Research Article

Issues in the Development of Location-Based Games

João Tiago Pinheiro Neto Jacob and António Fernando Coelho

Faculdade de Engenharia, Universidade do Porto—DEI/INESC, Porto 4200-465, Portugal

Correspondence should be addressed to João Tiago Pinheiro Neto Jacob, joao.jacob@fe.up.pt

Received 30 September 2010; Accepted 25 January 2011

Academic Editor: Narendra Chaudhari

Copyright © 2011 J. T. P. N. Jacob and A. F. Coelho. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

As new and more powerful mobile devices arrive, such as smartphones, tablets and PDAs, so do new ways of interaction emerge for the users and developers to explore. These devices allow for the creation of never-before-seen applications, using the broad range of inputs and sensors these devices integrate (such as accelerometers, light sensors, electronic compasses, GPS sensor among others), that allow for more ubiquitous and personalized experiences. This paper explores issues that arise during the development and design phases of such games, as well as possible solutions for them. Finally we present, a framework for the development of location-based games that is sensible to these issues. This framework was created and tested thanks to a proof-of-concept location-based game (Geo-Wars).

1. Introduction

All software applications are bound to being limited or to introduce some issues to the users, which may render them, under certain conditions, nearly useless. Games are not all that different, often becoming a frustrating experience rather than a joyful one.

Location-based games are often unpredictable, as they often use the player's real location as a means of input, and so, they frequently provide very discrepant gaming experiences. Since they rely upon location-based services for working, or more simply and usually, a GPS module, these games are almost solely found on mobile platforms, as only these platforms provide the computational requirements for the game and the necessary connectivity modules. Alas, certain issues are sure to arise due to the usage of such unpredictable data (the location-based data). These games are known to become unplayable or to provide a limited gaming experience in several circumstances, such as when the player is inside of buildings, as the position is hardly determinable by the GPS module, or when the location-based data is unavailable or poor for the given player's coordinates (such as when there is no information for the specific player's location, or the information available does not provide a good expectable gaming experience). Also, since there is

no absolute control on level design, because it depends on the player's location, it is very difficult for the developers to guarantee an even game experience. Additionally, the player's physical aptitude must be considered if the location-based-game requires the player to move around the real world. As an example, a location-based game that requires the player to move from two places may be considered as difficult, boring or tiresome for some of the more lazy players, whereas the more athletic ones may consider it a challenge too easy to be bothered with. Since these issues are often somewhat more or less serious depending on the location-based games' concepts, the appliance of agile methodologies that allow for prototyping and the gathering of feedback is advisable so as to determine the severity of these issues in the game and if they are worth solving so as to provide an evenly good gaming experience for everybody.

2. Location-Based Games

A Location-based game is a game that uses the player's physical location, usually via a GPS sensor module, as an input or as a base for the generation of the game level or access to location-specific information (such as maps, weather, or location-based services). Due to the connectivity

requirements these games often impose, they are almost exclusively available on mobile platforms.

The first location-based game to see commercial use was *Botfighters* [1], also a pioneer in the pay-per-locate business model. In this game, the player would take control of a robot with the mission of destroying other robots, controlled by other players. The players would have to wander around the streets in order to find and destroy their enemies, while being on their toes as the other players might have them on their sights. Each kill would provide the executioner with credits which could be spent at the game's website to improve the robot's capabilities further.

One of the most popular location-based games in the world, amassing more than a million players worldwide, is *Geocaching* [2]. The game itself is simple, and in some cases, it does not even require a GPS enabled device in order to be playable. Its concept is somewhat similar to that of treasure-hunting games: while some players will be stashing "caches" (a container of some kind that holds a "treasure") and logging information regarding the location of the cache, other players will be attempting to discover those caching, often having to explore the area where the cache is at and solving riddles in order to determine its exact location.

Another interesting location-based game is *Pac-Manhattan* [3] which is basically a *Pac-Man*-style game but adapted for using the players' location. The labyrinth is actually an accurate representation of a portion of Manhattan's grid of blocks, where the players will have to physically wander around. This game requires 10 players in order to work, as while five will be *Pac-Man* and the four ghosts, running around on the streets of Manhattan, the other five will each be one of the other on-the-field players' controllers, meaning that only the controllers will have the knowledge of how the game is developing and will try to guide their respective field player (be it ghost or *Pac-Man*) through the maze in order to escape the ghosts and catch pellets or catch *Pac-Man*.

With the success of location-based games, a few mobile location-based game platforms have appeared. An example of worldwide success is *Groundspeak* [4], a platform that is known for owning the *Geocaching* game, but also for the *Whereigo* [5] location-based game creating tool, a tool that allows for the creation of distinct GPS-enabled games and their distribution among the community. Another example, although not yet on par with *Groundspeak*, is *Locomatrix* [6], a platform currently offering only two games: *Fruit Farmer*, a game where the player must collect fruits scatter around a certain area by moving around in that area, and also avoiding that other players steal those same fruits, and *Treasure Hunter*, a treasure hunting game, somewhat similar to *Geocaching*, where the player has to use pictures to solve riddles and progress.

As location-based games become more widespread, these communities grow bigger, and newer location-based games are added to these platforms.

3. Location-Based Games' Issues

It is common for both games and software to present their users with some issues. In the case of games, these are mostly

related to the game's rules or playability that incur in some kind of unwanted limitation or become unbalance for its players.

So, it comes as no surprise that location-based games suffer, notably, from issues common in both these areas (game design and software development) but also due to hardware limitations (such as GPS signal unavailability, the device's usual low processing capabilities, and small storage capacity, small screen size) and as these games rely on location-based data, such as maps or weather, that information may be incomplete or unavailable even if the player's location is determinable. Additionally, since it often relies on the physical prowess of its players, due to the player's fitness and pace.

The game's design-related issues usually arise due to the game's mechanics not being correctly adapted to the location of the player, meaning that even though the game is exploring the player's location as expected, it provides a poor experience. For instance, imagine a turf-war type of game. In it, the players should expand their territory by physically invading other players' turfs or areas. Being a multiplayer location-based game, the game, although using the player's location in order to play the game and provide a very unique experience, relies upon it too much, as there will probably be players somewhere around the globe that will be playing the game practically alone, as they will not have any other turfs besides theirs to conquer nearby. Also, the game design must take into account that the player is still bound by physical (and legal) limitations, and so, a location-based racing game, where the player would have to race against the clock or against other opponents, will probably put its players' and bystanders' health on the line.

In location-based games, hardware limitations are also to be considered, as they will make the difference between a playable game and an unplayable one. Since most location-based games rely on GPS signal in order to determine the player's current position, this means that most location-based games become unplayable in areas or situations where this signal is unavailable. Furthermore, since these games are meant to be played in handheld devices, their other hardware limitations such as low amount of available RAM, video memory, or slow CPUs usually mean that even a simple game will probably somewhat strain the device and seriously affect the device's battery charge.

And in the event that the player's location is determinable, if there is location-related data to be accessed, such as maps, weather or other information, it is sometimes unavailable for the player's region or does not contain enough game-relevant data (for an instance, imagine a location-based game where the player is a paramedic and must physically move around the area in order to find virtual casualties and take them to the nearby real hospital. If the game relies on real hospital locations in order to position them on the game, the game will most likely become unplayable in some areas, where the number of nearby hospitals is slim). Also, accessing this information means that it either is stored locally, meaning that it will eventually be outdated or limited to only some locations, as these mobile devices' storage capacity is still somewhat limited, or that

this data is accessed remotely via data connection (be it HSDPA or Wi-Fi), meaning that the player will have to pay for accessing it (if he lacks a data plan).

Furthermore, location-based games often neglect the fact that they are to be played actively, and thus the physical condition of their players plays an important role in the game. As it was already mentioned, the same game (considering a game that requires the player to physically move around with haste), even if played in the same location and using the same settings by two players will be considered as being an easier game by a fit player, whereas it will be considered as being harder by a lazier one. Also, most location-based do not take into consideration the player's stamina into account. As a player further plays a location-based game, pace will eventually slow down. If the game does not take that into account, it may indirectly force the players to strain themselves since the game has become suddenly harder to play.

Finally, an issue that is also deterrent for some people is the usage of the player's location-related data. As it is known [7], privacy issues do matter, especially when related to location-based games as these games are likely to be played by children. Applications that allow the sharing of the user's private information are expected to handle that information with care and explicitly inform the user how it is being used and for what end.

These issues can be found in most location-based games and are what often moves players away from them and should be taken into account during the development and design phase of the game. A good analysis of what are the issues of most location-based games, particularly in terms of marketing and availability, from a commercial point of view, can be found here [8] and serves as a complement of some of the issues here listed.

4. Solutions for Location-Based Games' Issues

The aforementioned issues can be summarized as follows:

- (i) game-design issues,
- (ii) hardware limitations,
- (iii) location-related information availability and suitability,
- (iv) player's fitness and pace,
- (v) player's data protection.

There are several possible solutions for these issues.

4.1. Game Design. Regarding game design issues, it is important to keep in mind that location-based games involve player exposure to physical interaction with the real world. Meaning that it is important to consider where and how the player will attempt to play the game and try to keep the player safe. Being a location-based game, it always involves some degree of unpredictability. However, that unpredictability can be reduced if the player's behaviour is limited thanks to the gameplay. For an instance, setting a racing location-based game that involves the player to go from A to B may

be tempting for some players to attempt to break the record using a motorized vehicle and thus risking their safety and that of others. However, if a restraint is set, by determining another objective like, considering the previous example, stating that the player's top speed must not exceed 30 Km/h, the risk of a player speeding their way through the game is reduced. Additionally, and in order to ensure that the game is played on foot, using the device's sensors such as the accelerometer to work as a pedometer may help. These solutions do not alter the game altogether but help ensure that the gaming experience remains constant and that the game's unpredictability is lowered. Since game design is such a vast area, it is difficult to analyse location-based game design issues without going through case by case analysis. Even so, the important thing is to keep in mind what possible issues may arise from the conceptualized gameplay.

4.2. Hardware Limitations. Hardware limitations are difficult to overcome. In regular mobile games, the developer might limit the user's actions or the game's requirements. However, in the case of location-based games, since other hardware is often used, it is mandatory to have them taken into account. Since these games often rely on GPS or data connections in order to be playable, it is often the case where the player, unable to have one or neither of these services available, is incapable of playing the game. However, if the game's mechanics allowed playing the game without using GPS, but still using some location as an input (such as from logs or via direct input by the user), this situation could be avoided, sacrificing gameplay experience. On the other hand, the game could be played indoors or without any preparation whatsoever, which would be sure to please the most casual of gamers. Regarding data connections, more specifically via GPRS/HSDPA, since these are usually paid-for connection and have a limited coverage, the most adventurous of the players will probably end up in a situation where the access to this service is precarious or too expensive. In that circumstance, allowing the player to still play the game, either using cached location-related data to be used or, in case that location-specific data is lacking, using other location data. This solution enables the game to still be a location-based game, however, not based on the real position of the player. Such approach would allow the player to overcome this issue and still play a location-based game. In the case of the game using location data that is not specific for the location of where the game is being played, the game can still use the GPS signal to move the player's avatar in the game (if applicable). Obviously, although mapping the player's movement in the game, the game itself will not be considering the player's true surroundings, and it should emulate the input. If the game requires some data to be transferred, such as scores, statistics or saved games, this data transfer can often be postponed with no harm to the game. Of course, there are games [9] that accept the absence of a valid GPS signal as a hardware limitation and attempt to make the game playable only on predefined locations so as to avoid negative experiences. However, this severely narrows down the number of players able to play the game at all.

4.3. Location-Related Information Availability and Suitability. Since location-based games often use maps, weather information, or any other kind of location-related information in order to make the game truly unique and location-based, it is not uncommon to find location-based games that are rendered unplayable in many parts of the globe (due to requiring information that is either not available or that is not relevant and cannot be used in the game). Creating such a game, like Pac-Manhattan [3], means that the game is unplayable outside of the area it was designed to be played, and thus having a narrowed-down target audience. There are three possible solutions for this issue: one being not using location-related information, generating the needed information randomly, by using the player's GPS position as an input (so that even though the location's true novelty is lost, a new virtual novelty is created for that location), guaranteeing that the game is playable worldwide. Alternatively, and since mobile devices storage capabilities are now into the tens of gigabytes, it is possible to store data for all/several of the needed locations. However, and while this may ensure that the novelty of the location is used (even if still not available for some people), that information is destined to become out-dated, particularly if that information is ephemeral, such as that of weather or traffic. Ultimately, the usage of remotely stored location-related content that is often updated via web services will guarantee that the needed information is up to date and available to everyone without sacrificing the device's limited storage. However, this implies the usage of data connections, limited location coverage (as the information may not still be available for every playable location), and possibly remotely inaccessible servers. Often, even if the location-related content is available, it may still not be of any use if the game's mechanics does not take it into account (using the previous example of the location based Hospital game, if the available location-data only contains information regarding police stations, it is not usable since the game's logic does not take it into account). Some games gather the location-based information retrieved by the player when playing the game for purposes other than the game itself [10].

4.4. Player's Fitness and Pace. As it happens with most location-based games, the player's movement around the real world is used as an input, often to move their avatar around the virtual world. And like some games, such as Fruit Farmer [6], the player's speed and stamina is determinant for the outcome of the game. Unfortunately, this means that many of such location-based games are too difficult or altogether unplayable by the unfit player, providing an unbalanced gaming experience. If, however, the game was able to take the players pace into account, balancing its difficulty in real-time, the player would feel that the game was customized, and was still challenging without being too easy or overwhelmingly difficult. In the case of the "Zombies, Run" [11] game, (a game where the player must go from point A to point B in a real map by physically moving from the real point A to real point B while avoiding the hordes of Zombies that will be pursuing) if it automatically adjusted the number of zombies and their speed to the pace of the player (taking into

account the distance travelled, the average speed, and the current speed), the game would provide a custom-tailored experience. Of course, the game would still value the fastest of the players via metagaming (such as scores or boards), but the game would be playable for everyone. Something often overlooked by location-based games is that the player will probably need to stop to catch their breath. If possible (with the exception of real-time multiplayer location-based games), the game should be paused automatically whenever the player's stopped, or, if not pausing, slowing the game down significantly, giving the player the chance to gather their strength, as opposed to losing the game due to pausing for a few seconds. Other games [12] will require the player to keep a steady speed and follow navigation instructions (such as "turn left") within a narrow window of a few seconds in order to be played correctly. However, these games are meant to be played at high paces, and their players expect it to be so.

4.4.1. Player's Data Protection. When playing most location-based games, the user's location information, be it coordinates or fully readable addresses, is often cached either for future use or for statistical purposes [7]. Be as it may, the people's data privacy is an increasingly more important issue. The suggested approach is the storage of this information either locally or remotely, but preferably remotely, as this provides a safer place for the player to store personal data, particularly considering how easy it is to lose a mobile phone or someone unauthorized accessing its contained information. However, even if opting to cache information locally, the user should be prompted to enter a username and password (which will be checked out against the user login information present in the remote server) prior to the game granting access to this data or, in an effort to avoid the usage of data connections of any sort, minimize the access to the player's information when offline (not allowing to check, for an instance, the user's favourite locations for playing that game, if applicable). As always, the information stored varies from game to game. Still these games should guarantee the safety of the player's geoinformation (let alone the usual information needed when playing multiplayer online games). If not, at least alert the player about the risks, so that they are not caught by surprise by these issues. Additionally, when using location information for statistical purposes, such as checking for trends, or validating game models, the player should be informed about what data is being gathered and the purpose of said anonymous information.

5. Solution's Architecture

A framework for the creation of mobile location-based games that would incorporate some of these solutions was designed. The goal of this framework was to simplify the development of location-based games while providing the mechanisms the developers may opt to use to solve the above issues.

The created framework incorporates several remote location-based services in order to provide a near global coverage of the usually needed location-related data. So, currently, the solution offers the developer easy access to

FIGURE 1: A map and a visual representation of the mined data.

weather, maps, and georeferencing (the ability to convert a pair of latitude-longitude coordinates into a country/city/street name as needed), while offering also the tools for the developer to easily add custom web services. Furthermore, the framework itself manages to provide different solutions for the previously identified issues.

- (i) If no GPS signal is available it is possible to access a local cached map or get a new map via data connections provided an address. This solution has proved to be an acceptable one particularly from the standpoint of the player that inputs the address [13]. Additionally, since no GPS signal is available, the solution also allows for the player to move their avatar via another input (such as accelerometers or touch screen).
- (ii) If no data connection is available, it is possible to access cached maps while using the GPS to move the player's avatar around, even though the map that is being used will most likely contain inappropriate data, as the map will not contain data relevant for the player's location.
- (iii) By comparing the current position with the last one, it is possible to determine the current pace of the player, and so (if the developer decides to implement that) the game's difficulty may be changed accordingly.
- (iv) Information from the map's images may be obtained via the framework thanks to a binarization algorithm that allows for the distinction of streets, buildings, water, and green areas from each other. This information may be after that used in the game itself, as can be seen in Figure 1.

Additionally, the framework creates a layer above the DirectX mobile framework, allowing the developer to easily create, add, remove, and animate sprites, as well as provide simple collision handling mechanisms.

6. Geo Wars—A Location-Based Game

In order to both test and prove the framework's capabilities, a location-based game was created on top of it.

The game, Geo Wars [14], is a free-to-play location-based tower-defence game. In it, the player takes the role of a general with the goal of defending their sector (a portion of a map) from enemy forces, depicted in Figure 2.

The player must resist several waves of enemies, either by physically moving around, evading enemy fire, or luring them into friendly crossfire situations, or by building defensive towers, each with its unique strengths and weaknesses. The enemy attacks by land, air, and sea, using tanks, soldiers, airplanes, and cruisers. While tanks and soldiers can only move around streets or parks, aircrafts can move anywhere in the map. Each unit has its own firepower, primary and secondary objectives (some units may prefer targeting specific towers rather than the general), and A.I. Player uses money to build towers, money that can be gained over time, by destroying enemy units, or by physically moving to the locations of virtual bags of money displayed on screen. The game loads saved games settings via a web service that accesses a remote database that contains player-related saved games created using the online portal. This serves the purpose of guaranteeing the user's data privacy and safety. The user can easily load their saved games/submit high scores for a played game provided that they insert their login credentials. As far as location-based content, the game uses Google's georeferencing webservice to determine the current city the player is at (in order to download weather data for it), Google map's static API to access the map of the game, and webservicex.net Weather web service to determine the weather for any given city, as shown in Figure 3.

The game also takes into account weather (increasing the enemies speed if the weather is sunny and slowing it if it's cloudy or rainy) and allows the game to be played indoors by specifying an address for the game to be played or a cached map and by disabling the general's movement (which can make the game a bit harder to complete).

The testing of this game has been made with both a small group of testers and analysing their game experience and their feedback on the field and through a thread in the xda forum that allowed for a broader group of users to test the game and provide some comments and suggestions. This thread can be checked at: <http://forum.xda-developers.com/showthread.php?t=694276>. It was viewed over 8000 times, and the Geo Wars game was downloaded and tested by over 1000 people. The video depicting the game is also available in that thread.

However, since the game does not incorporate any feature that provides us with user statistic, the only feedback gathered was through comments or private messages through xda developers. Most notably was the desire of users to participate in multiplayer games (a feature current lacking

FIGURE 2: Geo Wars look and feel.

FIGURE 3: Geo Wars physical architecture.

in the framework), to cache games for later use and to use a custom address for the game to be played at. Fortunately, these features were easily achieved, as they were already contemplated in the framework. Since the game relies upon Google maps static API, the game is thoroughly available in most locations, a feature that was highly appreciated by the users of the forum. Remarkably, no tester seemed to be preoccupied with the data connection that was often required to play a noncached game or to include the local weather within the game. An issue that was easily identified was the need to zoom in or zoom out of the map, as in street crowded cities, the game, if not played with a high map zoom option, would be hardly playable. Likewise, if the game was to be played in a rural area, the player would need to zoom out a bit, in order to include more streets for making the game more challenging. This can be considered as the drawback of playing a game that uses such wide coverage of maps and playable locations. In fact, some users reported that in some locations, the game was either too hard or

too easy, demonstrating that the game needed a balancing mechanism. Thus, the implementation of the weather feature (as with bad weather the game becomes easier) and the pause option/slow pace down feature. This allowed for the game to be more playable and the experience more enjoyable. Other users incited us to make an option for the game to be static, without moving the player's avatar at all. This made the game to be playable while travelling by train or without worrying about losing/not having GPS signal. This feature was one that gathered the most consensus.

Some of these suggestions and issues were noted only during the testing phase of the Geo Wars game and were yet to be present in the framework. However, whenever possible they were implemented in the framework and used in the game, allowing future games to also profit from these solutions. Also noteworthy is the fact that not all location-based games may benefit from these solutions or even have these issues. On the other hand, this game itself may also not have made noticeable other problems or solutions that may be used by other location-based games. However, and considering the given feedback, it is possible to assert that all of the found issues were solved using realistic solutions, and pleasing the location-based games players' community.

7. Conclusions and Future Work

Due to the unpredictable nature of location-based games, some issues and limitations are difficult to be overcome completely. Instead, most solutions are based on alternatives that allow a game to still be playable rather than a solution to the issue itself (such as when the GPS signal is unavailable but the game is still playable by simulating GPS input some other way). However, such difficulty appeared only regarding hardware limitations. Regarding game balancing issues, the solution of allowing the game's difficulty to be constantly balanced by measuring the player's performance is doable and is applicable to these games. The developed framework incorporated most of these solutions, and thanks to the proof-of-concept game Geo Wars, it was proved to be a viable

solution considering the provided positive feedback of its users. Geo Wars was tested thoroughly by the community at xda-developers.com, with feedback being provided in the respective forum. The game's solutions and issues were validated and identified thanks to this community.

Since the framework proved to be a good solution, extra effort is being put in porting it to 3D (with mesh loaders and renderers already implemented) as well as including a multiplayer module (that it currently does not have) as these features were considered by many to be important in such games. Afterwards, Geo Wars itself will also be ported to 3D.

References

- [1] S. Dodson, "(Ready, aim, text), The Guardian," 2002, <http://www.guardian.co.uk/technology/2002/aug/15/electronicgoods.games>.
- [2] GroundSpeak, "Geocaching," 2010, <http://www.geocaching.com>.
- [3] Pac Manhattan, October 2009, <http://www.pacmanhattan.com/about.php>.
- [4] Groundspeak, 2010, <http://www.groundspeak.com>.
- [5] GroundSpeak, "WhereIGo," 2008, <http://www.wherigo.com>.
- [6] Locomatrix, 2010, <http://www.locomatrix.com/about.html>.
- [7] K. Eaton, "Rumor: Facebook's Location-Based Game Effort Due For Unveil Wednesday," 2010, <http://www.fastcompany.com/1682725/facebook-location-foursquare-check-in-social-networking-4/1/2011>.
- [8] S. Drab, J. Kroesche, C. Grün, and A. Jakl, *Pushing Location Based Games Further—How To Gain End User Suitability*, FH Joanneum, Graz, Austria, 2007.
- [9] S. Benford, R. Anastasi, M. Flintham et al., "Coping with uncertainty in a location-based game," *IEEE Distributed Systems Online*, vol. 2, no. 3, pp. 34–41, 2003.
- [10] S. Matyas, C. Matyas, C. Schlieder, P. Kiefer, H. Mitarai, and M. Kamata, "Designing location-based mobile games with a purpose—collecting geospatial data with city explorer," in *Proceedings of the International Conference on Advances in Computer Entertainment Technology (ACE '08)*, pp. 244–247, December 2008.
- [11] E. Fikkert, "Zombies, Run," September 2010, <http://www.androidapps.com/t/zombies-run>.
- [12] J. Hielscher and J. Heitlager, "Wanderer—location independent GPS-game," in *Proceedings of the 3rd International Workshop on Pervasive Gaming Applications (PerGames '06)*, May 2006.
- [13] S. Benford, W. Seager, M. Flintham et al., "The error of our ways: the experience of self-reported position in a location-based game," in *Proceedings of the 6th International Conference Ubiquitous Computing (UbiComp '04)*, 2004.
- [14] A. Coelho and J. Jacob, "Geo Wars—the development of a location-based game," in *Proceedings of the VideoJogos*, 2010.