

An online tutorial teaching social science researchers how to use a simple free tool for conducting research online

Marianne Promberger
University of Pennsylvania
3720 Walnut St
Philadelphia, PA
mpromber@psych.upenn.edu

ABSTRACT

The possibilities of conducting social science research on the web are vast, but also underused. Many social science researchers lack computer skills that would enable them to implement their web research from scratch, and they often turn to expensive solutions that may limit the control they have over experimental design. Jon Baron's JavaScript template for web experiments is free and publicly available, and researchers can modify it to suit their research project. It is, however, daunting to those unfamiliar with looking at programming code or with the basic workflow of working on a web page using a text editor. I am in the process of writing a tutorial that aims to bridge this gap, that would enable the less *tech-savvy* researcher to learn how to modify the template using a cookbook-method approach for fast results, and that would also relate aspects of usability and data security.

1. INTRODUCTION

The internet offers plenty of opportunities for conducting research efficiently. Participants can be contacted quickly and with a minimum of hassle via e-mail, studies can be put online and participants can fill in questionnaires on the WWW using a web browser. The data are stored electronically and are immediately available for analysis.

The opportunities available are arguably underused. Many researchers lack an understanding of the technical aspects to fully implement online research "from scratch." One solution is to expensively buy outside expertise by hiring web design experts who implement the design. Another solution is to use available online survey services such as *SurveyMonkey* (www.surveymonkey.com). Both of these solutions have drawbacks, most notably: (a) they are costly; (b) they may not give the researcher full control over the resulting method of data collection, sometimes also not over data storage.

There is a third way out of this dilemma: researchers could

learn to use available free and open source tools to conduct their research online. Arguably, many researchers already do this. However, there seems to be a gap between what we might call the *tech-savvy* and the *less-tech-savvy*. To oversimplify: The tech-savvy are not afraid of working with computer code, and they can develop a system entirely from scratch. The less-tech-savvy are afraid of code and would rather not touch anything that does not have a graphical user interface. An escape from this dichotomy is to empower the less-tech-savvy to have a minimal understanding of code, and to give them the bits and pieces they need to put their research together, along with a cookbook recipe to do this. They also need to be equipped with a basic understanding of the implications for usability and confidentiality.

Jon Baron has taken the first step to this solution by publishing the JavaScript template he uses for his web experiments online, available at <http://finzi.psych.upenn.edu/~baron/ex/template.htm>. I have used this method for all my graduate research, and am currently using it for my dissertation research.

2. AIM OF THE TUTORIAL

The file of the template itself contains comments on its usage. However, these are neither intended nor sufficient for a social science researcher unfamiliar with HTML and JavaScript code, maybe even with the process of working on code in a text editor altogether. To fill this gap, in collaboration with Jon Baron, I have started to write an online tutorial for Jon Baron's template, located at <http://psych.upenn.edu/~mpromber/jstemplate.html>.

The tutorial is specifically aimed at social science researchers who wish to conduct research online, but who have no extensive computer or programming knowledge. We can reasonably assume that these researchers would by default be likely to either use commercial services such as SurveyMonkey, or to consult professional IT help of some sort. Arguably, Jon Baron's template has strong advantages over these methods: it is open source, and written in the widely used languages HTML and JavaScript, using a minimal PHP script to store the data.

This not only means that it can be downloaded and used free of charge. More importantly, it means that researchers can modify the code whichever way they want – to implement different experimental designs (the use of JavaScript allows for great flexibility), to change the way things look, or even

to create their own research tool that makes use just of a small part or some code snippet from the template. If they want to go back ten years later to repeat the study, they have full access to the material they used, and can tweak any aspects they like. At the same time, an open source tool can enhance the openness and replicability of research with respect to the research community and to the public. If they so wish, researchers can easily publish the materials they used online. Other researchers can look at the material, and can try to replicate the findings with minimal hassle. For example, the template for one experiment can easily be modified to implement the same research design with different stimulus material or question wording.

3. CHALLENGES

In theory, researchers could learn to use the template themselves. It already contains an annotation of comments explaining what parts of it do. Information about HTML and JavaScript, even tutorials for those languages, are readily available online and can easily be found with a search engine.

In practice, however, many researchers, especially in the social sciences, have limited, if any, experience with computer languages such as HTML and JavaScript, and probably have trouble knowing where to start looking for information. They will also likely have the impression that the workload of manipulating the template is too great to justify its advantages over other available tools. We would argue that the perception of workload is skewed, especially if the researcher takes into account the number of experiments he will conduct during his career. The learning curve for starting to write an experiment in a computer language, for someone who has never done any coding, is steep at the beginning, but it declines quickly. After a few experiments, the marginal cost of doing each additional experiment will be lower than putting together an additional experiment with a tool that was easier to use initially.

4. APPROACH

Jon Baron's template can alleviate the initial steepness of the learning curve, because researchers can modify it instead of creating something from scratch. However, for researchers unfamiliar with how it works, and unfamiliar with JavaScript, it is not straightforward what parts they should modify, how modifying them will affect the experiment and data storage, and to what degree they have to understand parts of the template that they do not modify in order to use it for their own research. The tutorial we are writing tries to fill this gap.

As a first step, we are trying to be as explicit as possible. To many researchers, it may not be clear that "to modify the template" means to open it in a text editor, change things, and then look at the result in a web browser – so we are saying so explicitly. Even less obvious is the fact how to deal with errors, or even how to find out about them in the first place. Firefox has a very useful built-in error console. However, once again, even if they know about it, most researchers will not understand what they mean, or how to deal with them. To someone who has never done any debugging, it will not be obvious that it is most sensible to deal with the first error message, then see whether the other

ones go away (which they may well do), instead of trying to correct all errors at once.

Basically, the template does two things for the researcher: (1) It presents questions to the subject and collects the subject's answers, and (2), it submits these answers to the researcher by appending them to a file on the researcher's server. Technically, this second step is done by a PHP script on the researcher's server that is called from the template after the last question. The first step is where subjects will want to modify things: they want to ask different questions, and implement different experimental designs. They may also want to change the form of presentation, for example change the font for headers, or include images with questions. The second step, submitting the data on the server, is something that will need to be modified only a little, and that can subsequently remain unchanged over many experiments. Most researchers will be even less interested in how the second step works than in the first step.

Nevertheless, submitting the data and storing them on the server has implications for subject confidentiality. The tutorial will try to make these clear, point to solutions and be clear about the trade-offs involved. As with all security questions, technology is only part of the picture, and more important is an understanding of the probability of someone trying to compromise the data, the harm that this could entail, and the trade-offs with other goals that researcher and subject have. Ultimately, the researcher may want to consult with the IT department that handles security on the server, but they need to have a relatively solid understanding of the issues first. The tutorial will try to give them this understanding as far as possible.

The main bulk of the tutorial however will deal with the first step: asking questions in different experimental designs. Probably the most effective way to learn any type of programming is examples. The original template as you can currently download it from the web contains a very basic design: three proposals are presented in random order, one at a time, with the same two questions for each of them. The tutorial will include several different examples of different experimental designs, and guided solutions on how to implement them. The first examples will be more elaborate, with more guidance, and for subsequent examples we will assume that the reader can figure out what the solution means. Implementing different experimental designs and randomizations is in fact a strong point with JavaScript, but it will be hard to figure this out for researchers who have no prior experience in JavaScript. As far as possible, we will also include considerations of presentation. The template in its current form is very simple, with minimal "styling." We feel that this is an advantage, and that deviations from this styling need to be considered carefully with usability and accessibility in mind.

It is clear that we cannot be exhaustive on all the points. However, ideally, we will be understandable to researchers who have no prior programming knowledge without being condescending, and we will especially help them get to a point where they can ask the right questions, either when learning more about using the template or when considering alternative solutions.