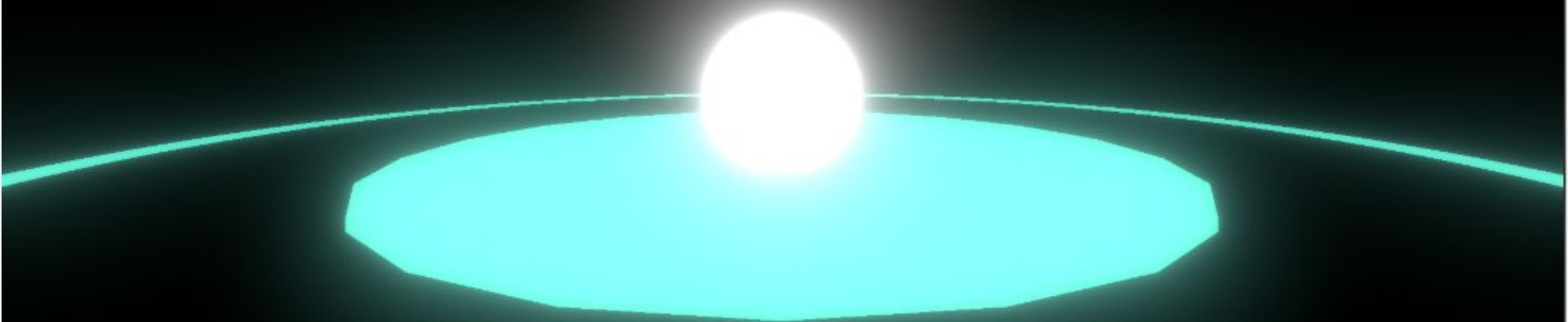


Direction Bowl



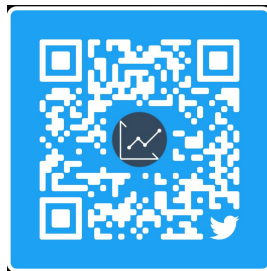


自己紹介

しましゅん
鈴鹿高専総合イノベーション工学専攻 2年

担当
Unityクライアント ネットワークマネージャー

Twitter: konnyaku256
GitHub: konnyaku256



やましゅー

東京電機大学 情報メディア学科 4年

担当
Unityクライアント ゲームロジック

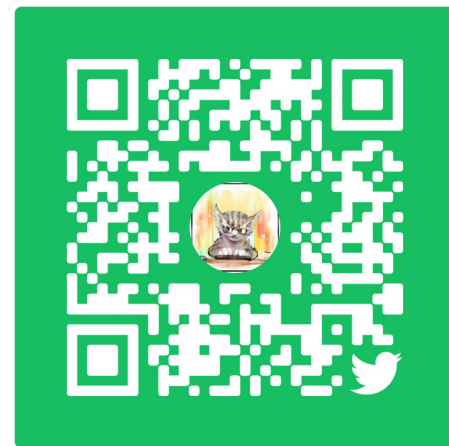
Twitter @AblerBiri
GitHub Gamu2059



たけちゃ

担当
サーバーサイド・インフラ

Twitter: take_btc
Github: BambooTuna

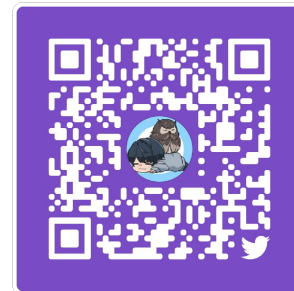


あくび

中京大学工学部情報工学科 4年

担当
サーバサイド

Twitter: yawn_yawn_yawn_
Github: yawn-yawn-yawn





作ろうと思ったきっかけ

これに注目

きっかけ

ルール

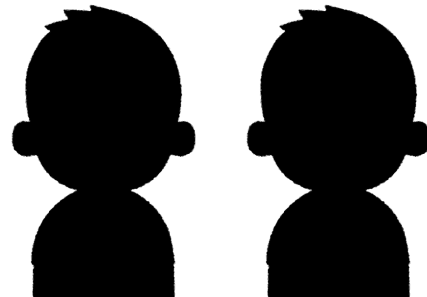
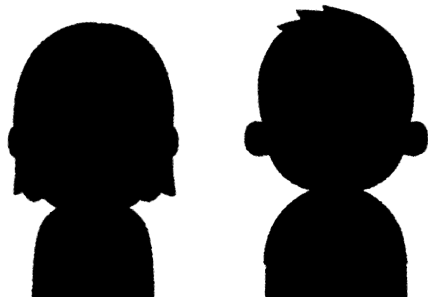
といっても、ルールはほとんどありません。

5/6 15時になったら発表会があるので、それまでに作りたいものを作りましょう💪（コアタイム等も定めません）

テーマは「お家で楽しめるなにか」です。

アイデアに困ったときは [#brainstorm](#) も合わせてご利用ください...！

- Unityだしリアルタイム通信使いたくね？
- 玉転がし×役割の分担をできたら面白くね？



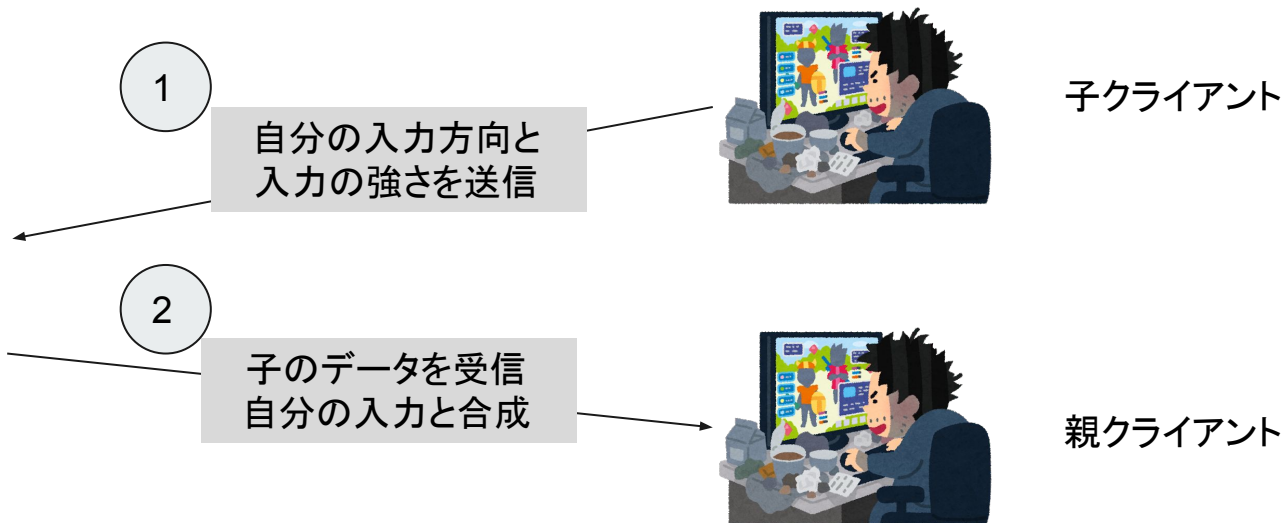
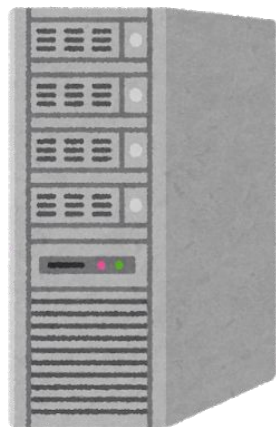


デモ

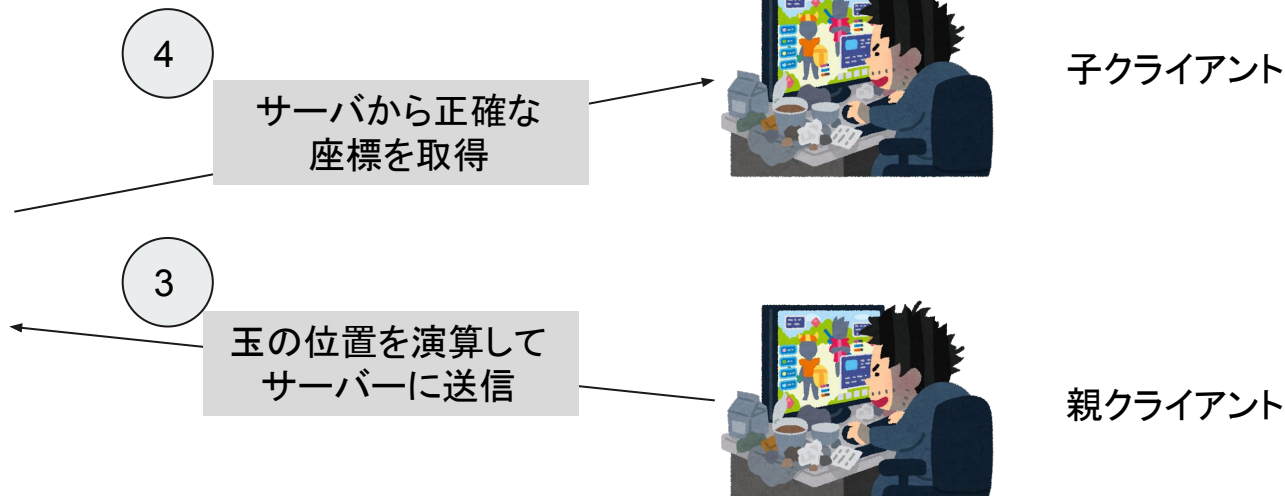
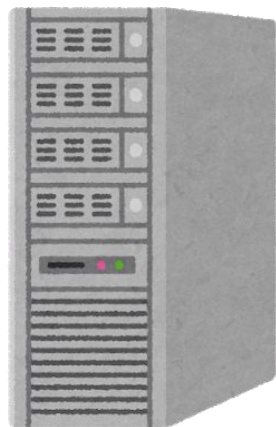


技術周りのこと

クライアント/サーバ間のリアルタイム通信



クライアント/サーバ間のリアルタイム通信





gRPC for Unity

gRPC C#はUnity向けが試験的にサポートされている

Linux / macOS / Windows / Android / iOS 対応 (WebGLもやろうとしたけどサポート対象外)

ProtocolBuffersもちろん使える！

→ クライアント/サーバ間のリアルタイム通信にgRPCのStreamingが使える

クライアント側から逐次的にリクエストストリームに書き込みしたい場合、

Bidirectional Streamingを使う必要があった

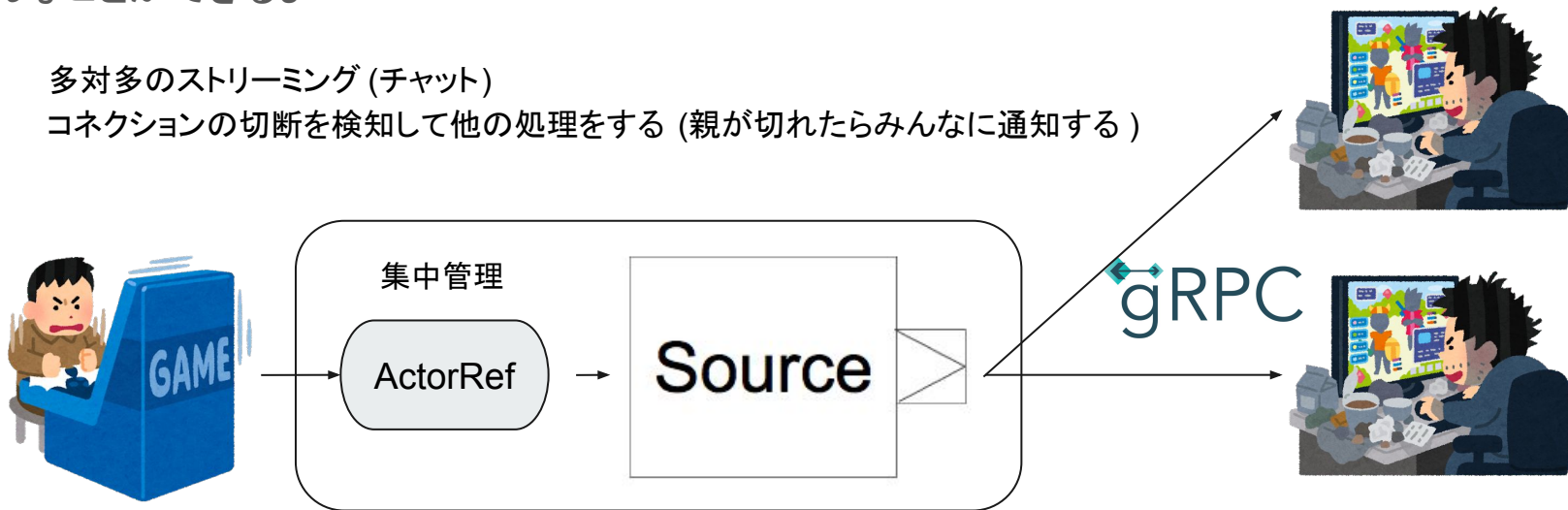
(最初、Client-side Streamingで実装している箇所があったが、これだとリクエストストリームの書き込み完了を待つ必要があり要件を満たせなかった)

Akka Stream

～とは?: ノンブロッキングでback pressureが可能な非同期ストリーム

こんなことができるよ～

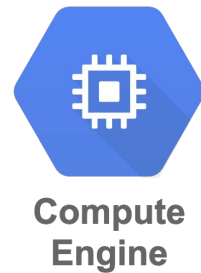
- 多対多のストリーミング (チャット)
- コネクションの切断を検知して他の処理をする (親が切れたらみんなに通知する)





インフラ・CI/CD

GithubActions





ありがとうございました

SlackにゲームのURLを貼っておくので
皆さんも遊んでみて下さい！