Master thesis

Creating Innovative Pedagogical Opportunities via “Treasure-HIT”

Mobile Treasure Hunt Games Generator

Author: Shavit Cohen
Supervisors:
Prof. Miky Ronen,
Dan Kohen-Vacs
Examiner: Dr. Marc Jansen
Date 30th January 2015
Subject: Social Media and Web Technologies
Level: Master
Course code: 5ME01
Abstract
In this thesis we are presenting a design-based research aimed to extend an existing mobile-based environment named “Treasure-HIT”. “Treasure-HIT” is an environment which uses the available mobile technology to enable teachers to conduct pedagogical “Treasure Hunt” games for their students. At baseline (January 2013), “Treasure-HIT” was at its early stages and has not yet been officially released; only a few teachers were granted access to it. The goal of this thesis was to test the system with teachers in order to understand the actual needs to detect difficulties and usability limitations and to improve the system accordingly. The study was conducted as a design-based research that included three iterations. Within each iteration we have identified needs expressed by teachers and improved the relevant system features. The outcome of this thesis was the design, development and implementation of 7 different features that increased the usability of the “Treasure-HIT” environment and its potential to effectively support the creation and enactment of pedagogical location based activities.

Acknowledgements
I would like to acknowledge the support provided by my wife and family during this study. I would have not even reached halfway without the ongoing support, encouragement, and patience from my wife Keren. My supervisors, Professor Miky Ronen and Dan Kohen-Vacs have provided essential guidance and advice along the way. And my thanks to Holon Institute of Technology for provided me with the foundations from which I could grow. Special thanks to my friend and colleague Robert Ferentz for his help and patience with proofreading my work.
Contents
List of Figures ........................................................................................................ iii
List of Tables ......................................................................................................... iv
1 Introduction ....................................................................................................... 1
  1.1 Motivation ........................................................................................................ 1
2 Aims and objectives .......................................................................................... 3
  2.1 Aims .................................................................................................................. 3
  2.2 Objectives ......................................................................................................... 3
3 Overview ........................................................................................................... 4
  3.1 Used technologies ........................................................................................... 4
4 Literature review ............................................................................................... 5
  4.1 Mobile learning ............................................................................................... 5
  4.2 Seamless learning ........................................................................................... 6
  4.3 Playing and mobile educational gaming ......................................................... 6
     4.3.1 Playing as a learning opportunity .............................................................. 6
     4.3.2 Mobile educational gaming .................................................................... 7
  4.4 Location-based services .................................................................................. 7
  4.5 “Treasure Hunt” .............................................................................................. 8
     4.5.1 Coupling “Treasure Hunt” with mobile technologies ................................ 8
     4.5.2 Review of existing mobile “Treasure Hunt” like environments ............... 8
     4.5.3 Summary and comparison ..................................................................... 10
5 Introduction to “Treasure-HIT” ..................................................................... 11
  5.1 Game description ............................................................................................ 11
  5.2 Rules and game flow ...................................................................................... 11
  5.3 Game environments ....................................................................................... 11
     5.3.1 The Mobile application ........................................................................... 12
     5.3.2 Author environment ............................................................................... 15
  5.4 “Treasure-HIT” technical aspects ................................................................ 21
     5.4.1 General architecture ............................................................................. 21
     5.4.2 Understanding the Application layer ...................................................... 25
6 Research questions and hypothesis ................................................................. 29
7 Methodology .................................................................................................... 30
  7.1 Design based research .................................................................................. 30
8 Data collection, analysis and results ................................................................. 31
  8.1 Iteration #1 – Interviews with first adopters ................................................. 32
     8.1.1 Designing and implementing the solutions ............................................ 32
     8.1.2 Results of iteration #1 ............................................................................ 37
  8.2 Iteration #2 – Periodic teacher training at “Beit Berl” 38 ......................... 40
     8.2.1 Designing and implementing solutions .................................................. 40
     8.2.2 Results of iteration #2 ............................................................................ 42
  8.3 Iteration #3 – TEL assimilators of ”Tel Aviv” .............................................. 44
     8.3.1 Designing and implementing solutions .................................................. 44
     8.3.2 Results of iteration #3 ............................................................................ 48
9 Summary .......................................................................................................... 51
10 Limitations ...................................................................................................... 52
11 Further Work .................................................................................................. 53
12 References ...................................................................................................... 54
13 Appendix ........................................................................................................ 58
  13.1 Elaboration about the tools and technologies in use by the author environment ........................................................................... 58
     13.1.1 HTML5 .................................................................................................... 58
     13.1.2 JavaScript and jQuery libraries ............................................................ 58
     13.1.3 AJAX (Asynchronous JavaScript and XML) ......................................... 58
     13.1.4 Third party Google’s APIs .................................................................... 59
  13.2 Elaboration of tools and technologies being used by the mobile app ....... 59
     13.2.1 “PhoneGap” ........................................................................................... 59
     13.2.2 Mobile App JavaScript framework ......................................................... 59
List of Figures

Figure 2.1 Schema of the set of objectives .......................................................... 3
Figure 5.1 Schema displays the game flow ......................................................... 11
Figure 5.2 The home screen as it seen on Android devices, the player has a choice how to access the game ................................................................. 12
Figure 5.3 Login a game via game code ............................................................... 12
Figure 5.4 Login a game by searching available games near the players' location ... 12
Figure 5.5 Registration of a new team screen ................................................... 12
Figure 5.6 Welcome message ............................................................................. 13
Figure 5.7 Available clues .................................................................................. 13
Figure 5.8 The content of Text typed clue ......................................................... 13
Figure 5.9 The content of an Image typed clue .................................................. 13
Figure 5.10 A station location check by the GPS ............................................... 13
Figure 5.11 A negative feedback for not finding the station .............................. 13
Figure 5.12 A positive feedback when the player reached the station .............. 13
Figure 5.13 A QR-Code scanner allowing the player to verify indoor stations ..... 14
Figure 5.14 A single selection question task .................................................... 14
Figure 5.15 An open question task ................................................................... 14
Figure 5.16 The delay mechanism ..................................................................... 14
Figure 5.17 The treasure .................................................................................... 15
Figure 5.18 The author portal ............................................................................ 15
Figure 5.19 A Game settings screen ................................................................. 16
Figure 5.20 A station settings screen ................................................................. 18
Figure 5.21 Pinpointing a station with Google Street view API .......................... 19
Figure 5.22 “Treasure-HIT” Architecture illustration ........................................ 21
Figure 5.23 System architecture – layers scheme ............................................. 22
Figure 5.24 UML class diagram of "Treasure-HIT" ........................................... 24
Figure 5.25 ASP.NET MVC request flow ........................................................... 26
Figure 5.26 “Treasure-HIT” Mobile application architecture ............................ 27
Figure 8.1 The flow of the design based research cycles ................................. 31
Figure 8.2 QR-Code structure ........................................................................ 33
Figure 8.3 A state UML diagram showing the teacher flow for creating a station.... 34
Figure 8.4 A screenshot of the station's settings section at the author environment ... 34
Figure 8.5 A screenshot of the printable barcode page after clicking the "Print barcode" button ......................................................................................... 34
Figure 8.6 A state UML diagram showing the flow at the player side when arriving a station .......................................................................................... 35
Figure 8.7 Clues screen for an indoor station .................................................... 36
Figure 8.8 The QR-code scanner inside the app in action ................................. 36
Figure 8.9 A negative feedback shows the deviation to the station .................... 36
Figure 8.10 The algorithm to calculate air-line distance between two points ......... 37
Figure 8.11 Pie chart shows the percentage of games which indoor stations are defined as part of the game ................................................................. 38
Figure 8.12 The teachers are trying to solve a riddle using the campus map ......... 39
Figure 8.13 A team has reached the correct station .......................................... 39
Figure 8.14 A team is trying to answer an open question task, notice that all team members are participating, by searching the answer on the web, using their smartphones ...................................................... 39
Figure 8.15 The settings screen ...................................................................... 40
Figure 8.16 The teacher login page on the app ............................................... 40
Figure 8.17 The teacher in-app portal ................................................................. 40
Figure 8.18 The stations of a game are nested and open as an accordion .......... 40
Figure 8.19 The functionality of a station is nested inside it .............................. 41
Figure 8.20 A warning confirmation popup being shown after the teacher taps "sample" ........................................................................................................ 41
Figure 8.21 The device GPS is trying to improve accuracy to at least 15 meter .... 41
Figure 8.22 A success notification telling the teacher that the location has been changed ...................................................................................................... 41
Figure 8.23 Example of a delay for answering a wrong answer .......................... 42
Figure 8.24 Introduction lecture to "Treasure-HIT" ........................................... 43
Figure 8.25 A team of two assimilators trying to solve a clue ............................ 43
Figure 8.26 A team of 3 assimilators trying to solve a task ............................... 44
Figure 8.27 The flow that requires constant internet connection ................. 45
Figure 8.28 The flow that requires internet connection only upon registration 45
Figure 8.29 Table of results .......................................................................... 46
Figure 8.30 JSON format of a Group object which completed a game .......... 47
Figure 8.31 Dialog for entering the colleague email in order to send them a game 48
Figure 8.32 Dialog for summary and confirmation for sending a game .......... 48
Figure 8.33 A map of the game that played in kibbutz "Nir Yitzhak" at April 2014 showing stations spread all over the kibbutz, event where there is no WiFi coverage ................................................................................................. 49
Figure 8.34 A summary session, the headmaster presenting the score table to students .................................................................................................. 49
Figure 9.1 “Treasure-HIT" registered editor’s graph until April 2014 .............. 52

List of Tables
Table 4.1 Comparison between the reviewed environments ............................. 10
Table 8.1 Requirements raised by teachers ..................................................... 31
Table 13.1 Features comarison of Backbone Ember and AngularJS ............. 60
1 Introduction

Formal and informal learning opportunities are presented as learning experiences across different planes and environments, which correspond to different pedagogical strategies like competition, peer learning and collaborative learning. Available technology may aid in order to conduct pedagogical processes across formal and informal learning environments (Pea, 2009). Many academic projects have successfully used mobile technologies to link virtual and physical spaces such as "Can You See Me Now?" (Benford, Crabtree, Flintham, Drozd, & Paxton, 2006), which demonstrated how the technology can bridge real and virtual worlds and allow participants to experience them both at the same time. Some educational researchers refer to these ‘bridging technologies’ as a way of enhancing teaching and learning activities, for example MyArtSpace (Vavoula, Sharples, Rudman, Meek, & Lonsdele, 2009) shows how relatively simple mobile technologies can effectively support innovative learning practices that can take learning outside the classroom. The use of mobile technology to facilitate enquiry-based activities allows students to be engaged in active enquiry, using realistic tools in authentic environments and become active players in the learning environment using their personal mobile accessory. This ability provides learners with opportunities to manage their learning process over time, to engage in collaboration, and to relate information to situated problems (Chen, Seow, So, Toh, & Looi, 2010). According to Huizenga, Admiraal, Akkerman, & Dam, (2009), games are one particular type of activity that has been used to engage learners and focus their attention. It was shown that game-like experiences have the power to motivate learners to take part in a learning activity and may provide a suitable platform for many activities in the field. "Treasure Hunt" is one of the approaches that may combine the outdoor/indoor space for learning and at the same time scale up the learners’ motivation and enthusiasm. "Treasure Hunt" activities for example represent a classic and playful approach that was first practiced with no technology (Baden-Powell, 1908). During the game, students are faced with "riddles" leading them to different landmarks (stations). "Treasure Hunt" games could be played according to different social settings: individually, in groups or as a team where the game can have an exploratory nature, competition among individuals, competition between teams or a combination of all of the above. “Treasure Hunt” may offer rich pedagogical potential including action learning, a large degree of exploration (Inquiry Learning) and collaborative learning (Kim & Yao, 2010). Mobile technology can foster “Treasure Hunt” games by utilizing the device’s special settings, such as GPS, large screen, camera and capture utilities.

1.1 Motivation

The department of instructional technologies at the Holon Institute of Technology developed an educational tool named "Treasure-HIT". This tool aimed to enable teachers to create and enact pedagogical “Treasure Hunt” games for their students. "Treasure-HIT" has been developed in order to meet the following requirements:

- Exploit the pedagogical nature of the traditional “Treasure Hunt” game to encourage students to explore their environment while playing the game.
- Expands teachers’ tutoring tools with an enabler for conducting “Treasure Hunt” games, and encourage them to create learning opportunities for their students outside and beyond the boundaries of the traditional classroom.
"Treasure-HIT" was introduced by Kohen-Vacs, Ronen & Cohen, 2012 and tested among a few teachers from elementary schools. Whenever the environment was presented to teachers the responses were very positive and enthusiastic. However, when teachers tried to create and enact meaningful pedagogical activities they seemed to encounter some challenges. This research aimed to improve the "Treasure-HIT" environment and enable it to become a useable teaching tool for schools and institutions worldwide.
2 Aims and objectives

2.1 Aims

From the discussion above we fleshed out the following aims:

- Understanding the challenges teachers experience while trying to use the "Treasure-HIT" environment for creating and conducting educational games.
- Coping with these challenges by adding new and revised features to the environment.
- Designing, developing and implementing these features.
- Evaluating the revised "Treasure-HIT" environment.

2.2 Objectives

Once the aims were established, we proceeded with the specification of a set of objectives that we considered necessary to be followed in order to be able to conduct the research:

1. Conduct interviews with teachers who used the "Treasure-HIT" environment in order to identify the problems and challenges they encounter while creating activities for students.
2. Perform analysis of responses and determine necessary improvements and additions as solutions to the problems.
3. Design, develop and embed the solutions.
4. Test the solutions with teachers that will create and enact activities with students and gain insights regarding further improvements needed.
5. Repeat phases 2-4.
6. Summarize the study.

![Figure 2.1 Schema of the set of objectives](image)
3 Overview

This thesis is divided into 8 parts:

- Literature review;
- Introduction to “Treasure-HIT”;
- Research questions and hypothesis;
- Methodology;
- Data collection analysis and results;
- Summary;
- Limitations;
- Future work.

A set of 5 relevant topics are reviewed in the chapter “Literature review”, each topic discusses a unique perspective that we considered for our research. We start by discussing “mobile learning”, as a concept and as a learning pattern. Next subcategory discusses the topic of “seamless learning” as a design guideline to the “Treasure-HIT” approach. Then we proceed with discussing “mobile educational gaming” as an approach for establishing learning. Then we discuss how location based services are being used today to support this approach. The last topic in our literature review covers “Treasure Hunt”, this topic is divided into two parts, the first one discusses the traditional game which has been played since 1908, and the second part is a review of 7 existing mobile platforms that use “Treasure Hunt” gaming mechanism to support learning. Lastly, we summarize and compare these environments.

3.1 Used technologies

In order to strengthen the context of the technological work that have been made as a result of this thesis, we would like to emphasize that at the baseline, “Treasure-HIT” infrastructure relied on the following technologies and tools: C#.NET, MVC framework, MSSQL, HTML5, Angular.JS, jQuery, CSS3 and Phonegap. All the modifications and additions that were made during this thesis, kept relying the existing infrastructure which haven’t changed. The enhancement that have been made during this work did not required modifying the infrastructure, but to add / modify the code of the different modules in the system.
4 Literature review

4.1 Mobile learning

Mobile learning is a concept that refers to the use of mobile devices for enrichment of the pedagogical process and learners’ experience. Mobile devices can make learning portable, personal, and spontaneous. A number of high level definitions have been made, for example: O'Malley, et al. (2005) describes mobile learning as any sort of learning which occurs when the learner is not at a fixed place, or when the learner takes advantage of learning opportunities offered by mobile technologies. Scanlon, Jones, & Waycott (2005) argued that the important feature of mobile learning is that the learner is on the move rather than just using the device. Additionally, Sharples, Taylor, & Vavoula, (2005) emphasized this approach in considering the main factor which creates effective mobile learning activities. The mobility enhances the learning process in the sense that learners may engage with educational activities without the constraints of classroom boundaries (Kukulska-Hulme & Traxler, 2007). Mobile learning research refers to the study of patterns in which mobility of learners are augmented by personal and public technology. This mobility can contribute to the process of gaining new knowledge, skills and experience, and moreover, allows meeting the needs of lifelong learning (Georgiev, Georgieva, & Smrikarov, 2004).

When learners are on the move, within, and between multiple contexts such as physical location, time, topic, and levels of engagement, they may benefit on an individual level and gain knowledge through conversations and explorations across multiple contexts amongst people and interactive technologies (Sharples, Lonsdale, Meek, Rudman, & Vavoula, 2007). While considering mobility from the learner’s point of view, it can be argued that mobile learning happens everywhere, for example: pupils studying to exams on the bus to school, brokers assessing the stock market trends, academia lecturers refreshing their professional knowledge in between classes, language students improving their language skills while travelling abroad. All these instances of formal or informal learning are taking place when people are not in a fixed location, meaning that with the use of mobile devices, learners can learn anywhere at any time (Crescente & Lee, 2011). Over the past decade mobile learning has grown from a minor research interest to a set of significant projects in schools, workplaces, museums, cities and rural areas around the world. The mobile learning community is still fragmented, with different national perspectives, differences between academia and industry, and between the school, higher education and lifelong learning sectors (Singh M., 2010). Sharples, Arnedillo-Sanchez, Milrad, & Vavoula, (2009) argued that the key factor of mobile learning is mobility which has several references:

- **The physical plain**: when people are mobile they try to conduct learning within the gaps of daily life. The location may be relevant to the learning, or take part as a reference.
- **The technological plain**: mobile tools are usually lightweight and can easily be carried around. The technology enables to transfer data across devices. Sharing can be done from laptop to mobile & notepad devices.
- **The conceptual plain**: learners’ attention shift between many topics and themes and from one conceptual topic to another according to the learner’s personal interest, curiosity and/or commitment.
- **The social plain**: learners are active in various social groups, such as family, colleagues from work or classroom.
Understanding these references might assist researchers to not only provide content to the learner in whatever context they may be, but also to utilize the location itself as part of the interactive learning activities. Furthermore, these mobility references meet socio-constructivist approaches to learning, which emphasize the importance of learners engaging in authentic, complex problem-solving activities in order to allow meaningful learning to take place. The widespread of portable devices and the relatively low cost of these devices (Earnshaw, 2011) contribute to seamless learning approaches (Chan, et al., 2006).

4.2 Seamless learning
The use of mobile learning may foster an approach that could be integrated in learning methods by providing a seamless experience. Adopting this approach is possible when mobile devices are commonly assimilated among students (Chan, et al., 2006). In the past, when everyone could afford learning tools like a book and a pencil, the way people learned had changed (Papert, 1980). A similar shift could happen if everyone owns a personal mobile computing device and embrace it as a commonly used learning tool. The mass adoption of mobile devices over the past years creates many new opportunities, which can contribute to the pedagogical potential of seamless learning. It enhances the learning experience by suggesting that students may learn whenever they are curious, and in a variety of easy to change scenarios (such as formal and informal learning, personal and social learning, etc.). These personal devices can be used as a technological mediator between learners and their learning environments. The use of personal devices enables the learner to be an active player within the learning situation (Chan, et al., 2006). This enables learners with new opportunities and tools for managing their learning process over time, engaging in collaboration, and relating information to situated problems (Chen, Seow, So, Toh, & Looi, 2010). One of the approaches to facilitate learning seamlessly is by mobile educational gaming (Spikol & Milrad, 2008).

4.3 Playing and mobile educational gaming
4.3.1 Playing as a learning opportunity
Teachers have always been looking for new ways to improve their students’ understanding (Veenema & Gardner, 1996), and locating various teaching methods using tools that can reach their students at multiple levels (Tan, Lewis, Avis, & Withers, 2008). Sandor & Klinker (2005) argued that when children play they establish learning due to the use of most of their senses. Play becomes an important activity which assists to improve and develop children physically, mentally, emotionally, and socially. For children, playing is a natural way to learn, solve problems and understand their environment (Rapeepisarn, Wong, Fung, & Depickere, 2006). Play can help children experience a greater form of learning than when they are in the classroom (Gee, 2003). Play can stimulate children to understand new concepts that would otherwise be difficult to reproduce (Squire, Barnett, Grant & Higginbotham, 2004). Play is an enjoyable activity that serves as a medium to develop capabilities, abilities, etc., through active participation (Blecic, Cecchini, Rizzi, & Tronfio, 2002). Play is amusing and enjoyment is important when trying to achieve learning goals because what is enjoyably learned is less likely to be forgotten (Squire & and Klopfer, 2007). Squire & and Klopfer (2007) argue that game-like experiences have the power to motivate learners to take part in a learning activity, and may provide a suitable scaffold for their activities in the subject of the
game. Educators can incorporate powerful tools into their teaching activities that can enrich and complement students' skills through play.

4.3.2 Mobile educational gaming
The combination of games and mobile platforms for achieving learning is an area that suffered from lack of best practices on how to design and provide comprehensive, effective, and appropriate support for learners (Facer, et al., 2004). Mobile devices that were once mainly communicational accessories, became a potential game platform (Flintham, et al., 2007), which allows learners to interact simultaneously with the physical world and digital information. This enables new forms of educational experiences by coupling the physical environment with digital resources. It makes sense to consider mobile learning and games-based learning as experiences that could fruitfully be combined (Facer, et al., 2004). At the same time, connectivity via wireless technology shifted the focus away from the individual player to the connected player, which enabled players find fellow gamers anywhere on the globe (Flintham, et al., 2007). Furió et al. (2013) outlined that mobile environments are an ideal platform for educational gaming, which improve lifelong learning as well as provide for more versatile educational methods. One of the most powerful properties of mobile educational gaming is the learner's interaction with his/hers surroundings, available due to location-based services.

4.4 Location-based services
Location-based services integrate a user’s geographic location with the general notion of services. Location-based service can be conceptualized as the ability to find the geographical location of a mobile device (e.g. smartphones, notepads) through the mobile network, providing services based on location and time information. The location of a mobile device is constantly retrieved by one of the following methods: the wide spread inclusion of GPS (Global Positioning System), which is based on a satellite navigation system, or GPRS (General Packet Radio Service) that is based on cellular network tracking (Brimicombe & Chao, 2009). Well known location based applications include: location based social networks (e.g., Foursquare, Google latitude); location based advertisements (e.g., Google ads, Facebook ads); navigational GIS systems (e.g., Waze, Google maps); augmented reality platforms (e.g., Layer, Junaio); social recommendation systems (e.g., Yelp, Wikiloc); photo sharing (e.g., Picasa); location based games, and educational applications. Location-based applications have the potential to generate learning environments, accessible at anytime, anywhere (Brown, et al., 2011). Location based services opens new opportunities to combine playful gaming with educational goals. Various projects have shown how mobile technologies can help take these opportunities out of the classroom and into the field (Kurti, Špikol, Milrad, Svensson, & Petterson, 2007). “Treasure Hunt” is one type of a game which facilitates inquiry-based activities that prompt students to become engaged in active exploration using realistic tools within authentic environments.

4.5 “Treasure Hunt”
“Treasure Hunt” activities represent classic and playful approach that existed long before technology was able to offer it, for example: Baden-Powell, (1908). At the game, students are faced with clues leading them between landmarks (stations). The game rules enable landmarks to be linked with specific tasks, which are required in order to advance in the game. “Treasure Hunt” games could be played according to different social settings: individually, in groups or as a team. When played as a team,
the game can have an exploratory nature, competition among individuals, competition between groups or a combination thereof (Kohen-Vacs, Ronen, & Cohen, 2012). This kind of game offers rich pedagogical potential, including action learning, degree of exploration (Inquiry Learning), and collaborative learning (Kim & Yao, 2010).

4.5.1 Coupling “Treasure Hunt” with mobile technologies
Mobile technologies enhanced by location based services can foster Treasure Hunt games by utilizing the device’s special settings, like GPS, large screen, capture and record abilities. The use of mobile technologies to enable outdoor inquiry-learning aligns well with complexities and skills required in the 21st century (Dede, Korte, Nelson, Valdez, & Ward, 2005). Today, there are already mobile learning approaches which support location-based inquiry learning and offer an outdoor learning experience. We reviewed the main mobile applications that offer Treasure Hunt like learning experience as reference that should be taken in consideration while approaching our challenges.

4.5.2 Review of existing mobile “Treasure Hunt” like environments
4.5.2.1 Lemonade
LEMONADE ("Learning Environment for Mobile Network-Able Devices") is a framework for planning and conducting field trips with mobile devices. It comes with an authoring interface that allows for defining certain trips with specific tasks for student groups and locations. LEMONADE provides a full cycle approach of a pedagogical trip activity which consists from 3 phases: the planning of a trip (pre-trip), the trip itself and the after trip (post-trip). Each phase can contains a set of activities and sub-activities. The first phase, will conduction mainly by the teacher. The second phase involves students which go into the field and conduct different activities that may involve data collection, for example taking pictures. This data will be used in the third phase. The “post-trip” is mainly for follow-up activities reporting/resenting the results, discussion and reflecting the trip. (Giemza, Bollen, Seydel, Overhagen, & Hoppe, 2010)

4.5.2.2 7scenes
"7scenes.com" is an environment relying on a community platform for multi-user real-time mobile gaming. It offers a mobile application that transforms any smartphone into a mobile storytelling platform. A playground is being set by an authoring system, which enables the placement of media on a map interface (FitzGerald, Taylor, & Craven, 2013). This enables artists, schools, museums and others to develop interactive mobile tours and games. "7scences" provides many types of games for pedagogical settings; “Treasure Hunt” is one of them. It allows users to combine routes through their physical surroundings using pictures, videos, audio, texts and assignments (Tol, 2008). Based on different game templates with specific game rules, the players can add photos, videos, audio, notes and tasks to locations with which they can create narratives. Students can subsequently create a competition by setting up teams, navigate the city, discover the storyline, collaborate with other players, perform tasks, record and upload media and score points. The entire game is broadcasted live on the web and archived so other players (and teachers) can play back all interactions and reflect on them with other players and the media that was created (7scenes.com, 2013).

4.5.2.3 Geo Caching
"Geo Caching" environment is a geo-social game. The gaming aspect of "Geo Caching" motivates the player to find virtual/physical tokens at certain spots and
collect them. A typical cache consists of a small physical waterproof box which can be hidden anywhere in the world. The latitude and longitude coordinates are noted and then published on a web site\(^1\) and can be discovered by following a map from the smart-phone application. A typical cache contains a “log book” that should be signed by players who find the cache and potentially a “treasure” depending on the size of the box. The “treasure” is usually low in value, such as small toys and coins. Usually, when a participant finds the box they should exchange treasures – if they take something from the cache they should leave something of similar value. One variant of a catch is a Multi Cache in which participants have to find several caches when each catch has clue/coordinates to the next one until the cache containing the logbook is finally found. A second type of a cache is the “Puzzle Cache”. In this cache the players solve a puzzle in order to discover the coordinates of the cache. The “Geo Caching” website contains the list of available caches all over the world where each cache has its own dedicated web page. In addition to the coordinates, the web page contains contextual information about the cache, like maps of starting points, clues and riddles to detect the coordinates, and useful tips or things to look out for (O’Hara, 2008). Players who decide to take part in “Geo Caching” can go to the website, find the closest caches in their surroundings and start playing; there is no element of competition, just a personal satisfaction.

4.5.2.4 Skattjat

Skattjat (Treasure Hunt in Swedish) is a game meant to develop informal skills of map reading and enrich learning with local history. In Skattjat the participants are divided into teams equipped with a mobile phone. Their task is to find landmarks containing clues in a form of text or audio via the game mobile phone by navigating using a mobile-based map. Skattjakt has been conceived and developed to encourage young students to become physically active by solving a mystery surrounding a castle located on the Linnaeus University campus at Sweden. The game is inspired by the ideas behind Treasure Hunt activities and the sport of orienteering - a traditional Scandinavian running sport involving navigation with a map and compass (Spikol & Milrad, 2008).

4.5.2.5 The Wandering

“The Wandering” is an environment designed to facilitate outdoor, authentic, and interactive learning via the creation of location-based interactive learning objects (LILOs). “The Wandering” was integrated as part of an environmental education program amongst middle school students. The environment aims to encourage students to “wander around”, while exploring new places and interacting with their environment and with one another. “The Wandering” allows users to create LILOs on their own or arrive to a location they wish to visit (city, street, park, museum, etc.) and see the LILOs exist at that location. Students may decide whether to see all the LILOs available at a location or only some that are certified (excellent LILOs that were tagged as ‘certified’ by experts). Students can choose to explore individual stations or search routes (an organized order of LILOs) and by doing so play some kind of a Treasure Hunt game. When a user (student or teacher) finds an interesting location he/she can: mark it on the mobile application map, create some kind of pedagogical activity about it, add layers of information to the LILO, and finally share it with other users (Barak & Ziv, 2013). “The Wandering” is not a dedicated Treasure Hunt environment but it can easily be integrated into one.

---

\(^{1}\) “Geo Caching” website address - www.geocaching.com
4.5.2.6 “fAR-Play”
“fAR-Play” (for Augmented Reality Play) is an augmented reality “Treasure Hunt” like game that supports outdoor activities (using GPS) as well as indoors. “fAR-Play” is one of the AARGs (Augmented and Alternate Reality Games) that expand traditional gaming by being played in multiple worlds in parallel, where the interaction between the players and the worlds is mediated by smart phones. Players interact with the game in the real world through clues communicated to them through mobile phones which are aware of the player’s locations using GPS or QR tags. Sometimes the virtual world may reflect the real world and sometimes augment it. "fAR-Play" requires 4 dedicated applications for AR (Augmented Reality) supports and a special AR browser named "Layer". "fAR-Play" has an author environment that may enable teachers to create games for their students. The main difficulty though is that the creation of “Treasure Hunt” activities is complicated and requires rich experience in creating AR models (Gutierrez, et al., 2011).

4.5.2.7 Frequency 1550
Frequency 1550 is a location-based game which being played in Amsterdam. It developed by the Waag Society - a Dutch IT research foundation. The context of the game is the medieval Amsterdam (year 1550) which the players have to imagine. The game is being played as a competition between teams of students during a single school day as part of the History lessons in secondary education. With the help of the Internet, smart phones and GPS technology, Amsterdam becomes a medieval playground. During the game, the students should conduct small location-based media tasks in order to explore areas, skilled the use of different maps, and master different areas in Amsterdam. (Admiraal, Huizenga, Akkerman, & Dam, 2011)

4.5.3 Summary and comparison
We reviewed 7 “Treasure Hunt” mobile environments. Some support learning and aim to serve educational goals and some might be modified but are not dedicated to that purpose. Table 4.1 shows a comparison between the reviewed environments. We composed criteria for the table that we believe emphasizes most of the differences between the reviewed environments.

<table>
<thead>
<tr>
<th>Criteria / Reviewed environment</th>
<th>Lemonade</th>
<th>7Scenes</th>
<th>GeoCaching</th>
<th>Skattjat</th>
<th>The Wandering</th>
<th>fAR-Play</th>
<th>Frequency 1550</th>
</tr>
</thead>
<tbody>
<tr>
<td>Treasure-hunt dedicated environment</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Environment dedicated to pedagogy</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Provides an author environment</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Offers conducting activities in an authentic pedagogical settings</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Provides a variety of pedagogical tools</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Free of charge for pedagogical uses</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Easy to install and use</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 4.1 Comparison between the reviewed environments

According to Table 4.1, none of the reviewed environments support all the criteria. In the next section we will present additional Treasure Hunt environments which aim to support all the criteria and to enrich teachers’ available pedagogical tools.
5 Introduction to “Treasure-HIT”
Before we present the environment, we would like to emphasize that some of the features presented in this chapter were designed, developed and implemented as a result of this study. We decided to introduce "Treasure-HIT" as it is available for use today in order to highlight its strengths and the opportunities it offers.

5.1 Game description
This subsection describes “Treasure-HIT”, an environment enabling teachers to design and enact mobile “Treasure Hunt” games.
“Treasure-HIT” was developed by the department of instructional technologies at the Holon Institute of Technology (H.I.T) in 2012 and was first introduced by Kohen-Vacs, Ronen, & Cohen, (2012) as an environment that enables teachers to create Treasure Hunt games outdoors. "Treasure-HIT" was developed to serve two main goals:

1. Provide an easy to use, multi-language environment for teachers encouraging them to extend the learning environment outside the traditional classrooms and thus intensify the students’ learning experiences.
2. Exploit the available mobile technologies students possess and practice mobile seamless learning.

5.2 Rules and game flow
"Treasure-HIT" game rules are very similar to the rules of the traditional, non-technological game (see subsection 4.5). The difference is that clues or any other communication which is supposed to lead the players across different stations on their way to the treasure are not physical materials and are presented on mobile devices. But similar to the traditional version of the game, “Treasure-HIT” still requires physical presence of the players across the different landmarks in a search for the treasure. A game might be played in teams/singles competing against each other or with no competition at all. Any game is composed of stations, and after finding the last station, the "treasure" is revealed. Figure 5.1 points out the flow of a "Treasure-HIT" game.

![Figure 5.1 Schema displays the game flow](image)

A game starts with a clue/s, which leads the player to a station; at the station task/s may require actions from the player. Only when the player completes the tasks successfully he/she will be able to proceed. This cycle might repeat itself a few times, until the player arrives to the last station – the "treasure".

5.3 Game environments
Conducting a "Treasure-HIT" game requires some preparations from the teacher’s side, and a few minor steps from the players. A dedicated "Treasure-HIT" author environment is available for teachers in which they can create and set “Treasure
Hunt” games. Only after a game is fully set, it can be played. In order to participate in a game, the players should install the “Treasure-HIT” mobile application on their smart-phone devices. At the following subsections we will present both of the "Treasure-HIT" environments:

### 5.3.1 The Mobile application

The “Treasure-HIT” mobile application enables the player to access the games created and published by a teacher. The application is supported by any device that operated by “iOS” or “Android” operating system and requires installation prior to playing. The app may be downloaded at the corresponding “app stores” of the different providers.

#### 5.3.1.1 Accessing a game

When activating the application, the home screen is being presented. It gives the players a small menu button from which they can choose to set the interface language or access the setting. There are two different ways to login to a game (Figure 5.2): 1. **By Code** - 5 digits code. Figure 5.3 illustrates when the player chose to login to a using a code. 2. **Selecting a game nearby** - relative to the player’s surroundings. Figure 5.4 illustrates the preview when the player chose to browse available games nearby within a distance of 1km. The list presented is sorted according to the diviation of the location of the game from the player’s current location. After selecting a game, the player will be asked to register a new group (Figure 5.5). This will allow presenting the performance of the player in the game at later stages. “Treasure-HIT” assumption is that there are no “solo” players, and the teacher will team students together for many pedagogical and logistic reasons.

![Figure 5.2 The home screen as it seen on Android devices, the player has a choice how to access the game](image1)

![Figure 5.3 Login a game via game code](image2)

![Figure 5.4 Login a game by searching available games near the players’ location](image3)

![Figure 5.5 Registration of a new team screen](image4)

#### 5.3.1.2 Clues

After the player registered a new group, the “welcome message” pops up (Figure 5.6). “Start playing” will lead the player to the clue/s for the first station. Figure 5.7 illustrates the available clues in forms of icons each implies the clue media-type. Tapping on a clue shows its content. Figures 5.8 and 5.9 illustrate a text typed clue and an image typed clue. Additionally, a status bar is available at the top of the screen telling the player the current progress in the game.

---

2Available only for Android devices since Apple app-store does not allow games activation via code.
5.3.1.3 Verifying location at the station

During the game, when the players believe they have solved the clues, they should physically go to the station the clue led them to. Continuing the game will be possible only after the station location verification process has been completed. This process verifies that the player is exactly at the location of the station. There are two methods available for verifying station arrival, the teacher can prioritize one:

1. Location verification via the GPS sensor. When the player taps the “Check location” button, the application will use the device GPS tracking data in order to verify the station arrival. This method requires the teacher to set the station outdoors, where the GPS satellites signal is high and available. The application will try to achieve a minimum of 15m' satellite accuracy; a screen showing the player the GPS accuracy will be displayed and get updated every second (Figure 5.10). When the GPS is accurate, a feedback on the player's location will be shown. If the player is not at the location of the station a negative feedback message will be displayed, set by the teacher prior the game (Figure 5.11). The message includes also information about the deviation of the player from the station. When the player is in the right location, a positive feedback will appear with a “Continue” button leading to the next stage (figure 5.12).
2. Location verification using a “QR code scanner”: this method was created in order to enable teachers to pinpoint stations whenever the GPS satellite signal is low or not available (e.g., indoors). This method requires additional preparation from teachers prior to the game because they have to physically attach the QR-code image to the station location. When players try to check their location, an “in-app” QR code scanner will pop up and enable the players to scan the QR code image (Figure 5.13).

5.3.1.4 Tasks
After the player has found the station, a pedagogical task might show up in a form of a question. A correct answer is necessary in order to continue the game. Each station may be attached to one task or more. There are 3 types of tasks that can be modified by the teacher: single/multiple selection (Figure 5.14), and open text question (Figure 5.15). The player must answer all correctly in order to proceed.

5.3.1.5 Delay
In order to achieve maximum player's attention to a task and prevent guesses, a delay mechanism is established as some kind of a punishment. Thus, when players answers the wrong answer they must wait 30 seconds before allowed to proceed in the game. Since most of the time "Treasure-HIT" is played as a competition between singles/teams, it was found that this mechanism is strong enough to deny players from guessing randomly and at the same time encourage exploration in order to reach the right answer. Figure 5.16 illustrates the delay preview: at the bottom, a countdown of the time left for the delay is shown.
5.3.1.6 Finding the treasure
Once participants found all stations and completed all tasks, the last question will be the treasure. The treasure is a message modified by the teacher and an image of a chest full of gold (Figure 5.17).

5.3.2 Author environment
The authoring environment is a simple web based application, which enables teachers to create, manage and share games. Teachers can register and log-in to the authoring environment at treasure-hit.net and access their own portal, in which all the games they own are available.

5.3.2.1 Teacher’s Portal
When teachers logs in to the environment, they are redirected to their personal portal (Figure 5.18), which displays all the games they own. Via the portal, teachers can create, update, publish and share their games with other authors, simulate a game, and view the results of games that have been played.

Portal’s Columns review
- **Publish** – Only activities that are marked as published are visible to the players and accessible by code or by browsing games nearby. This function enables teachers to determine whether the game is visible and playable.

- **Game code** - Each game has a unique 5-digit access code. When players attempt to access a game by code, they should know the digits in order to login to the game. When browsing for games nearby, the game code may assist in finding a specific game (e.g. Figure 5.3 & Figure 5.4).
**View** - Opens a web based mobile emulator that simulates the game flow for the teachers. This function gives the teachers the exact look and feel of how the game looks like without physically "playing" the game. Additionally, it provides an opportunity to test the game, before publishing it.

**Results** – When a game was played by more than two players, the results icon is shown and enables access to the results screen which displays a score table with general performance data: general time for completing the game, tasks performance, reaching stations performance, and a table score for all players. An in-depth overview of this feature is available at subsection 9.1.3.

**Management** - Enables to duplicate, delete or share a game with other teachers.

5.3.2.2 **Editing a game**
The “edit game” screen displays the game settings and enables modifying it (Figure 5.19). At the following sections we will elaborate on the different settings.

![Game settings screen](image)

**Game settings**
The game settings screen enables the teacher to change the game’s name, and modify the welcome and ending messages. The welcome message is displayed at the beginning of the game (Figure 5.6) and the ending message is displayed when the player finds the treasure (Figure 5.17).
**Route properties**

Since each game requires the players to physically travel between different stations, the order of the stations may be an important issue to control and modify according to the settings of the game and the pedagogical goals. Thus, the teacher can select one of the following route types:

**Identical:** the route will be identical for all players; the order of the stations will be the same. The teacher can set the order of the stations by dragging each station up and down.

**Random:** the order of the stations will be random, each participant will have a random route, and the teacher does not control a specific starting or ending station. The route is being calculated with an algorithm assuring that all participants have the same distance to the treasure.

**Random with identical end station:** all the stations will be random besides the ending station, which will be chosen by the teacher. This route is the most preferable route by teachers because it enables them to welcome the participants at the last station, gather them all in the same space and reflect on the game and its content.
Route properties columns review:

**First column**: this is a dynamic column that changes according to the route type. It enables sorting whether the route is identical, it is invisible when the route is “random”, and it displays radio buttons to enable selecting the final station for the "random with identical ending station" type.

**Address**: displays the address of the station (if available) which being retrieved from Google Maps API service. This field is not being set by the teacher.

**Edit**: The edit column enables modifying or deleting stations.

5.3.2.3  **Modifying a station**
When clicking on the station’s name, it can be modified. The station modification screen is divided into 4 main sections, and holds all the settings and the data regarding the station: location, verification method, clues, feedbacks and tasks (Figure 5.20).

![Figure 5.20 A station settings screen](image)
Defining a station
The left side of the screen holds data on the location of the station. The location of a station is set by a simple web interface that embeds map enabling the teacher to place the station (as a marker) over a landmark and edit its settings. The teacher must define the minimal required distance from the station (tolerance). Since the game may require exact points of interest that may be difficult to pinpoint in a regular two-dimensional map, further position refinement could be achieved using Google Street View Interface. Figure 5.21 shows how pinpointing a station using Google Street view API looks like.

Verifying arrival method
The teacher can determine the arrival verification method for the player when arriving to a station.

**GPS:** recommended when the station is outdoors, and the GPS satellites signal is high. Using this option means that the teacher does not need to attach any physical object to the station. When the player arrives to the station, the "Check location" button on the mobile apps activates the GPS sensor and calculates the player’s distance from the station and the tolerance defined by the teacher.

**Barcode (QR-code):** an alternative whenever the GPS satellite signal is low / not available. Selection of Barcode replaces the tolerance bar with an image of a QR code, clicking it will display it in a large printable preview. The teacher then should print the QR-code and physically attach it to the station. When the player tries to verify location, an “in-app” QR code scanner will pop-up and enable scanning the QR code image (see Figure 5.13).

**Clues**
When defining a station, the teacher is required to add clue/s (hints) that should help the players in finding the station. The environment enables the teachers to use any type of media as a clue, e.g., simple text, an image, YouTube video, website or audio. There are no limitations to the number of possible clues for each station.

**Feedbacks**
The teacher is required to set a success feedback that will be displayed when players find a station, as well as a failure feedback that should instruct / motivate the players (see Figures 5.11 and 5.12).
Tasks
Tasks are pedagogical events that the player must complete in order to continue playing the game. When a task has been assigned to a station, the arrival to the station will trigger the task. The player must complete the task successfully in order to continue playing the game. The available tasks are single/multiple questions or an open text question.

5.3.2.4 Additional tools for set/edit stations
Sometimes neither the two dimensional map nor the street view contains enough details in order to pinpoint the exact location of a station. In order to provide an alternative, the teacher is able to access the "Treasure-HIT" mobile application and register as an editor. An interface enabling him/her to pinpoint a station from the exact location in the playground, using the device GPS is built for this purpose.

So far we introduced the "Treasure-HIT" environment and presented the most important functionality for both the player and the teacher. In the next chapter we will review the technical aspects of “Treasure-HIT” in order to have a better understanding of how it works as a system.
5.4 “Treasure-HIT” technical aspects
This subsection reviews the technical aspects regarding the system including an end to end technical review and an overview of the technologies being used.

5.4.1 General architecture
Figure 5.22 illustrates the “Treasure-HIT” Architectural components. There are two front-end environments: the author environment as a web browser app and the player mobile application. The term “front-end” refers to the interface responsible for collecting inputs from the teacher/player and parse it to conform to the specification that the back-end (the “Treasure-HIT” server) can use. Both front ends are communicating with the “Treasure-HIT” server as well as with other components (services). A common scenario of how the components are used may be as follows: the teacher uses the author environment in order to create a game; during the creation of the game the author environment uses additional third party services such as Google services and the QR-Code generator. Once the game is ready, the meta-data of the game is sent to the “Treasure-HIT” server which stores it in the DB. When players use their mobile environment to play, the app communicates with the “Treasure-HIT” server and downloads the game meta-data including the game assets (images, audios) which are being stored locally at the mobile device. During the game, the app is using some of the device sensors, like the GPS and camera. Basically, the main business logic of the system lies in the “Treasure-HIT” server which is implemented with Visual Studio MVC 4.5. This framework enables establishing SOA (Service Oriented Architecture) and MVC (Modules Views Controllers) presentation pattern.

![Figure 5.22 “Treasure-HIT” Architecture illustration](image)

**Why “Treasure-HIT” architecture based on SOA?**
Before we explain why SOA is the architectural approach selected for “Treasure-HIT”, we would like to briefly explain what SOA is: according to Brown, Johnston, & Kelly, (2002), this pattern should enable the system to provide services to either end-user applications or other services through published and discoverable interfaces. Rosen, Lublinsky, Smith, & Balcer, (2008) argued that the real power of SOA is the coupling of reusing services to create flexible, agile business processes. SOA is a great way to integrate with systems that the organization does not control. (For
example, to provide services for other organizations to consume or to consume services supplied by other organizations.) (Brown, Johnston, & Kelly, 2002).

“Treasure-HIT” was planned in phases, and from the understanding that it will be scaled up, in terms of system requirements and quantity of users. SOA approach contributed to the scalability and maintainability of the system, in sense that “Treasure-HIT” provides independent services which can easily be modified/scaled with minimum scenarios of breaking changes. Additionally, it was clear that the system should be able to reuse logic since it has more than one interface. Moreover, the existence of large variety of possible consumers (i.e., schools, academic institutions, non-pedagogical institutions) forces “Treasure-HIT” as a system to provide independence of what to do and how to present the consumed content (as an API which expose logic).

**“Treasure-HIT” layers model**

For better understanding the different tiers of the system, and how they communicate with each other see Figure 5.23, which illustrate the system’s layers and the different API’s. Each layer consists of additional components marked in yellow.

Generally, “Treasure-HIT” consists of three layers: any data that flows from the application layer, to the storage layer transfer via public / internal API. Each layer is responsible for different specification as follows:

**Storage layer:** Holds the components responsible for storing data and assets. The main storage DB (Data Base) is an MSSQL Server that holds data of the system entities (see subsection 5.4.1.1); the local FS (File System) contains assets uploaded by the users such as images and audio files. The CRUD (Create, Retrieve, Update and Delete) actions to the DB are being performed using the storage API that enables safe, efficient and reusable interface for writing to the DB.

**Storage API** – A programing interface enabling the business logic layer to write and retrieve data from the DB and the FS. ASP.NET MVC 4.5 provides the “Entity Framework” - a data access library which contain many useful features such
as defining entities without requiring base classes or data persistence attributes, lazy loading support, LINQ operator Support, etc.

**Business Logic Layer** – Holds the entire business logic of “Treasure-HIT” and functions as the “brain” of the system. In this layer, the security rules are configured, and so is the entire system behavior. Since “Treasure-HIT” architecture is service oriented, the business logic layer holds modules that provide services to different consumers. Each one of the modules listed inside is an independent module which exposes reusable independent services to different consumers, for example: the authorization module serves mainly the teachers when they log in via the web author environment but is used again when teachers is trying to log in via the mobile app as authors. The game management is a module that provides services to the editor when defining the games, and to the mobile app when consuming the games’ data.

“Treasure-HIT” Public API - exposes logic from the business layer to any authorized entity. The API is based on HTTP, and support HTTP methods (GET, POST, PUT, DELET). Each HTTP request to the API can contain data in a form of JSON data structure. The Public API is implemented with ASP.NET MVC 4.5 framework that provides a useful feature called "Web API", which enhances the developer with an efficient and secure API mechanism. The main goal of this architectural approach is to provide services not only to the client environments (“Treasure-HIT” web & mobile application), but also to any organization that suggests any additional pedagogical usage of the “Treasure-HIT” abilities and by that potentially enables other developers to mash up “Treasure-HIT” logic in their environments.

**Application layer** – The Application Layer is the highest layer of a system, it does not provide services to any other layer, but only consumes services. The web application was deployed for the teacher to allow him to create and manage games. The mobile application was deployed mostly for the player as an interface for playing games.

5.4.1.1  *Treasure-HIT Data model*

Before covering the architectural aspects of the applications environments, we would like to present the data model and the relationships between the different entities in the system. An entity is a “thing” which can be distinctly identified. A specific editor, game, or station is an example of an entity. A relationship is an association among entities. For instance, “father-son” is the relationship between the Editor and the Game in scenes where each father (Editor) can have multiple sons (Games). The “Treasure-HIT” database contains relevant information regarding the entities and their relationships. Figure 5.24 shows a UML Class diagram that maps the system’s entities, their relationships and the attributes concerning them.
**Editor** – The editor is the most fundamental entity in the “Treasure-HIT” data structure and the primary ancestor of all other entities. Regardless of the administrative properties, each editor can have one or more Games.

**Game** – A game is the entity that holds all the properties concerning a game, it is not an independent entity since it can only be a child of an Editor. A Game is basically a combination of one or more stations, and has one or more players which can play it. Game is the parent entity of Stations and Players.

**Station** – A station is an entity that holds properties about a certain location in the game. It is a child of a Game (each Game can have one or more Stations). The properties of a station actually affect the player and the behavior of the mobile application at the station, for example: the property “VerificationMethod” defines whether an arrival to the station will be by using the device GPS or the barcode scanner. The station is the direct parent of Clues, which helps the player finding the station, and Tasks, which is the pedagogical activity that the player performs after the station is found.

**Clue** – A clue is a child entity of a station. Since there are different types of clues this entity is being extended to support all available types. The basic entity of a Clue holds the common properties for all clue types, and when a complex type is

![UML class diagram of "Treasure-HIT"](image)
used, it extends the basic entity of Clue. The available Entities are: “Website Clue”, “Image Clue”, “Audio Clue” and “Video Clue”.

Task – Similarly to the Clue, a Task is also a child of a station, and each station can have one or more tasks. The supported tasks are single/multiple selection of questions (close questions) or open questions. When the task type is a close question it supposed to have distractors. Thus, Distractor is a child entity of a task.

Distractor – A Distractor holds the following properties: the text of the distractor and whether the distractor is the right answer. This entity is used in two different ways:

1. Close questions: enable unlimited distractors to a Task and unlimited correct answers to a Task.
2. Open questions: enable unlimited forms of answers.

Player – A Player is an entity that represents the participant who registered to a game. When a players want to participate in a game they should select a game and register to it. The Player entity holds all the data essential to identify the players and their performance during the game. During the game all the players’ actions are being logged and sent to the database.

Log – An entity that holds properties about certain action of the player. Each action is being documented and sent to the server. Later on, by questioning the Logs regarding a game, the system is able to calculate the score table, and recapture the entire player moves.

5.4.2 Understanding the Application layer
After we have reviewed the system’s entities of “Treasure-HIT” we continue reviewing the application layer which contains the web based author application and the player mobile application:

5.4.2.1 Web based author application
The author application is a browser-based web interface that allows the teacher to execute administrative actions such as, log-in, retrieve, edit, create and delete instances of the “Treasure-HIT” entities for example, create a game, edit clues, delete task distractors, etc. The web base author application was implemented with the ASP.NET MVC framework which was built to support fast and efficient MVC presentation pattern development. MVC presentation pattern is a paradigm used for separating application logic from the data and presentation components. It is a deployment pattern whereby application code is separated into three distinct areas: those concerned with the core data of the application (the Model), those concerned with the interfaces or graphical aspects of the application (the View), and those concerned with core logic (the Controller). MVC works across multiple tiers: it allows application changing the design easily without affecting the control or data components of the client (Governor, Hinchliff, & Nickull, 2009).
When the author web application makes a request to the ASP.NET server, it is handled by the “Routing System” which interprets the URL and passes the request to the controller; the controller processes the request, operates logic on the request’s data and then passes a unit of data (model) to the view engine which uses it to generate a valid HTML page. When the page is fully generated it responses back to the client (MSDN, 2013). For example: The request for http://www.treasure-hit.net/he-il/Games holds two main URL parameters:

**He-il** - a “route parameter” used by the view-engine telling it in which language it supposed to generate the page.

**Games** - used by the Routing system, this parameter is basically the name of the controller that supposed to handle the request. The controller uses the storage API in order to retrieve a list of all the games owned by the user. Then, this list is passed to the view engine which uses it to loop throughout the entire games and generate the HTML page which presents them. Finally, the page is returned to the client (For example: Figure 5.18).

“Treasure-HIT” view engine uses ”Razor”, which was created to enable developers to write static HTML and make it dynamic by adding server code to it. One of the core design goals behind Razor is to make the coding process fluid, and to enable developers to quickly integrate server code into the HTML markup with a minimum of effort (Guthrie, 2010).

**Tools and technologies**
The author web based application uses the following tools and technologies in order to create full and extensive user experience: HTML, JavaScript and jQuery libraries, AJAX, Third party Google’s APIs. Further information regarding these tools and technologies can be found in the Appendix.

5.4.2.2 The player mobile application
After reviewing the author application and the technologies involved, we continue with the player application technologies and architecture.

**Mobile development approach**
The player application was targeted for mobile devices and so was the technology selected for implementing it. It was decided that the technology for the “Treasure-HIT” player mobile app will be web based that include HTML5, JavaScript and CSS3. For justifying this decision we would like to present some mobile development alternative approaches and background:
At the beginning of the smart-phone and tablets boom, it was popular to create mobile applications using the native development platforms such as C++, JAVA and Objective-C. Although it is still very popular, native platforms development holds some limitations that should be taken into account: cross platform incompatibility, code management and the need of development expertise in each native platform. An alternate solution for these problems is developing one application that has the ability to run on all mobile devices regardless of the operating system that the device is using or its screen resolution. Thanks to HTML5, CSS and JavaScript this is possible. Thus, as long as the target device has a web browser, it will be able to run the HTML based applications (Ohrt & Turau, 2012). Compared to native language, HTML5 does not provide an API for accessing the devices native functionality, such as the battery status or file storage. These functionalities are currently under development of the Device APIs working group (W3C) and might end up as supported features in browsers in the near future (Hirsch, 2013). Before the HTML5 API will be ready for cross platforms devices use, there are some challenges to overcome. Luckily, there is a hybrid solution called “PhoneGap”. Hybrid applications are mobile web apps placed in a native container that enable them to access the devices native functionality. Hybrid solutions are most suitable for projects with lower budgets but still require access to the devices native functions. Hybrid apps are faster to develop than native applications; they provide more functionality than a Web app and they can be uploaded to an OEM (Original Equipment Manufacturer) app store like Apple App Store and Google Play Store.

“Treasure-HIT” Mobile App Architecture
After elaborating on the selected JavaScript framework we would like to present the mobile application architecture. Figure 5.26 attempts to illustrate the app architecture.

AngularJS as a framework enables client-side development using the MVC approach which enable developing the application while separating the views (templates), controllers (business logic) and models (data objects).
Custom Services
AngularJS also suggests a mechanism for enabling the developer to create services, which are pieces of data and logic available by dependency injection (DI) across the entire lifecycle of the app. “Treasure-HIT” mobile app uses three main services:

**Games data service:** Responsible for handling the data regarding the app. Few common examples are:
- When a player registers to a game, the data regarding the game is downloaded, parsed and later saved to the device file system.
- During the game, additional data regarding the player’s actions is logged.
- It is possible to access multiple games from one device, and in each game the progress of the player is saved.

This service is responsible for parsing and manipulating the data before saving it to the device file system or before sending it back to the server.

**Common Logic:** holds logic regarding general actions that are triggered from everywhere, e.g., presenting a popup message, exiting the game, settings screen, encode /decode strings, etc.

**Device Features:** holds logic regarding the “PhoneGap” API. An example of the service responsibility would be:
- Activating the **Camera** for scanning a QR-code.
- Activating the **GPS** and processing its data.
- Activating the **Audio**.
- Saving data to the **Device File System**.

Any activation of the devices’ sensors is triggered only by this service. This service is being practically used as the mediator between the app and the devices’ sensors.
6 Research questions and hypothesis
Based on the description of the problem highlighted in section 1.1 and the aims of the research, our work may be summarized with the following research questions:

1. **Which modifications and additions are required in the "Treasure-HIT" environment in order to extend its pedagogical opportunities?**
   a. What are the design implications of these changes?
   b. What are the technological implications of these changes?
2. **How were the changes adopted by users (editors and players) and how did these changes enable a richer and more effective use of the system?**

**Hypothesis:** Understanding the needs of the teachers and modify "Treasure-HIT" accordingly, will enable a more effective and meaningful use of the system for educational purposes.
7 Methodology

In order to answer our research questions we followed an approach that starts with a literature review, continued with mapping the "Treasure-HIT" environment and reviewed similar platforms. We tried to identify specific actions and issues that have been reported as challenges by the teachers who have experience working with the system. After having identified the problems we continue to further plan a design-based research model which we could use and gather data to solve such issues.

7.1 Design based research

"Design Based Research" (DBR) is used to investigate a learning environment and examine it by several iterations. DBR involves creating "theory-based" educational innovations, often by the development of learning environments for solving local problems and issues. The design complexity results from gathering evidence in any iteration and thus promotes the final design research process (Barab, Design-Based Research, 2006). DBR can be used to understand instructional materials especially technology enhance learning environments. DBR is being conducted in respect to educational goals and guided by instructional frameworks such as knowledge integration. DBR supposed to be conducted in authentic settings and collect facts regarding the influence of different causes on the learners learning process and perception. DBR can improve the usability of an interface, or a learning process by focusing on how people think, know and learn. DBR not only recognizes the importance of contexts locally but focuses on the changes of these contexts as evidence to the existence of a theory (Barab & Squire, 2004). Wang & Hannafin (2005) argued that one of the methods to accomplish results is by iterative cycles of: design, enactment, analysis, and redesign.
8 Data collection, analysis and results

After a short review of DBR and its characteristics, we present, in detail, the steps taken to detect the limitations of “Treasure-HIT”. We offer and implement solutions and test how the implemented solutions were being used. Our research included 3 design-based research cycles (iterations); Figure 8.1 illustrates the flow of the design-based research iterations.

The research included 3 iterations which aimed to improve the "Treasure-HIT" environment and adjust it to best cater teachers’ needs. The data collection before each iteration led to the implementation of different features. A summary of all the requirements collected during this research is presented in table 8.1.

<table>
<thead>
<tr>
<th>DATA COLLECTION</th>
<th>REQUIREMENTS RAISED BY TEACHERS</th>
<th>SUBSECTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1 Interviews of teachers-first adopters</td>
<td>• An ability to set indoor stations by using QR code scanners.</td>
<td>8.1</td>
</tr>
<tr>
<td></td>
<td>• A presentation of the distance between the player and the station for outdoor stations.</td>
<td></td>
</tr>
<tr>
<td>#2 Periodic teachers training at &quot;Beit Berl&quot; college.</td>
<td>• The need to be able to pinpoint the exact location of a station on site and not only from a distance, by using Google map.</td>
<td>8.2</td>
</tr>
<tr>
<td></td>
<td>• The need to discourage random guessing when answering the game tasks. A &quot;punishment&quot; mechanism, which is activated after failure in a task.</td>
<td></td>
</tr>
<tr>
<td>#3 Periodic TEL assimilators training at &quot;Nofy Yam&quot; elementary school.</td>
<td>• Removal of the dependency on constant Internet connection during the entire game.</td>
<td>8.3</td>
</tr>
<tr>
<td></td>
<td>• Presenting the game results in various ways.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Game sharing between teachers.</td>
<td></td>
</tr>
</tbody>
</table>

It is important to note that besides the specific events mentioned in Table 8.1 as "data collection" we had continuous contact with active editors who used the system for creating games for their students, and received feedback on their experience of using the system and on the efficacy of the changes introduced during the iterations.
In the following subsections we will detail each iteration in the following order:

1. Overview of data collection and analysis.
2. Solutions and implementation of technical aspects.
3. Results of the iteration including information about its contribution.

8.1 Iteration #1 – Interviews with first adopters

In October of 2012, a series of interviews was conducted with 5 teachers who were amongst the first users of "Treasure-HIT" and conducted at least one game with their students. The goal of the interview was to discover the limitations of the environment and to identify the core problems encountered by these users during their attempts to develop and enact games. During the interviews, the teachers shared their experience as authors and their students’ comments on the game. Two major issues were raised by the teachers:

The environment limitation: The existing environment supported pinpointing only outdoor stations. In location-based games designed for educational purposes, there is a need to also include stations that may be located indoors, where detection by GPS is not applicable. The ability to include such stations in a game would enrich the pedagogical potential by addressing specific exhibits and objects that might be presented indoors (in school corridors, in an exhibition or museum etc.)

The mobile app: When confirming arrival to a station the existing app provides a dichromatic feedback: confirmation or denial of arrival with no additional details. Teachers who accompanied students playing the game reported that finding a station was sometimes frustrating because the players did not get any indication on their distance from the target. The teachers recommended that the app help players by providing additional information displaying their deviation from the actual target.

8.1.1 Designing and implementing the solutions

8.1.1.1 Enabling indoor stations

In order to enable teachers to include stations that may be located indoors, we had to think of a solution that will not rely on the use of GPS. This solution has to be easy for the teacher to define via the author environment, and easy for the player to use via the mobile app. We came up with two possible solutions:

1. The author environment will generate a 5-digit code for each indoor station, and the teacher must physically paste the code in the station's location. When the players arrive at the station, they have to insert the code to let the app verify that they are at the right location.
2. The author environment will generate a QR-Code image, which the teacher should paste at the station. When the players arrive at the station, they would then scan it in order to verify their location.

We decided to implement the QR-Code solution since in terms of usability it was easier for the players to capture using the device camera then use the keyboard and type digits. Moreover, QR-Codes are not readable by humans, and therefore players cannot decipher them or transfer them to other players. The next sections review the technical aspects of this solution and its implementation in the system.
QR-Code
Current smartphones by default contain applications that take photos and record videos using the phone's embedded camera. The embedded camera can be used as a new input interface for symbol recognition such as QR-Codes. This code combines symbols that represent a string of characters. Scanning it can be used for easily accessing web pages or WiFi network services by reading URLs or such address characters. QR-codes have a large capacity, small print out size and high speed scanning. (Ohbuchi, Hanaizumi, & Hock, 2004).

Figure 8.2 illustrates the structure of QR-Codes, which are comprised of the following patterns: finder pattern, timing pattern, format information, alignment pattern, and data cell. The reorganization process is based upon the found pattern and alignment, and the alignment is used for adjustment of code deformation (Ohbuchi, Hanaizumi, & Hock, 2004).

Technical aspects of enabling teachers to define indoor stations:
We had to design a solution that enables the teacher to add indoor stations without affecting the existing functionality of defining outdoor stations. Figure 8.3 is a state UML diagram that shows the teacher’s steps of creating a station. The steps marked with a dotted frame highlight the functionality of creating indoor stations that we implemented as a result of this iteration. It can be noticed that after the teacher decides to add an indoor station, the author environment should generate a QR-Code image representing a unique charset of the station. In order to do so, we used an API by: qrfree.kaywa.com, which requires a string of characters as an argument and size of the image, it then returns the same string coded in the form of QR-Code image. Since each game and station has a unique ID at the Database, we used both in order to generate the QR-code image.
For example, if the game unique ID is 34 and the station unique ID is 213 the use of the API might look like:

```
<s src="http://qrfree.kaywa.com/?s=14&d=34,213" alt="barcode">
```

The s argument contains the size of the image, and the d argument contains the string which the QR-Code image should represent. Figure 8.4 and 8.5 illustrate the final implementation of this feature within the author environment:

According to Figure 8.4 the teacher can modify the "arrival verification" mechanism and choose between GPS or Barcode. When Barcode is selected, a small icon of QR-Code is shown, a click on the icon will redirect to a page that holds the printable version of the QR-Code (Figure 8.5). The teacher should print it, cut the name of the
Technical aspects of enabling players to verify indoor stations:
The challenge for the player side was considerably greater. Figure 8.6 is a UML state diagram that shows the flow regarding the player when arriving to a station. The steps marked with the dotted frames on the left-hand side of the diagram shows what the player should do when arriving to a station based on a QR-Code verification mechanism. In order to scan the QR-Code, the app should activate the camera, capture the QR-Code and interpret it to text. As described in subsections 5.4.2.2 the player’s application is based on HTML5 and JavaScript. Interpreting the QR-Code is an action available only by image recognition, which JavaScript does not support. However, “PhoneGap” provides a QR-Code/Barcode scanner plug-in which can be compiled with the app files. The QR-Code plugin exposes a simple API that activates the camera, scans the QR-Code image, interprets it and returns its string value.

![Figure 8.6 A state UML diagram showing the flow at the player side when arriving a station](image)

Figure 8.7 is a screenshot of the player app when the station set to use a QR-Code verification mechanism. Figure 8.8 illustrates the QR-Code scanner activated on the mobile app, and the player is about to scan the QR-code.
8.1.1.2 Providing additional information displaying the deviation from the station

Teachers reported that the denial feedback when players fail to arrive at an outdoor station may not be sufficient and that some additional information is required in order to prevent frustration and help students identify the station. In order to solve this problem we decided to add information about the player’s deviation from the station. Figure 8.9 illustrates the implementation of this feedback.

Technical aspects of displaying the deviation of the player from the station
When the player is trying to reach a station, the app already has the coordinates of the player and the station. For calculating the deviation we used a simple math calculation from "Movable Type Script" that calculates the distance as the crow flies between two coordinates, by considering the radius of the globe (Figure 8.10):
The $d$ variable represents the final distance (km).

### 8.1.2 Results of iteration #1

After Iteration #1 the system was expanded to include indoor stations.

Game editors, old and new, embraced this feature and used it to create rich and versatile games. One teacher said: "Treasure-HIT" becomes an environment that supports three types of games: outdoors, indoors and a combination of the two". Following, are two examples of activities conducted by teachers that were possible due to changes introduced in iteration #1:

- **Classroom math game** – a game conducted to middle school students, as part of a math lesson. The teacher created a game combined of 20 stations; the clue for each station was a math problem. The answers to the problems were spread in different places on the walls of the classroom. A station’s QR-Code was pasted next to each answer. The students were divided to teams of two, which competed with each other. The teacher reported that the activity was very successful and the students were very motivated.

- **Meet your village** – a game that was conducted as part of a weekly "Young Maccabi Movement" activity at "Nili" village. The instructors directed an activity in which the students had to solve clues about the history of the village and find places of interest. The game combined both outdoor stations like playgrounds or gardens at the village and indoor stations like the gym, the library, the school, and the fire department offices.

From observing the teachers' activity and based on communication with some of the teachers it seems that they felt comfortable with the process of defining an indoor station, some reported that it is easier than setting an outdoor station and much more accurate.

A retrospective analysis of the games created with the system after Iteration #1 and until 2014, emphasizes the impact of this change (Figure 8.11). It seems that most of the games (56%) created with the system included indoor stations detected by QR-Code while 12% were based ONLY on indoor stations.

```javascript
var R = 6371; // km
var dLat = (lat2-lat1).toRad();
var dLon = (lon2-lon1).toRad();
var lat1 = lat1.toRad();
var lat2 = lat2.toRad();

var a = Math.sin(dLat/2) * Math.sin(dLat/2) +
    Math.sin(dLon/2) * Math.sin(dLon/2) * Math.cos(lat1) * Math.cos(lat2);
var c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1-a));
var d = R * c;
```

Figure 8.10 The algorithm to calculate air-line distance between two points
Another enhancement introduced in Iteration #1 provided additional informative feedback in case of missing an outdoor station. Teachers that enacted games with their students reported that displaying the actual deviation from the target was effective since it increased player’s sense of orientation and provided additional scaffolding for identifying the target. The students reported that this feature is helpful and important, and some even wished that such feedback could be available for indoor stations as well.

8.2 Iteration #2 – Periodic teacher training at "Beit Berl"

On the 30th of April 2013, the second iteration of the research was started. An event which included data collection, took place as part of a periodic training of teachers under the topic "Location-based learning".

About the audience: In Israel, there is a special program, named "Hotam program", led by the ministry of education. It aims to attract young, educated people to work as teachers for the ministry of education, under the slogan that “every student deserves a good teacher”. The teachers who joined the program are enthusiastic, technology-oriented, and relatively young, between the ages of 25 to 32. As part of the program, the teachers participate in training sessions focusing on various educational issues.

The “Treasure-HIT” workshop was conducted as part of training on "Location Based Learning". 35 teachers from the program participated in a full day workshop.

The workshop program included the following stages:

**Experiencing a “Treasure-HIT” game as players.** Prior to the workshop, we created a game on the "Beit Berl" campus. The goal of the game was to demonstrate a location-based activity and to introduce the "Treasure-HIT" environment to the teachers while highlighting its potential. We created a game consisting of 5 stations, on the topic of "Beit Berl” campus local history", 3 stations were outdoors and 2 stations were indoors. In order to solve the clues to the different stations, the teachers had to use the campus map (Figure 8.12) and search for data on the web. We asked the teachers to form teams of 2-3 each with at least one smart-phone per team. The
setting of the game was a competition between teams whereby the first team that finds the treasure, wins. It took 45 minutes for all teams to complete the game. Some ran, some walked, but they were all excited playing the game.

**Figure 8.12** The teachers are trying to solve a riddle using the campus map

**Figure 8.13** A team has reached the correct station

**Figure 8.14** A team is trying to answer an open question task, notice that all team members are participating, by searching the answer on the web, using their smartphones

**Reflections as players.** When the game was over, the teachers shared their feelings and thoughts about the experience, mentioning the potential relevancy of the tool to the various subject domains taught at school.

**Creating a game as editor.** A hands-on instruction session on how to build games using the “Treasure-HIT” environment. The participants were instructed to create a sample game including two stations focusing on subjects of their personal choice.

**Immediate reflections on working with “Treasure-HIT” as an author.** After the teachers themselves had built a game using the author environment, they were asked to reflect on the author interface and point out potential difficulties and limitations.

**Later reflections:** After the workshop we kept contact with the participants that had actually created games for their students and interviewed them personally in order to gain more accurate feedbacks based on real experience with students.
Insights from participants of Iteration #2

The following issues were raised by the workshop participants:

1. It is difficult to define the exact location of an outdoor station when Google maps does not provide detailed information for this place and when Street View is not available, for example: in parks or public playgrounds, within a campus or small alleys. Furthermore, teachers mentioned that preparing a successful game involves a preparatory field trip. During such a trip, ideas for new stations may pop up, therefore; it would be useful if one can define the location of a station while touring the actual site.

2. Some of the teachers admitted that they had "cheated" during the game. Since the setting of the game was competition, whenever the teachers encountered multiple / single selection questions in some of the tasks, some teams just guessed the right answers without even reading the question or trying to solve it, fearing that delving into each task might slow them down. They were sure that if they "cheated" their students will probably cheat too. The teachers suggested adding a punishment mechanism that would discourage random guessing.

8.2.1 Designing and implementing solutions
8.2.1.1 Enabling the editor to define a location of a station via the mobile app
In order to overcome the teachers’ difficulty in defining outdoor stations in areas that are not covered by Google Street View, and enable the teachers to define outdoor stations spontaneously, we decided to provide functionality for sampling the coordinates of a station on site. For this purpose, we extended the "Treasure-HIT" player application and added a "mini" author environment with limited functionality within the app. In order to access the author environment the teachers have to tap "Edit games" on the setting screen (Figure 8.15), then enter their author credentials (Figure 8.16), which will redirect them to their mobile app portal (Figure 8.17). All games created by the author environment are available for editing. Teachers can watch available games, add a new game, and by tapping a game, watch its stations (Figure 8.18).

When tapping a station, there are 2 options available (Figure 8.19):

- Rename – renaming a station.
- Sample – sampling the coordinates of the station and register coordinates in the “Treasure-HIT” database.
When the teacher clicks the "Sample" button, a confirmation window pops up (Figure 8.20). The app will attempt to communicate with the GPS and update the teacher on its accuracy (Figure 8.21), which must be at least 15 meters in order to sample a location. When the location has been sampled, a success notification pops up, instructing the author to continue editing the station from the website interface (Figure 8.22).

**Technical aspects of enabling teachers to define a location from the mobile app:**
The development and implementation of this feature required only some modifications of the mobile app: since the author environment is just an interface working with the public API, we were able to use the API as it is and not extend it, which means that the author web interface and the author mobile interface share logic regarding the station CRUD (Create, Retrieve, Update Delete).

**Security issues:**
Since the public API service which is responsible for updating the database is available from any end, we had to make sure that only the genuine author will have access to it. The author validation mechanism works as following:
When the teachers log-in to the app author environment, their credentials are sent to the server for validation, and the server responds with an API token. This token is used as a key to identify the teachers, so every time a teacher tries to CRUD from the mobile app, the token is being attached to the request. CRUD actions are allowed only with an API token.

**8.2.1.2 Adding a delay mechanism to the game**
In order to prevent players from "cheating" by random guessing, we took the teachers' advice and created a delay mechanism. Since the activity is being presented as a competition, time is an important factor for players. This mechanism prevents the player from making any progress in the game for 30 seconds after making two mistakes in the same item (Figure 8.23).
Figure 8.23 Example of a delay for answering a wrong answer

Technical aspects of the delay mechanism:
The delay mechanism is a global function being called when the player makes a mistake twice in a row for the same item. It disables all buttons and counts down until the delay is over. The clock being displayed is an animated gif.

8.2.2 Results of iteration #2
After the features were developed they were tested and released. The teachers who participated in the training event at "Beit Berl" were told that the improvements they suggested were implemented in the new version of the system. We monitored the teachers' activity in the "Treasure-HIT" environment, and interviewed 5 teachers who conducted games in order to have a feedback on the new features.

After Iteration #2 the system was expanded to enable defining an outdoor station via the mobile app. The teachers' feedback on the "in-app" authoring environment were very positive. Teachers reported that they used this option as an alternative when Google data on the area was limited. In addition, they reported that when they are in an interesting location, they can spontaneously and easily save the location as a station and combine it later in a game. These behaviors exercise seamless learning in the sense that teachers can use their smartphones and sample "Treasure-HIT" stations whenever they are in a place they recognize as a point of interest with a pedagogical value, regardless of their current context.

The other change introduced in Iteration #2 is activating a delay in case of repeated errors in the same question, in order to discourage random guessing and encourage coping with the content. Teachers that enacted games with their students reported that the delay mechanism was quite effective for preventing participants from trying to guess the answers. The teachers also reported that the "guessing behavior" that characterized students, decreased and that the majority of the players really tried to solve problems instead of guessing and lose time for guessing incorrectly. Some teachers suggested improving this feature by allowing the teacher some kind of control over the punishment, for example: to modify the delay by selecting the length of the delay.

8.3 Iteration #3 – TEL assimilators of "Tel Aviv"
On 11th of July 2013, the third iteration had started as part of a training session of TEL assimilators at the county of "Tel Aviv".
About the audience: TEL (Technology Enhanced Learning) assimilator is a relatively new position offered by the Ministry of Education of Israel. Their teachers volunteer to be responsible of the assimilation of TEL in their schools. Each school in the "Tel Aviv" county has at least one TEL assimilator. In each quarter there is an event at which the assimilators learn and experiment with new TEL tools.

Due to “buzz” generated by the successful workshop of iteration #2, we were invited to the event in July 2013 which took place at the elementary school "Nofey-Yam" to present "Treasure-HIT" and teach the assimilators how to use it and embed it in their school activities. 42 TEL assimilators from elementary schools participated in the event which included the following stages:

Demonstrate "Treasure-HIT" by conducting a game inside the school boundaries with the TEL assimilators as players. Prior to the workshop, we created a game within school boundaries. The goal of the game was to demonstrate a variety of disciplines and to try to expose the assimilators to various uses that can easily be transferred to their own pedagogical needs. We created a game with 5 stations. The stations of the game were points of interest that were available at the school e.g., the library, the art class, the school yard, etc. In order to solve the clues of the different stations, the participants had to search for answers on the internet, search for information in books at the library, and pay attention to their surroundings. We opened with a short background about "Treasure-HIT" (Figure 8.24), we then asked the participants to form teams of 3-4, similarly to iteration #2. It took 40 minutes for all teams to complete the game.

Discuss the activity and the relevancy of “Treasure-HIT” in elementary schools, its potential and how it could be integrated in elementary schools curricula.

An instruction session on how to build and conduct games using the environment. As an exercise, each assimilator built a short game on school grounds.

Reflections of the experience. Once the practical part ended, we gathered again and discussed the author environment and documented the participants’ feedback.

Figure 8.24 Introduction lecture to “Treasure-HIT”

Figure 8.25 A team of two assimilators trying to solve a clue
Figure 8.26 A team of 3 assimilators trying to solve a task

Insights from participants of Iteration #3

The contribution of this audience was very different from the previous ones. Most of the participants were TEL oriented who had prior experience with many TEL environments. Their suggestions originated from their experience and their perception of how the environment can be embraced by their fellow teachers. Three suggestions which later translated into requirements were offered by the majority of the audience:

1. The mobile app dependency on a constant internet connection limits the game potential. During the game, when participants are out of WiFi range they can play only if they possess mobile internet package, which can be expensive and not available to all students. This limitation forces the teacher to take in consideration the WiFi range when planning the game. A better approach is to enable players to play with no constant internet connection all along the activity.

2. Lack of closure: the teachers confirmed that a mobile “Treasure Hunt” game is a very motivating educational activity. Nevertheless, the teacher should be able to follow and monitor the specific activity of each team and to present the performances as part of a summary session of the activity.

3. Sharing games with colleagues – the teachers pointed out the need to offer some kind of sharing mechanism that will allow collaboration between fellow-teachers in order to learn from each other.

8.3.1 Designing and implementing solutions

8.3.1.1 Remove the dependency of constant internet connection from the game

When a player registered to a game, the mobile app communicated with the public API in order to save the player data to the database, and then retrieved only the data and assets for the first station. When the player finds the first station and completes the tasks, the app makes another request to the API in order to retrieve the data and assets of the next station. This behavior can frustrate players who are supposed to receive the clues for the next station but are out of WiFi range. Only players with an internet communication (3G) package can continue the game, which creates an unfair situation and limits the teacher to pinpoint stations only in places covered by WiFi communication. In order to solve this challenge we decided that all the game data and assets will be downloaded when the player is registering to the game. Only then is an internet connection is required, and not during the rest of the game.
Technical aspects of removing the dependency of constant internet connection:

Figure 8.27 shows the flow where a constant internet connection is required. Figure 8.28 shows the revised flow of a game after we improved it. We can see that an internet connection is required only upon registration, and then the data and assets of the entire game are saved locally and available throughout the game. We implemented this solution by retrieving a JSON object that represents the game (Activity) which includes all stations of the game; we then set a pointer to the current station. We save the JSON object for each progress of the player in the game to a file inside the device file-system. If during the game players receive a call or for some reason the app is closed, they are able to continue exactly from the place they left the game. By doing so we have also improved the continuity of the game.

8.3.1.2 Solving the lack of game closure

One of the main insights from teachers' feedback was the lack of closure and that there is a need to present a detailed summary of players' performance at the end of the game.

Since the system is designed for educational purposes, it was important to take into consideration not only the timing, but also the success in coping with the content challenges, as reflected by the questions presented at the stations. Another challenge was to support various possible enactment scenarios of a game. A game may be conducted on different days with different groups of students, for example, different classes in a school. The teacher should be able to decide how to present the game summary in order to provide a relevant grading of the results, for example: to present the grading of each class separately.

Figure 8.29 presents an example of the summary page we designed and implemented. The game’s summary page is designed as a table that presents all teams/players performance. Teams/players are sorted according to overall results in the game. Since a game can be repeated and played on different occasions, we decided to enable presentation of results within a certain time-frame. The teacher can set the relevant time period to match the time that a certain group of students played in order to
display only the results of this group. A display of all the results accumulated for this game is also available.

The system calculates the actual total time of a team for completing a game and displays it independently.

The relative grading of the teams is based on the following algorithm that takes into account completion time, location errors, and content errors. Each content error is weighted as 6 min delay and a location error as a 3 minutes delay. Then the weighted time is recalculated and used to define the group's ranking, relative to the other groups included in the configured time period. For example, Figure 8.29 shows that "Jonathan and Dana" completed the game in 32:59 minutes which is faster than "Beauty and the Beast", but still, they ranked lower than "Beauty and the Beast" because their performance in the tasks and at discovering the stations was better.

**Technical aspects of the results table:**

On the server side data model, a "Player" is an entity that contains a few attributes and a list of a different entity named "Log". When a Player registers to the game, the Player data (name of the team and the game ID) are sent to the Database via the public API. During the game, every action the player takes is monitored and saved to a "Log" object. If an internet connection is available, the Log data is sent from the device to the server, if there is no internet connection, the app waits for an available internet connection and then sends the unsent logs. When the game is over, the group performance data is cached and the following results are saved to the Group entity:

- Total time to complete the game
- Number of location errors
- Number of content errors
- Total time after adding extra time according to errors

When the teachers display
the results they get a sorted list by the total time + extra time according to errors. Figure 8.30 illustrates a Group entity structure as a JSON format.

![Figure 8.30 JSON format of a Group object which completed a game](image)

From the JSON structure of the group we can see that the "TotalGameTime" attribute is 00:32:59, but, the results are sorted by the "TotalGameTimeAfterCalculation" attribute which has a value of 01:20:59.

### 8.3.1.3 Sharing functionality enable teachers to share games with each other

The assimilators suggested that when teachers conducted a successful "Treasure-HIT" game, they should be able to share it with others. They suggested a sharing mechanism similar to web applications that they are familiar with, like "Google Drive" documents, which enables users to share documents with other users and simultaneously edit them. We agreed that adopting this functionality would probably be the best solution for a sharing mechanism but at this point, in terms of time and development resources, the cost is too high. Instead, we temporarily enabled a much simpler functionality that enables teachers to send games they had created to other teachers by technically duplicating their games, and transferring ownership of the duplicated games to a different teacher. When the author clicks on the sharing icon on the management column, a dialog box pops up. Figure 8.31 shows the dialog box asking them to type the colleagues’ email. If the email exists in the system, a different dialog pops up asking the teachers to confirm the action (Figure 8.32).
After confirmation, an email is sent to the teacher who "received" the game, telling him/her that a game was sent by a colleague. From this point on, the game is available in the teacher’s games repository, and the teacher can change it since he/she now owns it.

**Technical aspects of the sharing enabler:**
Technically we reused logic in order to support this feature. Game duplication was enabled from the first version of "Treasure-HIT": on the management column of the games portal (Figure 5.18), the teacher can press the duplication button for duplicating an entire game including its stations, clues and tasks. The only property not being duplicated is the game code, which is unique to a game. Each game entity has a property name "EditorID" that points out a relation of one to many between the editor entity and the game. In a regular game duplication, this property is duplicated as well. In order to enable sending a game to a different editor, we re-used the duplication logic except for changing the "EditorID" property to point to the editor the game was sent to.

8.3.2 **Results of iteration #3**
Like the previous iterations, we announced the release of new features to the group of authors. We monitored teachers who used the environment and interviewed 5 of them in order to assess the modifications' contribution to the teachers and their students; the results were as follows:

The removal of the dependency on a constant internet connection extends the teachers’ opportunities while conducting a game. Teachers no longer had to plan games at locations covered with WiFi internet connection for students that do not have a 3G internet package. When the mobile app was not dependent on an internet connection the playground size became larger.

An example to the usefulness of this feature is a game made for students of the “Nir Yitzhak” kibbutz, on April 2014 on the topic of Passover. The stations in this game were spread all over the kibbutz (Figure 8.33). The dining room (the largest red marker), the only place in the kibbutz that has public WiFi, was defined as the starting point where the students registered to the game. The other stations are not covered by public WiFi. The only way such game could be defined without worrying that all players have a mobile internet package is possible only due to this implementation.
The second enhancement that was introduced in Iteration #3 is the most meaningful addition according to teachers. Teachers reported that it provides a good closure, which was previously unavailable. The score table was usually presented during a summary session at the end of the game. Figure 8.34 shows an example of such a session held in the school main theater during which the headmaster presented the score table to the students.

The filtering tools which allow reviewing classes’ results according to a time frame allowed the teachers to conduct one game for different classes, and at the end of the game present to each class only the results that were relevant to the class members. All the teachers we interviewed agreed that the score calculation algorithm is reasonable since it reflects not only the time but also the tasks’ performance, which is important from an educational perspective. A few teachers suggested adding a detailed statistical analysis of the specific replies to the questions.

The last enhancement introduced in Iteration #3 allowed teachers to send each other games, reuse activities, and learn from peers. Whenever a game is shared, the system sends an email to the teacher with the information about the game shared.
During the period of October 2013 to April 2014 57 such emails were sent by 39 different teachers, providing clear evidence that teachers used this feature to share activities amongst themselves.

The teachers we interviewed agreed that the sharing feature helped them reuse a successful game created by a colleague and refine it to their pedagogical needs. According to some of the teachers, there is still work to be done in order to make the environment sharing-oriented. They suggested creating a portal containing best practice games, for example, so teachers would be able to select a game they like from a list, copy it to their repository, and use it.
9 Summary
This research focused on improving the first versions of the “Treasure-HIT” environment in order to meet teachers’ pedagogical needs. The research addressed the following questions:

1. Which modifications and additions were required in the "Treasure-HIT" environment in order to extend its pedagogical opportunities?
2. How were the changes adopted by users (editors and players) and how did these changes enable a richer and more effective use of the system?

The research was based on three consecutive cycles of testing the system with teachers, collecting and analyzing their comments as game creators, identifying meaningful problems encountered by the users, designing and implementing solutions to these problems, and then retesting the system with users. The research lasted 16 months and resulted in the development and incorporation of seven new system features:

Enable indoor stations: this feature was the most important and most “game changing”. Teachers embraced this feature and used it to create richer activities that included referring to information and exhibits which are displayed indoors. 66% of the activities created with the system included indoor stations.

Provide additional informative feedback in case of missing an outdoor station: Teachers that enacted games with their students reported that displaying the actual deviation from the target was effective since it increased player’s sense of orientation and provided additional scaffolding towards identifying the target.

Enable defining an outdoor station via the mobile app: The teachers’ feedbacks on the "in-app" authoring environment were very positive. Teachers reported that they used this option as an alternative when Google data on the area was limited.

A "delay mechanism" on the player app, activated after two consecutive wrong answers to a question: Teachers that enacted games with their students reported that the delay mechanism was quite effective for preventing participants from trying to guess the answers.

Removal of the dependency on a constant Internet connection during the entire game: This feature extended the teachers’ opportunities while conducting a game. Teachers no longer had to plan games at locations covered with WiFi internet connection for students that do not have a 3G internet package.

Score table: Enabling the teacher to present the game results in various ways. This feature provided a closure to the game and was used during the summary session for conducting a discussion and reflection about the activity.

Sharing functionality: This feature enabled teachers to share games with each other, reuse activities, and learn from peers.

During the period of this thesis teachers nationwide started to adopt “Treasure-HIT” as a tool for creating location-based educational activities. The graph below (Figure 9.1) shows the “Treasure-HIT” adoption with respect to the progress of the research.
Since the end of this research, many more teachers registered to “Treasure-HIT” and used it to create educational games for their students. After December 2013 about 100 new teachers joined each month, as a result of teacher training programs held in various places and of personal recommendations of peers who used the system successfully. Even if the growth in system's adoption cannot be attributed to specific changes and system features, the work described in this thesis has contributed to the creation of a more effective and usable system that teachers are willing to use, and recommend to their peers.

10 Limitations
The “Treasure-HIT” environment has more than one target audience. First and foremost it was created as a tool to support learning and provide teachers with a location based alternative to the traditional learning methods. In this work, we focused mostly on the teachers’ side by modifying the environment to be suitable to teachers’ needs. We must not forget that the teachers’ target audience is their students, and we believe that extensive research is required in order to understand how to adjust the player environment for better support of their learning. Furthermore, we conducted only three iterations, any further attempts might discover additional needs and options that can be supported by “Treasure-HIT”.

Figure 9.1 “Treasure-HIT” registered editor’s graph until April 2014
11 Further Work

The widespread use of the system led to new teachers' applications aiming to diversify the type of activities. The department of instructional technologies of HIT (Holon Institute of Technology) is conducting ongoing work with teachers in order to understand these needs. Further study on the “Treasure-HIT” environment should attend to the following topics:

**Locating meaningful games** - The extensive adoption by teachers created difficulty in finding meaningful pedagogical games among the hundreds of games available. Since the “Treasure-HIT” mobile app is available for download from the app stores, users can download and use it freely regardless of any relation to a certain pedagogical school activity and can browse the games nearby (see Figure 5.4). By tuning the range of search to within 100 km, the results can be dozens of games, of which only a few are meaningful. For users who do not know the games or the teachers, finding a random meaningful game can sometime be impossible. We believe that the meaningful games should be published as such and may be played by users. An additional extension to “Treasure-HIT” is required in order to locate meaningful games and help people publically find them. This platform could show a map with all the available games that can be played regardless of any pedagogical institution.

**Provide teachers a more in-depth configuration of a game** - Many decisions hardcoded in the environment behavior should be configured by the teacher. For example: we decided that every game, by default, has a competitive nature, and so the delay behavior is always activated. We observed teachers who created activities for their students with no competitive nature, and could not configure the game to not display the delay or configure it to be more suitable to a non-competitive game.

**Improve the sharing mechanism** - Improving the sharing mechanism is required in order to allow teachers to be exposed to pedagogical activities without depending on a fellow teacher to share it. A mechanism that would offer teachers the option to browse other teachers' shared games and adopt and adapt them for their needs.
12 References


http://7scenes.com/platform/


Dunn, J. (2012, 10 17). *The 88 Best iOS Apps For Mobile Learning.* Retrieved 10 19, 2013, from Edudemic - connecting edducation and technology:


13 Appendix

13.1 Elaboration about the tools and technologies in use by the author environment:

13.1.1 HTML5
HTML5 is the fifth revision of HTML. It is used for structuring and presenting web content. HTML5 aims to take a few steps towards the future by providing a good support of multimedia features and at the same time preserve the language readability by humans, browsers, parsers, etc. HTML5 is not a new language; it is an addition to the HTML tags with useful tags and deprecation of unused tags. It attempts to define a single markup language that can be written in HTML syntax. It extends the markup in a way that allows machines to better understand context of a content by adding many new elements, such as `<section>`, `<article>`, `<header>` and `<nav>`. Moreover, it includes the new `<video>`, `<audio>` and `<canvas>` elements, as well as the integration of Scalable Vector Graphics (SVG) content and introduces markup and application programming interfaces (APIs) for complex web applications. (Lubbers, Albers, & Salim, 2011). “Treasure-HIT” author environment uses HTML5 semantic tags and multimedia tags.

13.1.2 JavaScript and jQuery libraries
JavaScript is the most popular client-side scripting language for Web applications (Guarnieri, et al., 2011). JavaScript is an object-oriented language designed in 1995 by Brendan Eich at Netscape to allow non-programmers to extend websites with client-side executable code. (Richards, Lebresne, Burg, & Vitek, 2010). Due to the variety of applications that JavaScript serves and the variety of programming needs, JavaScript Frameworks have been developed in order to facilitate the work of web programmers. jQuery is a fast, small, and feature-rich JavaScript library. It makes HTML document objects (DOM) manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers with a combination of versatility and extensibility (jquery.com, 2014). jQuery is used extensively in every part of the author environment, the features being used the most are the DOM element selection and the Ajax API. An additional jQuery library that uses jQuery and is used in every page of the author environment is “jQuery UI”, which provides many useable cross-browser UI features (widgets). The widgets used in the author environment are: alternative style to the radio buttons and checkboxes, slider, date picker, spinner, progress-bar, dialog and tooltip. “jQuery UI” was the reason that jQuery was selected as the JavaScript framework for the author environment, a few additional libraries were tested, but none of them provided such rich UI widgets as jQuery UI.

13.1.3 AJAX (Asynchronous JavaScript and XML)
Allows the browser to communicate with the web without forcing page refreshes and thus enables client-side richness. AJAX influence on web development was great: many services implemented collaborative software solutions directly on the web like Google Docs, and implemented applications using SAAS (Software As A Service) philosophy, which sometimes compete with common popular desktop software such as Microsoft Office. A popular example of AJAX that can often be seen in “Treasure-HIT” is the autocompletion text field search bar, where a user enters characters into the search field, the client contacts the server for addresses containing these characters, and it displays a list of suggested addresses. This behavior enhances usability, and potentially can decrease the overall bandwidth of network access.
communication, improve interactivity and responsiveness. (Singh A. K., 2012). In the author environment, AJAX is used everywhere, for example: changing the game name or station landmark does not require a whole page reload. Small chunks of data are sent to the “Server API” via AJAX calls, instructing it to add, delete, rewrite, or retrieve data. Then, when the response gets back to the client with chunks of data which were parsed on the server, JavaScript is used to manipulate the DOM (Document Object Model) – adding, changing or removing elements according to the response received from the server. This behavior makes the environment act more like a desktop application and less like a website.

13.1.4 Third party Google's APIs
The author environment was designed to enable the teacher to create games without the necessity of being in the playground physically, the stations should be pinpointed using the web based environment only. This requirement forces the use of a mapping service that has a rich API and functionality. Google Maps is considered as a rich, flexible and accurate third party mapping service that can be used freely. Before a decision was made to use Google Maps API, an additional mapping service named “AtlasCT” was tested but had little functionality compared to Google Maps. Using Google Maps enabled easy embedding of additional Google services like “Google Places” and “Google Street View” that are also used in the author application.

13.2 Elaboration about the tools and technologies being used by the mobile app
13.2.1 “PhoneGap”
“PhoneGap” is an open source framework that enables developers to control the device operating system’s native functionality with a set of web APIs using JavaScript. “PhoneGap” provides a hybrid solution by wrapping a web app in a native container, so it is installable on any device just like a regular application developed using native operating system language. Developers can use HTML and CSS for rendering the UI and JavaScript for the application logic and DOM manipulations. They can then use the “PhoneGap” API which is responsible for the communication with the native operating system. “PhoneGap” basically serves the native container as a layer that allows deploying a web app to any app store which otherwise it would not accept. In 2012 “PhoneGap” announced a new service named “PhoneGap Build” that allows developers to upload their web app code directly to the cloud, compile the app and generate packages of the app for a few operating systems. The app can then be directly downloaded and installed on the device. Currently, “PhoneGap” is able to package applications to the following operating systems: iOS, Android, Windows Phone, BlackBerry, WebOS, Symbian Ubuntu and firefox OS. (PhoneGap, 2014). Thus, it was decided that the “Treasure-HIT” mobile app will be implemented as a mobile web app and will be compiled using the “PhoneGap” Build service. The next section will cover the JavaScript framework selected for implementing the app and its client side architecture.

13.2.2 Mobile App JavaScript framework
Like many HTML based applications “Treasure-HIT” mobile app uses a JavaScript framework, which similarly to a server side framework proposes easy implementation of a certain design pattern and best practices logic. Three JS framework were candidates to be the framework for the “Treasure-HIT” mobile app: Backbone,
Ember and AngularJS. Table 5.1 highlights the differences between the frameworks’ features:

<table>
<thead>
<tr>
<th></th>
<th>Backbone</th>
<th>Ember</th>
<th>AngularJS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Presentation pattern</td>
<td>MV + VC</td>
<td>Pure MVC</td>
<td>MV* (The * make the design pattern flexible to the developer flavor for example MVVM, MVC, ext.)</td>
</tr>
<tr>
<td>Bi directional Binding</td>
<td>No</td>
<td>Yes (But, requires handlebars )</td>
<td>Yes</td>
</tr>
<tr>
<td>Routing</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>JavaScript library</td>
<td>underscore.js</td>
<td>Handlebar.js and jQuery 1.7 or higher.</td>
<td>None</td>
</tr>
<tr>
<td>Compatible with other frameworks</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Additional Features</td>
<td></td>
<td></td>
<td>DI, mocking, directives, services, watch expressions and its binding, HTML5 validations</td>
</tr>
</tbody>
</table>

Table 13.1 Features comparison of Backbone Ember and AngularJS

According to Table 13.1 AngularJS offers the highest flexibility compared to Backbone and Ember. The growing popularity and community of AngularJS were a solid convention of this framework, and thus it was selected. The following section briefly introduces AngularJS framework and its properties.

13.2.2.1 AngularJS
AngularJS framework offers MV* client-side development, written in JavaScript. It enhances developers writing a single page web application which fits well on mobile. AngularJS offers many features but it is most useful when used to write CRUD (Create Read Update Delete) based applications. AngularJS offers a novel code styling to handle templates by extending the HTML and offers bi-directional data binding which makes the framework powerful and easy to use. Developers who use AngularJS report on dramatic reduction in the number of lines of code compared to other popular JavaScript frameworks. AngularJS developers invested great effort to support testability and encourage Test Driven Development (TDD). Today, AngularJS is used in complex projects involving many pages and client side logic. AngularJS, as any other JavaScript framework, has its own set of idioms, patterns, and best practices that have been uncovered by the community, based on their collective experiences (Kozlowski & Darwin, 2013). AngularJS has very solid engineering practices, its template system is unique in the following aspects:

- The template engine language is based on HTML.
- AngularJS is capable of tracking user actions, browser events, and model changes by figuring out when and which templates to refresh, so it does not require any DOM refresh when the data changes.
- AngularJS comes with an impressive set of extensible components (directives) which are capable of “teaching” the browser how to interpret new HTML tags, attributes or classes.

AngularJS also has useful additional features such as: dependency injection (DI). The built-in support for DI makes unit-testing easy, even for a large scale application, and lazy loading potential. (Kozlowski & Darwin, 2013).