Convenient Mobile Usability Reporting with UseApp

Johannes Feiner
IIT[1]
johannes.feiner@fh-joanneum.at

Keith Andrews
IICM[2]
kandrews@iicm.edu

Elmar Krainz
IIT[1]
elmar.krainz@fh-joanneum.at

IICM[2], Institute for Information Systems and Computer Media (IICM), Graz University of Technology, Austria
IIT[1], Institute for Internet Technologies & Applications (IIT), FH JOANNEUM, Austria

Abstract

Usability reporting is necessary to communicate the results of usability tests to developers and managers. Writing usability reports (data aggregation, interpretation, formatting, and writing for specific readerships) can be tedious. Particularly for mobile usability evaluation, where recording user task performance outside a lab is often necessary, testing and reporting can be costly. In many cases, automated extraction of usability findings would be helpful, but is rather difficult to achieve with commonly used report formats such as Word or PDF.

UseApp is a tablet-based web application developed to overcome some of these limitations. It supports the capture of usability data *in the field* during testing, simplifying data collection and aggregation. Live-reports are generated on-the-fly and usability findings can be exported electronically to bug tracking systems.

1 Mobile Usability Reporting

Usability evaluations are performed to validate the usability (and user experience) of software products. For example, experts might conduct heuristic evaluations (HE) to detect potential flaws in applications based on their experience and judgement. Thinking aloud (TA) tests might be conducted with representative test users to discover problems in realistic usage scenarios.

Smaller software development teams often do not have the resources to conduct extensive user studies. Furthermore, the modern practice of agile software development encourages rapid, incremental testing. In both cases, testing has to be simple and efficient. A tool supporting electronic capture of usability data as easily as using a pen and paper can be of great benefit.

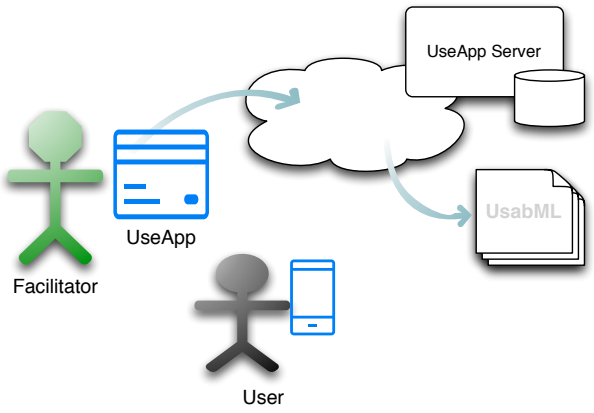

Figure 1: UseApp helps evaluators record the performance of users electronically during usability testing.

Nowadays, many applications are designed to run on mobile phones, shifting the focus of usability testing to mobile usability testing. This shift requires a tool set supporting mobile reporting. Reports in structured formats such as UsabML [FAK10] allow evaluation results to be processed electronically: findings can be extracted and then imported into bug tracking systems automatically.

2 Related Work

Many methods for evaluating user interfaces have been developed over the past three decades [DR93; Nie95]. Formative usability evaluation [Red+02] seeks to discover potential usability problems during the development of an interface, so they can be fixed. Of the formative evaluation techniques, Heuristic Evaluation (HE) [NM90; Nie94b; HLL07] and Thinking Aloud (TA) [RC08] testing are particularly widely used. Nielsen [Nie94a] suggested that some, simplified usability evaluation is always better than

In: W. Aigner, G. Schmiedl, K. Blumenstein, M. Zeppelzauer (eds.): Proceedings of the 9th Forum Media Technology 2016, St. Pölten, Austria, 24-11-2016, published at http://ceur-ws.org

none. Brooke [Bro96] proposed the System Usability Scale (SUS) to make assessment through questionnaires simpler and results comparable through normalised scoring.

Usability reporting feeds back the findings of usability evaluations to development teams [Que05; Her16]. According [Yus15] and [YGV15] using conventional bug tracking systems for usability reporting does now work well. Structured written reports have traditionally been used [FH03; LCA97]. Some efforts have been made to standardise the structure of such reports, including the Common Industry Format (CIF) [NIS99] for formal experiments. However, usability reports are still largely delivered in traditional document formats such as Microsoft Word and PDF, which are extremely hard to process automatically. UsabML [FAK10] is a structured, XML-based format format for usability reports, which allows tool-based extraction and/or conversion and thus fosters simpler automation and reuse.

Reporting usability defects to software developers (cmp [Hel+11]) is a challenge still. [YGV16] investigated reporting and analysed 147 responses. They detected a gap between the what reporters provide and what developers need when fixing defects. UseApp aims into the same direction, as it narrow this gap by supporting semiautomated handover of usability results into bug tracking systems.

Modern usability evaluation shifted towards open use situations and takes the mobile context into account, as discussed in [BH11], [KSV12] and [Lan13]. ISO standards support objective measurement of the usability of mobile applications as reported in [MIA16]. Several tools to assist mobile usability testing can be found in literature. [Sto+15] present MARS, a mobile app rating scale. The tool helps assessing and classifying apps in health sector. The challenges of automatic UI observation and event logging to improve usability on mobile apps can be found [Ma+13], but the support for usability engineering methods (like TA or HE) is missing. Frameworks with a set of different tools and methods to support mobile usability evaluation can be found at [And+01] and [Che16].

Also, some commercial products are on the market. For example, Usertesting[1] is a product which helps to add usability testing on mobile platforms. Beside premium/paid support for testing, simple test can be created with the help of an online tool. Another tool for testing-support of mobile web sites is UXRecorder[2] which supports recording of users touch and facial impressions.

The systematic mapping study [ZSG16] about mobile application testing techniques categorised the different approaches and stated that 19 out of 79 studies employed usability testing. The paper discusses many challenges of mobile testing, such as context-awareness, lab vs. in-the-wild testing, video recording or mobile eye-tracking. One of the main challenges addressed in several papers was *Improving the test suite*. Furthermore, [ZSG16] refer to research groups working on improved toolkits and testing frameworks: [Can+13] for Advanced Test Environment (ATE), a platform which supports automatic execution of user experience tests, [LH12] for a toolkit for unsupervised evaluation of mobile applications, [BH09] a logging based framework to evaluate usability of apps on mobile devices, and [VCD15] for automated mobile testing as a service. For research crowdsourcing in mobile testing [Sta13] created the lightweight *Cloud Testing of Mobile Systems* (CTOMS) framework.

In contrast to this work, which focuses on reporting, only few of the cited approaches mention post-processing and reuse of reports at all.

Criteria	Description	Usage in UseApp
Simple and Fast	Minimise input, use templates.	No pen and paper required. Placeholders and default values.
Context Awareness	Sensor support (GPS), timing.	Auto-timing of task duration.
Don't Repeat Yourself (DRY)	Manage and store project and user details.	Reuse existing user details, questionnaires.
Export and Reuse	Structured formats, post-processing.	Export as UsabML.

Table 1: Selected design criteria for a mobile usability reporting tool.

3 UseApp Concept

UseApp is a client-server web application, as shown in Figure 1. The facilitator manages the mobile user testing and typically enters data into the system using a web browser on a tablet. The criteria used to design UseApp are shown in Table 1. Data entry should be fast and simple, through a minimal interface and use of templates, placeholders, and default values. Sensors should be used to automate procedures as far as possible. Data should only have to be entered once. Overviews and reports should be generated on-the-fly and results should be exported in a structured reporting format.

Recipes for common evaluation tasks, such as a thinking aloud test or administering a standard questionnaire should be available pre-canned. The interface should support focus and context: giving an overview whilst simultaneously allowing the facilitator to focus on the details of current actions. Colour-coded indicators should give feedback about already completed sections, and highlight where data is still

[1] https://www.usertesting.com/.
[2] http://www.uxrecorder.com/.

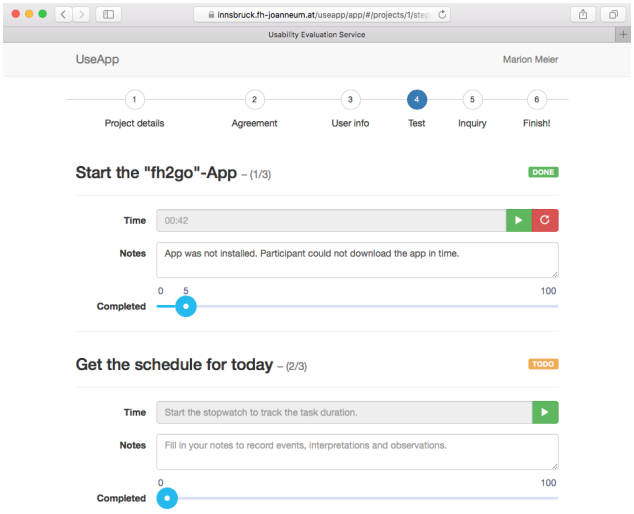

Figure 2: Just six steps – from setup, to data entry to a final report – motivates even small development teams to perform mobile usability tests.

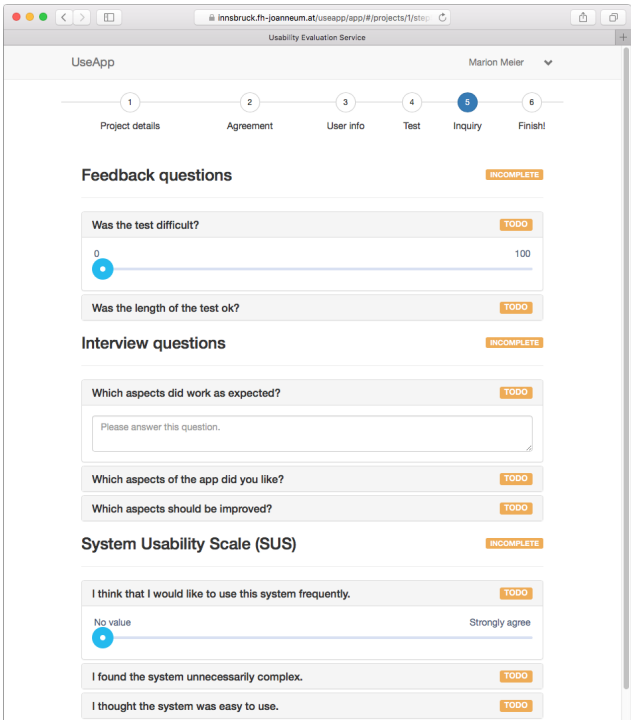

Figure 3: Many built-in placeholders and the timer functionality allow simple and fast reporting.

incomplete. To remove the need for pen and paper, everything should be possible directly on the tablet: from signing a consent form with a stylus or via audio, to answering questionnaire questions by tapping.

4 UseApp Implementation

UseApp currently has built-in support for running Thinking Aloud (TA) tests and administering SUS [Bro96] questionnaires. In future versions, support for Heuristic Evaluations (HE) and other questionnaires and rating scales will be added.

The client implementation uses many features of modern HTML5 web technologies, in order to support the features outlines in Section 3. Responsive web design is used to support several screen resolutions and provide sensible fallbacks where features are not supported by a particular device or browser. Offline storage, sensors, audio input and output, and canvas-based charts are all used.

The UseApp server is built in Ruby/Rails and exposes a restful application programming interface (API). Thus, the client only retrieves and stores data on the server, but the layout and rendering are completely server independent.

The workflow for a TA test comprises six steps (Project Details, Agreement, User Info, Test, Inquiry, and Finish), as indicated in the top bar in Figure 2. The workflow starts with entering the project details. Test users then give their consent and answer demographic and background questions.

The facilitator can track individual or collective performance directly with help of *UseApp*. Placeholders and templates support and speed up facilitator input as shown in Figure 3. Timing of task duration is supported by built-in timers. Task completeness can be indicated just be moving a slider. After completing tasks, the users are asked for feedback. As the questions have all been prepared and assembled in advance, the answers are collected in electronic form.

The results can be viewed for single participants, or for a group of participants, including means and summaries. Multiple charts are available to support interpretation and communication of the results. Figure 4 shows an example. Notes and annotations can be added by the facilitator.

5 UseApp in Action

UseApp was trialled for a number of mobile usability evaluations. The UseApp server was set up in-house and the installation of the web app on an iPad was prepared in advance. The manager of each study entered the project details, task descriptions, and questionnaire questions in advance. As users performed their tasks, the facilitator had their iPad in hand to guide the session, enter observations, and record task duration. After completing the tasks, an interview was conducted and a questionnaire was filled out. Immediately after each test user has finished, the usability managers had access to the results and could add any comments or notes relevant to that test.

Feedback from the first users of UseApp (the usability evaluations managers and facilitators) has indicated some of its benefits and limitations:

- *Feedback*: the top bar indicating the six steps to completion was useful feedback.

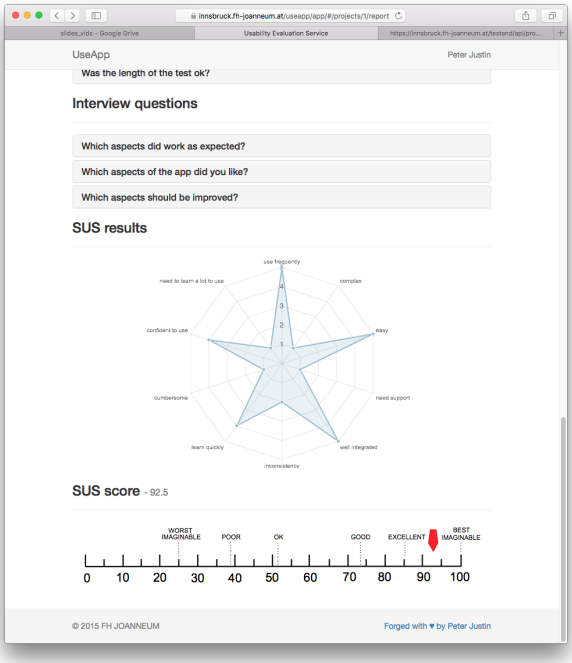

Figure 4: Results are calculated on-the-fly and make results available instantly.

- *No Paper*: no need for paper keeps the test environment uncluttered.
- *Re-Use*: where user testing is required multiple times, the reuse of already prepared evaluation documents, such as same or similar questions for the questionnaire, is time saving.
- *Export*: software developers liked the idea of post-processing reports. After exporting the usability reports in structured UsabML, automated import into bug tracking systems is not difficult.

UseApp acts as a practical companion when running a mobile usability test. Although UseApp can help, preparing and conducting usability tests still takes time and effort.

A minor limitation was the lack of support for freehand writing when signing the consent form. A tablet supporting a stylus might be useful for future versions instead of forcing users to draw with their fingers.

6 Concluding Remarks

UseApp has the potential to support usability evaluators in multiple ways. It simplifies data entry when conducting mobile usability tests, provides templates for input, automation for recording tasks, and reuse of project data. Instant reporting and flexible export into structured UsabML help accelerate the provision of usability findings by the test team to the appropriate software developers.

Ongoing improvement of UseApp will expand evaluation methods supported and the palette of built-in templates. The use of GPS sensors to track location might also be useful in some evaluation contexts.

References

[And+01] Terence S. Andre, H. Rex Hartson, Steven M. Belz, and Faith A. McCreary. “The User Action Framework: A Reliable Foundation For Usability Engineering Support Tools”. In: *International Journal of Human-Computer Studies* 54.1 (Jan. 2001), pages 107–136. doi:10.1006/ijhc.2000.0441 (cited on page 2).

[BH09] Florence Balagtas-Fernandez and Heinrich Hussmann. “A Methodology and Framework to Simplify Usability Analysis of Mobile Applications”. In: *Proc. IEEE/ACM International Conference on Automated Software Engineering*. ASE 2009. IEEE Computer Society, 2009, pages 520–524. ISBN 9780769538914. doi:10.1109/ASE.2009.12 (cited on page 2).

[BH11] Javier A. Bargas-Avila and Kasper Hornbæk. “Old Wine in New Bottles or Novel Challenges: A Critical Analysis of Empirical Studies of User Experience”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’11. Vancouver, BC, Canada: ACM, May 7, 2011, pages 2689–2698. ISBN 9781450302289. doi:10.1145/1978942.1979336 (cited on page 2).

[Bro96] J. Brooke. “SUS: A Quick and Dirty Usability Scale”. In: *Usability Evaluation in Industry*. Edited by Patrick W. Jordan, Bruce Thomas, Bernard A. Weerdmeester, and Ian L. McClelland. Taylor & Francis, 1996. Chapter 21, pages 189–194. ISBN 0748404600 (cited on pages 2–3).

[Can+13] Gerado Canfora, Francesco Mercaldo, Corrado Aaron Visaggio, Mauro DAngelo, Antonio Furno, and Carminantonio Manganelli. “A Case Study of Automating User Experience-Oriented Performance Testing on Smartphones”. In: *Proc. 6th International Conference on Software Testing, Verification and Validation*. ICST 2013. Mar. 18, 2013, pages 66–69. doi:10.1109/ICST.2013.16 (cited on page 2).

[Che16] Lin Chou Cheng. “The Mobile App Usability Inspection (MAUi) Framework as a Guide for Minimal Viable Product (MVP) Testing in Lean Development Cycle”. In: *Proc. 2nd International Conference in HCI and UX on*

Indonesia 2016. CHIuXiD 2016. Jakarta, Indonesia, Apr. 13, 2016, pages 1–11. ISBN 9781450340441. doi:/10.1145/2898459.2898460 (cited on page 2).

[DR93] Joseph S. Dumas and Janice Redish. *A Practical Guide to Usability Testing*. Ablex, Dec. 1993. ISBN 089391990X (cited on page 1).

[FAK10] Johannes Feiner, Keith Andrews, and Elmar Krajnc. "UsabML - The Usability Report Markup Language: Formalising the Exchange of Usability Findings". In: *Proc. 2nd ACM SIGCHI Symposium on Engineering Interactive Computing Systems*. EICS 2010. Berlin, Germany: ACM, May 2010, pages 297–302. ISBN 1450300839. doi:10.1145/1822018.1822065 (cited on pages 1–2).

[FH03] Andy P. Field and Graham Hole. *How to Design and Report Experiments*. Sage Publications, Feb. 2003. ISBN 0761973826 (cited on page 2).

[Hel+11] Florian Heller, Leonhard Lichtschlag, Moritz Wittenhagen, Thorsten Karrer, and Jan Borchers. "Me Hates This: Exploring Different Levels of User Feedback for (Usability) Bug Reporting". In: *Proc. Extended Abstracts on Human Factors in Computing Systems*. CHI EA 2011. Vancouver, BC, Canada: ACM, May 7, 2011, pages 1357–1362. ISBN 9781450302685. doi:10.1145/1979742.1979774 (cited on page 2).

[Her16] Morten Hertzum. "A Usability Test is Not an Interview". In: *interactions* 23.2 (Feb. 2016), pages 82–84. doi:10.1145/2875462 (cited on page 2).

[HLL07] Ebba Thora Hvannberg, Effie Lai-Chong Law, and Marta Kristín Lérusdóttir. "Heuristic evaluation: Comparing ways of finding and reporting usability problems". In: *Interacting with Computers* 19.2 (2007), pages 225–240. doi:10.1016/j.intcom.2006.10.001. http://kth.diva-portal.org/smash/get/diva2:527483/FULLTEXT01 (cited on page 1).

[KSV12] Artur H. Kronbauer, Celso A. S. Santos, and Vaninha Vieira. "Smartphone Applications Usability Evaluation: A Hybrid Model and Its Implementation". In: *Proc. 4th International Conference on Human-Centered Software Engineering*. HCSE 2012. Toulouse, France: Springer-Verlag, Oct. 29, 2012, pages 146–163. ISBN 9783642343469. doi:10.1007/978-3-642-34347-6_9. http://dx.doi.org/10.1007/978-3-642-34347-6_9 (cited on page 2).

[Lan13] Tania Lang. "Eight Lessons in Mobile Usability Testing". In: *UX Magazine* 998 (Apr. 10, 2013). https://uxmag.com/articles/eight-lessons-in-mobile-usability-testing (cited on page 2).

[LCA97] Darryn Lavery, Gilbert Cockton, and Malcom P. Atkinson. "Comparison of Evaluation Methods Using Structured Usability Problem Reports". In: *Behaviour & Information Technology* 16.4 (1997), pages 246–266. doi:10.1080/014492997119824 (cited on page 2).

[LH12] Florian Lettner and Clemens Holzmann. "Automated and Unsupervised User Interaction Logging As Basis for Usability Evaluation of Mobile Applications". In: *Proc. 10th International Conference on Advances in Mobile Computing & Multimedia*. MoMM 2012. Bali, Indonesia: ACM, 2012, pages 118–127. ISBN 9781450313070. doi:10.1145/2428955.2428983 (cited on page 2).

[Ma+13] Xiaoxiao Ma, Bo Yan, Guanling Chen, Chunhui Zhang, Ke Huang, Jill Drury, and Linzhang Wang. "Design and implementation of a toolkit for usability testing of mobile apps". In: *Mobile Networks and Applications* 18.1 (2013), pages 81–97 (cited on page 2).

[MIA16] Karima Moumane, Ali Idri, and Alain Abran. "Usability Evaluation of Mobile Applications Using ISO 9241 and ISO 25062 Standards". In: *SpringerPlus* 5.1 (2016), page 1. doi:10.1186/s40064-016-2171-z (cited on page 2).

[Nie94a] Jakob Nielsen. *Guerrilla HCI – Using Discount Usability Engineering to Penetrate the Intimidation Barrier*. Jan. 1994. http://www.nngroup.com/articles/guerrilla-hci/ (cited on page 1).

[Nie94b] Jakob Nielsen. *Ten Usability Heuristics*. Nielsen Norman Group. 1994. https://www.nngroup.com/articles/ten-usability-heuristics/ (cited on page 1).

[Nie95] Jakob Nielsen. "Usability Inspection Methods". In: *Conference Companion on Human Factors in Computing Systems*. CHI '95. Denver, Colorado, United States: ACM, 1995, pages 377–378. ISBN 0897917553. doi:10.1145/223355.223730 (cited on page 1).

[NM90] Jakob Nielsen and Rolf Molich. "Heuristic Evaluation of User Interfaces". In: *Proc. Conference on Human Factors in Computing Systems*. CHI '90. Seattle, Washington, USA: ACM, 1990, pages 249–256. ISBN 0201509326. doi:10.1145/97243.97281 (cited on page 1).

[NIS99] NIST. *Common Industry Format for Usability Test Reports*. National Institute of Standards and Technology. Oct. 1999. http://zing.ncsl.nist.gov/iusr/documents/cifv1.1b.htm (cited on page 2).

[Que05] Whitney Quesenbery. *Reporting Usability Results – Creating Effective Communication*. Tutorial Slides. Dec. 2005. http://www.wqusability.com/handouts/reporting_usability.pdf (cited on page 2).

[Red+02] Janice (Ginny) Redish, Randolph G. Bias, Robert Bailey, Rolf Molich, Joe Dumas, and Jared M. Spool. "Usability in Practice: Formative Usability Evaluations — Evolution and Revolution". In: *Extended Abstracts on Human Factors in Computing Systems*. CHI EA 2002. Minneapolis, Minnesota, USA: ACM, 2002, pages 885–890. ISBN 1581134541. doi: 10.1145/506443.506647 (cited on page 1).

[RC08] Jeffrey B. Rubin and Dana Chisnell. *Handbook of Usability Testing: Howto Plan, Design, and Conduct Effective Tests*. 2nd edition. John Wiley & Sons, May 2008. ISBN 0470185481 (cited on page 1).

[Sta13] Oleksii Starov. "Cloud Platform For Research Crowdsourcing in Mobile Testing". Master's Thesis. East Carolina University, Jan. 2013. http://thescholarship.ecu.edu/bitstream/handle/10342/1757/Starov_ecu_0600M_10953.pdf (cited on page 2).

[Sto+15] Stoyan R. Stoyanov, Leanne Hides, David J. Kavanagh, Oksana Zelenko, Dian Tjondronegoro, and Madhavan Mani. "Mobile App Rating Scale: A New Tool for Assessing the Quality of Health Mobile Apps". In: *JMIR mHealth and uHealth* 3.1 (2015), e27. doi:doi:10.2196/mhealth.3422. http://mhealth.jmir.org/2015/1/e27/ (cited on page 2).

[VCD15] Isabel Karina Villanes, Erick Alexandre Bezerra Costa, and Arilo Claudio Dias-Neto. "Automated Mobile Testing as a Service (AMTaaS)". In: *Proc. IEEE World Congress on Services*. June 2015, pages 79–86. doi:10.1109/SERVICES.2015.20 (cited on page 2).

[Yus15] Nor Shahida Mohamad Yusop. "Understanding Usability Defect Reporting in Software Defect Repositories". In: *Proc. 24th Australasian Software Engineering Conference*. ASWEC ' 15 Vol. II. Adelaide, SA, Australia: ACM, 2015, pages 134–137. ISBN 9781450337960. doi:10.1145/2811681.2817757 (cited on page 2).

[YGV15] Nor Shahida Mohamad Yusop, John Grundy, and Rajesh Vasa. "Reporting Usability Defects: Limitations of Open Source Defect Repositories and Suggestions for Improvement". In: *Proc 24th Australasian Software Engineering Conference*. ASWEC ' 15 Vol. II. Adelaide, SA, Australia: ACM, Sept. 28, 2015, pages 38–43. ISBN 9781450337960. doi:10.1145/2811681.2811689 (cited on page 2).

[YGV16] Nor Shahida Mohamad Yusop, John Grundy, and Rajesh Vasa. "Reporting Usability Defects: Do Reporters Report What Software Developers Need?" In: *Proc. 20th International Conference on Evaluation and Assessment in Software Engineering*. EASE '16. Limerick, Ireland: ACM, May 2016, 38:1–38:10. ISBN 9781450336918. doi:10.1145/2915970.2915995 (cited on page 2).

[ZSG16] Samer Zein, Norsaremah Salleh, and John Grundy. "A Systematic Mapping Study of Mobile Application Testing Techniques". In: *Journal of Systems and Software* 117 (2016), pages 334–356. http://www.ict.swin.edu.au/personal/jgrundy/papers/jss2016.pdf (cited on page 2).