Kaya Peng, Lloyd Deng, Tony Moon
CSS 432
Project Proposal
October 3rd, 2019

1. Team Cookies

2. Member information
   a. Kaya, Computer Science and Software Engineering major
   b. Lloyd, Computer Science and Software Engineering major
   c. Tony, Computer Science and Software Engineering major

3. Project description
   - We plan to build a Sea Battle game to be played on the console. Each player has their own board of empty squares, which the server will randomly populate with 8 ships. They can also see their opponent's board, but not the ships on that board. Players alternate turns entering locations they believe the opponent's ships reside, and will receive confirmation and a board marker from the server for each guess. Before entering their location, a player may also enter a message they wish to be seen by the other player. Once a player has correctly guessed all enemy ship locations, the game ends, and that player wins the game.
     - Two-player board game
     - Text-based
     - C++
     - Server-client architecture
   - The game and it's code will reside on the server. Players will send messages and attack instructions (a1, b2, c3) in their console, and receive back the game state from the server. In this manner, players are able to interact with the game.

Example:

```
uw1-320-10:~$ here?
> d2
##################

player:  opponent:
 abcdef   abcdef
1□□□■□□  1■□□□□□
2○○○●■□  2●●●●□□
3□□□■□□  3□□□□■□
4□□□□○□  4□□□□□□
5○●□□○□  5□■□□□□
6○●□□○□  6□□□□□□

turn 7: hit and
sink!
sent message: here?
waiting for
```

```
opponent.
##################
```

4. Tentative schedule
   Week 1: Prepare game assets (graphics, console messages, etc.)
   Week 2: Study networking fundamentals
   Week 3: Practice networking in C++
   Week 4: Practice server-client architecture in C++
   Week 5: Create “mock” game allowing players to send messages over the server
   Week 6: Begin Sea Battle development on top of mock game
   Week 7: Unit tests, integration tests
   Week 8: Submit project, implement extra credit material if ahead of schedule

5. Tentative tasks for each member
   a. Kaya
      Server-client implementations, unit tests.

   b. Lloyd
      Game assets, Sea Battle development, integration tests, server-client implementations.

   c. Tony
      Server-client implementations, create mock game.

6. Repository URL
   https://github.com/kpeng09/CSS432_19A_TeamCookies