

“Designing for the Web” Revisited: A Survey of Informal and Experienced Web Developers

Mary Beth Rosson¹, Julie F. Ballin¹, Jochen Rode², and Brooke Toward¹

¹ Pennsylvania State University, Information Sciences & Technology
330 IST Building, University Park, PA 16802
{mrosson, jfb15, bet133}@psu.edu

² Virginia Polytechnic Institute and State University, Center for Human-Computer Interaction
3160 Torgersen Hall, Blacksburg, VA 24061
jrode@vt.edu

Abstract. We report a subset of findings from a survey of over 300 web developers – a mixture of professional and more casual developers – targeted at understanding the needs, problems and the processes that developers follow and the tools they use. The prototypical web developer from our sample is meticulous about the quality of the web sites she produces, considers usability issues but neglects accessibility concerns. Web developers have many similar interests regarding web applications or features such as authentication, databases, online surveys or forms. They value ease of use as the most important property of a web development tool but mention many other needs such as integration with other tools, strong code editing features, or WYSIWYG facilities. This report details findings regarding process, tools, quality control, and learning.

1 Introduction

The diversity within the web development community is changing rapidly. For instance the number of end users who build web applications is increasing as general computing skills become more sophisticated (e.g., through use of spreadsheets, CAD, scientific visualization, and so on). Indeed the ubiquity of the web and the resultant ease of publishing content to a huge audience has been an important element in motivating web users to learn more powerful development techniques. The rapidly expanding population of web developers presents a mixture of opportunities and challenges for researchers and engineers building web development tools.

Despite the pervasiveness of the web – and the breadth of the associated developer population – little empirical work has studied web developers as “users”. In 1998, Vora [7] surveyed professional web developers to better characterize prototypical professional web development practices. For example, Vora queried web developers about the methods and tools that they use, and the problems that they typically encounter. He summarized a number of technical problems, including web browser interoperability and usability, and lack of standards compliance of WYSIWIG editors.

In this work, we build on the Vora survey, but with the goal of reaching out to the combined population of professional and more casual web developers. The survey is done within the context of a research program aimed at analyzing and supporting the needs, problems, and preferences of “informal web developers” (individuals not explicitly trained in web programming techniques). Our sampling is intentionally biased towards these casual (nonprogrammer) web developers and therefore care should be taken when viewing the results in the context of professional web development.

2 Related Work

In general, the analysis of web developers’ needs has received only little attention in the web engineering literature. In two of our pilot studies (using both survey and interview methods), we analyzed the experiences and concerns of experienced web developers within a university context [3] [4]. These developers reported a number of problems that overlap to some extent with those report by Vora [7] and include security, cross-platform compatibility, integrating technologies (such as HTML, CSS, JavaScript, Java, SQL) and debugging of web applications.

As one phase in their user-centered design of the Denim prototyping tool for web site development, Newman et al. [1] interviewed 11 web development professionals. They found that these experts’ design activities comprise many informal stages and artifacts. Expert designers employ multiple site representations to highlight different aspects of their designs and use many different tools to accomplish their work. They concluded that there is a need for informal tools that help in the early stages of design and integrate well with the tools designers already use.

In a study of web development in a community computing context, we interviewed 12 informal web developers about how they came to be doing web development, how they acquired their skills, the kinds of projects and programming issues they encountered in their everyday development, and what concerns, if any, they had about the tools they used [5]. We found that these individuals’ development activities are situated in a collaborative context in which they depend on colleagues for content, expert advice, and testing. Their choice of tools was often based on organizational issues such as cost or who else was using the tool, rather than their own preferences or analysis of tools available. They learned new skills in an informal and as-needed fashion, often by tracking down and adapting or modeling the examples of others.

3 A Survey of Web Developers

To develop a broad characterization of the current web developer population – both professional and casual – we conducted an online survey and recruited participants from a variety of web development communities. The survey was based on our prior surveys and interviews of local web developers; it contained questions about web development experiences, including problems encountered; whether and how testing was carried out; desirable features or applications to incorporate in web development (e.g., databases, authentication); development style, including individual working style variations, and basic demographics (for a full list of the 37 questions see [6]).

We took two general approaches in recruiting participants. First, we contacted user groups associated with web tools (e.g., Macromedia, Frontpage); second we searched the web for other organizations that seemed to be oriented towards web use or even computer use in general. We particularly sought out organizations that might rely on *informal* developers (e.g., clubs or community organizations), but our survey invitation was aimed at *both* professional and casual developers.

We initiated contact with 591 organizations: approximately 30% product-centered groups (Coldfusion, Frontpage, etc.), 20% platform-centered (Mac, Linux, etc.), 38% hobby or ‘computer club’ type groups, and the remaining groups falling into language-oriented (e.g., ASP), professional/networking organizations and specific web-

sites. We sent our email invitation to the listserv contacts, asking them to forward it to their members; the email summarized the study, data security/privacy, and the drawing for cash prizes (10 prizes of \$50) used as an incentive for participation.

4 Survey Results

We received 334 responses to the survey. In this report we highlight a subset of the findings that seem especially relevant to the web engineering community (see [6] for additional results). In the following, question numbers refer to the actual position in the survey, so that interested readers may integrate the results reported here with the full survey and summary results available online. Note that percentages reported in this paper are the percentage of respondents who answered a particular question, not the percentage of the entire survey population with missing responses. Many respondents skipped one or more questions, so we follow the norm of including the relevant sample size as each percentage result is reported.

Interestingly, the answer to whether or not a respondent self-identified as a “programmer” was not often a useful grouping variable for the web activities and problems summarized here. For this reason the results discussed use the entire dataset.

4.1 Participants

The survey population included both men and women (70% and 30% respectively); most respondents (86%) reported their race or ethnicity as white/Caucasian. As one would expect, this web developer sample was relatively highly educated: 29% of respondents reported that they had completed an undergraduate degree and an even larger proportion (35%) reported completing at least some post-graduate education.

There was considerable age diversity in our sample (remember the survey’s bias towards informal web developers). Interestingly, the single largest group of respondents age-wise was those who identified as age 60 or older (21%). In combing for computer related groups to whom we wanted to promote the survey, we discovered many groups oriented towards or run by senior citizens; this may explain the large proportion of older respondents. Other respondents were spread relatively evenly across age categories of 26-30, 31-35, and so on up through the age group 56-59. Only 6% of the sample reported their age as 25 or younger.

A small majority of respondents (54.7%) reported that “work” was the most common reason for them to develop and maintain websites. This is interesting as it emphasizes that, although considerable web development is being carried out in professional contexts, there is a sizable number of projects underway for other purposes. The two next most common motives were “special interest/hobby” (16.6%) and “civic, volunteer, or community work” (12.4%).

4.2 Perceived Value of Web Functionality

One question aimed at understanding web developers’ current needs asked them to rate the perceived value of a number of predefined features (Figure 1; these items were developed through our pilot studies). As indicated in Figure 1, access restrictions, online databases, member registration systems, and online surveys/forms are

seen as particularly valuable to our respondents, all being well above the mid-point on a range from 1 (not valuable) to 5 (extremely valuable). Communication-oriented features like discussions and chat are seen as relatively less valuable.

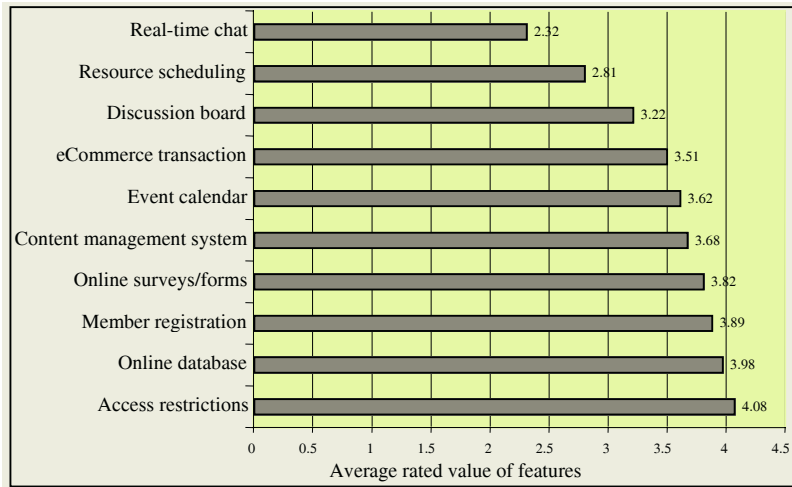


Fig. 1. Results from Question 5: “The following question asks you to judge the value of these same 10 features in your web development projects, regardless of whether you have worked with them yet or not.” (N=314 to 318)

4.3 Characterizing the Web Development Process

To gain insight into the typical web development process and attitudes towards web development, question 16 asked the respondents to rate their agreement with a series of statements (1=strongly disagree, 5=strongly agree).

Our respondents tended to agree with the statement: “I spend a lot of time making sure my site’s layout, formatting, content, and interactive elements are just right before I “go live”” (mean=4.18, SD=0.93, n=274). They voice similarly strong agreement with: “After my websites “go live”, I check back frequently to make sure that everything works like it should (links, images, forms, etc.)” (mean=4.11, SD=1.0, n=274). These responses suggest that attention to the details of a web page is high on these developers’ list of concerns.

Respondents tended to disagree with the statement: “When taking on a new web project, I immediately start constructing pages” (mean=2.43, SD=1.25, n=272), implying that they take steps to plan their project before jumping into building web pages. However the statement: “When working on a web site, I have a systematic process I follow” evoked a rather neutral response, only slightly biased towards “agree” (mean=3.56, SD=1.12, n=273). This is an area we hope to further explore in later research.

Most respondents also agreed with the statement: “As I work on a web project, I think about how I might come back later to change or expand it” (mean=4.16, SD=0.96, n=274). This is a promising result as it implies that they are planning for enhancement or other maintenance activities. See Table 1 for details.

Table 1. Question 16: statements ranked from 1 (strongly disagree) to 5 (strongly agree)

	1	2	3	4	5	Average
I spend a lot of time making sure my site's layout, formatting, content, and interactive elements are just right before I "go live"	1% (3)	4% (11)	17% (47)	32% (87)	46% (126)	4.18 (n=274)
After my websites "go live", I check back frequently to make sure that everything works like it should (links, images, forms, etc.)	0% (1)	7% (20)	20% (56)	24% (67)	47% (130)	4.11 (n=274)
When taking on a new web project, I immediately start constructing pages	29% (79)	28% (77)	20% (55)	15% (41)	7% (20)	2.43 (n=272)
When working on a web site, I have a systematic process I follow	2% (5)	18% (50)	28% (77)	26% (70)	26% (71)	3.56 (n=273)
As I work on a web project, I think about how I might come back later to change or expand it	1% (4)	5% (13)	16% (45)	31% (86)	46% (126)	4.16 (n=274)

Question 15 was, in part, targeted at the issue of code reuse and participants were asked to rate how often particular statements are true (1=hardly ever; 5=quite often). The statement "I consult and reuse/copy code I have previously written myself" received a relatively high rating (mean=3.90, SD=1.36, n=273). This can be contrasted to their ratings for reusing others' code: "I search the web for snippets of code that I can directly copy, paste and edit" (mean=3.01, SD=1.33, n=273).

4.4 Web Development Tools

Question 6 asked: "What is the primary development tool you use for working on your site(s)?" 42.1% of the respondents cited Macromedia Dreamweaver. Microsoft Frontpage tied with HTML editors (Bbedit, Homesite etc.) at 12-13% each, followed by Text editors such as notepad or vi with 9.7%. No other tool exceeded 3%. Note that the relatively high proportion of Dreamweaver users is likely biased by our recruiting strategy (the Macromedia user groups were large and had good response rate). Of course, this predilection for Dreamweaver should also be considered when interpreting responses to questions concerning tool likes and dislikes.

Question 8 asked: "What are the three things you like MOST about your primary web development tool?" Three open response fields were provided and we received 286 responses for the first, 272 for the second, and 246 the third – a total of 804 individual responses, typically just a few words long. We coded the results by first scanning all responses and establishing categories. Next, we coded all comments according to the previously established categories. Fig. 2 visualizes about 90% of grouped comments (719 responses). 10% of developers' comments were coded as "other" because they were too diverse to be grouped in a meaningful fashion.

Question 9 asked: "What are the three things you like LEAST about your primary web development tool?" Again, three open response fields were provided and we received 259 responses for the first, 193 for the second, and 143 the third, for a total of 547 individual responses (excluding 48 responses such as "nothing" or "n/a").

We used a similar coding strategy, resulting in 16 categories. Not surprisingly, many of the comments made in response to things liked least can be seen as the inverse versions of things liked most (e.g., the number one group in both cases is related to the rather general evaluation of ease of use). Interestingly however, while feature

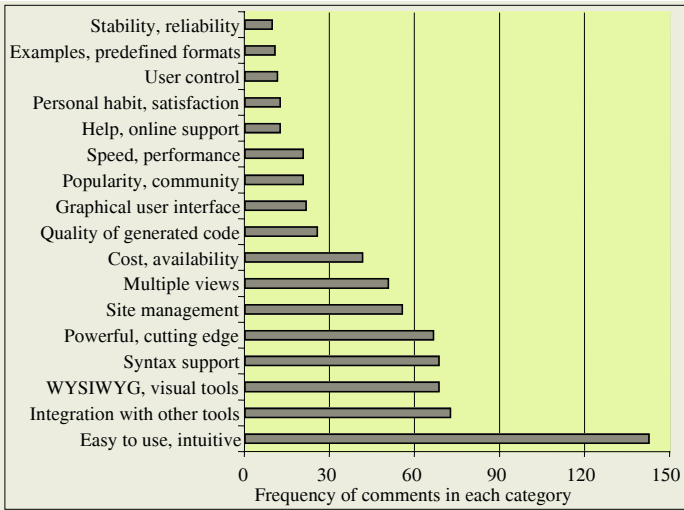


Fig. 2. 90% of responses to question 8 “What are the three things you like MOST about your primary web development tool?” were coded into 17 categories

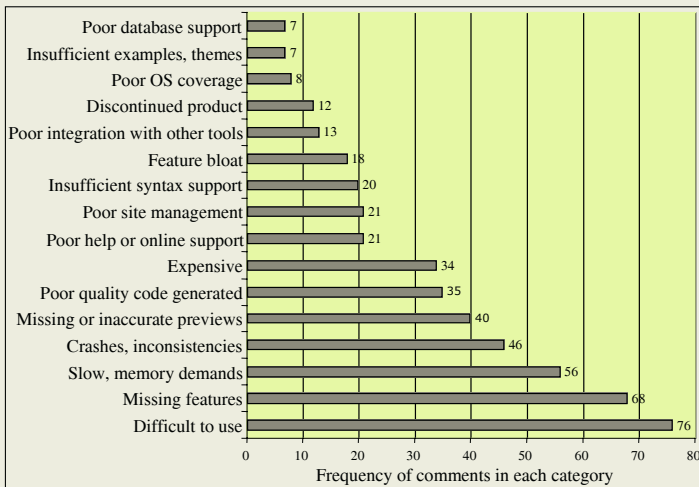


Fig. 3. 88% of responses to question 9 “What are the three things you like LEAST about your primary web development tool?” were coded into 16 categories

coverage was rarely mentioned as a reason to *like* a tool, it was the second most common category for *disliking* a tool.

4.5 Problematic Development Situations

To explore the problems that web developers may encounter we asked our respondents to rate eleven problems according to how frequently they occur. As with the

features probed in Question 5, this list of issues was based on our earlier surveys and interviews that probed problems in web development.

Fig. 4 shows the results. None of the issues stands out as a particularly frequent problem, except perhaps of “getting content in a timely manner from others...” (mean=3.32, SD=1.41, n=272). This is interesting in that it is the one issue that is very much related to the developers’ collaborative context – that is, to their dependencies on others. We had learned in our interviews with community webmasters that this was a particularly vexing problem for these relatively informal web developers [5]; it appears that it is a similar problem for a much more diverse population.

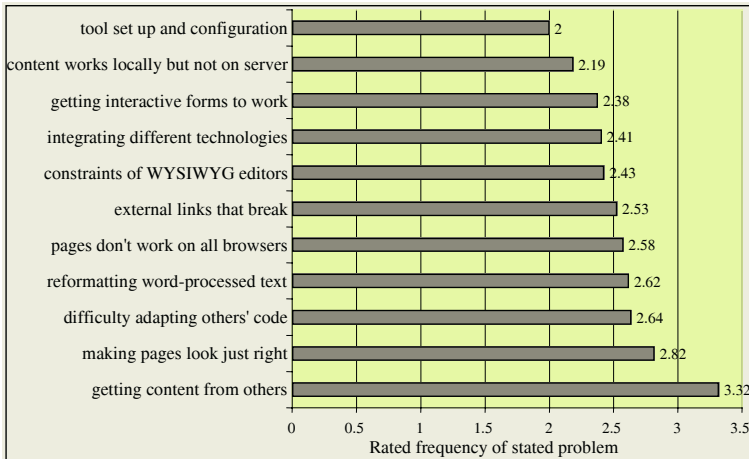


Fig. 4. Responses to Question 14 “How often do you experience problems with the following kinds of issues that sometimes arise in web development work? Please use a scale from 1 (one) to 5 (five) where 1 means hardly ever, and 5 means quite often.” (n=267 to 276)

4.6 Attention Directed to Quality Control

To understand the extent to which quality control is a concern for our sample of web developers, we asked respondents to tell us how often they performed certain testing tasks (1=never, 5=always; “When working on websites, how often do you test to make sure...”; Question 12). An overwhelming majority of respondents agreed that they evaluate the general *usability* of their websites always or almost always: “...it is easy for users to do what they want to do on the site and to find what they might be looking for (usability)” (mean=4.33, SD=0.93, n=276). However, they seem to be much less likely to worry about universal access: “Users who might have disabilities will be able to use your site (ADA compliance, section 508, Equal Access, etc.)” (mean=2.75, SD=1.41, n=276).

Although most developers appear to test for platform and browser compatibility, not all of them do so routinely (“It will work across different operating systems and different web browsers such as Internet Explorer, Netscape Navigator, Safari, etc.”; mean=3.75, SD=1.26, n=276).

The three items analyzed above represent a relatively superficial assessment of developers’ testing processes. To probe more deeply we included an open-ended ques-

tion “Please briefly describe when and how you test the websites you build or maintain”. This generated 514 comments; 415 addressed *how* testing is done and 99 *when*. We coded responses by first scanning all responses and establishing categories and then classifying comments into these categories. Some respondents did not answer the question and answers from a single respondent often contained multiple codes. 17% of responses were coded as “other” because they were too diverse to be grouped in a single category or were not specific enough. The distribution of the *how* comments is graphed in Fig. 5. The majority of responses related to browser compatibility and operating system concerns (note that we had just raised these concerns in the immediately preceding survey items). These data are also consistent with a small interview study of expert web developers where compatibility was among the most frequently cited challenges in web engineering [4].

Our prior study of community webmasters had indicated that nonprogrammers working in this context often use informal testing mechanisms (e.g., asking friends or coworkers to critique a site, [5]). Similar strategies are apparent in this larger survey: Many respondents said that the main test method they use is to simply “eyeball” the site before going live. We were surprised that only *one* respondent mentioned testing of security. This is particularly interesting given our prior interviews with web development experts who listed this as their primary concern [4]. Perhaps this is indicative of how difficult security testing is for many web developers, even though it is recognized as a central issue.

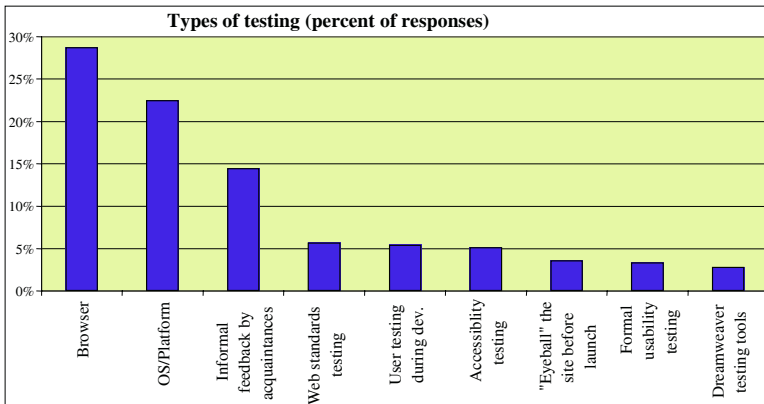


Fig. 5. Nine categories of testing *how* comments account for 83% of responses (N=415)

Our open-ended survey probe also asked for information about *when* website testing is done. We developed and applied a coding scheme (eight categories, see Fig. 6) for the content that described *when* website testing is conducted. Of the 99 responses received, 7% were coded as “other” because they were too diverse to be grouped in a single category or were not specific enough. The most common response (32%) was that testing was carried out after every change or update to the site. An almost identical number (30%) said that testing is simply conducted throughout development. Only a few mentioned a specific time interval at which they test; three respondents did volunteer that they never test their work!

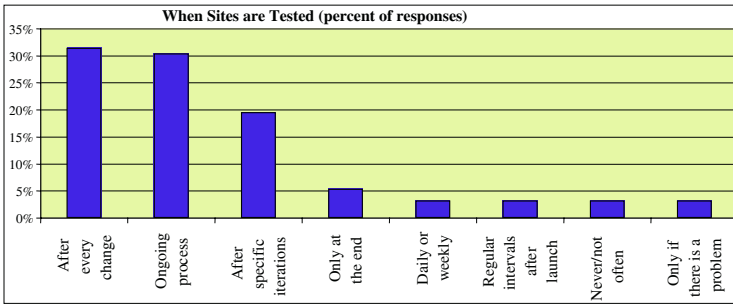


Fig. 6. Eight categories of testing *when* account for 93% of the comments (N=99)

4.7 Learning New Web Development Skills

We asked participants to rate how likely they would be to consult particular resources for assistance in case they needed to learn something new (Question 11; 1=not likely; 5=very likely). “FAQs, books, or tutorials” were rated most highly (mean=4.53, SD=0.88, n=257), followed by “Examples of similar sites from which you can get ideas and copy code” (mean=3.97, SD=1.18, n=259), and “A friend or coworker who knows how to do it” (mean=3.76, SD=1.26, n=259). Respondents indicated that they would be less likely to consult sources such as interactive software wizards, software agents, seminars, or support hotlines.

5 Discussion

Our survey yielded a diverse sample of respondents – a mixture of professional and more casual developers, representing a wide range of ages, who seem to be pursuing projects in rather different web development contexts. However, despite the variation among the respondents, there are a number of implications that we see in our results.

For example, with respect to perceived value of different web functionality, most developers rated access restrictions, online databases, survey and forms as valuable elements for their web presence. Unfortunately, many of the features and applications that developers see as valuable are not easily implemented. For casual or informal web developers, providing access restrictions may be conceptually simple and obvious, but current tools make its implementation quite challenging. One of the other highly valued features—online databases—seems to be even more difficult to implement than access restrictions. Again, although the interactions with databases may be conceptually simple (e.g., consisting of overview and detail pages, a search function and some data input and edit forms), they are typically beyond the implementations skills for casual web developers. Current web development tools do not sufficiently abstract technical concepts such as session management, input validation or URL parameter passing. This requirement underscores an opportunity to develop more powerful web development tools designed for end users, tools that would raise the ceiling on what is achievable for nonprofessionals.

Our analysis of questions about respondents’ web development process suggests that – at least in our sample – the prototypical web developer is meticulous and particular about the quality of the web sites she produces and maintains. Also, generally

our web developers seem to invest some thought before embarking on a new project rather than implementing web pages ad-hoc, although they may or may not follow a strict process. Web developers also appear to frequently reuse code they wrote earlier but only occasionally search the web for example code to copy and use. These general findings are an encouraging indication that even an increasingly diverse web developer population is attuned to the “traditional” concerns of software engineering such as design and quality assurance.

The responses to the question about features most liked in web development tools show that this sample of web developers value ease of use as the most important property of a web development tool. They also clearly appreciate a tool that integrates well with other tools and provides frequently needed site management features such as integrated file upload. While they highly regard powerful WYSIWYG visual design and code generation features, they also demand support for viewing and *editing*, testing, and previewing the code behind the scenes. They appreciate code auto formatting and tag completion but at the same time expect to have full control over the layout of hand-written code.

At the same time, the responses to the question about what developers least like about their web development tool(s) show that many web developers are still not satisfied with usability aspects of their tools. While many respondents request more powerful features, such as more extensive WYSIWYG support, others complain about feature bloat. Across all comments, concerns about performance problems and faulty behaviors take the lead in complaints about tools. Another common complaint refers to automatically generated code that appears “messy”, “bloated”, and non-compliant to standards.

Regarding the typical problems that web developers encounter we were not able to detect any major distinctions in developer’s experiences. Only the issue of “getting content in a timely manner from others...” was rated above the mid-point on a frequency rating scale. This concern is interesting, as it is much more social in nature (being dependent on a colleague for input) than most of the other concerns. It may be that social problems of this nature plague everyone, whereas the other listed problems are much more dependent on the types of applications or work contexts in which developers operate. Our future research will investigate these problematic aspects of web development more carefully, for example also probing perceived severity of individual problems, connecting problems to developers’ working context, and providing an opportunity to describe problems in an open-format question.

Questions about the quality control process show that the vast majority of developers from our sample routinely validate website usability (although we do not know about the procedures they employ and standards they maintain) and sometimes check for cross-platform issues but rarely for accessibility problems. These accessibility checks may be omitted because of lack of awareness and concern, but it may be at least partly due to the relatively tedious and time-consuming tool support for such checks (too verbose, reporting many false positives; lack of automation).

6 Future Work

The results from this survey paint a high-level picture of today’s web developers, their needs and habits. In a more detailed fashion we hope to investigate the specific

processes that informal and professional web developers follow for planning, development, testing and debugging. Another dimension worthy of investigation is the collaborative aspects of web development – with respect to general problems, the most frequently reported problem was obtaining content from other people, which reinforces the importance of the organizational context in which web work is done.

Another direction for future work is to refine our analysis of professional versus casual developers, so as to better distinguish among their experiences and concerns. Our high-level contrast of developers who do or do not self-identify as a programmer did not prove to be an important categorical factor in their responses. However, there are a number of other issues related to this self-judgment (e.g., the projects a developer undertakes, the context in which a project takes place, the training the person has received); these variables may combine in complex ways to reveal subgroups within this diverse community. If we can identify such distinctions, we may be able to create a more refined picture of the habits and needs – and develop the corresponding tools – for the large and growing population of web developers.

References

1. Newman, M., Lin, J. Hong, J., Landay, J. (2003). DENIM: An Informal Web Site Design Tool Inspired by Observations of Practice. *Human-Computer Interaction* 18: 259-324.
2. Pitkow, J. and Kehoe, C. (1998). Gvu 1998, 10th WWW User Survey. http://www.cc.gatech.edu/gvu/user_surveys/survey-1998-10/.
3. Rode, J. and M.B. Rosson (2003). Programming at Runtime: Requirements & Paradigms for Nonprogrammer Web Application Development. *IEEE HCC* 2003.
4. Rode, J., M.B. Rosson, M. A. Pérez-Quñones (2002). The challenges of web engineering and requirements for better tool support. *Virginia Tech Computer Science Tech Report #TR-05-01*.
5. Rosson, M.B., J. Ballin, H. Nash. (2004). Everyday programming: Challenges and opportunities for informal web development. *IEEE VL/HCC 2004*: 123-130. New York: IEEE.
6. Rosson, M.B., J.F. Ballin, J. Rode (2005). Survey of Informal and Experienced Web Developers. Results summary. <http://eudweb.cs.vt.edu/webdevelopersurvey/summary.html>
7. Vora, P. (1998). Designing for the Web. *ACM interactions*, 13-30.