

Yumin Village Minecraft mod

目的

既存のマルチフロントエンドシステムにおける、Minecraft modを使用した新規フロントエンド追加検証

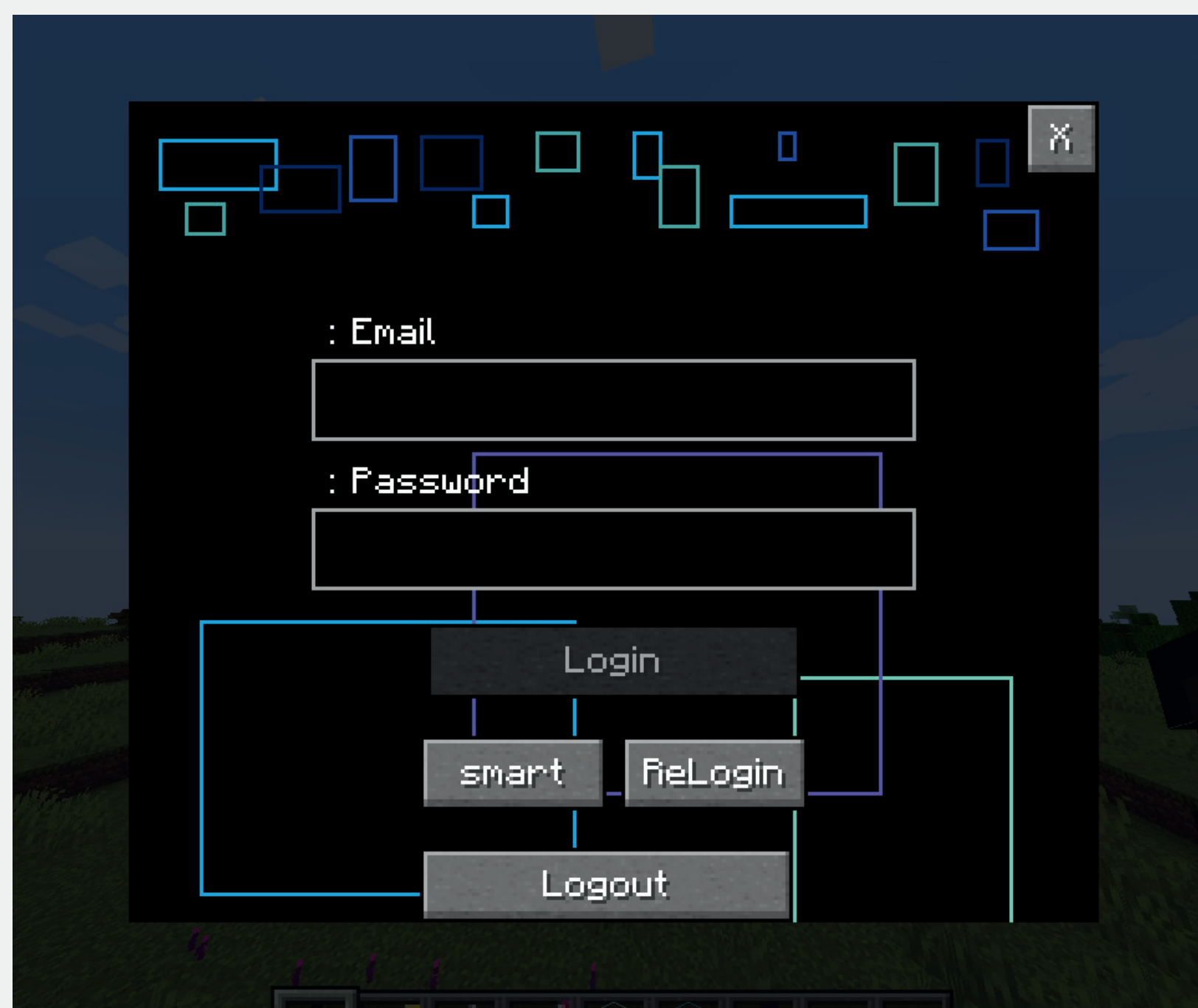
概要

modではアイテム追加をメインに行い、既存システムのWeb APIを使用したシステムの連携と、取得したデータによるゲーム体験の変化を実装する。

システムの連携



アイテム“ログインキー”を使用時にフォームを表示し、Web APIを通してYumin Villageにログイン、データを取得する。



Login Data Holder

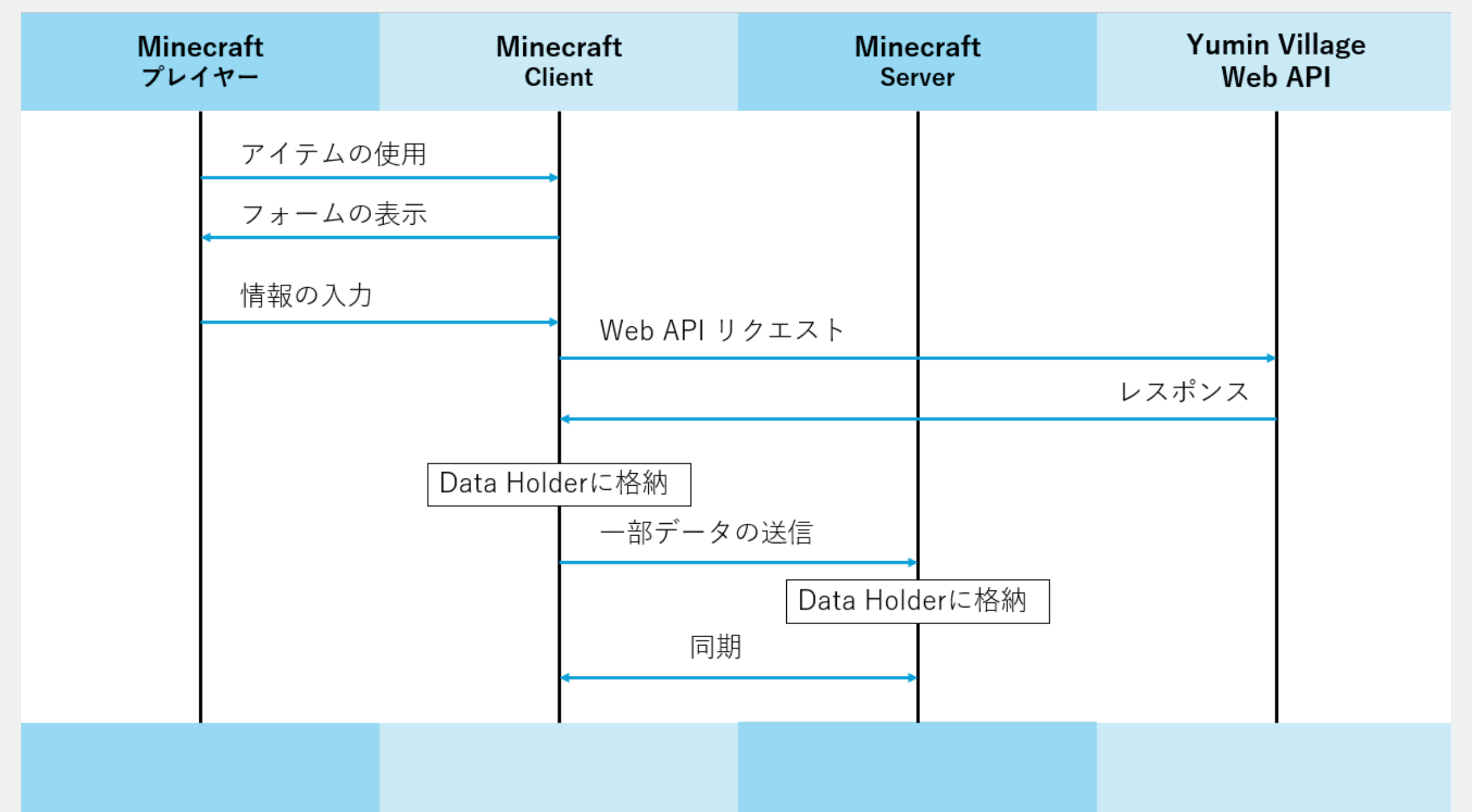
UUID : UUID
 メールアドレス : String
 パスワード : String
 アクセストークン : String
 リフレッシュトークン : String
 ログインステータス : Integer

Member Info Data Holder

キャラクターID : Integer
 メンバーID : Integer
 メールアドレス : String
 名前1 : String
 名前2 : String
 名前3 : String
 名前4 : String

Chara Info Data Holder

キャラクターID : Integer
 ニックネーム : String
 所属1 : String
 所属2 : String
 所属3 : String
 指導者 : String
 請け負える作業レベル : Integer
 過去実績1 : String
 過去実績2 : String
 過去実績3 : String
 自由記述1 : String
 自由記述2 : String
 自由記述3 : String
 自由記述4 : String
 自己PR : String
 過去実績スコア : Integer
 自由スコア : Integer



使用例



開発事例



・ポリデント

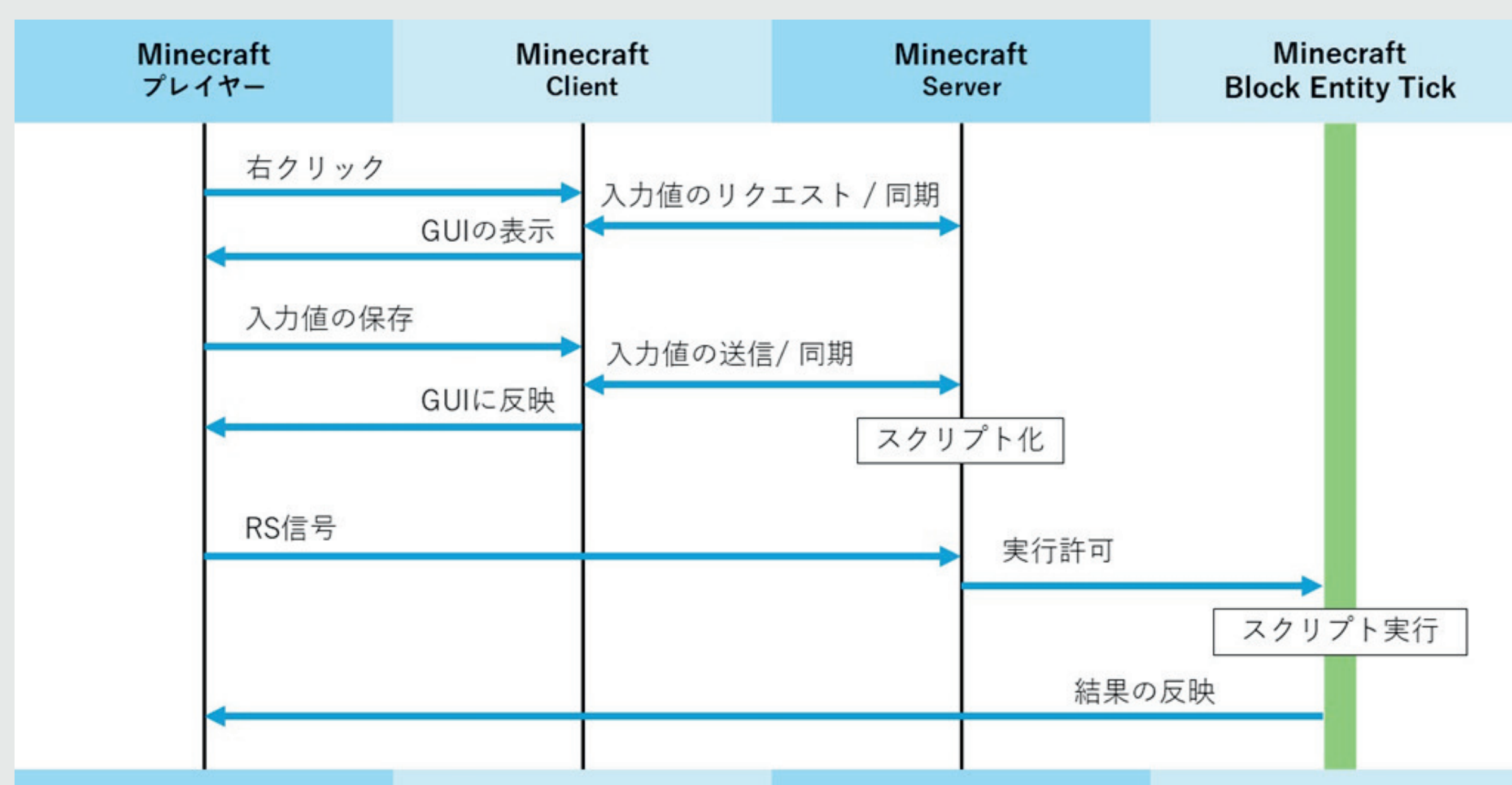
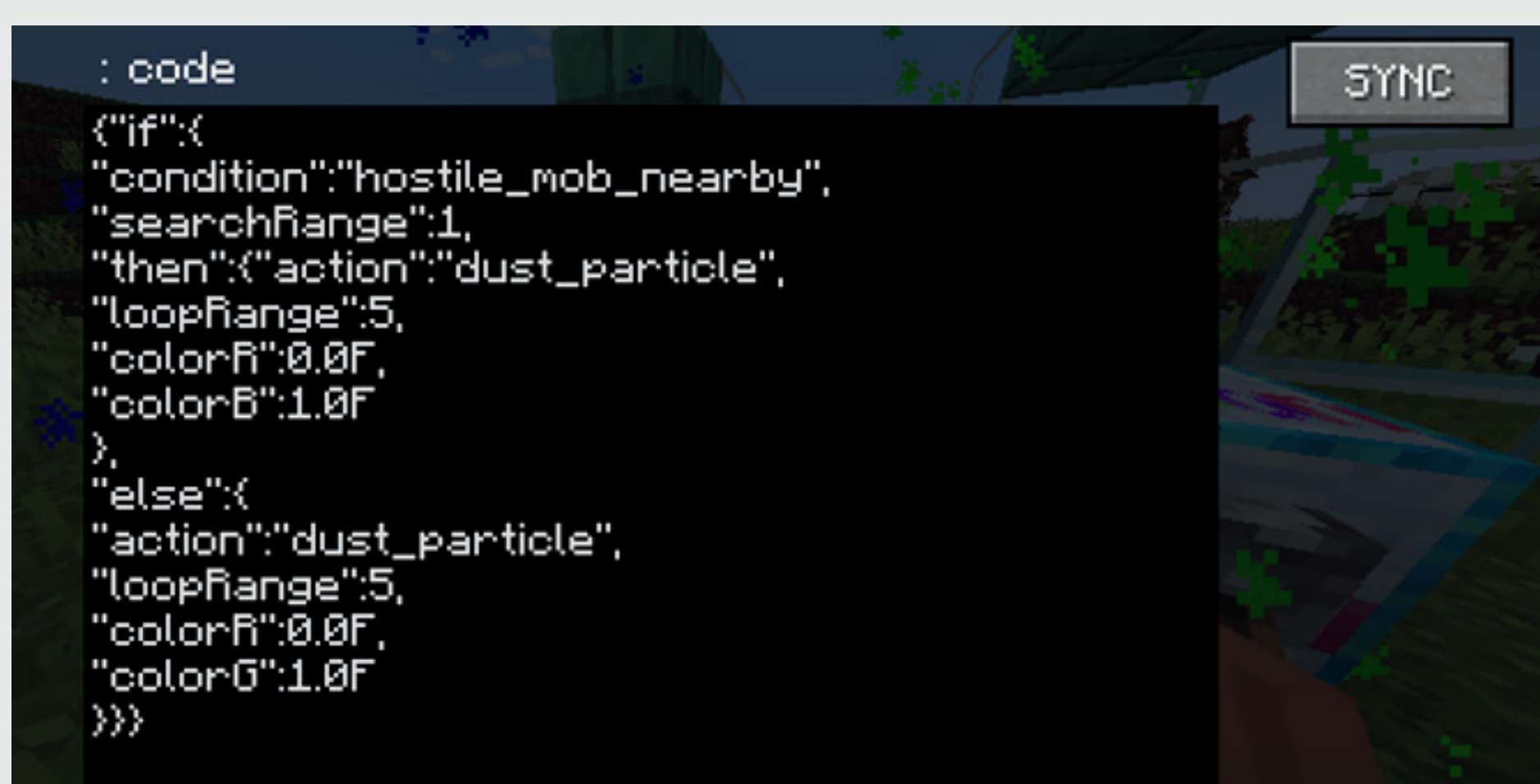
- level work だけ1度に投擲
- 投擲された周囲のmobに移動速度低下を付与



・チャージソード

- 14 - level work * 2回攻撃すると貯まる
- 特殊(爆発)攻撃が可能

- ・ゲーム内から動的にプログラミング/実行可能なブロックを作成
- ・if条件/action内容をサンドボックス化することでゲーム内からプログラム可能な仕組み
- ・近接mob(player/敵対/友好)の(有無/人数/yuminログイン有無)、ランダム等の条件によってアクション：パーティクル生成やメッセージ送信、状態異常付与などが任意の色や頻度、条件、位置、種類が設定可能
- ・ネスト可能なデータモデルにより、複雑なプログラム可能
- ・コーディングはjson形式で行い、実行にあたりスクリプト化



- ・GUI / 入力値はクライアント依存であるが、マルチの同期に当たって通信を作成。サーバーで管理
- ・スクリプト化はサーバーでのみ行い、Tickメソッドを通じてプログラミングしたスクリプトの実行を実現
- ・ネストは再帰処理、ifとactionは分岐処理で汎化や共通化、拡張性を意識した設計

