IO1: Gamification Scheme for Programming Exercises

May 25, 2019

Document Manager	
Jakub Swacha	US

Document Version:	1.0	Date:	2019-05-25

Published Filename:	FGPE_IO1_Gamification_Scheme_for_Programming_Exercises.pdf
Intellectual Output	IO1
Storage location / link	

License

This work is licensed under a Creative Commons Attribution 4.0 International License.

Disclaimer

The information in this document is provided "as is" and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

The contents of this document are the sole responsibility of its authors and can in no way be taken to reflect the views of the European Union, the Erasmus Plus programme or its National Agencies.

The FGPE Consortium consists of the following partners:

Participant no.	Participant organization name	Short name	Country
1	University of Szczecin	US	Poland
2	CRACS University of Porto	CRACS	Portugal
3	Aalborg University Copenhagen	AAU	Denmark
4	University of Napoli Parthenope	UNP	Italy

Revision history

Version	Author	Notes	Date
0.1	Sokol Kosta (AAU)	Template defined. Introductory structure defined.	March 29, 2019
0.2	Jakub Swacha (US)	Internal structure defined. Tables contents defined. Section 3 content edited based on internal project documentation.	April 01, 2019
0.3	José C. Paiva (CRACS) Ricardo Queirós (CRACS)	Section 4 content edited based on internal project documentation. Section 5 content edited based on internal project documentation. Content validated.	April 26, 2019
0.4	Jakub Swacha (US)	Document structure modified. Section 1 content edited. Section 2 content edited based on internal project documentation.	April 29, 2019
0.5	Sokol Kosta (AAU)	Internal review of the document.	May 6, 2019
0.6	Jakub Swacha (US)	Technical corrections.	May 17, 2019
1.0	Jakub Swacha (US)	Content of sections 3, 4 and 5 corrected and extended.	May 25, 2019

Table of Contents

1. Executive Summary

This document provides a catalogue of gamification techniques adapted for the purposes of programming education. It is a direct result of the work done during the first phase of the Framework for Gamified Programming Education project. It constitutes a necessary base for the development of the further project outputs, primarily, the format for sharing the gamified programming exercises and the tools for handling them.

Section 2 provides more details on this document's background, purpose and scope.

Section 3 defines basic gamification concepts to be used in the further sections, with distinction of the areas of: course organization, player goals and player rewards.

Section 4 defines gamification concepts intended for application on exercise level, with distinction of exercise-level challenge types, reward mechanisms and player metrics.

Section 5 defines gamification concepts intended for application on course level, with distinction of course-level challenge types, reward mechanisms and player metrics.

2. Introduction

2.1 Background

The combined use of automated assessment, which provides fast feedback to the students experimenting with their code and gamification, which provides additional motivation for the students to intensify their learning effort can help pass the barrier of difficulty in learning programming. In such environment, students keep receiving the relevant feedback no matter how many times they try (thanks to automated assessment), and their engagement is retained (thanks to gamification).

Learning programming relies on practicing programming. While there is a number of open software and programming exercise collections supporting automated assessment, up to this date, there are no available open collections of gamified programming exercises, no open interactive programming learning environment that would support such exercises, and even no open standard for the representation of such exercises so that they could be developed in different educational institutions and shared among them.

The primary objective of the Framework for Gamified Programming Education (FGPE) project is therefore to provide a framework for application of gamification to programming education, including the necessary specifications (of the gamification scheme and the exercise definition format), collection of gamified exercises (for popular programming languages) and software (a toolset for editing the exercises and an interactive learning environment providing them to the students).

2.2 Purpose

The true gamification cannot be equated with just "points, badges and leaderboards". In order to increase the students' motivation to learn programming, more advanced gamification techniques should be employed than simply adding points for every completed exercise. The envisaged scope of such scheme includes the areas relevant to gamification of individual exercises (e.g. in-exercise challenges such as "solve the problem without using loops" or "solve the problem with a single statement" and the rewards for passing the respective challenges), and the whole course (e.g. types of badges and virtual items, linking the course completion levels to awards such as badges or virtual items, how the progression is visualized to the student).

The purpose of this document is to provide a catalogue of gamification techniques tailored for programming education. According to the best of our knowledge, currently there is no such a catalogue. It has been devised primarily as a necessary preliminary requirement for the development of the format for sharing the gamified programming exercises and the tools for handling them, as in order to store and process the gamification-related data one needs to know what data to expect. However, we believe that, as it is, it can also help the gamified programming course designers (showing them what they can choose from).

2.3 Scope and related documents

This document covers gamification concepts relevant to programming education. The key sections 4 and 5 list concepts applicable at the level, respectively, of a single exercise and the whole course. Section 3 introduces general gamification concepts referred to in the subsequent sections.

As gamification is a rapidly developing field, the contents of this document are subject to change in future. Please consult the FGPE project website (http://fgpe.usz.edu.pl) for its up-to-date version.

This document does not cover the definition of the data interchange format for gamified programming exercises and courses, which is to be covered in the Data Exchange Format for Gamified Programming Exercises document (scheduled for August 2019).

This document does not cover the documentation of the tools supporting editing and conversion of gamified programming exercises, which is to be covered in the Tools Supporting Editing and Conversion of Programming Exercises document (scheduled for February 2020).

This document does not cover the documentation of the online platform providing gamified programming courses, which is to be covered in the Programming Learning Environment featuring Gamified Exercises document (scheduled for July 2020).

This document does not cover the presentation of the gamified programming exercises developed within the FGPE project, which is to be covered in the Programming Courses featuring Gamified Exercises document (scheduled for February 2021).

2.4 Acknowledgements

Although the editors of this document are listed on page 3, its content reflects the results of the intellectual work of all the involved project team members:

- on behalf of University of Szczecin: Jakub Swacha, Karolina Muszyńska, Mateusz Piwowarski, Agnieszka Miluniec,
- on behalf of CRACS University of Porto: Ricardo Queirós, José C. Paiva and José Paulo Leal,
- on behalf of Aalborg University Copenhagen: Sokol Kosta, Per Lynggaard, and Reza Tadayoni,
- on behalf of University of Napoli Parthenope: Raffaele Montella, Giulio Giunta, Livia Marcellino.

Numerous other persons indirectly contributed to its development as survey participants.

3. Basic Gamification Concepts

This section provides practical definitions for gamification-related concepts which can be used in more advanced gamification mechanisms described in further sections.

3.1 Course organization

Table 3.1 Gamification concepts related to course organization.

Concept name	Concept description
Course Module	A subset of course content related to specific topic (and/or learning stage). Further modules may be locked by default.
Exercise Type	A kind of programming exercise to be solved.
Exercise Mode	Non-default exercise modes define special guidelines regarding how an exercise should be presented or provide modified requirements to make the student interested in solving an exercise again
Locked Content	Player is aware of locked content but cannot access it before certain requirements are met.
Secret	A hidden content about existence of which player is not informed about beforehand (only after making certain defined steps). Its revealing may be random or fixed depending on certain conditions.
Difficulty Level	A subset of course content on a similar difficulty level. Higher levels may be locked by default.

3.2 Player goals

Table 3.2 Gamification concepts related to defined goals.

Concept name	Concept description
Challenge	A single programming exercise to be solved
Requirements	Result conditions that define a correct solution and additional conditions on results or source code that make it more difficult to pass a challenge
Progress	Player's progress within a course measured with a number of metrics (see section 4.3).
Record	The highest value of a certain metric achieved by this player or any player before

3.3 Player rewards

Table 3.3 Gamification concepts related to defined rewards.

Concept name	Concept description
Point	Increases player's score, showing their progress. There may be different types of points
Level	Depends on player score, possibly on other factors. Higher levels may be required to access certain course areas or difficulty levels
Held Record	The fact that the highest value of a certain metric till now has been achieved by this player. Visible to other players
Current Rank	The current rank of this player in their group leaderboard. Visible to other players (possibly with limitations, e.g. only the top of the leaderboard and the neighbours)
Badge	Graphics certifying player's achievement. A single type of badge may have levels
Virtual Item	Received for certain achievements. In contrast to badge, it can possibly be lost, traded, combined, or used for certain purposes; it also does not have to be visible to other players. An example of a generic virtual item is Course Coin
Coupon	Received for certain achievements. In contrast to Virtual Item it is intended only for practical purposes, not a kind of collectible: it is only for a single-use for certain purpose (e.g. getting a hint or unlocking a course area). It may (and usually should) have a defined validity period.
Content Discovery	Reveal secret content which was hidden beforehand.
Content Unlock	Unlock access to some content. Note: unlocking can be direct, i.e. granted as a reward for completing e.g. a course area, or indirect, by granting coupons or items that can be used to unlock content (possibly chosen by player), or as a result of player level having been increased.
Hint	Text (possibly also revealed snippet of a correct solution source code) displayed to player failing to solve a challenge: on his/her own choice (possibly paid with specific coupons) or automatically after a failure.
Congratulations	Text (possibly accompanied with visual and/or sound effects) congratulating player on achievements

3.4 Reward conditions

Table 3.4 Types of conditions upon which rewards are granted.

Concept name	Concept description	Applicable to (reward types)

Concept name	Concept description	Applicable to (reward types)
Attempt	Something player tries to achieve (e.g. opens an exercise).	Badge, Content Discovery
Achievement	Something player achieves (e.g. solves an exercise, completes a quest).	Point, Level, Badge, Virtual Item, Coupon, Content Discovery, Content Unlock, Held Record, Congratulations
Failure	Something player fails to achieve.	Hint, Content Discovery
Progress threshold	Player passes a specified threshold of a given progress metric (see section 4.3).	Level, Badge, Virtual Item, Coupon, Content Discovery, Content Unlock, Congratulations
Progress in competition	Player passes another player in a given progress metric (see section 4.3).	Current Rank, Held Record, Badge, Virtual Item, Coupon, Congratulations

3.5 Progress visualization

Table 3.5 The ways of visualization of participants' progress.

Concept name	Concept description
Player Profile	Students may have a public profile displaying their progress, latest achievements, and collected badges to make others envy.
Leaderboard	A list of players belonging to some group (e.g. participants of the same course) ranked based on selected metric (e.g. the number of collected points)
Avatar	Students may have an avatar. Its graphics would be displayed in Player Profile, and, next to his/her name, on Leaderboards and Forums. The avatar may be improved with student's progress (e.g. adding a crown above the head of the ranking leader's avatar).
Progress Bar	A bar showing the part of a course that has been completed so far by player.

4. Exercise-level Gamification Concepts

This section provides the proposed gamification-related concepts to be applied on the exercise-level.

4.1 Types of exercise-level challenges

Table 4.1 Types of gamified programming exercises.

Concept name	Concept description
Blank sheet	This kind of exercise provides a blank sheet for the student to write his/her solution source code from the scratch.
Code Extension	This kind of exercise provides partially finished solution source code (the provided parts are not subject to change by the student) which the student has to complete.
Code Improvement	This kind of exercise provides correct initial source code which does not yet achieve all the goals specified in the exercise specification, so the student has to modify it to solve the exercise.
Buggy code	This kind of exercise provides code with bugs (and failed tests) to foster the student to find the right code.
Fill-in the gap	This kind of exercise provides code with missing parts and asks students to fill them with the right code.
Mixed code	This kind of exercise breaks a solution into several blocks of code, mixes them, and asks to students to sort them (e.g., quicksort algorithm).
Show me	This kind of exercise defines a small set of primitives that can be used to solve the challenge (e.g., move('red','left') meaning move red block to left stack) and provides a visual animation of the code execution.
Spot the bug	This kind of exercise provides code with bugs and asks students to merely indicate the location of the bugs.

Table 4.2 Types of gamified programming exercise modes.

Concept name	Concept description
Shapeshifter	This kind of exercise is composed by a set of very similar exercises which switch between them when the metamorphosis timer end. This motivates students to solve the exercise fast but can also help if they are struggling with minor issues less related with what we want them to learn.

Concept name	Concept description
Shortening challenge	This kind of exercise provides a bonus reward for shortening the submitted solution to below specified number of lines offered if the submitted solution is correct but exceeds some threshold number of lines.
Speedup challenge	This kind of exercise provides a bonus reward for speeding up the submitted solution to below specified execution time, offered if the submitted solution is correct but exceeds some threshold execution time.
Hack the problem	This kind of exercise rewards students who solve the exercise in a tricky way without prior information about that.
Time bomb	This kind of exercise is only available for a certain amount of time once revealed.

4.2 Exercise-level badges

Table 4.3 Badges available to be granted on exercise level.

Concept name	Concept description
Hardworker	A badge awarded when a student fails more than n times to solve an exercise, but ends up with an accepted solution
Scientist	A badge awarded when the student makes several tests to check his solution, before submitting.
Keyword	A badge awarded when the student uses a specific keyword (e.g., array.map).
Straight	A badge awarded when the submitted solution has less or equal number of loops, for instance, than the specified correct solution.

4.3 Exercise-level metrics

Table 4.4 Exercise-level player activity metrics that can be used for gamification.

Concept name	Concept description
Solution Time	Time spent between starting a challenge and providing an acceptable solution to it.
Wrong Attempts	Number of attempts submitted before solving the exercise.
Correct Attempts	Number of correct submissions.

Concept name	Concept description
Memory Used	Memory usage of a program.
Execution Time	Execution time of a program.
Code Length	Size of the submitted code in characters.
Code Lines	Number of the submitted code lines.

Table 4.5 Exercise statistics that can be presented with its description.

Concept name	Concept description
Median Solution Time	Median time spent between starting a challenge and providing an acceptable solution to it.
Wrong Attempts	Number of attempts submitted before solving the exercise (of all students).
Correct Attempts	Number of correct submissions (of all students).
Least Memory Used	Minimum memory usage of a correct solution program measured.
Shortest Execution Time	Minimum execution time of a correct solution program measured.
Shortest Code Length	Minimum size of a submitted correct solution code in characters.

4.4 Miscellaneous

Table 4.6 Other exercise-level gamification concepts.

Concept name	Concept description
Time to relax a bit	When a student is struggling a lot to solve an exercise, show a 15 seconds animation encouraging him to look at the window and breathe deeply. It may also present an inspiring quote.
Exercise Narration	There may be a narration providing the motivation for solving the exercise.
Statistics	The exercise metadata includes updated statistics about students' results so far.

5. Course-level Gamification Concepts

This section provides the proposed gamification-related concepts to be applied on the course-level.

5.1 Types of course-level challenges

Table 5.1 Types of challenges spanning beyond a single programming exercise.

Concept name	Concept description
Duel	Students can challenge another online student for a 3 exercises' match. The exercises must be related to the concepts being studied. It may also involve exhaustible resources to limit the number of challenges.
Quest	A set of conditions on player's progress which when met result in granting a reward. Player is aware of his/her active quest and its requirements (and possibly can choose the next quest after completing one).
Streak	A sequence of time units in which certain goals were consistently achieved by player
Story	Set of exercises wrapped in a storyline which develops as students complete an exercise. May be connected to one or more quests.
Tournament	Instructors may schedule tournaments composed of a set of exercises where every student can enroll, and winners are picked according to a predefined criterion.
Mystery Track	Exercises that reveal additional exercises about the same (or related) concepts to offer more opportunity to practice.

5.2 Course-level badges.

Table 5.2 Badges available to be granted on course level.

Concept name	Concept description
Solver	A badge awarded by solving N exercises in a row without a wrong submission, for N in {3, 5, 10, 15, …}.
Man of duty	A badge awarded by following a streak for N time units, for N in {3, 5, 10, 15, …}.
Runner	A badge awarded to students who were the first to complete a full course or one of its modules.

Concept name	Concept description
Explorer	A badge awarded to students who revealed a specified number of secret content elements in a course.
Pathfinder	A badge awarded to students who were the first to complete a specified number of exercises in a course.

5.3 Player metrics

Table 5.3 Player activity metrics over the course level that can be used for gamification.

Concept name	Concept description
Exercises solved	Total number of exercises solved.
Solved first	Number of exercises that player solved the first.
Good series	The current number of exercises solved without wrong submissions in a row.
Streak length	Number of time units the current player's streak lasts.
Uncovered secrets	Number of hidden content elements uncovered in a course.
Active time	Time during which the student was active in the course.
Modules completed	Number of modules completed.
Total solution time	Sum of the time spent between starting a challenge and providing an acceptable solution to it

5.4 Miscellaneous

Table 5.4 Other course-level gamification concepts.

Concept name	Concept description
Most starred exercises	Students can give stars to exercises and show top exercises
Course coins	A kind of virtual item. Students may receive coins either for completing tasks or winning fights/tournaments. These can be used, for instance, to buy items for the avatar or book a seat in a tournament.

Concept name	Concept description
Achievements Feed	Feed with latest achievements of student and colleagues (e.g., X solved 5 exercises in a row, Y is missing 1 exercise to finish part A, ...).
H2H comparison	Students can compare certain statistics with another student in a H2H-like view.
Forum/Chat	Students may help each other through forum discussions or instant messages.
Course Narration	There may be a narration providing a background story for a course and linking its exercises in a logical manner.
Wheel of Fortune	Students who come to the course every day may get a chance to win a collectible item or virtual economy's value. This chance may be higher for students who have tried to help others in the day before.