Project 4 – Deployment Plan

The initial deployment of our game will be as a browser-based multiplayer strategy game. A central server will determine the starting map conditions, and then send that information to the client-side of each player. Player one will be given a turn, and at the end of their turn, the map status will be sent back to the server and then to the other player. Player 2 then makes their turn and the cycle repeats. The centralized server would allow for more reliable anti-cheat and connectivity.

The anticipated market includes strategy-game enthusiasts 6 years and older. While the game has a robust combat system, the complexities are largely hidden from the players, making it widely accessible, easy to learn, and hard to master. Our choice of graphics – an SVG map with military symbols in place of typical skeuomorphic troop icons – provides a cartoonish charm that ensures it’s appropriate for children despite being a wargame. Simple combat animations add a visual flair without being graphic, preserving that charm. Adding additional mechanics (regional fortifications, morale system, additional troop types, etc) to allow for detailed, complex strategizing would help us capture the adult segment of the strategy-game market.

We would first need to implement servers, anti-cheat, and a matchmaking system. This would require either salary costs for us to continue the game’s development, or to hire other software engineers to handle these tasks for us. We expect that a team of 3-5 developers would be able to complete the game in 1-2 months. Once complete, servers would be purchased from a vendor such as DigitalOcean or AWS. From the players laptop or PC they will get sent a starting condition or starting screen to play the game. Initially, we could start with a single DO droplet ($5/month) and a cheap domain name (we could presumably find something esoteric enough that we wouldn’t need to pay more than $10-$15/year from a vendor like NameCheap). Based on the market size and our networking solution, we would scale either horizontally, vertically, or both, to serve a larger player base; either way, we will need to lock in with a vendor that allows elastic scaling of resources, and invest in the development of a high-performance networking solution.

Because we are not deploying on an app store or game marketplace (e.g. Xbox live / Microsoft store), we can avoid a lot of traditional costs-of-entry that would make this sort of game more expensive. This makes our monetization plan much simpler: to monetize, we may either add advertisements to the game webpage, or implement an account system to play (similar to Chess.com). In the latter case, we would also assure players that their accounts would give them access to future games from our development team in order to justify some cost of entry / subscription fee (perhaps $10 one-time, or $2/month recurring). To keep with the lean-and-mean model, we would avoid attending conventions and conferences, instead focusing on low-cost, high-impact online campaigns on sites like Reddit and HackerNews (ycombinator). Depending on anticipated revenue, we may also start ad campaigns via Google Ads, Bing, Facebook, et cetera.