Multimodal Technologies and Interaction

MDPI

Article

A Comparative Analysis of Low or No-Code Authoring Tools for Location-Based Games

Christos Batsaras and Stelios Xinogalos *

Department of Applied Informatics, University of Macedonia, GR-54636 Thessaloniki, Greece; mai20042@uom.edu.gr
* Correspondence: stelios@uom.edu.gr

Abstract: This article presents a comparative analysis of four low or no-code location-based game (LBG) authoring tools, namely Taleblazer, Aris, Actionbound, and Locatify. Each tool is examined in detail, with an emphasis on the functions and capabilities it provides for the development of LBGs. The article builds on the history and purpose of LBGs, their characteristics, as well as basic concepts and previous applications, placing emphasis both on the technological and pedagogical dimensions of these games. The evaluation of the tools is based on certain criteria, or metrics, recorded in the literature and empirical data collected through the development of prototype games for each tool. The tools are comparatively analyzed in terms of the LBG's constituent features they incorporate, the fundamental and additional functionality provided to the developer, as well as the existence or absence of features that captivate players in the game experience. Moreover, feedback is provided based on the practical use of the platforms for developing LBGs in order to support prospective developers in making an informed choice of an LBG platform for implementing a specific game. The games were created by taking advantage of as many features of the tools as possible in order to have a more fair and complete evaluation. This study aims to highlight the affordances and limitations of the investigated low or no-code LBG authoring tools, enabling anyone interested in developing an LBG to choose the most appropriate tool taking into account their needs and technological background or designing their own LBG authoring tools.

Keywords: location-based games (LBGs); low or no-code authoring tools; comparative analysis

Citation: Batsaras, C.; Xinogalos, S. A Comparative Analysis of Low or No-Code Authoring Tools for Location-Based Games. *Multimodal Technol. Interact.* **2023**, *7*, 86. https://doi.org/10.3390/mti7090086

Academic Editor: Anton Nijholt

Received: 2 August 2023
Revised: 20 August 2023
Accepted: 25 August 2023
Published: 1 September 2023

1. Introduction

Location-based games (LBGs) are a form of experimentation with locative media involving location-aware mobile technologies and the disclosure of user locations. By integrating digital content with physical space through GPS technology, a hybrid space is created, offering a multi-dimensional experience [1,2]. Locative games exemplify the ludic trend in contemporary culture. The use of locative media further enriches urban spaces by providing location-based information and hidden narratives. These developments originated from artistic experimentation with GPS technology, since artists explored hybrid spaces by integrating data packets or metadata into physical environments.

The rapid technological advancements in communication media, such as GPS technology, augmented reality (AR), ubiquitous computing, and smart devices, have brought significant changes to how we perceive both physical and virtual spaces, as well as the concept of gaming. The emergence of the hybrid space and AR has laid new foundations for game design, and the process of designing LBGs has become participatory through open-access tools and platforms, allowing users and players to create their own games.

While attempts have been made to analyze and compare the development tools of location-aware games for their effectiveness and suitability as entertainment and educational tools [3–5], the work presented in this article aims to update the list of previously investigated tools and games, some of which have not been adequately analyzed in previous studies or have not been analyzed at all due to their recent creation. This update

is essential given the ephemeral nature of these digital tools, as many are frequently replaced by new ones that offer different capabilities. To address these research needs, LBGs have been developed to showcase the potential of each platform and highlight both their entertainment and educational aspects.

More specifically, the purpose of this study is to evaluate and comparatively analyze the most recent and popular low or no-code development tools for LBGs. The study applies a theoretical approach for exploring the features, technologies, pedagogical dimensions, and design of these tools based on metrics defined in existing literature [4–11], and an empirical approach that involves developing prototype games using each of the examined tools. The ultimate goal of the study is to compare and contrast the investigated tools to draw conclusions about their effectiveness and explore their positive and negative attributes. More specifically, this comparative analysis aims to investigate the following research questions:

RQ1. How do LBG constituent features map to the features of the investigated tools?
RQ2. What LBG development functionality (basic and additional) is provided by each tool?
RQ3. What features for captivating players' interest are provided by each tool?
RQ4. What is the overall support provided by each tool to inexperienced users when authoring LBGs?

The rest of the article is structured as follows. In Section 2, related work is presented, including a background on the evolution of LBGs and their characteristics, LBG authoring tools and the necessary developer functionality that must be offered by them, and previous work on analyzing such tools. In Section 3, the methodology of the study is analyzed. Next, the main features of the four LBG authoring tools are presented, along with the prototype games that were developed with them (Section 4). In Section 5, the results of the comparative analysis are presented, and finally, in Section 6, conclusions are drawn.

2. Related Work

2.1. Background on LBGs

In the world of gaming, one of the most intriguing and innovative genres to emerge in recent years is that of location-based games (LBGs) or location-based augmented reality games (LBARGs). These games utilize the player's physical location and real-world surroundings as integral elements of the gameplay, blurring the boundaries between the virtual and the physical. This interactive gaming experience has captured the imagination of players and developers alike, fostering a new era of immersive entertainment.

The first LBG that was created was Geocaching (2000) [12] in the USA. Geocaching is a treasure hunt game that uses hidden containers, known as "geocaches". The players are "*provided with the GPS coordinates of a hidden container (a 'cache')* via *a website*" and they have to find the caches through a "*GPS-enabled mobile device*" [13] (p. 162). In fact, the players have to "*prove their presence at the cache*" by signing a logbook at its location, while also signing in at the Geocaching.com website [14] (p. 84). Geocaching is a mixture of technology, exploration, and community engagement, with players sharing their experiences with the community as well. Significantly, this game paved the path for the production of similar scavenger hunt games such as Munzee (2011) [15] that "*utilizes the correlation between the material landscape and digital space*", but also take advantage of the capabilities of smartphone technology, such as built-in location awareness, as will also be shown further down.

However, back in 2001, when GPS was made available to the public, two more LBGs were created on the other side of the Atlantic. Those were "Botfighters" in Sweden by the "It's Alive" team and "Can You See Me Now?" in the UK by Blast Theory. These games were played on portable devices, such as personal digital assistants (PDAs) and laptops [2].

During this period, another significant milestone was reached with the creation of the first commercial LBG, "Mogi", which operated using GPS technology [16]. These early games acted as precursors to modern LBGs and introduced novel features that would shape the genre's evolution in the years to come.

As technology rapidly advanced, particularly with the advent of "smart" mobile phones in 2008, equipped with advanced features and operating systems such as Android and iOS, a surge in mobile applications (mobile apps) followed. This technological shift marked the transition of LBGs from experimental artistic endeavors to commercially produced software for mass consumption [17].

In the early 2010s, a new generation of LBGs emerged, distinguished by their availability on mobile phones [18]. Prominent among these games were "Parallel Kingdom", introduced in 2008, and "Shadow Cities", released in 2010 [19]. However, the definitive moment for the genre came when Niantic, Inc., a subsidiary of Google, developed "Ingress" [20] in 2012. This game, followed by "Pokémon GO" [21] in 2016 and "Harry Potter: Wizards Unite" [22] in 2019, marked the mass commercialization of location-based gaming [17]. These games took advantage of multiple devices and platforms that players could use to immerse themselves in location-based gameplay [23].

In the quest to precisely define this innovative gaming genre, researchers and enthusiasts have employed various terms to describe LBGs based on the specific features and characteristics each game exhibits. Some of the terms include pervasive games, urban games, location-based games, augmented reality games, and alternate reality games [24]. Due to the diversity of gameplay elements and the technology employed, LBGs often exhibit hybrid characteristics, combining augmented reality with GPS-based mechanics or even integrating elements from massively multi-player online role-playing games (MMORPGs).

In conclusion, LBGs have come a long way since the introduction of "Geocaching" in the early 2000s. From experimental artistic projects played on portable devices to the mass-produced mobile apps and highly immersive experiences we see today [18], location-based gaming has transformed the gaming landscape. With the continuous advancement of technology, we can only anticipate further evolution and innovation in this captivating genre of gaming.

2.2. LBGs Characteristics

Spatial, temporal, and social dimensions are expanded through LBGs [24], which are characterized by several key elements:

- *Spatiality*: Players experience interactions with physical space, locations, and objects. The "magic circle" of gameplay extends beyond the virtual space, incorporating the urban environment [25].
- *Temporality*: Players can engage in the game at any time as it lacks a limited duration.
- *Mental dimension*: The game presents cognitive challenges, requiring problem-solving and strategic thinking during gameplay.
- *Sociality*: Players engage in socialization, forming teams and collaborating to overcome challenges in the game.
- *Immersion*: Players are deeply engaged with both the virtual environment on mobile screens and their physical surroundings, creating mixed reality experiences [24,26].

LBGs utilize technologies such as GPS, Bluetooth beacons, and AR [24], and exemplify ubiquitous computing, where devices are interconnected in a network. They also have significant educational potential, offering interactive learning experiences and incorporating knowledge about the physical environment [27]. Practical applications include treasure hunts and educational games in cultural spaces and museums. In summary, LBGs integrate technology and education to create interactive experiences, allowing players to explore the physical world while engaging with digital content and each other.

2.3. LBG Authoring Tools

The special features of LBGs make their development a rather challenging process. This resulted in the development of LBG authoring tools to support users in LBG design and development. The history of LBG authoring tools can be traced around the same period that the first location-aware games came out, that is, the early 2000s, when selective availability of the GPS was turned off and GPS technology was transferred "*from the military*

to the public" [18] (p. 34). In 2002, the platform Mscape (2002–2010) [28] was initiated by Hewlett-Packard, offering its users the opportunity to attach multimedia content to locations in order to create their own stories and games, which could be experienced via a PDA and later on a GPS-enabled mobile phone. It was not until later, though, that such platforms went mainstream with the release of the Wherigo platform in 2007 by Groundspeak, Inc. [29], with which users could construct stories, games, and guided tours. A year later, with the advent of smartphone technology, the Far-Play (2008) platform was designed by the fAR-Play Team, which facilitated the creation of smartphone-based treasure hunts and games that involved the use of augmented reality and QR codes [30]. Similar platforms were designed in the next decade, such as FreshAir (2013–2019), which could also be used in education.

As a matter of fact, numerous LBG authoring tools were developed to fulfill different needs. Fidas et al. [31] in their survey analyze five authoring tools that can be used for implementing cultural heritage experiences: Hoppala, ARIS, TaggingCreaditor, LoCloud, and CHEF. In [5], the following tools developed in the context of research projects are briefly analyzed: COLLACE and Games Atelier, which were targeted at secondary education students; TOTEM, which supports AR and mobile mixed reality games; and MAGGELAN, which supports the creation, creativity, sharing, and execution of LBGs. The aforementioned tools, developed in the context of research projects, are no longer available but contributed to the evolution of LBG authoring tools.

Contemporary LBG authoring tools include both free and commercial tools, as well as commercial tools with free or free-trial versions: ARIS [32], Taleblazer [33], Action-bound [34], Locatify [35], Motive [36], STQRY (previously known as MY TOURS and initially created as 7Scenes) [37], ARLearn [38], CREANDO [39], LAGARTO [40], and TraceReaders [41]. Some of these tools have been evaluated with positive results, while others have been used for creating commercial location-based apps and games. Moreover, several of the aforementioned tools focus on education, such as CREANDO, LAGARTO, and TraceReaders. CREANDO aims at creating pervasive games in closed spaces with the ultimate goal of increasing motivation for learning in higher education students, a goal that was confirmed in a case study with 20 students [39]. LAGARTO [40] is an LBG authoring tool enhanced with AR features that supports the development of multiplayer games even by users without programming experience, as showcased in a study with 14 software developers and 14 participants without any programming experience. TraceReaders [41] is a tool that aims to support the development of apps for location-based AR inquiry learning and showed promising results in two case studies with primary and high school students. From the aforementioned tools, ARIS and Taleblazer seem to have attracted great attention, probably due to the support of these tools by the Wisconsin Center for Education Research at the University of Wisconsin—Madison and the MIT Scheller Teacher Education Program (STEP) lab, respectively.

Finally, we should note that besides the tools that are dedicated to authoring LBGs, widely known game engines provide support for the development of LBGs. A typical example is the Unity game engine with the support of the Mapbox SDK, analyzed in [5].

2.4. LBG Authoring Tools Functionality

To simplify game development and achieve dynamic gameplay experiences, LBG authoring tools should provide the following functionalities, which are presented below as described in [7,8] and reviewed in [4]:

1. *Non-linear authoring*: Each playthrough of the game should offer a different experience, depending on the player's role, given scenario, or past choices. This means that the authoring tool offers opportunities for event handling and corresponding triggers that result in different paths or experiences each time the game is played.
2. *Visual authoring*: Designers should be able to construct applications without extensive programming knowledge, modifying character properties and placing objects on the map easily.

3. *In situ authoring*: Designers can be physically present in the game's location via a mobile device, enabling them to create and modify the game for a better understanding of player experiences.
4. *Re-use and re-editing*: Novice creators should have access to a game framework, making it simpler to reuse and modify existing games.
5. *Content adding and management*: Integrated mechanisms for adding, managing, and storing content within the application should reduce the complexity of game creation.
6. *GUI customization*: Users of different ages and technical knowledge should be able to customize the program's interface according to their preferences.
7. *Simulation mode*: Game creators can perform functionality checks remotely without needing to publish the game.
8. *Game analytics*: Access to statistics on game performance and player behavior allows designers to adjust the game, scenarios, educational content, and difficulty levels accordingly.
9. *Map authoring*: Simplified representation of in-game events directly on the map aids in the development of LBGs.
10. *Visual programming or programming interface*: Offering possibilities for extending and enhancing game elements in an easily understandable way, thereby optimizing player interaction.

By incorporating these functionalities, game development becomes more accessible, resulting in engaging and diverse gameplay experiences for players. To enhance player familiarity and immersion in games, the following features and tools should be available to players [4–7]:

1. *Multiple media formats*: Games should utilize text, audio, visuals, and 3D graphics to entertain players.
2. *Multiple sensing technologies*: Games can adjust based on players' physical location, using technologies such as GPS, QR codes, Bluetooth beacons, augmented reality (AR) markers, and RFID.
3. *Reflection*: Players' actions and creations within the game should be stored and organized for later group discussions, feedback, critical thinking, and conclusions.
4. *Communication*: Games should enable communication between players, facilitating goal achievement and information exchange through messaging or voice transmission.
5. *Collaboration*: Tools that provide information, location data, and multimedia exchange support players in working together to progress smoothly using their unique skills and knowledge.
6. *Multiple players*: Continuous internet connection enables multiplayer interactions and unified game states for a diverse player base.
7. *Multiple operating systems*: Supporting various systems, such as Android and iOS, broadens the player's audience.
8. *Connectionless gameplay*: Allowing offline play accommodates player preferences and situations where internet access is limited.
9. *Outdoor–indoor gameplay*: Games should function in both indoor and outdoor environments, utilizing Bluetooth, QR codes, and GPS.

By providing these features to players, game developers can enrich the gaming experience and encourage meaningful interactions and engagement.

2.5. Previous Studies

One of the first and fundamental studies exploring the playful side of communication media through locative gaming is the 2009 volume edited by Adriana de Souza e Silva and Daniel M. Sutko entitled "Digital Cityscapes: Merging Digital and Urban Playspaces" [42]. The authors provide an overview of the categories and characteristics of LBGs, considering their pedagogical aspects as well. They emphasize some fundamental principles related to the design of these games.

Postgraduate and doctoral dissertations, as well as articles, have also been conducted to comparatively analyze the tools for creating LBGs. Xanthopoulos and Xinogalos [7] comparatively analyze ARIS and Taleblazer in terms of their authoring functionality, admin tools, and client-app features. Metikaridis [4] in his master's thesis comparatively analyzes Taleblazer, ARIS, and Unity with Mapbox. The comparative analysis is based on metrics recorded in the literature and the practical experience gained through the development of prototype games. This work was refined by utilizing an extended list of metrics for the comparative analysis of the tools [5], which was adopted for the comparative analysis presented in this article as well. Komninos Papaevangelou [3] analyzes ActionBound, PlayVisit, TaleBlazer, and Mapbox through the development of prototype games and the metrics defined in [7].

The aforementioned studies serve as a starting point for the present work, and its notable difference lies in the selection of analyzed tools, which have not all been the main focus of the previous works and have not been extensively studied in the existing literature.

3. Materials and Methods

The study presented in this article aimed at comparatively analyzing low or no-code LBG authoring tools that are widely known and used for the development of LBGs. The tools comparatively analyzed are TaleBlazer, ARIS, Actionbound, and Locatify and were selected based on the following criteria: Being low or no-code tools; being freely available, or at least offering a functional free version (and not a limited-time trial version); being exclusively used for the creation of location-based games/apps and not video games in general. This means that widely known game engines, such as Unity (with Mapbox), Unreal Engine, GDevelop, and Construct, were not legible for our study. In contrast with previous works in the field [3–5] we decided to focus exclusively on widely known low or no-code tools and comparatively analyze them with an extended list of metrics proposed in [5] that was based on reviewing relevant literature. Although ARIS and Taleblazer have been analyzed in previous work [5,7], we considered it important to include them in our analysis not only because they fulfill all the selection criteria and are two of the best-known tools, but also because the landscape of LBG authoring tools is constantly changing. Actionbound has also been analyzed before [3], but utilizing a subset of the metrics used in our study, while Locatify was not analyzed in the reviewed studies.

The study aimed to investigate the following research questions:

RQ1. How do LBG constituent features map to the features of the investigated tools?
RQ2. What LBG development functionality (basic and additional) is provided by each tool?
RQ3. What features for captivating players' interest are provided by each tool?
RQ4. What is the overall support provided by each tool to inexperienced users on authoring LBGs?

The methodology proposed by Metikaridis [4] and Metikaridis and Xinogalos [5] was adopted in our study. The methodology proposed consists of three steps: (1) Defining the metrics that will be used in the comparative analysis of the tools based on relevant literature; (2) designing, implementing, and testing LBG prototypes using the tools; and (3) using the predefined metrics for comparatively analyzing the tools based on data from the literature, tool manuals, and documentation, as well as empirical data. In our study, the first step was skipped, and the metrics already proposed were utilized.

The prototype games were carefully designed in order to utilize as many features incorporated in the tools as possible. The prototype games were developed by the first author of this paper exclusively for the needs of the study presented in it. The developer holds a BSc in "Geoinformatics and Surveying" and an MSc in "Applied Informatics". In the context of the MSc, he had attended a course on "Serious Games Programming", but he had no experience neither in implementing LBGs nor in the LBG authoring tools investigated in the study. The prototype games are available through the platforms: in the ARIS app, you search for the game with the title "Time Keepers"; in the Taleblazer app, you

search for the game code "gjedgrr"; in Locatify (Turfhunt app [43]), you enter the invitation code "CZKAWJ"; and in Actionbound, you search for the game "Curiosity Bound".

The platforms were comparatively analyzed on the following axes:

- *LBG constituent features*: Mapping of LBG constituent features as encountered in the literature with the features incorporated in the tools (Section 5.1).
- *Development functionality*: It is checked whether each platform provides the necessary functionality to the user in order to create games easily and quickly, as well as additional functionality that further supports users in developing games (Section 5.2).
- *Player captivation*: Features that captivate players in the game experience and contribute to their immersion are presented for the investigated platforms (Section 5.3).
- *Overall support*: Metrics that are considered important for making an informed choice of an LBG platform for implementing a specific game are presented based mainly on the practical use of the platforms for developing games (Section 5.4).

As already mentioned, the metrics utilized by Metikaridis and Xinogalos [5] were adopted in the comparative analysis of the platforms across all the aforementioned axes. The extended list of metrics proposed by Metikaridis and Xinogalos [5] was based on relevant literature [6–11] and metrics proposed by the authors themselves based on their practical experience of implementing LBG using the tools.

4. Implementation of LBG Prototypes

Taleblazer, ARIS, Locatify, and Actionbound are ideal platforms for students, offering comprehensive experiences and active learning opportunities. Taleblazer is suitable for educational games; ARIS is ideal for storytelling games; and Locatify and Actionbound are great for treasure hunts and guided walks. These platforms require low or no-code writing, making them ideal for users with little or no programming knowledge. They also support remixing games, allowing users to create non-standardized experiences. ARIS and Taleblazer are free, with the first one providing its source code for user contributions, while Locatify and Actionbound are free to use with limitations.

In the subsections that follow, the main features of each tool and the prototype game implemented utilizing these features are presented for Taleblazer (Section 4.1), ARIS (Section 4.2), Locatify (Section 4.3), and Actionbound (Section 4.4).

4.1. TaleBlazer

4.1.1. The Main Features of TaleBlazer

Taleblazer is a free augmented reality (AR) game platform that allows players to play both indoors and outdoors. It is a low-code platform designed for users who are novices in programming since it uses visual block programming. Taleblazer uses global positioning system (GPS) technology to locate players' positions and allow them to interact with virtual content. Taleblazer's hybrid experiences can be used for storytelling, education, social participation, and entertainment.

To create a game, users need to define locations in the real world and add elements such as agents representing characters or objects. Taleblazer consists of five software components: An online game editor, a game repository, a server for multiplayer games (currently under development (2023)), a mobile application, and an analytics server for game statistics. As shown in Figure 1, the platform comprises six tabs: map, agents, player, world, settings, and beacons.

The game's *map* section defines the real-world location of the game, which can be based on Google Maps or a map of the current location. *Agents* are visible as icons on the map, and the game can be played without an Internet connection. Multiple regions or the same region with different agents can be defined multiple times, as can internal regions.

The game designer can define characteristics and actions for the player, world, and agents. These can be visible or invisible, depending on the designer's choice. *Players* can interact with agents in five ways: GPS proximity; selecting the agent's image on the map; "Heads Up" that uses the compass and GPS readings of the device to show nearby agents

as markers overlaid onto the video camera display; selection from the inventory catalog; and Clue Code, which is a secret code that can be found throughout the game and you can input it on the related tab, so you can reveal secret agents. Roles can have names, images, descriptions, characteristics, and actions.

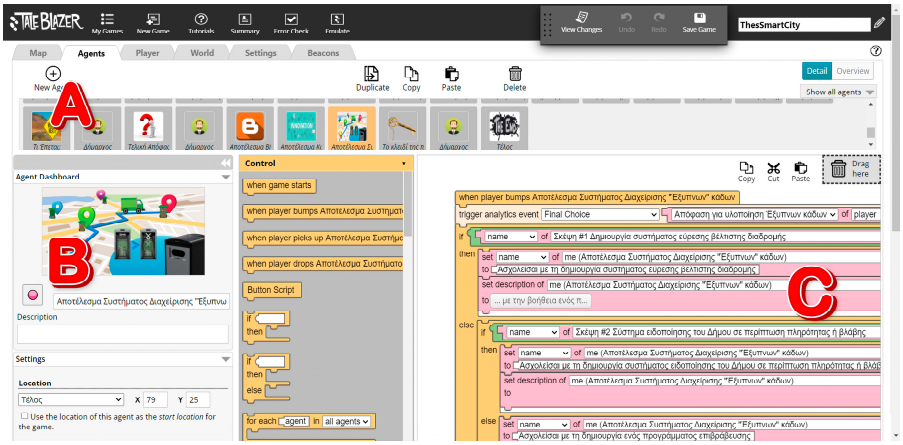

Figure 1. Taleblazer web editor, (**A**) Agents with their names from left to right: What's next? Mayor, Final Decision etc. (**B**) Name field: defining the name of the selected agent with current value "Result of "Smart Bin Management System", (**C**) Sample code segment composed of blocks: the text in Greek refers to identifiers' names and values assigned to them.

Scenarios are variations that the player selects at the beginning of the game. The designer can define different variables and parameters based on the chosen scenario. The game world encompasses settings, characteristics, and actions that are displayed to all players regardless of their location. The *settings* tab contains global settings that apply to all elements of the game and is divided into four subtabs: "Mobile Tabs", "Properties", "Bluetooth Reliability Level", and "Beacon Table". A Bluetooth beacon is a small wireless device that works based on Bluetooth Low Energy. It repeatedly transmits a constant signal that other devices can see.

The Taleblazer platform allows individuals without programming knowledge to create functionality for their games using a visual programming language called Blocks. The blocks are divided into eight categories, with the user's puzzle of blocks creating a logic similar to that of traditional programming languages (Figure 2). The game's control, operators, game, looks, movement, traits, time, and Bluetooth technology categories of blocks are all designed to help users create and manage their game.

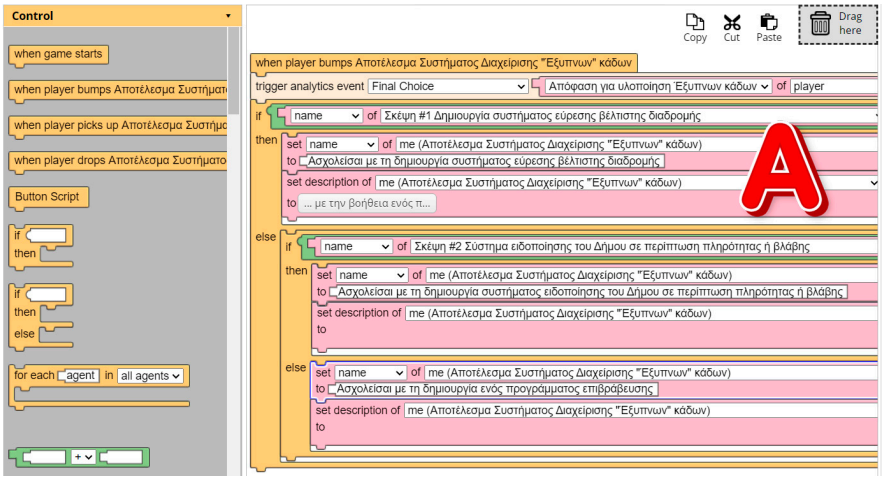

Figure 2. Taleblazer blocks programming language, (**A**) Sample code segment composed of blocks: the text in Greek refers to identifiers' names and values assigned to them.

4.1.2. The Educational LBG ThesSmartCity

ThesSmartCity is an educational game that focuses on sustainable development, specifically, in waste management, in Thessaloniki, Greece. The game involves players interacting with fictional characters, gathering information, and making thoughtful decisions to achieve their goals (Figure 3). To win the game, players must choose a series of actions to create a comprehensive proposal to submit to the mayor, obtaining the key to the city. The game is designed to enhance critical thinking skills and encourage team discussions.

Figure 3. Taleblazer ThesSmartCity options, (**A**) The task assigned to the player in Greek, (**B**) Multiple choice question (**C**) Thought #1 (decision for achieving the task), (**D**) Thought #2, (**E**) Thought #3.

The game's structural elements include a game region, a game introduction, 42 agents (Figure 4), nine different scenarios, 40 multimedia elements, one trait for each player, agent, and world, and one to five actions for each player, agent, and world. The game uses the scripting programming language "BLOCKS" to achieve interconnectivity between the game objects and smooth progression. The game was created in order to assess the capabilities of the Taleblazer tool in the development of LBGs.

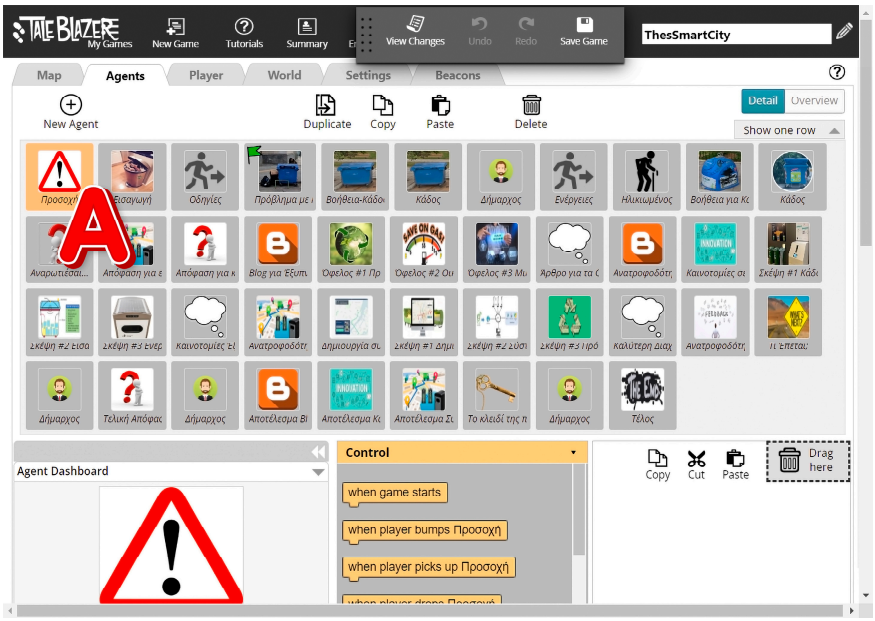

Figure 4. Taleblazer ThesSmartCity agents (**A**) Agent with their names in Greek. Agents from left to right: Caution, Introduction, Instructions, Problem with trash bins, etc.

4.2. ARIS

4.2.1. The Main Features of ARIS

ARIS is a web-based, open-source platform that was developed by Field Day Lab (Figure 5). The platform can be used for educational and touristic purposes, as it allows the creation of fantasy games in which users may interact with real-world locations while learning information about them at the same time. Another advantage of this platform is that game developers may monitor players' in-game actions and responses, which permits them to make amendments to the game.

Some of the ARIS platform features can be found below:

1. Game development through the use of a drag-and-drop interface; no knowledge of programming is required.
2. Built-in scripting language for the creation of complex in-game interactions involving riddles and branching narratives.
3. Mobile app availability: The players can take advantage of the affordances of mobile devices, such as GPS technology, cameras, and sensors, since ARIS games are available on iOS mobile devices through the ARIS app.
4. Social interaction and collaboration during gameplay: Players achieve their goals through collaboration with other players, forming their own groups.
5. Customizable game features: Players construct their own avatars and virtual items and choose in-game locations.
6. Analytics: The platform offers analytics about game usage, player conduct, and progress.
7. Rewards: Points and badges are granted to players as rewards for goal achievement and task completion.
8. Game templates and scenarios: Players are given access to pre-set game templates and scenarios that they themselves can adjust to their preferences.

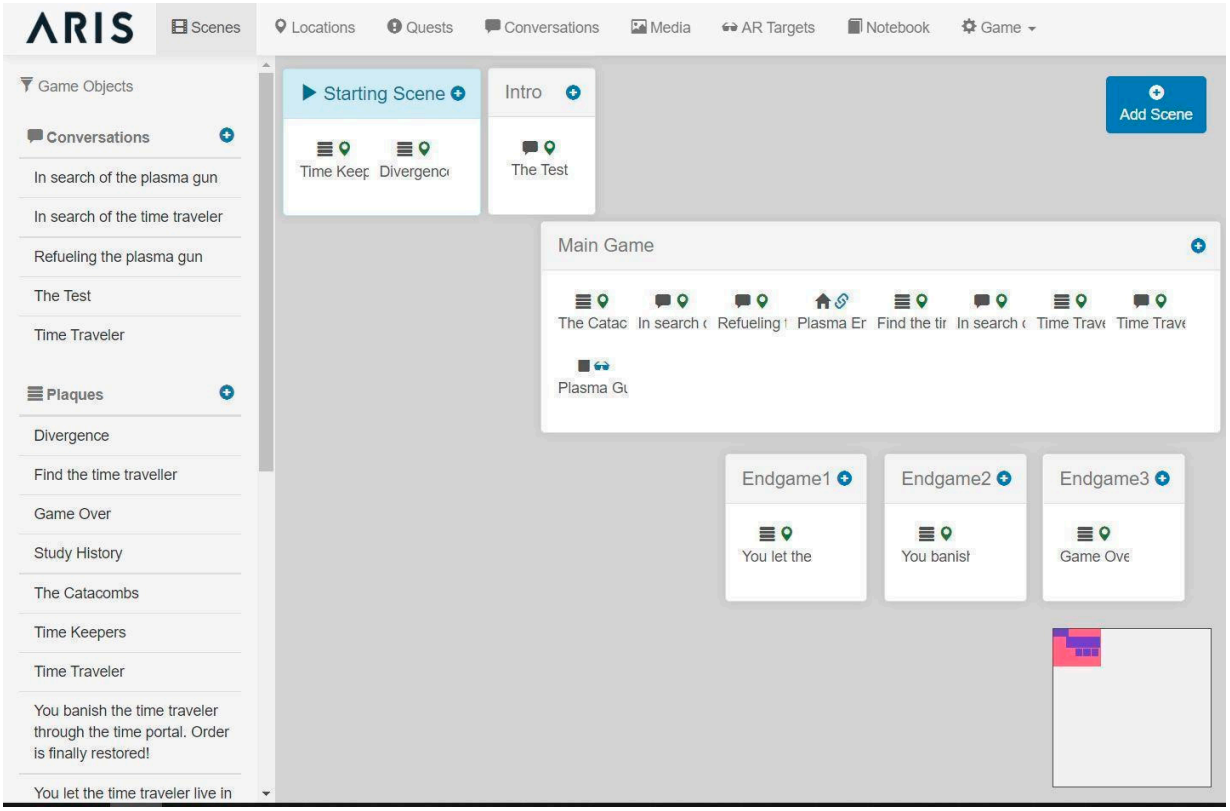

Figure 5. ARIS Time Keepers web editor.

Nevertheless, the platform heavily relies on GPS technology, a fact that might cause accessibility problems in areas with low GPS accuracy. Another disadvantage may have to do with the relatively high level of technical expertise that the design of a game requires, which might prove discouraging and time-consuming for users.

4.2.2. The Fantasy LBG Time Keepers

"Time Keepers" is an entertainment fantasy role-playing game (RPG) that involves decision making, interaction with fictional characters, and information collection (Figure 6). The game can be played several times, offering players a number of choices that will lead to three different endings and outcomes, which players are invited to explore in a hypertext fashion. The plot of the game revolves around a group of agents/guardians trying to send secret time travelers back to their timelines in order to prevent anomalies in the time continuum that will potentially lead to world-ending calamities and fatal changes in the course of human history. Historical monuments in the city of Thessaloniki, Greece, serve as portals with which players must interact in order to play the game (Figure 7).

There is a group of agents/guardians trying to send secret time travelers back to their timelines in order to prevent anomalies in the time continuum that will potentially lead to world-ending calamities and fatal changes in the course of human history. The players are introduced to the character "Anton", the leader of the time guardians, who requires the players to pass a test to receive the time guardian's badge. If the players answer incorrectly, the game ends. If they answer correctly, the next mission is to find a plasma gun to open a time portal and send the time travelers back to their timelines. The players must collect randomly generated plasma energy within a specific timeframe to charge the gun. Once the gun is charged, the players receive information that the targeted agent is near the Church of Hagia Sophia in Thessaloniki. After apprehending the agent, the players engage in a dialogue and must choose whether to exile the agent to their own timeline or allow them to stay, requesting that they cause no trouble. In the first case, the players open a time portal using the gun and exile the agent, while in the second case, the agent is left free to leave.

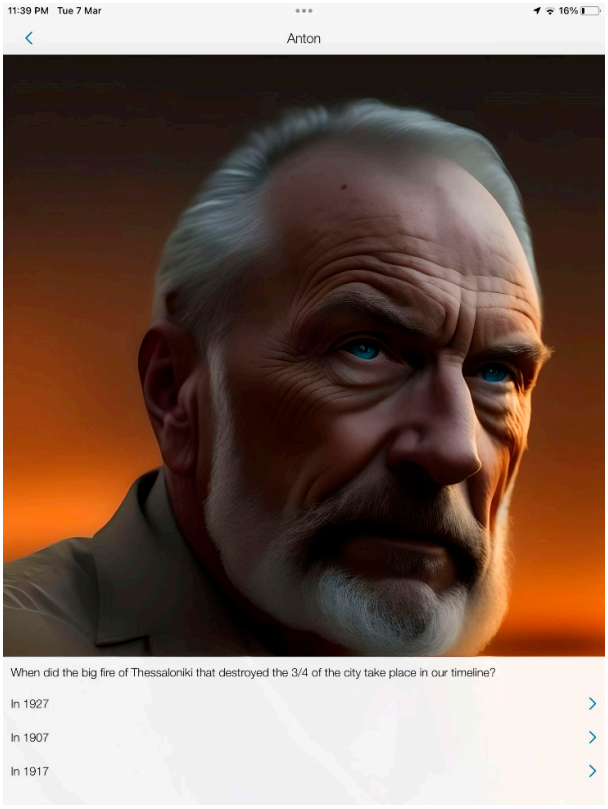

Figure 6. ARIS Time Keepers—player interaction in the form of multiple choice questions.

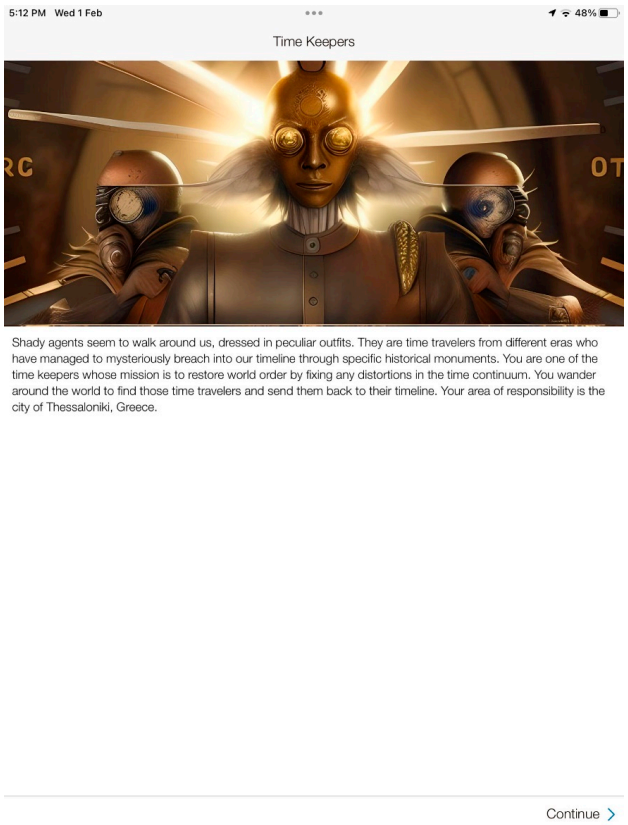

Figure 7. ARIS Time Keepers player storyline.

The "Time Keepers" game combines fantasy elements with historical aspects, guiding the players through various historical locations (Figure 8) in Thessaloniki. Players are required to fulfill various missions, such as plasma gun searches and refueling, that will result in sending them back to their era or permitting them to continue living in the current timeline with no repercussions for humanity.

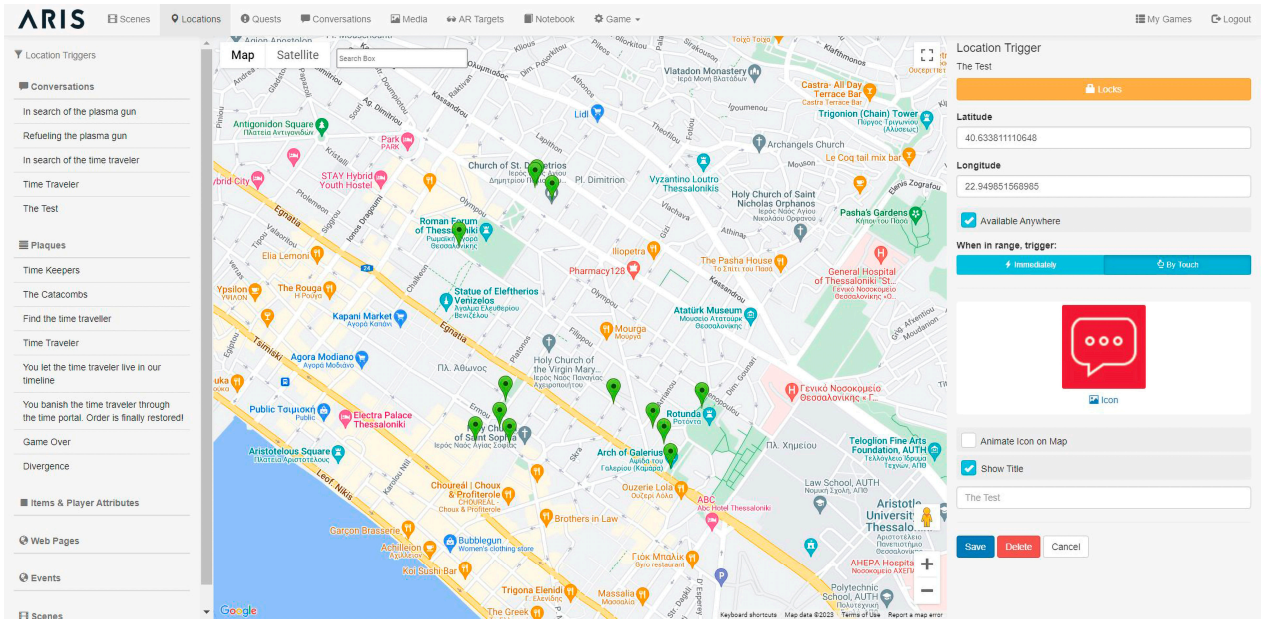

Figure 8. ARIS Time Keepers locations.

The game resorts to the use of augmented reality (AR) and GPS technology, requiring the players' exact location on the map.

4.3. Locatify

4.3.1. The Main Features of Locatify

Locatify is a no-code platform that can be used to create indoor and outdoor location-based applications. As a no-code platform, it is ideal for users without programming skills. Locatify uses GPS and Bluetooth beacons to get an accurate position of its users. The platform is customizable on the user interface, map styles, and notification settings, although some are paid features. The interface is user-friendly (Figure 9), making the creation of applications a very easy process. Locatify is also very well documented, with many tutorials to help its users along their development process.

The most important features of the Locatify platform (free version) are the following:

1. Customization: User interface, map styles, and notification settings can be customized according to the user's needs. Some features are paid for though.
2. Real-time data: The platform supports real-time user location data, making the development of dynamic applications possible. This makes it ideal when we want to build more personalized applications.
3. Multilingual support: Locatify supports multiple languages, making it available to more users worldwide who can create applications in their local language.
4. Navigation: The platform can be used to create indoor and outdoor applications with the help of GPS and Bluetooth beacons.
5. Indoor mapping: Indoor applications were recently added to the Locatify platform, and they are ideal for museums, universities, and other buildings. They can provide guidance to users based on floor layouts.
6. Gaming: Locatify can be used for the creation of treasure-hunt games that engage users to explore the physical space and cooperate.

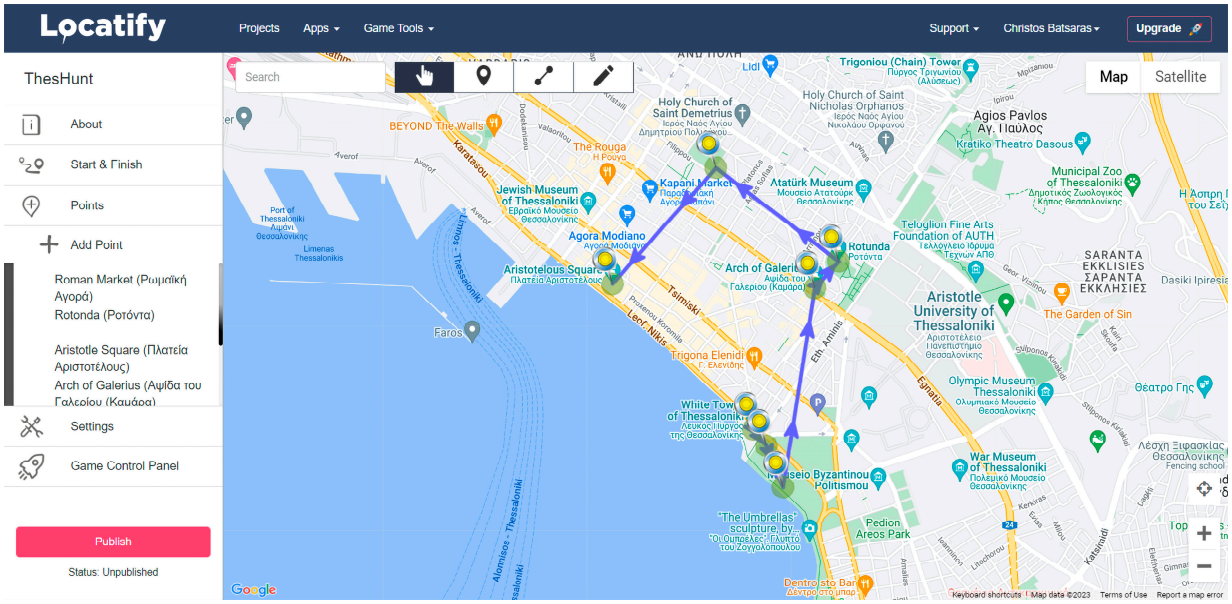

Figure 9. Locatify ThesHunt web interface.

In conclusion, Locatify can be used for various projects, from tour guides to LBGs and, recently, indoor applications. We ought to mention that in the paid version, users have extended features and customization options.

4.3.2. The Treasure Hunt LBG Game ThesHunt

The Locatify platform offers three sections: Outdoor routes, treasure hunt games, and indoor projects. The "ThesHunt" game, based on Thessaloniki (Greece), combines entertainment and educational elements (Figure 10). It is available on iOS and Android devices through the "Turfhunt" application. The application supports audiovisual content and is accessible to people with disabilities, including visual impairments. It also features screen-reading features such as VoiceOver and TalkBack. The current availability of "Turfhunt" is limited, so it would be beneficial to expand it to more cities and provide regular updates to keep the game engaging.

Figure 10. Locatify ThesHunt game intro, (**A**) Description of the game in Greek.

In the game, players select a nickname and a team and navigate through the designated map area to find objects, solve puzzles, and answer questions for points (Figure 11). The application promotes social interaction through playing with friends, forming groups, and sharing the scoreboard results on social media.

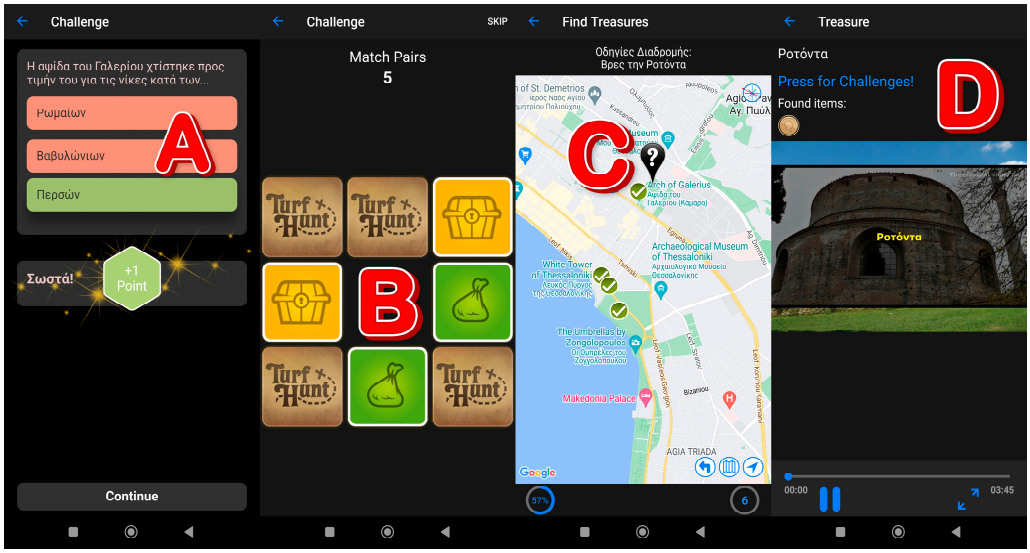

Figure 11. Locatify ThesHunt challenges, (**A**) Multiple choice question, (**B**) Match pairs game challenge, (**C**) Head to the next POI, (**D**) Descriptive video of the POI.

"ThesHunt" consists of 7 points of interest, 8 pictures, 7 challenges, and 3 multimedia files. The creation of the game consists of the following steps: Creation of a new project, adding instructions and information about the game, placing the points of interest on the map at specific locations, creating the challenges, adding multimedia content for each point of interest, adjusting the settings of the game, and publishing the game. At the end of the game, a scoreboard appears to display the achievements of a player or team, as seen in Figure 12.

Figure 12. Locatify ThesHunt scoreboard.

4.4. Actionbound

4.4.1. The Main Features of Actionbound

Actionbound is an easy-to-use, no-code platform for the creation of treasure hunts, surveys, and quizzes. The platform consists of a very intuitive drag-and-drop interface (Figure 13) that enables users to build applications with ease. Actionbound can be used to create educational as well as entertainment applications easily and quickly.

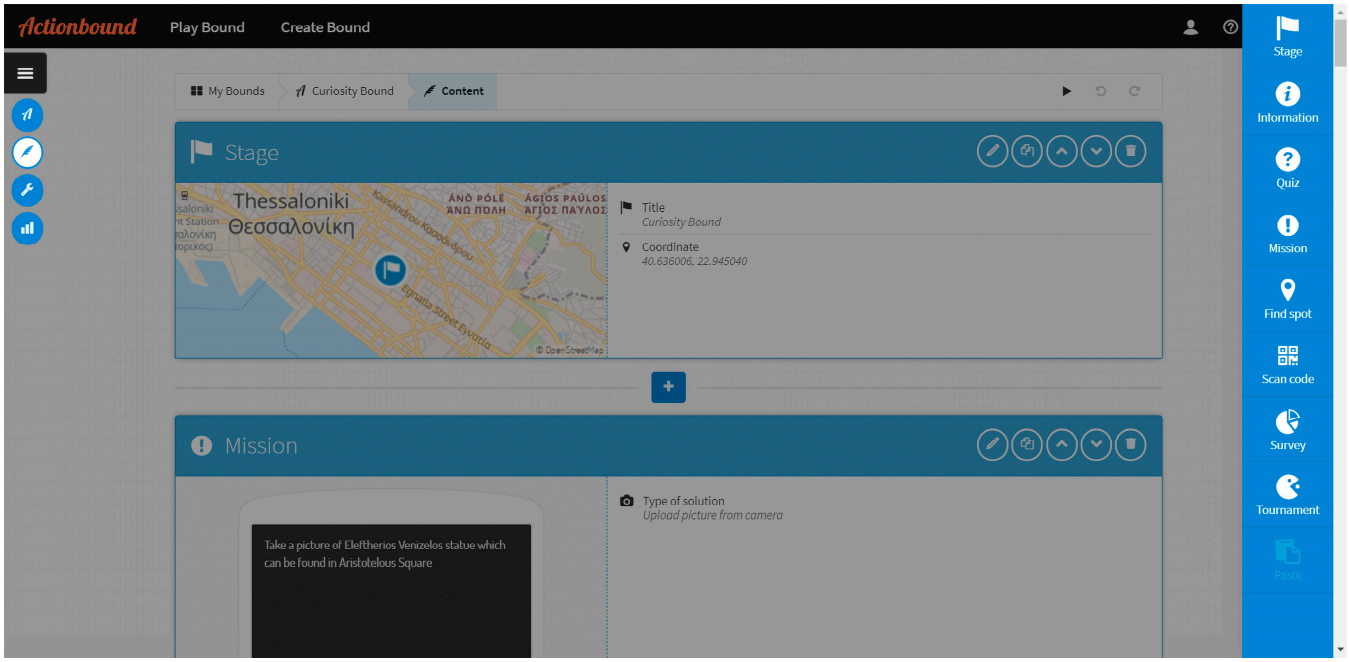

Figure 13. Actionbound Curiosity Bound web editor.

The Actionbound platform offers the ability to incorporate multimedia such as images, videos, and sounds inside the applications, providing a more immersive user experience. Another feature is the ability to create GPS missions where users need to physically travel to a given point in order to complete the mission. This is useful in cases such as treasure hunts, parks, and museums. Moreover, the platform has feedback and tracking capabilities, which are useful for educators to track the students' progress. Actionbound is multilingual, making it ideal to cover a larger user base, and supports iOS and Android devices. The platform also offers a paid plan for those who want custom branding for their application, need useful analytics, and want no limit on the number of applications that they can create.

4.4.2. The Treasure Hunt LBG Game Curiosity Bound

Facilitating the Actionbound platform, we created the "Curiosity Bound" treasure hunt game. The purpose of the game is not only entertaining but also educational. It does so by providing historical and cultural information about the city of Thessaloniki in Greece. Individuals or teams can play the game when they first choose a team and provide a nickname (Figure 14). To win, players/teams must complete the bound within a specific time set by the creator, and their ranking is displayed based on their score.

Apart from being entertaining, the game is also educational. Its objective is to entertain players while also acquiring cultural, historical, and encyclopedic knowledge about Thessaloniki. Players navigate through specific points of interest and solve various challenges to finish quickly and with a higher score than other players. Examples of game challenges can be seen in Figure 15.

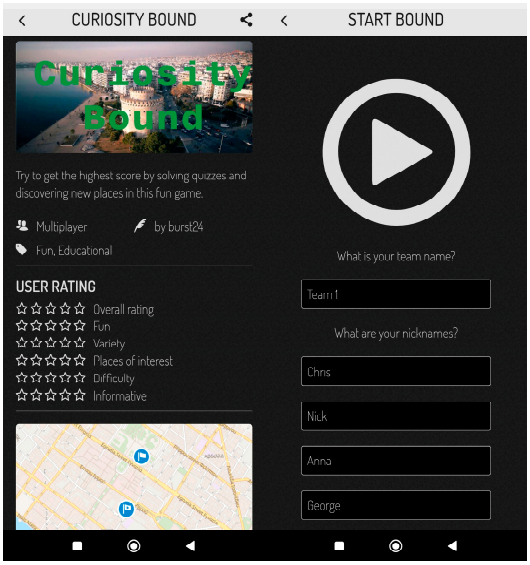

Figure 14. Actionbound Curiosity Bound game start.

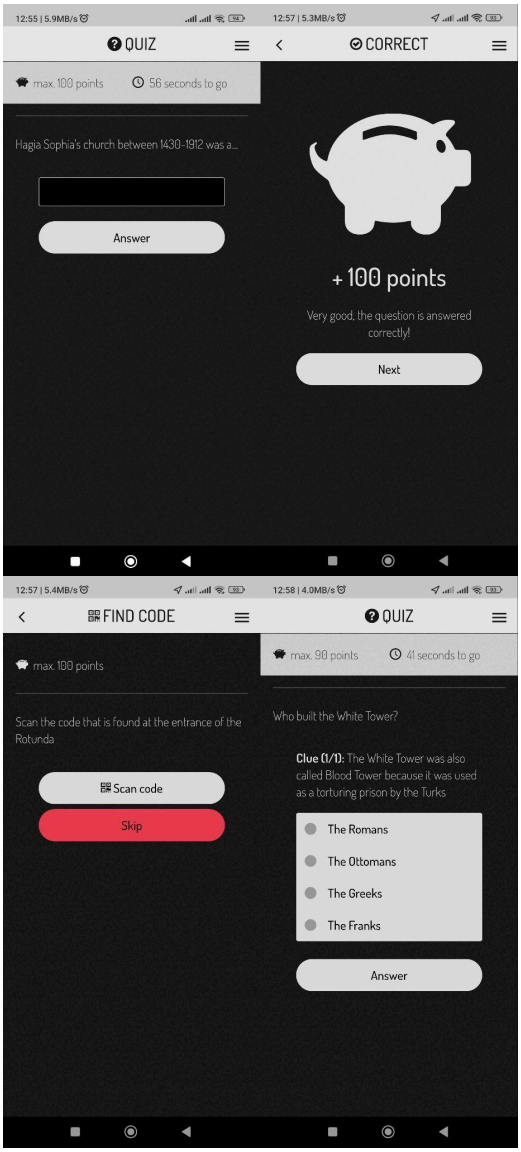

Figure 15. Actionbound Curiosity Bound challenges.

Teams start from a designated starting point. After reading the information provided through the "Info" element, players must answer the "Quiz" challenges correctly to earn as many points as possible within a limited time. In addition to the challenges, players are given "Missions" where there is no right or wrong answer. These missions can include tasks such as writing a text or uploading multimedia content. For example, players are asked to upload a photo of Eleftherios Venizelos, located in Aristotelous Square. Another type of challenge is finding a specific location based on coordinates, such as visiting the Church of Hagia Sophia. The game continues with successive pieces of information and challenges, and players are asked to scan a QR code located at the entrance of the Rotunda. The game ends with the display of achievements, scores, and a "Feedback" survey where players can evaluate the game (Figure 16).

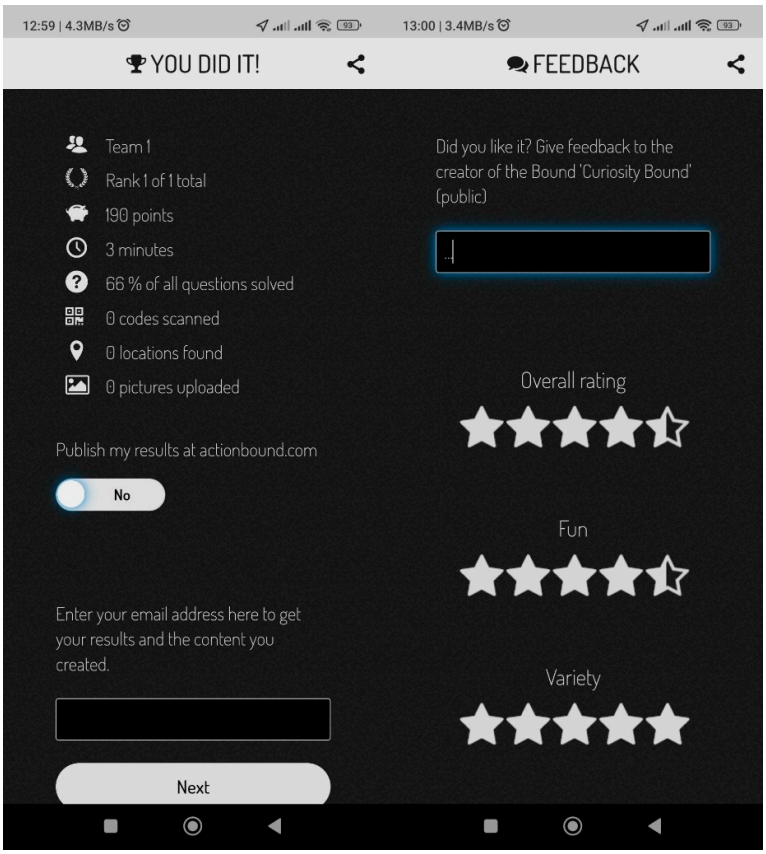

Figure 16. Actionbound Curiosity Bound game over.

The game utilizes various elements provided by the Actionbound platform, including scenes, missions, location findings, information, quizzes, QR code scans, and surveys. These elements contribute to players' scores, except for scenes, information, and surveys.

5. Comparative Analysis

In this section, a comparative analysis of the four LBG platforms is presented. Specifically, the platforms are comparatively analyzed in the following axes: LBG constituent features; development functionality; player captivation; and overall support provided for inexperienced users.

5.1. Mapping of LBG Constituent Features with Platform Features (RQ1)

Table 1 presents the constituent features of LBGs, along with the corresponding terms used by each development platform. Specifically, Table 1 shows how these basic features are implemented on each platform. For example, the concept of "player" is referred to as "player" on all platforms, while the concept of "world" varies from platform to platform,

being referred to as "world", "scene", "project", or "bound". The features included in Table 1 are defined in previous works [5,6]. Specifically, Xanthopoulos and Xinogalos [6] proposed as constituent LBG features the following:

- The "Map" that corresponds to the game space.
- The "Area" that usually represents a field in the real world.
- The "Quest" that corresponds to a challenge/mission within the game.
- The "Track" that indicates the path followed by the player.
- The "NPC" that refers to non-player virtual characters.
- The "Bump" that denotes that the player is about to enter the space of another game entity.
- The "Item" that refers to objects that can be picked up and carried in the inventory.
- The "Unlock" denotes the activation of a previously unavailable option.
- The "Dialog" represents a conversation, usually with an NPC.
- The "Progress" indicates the progress of the player in the game.

Table 1. Mapping of LBGs features with platforms' features.

Game Element	TaleBlazer	ARIS	Locatify (Free Version)	Actionbound (Free Version)
Player	Player	Player	Player	Player
World	World	Scene	Project	Bound
Class	Role	Group	-	-
Scenario	Scenario	Player attribute/hidden item	-	-
Map	Map	Map/locations	Map	Map
Area	Region	Plaque	Location	Stage
Quest	Player trait + agent button	Quest	-	-
Track	-	-	Path	Track
NPC	Agent	Character	-	-
Bump	Bump settings	Map trigger	Map trigger	Map trigger
Item	Agent	Item	Treasure	Points
Unlock	Block scripting (trait button Visibility/agent inclusion)	Locks	-	-
Dialog	Agent description + agent button	Conversation	-	-
Progress	Player/world trait	Player attribute/current scene	Progress bar	Progress bar
Operations	Buttons/ block scripting	Events/ARISjs	-	-
Label	Private traits	Tags	-	-
Page	Block scripting (show/launch url...)	Web pages	-	-
Spawn	Block scripting (every.../create clone...for each clone do...)	Factories	-	-
Timer	Block scripting (every.../elapsed time...)	Timer trigger	Timed questions	Timed questions

Metikaridis and Xinogalos [5] adopted the aforementioned features in their study and proposed some additional features for a better comparison of relevant tools:

- The term "Player" refers to each virtual character in the game controlled by a user, rather than any form of AI.
- The "Class" of the player refers to the category to which they belong and is used to highlight the character's attributes, advantages, and disadvantages.

- The term "World" refers to the overall physical and digital space in which the player can move in the game, including different missions.
- The term "Scenario" refers to all possible paths that the game can take in terms of duration, plot, or termination, determined by the player's initial choices.
- The term "Operations" refers to the programmed or improvised activities within the game.
- The term "Label" is the characteristic that helps distinguish similar objects.
- The concept of a "Page" refers to the ability to display a webpage within a game.
- The concept of "Spawn" denotes the creation of new objects within the game when specific criteria are met.
- The term "Timer" refers to recording the elapsed time from a specific moment with the goal of executing an action in the future.

Regarding the number of supported features and the correlation of concepts in LBGs, we observe that most concepts are identified in the Taleblazer and ARIS platforms, while fewer concepts are found in Locatify and Actionbound (free versions). It is also noted that ARIS has automated the use of most concepts, reducing the need for programming, while Taleblazer requires programming to connect individual concepts and develop the necessary functionality. However, the use of the Blocks programming interface makes the process easier. The remaining platforms do not allow programming.

5.2. Development Functionality (RQ2)

The results of the comparative analysis of the platforms in terms of the necessary functionality provided to users who create location-aware games are presented in Table 2. The metrics utilized are the ones used in the study by Metikaridis and Xinogalos [5], which in turn were based on the work of Winter [8] and Xanthopoulos and Xinogalos [7].

Table 2. Development functionality.

Developer Functionality	TaleBlazer	ARIS	Locatify (Free Version)	Actionbound (Free Version)
Non-linear authoring	Yes (*if-then* block)	Yes (locks/ triggers/events)	No	No
Visual authoring	Yes (GUI editor)	Yes (GUI editor)	Yes (GUI editor)	Yes (GUI editor)
In situ authoring	No	No	No	No
Re-use and re-editing	Yes (remix/copy)	Yes (import/duplicate)	No	Yes (Copy/ import/duplicate)
Content adding and management	Partial (cannot delete/ rename multimedia)	Yes (*multimedia* tab)	Yes (*media manager*)	Yes (*media Library*)
Editor customization	Partial (foldable menus)	No	No	No
Simulation mode	No (discontinued)	No	Yes (*test Version*)	Yes (*test bound*)
Game analytics	Partial (official partners)	No	Yes (*game control panel*)	Yes (*results*)
Map authoring	Yes (*map* Tab)	Yes (*locations* tab)	Yes (*Map* tab)	Yes (*map* tab)
Visual programming	Yes (block scripting language)	Partial (locks/events)	No	No
Programming interface	No	Yes (JavaScript)	No	No

It is notable that none of the platforms supports the capability of in situ authoring, while ARIS is the only platform that does not offer detailed statistical data (analytics) for the game. TaleBlazer supports such features only for official collaborators and is the only platform that allows partial customization of the interface's graphical elements (editor customization). TaleBlazer is also the only platform that fully supports a visual programming language, while it does not have a programming interface for non-visual

languages (programming interface). ARIS seems to be in the middle, seeking to combine both the visual programming system using locks and events, as well as the non-visual programming system through the ARIS programming interface using JavaScript. Finally, we observe that the platforms Locatify and Actionbound do not offer programming capabilities either with visual or non-visual languages or some form of non-linear game design (non-linear authoring).

Table 3 summarizes the results of the comparative analysis of the additional functionality offered by the LBG platforms. The metrics used are adopted from the work by Metikaridis and Xinogalos [5] and include functionality features that were defined by the authors themselves, as well as other researchers [9–11].

Table 3. Additional developer functionality.

Additional Functionality	TaleBlazer	ARIS	Locatify (Free Version)	Actionbound (Free Version)
Game start event	*Settings/introduction* Block scripting (when game starts)	Starting scene	Start	-
Game end event	Block scripting	Quest + end scene	Finish	-
Scheduled and random events	Block scripting	Timer trigger	-	*Stages/random mode*
Tap to bump	*Settings/bump settings/allow tap to bump* tab	*Game/settings/offsite mode* tab	*Settings/map and location settings*	-
On-screen navigation	Map tab	Map tab	Map tab	Map tab
User input	Buttons, clue code tab, trait and action passwords	Conversation choices, QR scanner, decoder codes	Buttons, text boxes	Buttons, text boxes, QR scanner
Data collection	-	Notebook	-	-
Multimedia sharing between games	Stored in MyFiles and MyIcons	-	-	-
User application customization	User tabs only	User tabs only	-	-
Automated version control	Game revision history	-	-	*Publish/version control*
Shared editing	-	*Game/sharing* tab	-	-
Game export	Summary	*Game/settings/export* tab	*Projects/export*	-
Help button	Tutorials	-	*Support/help center*	*i (info)*
Error checking	Error check	-	-	-

As for the additional development functionality, it is worth noting that TaleBlazer is the only platform that supports multimedia sharing between different games and also provides an error-checking system. Its most significant drawback is the lack of support for shared editing, which would allow multiple users to collaborate on a project, enabling faster completion. ARIS is the only platform that supports data collection using a notebook, but it lacks user assistance (help button). TaleBlazer and Actionbound are the only platforms that offer an automated version control system, allowing users to revert to a previous version of their project. TaleBlazer and ARIS are also the only platforms that allow user application customization, enabling users to customize the tool to their preferences. Locatify has the drawback of lacking scheduled and random events during gameplay and the inability to share a project for parallel development by more than one person. Finally, Actionbound lacks a game start event and game end event, a tap-to-bump feature for visits, and the ability to export the game in any form.

5.3. Player Captivation Capabilities (RQ3)

In this section, the results of the player captivation capabilities of the comparatively analyzed LBG platforms are summarized (Table 4). The list of capabilities was composed by Metikaridis and Xinogalos [5] based on relevant literature [8–10].

Table 4. Player captivation capabilities.

Player Captivation	TaleBlazer	ARIS	Locatify (Free Version)	Actionbound (Free Version)
Multiple media formats	Yes (images: jpg, png, jpeg, gif. Sounds: wav, mp3, m4a, wma, 3gp. Videos: mp4, wmv, mpg, mov, 3gp.	Yes (images: jpg, png, jpeg, gif, .bmp. Sounds: wav, mp3. Videos: mp4)	Yes (images: jpg, png, jpeg, gif, bmp. Sounds: wav, mp3, ogg. Videos: mp4, webm)	Yes (images: jpg, png, jpeg, gif, bmp. Sounds: wav, mp3. Videos: mp4, webm. Location data: GeoJSON, GPX, KML)
Multiple sensing technologies	Partial (GPS, BB)	Yes (GPS, QRC, BB, AR targets)	Yes (GPS, BB, UWB)	No (GPS only)
Reflection	Yes (*log/history* tab, analytics)	Yes (notebook)	Yes (game observer, game control panel)	Yes (results)
Communication	No (no chat system)	No (no chat system)	No (no chat system)	Yes (chat system)
Collaboration	No (no multiplayer)	Partial (+player Locations and player classes. No trade/ diplomatic system)	Partial (+player locations. No trade/ diplomatic system)	Partial (+player locations. No trade/ diplomatic system)
Multiple players	No (multiplayer under development)	Yes (multiplayer options In *game/settings* tab)	Yes (mutiplayer options in app)	Yes (mutiplayer options in app)
Multiple operating Systems	Yes (Android, iOS)	No (IOS only)	Yes (Android, iOS)	Yes (Android, iOS)
Connectionless gameplay	Yes (pre-download possible)	No (constant server connection)	Yes (pre-download possible)	Yes (pre-download possible)
Outdoor–indoor playability	Yes (GPS, BB)	Yes (GPS, BB)	Yes (GPS, BB, UWB)	Yes (GPS, QR)

As for player captivation, we observe that ARIS and Locatify are the only tools that use multiple sensing technologies for player positioning. ARIS does not support multiple operating systems, requiring a continuous internet connection for gameplay (connectionless gameplay). Regarding player collaboration, TaleBlazer is the only platform that does not yet support multiplayer games.

5.4. Overall Support for Inexperienced Users (RQ4)

At this point, we will discuss some empirical metrics that emerged from the research by Metikaridis and Xinogalos [5] and serve as criteria for selecting the best platform for developing LBGs. The respective results are presented in Table 5, while in Table 6, the results for some metrics, such as "Friendly Developer UI" and the classifications "Maximal/Moderate/Minimal", which are subjective and are based on the author's subjective judgment, are presented.

Regarding the TaleBlazer platform, its advantages include a friendly player UI, free usage, and great prospects for educational purposes. Its drawbacks include the absence of extensibility toolkits for additional programmatic functionality, lack of a digital gathering/discussion space (forum), and a low number of active community members.

Regarding the ARIS platform, its advantages are that it is open source and has an active forum and community. It also offers the most educational videos (video tutorials). The drawbacks include various programming errors (bugs) in the online editor, limited functionality for the iOS operating system, and the need for a continuous internet connection to use it.

Table 5. Overview and feedback.

Metric	TaleBlazer	ARIS	Locatify (Free Version)	Actionbound (Free Version)
Open source	N (no source code)	Y (source code)	N (no source code)	N (no source code)
Release date	October, 2018	March, 2011	September, 2009	2012
Latest version	March, 2020 (Taleblazer 3.2.14)	October, 2019 (ARIS 2.7.7)	August, 2023 (Turfhunt 9.0.1)	July, 2023 (Actionbound 2.16.4)
Costs/price	Completely free	Free until 100 players monthly	One published app limit, 10 teams in-game max, duration 48 h max	Free for personal use
Extensibility toolkits	-	SIFTR, Vuforia	-	-
Game dimensions	2D	2D	2D	2D
Development platforms	Chrome/Firefox	Chrome/Firefox	Chrome/Firefox	Chrome/Firefox
Server–client architecture	Yes	Yes	Yes	Yes
Editor usage	Cloud	Cloud	Cloud	Cloud
Official support	Email/outdated forum	No longer supported (since 2023)	Help center (live chat, requests, FAQ, new forum since September 2022 with low activity)	Help center (Tel., email, FAQ, support forum with fast replies)
Active community	Minimal	Maximal	Minimal	Maximal
Documentation and manuals	Maximal	Maximal	Maximal	Maximal
Video tutorials	Moderate	Maximal	Maximal	Maximal
Game demos	Maximal	Maximal	Maximal	Maximal
Playing system requirements	Moderate	Moderate	Moderate	Moderate
Development system requirements	Minimal	Minimal	Minimal	Minimal

Table 6. Author's subjective assessment.

Metric	TaleBlazer	ARIS	Locatify (Free Version)	Actionbound (Free Version)
Friendly developer UI	Moderate	Maximal	Maximal	Maximal
Friendly player UI	Maximal	Moderate	Maximal	Maximal
Learning curve	Moderate	Moderate	Minimal	Minimal
General advantages	Great educational potential Plenty of training material User-friendly GUI Extra developer Functionality	Great educational potential Plenty of training material User-friendly GUI Truly open source	Great educational potential Plenty of training material User-friendly GUI	Great educational potential Plenty of training material User-friendly GUI
General disadvantages	Reduced AR features No multiplayer games Limited multimedia management capacities	iOS only Internet connection required Minor editor bugs	Limited customization Limited functionality on free plan Medium learning curve	Limited customization Limited functionality on free plan Medium learning curve

As for the Locatify platform, the positives include a friendly developer UI and a friendly player UI, extensive educational material (documentation, manuals, video tutorials), and significant educational benefits for younger ages due to its user-friendly interface. On the other hand, the negatives include limited functionality in the free plan and limited customization of the user interface.

Finally, regarding the Actionbound platform, its advantages include free personal use, while it shares the same pros and cons as the Locatify platform.

6. Conclusions

In this section, we will present the conclusions drawn from the evaluation of the effectiveness of the LBG development tools we examined. This evaluation takes into account both the literature review and the results obtained from the personal exploration of the tools for creating prototype games. The conclusions drawn for each one of the four research questions are presented at the end of the corresponding subsections of the Comparative Analysis section, as follows: The conclusions regarding the mapping of LBG's constituent features with platform features (RQ1) are presented at the end of Section 5.1; the conclusions regarding the basic and additional development functionality of the tools (RQ2) are presented at the end of Section 5.2; the conclusions drawn on the player captivation capabilities offered by the tools are presented at the end of Section 5.3; and finally, the conclusions drawn on the support provided by the tool for inexperienced users are presented at the end of Section 5.4. In the following paragraphs, we try to synthesize the conclusions already drawn in the context of each RQ with the goal of drawing general conclusions regarding the investigated tools and their suitability for developing LBGs in the fields of research and education.

The four platforms that were studied both in the literature and through the implementation of prototype games are among the most widely used for developing LBGs with low or no code. In terms of educational aspects, TaleBlazer and ARIS are ideal due to their capabilities to provide a comprehensive learning experience for students while allowing them to take an active role in the learning process (active learning). Specifically, TaleBlazer is more suitable for educational games, while ARIS is well suited for narrative-based games. The other platforms, although they may lack some of the advanced features of the former two in their free versions, excel in simplicity and ease of use, making them ideal for younger students to create simpler LBGs more quickly compared to the former platforms.

Regarding coding requirements, it is worth noting that all platforms were designed to serve users with limited or no programming knowledge in order to achieve the desired functionality and dynamic behavior of game elements. While TaleBlazer is the only platform that requires code scripting (block scripting) to connect game elements and ensure smooth flow, the code writing process is relatively easy as it relies on the integration of logical building blocks. This does not require knowledge of traditional programming languages.

In terms of reusable games, the only platform that does not support this feature is Locatify. Reusable games refer to fully functional games in which differentiation of certain elements, such as the scenario or the multimedia content involved, is sufficient to create a different game while maintaining the same dynamics and mechanics. Creating non-reusable experiences on all platforms is easily and quickly achievable, as the tools provide free teaching materials on their respective websites. These materials include various guides and videos, offering necessary support to new users of the tools. Through these resources, users can gain an understanding of the essential concepts and discover the various possibilities offered by these tools.

Regarding the features of the tools, the most comprehensive experience can be provided by TaleBlazer and ARIS. Therefore, another distinction among the platforms lies in the significantly shorter learning time required for Locatify and Actionbound compared to TaleBlazer and ARIS, which require significant study of their functionalities and engagement for users to effectively utilize them. In terms of usability, TaleBlazer is an ideal tool for users who want to develop educational games, while ARIS is suitable for narrative-based

games. Locatify and Actionbound can be used for educational games, treasure hunt games, tourist guides, and indoor navigation.

In conclusion, it is worth noting that Locatify and Actionbound have reduced customization and functionality in their free versions. In contrast, TaleBlazer and ARIS are available with all their features for free, with the latter even providing open-source code for users to contribute to its evolution.

Finally, we have to note that the study presented has some limitations. The most important one lies in the fact that the prototype games were developed by one person, the first author of this paper, and the recorded experience cannot be considered as robust as if a number of end users had utilized the tools and commented on their experience. However, the results of the comparative analysis can contribute to a better understanding of the strengths and weaknesses of the investigated LBG authoring tools, which can support both experienced and inexperienced LBG designers as well as LBG authoring tool designers.

Author Contributions: Conceptualization, C.B. and S.X.; methodology, C.B. and S.X.; software, C.B.; investigation, C.B.; resources, C.B. and S.X.; data curation, C.B.; writing—original draft preparation, C.B.; writing—review and editing, C.B. and S.X.; visualization, C.B.; supervision, S.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All data are included in the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. De Souza e Silva, A. Mobile narratives: Reading and writing urban space with location-based technologies. In *Comparative Textual Media: Transforming the Humanities in the Postprint Era*; Hayles, N.K., Pressman, J., Eds.; University of Minnesota Press: Minneapolis, MN, USA; London, UK, 2013; pp. 33–52.
2. Gordon, E.; de Souza e Silva, A. *Net Locality: Why Location Matters in a Networked World*; John Wiley & Sons: Chichester, UK, 2011.
3. Komninos Papaevangelou, E. Location-Aware Mobile Entertainment Applications. Master's Thesis, University of Macedonia, Thessaloniki, Greece, 2020.
4. Metikaridis, D. Overview and Comparative Analysis of Location-Based Games Development Tools. Master's Thesis, University of Macedonia, Thessaloniki, Greece, 2020. (In Greek).
5. Metikaridis, D.; Xinogalos, S. A Comparative Analysis of Tools for Developing Location Based Games. *Entertain. Comput.* **2021**, *37*, 100403.
6. Xanthopoulos, S.; Xinogalos, S. Opportunities and challenges of mobile location-based games in education: Exploring the integration of authoring and analytics tools. In Proceedings of the 2018 IEEE Global Engineering Education Conference (EDUCON), Santa Cruz de Tenerife, Spain, 17–20 April 2018; pp. 1797–1805.
7. Xanthopoulos, S.; Xinogalos, S. An Overview of Location-based Game Authoring Tools for Education. In *Interactive Mobile Communication, Technologies and Learning*; Auer, M.E., Tsiatsos, T., Eds.; Springer: Cham, Switzerland, 2018; Volume 725, pp. 201–212.
8. Winter, M. Location Based Games Authoring. Literature Review to Inform the Development of a Location Based Games Authoring Tool for Secondary School Children. Project Deliverable, University of Brighton, Brighton, UK, 2009.
9. Maraki, M. Overview of Mobile Location-Based Serious Games for Educational Purposes. Master's Thesis, University of Macedonia, Thessaloniki, Greece, 2018. (In Greek).
10. Siakavaras, I.; Papastergiou, M.; Comoutos, N. Mobile Games in Computer Science Education: Current State and Proposal of a Mobile Game Design that Incorporates Physical Activity. In *Research on e-Learning and ICT in Education*; Mikropoulos, T.A., Ed.; Springer: Cham, Switzerland, 2018; pp. 243–255.
11. Heinz, T.; Schlieder, C. A Pattern-based Framework for Designing Location-based Games. In Proceedings of the AGILE Conference, Lund, Sweeden, 12–18 June 2018.
12. Groundspeak—Geocaching. Available online: https://www.geocaching.com/play (accessed on 18 August 2023).
13. Bunting, B.S., Jr. The Geocacher as Placemaker: Remapping Reality through Location-Based Mobile Gameplay. In *The Mobile Story: Narrative Practices with Locative Technologies*; Farman, J., Ed.; Routledge: New York, NY, USA, 2014; pp. 161–174.
14. Farman, J. *Mobile Interface Theory: Embodied Space and Locative Media*, 1st ed.; Routledge: New York, NY, USA, 2012.
15. Munzee. Available online: https://www.munzee.com/ (accessed on 18 August 2023).

16. De Souza e Silva, A.; Frith, J. *Mobile Interfaces in Public Spaces: Locational Privacy, Control, and Urban Sociability*; Routledge: New York, NY, USA, 2012.
17. Leorke, D. Location-based Gaming Apps and the Commercialization of Locative Media. In *Mobility and Locative Media: Mobile Communication in Hybrid Spaces*; De Souza e Silva, A., Sheller, M., Eds.; Routledge: London, UK; New York, NY, USA, 2015; pp. 132–148.
18. Delioglanis, V.N. Locative Media and Narrative in North American Literature and Culture. Ph.D. Thesis, School of English, Aristotle University of Thessaloniki, Thessaloniki, Greece, 2020.
19. Leorke, D. *Location-Based Gaming: Play in Public Space*; Palgrave Macmillan: Singapore, 2018.
20. Ingress. Available online: https://ingress.com/game/ (accessed on 18 August 2023).
21. Pokémon GO. Available online: https://pokemongolive.com/ (accessed on 18 August 2023).
22. Harry Potter: Wizards Unite. Available online: https://play.google.com/store/apps/details?id=com.nianticlabs.hpwu.prod&hl=en&gl=RU (accessed on 18 August 2023).
23. Leaver, T.; Willson, M. Social Networks, Casual Games and Mobile Devices: The Shifting Contexts of Gamers and Gaming. In *Social, Casual, and Mobile Games: The Changing Gaming Landscape*; Leaver, T., Willson, M., Eds.; Bloomsbury: New York, NY, USA, 2016; pp. 1–12.
24. Montola, M.; Stenros, J.; Waern, A. *Pervasive Games: Theory and Design/Experiences on the Boundary Between Life and Play*; Morgan Kaufmann Publishers: Burlington, MA, USA, 2009.
25. Hjorth, L.; Richardson, I. *Gaming in Social, Locative, and Mobile Media*; Palgrave Macmillan: New York, NY, USA, 2014.
26. Hinske, S.; Lampe, M.; Magerkurth, C.; Röcker, C. Classifying pervasive games: On pervasive computing and mixed reality. In *Concepts and Technologies for Pervasive Games—A Reader for Pervasive Gaming Research*; Shaker Verlag: Düren, Germany, 2007; Volume 1, pp. 1–20.
27. De Souza e Silva, A.; Sutko, D.M. (Eds.) Merging Digital and Urban Playspaces: An Introduction to the Field. In *Digital Cityscapes: Merging Digital and Urban Playspaces*; Peter Lang: New York, NY, USA, 2009; pp. 1–17.
28. Mscape. Available online: https://elmcip.net/platformsoftware/mscape (accessed on 18 August 2023).
29. Wherigo. Available online: https://wherigo.com/ (accessed on 18 August 2023).
30. Gutierrez, L.; Nikolaidis, I.; Stroulia, E.; Gouglas, S.; Rockwell, G.; Boechler, P.; Carbonaro, M.; King, S. Far-Play: A framework to develop augmented/alternate reality games. In Proceedings of the 2011 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops), Seattle, WA, USA, 21–25 March 2011; pp. 531–536.
31. Fidas, C.; Sintoris, C.; Yiannoutsou, N.; Avouris, N. A survey on tools for end user authoring of mobile applications for cultural heritage. In Proceedings of the 2015 6th International Conference on Information, Intelligence, Systems and Applications (IISA), Corfu, Greece, 6–8 July 2015; pp. 1–5.
32. Field Day—ARIS. Available online: https://fielddaylab.org/make/aris/ (accessed on 18 August 2023).
33. Taleblazer. Available online: http://taleblazer.org/ (accessed on 18 August 2023).
34. Actionbound. Available online: https://en.actionbound.com/ (accessed on 18 August 2023).
35. Locatify Apps, Tour Guides & Treasure Hunt Games. Available online: https://locatify.com/ (accessed on 18 August 2023).
36. Motive.io. Available online: https://www.motive.io/ (accessed on 18 August 2023).
37. STQRY—Connecting People, Places & Stories. Available online: https://stqry.com/ (accessed on 18 August 2023).
38. ARLearn. Available online: https://streetlearn.appspot.com/#/home (accessed on 18 August 2023).
39. Arango-López, J.; Valdivieso, C.C.C.; Collazos, C.A.; Vela, F.L.G.; Moreira, F. CREANDO: Tool for creating pervasive games to increase the learning motivation in higher education students. *Telemat. Inform.* **2019**, *38*, 62–73.
40. Maia, L.F.; Nolêto, C.; Lima, M.; Ferreira, C.; Marinho, C.; Viana, W.; Trinta, F. LAGARTO: A LocAtion based Games AuthoRing TOol enhanced with augmented reality features. *Entertain. Comput.* **2017**, *22*, 3–13.
41. Kyza, E.A.; Georgiou, Y. Scaffolding augmented reality inquiry learning: The design and investigation of the TraceReaders location-based, augmented reality platform. *Interact. Learn. Environ.* **2019**, *27*, 211–225.
42. E Silva, A.D.S.; Sutko, D.M. (Eds.) *Digital Cityscapes: Merging Digital and Urban Playspaces*; Peter Lang: New York, NY, USA, 2009; Volume 57.
43. Turfhunt. Available online: https://play.google.com/store/apps/details?id=com.locatify.treasurehunt&hl=el&gl=US (accessed on 18 August 2023).