Crowdsourced User Interface Testing for Multimedia Applications

Raynor Vliegendhart
Delft University of Technology
Delft, The Netherlands
r.vliegendhart@tudelft.nl

Eelco Dolstra
LogicBlox, Inc.
Atlanta, Georgia, USA
edolstra@gmail.com

Johan Pouwelse
Delft University of Technology
Delft, The Netherlands
j.a.pouwelse@tudelft.nl

ABSTRACT

Conducting a conventional experiment to test an application's user interface in a lab environment is a costly and time-consuming process. In this paper, we show that it is feasible to carry out A/B tests for a multimedia application through Amazon's crowdsourcing platform Mechanical Turk involving hundreds of workers at low costs. We let workers test user interfaces within a remote virtual machine that is embedded within the HIT and we show that technical issues that arise in this approach can be overcome.

Categories and Subject Descriptors:
H.5.2 [User Interfaces]: Evaluation/methodology

Keywords: crowdsourcing, usability study, A/B testing

1. INTRODUCTION

Conducting an experiment to test an application's user interface is a costly and time-consuming process. A lab setting is needed in which the experimenter has full control over the environment and technical setup and in which the participants can be instructed. This generally means the experimenter can only accomodate a small number of user subjects at a time due to limited capacity. Furthermore, in order to draw statistically significant conclusions, a large number of participants is needed. Hence, conventional usability studies do not scale well.

In this paper, we show that it is technically feasible to conduct a large scale usability study on Amazon's Mechanical Turk crowdsourcing platform (http://www.mturk.com) involving hundreds of participants at low costs. In our approach, we face the challenge of no longer having full control over the experiment. While we can maintain control over the technical setup (OS, browser, etc.) that is running the application under test, we cannot control the environment in which the worker is performing the task. We show, however, that we can design usability studies to accommodate for this lack of control.

While user interfaces of web pages are already being evaluated on Mechanical Turk using services like TryMyUI (http://www.trymyui.com), our work allows any application's user interface to be tested by workers from a crowdsourcing platform. We have implemented a prototype that presents Mechanical Turk workers a display of a virtual machine (VM) embedded within the web page of the Human Intelligence Task (HIT). This VM runs the graphical user interface (GUI) under test on a server operated by the experimenter, ensuring we have full control over the technical setup. Workers can interact with the GUI using the keyboard and mouse and are asked to execute a series of steps as described by the HIT (Figure 1). These steps are visually recorded by the VM. The resulting video can be used by developers for analysis, e.g., in case workers reported any problems.

To test whether our prototype can be used for usability studies, we ran A/B tests [1] for variants of Tribler, a multimedia sharing application [3]. In this usability study, we evaluated whether an experimental user interface feature inspired by previous work [2] would help users in finding multimedia content faster.

Implementation details of our prototype are outside the scope of this paper and are reserved for future publications. Some of the collected statistics that we present here are from a larger study which this work is part of. The focus of the larger study is not limited to usability studies, but also includes semi-automated continuous (e.g., periodic) testing.

The structure of this paper is as follows. We first describe technical factors that impact the HIT design and affect the testing of user interfaces (Section 2). We then discuss the design and the results of our usability study (Section 3). Finally, we summarize our findings (Section 4).

2. TECHNICAL FACTORS

In a conventional lab setting, the experimenter has the opportunity to eliminate any environmental factors that may have an impact on the results of an experiment. In our approach, the experimenter only has full control over the technical setup running the user interface that needs to be tested. We cannot control for technical factors that play a role when workers are connecting to one of the remote virtual machines, such as: *a*) Bandwidth of the worker's connection; *b*) Latency of the worker's connection; and *c*) Screen resolution of the worker's display. We collected this information on each worker's technical setup as well as each worker's location in our larger evaluation study on crowdsourced GUI testing. The study involved 398 unique workers submitting 700 assignments from 32 different countries.

We found that our workers generally had a fairly slow Internet connection. Their connections had a median download speed of 48 KiB/s and had an average ping of 260 milliseconds. This factor has an effect on the task completion time, which needs to be accounted for when completion time is a key element in the usability study.

CrowdMM'12, October 29, 2012, Nara, Japan.
ACM 978-1-4503-1589-0/12/10.

We also found that our workers were using low resolution displays. The majority of the workers had a 1024x768 (25.3%), 1366x768 (20.7%) or 1280x800 (11.8%) screen. To accommodate for these screens, the display of the VM should also be small. If it is too large, the worker cannot see both the embedded VM and the HIT's instructions simultaneously, which negatively impacts the worker's workflow. We therefore chose to use a resolution of 640x480 for the usability study which we describe in the next section.

3. USABILITY STUDY

One aspect of usability is efficiency, e.g., how quickly can a user carry out a specific task. If our crowdsourced user interface testing approach were to be feasible, it is required that task completion times measured during experiments are reliable and are not influenced by external factors. However, as we have seen, there is a large variance in worker connection speeds (Section 2) which could cause a large variance in the task completion time.

To test whether we can detect significant differences in task completion times, we focused on A/B testing for the usability study. Workers were instructed to issue several specific queries to find and download multimedia content in Tribler. The VM server presented connecting workers either variant A or variant B of the Tribler application. The application was instrumented to log the time between each query and download action. We modified variant B to include an artificial delay of 2 seconds in displaying search results, expecting that each query-download task would take 2 seconds longer when compared to variant A.

We launched a HIT of 100 assignments (and thus 100 workers), which took 28h58m to complete and costed a total of US$25. This yielded 354 and 330 measurements for variant A (normal) and B (delayed), respectively. The median interval between searching and downloading was 19.6s for variant A and 21.7s for variant B, conforming to the artificial 2 second delay that was introduced in variant B. The same was not observed for the arithmetic mean due to extreme outliers in variant A (max: 748.9s), but discarding the 25% highest measurements to account for skew resulted in a statistically significant difference between the two trimmed arithmetic means (Student t-test, $P = 0.049$). We thus conclude that the variance on connection speeds is not an issue.

Following this conclusion, we repeated the experiment to evaluate a new feature. This time, variant B contained an experimental Tribler feature called "bundling". This feature groups related search results together based on one of a few different notions of similarity inspired by earlier work [2] such as filename or size, but we found no statistical difference in task completion time. The second HIT took 28h38m to complete. Thus the total runtime of both HITs was less than three days and costed in total only US$50.

4. CONCLUSIONS

In this paper, we have shown that it is feasible to carry out A/B tests for a multimedia application on Mechanical Turk. While technical factors impact the design of the HIT and the usability study, we can account for them. Using our approach, we have been able to involve hundreds of workers to evaluate an experimental user interface feature within a few days at reasonably low costs.

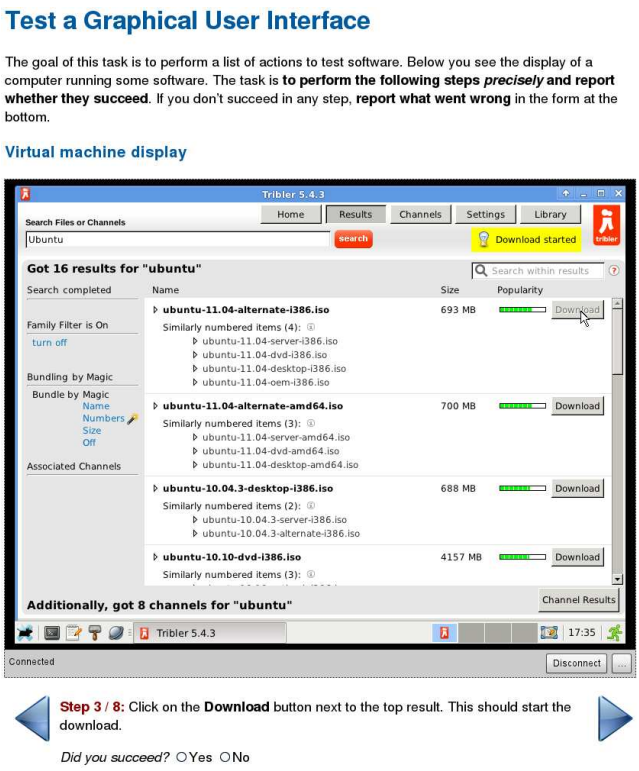

Figure 1: An example of a GUI testing HIT as it appears in a worker's web browser. The steps are shown below the embedded VM's display.

5. ACKNOWLEDGMENTS

We wish to thank Martha Larson for her advice on crowdsourcing and her comments on the design of the A/B test; Niels Zeilemaker for discussions and fixing bugs in Tribler; and of course all workers who participated in our HITs. This research was partially supported by: NWO-JACQUARD project 638.001.208, *PDS: Pull Deployment of Services*, as well as the NIRICT LaQuSo Build Farm project.

6. REFERENCES

[1] R. Kohavi, R. Longbotham, D. Sommerfield, and R. M. Henne. Controlled experiments on the web: survey and practical guide. *Data Mining and Knowledge Discovery*, 18:140–181, Feb. 2009.

[2] R. Vliegendhart, M. Larson, and J. Pouwelse. Discovering user perceptions of semantic similarity in near-duplicate multimedia files. In *Proceedings of the 1st International Workshop on Crowdsourcing Web Search*, pages 54–58. CEUR-WS.org, Apr. 2012.

[3] N. Zeilemaker, M. Capota, A. Bakker, and J. Pouwelse. Tribler: P2P media search and sharing. In *Proceedings of the 19th ACM International Conference on Multimedia*, pages 739–742. ACM, 2011.