Thesis: Cloudflare Workers for Gaming can help the large pool of indie game developers transition single-player games to more scalable multiplayer functionality by packaging development, security and hosting services.

The market:

- Dominated by AAA (games developed by mid-sized to large publishers)
    - Large budgets and infrastructure support from institutional partners
    - Not an attractive market for Cloudflare
- Indie (individual) games
    - Games developed by individuals or small groups of developers
    - Generally, they have low funding and lack of institutional support
    - Monetization
        - One of the largest and most common problems for indie developers
            - If the player base isn't large enough, the designers often try to create a freemium model that can drive some users away, such as required in-game purchases
        - Multiplayer application allows developers to avoid resorting to disadvantageous monetization models through an increased potential player base and cheaper client acquisition costs
            - Generate the majority of revenue from the game purchases

The user / decision-maker:

- Indie developer who wants to gain access to Cloudflare for Workers base functionality and is also willing to pay Cloudflare to handle the headache of the backend to enable multiplayer functionality
- Combination of game development, securitization, and backend services all in one place
- Indie game developers often don't have the manpower or money to handle the backend necessary to make their games multiplayer
- Most games are single-player and some have a co-op hosting option
    - Co-op is almost always required to be hosted by the player
        - It is expensive and difficult for the developer to handle hosting the servers

Production changes/additions:

- Offer a beta version for developers to test features
- Add Cloudflare hosting capacity to Cloudflare Workers
- Add analytic tools for developers to measure user interest, playtime, etc

- Ensure Cloudflare Workers for Gaming supports all the most common indie coding languages

Distribution and Revenue model

- Focus on existing games that want to expand to multiplayer
    - Demonstrate the benefit for existing games to entice full development on Workers
- Use a freemium model to allow developers free access up to a limit before charging incrementally based on their total usage

Further research:

- Determine roughly how many indie developers are handling the multiplayer functionality on their own or how much they are paying for a service to do it for them
    - Need to determine the percentage of indie developers would be interested in this product
- Estimate what the premium for an all-inclusive development, security, and backend support product would be
- Figure out the most common coding languages for Indie games and how best to make development easy for them
- Figure out the cost to Cloudflare to serve these markets

Metrics of Success:

- The percentage of the indie developers who fit the user description who use the product
    - Market share compared to competitors
- Profitability
    - Whether it is profitable and if so, how the margins compare to other Cloudflare products

Potential problems:

- Potential competitive response from competitive hosting services, such as AWS GameLift
- Indie gamers code in a variety of languages which might require the site to support more languages, although with some research we can determine which languages will provide the greatest market opportunity
- Whether it is profitable to provide these hosting services
    - Whether the developers can afford to pay more than the cost to Cloudflare