## Large scale seeking game using android phones

Group # 4

Colby Ansel Horn (20742291) - cahorn@uwaterloo.ca

Prerna Angrish (20737665) - pangrish@uwaterloo.ca

Yaron Yaacov Milwid (20420200) - yaron.milwid@uwaterloo.ca

Haoyuan Zhang (20659128) - h524zhan@uwaterloo.ca

### Part 1: Introduction

#### 1.1 Overview Of Application

This application is an android-based ecosystem of multi-player location based games. Users will be able to play different styles of hide-and-seek over a large area. Since the area is large, their phones will provide them hints as to their targets’ locations. The phone system will track whether or not players have been found. A tag requires human interaction.

#### 1.2 Interest Of Application

The application is interesting to users since most people enjoy playing games, and multiplayer games such as this serve multiple purposes. For example, people often use these games to meet people or as a fun, interactive way of spending time with friends.

The application is interesting architecturally since information must be repeatedly shared between many devices. This sharing requires consideration of many factors in order to ensure a high quality of service. For example, the amount and frequency of information shared must ensure seamless operation, without bloating the channels. Additionally, different architectures affect privacy in different ways. Finally, the architecture must be designed in such a way as to minimize the impact of network issues, requiring significant decisions related to CAP theorem1. For example, based on the fallacies of distributed computing2, fault tolerance is necessary. A BASE1 system is likely preferable to an ACID1 system for user experience; however, this could create short term issues for comparing “tag counts”.

#### 1.3 Suitability of Technology

Since users will be constantly moving, the game needs to be on a portable platform, implying that a smaller device is preferable to a computer. Additionally, since the location needs to be broadcast, this rules out a website. Because there needs to be a constant network connection, a cell phone is superior to a tablet since many cell phones have a data connection for when WiFi connections are lost. As a result of these three considerations, cell phones are the optimal platform for this game. Since Android has the largest market share in the smartphone market3, it makes the most sense to initially develop a smartphone game for Android.

### 

### 

### Part 2: Properties

#### 2.1 Functional Properties

1. Allow users to create and manage a *player account*.
2. Allow players to create *games*. Game creation includes:
   1. Listing participating players or otherwise defining how players can join. Participating players can be:
      1. Listed individually.
      2. Undefined; the game can be open to all players who want to participate up to a specified player limit.
   2. Selecting various rules of the game. These rules primarily fall into 2 categories:
      1. Selecting which players will have their locations reported, or be *visible*, to which other players. For example, if players wish to play a traditional game of hide-and-seek4, the player designated as the seeker would be set to receive hints as to all other players locations, while those other players would receive no such hints. Conversely, if players wished to play a game in the style of assassin5, each player would be set to receive hints as to the location of a single other player that the first is designated to find.
      2. Selecting the result of tagging another player. The above two variants illustrate the effect of this setting. If playing a game of hide-and-seek, a tag would be set to simply remove the tagged player from the current game. However, if playing the assassin variant, a tag would additionally be set to select a new target for the tagging player, and modify the player visibility accordingly.
   3. Defining an *area of play*. Players who venture outside this area are considered temporarily inactive until they reenter the area of play. While inactive, players neither send nor receive location data regarding their or other players’ locations and cannot be tagged. The application tracks how long a player is inactive and displays this at the end of a game.
   4. Defining the time period over which the game will take place.
3. Provide each player hints as to the location of other players, as per the game’s configured visibility rules.
4. Allow players to record tagging another player, and take the appropriate action for a particular game.

#### 2.2 Use Cases

##### 2.2.1 University Orientation

Alice is a resident assistant at her university. She is tasked with planning an orientation social event for new students. However, the students are housed in multiple separate building spread across campus. She creates a game of assassin, with the location boundaries set to the university campus as well as the residences, and the time limit set to one week. She then navigates through the game interface and invites the new students to join the game.

Once the game has started, students use location hints to approach their targets. Once they are near their targets, students need to talk to and get to know other students in order to find their targets. After tagging their targets, students inherit the targets’ targets and need to repeat this activity for the new targets. In this way, students find, play with, and get to know other students that they might not have interacted with otherwise. Additionally, students get to explore the university campus and residences. The game’s tag tracking allows Alice to easily keep track of the game’s progress and provide a celebratory prize to the winner, either the final player, or the player with the most tags at the end of gameplay.

##### 2.2.2 Day in the Park

Bob and a group of five of his friends are bored one sunday. They decide the weather is so nice that they would like to do something outside. They decide to go to Waterloo Park and play a game of hide-and-seek. Bob creates a new game and sets the location to the boundaries of Waterloo Park, he also sets the time limit to one hour. He then invites all five of his friends to join the game and they spread out in the park. The game randomly chooses one of the players (Charlie) to be ‘it’. Charlie follows the location hints to find the general locations of his friends in the 111 acre large park6. He then has to find them within this smaller location.

Charlie only manages to find four of his five friends before he runs out of time and feels dejected. However, when looking at game statistics, the friends find that Alice accidentally left the park for the entire game. As such, based on their group norms, the group decides that the game was not fair to Charlie; and therefore, Alice has lost the game. As a result of losing the game, Alice has to buy ice cream for all of her friends. In this way, the friends spend an enjoyable day in the park, get to explore a wonderful regional resource, and experience some healthy competition.

#### 2.3 Non-Functional Properties

1. As gameplay requires a real-time, continuous flow of data between players’ handheld devices and coordinating servers, data transmission efficiency is of prime concern. One way that efficiency will be attained is by minimizing data transmissions by both clients and servers:
   1. Limit the frequency of updates by only reporting significant location changes.
   2. Limit the size of transmissions by only providing clients with a subset of player location data, specified by a game’s visibility rules.
2. In order to support gameplay in a multitude of potentially large areas, and with player movement as a core mechanic, robustness to transmission failure is also of key importance. All devices will store sufficient information to continue to provide gameplay through short transmission failures and gracefully handle extended transmission failures.
3. To provide an ecosystem of location-based games, rather than a single mode of play, the backing data infrastructure will be sufficiently flexible to evolve to support multiple game modes. One way that this flexibility will be achieved is, rather than hard coding rules of visibility for each game type, each game will store a unique set of visibility rules in a *visibility matrix* that:
   1. Determines the circumstances under which each player will receive hints as to other players’ locations.
   2. Is easily configured before gameplay to support a particular game type and easily reconfigured during gameplay for joining players or tagging actions.

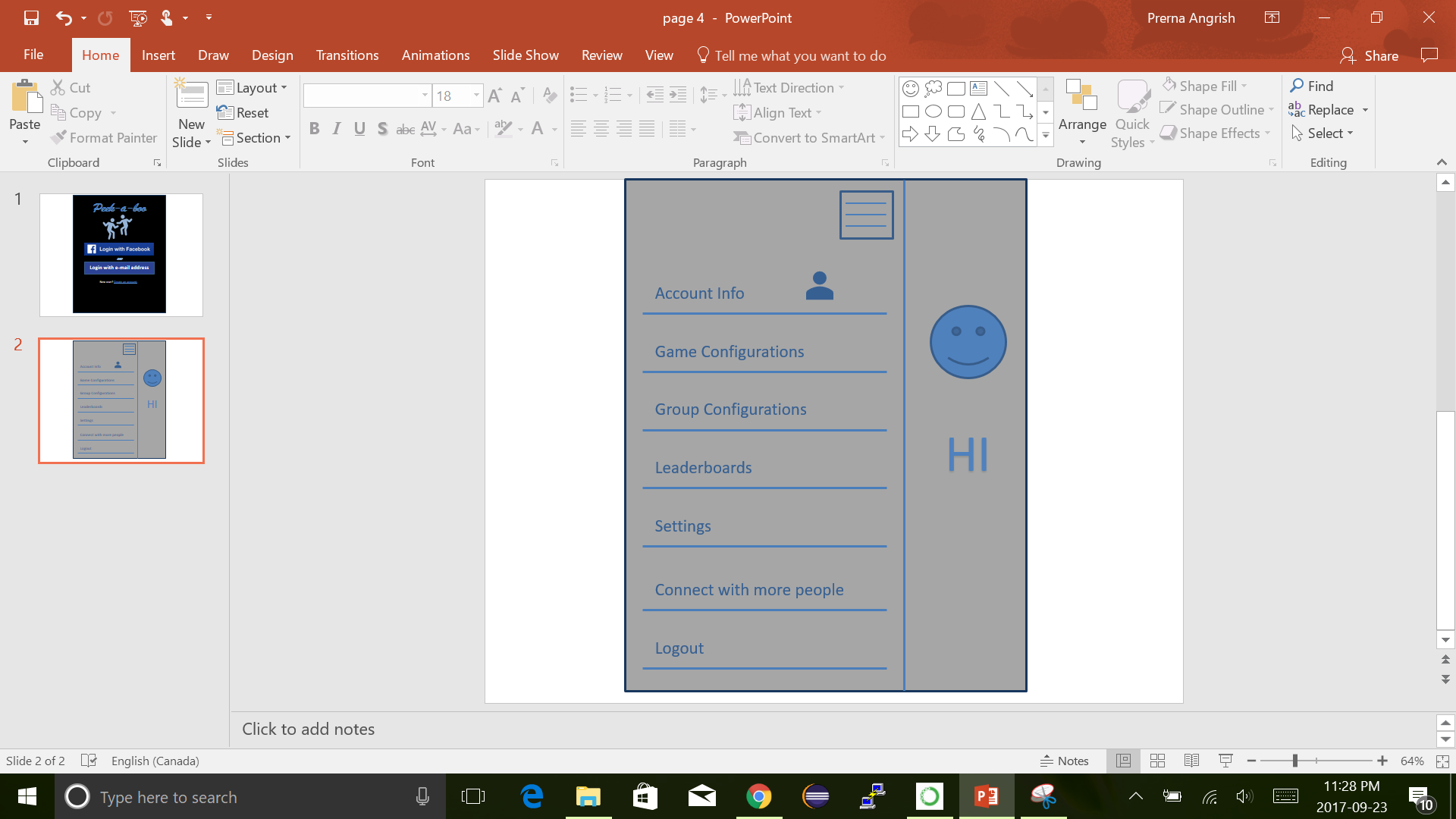
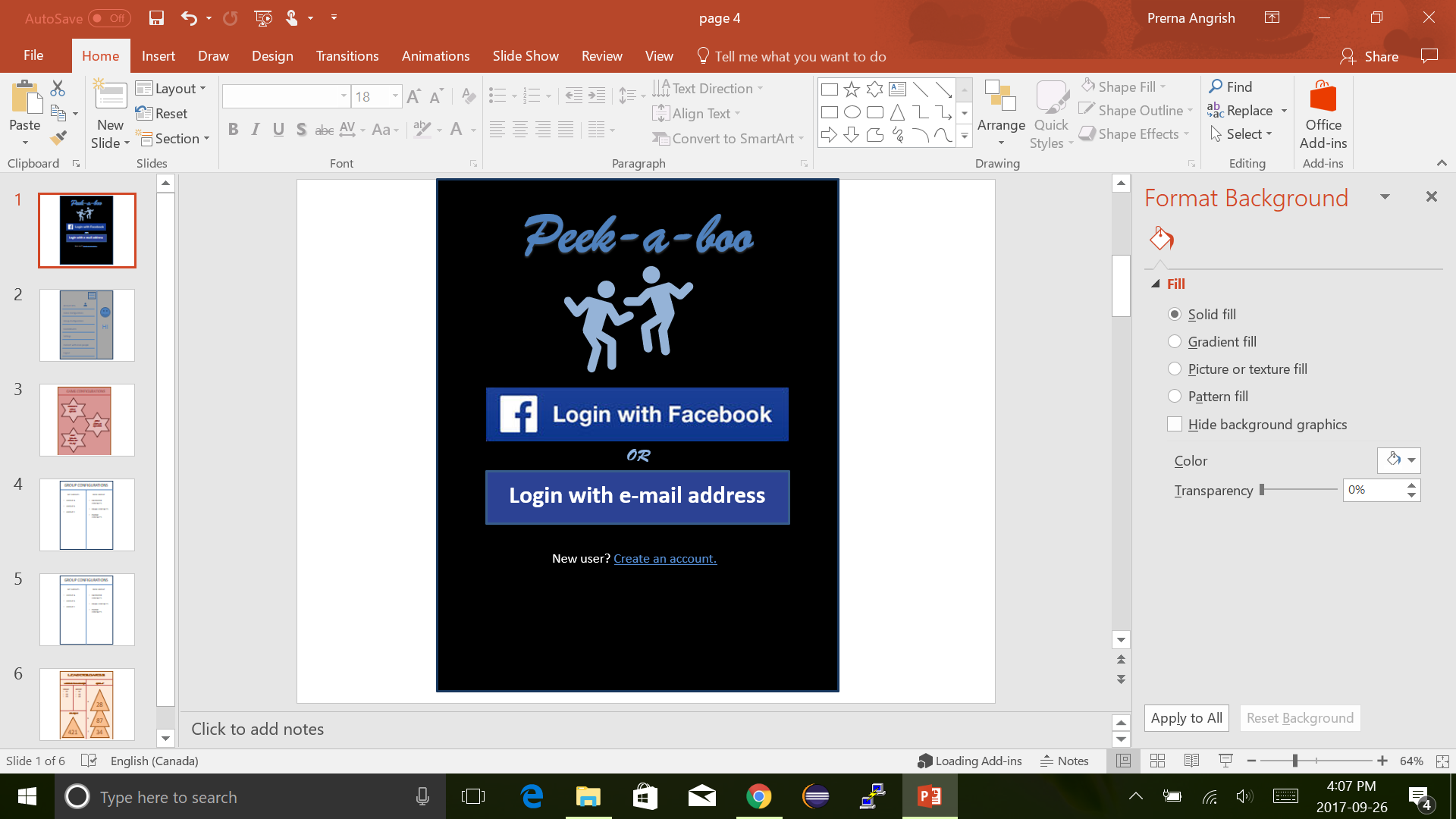
#### 2.4 Possible Extensions

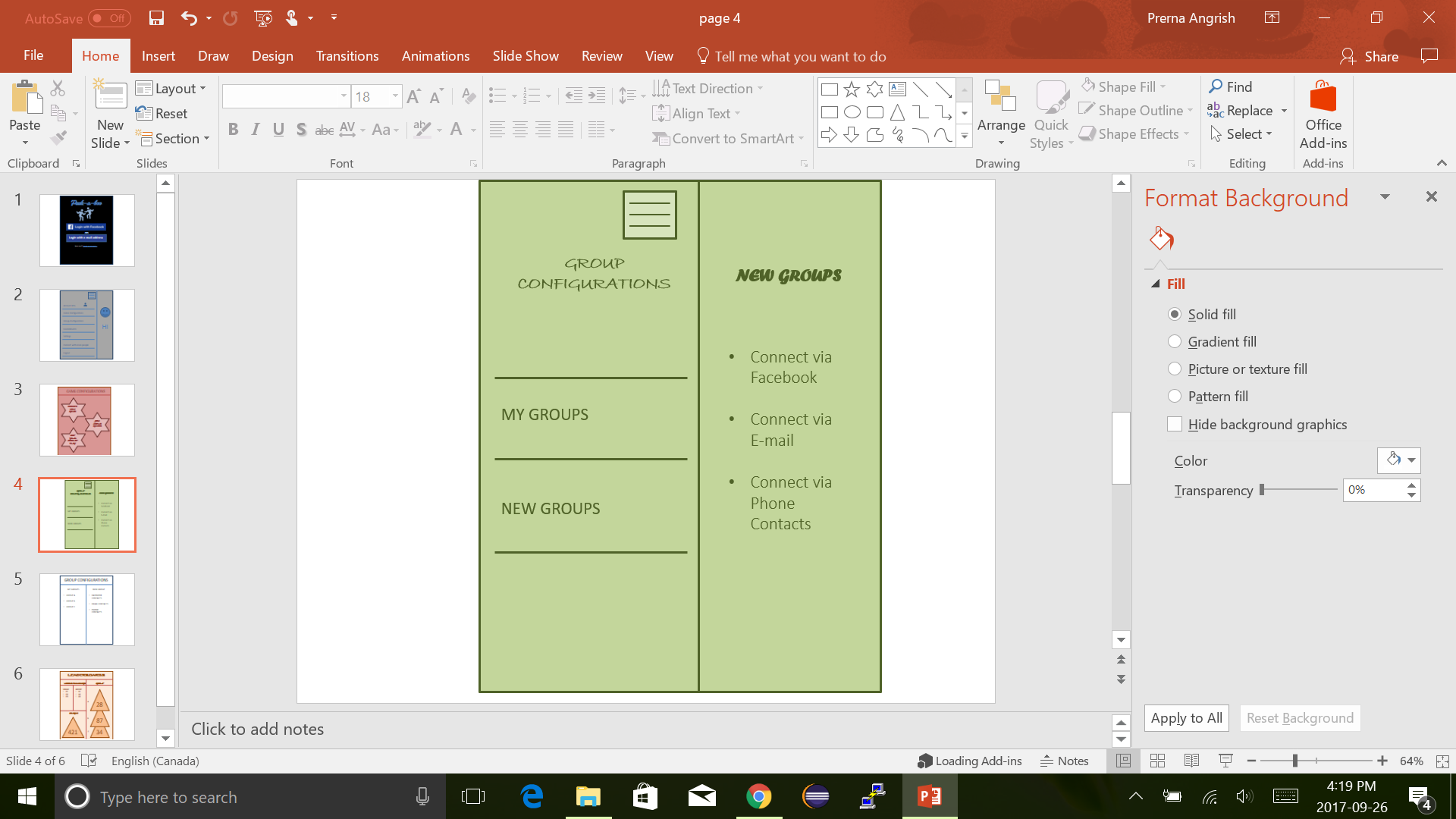
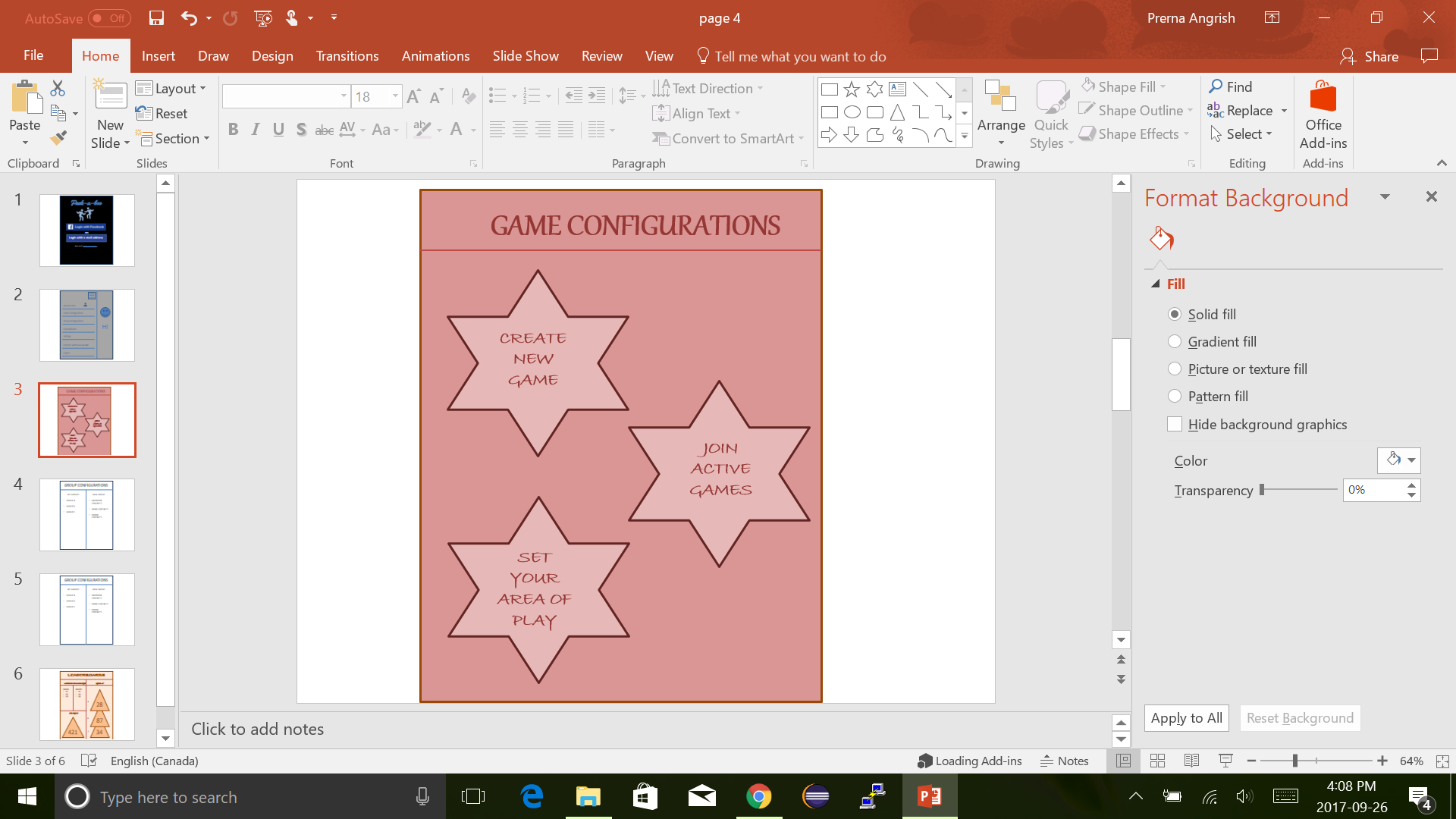
1. Link pictures to user accounts so that players can better identify each other and play together.
2. Add a leaderboard showing players who have garnered the most tags, and various other statistics.
3. Provide *ad hoc* control over visibility rules to allow novel, user-defined game types.
4. Provide players a way to manage *playing groups*. These groups can be constructed by individually identifying each player’s account or by importing group information from other communication or social media contact lists.

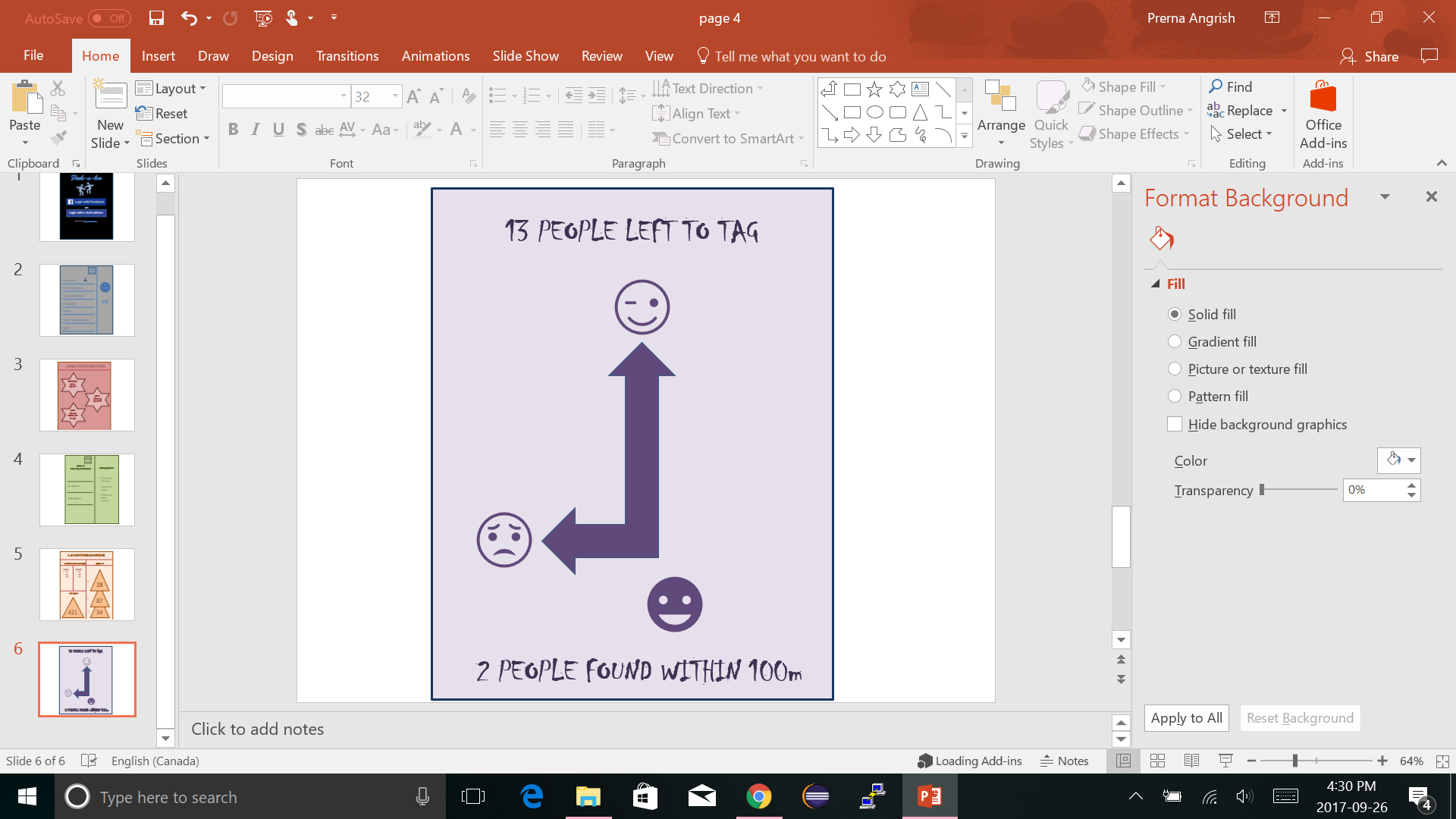
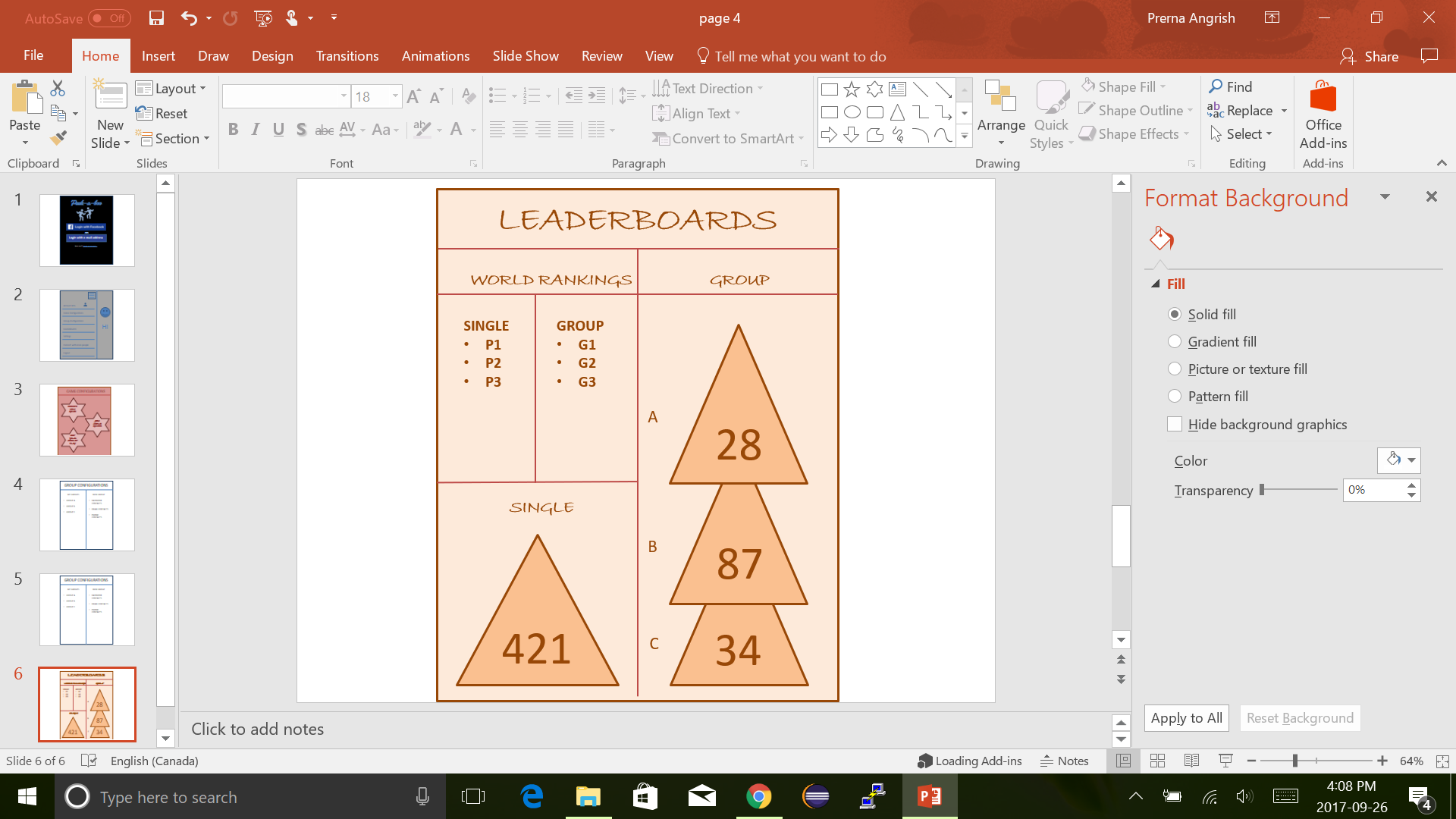
### 

### 

### Part 3: Graphical Depictions of Application







### References

[1]E. Brewer, "CAP twelve years later: How the "rules" have changed", *Computer*, vol. 45, no. 2, pp. 23-29, 2012.

[2]A. Rotem-Gal-Oz, "Fallacies of distributed computing explained", 2006.

[3]o. Global smartphone sales to end users from 1st quarter 2009 to 1st quarter 2017, "Smartphone sales by OS worldwide 2009-2017 | Statistic", *Statista*, 2017. [Online]. Available: http://www.statista.com/statistics/266219/global-smartphone-sales-since-1st-quarter-2009-by-operating-system/. [Accessed: 20- Sep- 2017].

[4]"Hide-and-seek", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Hide-and-seek. [Accessed: 26- Sep- 2017].

[5]"Assassin (game)", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Assassin\_(game). [Accessed: 26- Sep- 2017].

[6]"Waterloo Park", *Waterloo.ca*, 2017. [Online]. Available: http://www.waterloo.ca/en/gettingactive/waterloopark.asp. [Accessed: 26- Sep- 2017].