

# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,100

Open access books available

167,000

International authors and editors

185M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index  
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?  
Contact [book.department@intechopen.com](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.  
For more information visit [www.intechopen.com](http://www.intechopen.com)



# Game Development and Testing in Education

*Emília Pietriková and Branislav Sobota*

## Abstract

The following chapter presents an experience with academic education in computer games design and development based on game theory, emphasizing iterative development, rapid (paper) prototyping, and game testing. We demonstrate and comment on various educational activities focused on game testing, describing them in steps and including benefits and motivation. Iteration of card game rules, design of a board game, design of a computer game with a paper prototype, and testing of early-access games are supplemented with students' practical outputs in both text and graphical form. The results include feedback in the testing forms: Focus Groups, Playtesting, Usability Testing, and Quality Assurance.

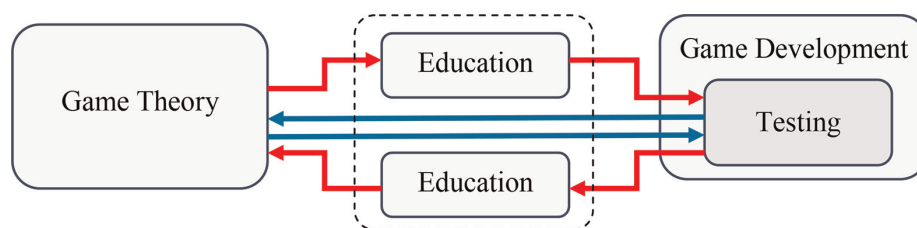
**Keywords:** game design, game theory, game testing, playtesting, education

## 1. Introduction

In addition to forming a game development basis, Game Theory can significantly affect the development of computer games. Some types of games have been known in Game Theory for a long time. However, some procedures are modified concerning modern technologies, allowing one to quickly solve some (up to now) complex tasks, e.g., independently of the computer graphics use. It is excellent when some principles from this theory are applied in the education process of computer games development, especially in the area of computer science education.

In terms of influence (**Figure 1**), we can observe mainly the following approaches to computer games testing:

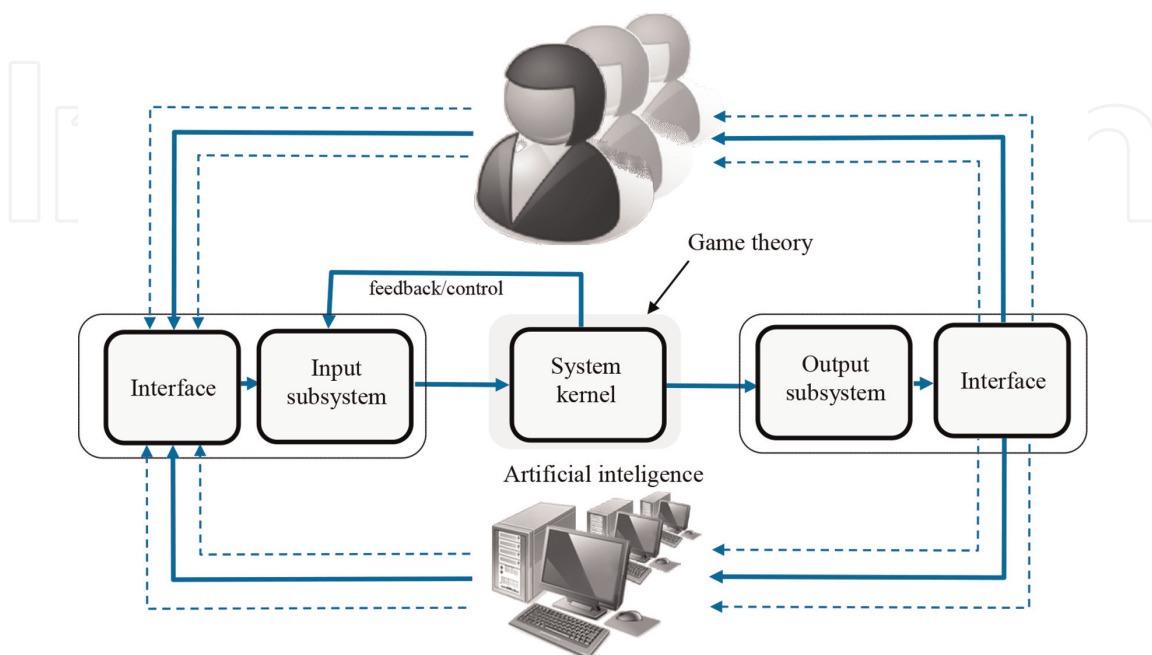
- *Direction GTh (Game Theory) → GTe (Game Testing):* The Game Theory formalism defines the framework of Game Testing. It can determine what to



**Figure 1.**  
*The effect of game theory on game development and vice versa.*

focus on during testing, notably with a specific type of game or what testing parameter to use. The formalism can specify possible threats and other aspects when including a SWOT analysis.

- *Direction GTe (Game Testing) → GTh (Game Theory)*: In this case, the testers define the game evaluation parameters by recording their notes (text, voice, video), which are subsequently processed and structured. Game designers and developers can afterward perform simulation and evaluation to optimize the game, game loop (**Figure 2**), improve its design, or change the formal apparatus. In addition, it is possible to obtain feedback within the game loop for inputs or outputs adjusting, including the possibility of adjusting the implementation of artificial intelligence, e.g., in the form of bots. The game loop implemented in this way also makes it possible to apply separate/simultaneous testing of individual blocks, and thus it increases the efficiency and speed of the testing itself. It also affects better processing, e.g., incidence matrix. In a game loop structured in this way, e.g., at the input level, it is also possible to implement various scripting languages, in the case of inserting automatic test scripts during game development. Of course, the elements of the Game theory are mostly applied in the System kernel. This process can help compare to what extent the game design corresponds to the formalism of Game Theory (Is the game playable, is it difficult, is the game and its reward strategy motivating, or does the symmetry of the game change?).
- *Direction Ed (Education) → GTe and GTh*: Determines how to teach CS students, so they can appropriately use the GTh knowledge in GTe, and subsequently in the practical development of computer games. This direction substantially impacts students in the sense of the GTh education and gamification, so they are not only “flooded” by the mathematical apparatus but can also transform it into programming practice.



**Figure 2.**  
Standard process of player(s) interaction in game loop.

Regarding teaching computer games development, some *GTh* results can help define roles from a *GTe* perspective (e.g. Observer, Playtester, Bugfixer). Each one may have its formal apparatus or different parameters (values, methods of use, or semantics), depending on how the result supported by the theory applies to individual *GTe* roles, rules, strategies, or incidences.

At the Department of Computers and Informatics FEEI Technical University of Košice, we have implemented the *Design and Development of Computer Games* and related courses for several years. To be successful in implementation, the CS students, as future creators of computer games, need to have at least the basic apparatus of Game Theory. As part of the education process, the students become familiar with various methodologies and technologies related to computer games design and development: high concept, market research, storytelling, prototyping, rules balance, MDA, or iterative game development, an essential part of which is *GTe*.

In the chapter, we would like to present the primary forms of this education process, focusing on *GTe*, including some of the technologies used. We will also present some forms and demonstrations of *GTe* performed by students, corresponding mainly to the first two directions. We processed some results with an SUS questionnaire using a 5-point Likert scale.

We will present activity outputs recorded during the actual education process with 3rd-grade students of the bachelor's degree in Informatics in a full-time attendance form of study. We believe the experience presented in this chapter will find its readers. We attach great importance to the fact that these are actual data processed over multiple years and thus can provide a basis for other studies. We will comment on the outputs of student activities from the educational process point of view.

## 1.1 Iterative game development and testing

Testing is one of the critical processes accompanying the design and development of computer games. Game designers and developers often confuse or simplify different testing processes, leading to unsatisfactory results, e.g. for students who encounter testing as part of the educational process, it is most often confusion of *playtesting* with *bugfixing* (*quality assurance*) or playtesting with *usability testing* (*interface*). Therefore, in the next chapter, we will formally clarify the testing process and present one of the ways to incorporate various forms of testing into the educational process in computer game design and development.

Well-known literature states that four types of testing are crucial to the design and development of computer games [1, 2]:

- *Focus Groups*: They play a role, especially in the initial stage of game design, consisting of meetings with potential players providing feedback, likes, and dislikes on the upcoming game topic. The literature [1] states that although Focus Groups can be helpful, they are unpopular. The reason is management's fear of "killing" their ideas, so the book [2] recommends switching from obtaining one-sided feedback to discussions generating new ideas. Such Focus Groups will allow iterating on existing ideas more effectively. Moreover, it does not have to be only the initial stages of game design. They can also be helpful later, e.g., when designing game levels or planning the marketing of an upcoming game, which is especially crucial in the final stage of game development.

- *Quality Assurance (QA) Testing*: This is standard bugfixing. The goal is not to play and enjoy the game but to identify all the problems and make sure the game meets all the predetermined requirements [3]. QA engineers look for potential bugs and report them. Since games are not a common type of software, a video, image, and detailed description can help understand steps, leading to the error situation.
- *Usability Testing*: One of the final aspects of a tested game is accessibility. Usability Testing focuses on whether the user interface is intuitive and easy to use. Players must understand and play the game without assistance. For this reason, testers best test the game during the first-time access [2]. Ideally, User Experience (UX) experts perform the testing.
- *Playtesting*: Often referred to as the most important of the four testing types. It is a process that designers and playtesters carry out throughout the game design to determine how the players experience the game. The team of designers and developers proves they are making a good game, even if it scares them [1]. Playtesting aims to guide the design and provide feedback to the designers on whether the game meets their goals and the players' expectations [4]. The forms are different, more or less formal.

*Game User Research (GUR)* intends to improve player experience [4]. *GUR* is a community of UX experts, game developers, and researchers, where the primary research subject is playtesting. In general, *GUR* supports the adaptation of standard HCI evaluation techniques in the playtesting practice, focusing on “entertainment” instead of “productivity” [5]. From the point of view of software engineering methods, *GUR* mainly recommends the *RITE* method [6]. The research applies other sophisticated methods such as *PLAY*, adapting various heuristic methods [7], or using artificial intelligence - AI Players. The goal is to predict player behavior and gaming experience [8].

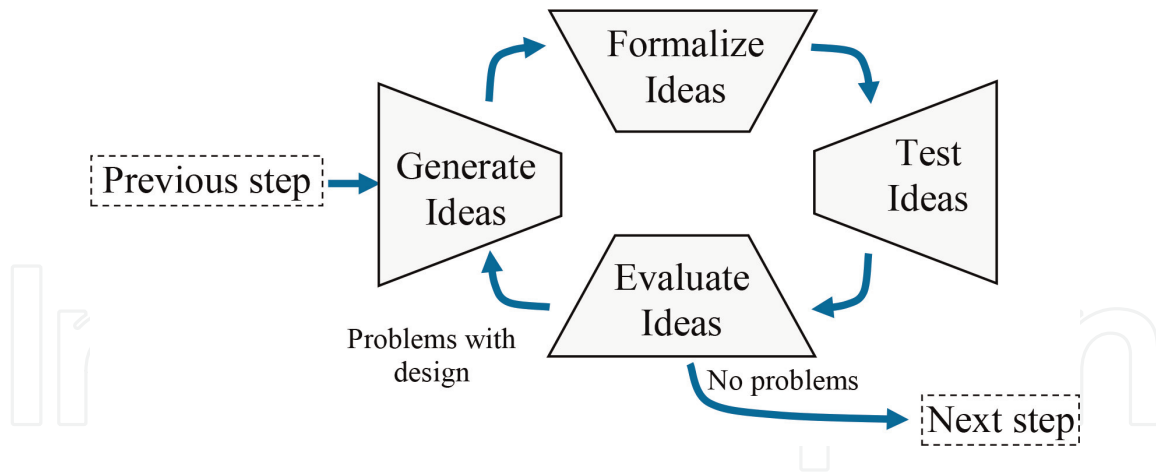
As already mentioned, testing is critical in the iterative process of game design and development. It helps detect problems early (it is still possible to fix them). But what exactly is an iterative process?

Rapid Iterative Testing and Evaluation (*RITE*) is a software development method whose main philosophy is to identify a problem as soon as possible and eliminate it. During the last two decades, the *RITE* method has become widespread, especially for products changing rapidly e.g. web, cloud products, and games [6, 9].

It is too late to test at the end of game development. Otherwise, the changes would be too lengthy and costly. The *RITE* method guarantees early identification and elimination of the problem [2]. It is mainly adopted by larger game studios and recommended by the *GUR* community.

There are multiple testing types in game development. Playtesting is key to the *RITE* method, although other types of testing also find their place in individual stages of game design and development (**Figure 3**). From a practical point of view, it is best for playtesting if playtesters can play the game as soon as possible. Rapid prototyping (including paper prototypes) is advantageous if the game is still in the design stage.

A prototype is a simplified but testable game version. Designers create it before they spend money on implementing game elements and features. In general, there are three types of game prototypes [10]:



**Figure 3.**  
*Process of iterative game design.*

- *Software*: Prepared in a game software like Unity,
- *Physical*: Physical interaction and activity with the playtester, and
- *Paper*: Board game, the most common type.

With smaller game industry studios, we encounter risk, so the game is published as soon as possible. In this case, they tend to skip the prototype and playtest only at the end of game development. Many rely on the fact that most game elements are copies of existing elements from other games. This risk results in the publication of many small, imperfect games. Fortunately, the game industry is changing thanks to the GUR community.

For playtesting, which is one of the game testing types, two people are essential:

- *Investigator (Observer)*: Person who manages the test,
- *Playtester*: Person who participates in testing and provides feedback.

The Investigator should talk to the players as little as possible to not influence their evaluation and create a questionnaire. The *Investigator* can track the answers' development over time after further iterations. A good *Investigator* should be able to identify playtesters' exit points and not be sad if they do not like the game ideas. The author of [1] recommends a scale of 1–5 instead of 1–10 or 1–7: *Terrible, Pretty bad, So-so, Good, and Excellent*.

On the contrary, the *Playtesters* should think out loud while playing, so the *Investigator* can record their thoughts. They should reveal their prejudices and self-analysis. So, in addition to evaluating something positively or negatively, they should also state the reasons. Good *Playtesters* should be able to comment on various game elements like graphics, music, story, mechanics, etc.

## 2. Game testing in education

As we have already mentioned, in the education process during game testing, we often encounter students confusing playtesting with *bugfixing* or *playtesting* with

*usability testing*. Therefore, in the next chapter, we will introduce multiple educational activities in designing and developing computer games, focused on implementing various forms of testing. The goal is to provide students with an experience and make them aware of the differences. The activities proceed from our personal experience with software engineering students in the course of Computer Games Design and Development. We assume the students have already mastered software engineering principles, including traditional software design, development, and testing.

In the following subchapters, we will present multiple outputs of student testing activities. Testers' opinions and feelings are part of testing (especially playtesting). Therefore, the listed results also contain the opinions and feelings of selected students and may not always coincide with our opinions and feelings.

## 2.1 Card game: playtesting & focus groups

Since computer game design and development correspond to Game Theory, it is appropriate in the education process if students work with typical card or board games. Analysis and playtesting of card games are especially suitable at the beginning of such courses.

**Goal:** To achieve the best possible gaming experience in the card game.

**Resources:** Card games (or board games).

**Testing:** Playtesting & Focus Groups.

**Methods:** Basic rules iteration, incidence matrix, SWOT analysis.

**Length:** 80–90 minutes.

### Steps:

- Students form teams of 3–5 players.
- Each team selects one existing card game, which they play for about 20–30 minutes. If the game is fast, teams can play it multiple times in a row.
- Group analysis of the game and feedback on the game experience. Here are some steps we recommend:
  - Write a list of card game rules.
  - Write short feedback on the gaming experience, including both positives and negatives.
  - Write SWOT analysis.
- Group iteration of the basic rules to improve the gaming experience.
- Shuffle teams. In each group, 1–2 original players will remain, and 2–3 will shuffle with other groups.
- New teams play card games with new rules.
- Group analysis of the game and feedback on the game experience:

- Write short feedback on the gaming experience, including both positives and negatives.
- Update a SWOT analysis.
- If necessary, prepare a new iteration.
- Compare game experiences both before and after game rules iteration.
- Compare SWOT analysis both before and after game rules iteration.

Feedback on the card gaming experience prepares students for playtesting computer games. Since card games are usually short and fast, students are often unaware of various game aspects, subsequently missing in their feedback. That's why we decided to do a SWOT analysis, the use of which is unconventional in this case. From our experience, the SWOT analysis multiplies the effect of the entire activity.

Since the activity consists of group steps, we can also talk about training Focus Groups. Here we follow the recommendation of [2] that Focus Groups participate in creating new ideas. So, in addition to feedback, students in teams prepare iterations of game rules, coming up with new ideas.

Working with a card game guides students from Game Theory to Game Design. At the end of the activity, they compare playtesting and SWOT analysis both before and after the basic rules' iteration. It points to the quality of the iterations with which the students will develop their evaluation. An alternative could be the same activity with an existing board game.

In the next subchapters, we will present multiple outputs of this activity.

### 2.1.1 Bang!

**Basic rules:** Bang! is a wild-west themed card game for 4–7 players in the basic edition. The goal is to kill the enemy according to the roles drawn at the beginning of the game. The sheriff must eliminate all the bandits and the renegade with the help of his assistants. The bandits must kill the sheriff before he kills them. The renegade wants to become the new sheriff, so his goal is to be the last one alive. So, he first takes out all the bandits, sheriff's assistants, and finally, the sheriff. While the sheriff is the only role revealing his identity at the beginning, the other roles hide their identity until they die.

**Positive playtesting comment:** Can entertain multiple players. It is diverse and strategic.

**Negative playtesting comment:** The game is difficult for newcomers. It can seem long-winded.

#### SWOT analysis:

	Useful attributes	Harmful attributes
Internal attributes	<ul style="list-style-type: none"> <li>• The rules are easy to learn.</li> <li>• If players understand the rules, the game is easy and intuitive.</li> <li>• The mistake should not occur since all players can see the steps of teammates and opponents.</li> <li>• Easily portable game.</li> <li>• It supports strategic and team thinking.</li> </ul>	<ul style="list-style-type: none"> <li>• Needs at least four players.</li> <li>• Some characters from the base pack are not entirely balanced.</li> </ul>



	Useful attributes	Harmful attributes
External attributes	<ul style="list-style-type: none"> <li>• The game with 6–7 players is more entertaining.</li> <li>• Cards quality (material).</li> <li>• If any card is lost, the game is still playable.</li> <li>• Add-on packs make the game more exciting and complement it with existing and new cards.</li> <li>• The possibility of making players' own cards.</li> </ul>	<ul style="list-style-type: none"> <li>• The game with four players becomes repetitive after a while.</li> <li>• Eliminated players can get bored.</li> </ul>

### Iterations:

1. *Four players:* In the basic game, it is easy to find players' roles after a while. There is the sheriff, vice, and two bandits. The sheriff will show his role before the game starts, so it is easy to determine who are a bandits and who are not. An alternative could be to shuffle the cards by adding a renegade instead of a bandit. The renegade must be the last player left in the game, so he is expected to switch sides

*Playtesting:* Iteration made the game harder. It was not immediately clear who had what role.

2. *Three players:* When the sheriff is against two bandits, the player of the sheriff is at a significant disadvantage so that he would have multiple advantages in addition to the original game. He would have two more lives (in the base game, the sheriff always has one more life). Likewise, he would draw one more card (in the base game, it is two cards)

*Playtesting:* After the iteration, the sheriff player was no longer at a disadvantage, which positively affected the game balance.

3. *Zombie Bang!* After his death, a person becomes a zombie. He has the same traits and cards but joins the person who killed him. This way, the game will speed up considerably and become more entertaining. Even eliminated players can continue to play

*Playtesting:* The game had more dynamics after the iteration. However, there was a problem that the game was too fast.

4. *Gangs:* Players remove the original roles and divide into two or more gangs that will fight against each other. Each has its leader, hidden from others and known only to his fellow members. The goal of the gang is to kill the leaders of the other gangs. If the leader dies, the entire gang automatically loses

*Playtesting:* This iteration allowed multiple players to play the game. However, with a smaller number of players, the game became repetitive quickly. Therefore, the minimum recommended number of players for this iteration is 6.

5. *New Sheriff:* If the renegade manages to kill the sheriff before the game ends, he can become the new sheriff, and the game continues. This rule can be extended, for example, that the sheriff's assistants become new renegades, or the original sheriff becomes the new renegade

*Playtesting:* Iteration finds its use in case the game ends quickly. After the iteration is applied, the game becomes longer and continues to be dynamic, which was the goal of the iteration.

6. *Deathmatch*: An alternative could be a form of a deathmatch where all players would play against each other regardless of role, which is unnecessary in this case

*Playtesting*: The game was more dynamic after the iteration.

7. *Remove weapon restrictions*: The players remove the original condition to fire only at players within a given range

*Playtesting*: With this change, we increased the game dynamics and shortened its duration.

### SWOT analysis after iterations:

	Useful attributes	Harmful attributes
Internal attributes	<ul style="list-style-type: none"> <li>• The rules are easy to learn.</li> <li>• If players understand the rules, the game is easy and intuitive.</li> <li>• Easily portable game.</li> <li>• It supports strategic and team thinking.</li> </ul>	<ul style="list-style-type: none"> <li>• Needs at least four players; in some versions, at least six players.</li> </ul>
External attributes	<ul style="list-style-type: none"> <li>• The game with 6–7 players is more entertaining.</li> <li>• Cards quality (material).</li> <li>• If any card is lost, the game is still playable.</li> <li>• Add-on packs make the game more exciting and complement it with existing and new cards.</li> <li>• The possibility of making players' own cards.</li> </ul>	<ul style="list-style-type: none"> <li>• Eliminated players can get bored.</li> </ul>

#### 2.1.2 Black Peter

**Basic rules:** Game for min. 3 players, 33 cards: 32 + 1 (Black Peter). The aim is to collect pairs of identical cards. The players then draw random cards from the left player. The player who has only Black Peter left in his hand loses the game.

**Positive playtesting comment:** The game is simple for players of any age, has a fast course, and offers a balance of chances.

**Negative playtesting comment:** When playing, we revealed low variability of the gaming experience along with high repetitiveness and the impossibility of tactics.

### SWOT analysis:

	Useful attributes	Harmful attributes
Internal attributes	<ul style="list-style-type: none"> <li>• A robust rule checking mechanism.</li> <li>• Portable and re-playable.</li> <li>• Simple rules for a different number of players.</li> <li>• A quick game for different ages.</li> </ul>	<ul style="list-style-type: none"> <li>• It is possible to play the game endlessly since the selection of cards by the players is random (for example, when there are two players with three cards, they can move one card around).</li> <li>• The game is repetitive.</li> </ul>
External attributes	<ul style="list-style-type: none"> <li>• It is not necessary to concentrate on the game, and it is possible to have a conversation during the game.</li> <li>• It is possible to play the game even if a card in the pack is lost.</li> <li>• It is possible to play the game with any deck of cards (Pharaoh cards, Joker cards).</li> </ul>	<ul style="list-style-type: none"> <li>• Due to the randomness of the game, one player can lose repeatedly.</li> <li>• Cards can get damaged.</li> </ul>

### Iterations:

1. *Action pairs*: Upon completing a pair, the player gets the option to “unload” it and perform an action that the pair allows. A player can only unload pairs after the move.

- Cards with a blue background have essential functions without actions.
- Cards with a purple background will change the game direction.
- Cards with a yellow background will allow a player to exchange one of their unwanted cards with any player’s card.
- Cards with a green background will move all cards in the player’s hand in the game direction.

*Playtesting*: New rules have increased the dynamics of the game. We found a problem where their given action was not always suitable to perform after unloading a pair.

2. The rules changed, so the player does not have to use a particular action after unloading an action pair

*Playtesting*: The game became calm. However, players were afraid to use the actions.

3. *Suggested action*: Draw one card from each player. The proposed change includes a new type of action pair with a red background, whose particular action the player must perform. These actions will be deliberately designed to be disadvantageous and annoying to players

*Playtesting*: We were pleased with the new dynamics and rule change, and the game provided an entertaining gameplay experience.

### SWOT analysis after iterations:

	Useful attributes	Harmful attributes
Internal attributes	<ul style="list-style-type: none"> <li>• A robust rule checking mechanism.</li> <li>• Portable and re-playable.</li> <li>• Simple rules for a different number of players.</li> <li>• A quick game for different ages.</li> <li>• Players need to think more.</li> </ul>	<ul style="list-style-type: none"> <li>• Small children may have a problem with such a game (they may forget some of the rules).</li> </ul>
External attributes	<ul style="list-style-type: none"> <li>• The game is more fun.</li> <li>• It is possible to play the game even if a card in the pack is lost.</li> </ul>	<ul style="list-style-type: none"> <li>• Cards can get damaged.</li> </ul>

## 2.2 Board game: playtesting & focus groups

Board games are suitable as introductory activities in game design courses, like card games. This activity is similar to the card game activity. This time, students are not working with an existing game but creating a new, custom board game that they play, analyze, and iterate with each other.

**Goal:** To achieve the best gaming experience when designing a board game.

**Resources:** Paper, pencil, figures, cubes.

**Testing:** Playtesting & Focus Groups.

**Methods:** Game design, iteration of basic rules, incidence matrix, SWOT analysis.

**Length:** 80–90 minutes.

**Steps:**

- Students form teams of 3–5 players.
- Each team designs and creates a board game on paper. The team can play the game; however, it is not crucial to this activity.
- Each team writes the rules of the board game.
- Shuffle teams. In each group, 1–2 original members will remain, and 2–3 will shuffle with other groups.
- New teams play board games for about 20–30 minutes. If the game is fast, teams can play it multiple times in a row.
- Group analysis of the game and feedback on the game experience. Here are some steps we recommend:
  - Write short feedback on the gaming experience, including both positives and negatives.
  - Write SWOT analysis.
- Group iteration of the basic rules to improve the gaming experience.
- Shuffle teams. In each group, 1–2 original members will remain, and 2–3 will shuffle with other groups.
- New teams play board games with new rules.
- Groups analysis of the game and feedback on the game experience:
  - Write short feedback on the gaming experience, including both positives and negatives.
  - Update a SWOT analysis.
  - If necessary, prepare a new iteration.
- Compare game experiences both before and after game rules iteration.
- Compare SWOT analysis both before and after game rules iteration.

Working with a board game guides students from Game Theory to Games Design, like card game activities. Students practice playtesting a board game design

(Figure 4), which is similar to playtesting a computer game design. Again, the SWOT analysis multiplies the effect of the entire activity.

Since the activity consists of group steps, we can also talk about training Focus Groups. At the same time, working with hand-made board games is similar to playtesting with paper prototypes of computer games, preparing students for the following course stages, which are focused on designing and developing computer games.

At the end of the activity, students compare playtesting and SWOT analysis both before and after the basic rules' iteration, developing a self-evaluation again.

### 2.3 Rapid prototyping: playtesting

This activity moves beyond Game Theory and directly relates to Game Design. After the initial processes like High Concept or Pitch, the game environment, rules, or character design comes next. Since changing these aspects of the game at a later development stage is quite expensive, it is essential to playtest the game as soon as possible.

**Goal:** To achieve the best possible gaming experience when designing a computer game.



**Figure 4.**  
*Board games activity in the classroom.*

**Resources:** Paper, glue, scissors, crayons, figures, Lego.

**Testing:** Playtesting.

**Methods:** Game design, basic rules iteration, incidence matrix, paper prototyping.

**Length:** 180+ minutes.

As mentioned earlier, prototyping is part of computer games' iterative design and development. Students must realize the importance of this stage through the paper prototyping activity.

**Steps:**

- Students work in existing teams of 3–5 designers and developers.
- Each group already has the basic game proposal, consisting of basic rules, characters, and environment, all prepared during some previous activities.
- Each group prepares a paper prototype for the proposed game (60–120 minutes).
  - Work with paper, glue, scissors, and crayons.
  - Students' creativity is welcome, e.g. they can use various figures or Legos.
- Shuffle teams. In each group, 1–2 original members will remain, and 2–3 will shuffle with other groups.
- New teams play prototype games for about 30 minutes.
- Group analysis of the game and feedback on the game experience. Here are some steps we recommend:
  - New team members are playtesters and they play the game. They talk out loud.
  - The original team members have the role of observers. They talk as little as possible.
  - Write short feedback on the gaming experience, including both positives and negatives.
- In their original composition, the teams analyze the playtesting results.
- In their original composition, the teams iterate aspects of the initial game design.
- If there is enough time, students can update the prototype and playtest the game again.

Testing is an integral part of the software life cycle. In the case of computer games, the design and development process is diverse from traditional software. We emphasize iterations, testing, and evaluation at each stage of development (RITE). A prototype is a practical means of testing, enabling one to carry out the playtesting before

the game development begins. Preparing a prototype is faster than implementing an actual computer game, so (not only) the students can get to playtesting much earlier (**Figure 5**).

Feedback on the prototype's gaming experience helps students iterate their computer game's design (**Figure 6**). Reminding individual teams not to be disappointed by negative feedback is significant as one of the playtesting goals is to help game designers but not to demotivate them. Identifying the playtester's "leave" is crucial since we do not want to know that something is good or bad but also why it is good and bad. So, in addition to thinking out loud, the playtester should reveal his prejudices and adequately justify each impression.

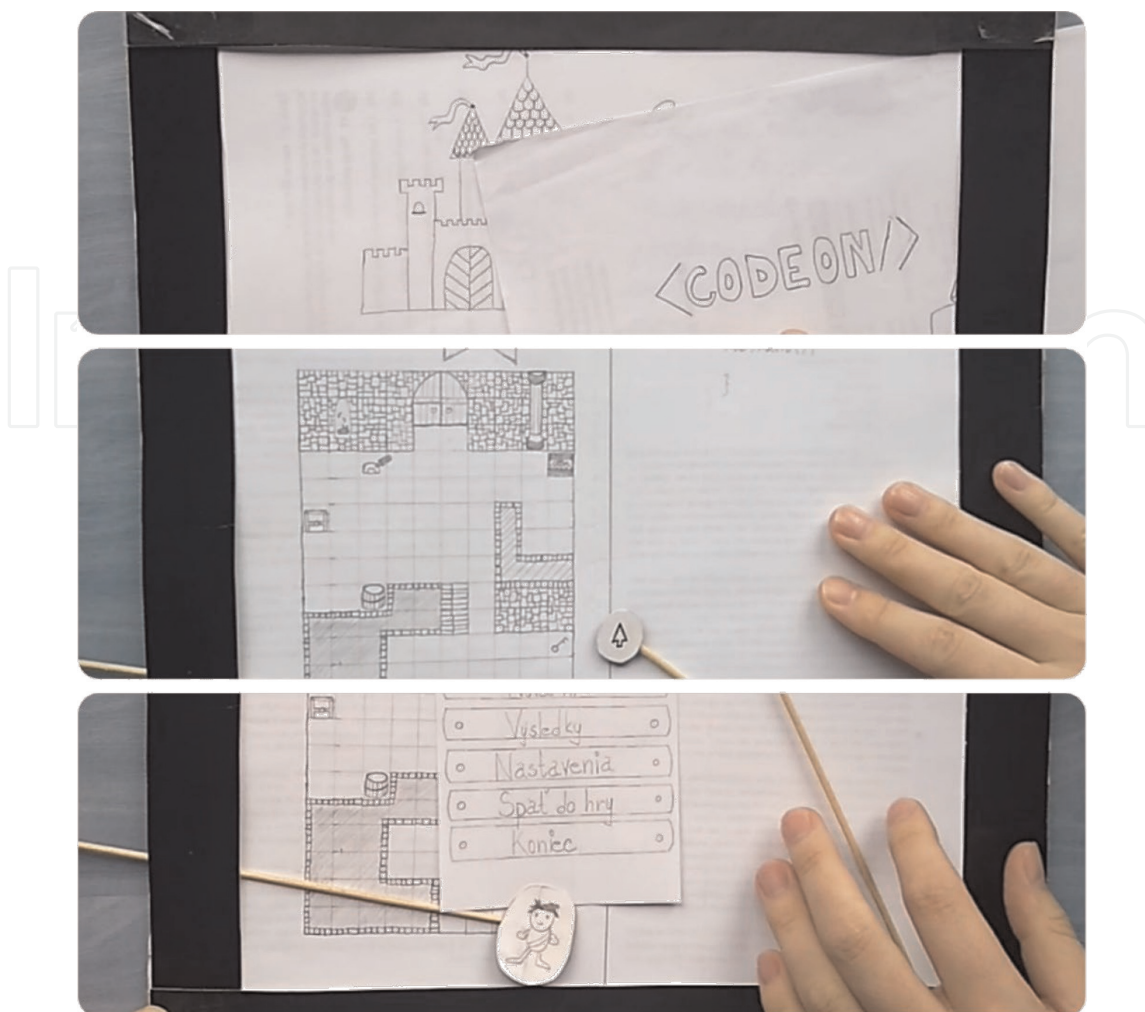
In this way, in addition to creating paper prototypes, students practice playtesting computer games, where they have the opportunity to work as both playtesters and observers.

## 2.4 Early-access games testing: playtesting, QA, UX

This activity aims at testing games at the end of their development. Each student should work as both a tester and an observer, that is, to try testing from both sides. Since three types of testing play a role at the end of game development, students work with all three: *playtesting*, *usability testing*, and *bugfixing* (QA).



**Figure 5.**  
*Paper & Lego prototypes activity in the classroom.*



**Figure 6.**  
*Example of game screenplay on paper prototype (Slovak menu).*

**Goal:** Practice the role of playtester and observer, achieve the best gaming and user experience, game debugging.

**Resources:** Computer, computer game.

**Testing:** Playtesting, Usability Testing, Quality Assurance (QA).

**Methods:** Questionnaire (SUS), thinking out loud, reporting.

**Length:** 180–240 minutes.

This activity can have two versions. If students create their games on the course, they test them among each other. If the course has an early-access game available, students test it and report it to the game developers. If possible, students get feedback on their testing from real game developers.

**Steps in playtesting of early-access games:**

- Students work in pairs.
- One student is an observer; the other is a playtester.
- Playtester prepares an information sheet with helpful information for the game authors:



- Technical parameters of the OS, computer, and other devices such as resolution, etc.
- Self-Persona of the playtester (characteristics, self-analysis).
- Students will also use the information sheet for other types of testing.
- A playtester plays the game and comments on his gaming experience out loud.
- The observer records observations from the gaming experience and does not interfere with the playtester.
- Together, the students write a playtesting report containing:
  - The positives of the gaming experience.
  - The negatives of the gaming experience.
  - Other observations.
- The playtesters fill out the questionnaire if requested by the development team.
- Each playtesting is different, which is why each playtesting is helpful for developers in another way. Therefore, feedback from the developers is essential (if they are willing, which is usually not a problem).

#### **Steps in playtesting of student games:**

- Students work in pairs or threes.
- One student is an observer and belongs to the game development team.
- One or two students are playtesters and do not belong to the development team.
- Playtesters play the game and comment on their gaming experience out loud.
- The observer records observations from the gaming experience and does not interfere with the playtesters.
- The development team analyzes the playtesting and prepares new game iteration based on the results.

#### **Steps in usability testing:**

- The procedure is similar to playtesting, focusing on the user interface.
- In the case of a student game, the development team prepares a SUS questionnaire with a Likert scale of 1–5.

- Testers add observations related to the game usability to the existing playtesting report.
- Testers prepare a video or an exact sequence of steps to describe problem situations better.
- The testers fill out the questionnaire if requested by the development team.
- In the case of a student game, the development team analyzes the testing and prepares a new game iteration based on the results.

#### **Steps in bugfixing:**

- There is no need to work in pairs or speak out loud in this part.
- Each student prepares an information sheet with technical parameters of the OS, computer, and other devices such as resolution, etc.
- Each student independently goes through the game multiple times and tries to interact with as many game elements as possible.
- Each student prepares a report and, if necessary, supplements it with a video or an exact sequence of steps in problem situations.
- In the case of a student game, the development team analyzes the testing and debugs the game implementation.

The essential part of this activity is that students practice the three types of game testing, being on both sides: testers and developers. So, students implement early-access game testing in a broader context.

We have already described the essence of playtesting. The observer should be able to identify the player's "leave" but should not be offended by any negative feedback. On the contrary, the playtester should think out loud, be able to reveal his prejudices, and justify why he likes or dislikes something about the game.

Students can test the game in an early-access version if one is available. Or they can test their games with each other. It all depends on how the educational course is structured.

In the next subchapters, we will present multiple outputs of this activity.

#### *2.4.1 Testing of early-access game bushfires: animal rescue*

This chapter presents an example of game testing of the early-access game carried out in 2021. The game is currently available on the Steam platform.

**Game:** Bushfires: Animal Rescue.

**Developer & Publisher:** Flying Butter Games.

**Version:** Early-Access.

#### **Playtester 1, male:**

- *Characteristics:* I consider myself 1/3 Achiever and 2/3 Explorer. I'd say I'm a Mid-core gamer, and I like a big open world to explore. I like games where the player has

to fulfill various tasks within a level. While playing, I want to logically think about how to pass the level with the highest score, and I enjoy the high-level difficulty.

- *Self-analysis*: For me, the important thing when deciding whether to buy/play a game is whether the game has an idea (it wants to draw the player's attention to a burning problem or sell the player a vital truth). I appreciate it when the developers leave a section in the game where they summarize the information about the theme on which they based the game (for example, in the game Brothers in Arms: Road to Hill 30, the developers left a database of official reports from military missions, according to which they created individual levels/missions in the game).
- *Top games*: Assassin's Creed: Brotherhood (Ubisoft/Ubisoft Montreal), Euro Truck Simulator 2 (SCS Software), LEGO Pirates of the Caribbean: The Video Game (TT Games Limited), Sims (all series, Electronic Arts), Brothers in Arms: Road to Hill 30 (Gearbox Software/Ubisoft), Blitzkrieg (Nival Interactive/1C Company).

Device	Resolution	OS	Memory	Graphics	Processor
PC	1920 × 1080	Win 10 (x64)	8 GB DDR4 RAM, 256 GB SSD	nVIDIA GeForce GTX 1050 2GB	Intel Core i5 7300HQ 2,5 GHz

### Playtester 2, female:

- *Characteristics*: I'm not a regular computer gamer, which means I play a game once a month, sometimes not even that. I cannot get obsessed with games and can play a game for 30 minutes. If the game is multiplayer, I can play it longer without getting bored. Most of the games I would play are from my "childhood," that is, older games. I do not like games where I only have to get from point A to point B.
- *Self-analysis*: If I must choose a new game to play, the visuals and overall graphics of the game and the sound of the game are essential to me - examples are Limbo or Ori And The Blind Forest.
- *Top games*: The Sims 1,2 (Electronic Arts), NHL 06 (EA Sports), Limbo (Playdead), Zoo Tycoon 1 (Blue Fang Games), Worms (Team17).

Device	Resolution	OS	Memory	Graphics	Processor
Note-book	1920 × 1080	Win 10 Home (x64)	8 GB DDR4, 128 GB M.2 SSD	NVIDIA GeForce 930MX/2GB	Intel Core i7 8550U

### Playtesting:

- Positive remarks:
  - A fascinating idea for the game. I like that it also points to a situation that happened and shows the players what Australia had to go through.

- It's great that one can also find basic information about animals that have already been discovered and rescued.
- Nice graphics. I like the animals look pretty realistic, and the environment is also very nicely depicted.
- I like the road signs, navigating where to go or what animals are on the road.
- Pleasant music that is not annoying.
- Sound effects when painting a car.
- Negative remarks:
  - I find the first map (level) more challenging, considering it is the first contact with the game. It would be nice if the first level was in the form of a tutorial or had an infinite time so the player could master the controls in peace.
  - I would not use the sound when selecting the level in the case of the locked ones.
- Other remarks:
  - The color design gives a person the impression of dryness and fire.
  - The game could include some checkpoints, although they might be difficult to implement now since time is an essential factor, and they could make it too easy to pass the level.

### **Usability testing:**

- At first, the car was sensitive to the controls (reminiscent of the ATV controls in the game GTA: San Andreas), but after one gets used to it, the game has a pleasant control. It is nice that it is possible to use both the keyboard and the mouse.
- I like that when hunting animals, time slows down, so I can better predict where the animal will probably go and think where to shoot the net.
- The colors of the animals are remarkably similar to the environment, which makes them difficult to see.
- It is more difficult to find if the animals are in the fire. It would be good if the animals were marked near the car, just like in the distance, and show the distance from the vehicle. As long as they do not move and the car gets close, the marker indicates where the pet is, and the player knows where to look for it, but once the pet gets closer and moves, it's harder to find (**Figure 7**).
- I'm missing a minimap somewhere in the corner that would speed up navigation. Add a minimap like in GTA or Mafia games, where animal icons would be displayed showing where the animals are located.



**Figure 7.**  
*Example of bushfires game - unmarked animals.*

### **Bugfixing (QA):**

- *Bug 1 & steps:* When I open the settings during the game and use the ESC key when going back to the game, it does not bring me back to the game, but it brings me back to the settings while I can no longer access the settings via SETTINGS. To go back to the game, I must go back to the menu via BACK from the settings and choose RESUME GAME or return from the settings to the menu and choose RESUME GAME; then it throws me back to the settings, and I choose BACK. (Students attached video).
- *Bug 2 & steps:* In level 4, the target and arrow are not visible in some places while shooting the net to catch the animal. It happens, for example, between two bridges (kangaroos) or at a fence under a hill (koalas). What is interesting is that in these places, the target is not visible if the car is going in the right direction; however, if the vehicle is going in the opposite direction, the target is visible. (Students attached video).
- *Bug 3 & steps:* In level 4, animals disappeared from the map several times. In the video, you can see how the koala disappeared and how the mark of the dingo dog is displayed at first, but then it disappears, and even after leaving, it is impossible to see his mark. (Students attached video).
- *Bug 4 & steps:* I accidentally drove the car through the texture: Over the fence on the bridge (level 3) and through the bridge (level 4). (Students attached video).
- *Bug 5 & steps:* The point that indicates where we are on the map at the end of the 3rd level has disappeared.
- *Bug 6 & steps:* On the map of Australia where the levels are displayed, I can hear sound effects when moving the mouse over them on the first load. When I pass a round or go to the main menu and click CONTINUE, I cannot hear them (besides the music in the background). I cannot hear them even when choosing the color of the car.

- *Bug 7 & steps:* Using a different icons for the same functionality in the settings and at the level start (**Figure 8**).

*Comment:* Bugs appeared during testing on all test devices.

#### 2.4.2 Testing of early-access game pillow bellow

This chapter presents an example of game testing of the early-access game carried out in 2021. The game is currently available on the Steam platform.

**Game:** Pillow Bellow.

**Developer & Publisher:** VOX Design.

**Version:** Early-Access.

#### **Playtesters:**

- The characteristics of playtesters are similar to the Bushfires game.

#### **Playtesting:**

- Positive remarks:
  - Very nice concept of the game with a unique world.
  - Refill life by collecting feathers.
  - I like that players can collect many different achievements in levels. For example, it is a lucky surprise to get the achievement of being lost in the woods.
  - I like it is possible to achieve also objects different from feathers, e.g., skins representing each game level.
  - It is exciting for a player that every level has a different theme, various traps, and enemies.
  - It is great that the way to the goal is not precisely defined. Mostly there are multiple paths, and to fulfill the level requirements, the player must explore all of them.



**Figure 8.**  
*Example of bushfires game – Different icons for the same functionality.*

- Using relaxation music which varies on all levels, there is little chance it will get on the player's nerves.
- Negative remarks:
  - The disadvantage is that there are no checkpoints in the levels, and it is always necessary to start over. It would be appropriate to add at least one checkpoint for each level.
  - The music at the Halloween level does not fit the theme. Something scary or indicating the Halloween theme would be more suitable.
- The third level represents the world of needles and threads. Sometimes the pillow gets stuck on a needle, and nothing happens. It would be good if the game reduced the player's health in such a case. If he does not jump off the needle at a defined time, he will die (like falling off the track).
- The third level represents the world of needles and threads. The player's pillow gets stuck in the object representing the threads, and it is challenging to move with it.
- Players cannot save collectibles (e.g., skins) and achievements in levels. Players cannot transfer skin (appearance) obtained in level 3 to other levels. It would be convenient to create a database of these objects and refer them to the main menu. The game would unlock the collectibles as the player found them in levels. The player would thus have an overview of his game activity. The player could "put on" the skin on his pillow whenever he wanted.
- In the Christmas special, the dialog with the villager switches by itself. I would allow the player to switch it so that he can read it. Not everyone reads quickly.

### **Usability Testing:**

- I like the game has a tutorial, and right at the beginning, it nicely explains what to expect in each level and how to control the pillow.
- Turning off the game using the X key in the level map simplifies the game control.
- On some levels, places in the dark or the shadows are too dark, and it is difficult to orientate there.
- If impaired mobility on a thread-type object happens on purpose, I'd reduce the step length or make the pillow take three steps to move (shear effect).
- The tutorial will start first instead of the main menu when restarting the game. When starting the game, the main menu should appear first, where there should be a link to the tutorial and the menu of levels.
- The menu under options is shown at the bottom that one can go back only with the right mouse button, but it also works with Backspace. However, it is not marked anywhere.

### Bugfixing (QA):

- *Bug 1 & steps:* Even though I turned off sounds and music, the sounds and music start at the beginning of each level. (Students attached video.)
- *Bug 2 & steps:* The counter of collected stars does not update in the Christmas special.
- *Bug 3 & steps:* In the Christmas special, I received a message about the successful completion of the level, although I did not reach the flag and did not even collect all the stars. The video shows that it was impossible to collect one of the stars. That is, I did not collect at least one star. (Students attached video.)
- *Bug 4 & steps:* In the level with the princess (level 5), my life got to zero after contact with the enemy, but even so, I was not sent back to the beginning and could continue the game. (Students attached video.)
- *Bug 5 & steps:* In the Halloween special, we need to collect 10 feathers, but the level counter shows we need to collect 15 (**Figure 9**).
- *Bug 6 & steps:* In the introductory tutorial and some level descriptions, there are typos.
- *Bug 7 & steps:* Moving objects sometimes disappear and appear in the levels. (Students attached video.)

#### 2.4.3 Testing of student multiplayer FPS game NSD

This chapter presents an example of game testing of a multiplayer FPS game carried out in 2022 (**Figure 10**). A group of students developed this game as part of a bachelor's project.

**Game:** NSD.

**Developer:** Students.

**Version:** Early-Access.

**Playtesters:**

- The characteristics of playtesters are similar to the Bushfires game.



**Figure 9.**  
*Example of pillow bellow game – Bugs.*



### Playtesting:

- Positive remarks:
  - Since this is a multiplayer game, chat and private messages are a plus.
  - Spawning immortality makes spawn camping impossible.
  - A different map for each match increases the dynamics of the gaming experience.
  - Considering this is an FPS game, it is good that the game has enough obstacles on the map.
- Negative remarks:
  - Lack of feedback in the menu (friend request and invite).
  - Spawning too close to other players occasionally.
  - The possibility of “jump shots”: A shot during a jump is accurate only if the player was standing before. So, it is possible to stand behind an obstacle, jump up, and shoot while the opponent is at a disadvantage.
  - After leaving the ladder, the player cannot shoot for a long time.
  - The shotgun is too weak (even in close combat).
- Other remarks:
  - The weapon movement based on the camera rotation looks interesting, but sometimes it is distracting.
  - One can use [Match] chat even if he is not currently in a match (and this chat corresponds to a previous match).



**Figure 10.**  
*Testing the multiplayer FPS game NSD in the classroom.*

- One cannot leave the ongoing match.
- The character model (third-person view) does not match what the player sees in the first-person view. For example, the player sees the character is wearing gloves while holding the weapon/knife, but there are no gloves on the model.

#### Usability testing:

- The possibility of setting the aiming sensitivity.
- Ability to jump onto the roof of the building using the surrounding obstacles.
- After respawning, the player must select a weapon again (the player is empty-handed). With frequent respawn, this is a user disadvantage.
- Jumping through the windows is possible but difficult.
- SUS questionnaire:
  - Questions focused on movement, shooting, weapon damage tuning, and the overall UI experience.
  - 41 users completed it.
  - The Likert scale was 1–5.
  - Overall evaluation was 76.16, which is positive (B).

#### Bugfixing (QA):

- *Bug 1 & steps:* Getting stuck in a wall by closing a door (another player must close it; a person cannot close it himself if he is inside it).
- *Bug 2 & steps:* The possibility to stand on top of another player if he is crouching; in other cases, one can pass through players.
- *Bug 3 & steps:* The ability to jump on a tree and stay on it (according to the developers, it should not have been possible).
- *Bug 4 & steps:* If the character is holding a knife, when one quickly clicks the mouse button, the animation of using the knife starts over again, but the attack does not finish.

### 3. Conclusion

In our department, we generally focus on education in software engineering. So, in the design and development of games, the implementation side, including the iterative process, is vital to us. We are looking for ways and activities to teach students to

transfer formalisms and critical aspects of Game Theory into practice. It is not only a transfer into source code but also a design or feedback.

This chapter presents our experience implementing the Design and Development of Computer Games and other similar courses. We emphasize several essential methods and aspects like iterative development, rapid (paper) prototyping, and game testing. In the chapter, we presented educational activities mainly focused on game testing: Iteration of a card game rules, design of a board game, design of a computer game with a paper prototype, and testing of early-access games. Activities are described in steps, including their motivation and benefits. They are supplemented with our commentary from the educational process point of view and students' practical outputs in both text and graphical form. The results include feedback in the testing forms: Focus Groups, Playtesting, Usability Testing, and Quality Assurance.

Critical thinking and the ability to transfer formalism into practice are crucial in the game industry and software engineering. Based on our experience, we can say that the students' outputs are mostly interestingly processed. They do not lack creativity or attention to detail, which we consider a positive result. The game creators processed early-access game testing. They are primarily experts working in the game industry who gave students feedback on how beneficial the testing was, being positive in most cases. Based on this, we perceive the results as contributinal.

We believe that our work will inspire lecturers from similar fields. Given that the results presented in this chapter are based on multiple years of experience with students, they can also be used as a basis for further studies and they will fulfill the title in terms of the way from theory to practice.

## **Acknowledgements**

This work has been supported by Slovak KEGA Agency under grant no. 002TUKE- 4/2021: "Implementation of Modern Methods and Education Forms in the Area of Cybersecurity towards Requirements of Labour Market" and under grant no. 048TUKE-4/2022: „Collaborative virtual reality technologies in the educational process“.


## **Author details**

Emília Pietriková and Branislav Sobota\*  
Technical University of Košice, Slovakia

\*Address all correspondence to: [branislav.sobota@tuke.sk](mailto:branislav.sobota@tuke.sk)

## **IntechOpen**

---

© 2022 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

## References

- [1] Schnell J. *The Art of Game Design: A Book of Lenses*. Third ed. NY USA: CRC Press; 2019
- [2] Fullerton T. *Game Design Workshop: A Playcentric Approach to Creating Innovative Games*. Fourth ed. NY, USA: CRC Press; 2018
- [3] McBain J. *Learning Game Design*. Kindle ed. Melbourne. Australia: ASIN; 2016
- [4] Choi JO et al. Playtesting with a Purpose. In: CHI PLAY '16: Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play. NY, USA: 2016. pp. 254-265
- [5] Lennart E et al. User experience (UX) research in games. In: CHI EA '19: Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems. NY, USA: 2019. pp. 1-4
- [6] Medlock MC. *The Rapid Iterative Test and Evaluation Method (RITE)*. First ed. San Francisco, USA: Oxford Press; 2018
- [7] Mirza-Babaei P et al. A postmortem on playtesting: Exploring the impact of playtesting on the critical reception of video games. In: CHI '20: Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems. NY, USA: 2020. pp. 1-12
- [8] Roohi S et al. Predicting game difficulty and engagement using AI players. In: Proceedings of the ACM on Human-Computer Interaction. NY, USA: 2021. pp. 1-17
- [9] Lewis J, Sauro J. *Ten Things to Know about the RITE Method*. Denver, USA: Measuring U; 2020
- [10] Adams E. *Fundamentals of Game Design*. Third ed. NY, USA: New Riders; 2013