DEVELOPING AN EDUCATIONAL BOARD GAME USING INFORMATION TECHNOLOGY

Jerzy Montusiewicz, Marek Miłosz, Jacek Kęsik, Marcin Barszcz, Krzysztof Dziedzic, Michail Tokovarov, Piotr Kopniak

Lublin University of Technology (POLAND)

Abstract

Gamification has become an established method of the educational process. Despite the great attraction of computer and video games, there is a continuous interest in games that require some active physical involvement.

The paper presents the use of various, mostly IT-related, technologies for the purpose of building a board game called "Architectural Jewels of Lublin". The game was created in connection with the 700th anniversary of Lublin's foundation. The aim of the game is to educate in the area of location and history of architectural monuments of the historic centre of Lublin. The essence of the game is to place on a map the three-dimensional models of the most important architectural monuments of Lublin. An essential part of the game is to provide the players with knowledge about the monuments (their history, architectural styles), and this is seen as best achieved by means of the challenges put before the participants by gamification. The devised board game is intended for either one or two players. Its technical structure is a complex system combining physical elements (board, 3D models of architectural monuments), a sensor network utilizing advanced localization algorithms, and an application with the software and sound system for the implementation of the competition.

The information technologies used in the preparation of the game deal with 2D modelling applied in developing board games, virtual 3D modelling of architectural monuments and physical replication of these models as 3D objects, construction of a sensory network using RFID technology with the Arduino software platform and the design and development of the software logic of the game on a palmtop. The paper addresses the problems arising during the implementation of the project in each of the identified areas.

Keywords: 3D modelling, 3D printing, RFID technology, education in cultural heritage, gamification.

1 INTRODUCTION

The occurrence of local historical anniversaries allows to use them for various promotional, business and educational purposes [1, 2]. The 700th anniversary of the granting of city rights to Lublin has become a pretext for the development of a board game that, despite the use of many computer technologies, takes place in real life [3]. The use of computer 3D modelling of historic architectural objects [4] for their 3D replication [5, 6] was combined with their automatic recognition when placed on the board. The whole was aggregated by a computer system that controls the game process. The main idea behind the whole project is the realisation of the concept of gamification by using modern IT technologies.

In article [7] there are many statistical summaries of various aspects of the use of gamification methods in teaching. It is clear from the content of the article that the methods of gamification are still being developed and still find new uses in the educational process. Work [8] enumerated four categories of gamified learning: game-based learning, simulation-based learning, using PBLs (points, badges, leaderboards) and gamification (use of rewards to motivate behaviour in an otherwise non-game context). Using the terms *gamification* and *game-based learning* in the paper is understood as referring to the use of games to achieve learning objectives. The authors of [9] state that certain trends are currently emerging in the use of certain configurations of game mechanics and rules of game design for education through gamification. However, there is no adequate technological support, which is one of the main obstacles to the use of the elements of gamification in education. In this article, we argue that a relatively moderate amount of work can be provided with adequate technological support to enable real-world gaming, using broadly based game learning for educational purposes. Selected sociological aspects of gamification are described in [10].

ISBN: 978-84-697-3777-4

The aim of this work is to present the computer technologies from the broadly understood IT field, as well as their interconnectedness, necessary for the preparation of a board game enabling the realisation of the game-based learning concept, which will be played in the real world.

2 THE IDEA OF THE BOARD GAME AND GAMEPLAY

"Architectural Jewels of Lublin" is a board game prepared in connection with the 700th anniversary of the granting of city rights to the city of Lublin. It is based on the competition between two participants, who in the two-stage gameplay endeavour to accumulate as many points as possible. The first step is always to place the selected historical object in the right spot of the board, the second – to answer the questions asked. The first question is answered by the person who correctly placed the object. The second question concerns both players and can be taken over by the competitor – the first person who presses this button obtains the right to respond. In the case of an incorrect placement, the second player takes over the initiative.

The board game is implemented using the following human-computer interface artefacts: Player, Board, Architectural objects, Answer button, Bank of objects and Places to insert objects.

The game is turn-based, i.e. the players perform similar actions in turn: they place architectural models taken from the object bank in appropriate locations of the model. In the case of an incorrect placement of an object, the system reads out its abbreviated historical description. This action aims at realising the educational purpose of the game. In order to increase the players' competition (gamification), an additional element was added to the game: a mini contest that tests messages about the object placed and the history of the city. The system asks a question and the players respond by using the Answer buttons and the response buttons on the player's panel. For correct answers, participants receive additional points. A wrong answer or no answer results in the loss of some of the points accumulated.

Each turn of the game is executed according to the following scenario:

1. Player with number n (initially n=1, then it passes through consecutive numbers) is asked to download one object and place it in the correct place on the board.

2a. The player has placed the object correctly – the object is removed from the object bank, the points are scored for the player, and the game is moved to step 3.

2b. The player has placed the object incorrectly – the historical information about the object is sounded and the player is asked to replace the object in the object bank. After its replacement, the game proceeds to step 1 with a change of player number.

3. Display (and reading) of questions about the object and expectation (countdown on the screen, max 30 seconds) by pressing the "Answer" button.

4. Registration of the player's answer to the question in point 3.

The final score of the game is the sum of the partial scores. The winner is determined according to the largest aggregate assessment. For identical values, a tie is announced. The table below shows the game evaluation elements.

Table 1. *Evaluation of the results of the activities of the players.*

No.	Item rated	Rating (points)
1	Correct object setting	2+n
2	Correct answer to question	2+n
3	No answer after pressing "I answer"	-2*i
4	Invalid answer	-2-n

Markings: n – number of incorrect settings of the object; i – number of missing/incorrect answers

The software provides an adaptive algorithm for evaluating the actions of game participants. This algorithm modifies the score from Table 1 on the basis of previous rounds of the same game, using weights assigned to individual historical results. These scales have the initial values of 1 for all evaluation elements. The weight of "Correct object placement" increases by 1 each time an object is placed incorrectly. For example, if you correctly place a re-selected object from the bank, you get 2+1=3 points. The "Correct answer to the question" score changes analogously. The score for not responding despite pressing the "Answer" button (item 3 of Table 1) is reduced by -2 * and for each i-th non-response despite pressing the "Answer" button, i.e. the first one costs the player -2, but the second already -4. This allows to penalise the participant for ill-considered use of the "Reply" button. The negative points for giving a wrong answer are -1, i.e. a repeated incorrect answer "costs" (-2)+(-1)==-3.

3 BOARD AND ARCHITECTURAL OBJECTS

3.1 Creating a board game

The game board is understood both as a carrier structure (providing gaming mobility) and an old town plan where 3D models of monuments will be placed. The accepted net size of the board of 1250×1420 mm includes the plan of the centre of Lublin, i.e. the historic old town extended by the surroundings. The frame support structure extends the overall size by 40 mm in each direction and is made from wooden boards reinforced with an additional truss also from boards that reinforces the whole and allows for fixing from the top a 6 mm thick plywood board. On this board, a map of the city is glued to the top, with RFID readers that verify the correctness of the object positions and all the other components that make up the game logic: Arduino boards, over 130m wiring, power strip and power supplies. The size of the board was chosen in such a way that the players could reach each place. The size of the printed 3D models of the monuments was large enough to be readily recognisable and easy to handle.

To make the graphics part of the game board, which will be printed on abrasion and scratch resistant material, a hybrid design method was used. It consists in using existing raster images and their vectorisation so that vector graphics objects are created to allow for any scaling. The method thus combines the advantages of high quality with the possibility of rapid modification and a relatively time-unconsuming workload. In addition, in the vectorisation process, it is possible to simplify the generated city plan to include only the streets and squares and the information needed to correctly identify the place of object insertion.

a) b)

Fig. 1. The game board: a) vectorisation on a raster background,
b) for printing after switching the background off.

To make the board the AutoCAD software was used, which allows to use the existing raster images and insert them into the program. Combining raster images with vector images makes it possible to place them as backgrounds in a drawing space. It should be added that bitmap files (in .jpg, .bmp and .tiff formats) are external objects (like links) and do not represent the actual part of the drawing, which only slightly increases its size. Preparing the old town game board consisted of the following activities:

- Personalisation of the created .dwg drawing file in AutoCAD by defining mm units for scaling inserted raster objects.
- Defining the layers, which served to aggregate objects of the same type and give them suitable properties and characteristics.

- Importing the map of Lublin in the .jpg format to the Map layer of the AutoCAD file and scaling the whole to the size of the board (1250x1420 mm).
- Performing the vectorisation process of Fig. 1a based on the imported background to the drawing file.
- Drawing the locations for placing the 3D models – true and false locations (blue wheels) and car parks (purple wheels).
- Transition from the working (drawing) area of the model to the printing (paper) area to define the printing device, the area and the print parameters.
- Printing the finished board with the layer visibility switched off and the plan background introduced in .jpg format, Fig. 1b.

3.2 Creation of 3D models of architectural objects

To model historic architectural objects for a three-dimensional board game, Autodesk's AutoCAD software was used. It allows to save a finished model to an .stl file, which is essential for the possibility of three-dimensional object replication. The basic research and application problem lies in the proper selection of methods and tools that will transform a three-dimensional, photorealistic model of an object perfectly displayed on a computer screen into a so-called "waterproof" model necessary for 3D printing. In the process of modelling and generation of the .stl format, the parametric model used in AutoCAD is replaced with a triangular representation of the object's surface geometry. The resulting shape is similar to that of the original model. Experiments have shown that obtaining the .stl format is best when all the elements of the object are added to a single body, and that requires prior conversion of surface objects into solid ones.

The second important aspect of the 3D modelling process geared towards 3D replication is the original size of the object (range of 20 to 40 m) and their rescaling to the size of their models after printing. Assuming that due to the size of the prepared board (1250x1420 mm) the size of the 3D copy of the monuments will be about 120 mm, their dimensions will be as many as 200 times smaller. Although 3D models are executed as vector objects and are fully scalable, such reduction will cause some architectural details to be very small, potentially below the precision of the 3D printer. Thus, familiarity with 3D printing technology and its parameters will allow for the proper layout of 3D modelling and the application of geometric simplifications of complex shapes and very small dimensions. This paper presents the main stages of 3D modelling of the Lublin castle courtyard Donjon.

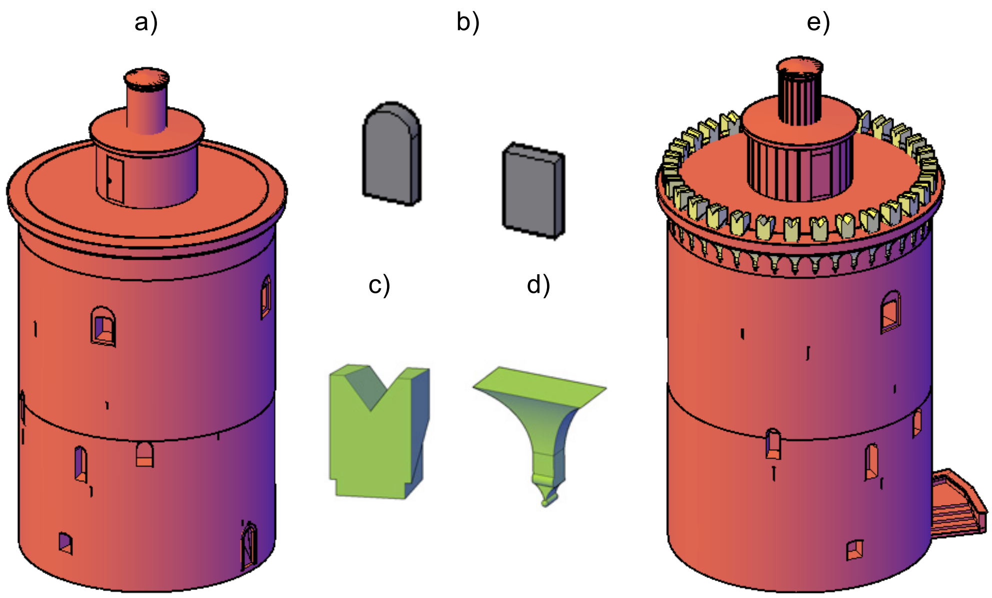

Fig. 2. Donjon 3D model: a) main body with windows and doors, b) window and door models, c) cradle model, d) cornice model, e) complete model.

3D modelling of the Donjon was made using the concept of creating a completely filled object (by means of 'primitives'). The Donjon is a defensive facility that was built at the end of the 13th century and decorated with Renaissance elements in the late 16th century. Its modelling required the following actions:

- Define the appropriate layers and give them their proper features.
- Apply the Cyllinder command to create the main body of the model, Fig. 2a. To design all the types of window and door openings the following commands were used: Polyline, Region, Extract (Fig. 2b) and Boolean operations – subtraction.
- Creation of bastions. A single element of the crenellation (Fig. 2c) was made by creating its outline (Polyline), converting it into a region type object and extracting it. Once placed on top of the tower, it duplicated by using Polar Array. The cornice (Fig. 2d) was modelled using the Cyllinder, Sphere, and Polar Array commands.
- Designing the stairs by using the instructions Cube and Plate.
- Adding all objects into one object – the Boolean operations command – addition, Fig. 2e.

3.3 Printing 3D models

For printing the architectural models used in the game the Fused Filament Fabrication (FFF) method was applied, due to the relatively low cost of both the printing device and the printing procedure itself. The FFF method is based on the formation of an object from melted plastic [6]. It involves placing layers of material (filament) heated to melting and extruded through a nozzle. The head, comprising of a filament moving motor, heating coil and a nozzle, moves in the XY plane when printing a single layer of the object. The software controls the direction of movement of the head and the extrusion speed of the filament.

Currently, the average thickness of the layer obtained by the available devices oscillates around 0.2 mm [11]. A disadvantage of this method of 3D printing is the necessity of placing a printed layer on the support. The dissolved filament must be supported before it hardens or it will deform under the influence of gravity. Therefore it is not possible to print the extended protrusions from the surface of an object without printing additional supports. The advantage of this method is the possibility of partly filling the interior of the printed item by means of the stiffening structure designs. This allows for producing light objects with high durability, which is desirable for objects used in the game. The process of manufacturing can be divided into several stages: the creation of the object model, selection of print parameters, conversion into slice model and creation of print execution file.

✦ *The creation of the object model*

As stated in 3.2, the object for print was prepared with the use of tools for computer-aided design (CAD). These tools offer the creation of a model in the form of a design composed of parametric objects [4], with a mathematical description of their particular shape. This solution, focused on the creation of a high-precision, mathematically described model, is not useful for FFF printing, where the print layers are created by analysing a triangle-based shell object model. It is therefore necessary to convert the parametric model into a triangle mesh, approximating its outer shell.

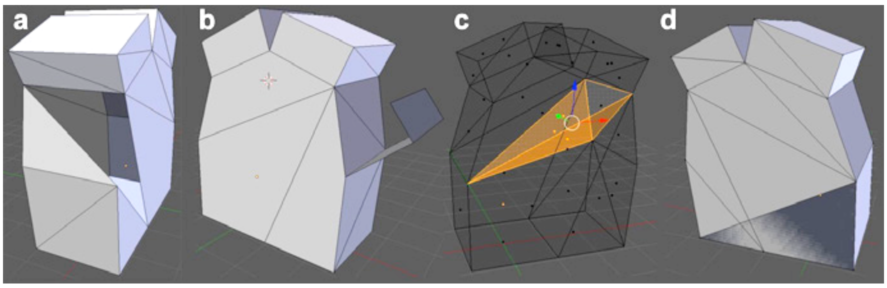

Fig. 3. Typical flaws of a 3D model, causing slicing problems: a) a gap in the object surface, b) a 2-dimensional element, c) inner geometry, d) overlapping triangles.

For the resulting triangle mesh model to be treated as a solid and be correctly converted to a layered model, a condition of manifold geometry must be met. A non-manifold geometry is an object having at least one of the following flaws: the surface of the shell object has a gap, the object is not “waterproof” (Fig. 3a), there exists an area of the object having a zero thickness (Fig. 3b), there are elements of the shell description placed inside the object (Fig. 3c), or there are overlapping areas of boundary representation description (Fig. 3d).

Although the tools for preparing a mesh model for printing have mechanisms for at least partial compensation of these defects, each of them may cause incorrect conversion to the slice model. Likewise, incorrectly defined sides for surfaces forming an object may be problematic. The external side of a face is indicated by a normal vector. When some of these vectors are directed towards the inner side of the object, it is difficult for the slicer to determine the correct volume of the body.

✦ *Selection of print parameters*

According to the principle of FFF printing, the object will be created with layers containing the model walls and the filling according to a fixed pattern. Walls set horizontally are treated as a floor or ceiling for which one can determine the thickness as a number of layers. The print parameters can be divided into two groups: the printout parameters and the auxiliary parameters.

The first group contains parameters that determine the shape of the end object. The overall height of the layer determines the global precision of the print, the number of layers per millimeter of print. In addition, the boundary size of the object features is determined. Fragments smaller than the limit value will be omitted when creating a contour model. Other parameters are: the thickness of the floor and the ceiling in layers, the thickness of the side walls in the form of the number of layers (shells), the way of filling the interior of the pattern (e.g. diamond, honeycomb, etc.) and its percentage filling.

The second group are parameters that define how printouts of a prospective support should be made to correctly print fragments not supported by the model itself. In general, each piece projecting at an angle of less than 45º relative to print base requires support. The exception are so-called bridges, where the fragment is anchored in the model on both sides. The parameters of the supports include, but are not limited to, the possibility of turning them on/off, the density of generating supports, the distance of the support from the model side surface and from the supported layer, the angle limit from which the support should be generated, and the maximal bridge length without supporting.

✦ *Conversion to contour model and printing*

The conversion follows the print parameters specified. The conversion algorithm requires a valid, error-free model. Otherwise, the generated contour model may contain undesired shortcomings. The last step is to save the contour model in the format required by the printing device and start the printing process. The printed model requires manual postprocessing in order to remove the supports and smooth out undesired rough fragments.

4 GAME ARCHITECTURE

The proposed solution of the game from the point of view of its electronic components and the control of the gameplay consists of three basic elements (Fig. 4).

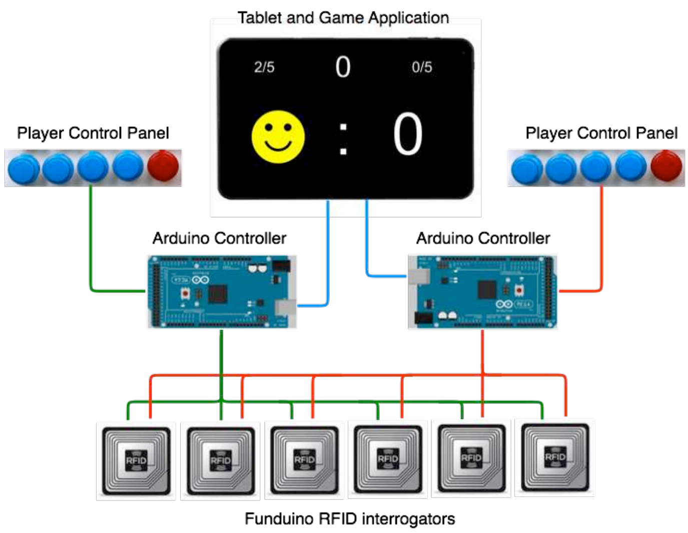

Fig. 4. Technical infrastructure of the game.

The idea of recognizing the models is based on the RFID technology. RFID stands for radio frequency identification, i.e. it is the technology allowing to recognize objects by means of radio waves [12]. While other radio wave based distance measurement techniques were considered, like WiFi RSSI [13] or Bluetooth [14], the exceptionally close distance required between receiving spots disqualified these.

An RFID communication pair consists of a transmitter/receiver, also called an interrogator, and an RFID tag. A tag contains a unique ID number, which ensures unambiguous identification of the tag by the interrogator. Depending on the tag operation principle, an RFID system may be active or passive. In an active RFID system an interrogator only receives radio waves emitted by a tag, the tag produces radio waves independently by constantly sending its ID number. This approach allows to achieve a higher transmission range while having a significant disadvantage: the tags, due to their independent operation, should be powered independently as well [15]. The second type of RFID system is called passive. On the one hand this type is more complex: the interrogator works in a two-stage cycle, not only receiving radio waves, but also emitting them. On the other hand it is simpler there being no need to power the tags independently as they use the power of the waves emitted by the interrogator to generate the response [15].

The project did not demand a high active range of the interrogators as recognition should be performed within a specific limited range (6-10 cm). Along with that, compact tags not demanding independent powering would be essential as the models should be light and wireless. These conditions determined the second type of RFID system to be selected for use in the project described in the present paper.

The electronic part of the board game consists of two identical sets. One set is used for recognising the models placed on the game board, while the other is responsible for recognising the models placed in the object bank. The elements of a set are presented in Table 2. The resistors mentioned in Table 2 are used in order to lower the voltage for the RGB LEDs, as the output voltage of Arduino Mega digital pins is 5 V, whereas the operation voltage of RGB LEDs is ~3 V.

Table 2. *Elements of an electronic set.*

No.	Title	Number
1	Arduino mega 2560 development board	1
2	Funduino RC522 RFID interrogator	12
3	RFID tag card	12
4	RGB LED	12
5	100 Ω resistor	24

The development board communicates with the interrogators by the means of an SPI (serial peripheral protocol). This protocol uses three common buses: a system clock clocking the communication system, MOSI (Master Output Slave Input), MISO (Master Input Slave Output) and SDA (slave select). In order to choose the slave for communication, the digital output pin, which the slave is connected to, should be set to LOW, and the rest of the SDA pins should be set to HIGH [16]. There is also an RST (reset) pin, it is used in order to reset the interrogator after it detects a card. Figure x.1 presents the schematics of connecting the interrogators to the development board. The logical scheme is presented in Fig. 5.

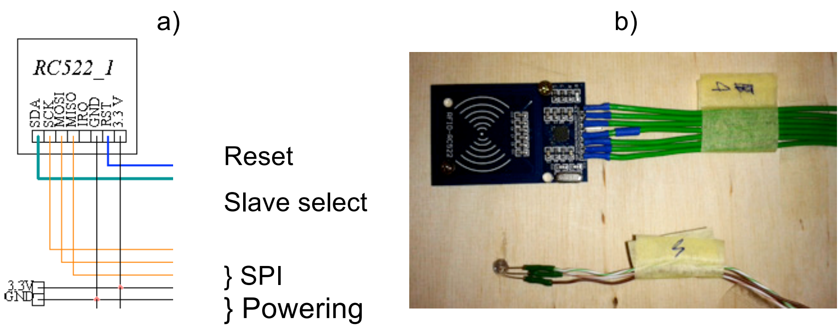

Fig. 5. RFID reader: a) scheme; b) actual image with RGB LED.

The program for controlling the electronic part was written in the C++ programming language, allowing to make it both fast and easily understandable for a programmer.

Game application is the main element of game supervision. It contains the implementation of the gameplay algorithm and the handling of a player's movements. The application installed on a Windows tablet communicates with the Arduino controllers via a USB connection. The created application provides:

(i) communication with Arduino controllers, gathering information about player movements and answers to questions, and controlling other aspects of the game,

(ii) proper operation of Arduino controllers recording events related to player movements and pressing buttons while selecting answers to questions and taking over the initiative in the game,

(iii) RFID-enabled tokens and sensors for proximity detection of actions of lifting from the base and placing target buildings on the game board.

The main tasks of the application are:

- Sending messages to Arduino devices to start and stop the game.
- Receiving and interpreting the messages sent by the Arduino controllers regarding player-related activities during the game, such as changing the figures on the board or pressing buttons on the control panels.
- Counting moves and points earned by players.
- Randomly displaying questions for players who have scored and interpreted answers to questions.
- Countdown time for traffic.
- Information on the completion and outcome of the game.
- Additional gamification.

To build the application, the Java programming language was used. This can be run under any operating system that has a dedicated Java virtual machine implementation. In this case, Windows was used because one of the requirements for non-functional games was to work on a tablet with a system that has a USB communication port. USB provides fast data transfer between the controllers and the tablet, and some uninterrupted communication, which could be a problem with wireless transmissions, for example via Bluetooth.

Java has its own Java Communications API [17] for communication through the computer's I/O ports. This library allows applications to access RS-232 serial ports and limited access to IEEE-1284 parallel ports, but due to the fact that it is limited to only three system platforms: Solaris SPARC, Solaris x86 and Linux x86, it could not be used. In the end, the jSerialComm [18] universal external jSerialComm library, which does not use native code, runs independently of the operating system selected. The standard Java Media Framework [19] is used for audio playback.

Messages sent between devices through the serial port are in the form of three numeric values separated by a comma and a hash (#). The first number is the player number, the second is the number of the function that should be called after the message is delivered, and the third is the value of the argument for that function. The set of functions and their meaning are shown in Table 3.

Table 3. *Functions and function arguments used in the communication protocol.*

No. of function	Values of arguments	Description
1	0 or 1	Start of the game – 1, stop – 0, e.g. message: 1,1,1#
2-9	NN	Reserved for future functions
10	0 or 1	Pressed main player button – 1, released – 0
11-14	0 or 1	Player button (one of answer buttons: 1, 2, 3 or 4), pressed – 1, released – 0
21-33	-1 or 0 or field number at the board	Information of historic building move. Building in the base – 0, elevated – (-1), 1-n – number of place where the moved historic building model was putted down

The game has a simple, clean user interface that does not distract the player from the game board with objects and provides good visibility from a distance. Proper movement of the player on the board, or the correct answer to the question is rewarded with positive points, emoticon smiley face and the sound of applause. Negative points give rise to a sad face, the sound of disappointment and negative points. Examples of the game screen shots are shown in Fig. 6 and the verification lighting of objects on the board (right place – green light and wrong place – red light) in Fig. 7.

a) b)

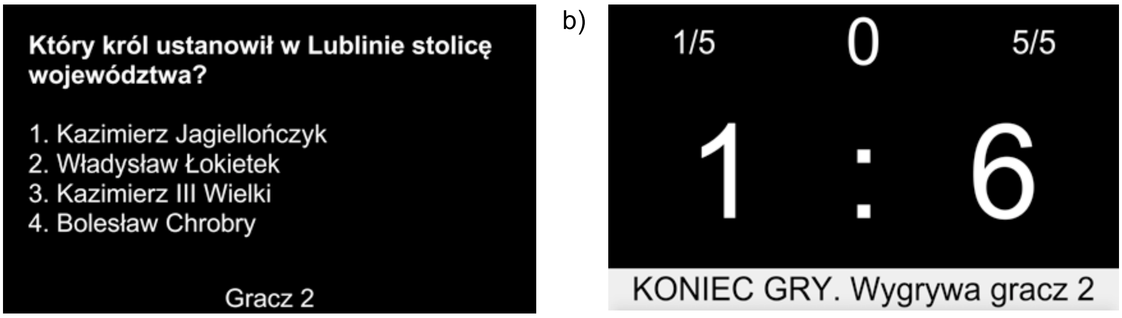

Fig. 6.Two screens of the game: a) question screen, b) result screen with information about winner of the game.

a) b)

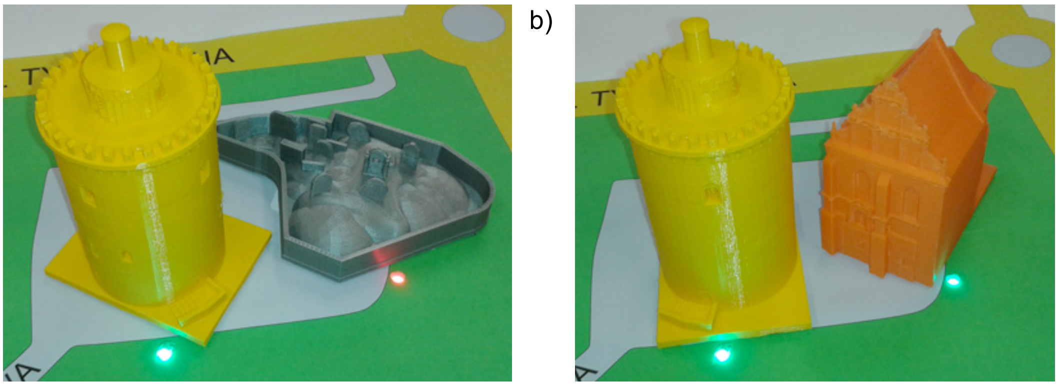

Fig. 7. Objects placed on the board: a) incorrectly, b) correctly.

5 CONCLUSIONS

The "Architectural Jewels of Lublin" board game is a multidisciplinary project, despite the fact that all the applied technologies belong to the broadly understood IT area. Thus it was necessary for its realisation to build a team of specialists, representing computer graphics, programmers, gamers, electronics and 3D printing technicians.

First of all, the attractiveness of the game was determined by the visual aspects, namely the accuracy of the prepared 3D models of historic architectural objects and the quality of their 3D replication. At the same time, it was necessary to adjust the accuracy of the detail projection to the role that the prints would have to perform: repeatedly and manually displaced verticals. It seems that the reality effect of the 3D models would be even better if 3D printing was used with multi-colour printing technology.

The preparation of a board game in which the game process is controlled by interconnected computer systems requires precise prediction of all possible interactions between the players and the computer and proper communication between the systems. Only on the basis of this knowledge is it possible to build an attractive scenario that encourages mutual competition between players.

REFERENCES

[1] Milosz M., Milosz E. (2012) Business simulation games during the summer schools – the case study Proceedings of 6th International Conference of Technology, Education and Development (INTED), Valencia, SPAIN, pp. 1577-1583.

[2] Plechawska-Wojcik M., Schmidtke M., Skublewska-Paszkowska M. (2016) Applying of gamification in rising social awareness of urban residents ICERI2016 Proceedings, pp. 8390-8396.

[3] Montusiewicz J., Barszcz M., Dziedzic K., Kęsik J., Milosz M., Tokovarov M. (2017) The concept of a 3d game board to recognise architectural monuments Proceedings of 11th International Conference of Technology, Education and Development (INTED), Valencia, SPAIN, pp. 8665-8674.

[4] Autodesk (2015) AutoCAD 2015: Complete Guide to What's New. www.cadmasters.com/pdf/AutoCAD%202015%20What%27s%20New%20Guide%20.pdf

[5] Montusiewicz J., Czyż Z., Kęsik J. (2015) Using 3d replication technology in preparing didactic aid sets in the area of cultural heritage Proceedings of 7th International Conference on Education and New Learning Technologies (EDULEARN), Barcelona SPAIN, pp. 1861-1871.

[6] Dudek P. (2013) FDM 3D Printing Technology in Manufacturing Composite Elements, Archives of Metallurgy and Materials, Vol. 58, Is. 4, pp. 1415–1418, DOI: https://doi.org/10.2478/amm-2013-0186.

[7] Caponetto I., Earp J., Ott M. (2014) Gamification and Education: A Literature Review ITD- CNR, Genova, Italy, *http://www.itd.cnr.it/download/gamificationECGBL2014.pdf*

[8] Emerging Strategy (2015) Current and Future Prospects for Gamification in the Education Sector *http://www.emerging-strategy.com/article/*current-

[9] Dicheva D., Dichev C., Agre G., Angelova G. (2015) Gamification in Education: A Systematic Mapping Study Educational Technology & Society, 18 (3), pp. 75–88.

[10] Simões J., Redondo R. (2013) Vilas A. A social gamification framework for a k-6 learning platform Computers in Human Behavior, vol. 2, no. 29, pp. 345-353.

[11] Sculpteo (2017) Layer thickness in 3D printing: an additive manufacturing basic, https://www.sculpteo.com/en/glossary/layer-thickness-definition/

[12] Szymczyk T., Montusiewicz J., Gutek D. (2016) Navigation in large-format buildings based on RFID sensors and QR and AR markers Advances in Science and Technology Research Journal, vol. 10, no. 31, pp. 263-273.

[13] Czerwiński D., Przyłucki S., Mukharsky D. (2016) RSSI-Based localisation of the radio waves source by mobile agents Communications in Computer and Information Science; No. 608.

[14] Sierszeń A., Przyłucki S., Wojciechowski R., Sturgulewski Ł. (2016) System of distance measurement and localization of wireless devices communicating based on bluetooth technologies Advances in Intelligent Systems and Computing; No. 389.

[15] Lei Y., Jiannong C., Weiping Z., Shaojie T. (2015) Accurate and effective object tracking based on passive RFID IEEE Transactions on Mobile Computing, vol. 14, no. 11, pp. 2188-2200.

[16] http://www.byteparadigm.com/applications/introduction-to-i2c-and-spi-protocols/ – last access 28-03-2017.

[17] Java Communication API http://www.oracle.com/technetwork/java/index-jsp-141752.html.

[18] jSerialComm http://fazecast.github.io/jSerialComm/ – last access 28-03-2017.

[19] Java Media Framework *http://www.oracle.com/technetwork/articles/javase/index-jsp-140239.html* – last access 28-03-2017.