

After the first time going through the tasks of the Conceptualization stage, the initial concept can be refined by returning to the design sphere tasks and updating them. For example, the choice of networking infrastructure in the development spheres may impact all the tasks within the design spheres, requiring changes. In turn, the development sphere tasks can then also be updated depending on the design sphere changes and so on up the spiraling design-development sphere feedback loop. The designers can continue this refinement process until they are satisfied enough with their concept to begin building software prototypes. The design and development categorizations thusly clarify the relationships between any given pair of tasks, simplifying and directing the process of propagation of changes in any one task to dependent tasks.

V. DISCUSSION AND FUTURE WORK

To demonstrate and refine the R4 framework and to demonstrate the construction of a methodology, we plan to develop a SWA for local, cultural social scenes, such as local music and art scenes, called localive. With localive, users can browse scenes by location and genre and be notified of events and scenes according to their own preferences. The application is designed to support and enhance existing local scenes and enable geographically diverse scenes to communicate with each other over genres or other common interests. We are currently finishing up the early design stages and are beginning to develop a prototype for initial user testing.

We are confident that 4Sphere is a useful and comprehensive conceptual map, and our initial experiences with 4Sphere have so far been consistent. As for the RRR Lifecycle, the (Re)Design stage is well-defined in each of the four spheres. However, the later stages of RRR lifecycle, Realization and Reformation, are not as well-defined in the related work. R4 can help organize the appropriate tasks and methods with which to comprise a methodology, but they require long-term or extensive case studies to describe in detail. We plan to research case studies for long-running web applications to refine and flesh out these important stages.

VI. CONCLUSION

In this paper, we have argued that a holistic design and development framework for SWAs is required to manage and relate all the concerns and issues relevant to the design and development process. We suggested requirements for such a framework. We introduced R4, a comprehensive framework for SWA development, based on these requirements. R4 consists of 4Sphere, a conceptual map of four main perspectives, and the RRR Lifecycle model. We have provided basic guidelines for design and development based on our framework and its relationships and shown how to use the R4 framework to construct a specific methodology. We believe that holistic frameworks that articulate the vision of R4, incorporating and relating the major perspectives and interests involved in the design and development of SWAs over their entire lifecycle, will be important theoretical and practical tools for the understanding of future SWAs.

REFERENCES

[1] 37signals, Getting Real: The smarter, faster, easier way to build a successful web application. 37signals, 2006, http://gettingreal.37signals.com/

[2] C. Anderson, The Long Tail: Why the Future of Business is Selling Less of More, Hyperion, 2006

[3] C. Anderson, Free: The Future of a Radical Price, Hyperion, 2009

[4] W. Baars, Project Management Handbook. Ver 1.1, The Hague, Netherlands: Data Archiving and Networked Services, July 2006, http://www.projectmanagement-training.net

[5] I. Bakx, "User-Centred Design Approach for a Community Website with Social Software," Proceedings of Human-Computer Interaction International, Springer, July 2007

[6] A.-L. Barabasi, Linked: How Everything Is Connected to Everything Else and What It Meas for Business Science and Everday Life. New York: Penguin Group, 2003

[7] D. T. Bourgeois and T. A. Horan, "A Design Theory Approach to Community Informatics: Community-Centered Development and Action Research Testing of Online Social Networking Prototype," Journal of Community Informatics, Vol. 3, No. 1, 2007

[8] D.M. Boyd and N.B. Ellison, "Social Network Sites: Definition, History and Scholarship," Journal of Computer-Mediated Communication, vol. 13, issue 1, article 11, 2007

[9] W. Chin, E-Dreams (documentary film). Cav Distributing, 2004

[10] O. T. Duncan, L. F Schnore, P. H. Rossi, "Cultural, Behavioral and Ecological Perspectives in the Study of Social Organizations," The American Journal of Sociology, Sep. 1959

[11] N. Eagle, A. Pentland, "Social serendipity: mobilizing social software," IEEE Pervasive Computing, Vol. 4, no. 2, pp. 28-34, Jan.-Mar. 2005

[12] J. Fenn and M. Raskino, Mastering the Hype Cycle. Cambridge, MA: Harvard Business Press, 2008

[13] D. Fisher, T.C. Turner, M. Smith, "Space Planning for Online Communities," Proceedings of the Second International Conference On Weblogs and Social Media, March 2008

[14] K. German, "Top 10- dot-com flops", http://www.cnet.com/1990-11136_1-6278387-1.html

[15] M. Gladwell, The Tipping Point: How Little Things Can Make a Big Difference. New York: Little, Brown and Company, 2000

[16] C. Ghezzi, M. Jazaveri, D. Mandrioli, Fundamentals of Software Engineering. 2nd Edition, Prentice Hall, 2002

[17] C. Hegedus and J. Noujaim, Startup.com (documentary film). Artisan, 2001

[18] G. Kawasaki, The Art of The Start. Portfolio, 2004

[19] K. Kelly, New Rules for the New Economy: 10 Radical Strategies for a Connected World, Viking Adult, 1998

[20] A. Jo Kim, Community Building on the Web. Berkeley, CA: Peachpit Press, 2000

[21] K. Ling et al., "Using social psychology to motivate contributions to online communities," Journal of Computer-Mediated Communication, vol. 10, issue 4, article 10, 2005

[22] M. Ma and R. Agarwai, "Through a Glass Darkly: Information Technology Design, Identity Verification, and Knowledge Contribution in Online Communities," Information Systems Research, vol. 18, no. 1, pp. 42-67, 2007

[23] S. Murgesan and A. Ginige, "Web Engineering: Introduction and Perspectives," in Web Engineering: Principles and Techniques, Ch. 1, Idea Group, 2005

[24] J. Musser and T. O'Reilly, "Web 2.0: Principles and Best Practices," White Paper, O'Reilly Radar, 2006, http://oreilly.com/catalog/web2report/chapter/web20_report_excerpt.pdf

[25] J. Porter, Designing for the Social Web. Berkeley, CA: New Riders, 2008

[26] D. Powazek, Design for Community: The Art of Connecting Real People in Virtual Places. Corte Madera, CA: Waite Group Press, 2001

[27] J. Preece, Online Communities: Designing Usability and Supporting Sociability. Hoboken, NJ: Wiley, 2000

[28] A. Rashid et al., "Motivating participation by displaying the value of contribution," Proceedings of the SIGCHI conference on Human Factors in computing systems, 2006

[29] H. Rheingold, The Virtual Community: Homesteading on the Electronic Frontier. Addison-Wesley, 2003

[30] D. Tapscott and A.D. Willaims, Wikinomics: How Mass Collaboration Changes Everything. Portfolio, 2006

[31] R. Vidgen, "Constructing a web information system development methodology," Information Systems Journal, vol. 12, pp. 241-261, 2002

[32] B. Wellman and M. Gulia, "Net Surfers Don't Ride Alone: Virtual Communities as Communities," in Communites and Cyberspace, P. Kollock and M. Smith, Eds. New York: Routledge, 1999

[33] K. Wilber, "Introduction to Integral Theory and Practice: IOS Basic and the AQAL Map," Journal of Integral Theory and Practice, Vol. 1, no. 1, Spring 2006

[34] http://www.amazon.com

[35] http://www.facebook.com

[36] http://www.flickr.com

[37] http://www.myspace.com

[38] http://www.twitter.com

[39] http://www.youtube.com