

# Chapter 10

## Markov Decision Theory– Based Crowdsourcing Software Process Model

Kamalendu Pal

 <https://orcid.org/0000-0001-7158-6481>

City, University of London, UK

### ABSTRACT

*The word crowdsourcing, a compound contraction of crowd and outsourcing, was introduced by Jeff Howe in order to define outsourcing to the crowd. It is a sourcing model in which individuals or organizations obtain goods and services. These services include ideas and development of software or hardware, or any other business-task from a large, relatively open and often rapidly-evolving group of internet users; it divides work between participants to achieve a cumulative result. It has been used for completing various human intelligence tasks in the past, and this is an emerging form of outsourcing software development as it has the potential to significantly reduce the implementation cost. This chapter analyses the process of software development at a crowdsourced platform. The work analyses and identifies the phase wise deliverables in a competitive software development problem. It also proposes the use of Markov decision theory to model the dynamics of the development processes of a software by using a simulated example.*

### INTRODUCTION

Crowdsourcing is the Information Technology (IT) mediated engagement of crowds for the purposes of problem-solving, task completion, idea generation and production (Howe, 2006; Howe, 2008; Brabham, 2008). The latest breakthroughs in Information and Communication Technologies (ICT) have ushered a new dawn for researchers to design innovative crowdsourcing systems that can harness Human Intelligence Tasks (HITs) of online communities. The prime aim of crowdsourcing is to facilitate the *wisdom of crowds*. The theory suggests that the average response of many people, even amateurs, to a question is frequently more accurate than the view of a few experts. In this respect, a community of individuals with common interests and facing the same tasks can deliver better products and solutions than experts alone

DOI: 10.4018/978-1-6684-3702-5.ch010

in the field. Information systems scholars Jean-Fabrice Lebraty and Katia Lobre-Lebraty confirmed that the “*diversity and impudence of the members of a crowd*” is a value addition to crowdsourcing operations (Lebraty & Lobre-Lebraty, 2013).

Therefore, the advantages of crowdsourcing lie mainly in the innovative ideas and problem-solving capacity that the diverse contributors – which may consist of experts and interested amateurs – can provide. The crowd can provide expert and faster solution to an existing problem. Depending on the challenge at hand, the solution provided may also prove innovative. In this way, crowdsourcing has emerged as a new labour pool for a variety of tasks, ranging from micro-tasks on Amazon Mechanical Turk (mTurk) to big innovation contests conducted by Netflix and Innocentive. Amazon mTurk today dominates the market for crowdsourcing small task that would be too repetitive and too tedious for an individual to accomplish. Amazon mTurk established a marketplace where requesters can post tasks and workers complete them for relatively small amount of money. Image tagging, document labeling, characterizing data, transcribing spoken languages, or creating data visualizations, are all tasks that are now routinely being completed online using the Amazon mTurk marketplace, providing higher speed of completion and lower price than in-house solutions.

Competitive crowdsourcing is reward based and has been used for variety of tasks from design of T-Shirts to research and development of pharmaceuticals and very recently for developing software (Howe, 2008; Lakhani & Lonstein, 2011; Stol & Fitzgerald, 2014). The mTurk is one of the best-known crowdsourcing platforms where HITs or microtasks are performed by thousands of workers (Ipeirotis, 2009).

There are different types of crowdsourcing platforms, such as virtual labour markets (VLMs), tournament crowdsourcing (TC) and open collaboration (OC), which each have different roles and characteristics (Estelles-Arolas & Gonzalez-Ladron-de-Guevara, 2012; Prpic, Taeihagh & Melton, 2014). Along with the growth of crowdsourcing, crowdsourcing platforms are very important to mediate the transactions. At the same time, IT-mediated platforms improve efficiency and decrease transaction costs and information asymmetry. However, these platforms are domain specific.

Crowdsourced Software Engineering derives from crowdsourcing. Using an open call, it recruits global online labour to work on different types of software engineering works, such as requirement elicitation, design, coding and testing. This emerging model has been claimed to reduce time-to-market by increasing parallelism (Lakhani et al., 2010; LaToza et al., 2013; Stol & Fitzgerald, 2014), and to lower costs and defect rates with flexible development capability (Lakhani et al., 2010). Crowdsourced Software Engineering is implemented by many successful crowdsourcing platforms, such as TopCoder, AppStori, uTest, Mob4Hire and TestFlight. Crowdsourced Software Engineering has also rapidly gained increasing interest in both industrial and academic communities.

In this chapter only, software development related crowdsourcing business activities and relevant platforms are considered. Software development is creative and ever evolving. Organizations use various software development process models and methodologies for developing software. A software process model (SPM) specifies the stages in which a project should be divided, order of execution of these stages, and other constraints and conditions on the execution of these stage (Sommerville, 2017). However, the software development methodology (also known as SDM) framework did not emerge until the 1960s. The system development life cycle (SDLC) is the oldest formalized framework for building information systems. The main idea of the SDLC has been “to pursue the development of information systems in a very deliberate, structured and methodical way, requiring each stage of the life cycle – from inception of the idea to delivery of the final system – to be carried out rigidly and sequentially (Elliott, 2004) within the context of the framework being applied. The main objective of this framework in the 1960s was to

16 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

[www.igi-global.com/chapter/markov-decision-theory-based-crowdsourcing-software-process-model/294465](http://www.igi-global.com/chapter/markov-decision-theory-based-crowdsourcing-software-process-model/294465)

## Related Content

---

### Modelling Self-Led Trust Value Management in Grid and Service Oriented Infrastructures: A Graph Theoretic Social Network Mediated Approach

Antony Brown, Paul Sant, Nik Bessis, Tim Frenchand Carsten Maple (2012). *Theoretical and Analytical Service-Focused Systems Design and Development* (pp. 122-139).

[www.irma-international.org/chapter/modelling-self-led-trust-value/66796](http://www.irma-international.org/chapter/modelling-self-led-trust-value/66796)

### Teaching Model-Driven Engineering in a Master's Program: Three Editions on a PBL-Based Experience

Alexandre Bragança, Isabel Azevedoand Nuno Bettencourt (2022). *Research Anthology on Agile Software, Software Development, and Testing* (pp. 1586-1611).

[www.irma-international.org/chapter/teaching-model-driven-engineering-in-a-masters-program/294532](http://www.irma-international.org/chapter/teaching-model-driven-engineering-in-a-masters-program/294532)

### Alert Driven Communications Management for Children Music Learning Based on Suzuki Method

Cheuk-Ting Chan, Kiko Tsz Lan Chengand Dickson K.W. Chiu (2015). *International Journal of Systems and Service-Oriented Engineering* (pp. 75-90).

[www.irma-international.org/article/alert-driven-communications-management-for-children-music-learning-based-on-suzuki-method/134435](http://www.irma-international.org/article/alert-driven-communications-management-for-children-music-learning-based-on-suzuki-method/134435)

### Web Application Vulnerabilities and Their Countermeasures

Kannan Balasubramanian (2018). *Application Development and Design: Concepts, Methodologies, Tools, and Applications* (pp. 1312-1342).

[www.irma-international.org/chapter/web-application-vulnerabilities-and-their-countermeasures/188258](http://www.irma-international.org/chapter/web-application-vulnerabilities-and-their-countermeasures/188258)

### A Web Backtracking Technique for Fraud Detection in Financial Applications

Tasawar Hussainand Sohail Asghar (2018). *Application Development and Design: Concepts, Methodologies, Tools, and Applications* (pp. 861-878).

[www.irma-international.org/chapter/a-web-backtracking-technique-for-fraud-detection-in-financial-applications/188238](http://www.irma-international.org/chapter/a-web-backtracking-technique-for-fraud-detection-in-financial-applications/188238)