Learning Web Development:
Challenges at an Earlier Stage of Computing Education

THOMAS H. PARK
thomas.park@drexel.edu

SUSAN WIEDENBECK
susan.wiedenbeck@drexel.edu

The iSchool at Drexel
College of Information Science and Technology

INTRODUCTION

Given the increasing importance of a computationally literate citizenry, educators, researchers, and policy-makers alike are striving to broaden participation in computing. Web development offers potential in this regard by providing an platform for introducing more elementary aspects of computing [4] and by situating computation in personally and socially meaningful contexts [2].

The goal of our study [5] was to better understand *how* web development can effectively serve an earlier stage of computing education. Our approach was to identify aspects of basic web development that are challenging to students who lack a programming background. Our research questions are as follows:

RQ1. What are the challenges that non-CS students encounter in an introductory web development course?

RQ2. What are the challenges that non-CS students encounter specific to manipulating code?

RQ3. How do these challenges relate to computational concepts?

METHODS

Our study focused on an introductory web development course offered to Master's of Library and Information Science students. Forums were available where students could seek help from classmates or the instructor. We operationalized challenges as the impetuses for seeking help on the forums, and collected each post's body, author, title, and time.

Two sections of the course, taught by the same instructor fall 2008 and fall 2009, were used for this study. These sections comprised 49 students authoring 747 posts.

We applied two levels of content analysis [3] to this dataset, first deriving broad categories of challenges reported in the help forums, and then a subset of development challenges pertaining specifically to the manipulation of HTML, CSS, and JavaScript code. To identify computational concepts underlying these challenges, we analyzed the dataset for relevant themes [1].

www.ischool.drexel.edu

RESULTS

Category	Description
Instruction	Asking questions about curriculum, instructions, and assessment
Content	Collecting, creating, and editing text, images, and multimedia
Design	Planning information architecture and visual design
Development	Creating and manipulating HTML, CSS, and JavaScript code
Technology	Selecting, installing, and configuring technology
None	Sharing general information and providing help

Category	Description
Links	Creating links to other resources
Images	Embedding images
Image Maps	Creating image maps
Tables	Creating tables
Lists	Creating lists of items
Forms	Creating forms with input elements and actions
Frames	Creating framesets or inline frames
Backgrounds	Setting background colors, images, or tiling
Fonts	Setting font styles
Layout	Positioning and aligning elements
Functions	Modifying functions, attaching event handlers
Objects	Instantiating objects
Source Files	Managing source code at the file level

The tables above summarize the codes resulting from the content analysis. On the left are broad categories of challenges, and on the right are challenges specific to development. The corresponding figures below illustrate how the help-seeking instances were classified using these codes.
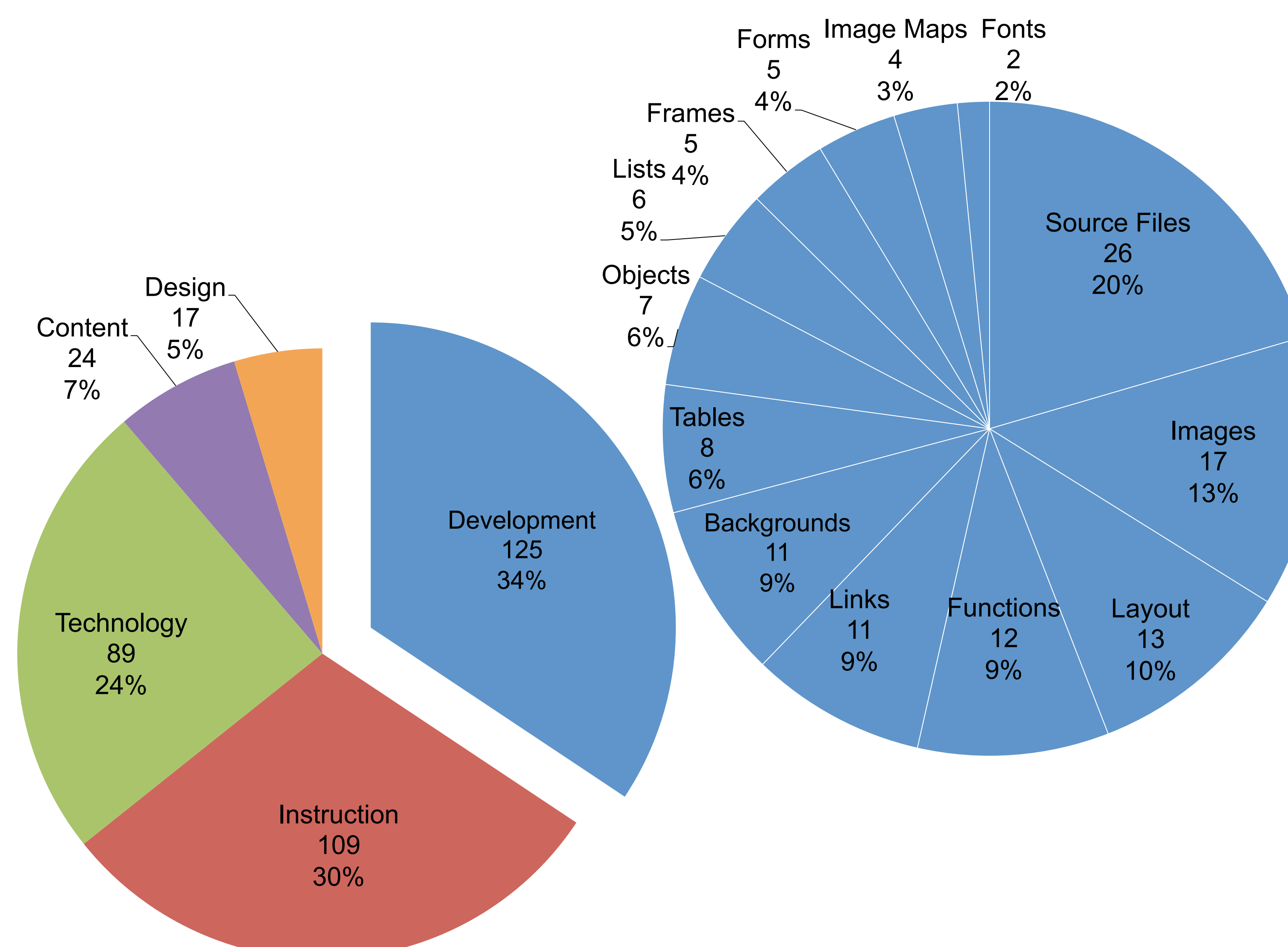
DISCUSSION

Our findings hold several implications for computing education.

Despite the motivational effect of a contextualized computing course, there are several tradeoffs in a web development context. Most significantly, many technical difficulties occurred while students were tasked with configuring shell accounts, home machines, and web servers. Although challenges can serve both as learning barriers and opportunities, students reported that these issues detracted from the main purpose of the course and contributed to feelings of frustration. Tradeoffs also manifested in complaints about the lack of adherence to web standards on certain topics and concerns about privacy and intellectual property laws.

Though many web development courses "sprint" toward programming, our results lend support for a basic course that is unapologetically dedicated to more elementary computing. First, a large proportion of challenges experienced by non-CS students were caused by basic topics such as links and images. Second, students demonstrated a low level of engagement and understanding of JavaScript topics such as functions and objects, which is not surprising given the time constraints under which they were introduced. Finally, we identified a number of computational concepts underlying the challenges, including *notation*, *hierarchies and paths*, *nesting*, *networks*, *parameters and arguments*, and *decomposition and abstraction*. These represent opportunities to educate students about computing through basic web development, setting the stage for future learning.

REFERENCES

[1] Braun & Clarke. 2006. Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3, 2, 77-101.
[2] Forte & Guzdial. 2005. Motivation and non-majors in CS1: Identifying discrete audiences for introductory computer science. *Transactions on Education*, 48, 2, 248-253.
[3] Krippendorff. 2004. *Content Analysis: An Introduction to Its Methodology*. Sage Publications, Thousand Oaks, CA.
[4] Lu & Fletcher. 2009. Thinking about computational thinking. In *Proc. of SIGCSE*, 176-180.
[5] Park & Wiedenbeck. (in prep.) Learning web development: Challenges at an Earlier Stage of Computing Education. In *Proc. of ICER*.

Drexel UNIVERSITY