一人でオンラインゲーム作ってみた！

created an online game by myself

東京都立南多摩中等教育学校
5年　佐藤　良

ゲームの概要

操作の概要を説明する。pcで動作し、初回起動時のみに予め発行済みのIDとpassを入力する。その後リアルタイムでマッチし戦闘する、シューティングゲームである。このログインシステムから戦闘システムまで全てを実装した。その概略と検討過程を報告する。

必要な要素って何だろう？

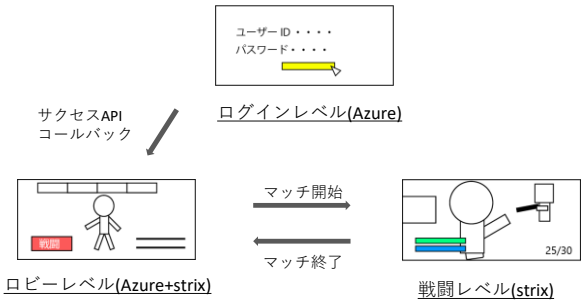
既存のオンラインゲームの種類は様々あるが、今回はオンラインの対戦ゲームを作る。その際必要となる要素の一部を羅列した

①アカウントシステム
②マッチメイキングシステム
③実際の対戦システム

これらの実装にはフレームワークを使用した。アカウントシステムとデータベースはAzure、マッチメイキングと実際の対戦はstrixを使用した。

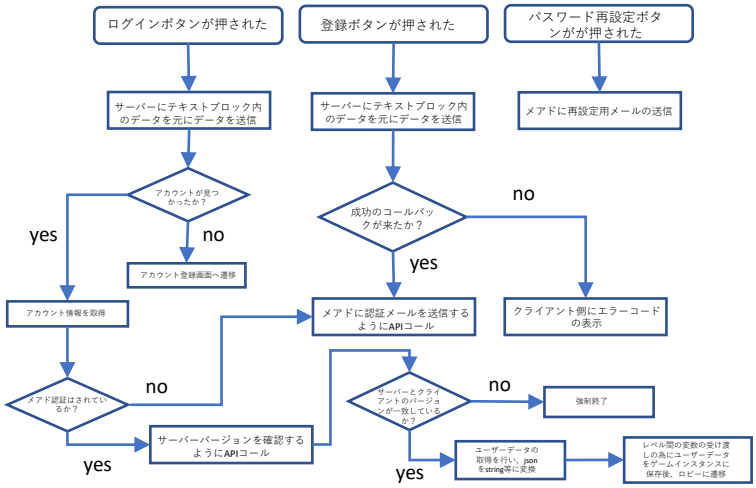
レベルを考える

一度にすべてのデータを使用者の端末にロードすると負荷が大きいため複数のデータ空間を遷移させて実行する。実際にはunrealengineを使用しており、本環境ではレベルと呼ぶ。



図,大まかなレベル遷移

実装したログイン処理の概略



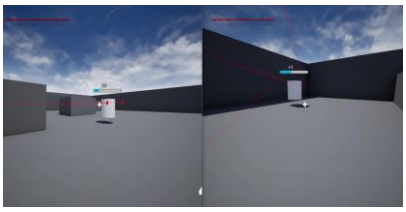
戦闘を考える

ゲームサーバはリレーサーバを利用する。リレーサーバでは各クライアントのデータを共有、レプリケーションしてゲームを進めていく。その際、レプリケーションのタイミングによって処理が大きく変わる。RPCを用いて関数の呼び出しを行うが、高スパンで呼び出すわけにもいかない。以下が発砲処理をした時の具体的例である。

その際の処理を検討した。そのうちの3つを以下に示す。

①弾丸自体をレプリケーションし、当たり判定だけ攻撃者側でとる
②各クライアントで非レプリケーションの弾丸を引数を元に弾道を決定し、判定を攻撃者側でとる。
③インプット以降の処理を全てレプリケーションする

右の写真が③を簡易的に実装した結果である。
攻撃者側(写真左)ではトレースが壁に当たっているのに対して、被攻撃者側(写真右)では大きく上にトレースがずれていることがわかる。
これは僅かなずれでも複数の値や座標が関わることでバタフライエフェクトが発生しているからである。



次に①の方法は、高速で弾丸を飛ばした場合、かなりのサーバ同期レートが必要になるため通信費が高額になる懸念に加えて、クライアント間での同期は完璧にはできないため不採用とした。

そのため②の方法を選択した。rpcの引数で攻撃者側で計算したvectorを取り各クライアントに送信し、ローテーションを計算して弾丸をスポーンさせている。

今後の課題

今後の課題として、1フレームごとにローテーションの変数を元にしているキャラクターのボーン調整部分がなぜかうまくいかず、写真のように高速ヘッドハンティングしている。恐らく補完による変数の上書きが行われているためだと思われる。今後もネットワーク周りの調査を続けていきたいと思う。
又、学校や知人間でテストプレイをした時は概ね正常動作していたが、フレーム落ちによる処理の失敗で正常に動作しないことがあった。今後はエラーが起きた時のリカバリー方法も考えていきたい。