ToolStackers: A Boardgame for SE Education

Barış Ardıç
Bilkent University
Department of Computer Engineering
baris.ardic@bilkent.edu.tr

Eray Tüzün
Bilkent University
Department of Computer Engineering
eraytuzun@cs.bilkent.edu.tr

***Abstract*—We propose a complementary board game that aims to familiarize students with usage of tools in software engineering, their functionality and how they fit together in order to form a tool-stack for modern development paradigms like Agile and DevOps for an Application Lifecycle Management course.**

***Index Terms*—game based learning, software engineering, application lifecycle management**

I. Introduction

Application Lifecycle Management (ALM) is an area of modern software development that is concerned with governance, development, and maintenance of application software [1]. Bilkent University provides an ALM technical elective course for the computer science department which focuses on familiarizing students with tools used in the software engineering industry alongside an overview of modern development paradigms like DevOps and Agile development. Throughout multiple semesters of this course, we have seen students struggle with the amount of content in this course mostly due to of a lack of familiarity with how the SE industry operates. Game-based learning has been gaining popularity in recent years for higher educational purposes [2] therefore in order to help breach this knowledge gap, we are proposing a board game that aims to give students a background regarding the tools that are used in SE, their functionality and how they fit together as a tool stack.

II. Learning Objectives

Our design aims to give students a good start with tool related software development practices. The main learning objectives that we desire to accomplish are defined as follows:

- Familiarization of students with basic process areas of ALM tools like project management or issue tracking and expectations from these tools.
- The popular choices of tools in their relative process areas. For example, Jenkins for continuous integration and Bugzilla for issue tracking.
- How using tools together from different process areas, hence a tool stack, enhances software development.

III. Game Design

The game is meant to be played in multiplayer fashion with at least two players. The upper-bound on maximum player size is defined by the number of cards we prepare. We aim for the game to be playable with 5 players at least in order for it to be played by a project group of the ALM course.

1) Cards:

1) Cards: There are three types of cards in the game; tools, upgrades and events. Each tool card represents an ALM tool where the card type is the tool's process area. There are 6 card types inspired from main process areas targeted by the lectures and common tool-stacks in the industry. These areas are project management, issue tracking, continuous integration, monitoring, testing and code review. Each card contains some flavor text regarding the tool it represents and its features in order to give additional information. Some tools can have multiple types as well. For example, Jira would have both issue tracking and project management types.

Upgrade cards increase the proficiency of a tool card the player has. A tool card drawn starts as tier 0 and can be upgraded thrice to tier 3, representing maximum proficiency.

Event cards describe an event regarding a process area. For example; "You need to automate your functional tests" event card would require at least a tier 1 continuous integration card and it would award the players with an upgrade if they posses such a card in their stacks. A harder version of this event would be "You need to adopt continuous deployment" and this event would require a tier 2 card.

2) Game Flow: The game is played in a turn based fashion. Each player starts with 6 random tool cards. The first purpose in the game is to complete a stack (meaning a tool card of each type). This can be achieved by players trading cards or by discarding a card in order to draw a new one from a the tool deck on their turn. If the player is not drawing a card from the tool deck, they will draw from the event deck consisting of event and upgrade cards. Players can benefit from event cards only after they complete their stack. However they still can choose to draw from the event deck hoping for an upgrade card providing some strategical depth. The intend to add more features in order to increase depth and replayability. The game ends when a player has a tier total of 10 on their stack.

3) Game Setting: We intend for this game to be in printed form and played by project groups just after they are formed in the beginning of a semester. This way it will also act as an team building exercise while giving the members a background on ALM tools.

References

[1] J. Rossberg, *Beginning application lifecycle management*. Apress, 2014.
[2] M. Qian and K. R. Clark, "Game-based learning and 21st century skills: A review of recent research." *Comput. Hum. Behav.*, vol. 63, pp. 50–58, 2016. [Online]. Available: http://dblp.uni-trier.de/db/journals/chb/chb63.htmlQianC16

978-1-7281-6807-4/20/$31.00 ©2020 IEEE