

# 要件定義書:タイピングRPG

## 1.プロジェクトの概要

項目	内容
プロジェクト名	タイピング形式のRPG
目的	日本語のタイピングを通じて、RPG形式でゲームを進行するシステムの開発
ターゲットユーザー	教育目的でタイピングを学ぶ人
使用技術	サーブレット & JSP、CSS、画像（PNG）、日本語タイピング
開発環境	Java、JSP、CSS、Webサーバー（Tomcat等）

## 2. システム要件

項目	内容
プレイヤー	HP、名前、画像
敵モンスター	HP、名前、画像
コマンド	txtファイルからランダム出題
タイピング	指定された日本語単語を正確にタイピング（完全一致）
戦闘ルール	正確にタイピング → プレイヤーの行動成功、不正解 → 敵の攻撃
状態管理	HttpSessionでHPや進行状況を保持
UIデザイン	CSSでHPバー、配置、敵・プレイヤー画像の表示

3.画面設計(JSP)

画面名	機能
login.jsp	start.jspユーザーIDとパスワード入力画面
start.jsp	ゲーム開始ボタン、プロジェクトの説明
game.jsp	プレイヤーと敵の情報、タイピングワード、入力フォーム
win.jsp	勝利画面（ゲームクリア時）
lose.jsp	敗北画面（HPが0になった場合）

4.サーバー処理(Servlet)

処理	内容
LoginServlet.java	ログイン認証処理（ID/パスワードチェック後、セッションにログイン情報を保存）
GameServlet.java	ゲーム進行ロジック（セッションからログインとHPを管理）、ログインチェック機能

5. ゲームの流れ(プレイサイクル)

login.jsp :	ユーザーIDとパスワードを入力
-------------	-----------------

LoginServlet :	成功 → LoginServlet → GameServlet?action=start にリダイレクト
	失敗 → login.jsp?error=1 に戻ってエラーメッセージ表示
	GameServlet 側で未ログインならアクセスを遮断
start.jsp :	ゲーム開始ボタン押下
GameServlet :	セッションを初期化 (HP、敵データ、ワードなど)
game.jsp :	現在のHP、敵情報、ランダムに出題されたタイピングワード表示
POST送信 :	ユーザーが入力 → GameServletで一致判定
正解／不正解処理 :	プレイヤーの攻撃または敵の攻撃
勝敗判定 :	HPが0になったらwin.jspまたはlose.jspに遷移
続行 :	続ける場合は再度ランダムワードを出題して戦闘続行

## 6.リスク管理

リスク	対応策
タイピング速度の違い	難易度調整オプションの実装
ゲームのバグ	ユーザーテストを通じたバグの特定と修正
サーバーダウン	サーバーの負荷テスト、監視システムの導入