

Alternate *ACM SIG* Proceedings Paper in LaTeX Format*

Geng Fu[†]
INI
Carnegie Mellon University
gengf@andrew.cmu.edu

Vinay Kumar Vavili[‡]
INI
Carnegie Mellon University
vvavili@andrew.cmu.edu

Yalei Song[§]
ECE
Carnegie Mellon University
yaleis@andrew.cmu.edu

1. INTRODUCTION

Public displays are ubiquitous. They can be seen in many public locations including malls, schools, and coffee shops. Public displays share these common characteristics:

- They appear in public places (as mentioned above)
- Their locations are usually fixed (not portable)
- They are usually large in size
- People pass by or gather around the display
- There are limited interfaces. Keyboards, touch screens are usually not provided
- No internal memory available (it is just a screen)

With those characteristics, public displays are often being used as a screen to display content. It could be static content like posters, images and Ads; or it could be dynamics content like news channels, trends or music videos. However, public displays are not necessarily aware of its surroundings. Most importantly, displays don't respond to the presence of people. In other words, public displays are not interactive. For instance, a public display in a mall could show music videos of Justin Bieber all day long, even though none of the people in the mall is a Belieber¹.

2. OBJECTIVES

The objectives of this project are to make public displays interactive. We set different scenarios, such as cocktail parties where people mostly mingle with others; or a coffee shop where people standing in line waiting to order. Specifically, we hope to achieve two goals. First, we would like to encourage interaction between people and display, and also to make displays and people aware of each other's presence. Second, we would like to foster interaction between people and encourage discussion.

¹A huge fan of Justin Bieber

In order to engage people, we started with designing an application that should attract people's attention. Many ideas were emerged, such as developing games or tools that will allow people to participate in activities. We believe that game is an efficient way to gather people's attention in a short period of time, hopefully in a pleasant way also.

The objectives of the game are very simple. It allows people to interact with display and promote interaction between people quickly. Since we set up our game in scenarios like cocktail parties and coffee shops, where people are usually occupied, we want to make sure that players can join the anytime. Moreover, players shall not feel obligated to stay in the game; she/he should be able to leave the game anytime. In additional, the game should not be too intellectual or difficult to play.

We conducted a series of brainstorming and came up with several game ideas, such as action game (e.g., darts, basketball), gambling game (e.g., roulette), word puzzle game and trivia game. We later realized that action requires too much commitment to the game and space for movement, which doesn't suit our scenarios; puzzle game involves too much intelligence and it is time consuming; whereas gambling game is basically random guess. As a result, developing trivia game(s) was a favorable idea.

We conducted a series of brainstorming and came up with several game ideas, such as action game (e.g., darts, basketball), gambling game (e.g., roulette), word puzzle game and trivia game. We later realized that action requires too much commitment to the game and space for movement, which doesn't suit our scenarios; puzzle game involves too much intelligence and it is time consuming; whereas gambling game is basically random guess. As a result, developing trivia game(s) was a favorable idea.

3. GAME IDEAS

To generate a ?sourceful? question pool for trivia game, we utilized Google APIs, which provide ample useful information.

3.1 Battle of Cities

Google radar search (part of Google Places APIs) provides number of a certain type of places (e.g., restaurant, bakery) in a given area. With this information we can develop a trivia game that is based on numbers of places in different cities. Here is a sample questions: Which city has more bars, New York City or Shanghai?

We select more than 10 high-profile cities in both the eastern and western atmosphere and 80 place types to generate

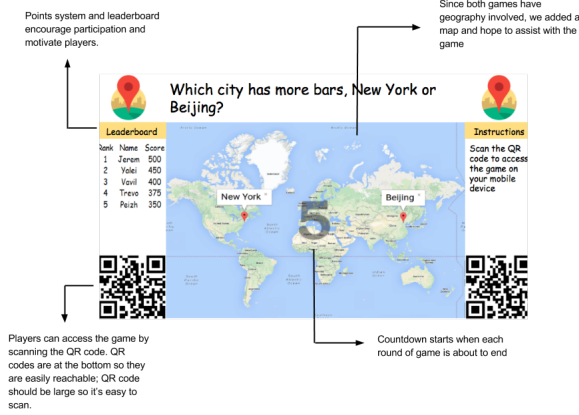


Figure 1: Preliminary Design

questions.

3.2 Translator Dilemma

Google translate game idea was emerged when browsing a website called Translation-telephone, which is similar as the traditional ?Telephone? game, where people pass a message around a circle and see how it turns out at the end [1. Reference the website]. In the translation-telephone, a message gets translated to random languages and back to original language.

We thought it is an interesting and wanted to develop a game based on it. Google Translate ? one of the most widely used translate engine, is a great tool to use in this game. In the same way as translation-telephone, we select random collection of languages and use Google translate API to generate translations. In addition, we have two collections (paths) of languages and provide one final translation. In the end, players will get to choose which path the final translation is coming from.

4. PRELIMINARY DESIGN

5. DESIGN GUIDELINES

Table 1 summarizes the design guidelines we hope will help in the design of games for interactive public displays.

The design guidelines were formulated based on user feedback and results from a user study that was conducted as part of the live deployment. We also used the results of the user study to evaluate if we met the objectives of the game. The study responses were scale responses scaled from 1 to 10. We normalized the scaled responses to standard[1]. In simple terms, the percentage calculated from the z-score represents the percentage of responses that are greater than or equal to the benchmark selected during the z-score calculation. We have selected a benchmark of 8 for the scale from 1 - 10.

“The questions should be intuitive or easy to understand”

“The answers should be correct or logical”

The first question we asked participants was, “How easy is it for you to understand the objective of the game?”. Fig-

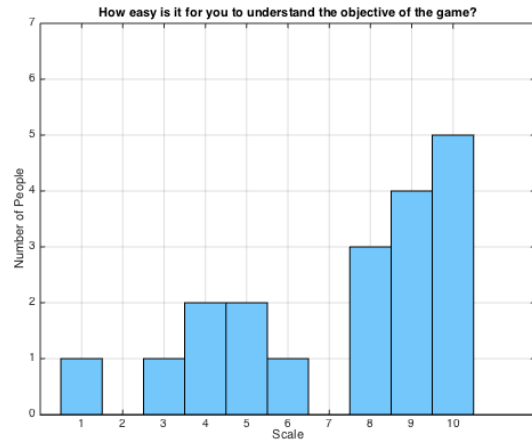


Figure 2: Battle of Cities - How easy is it for you to understand the objective of the game?

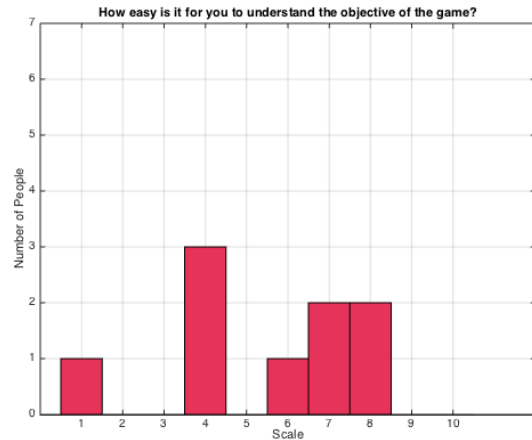


Figure 3: Translator Dilemma - How easy is it for you to understand the objective of the game?

ure 2 shows the the responses for the game Battle of Cities and Figure 3 is the responses for Translator?s Dilemma.

37.98% of the participants felt that the Battle of Cities game was easy to understand. This was primarily because the question was quick to read and easy to understand. There was higher participation in the game as a consequence. Compared to Battle of Cities, only 13.85% of the participants felt that Translator?s Dilemma game was easy to understand. As mentioned before Translator?s Dilemma game is based on the ?Telephone? game. During the user study we noticed that it was very hard for participants to understand the objective of the game as most of them had not played the “Telephone” game before. The question text was also long and complex in this case.

The following question we asked participants was, “Did you discuss with others during the game?”. Figure ?? shows the the responses for the game Battle of Cities and Figure ?? is the responses for Translator?s Dilemma.

6. LESSONS

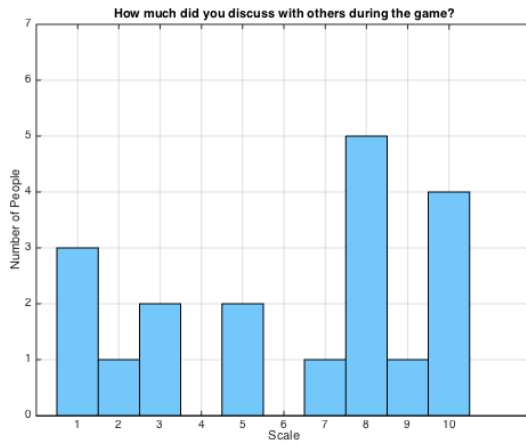


Figure 4: Battle of Cities - Did you discuss with others during the game?



Figure 5: Translator Dilemma - Did you discuss with others during the game?

Do not underestimate the efforts to implement web page on mobile devices. The challenge of making a fluid UI is to deal with different resolutions of screens. A basic solution is using relative units such as em or percent(%) instead of pixels(px). However, in deployment we found out that since the dimensions of the screen itself is different, the UI is not consistent across all screens. To handle the problem of different screen dimensions we used viewport, which arranges elements by the actual size of the screen. Users may hold the device in different orientations too. We used CSS media query to detect the orientation of the devices and then display the elements on the screen. Even though we spent a lot of efforts in device UI, the UI is not consistent on old devices.

Public display should be self-configurable. Initially, the URL of the backend server was hard-coded to ensure users can connect to backend servers. And the QR code was static. Consequently, when the backend server is deployed in another location, all those static components need to be configured before deployment. A solution to ease deployment is to leverage the information that web browser is aware of where the page is loaded from.

Long-duration deployment discloses bugs. The potential bug isn't seen until long duration deployment. In deployment, the problems may arise from unreliable dependencies, race conditions between multiple players operating at the same time etc.

Fault-tolerance at public display. In the deployment, we often don't have the direct control over the public display. The display may lose the connection with the server, in which situation the public display should be able to reconnect to backend server. If bugs are found in the frontend page, the display needs to reload the web page. A script which periodically reloads the web page is helpful in deployment.

Backend server should be publicly addressable. A common mistake in deployment is to deploy the backend server with a private IP address. Due to the NAT problem, only the devices that are in the same network can access the game. Therefore, a public addressable interface is essential to the backend server.

Alternative to allow device to access the game. In the deployment, we assumed that users will scan the QR code to find the game. It turned out that most of devices were not equipped with QR code scanner and players are reluctant to install one. Alternative way such as tiny URL is helpful.

Carefully choose the place to conduct user study. The place of deployment was in a student common area. Students mostly were in a rush. The unsolicited involvement was low.

6.1 References

Generated by bibtex from your .bib file. Run latex, then bibtex, then latex twice (to resolve references) to create the .bbl file. Insert that .bbl file into the .tex source file and comment out the command \sigproc.

7. ADDITIONAL AUTHORS

Additional authors: John Smith (The Thørväld Group, email: jsmith@affiliation.org) and Julius P. Kumquat (The Kumquat Consortium, email: jpkumquat@consortium.net).

8. REFERENCES

- [1] Standard score, Feb 2015.
- [2] Mic Bowman, Saumya K. Debray, and Larry L. Peterson. Reasoning about naming systems. *ACM Trans. Program. Lang. Syst.*, 15(5):795–825, November 1993.
- [3] Malcolm Clark. Post congress tristesse. In *TeX90 Conference Proceedings*, pages 84–89. TeX Users Group, March 1991.

Table 1: Design Guidelines Summary

| Aspect | Lesson |
|-------------------|--|
| Content | The questions should be intuitive or easy to understand The answers should be correct or logical They are usually large in size People pass by or gather around the display |
| Player Perception | Attention span of users is limited. The timing between questions should be short Make sure the player knows the status of the game Players should be motivated |