

## AUTOMATING USER EXPERIENCE TESTING IN MOBILE DEVELOPMENT TEAMS

**Eduard Nicolae BUDACU**

Bucharest University of Economic Studies, Romania

eduard.budacu@csie.ase.ro

**Abstract.** *Mobile development is a user centric activity. Designing user friendly interfaces requires the ability to predict the user behavior based on the best knowledge the team acquired about its target audience. Because people interact intuitively with mobile devices in a continuous learning process, mobile development teams are required to do intensive testing activities before releasing an application or a new feature in an existing app. This paper explores automation tools and techniques that help reducing testing effort. A testing framework is envisioned with focus on increasing user satisfaction.*

**Keywords:** User Experience, User Satisfaction, Mobile Development, Test Automation

**JEL classification:** L86, D80

**DOI:** 10.12948/ie2019.05.05

### 1. Introduction

Mobile devices have become integrated in our daily activity. Estimates show that there are more than 5.112 billion unique mobile users that means roughly 67% penetration in global population. This paper addresses the problem of testing user experience in mobile development teams.

In the first section the problem was defined. Second sections presents a background research touching on the current state of practice in mobile development teams. A user satisfaction measurement is proposed in the third section. The forth section assesses what automation tools are available. Results are discussed in the fifth section. Final section concludes and directions for further research are presented.

### 2. Background research

Succeeding when launching a mobile application is highly correlated with the ability to solve a specific problem or need for the targeted audience. A user-centric approach in an agile methodology starts with defining personas and user stories [1] to capture specific behaviors that can integrate activities using a mobile phone. Agile teams are driven by the purpose of satisfying the customers by delivering working products as early and often as possible. The term of minimum viable product (MVP) is usually referred when planning the initial launch of an app. Further, acquiring a large community of users involves a multi-disciplinary team that sustains and constantly improves the product.

A mobile development team performs different activities like understanding user needs, designing interfaces, implementing algorithms, creating data structures, coding, testing and deployment. Agile methods have proven to be effective for organizing application development, but it involves a lot of effort to shorten the time from idea to testing with real users. Also not having direct access to users makes getting feedback a challenge and measuring user satisfaction is therefore neglected. So who's role is to test is a common question within a team.

We further look into the definition of user experience (UX) to map the roles, responsibilities and knowledge specific to define execute this process in a mobile development team. In the user-centric design approach, user experience is defined as the total interactions a person has with a specific product to solve a specific need [2]. UX also captures the emotions and attitudes of the users regarding a specific product making it suitable for measuring user satisfaction. In figure 1 the standard for human-centered design process is presented as an iterative approach making it suitable for adaptation to agile oriented methodologies.

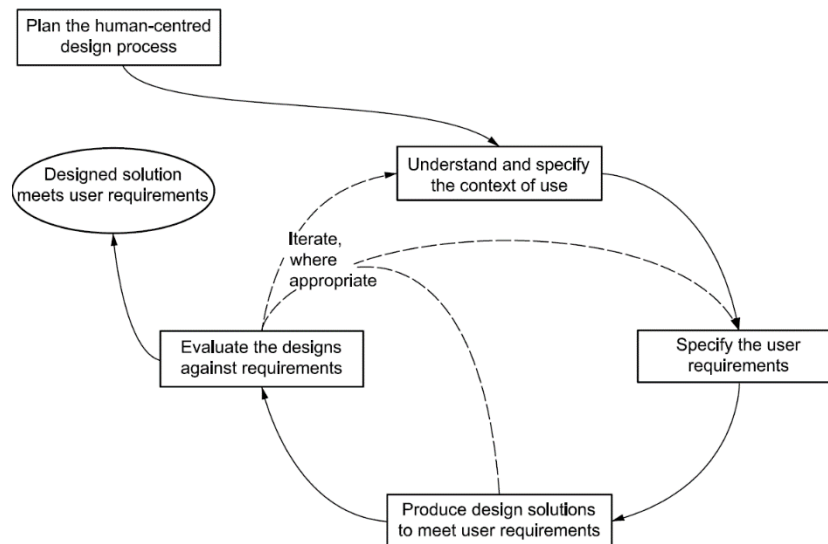


Figure 1. Interdependence of human-centered design activities (source: ISO 9241-210) [2]

In a typical Scrum team we have the roles of Product Owner, Scrum Master and Development Team. According to the Scrum guide the Product Owner is responsible for maximizing the value of the product resulting from work of the Development Team, and the Scrum Master is responsible with helping everyone understand Scrum theory, practices, rules, and values [3]. Recent development of the agile frameworks has led to acknowledgment of UX and design practices as very important in the development process [4], [5]. Currently UX activities are in the responsibility of a specific person within mobile development team or shared within the team members. Furthermore there is a difference between user experience and user interface (UI) roles, the former having specific responsibilities of implementing design sketches into programmable interfaces.

Activities that are usually performed in UX are:

- A/B or Split testing;
- Accessibility testing;
- Design prototyping;
- Evaluating design;
- Information architecture;
- Live chat with users;
- Focus groups and interviews;
- Surveying users.

In the age of the internet automation has evolved and allowed increased efficiency in handling information. Software development life cycle has become shorter as more tools reduced the complexity of writing software. In mobile development programmers rely on integrated development environments (IDEs) to write, test, build and deploy applications. The testing

phase is no longer the sole responsibility of the quality assurance department. In [6] authors review 29 studies that have reported evidence on test automation of mobile applications that cover usability, performance and security. Automation testing techniques fall in offline and online according to [7]. The authors of the researched proposed a framework for objective analysis of test efficiency. In [8] a Mobile Testing as a Service (MTaaS) cloud based system is proposed. Current state of the art in test automation is mostly focused on testing the output of the development team (i.e. Code produced or interfaces). According to [9] user feedback plays a large role in optimizing mobile applications.

Having this breakdown of responsibilities is not clear whose role is to measure customer satisfaction and to cater for the user needs. The opportunity of defining a user satisfaction measurement framework is given by the increased global adoption of mobile devices, internet connectivity, digital transformation of business and increased responsibilities in IT departments to satisfy customer needs.

### 3. User satisfaction measurement framework

In order to test the user experience the focus needs to be moved from actual application delivery process to the ecosystem formed by the users.

Special roles are often integrated in teams to act as user researchers or testers that impersonate users to mimic their behavior and provide feedback to programmers so that they adjust application flows, interfaces and design. Gaining real life feedback before each release is hard especially when the users are distributed on wide areas and teams don't have direct access like in the case of an internal application used within a company.

Figure 2 describes the framework for measuring user satisfaction.

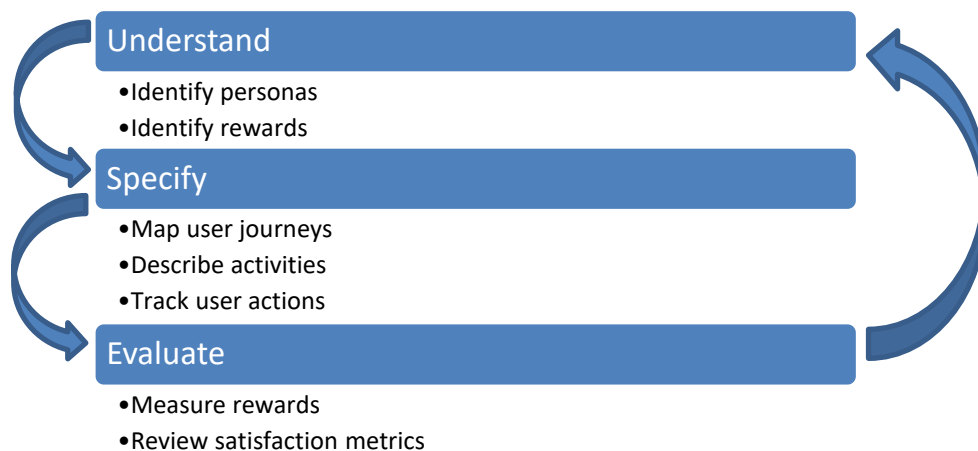


Figure 2. User satisfaction measurement framework

In the first step personas must be identified to understand the overall community of people that interact with the application. For example in a commerce oriented mobile application users may fall in one or many of this categories: Consumers, Creators, Buyers and Sellers. In order to understand the economics of the application we need to map the relations between the actors. Consumers are the main target of the application and represent the great majority of the users. Creators are ones that generate content in the ecosystem. If a transaction occurs the users become Sellers and Buyers. It's not necessary for a transaction to involve an exchange of real money, but it's important to identify them so that we can map rewards that lead to increase in satisfaction. Conversion rate (CR) is one of the metrics that indicate global user satisfaction.

$$CR = (Buyers / Consumers) \times 100$$

Next step involves identifying rewards. User experience is the sum of interaction with a mobile application. User satisfaction is determined by the reward that a person gets by utilizing a mobile application. The reward can be in the form of getting a task done, enjoying relevant content, acquiring a product, interacting with friends, gaining social recognition or other gains that the person using the application is pursuing. An example of reward is the achievement of target revenue (ATR). This is a metric that evaluates the degree of revenue generated by a user based on its individual target.

$$ATR = (\text{Generated revenue} / \text{Target Revenue}) \times 100$$

User journey mapping is a tool specific to agile product development that aims to describe user behavior and map it towards specific outcomes. User journeys are end-to-end flows that users are performing. Examples of a journey is acquiring a product, completing an online course, registering for an event, publishing content. The journey consists of several activities that a user performs in a step by step process like registration, browsing products, selecting a product, payment and thank you page. Activities can also be described as a continuous flow like consuming content generated by other users or might involve users inputting their own data into the application.

In app tracking tools must be setup to record user activity. This tools allow automatic recording of usage, session length, actions and data.

Rewards are triggered when certain conditions are accomplished. In order to measure rewards we need to correlate actions with outcomes envisioned in the user journey mapping. Data is stored and presented in visualizations for analysis.

Evaluation of satisfaction metrics requires highly skilled individuals that acquire knowledge about the users and application. Quantitative and qualitative satisfaction reports are presented to all parties involves to facilitate decision making.

#### 4. Automation tools and techniques

In order to automate user experience testing development team members that are responsible whit this process must be able to interact with real users. The logistics for configuration of a testing environment involves a lot of effort in providing the devices, installing applications, preparing test cases and executing the scenarios. Tools that are currently in use for the testing are oriented to automating scenarios that are predefined or recorded and then executed on emulators, real devices or devices in cloud. In table 1 a selection of popular tools is presented along with the approach taken to automate testing effort.

Table 1. Automation tools

Tool	Approach	Platform
Appium Studio	Record and replay scenarios	Android, iOS, Web
Selendroid	Scripting test cases	Android, Web
Bitbar Cloud	On demand devices in cloud	Android and Apple devices
SeeTest Cloud	On demand devices in cloud	Android and Apple devices
Respondent	Online focus groups and survey	Any platform
SurveyMonkey	Online survey tool	Any platform
Google Analytics for Mobile Apps	Analytics and tracking	Android, iOS

App Annie	Analytics, marketing	Android, iOS
-----------	----------------------	--------------

This tools capture only part of the entire user experience that is configured by the test analysts that design the scenarios. We need to extend the automation so that it will allow interaction with the real users of the application. A cloud based solution that organizes groups of users that subscribe to a journey testing would enable test analysts to capture more meaningful data regarding user satisfaction.

## 5. Discussion

To gain the full advantages of this framework a software platform is necessary using an architecture that is cloud based so that it would enable developers to address users located in different zones. Also the testing will generate a lot of data so it will need proper storage, security and computation power.

When we think about what can be automated and what still needs human intervention the tendency is to identify tasks that are simple to execute and don't involve creativity. But with intelligent algorithms that rely on machine learning even more complex tasks like generating multiple iterations of a specific design and test them with the users can be approached. This leads to the opportunity to create new development tools that embed artificial intelligence into the UX process.

Data gathering and visualization can be fully automated in mobile applications. Data gathering is enabled by integrating tracking codes that send events to a server where analytics can be performed. Visualization is available in analytics tools as graphs, maps that show the journey of the user, time spent on different activities, data input and other metrics. The framework would still need human intervention in defining personas and rewards. The last part of evaluating user satisfaction metrics involves human judgement to make decisions and act on future changes in the product.

Automation has the following advantages:

- Reduce work with repetitive activities;
- Allow team members to make decisions driven by data;
- Increase company competitiveness;
- Improve the quality of the product;
- Allow testing multiple scenarios;
- Better satisfy user needs.

Full automation in UX has limitations due to the human interactions and empathy necessary for understanding problems that a user is facing. Yet a lot of time consuming activities are performed by teams in the testing process that can be automated. Another limitation is having the necessary skills to perform the automations. In order to mitigate this limitations further research needs to be performed in the role of UX within mobile development teams. To gain more insights a comprehensive analysis of the activities in understanding, specification and evaluation of user satisfaction metrics and measurements is necessary.

Another important limitation of automating user experience testing is you cannot replace real users' involvement in the process. Even with the development of autonomous agents that are able to mimic user behavior in a mobile app the actual behavior is hard to predict.

## 6. Conclusions

In this paper the problem of automating user experience testing in mobile development teams was addressed. The current state of agile practices emphasizes the focus on customer satisfaction as the primary concern. Automation plays an important role in reducing the time to market of

a mobile application. A framework for measuring user satisfaction is proposed. Further research will be focused on development of an integrated development environment that allows understanding customer needs, specify user journeys, track activities and measure user satisfaction in mobile development teams. Increase automation in user experience testing will lead to improve software quality, increase user acquisition and reduce user churn.

## References

- [1] M. Cohn, *User Stories Applied: For Agile Software Development*, Addison-Wesley Professional, 2004, p 304.
- [2] ISO, *ISO 9241-210: Human-centred design for interactive systems*, Switzerland, 2010
- [3] J. Sutherland, K. Schwaber, *The scrum guide. The definitive guide to scrum: The rules of the game*, Scrum.org, 2013 Jul.
- [4] T. S. Da Silva, M. S. Silveira, F. Maurer, T. Hellmann, "User experience design and agile development: From theory to practice," *Journal of Software Engineering and Applications*, 5(10), 2012, pp 743.
- [5] D. DeMarco Brown, *Chapter 2 - Agile Methods + UX = Agile UX Agile User Experience Design*, Morgan Kaufmann, pp 39-69, 2013, ISBN 9780124159532.
- [6] S. Zein, N. Salleh, J. Grundy, "A systematic mapping study of mobile application testing techniques," *Journal of Systems and Software*, 117, 2016, pp 334-356.
- [7] D. Amalfitano, N. Amatucci, A. M. Memon, P. Tramontana, A. R. Fasolino, "A general framework for comparing automatic testing techniques of Android mobile apps," *Journal of Systems and Software*, 125, 2017, pp 322-343.
- [8] C. Tao, J. Gao, "On building a cloud-based mobile testing infrastructure service system," *Journal of Systems and Software*, 124, 2017, pp 39-55.
- [9] F. Elberzhager, K. Holl, "Towards automated capturing and processing of user feedback for optimizing mobile apps," *Procedia Computer Science*, 110, 2017, pp 215-221.