Finger Pop

**Background Survey:**

This section includes the problems as we perceive and faced during the development of this game and the algorithms that were researched to solve such problems.

**Game Format:**

The Finger Pop game is essentially a hangman-styled word guessing game, where multiple users try to guess the letters of a given word in real time, simultaneously. The game is a client-server model based design and intended to work on modern browsers with most of the work offloaded to client side (Thick clients). The game involves clients joining a session, and with enough number of users, the server initiates the game with a random word, presented to all users in current session. All users can simultaneously try to guess the letters of the word while only the users who have guessed the correct letters first gets points.

**Problems:**

Since this is a multiplayer game, we have to deal with the issue of concurrency and mutual exclusion **[[[expand on both here a bit ---Only on what is mutual exclusion and what is concurrency]]]**

**Mutual Exclusion:**

Where mutual exclusion can happen in this game, and how this is an issue,

**Concurrency:**

Where concurrency can happen, how it needs to be handled. This could be written for a page, both topics combined.

**Deadlocks:**

A part of concurrency, add to the previous section and expand a bit, in context to this game.

**Algorithms:**

**Mutual Exclusion:**

1. **Token Manager (half page)**
2. **Central Server algorithm (ha;f page)**
3. **Few Generic Algorithm(1 page)**

**Concurrency and Deadlock**

1. **Lock (Half page)**
2. **Timeouts (Half page)**
3. **Other methods of lock (two phase lock, etc) and algorithms (1 Page)**