Entwicklungsplan für das Snake-Spiel

Jan Osing, Gabriel Janich,
Hendrik Siemens

April 20, 2024

1 Projekt Setup und Vorbereitung

Ziele:

- Einrichten der Entwicklungsumgebung.
- Festlegung der Projektstruktur und Dateiorganisation.

Aufgaben:

1. Webserver (z.B. Apache oder Nginx).
2. MySQL-Datenbankserver einrichten.
3. PHP-Entwicklungsumgebung konfigurieren (z.B. XAMPP, LAMP).
4. Verzeichnisstruktur für das Projekt erstellen.

2 Datenbankdesign und -integration

Ziele:

- Erstellen und Konfigurieren der MySQL-Datenbank.
- Sicherstellen, dass die Datenbankverbindung effizient und sicher ist.

Aufgaben:

1. Ausführen der SQL-Skripte zur Erstellung der Datenbank und Tabellen.
2. PHP-Skripte schreiben, um eine Datenbankverbindung herzustellen.
3. CRUD-Operationen für Spieler und Scores als PHP-Funktionen implementieren.

3 Backend-Entwicklung

Ziele:

- Erstellen der Server-Logik zur Verwaltung von Spielsessions und Spielerdaten.

Aufgaben:

1. PHP-Skripte entwickeln für:
    - Anmeldung und Registrierung von Spielern.
    - Starten und Beenden von Spielsitzungen.
    - Aktualisierung und Abruf von Spieler-Scores.

4 Frontend-Entwicklung

Ziele:

- Design und Implementierung der Benutzeroberfläche.
- Entwicklung der Spiellogik.

Aufgaben:

1. HTML-Seiten erstellen für Spielansicht, Leaderboard und Spielerprofil.
2. CSS für responsive Design verwenden.
3. JavaScript für die Spiellogik entwickeln.
4. AJAX verwenden, um die Highscore-Tabelle live zu aktualisieren.

5 Testing und Qualitätssicherung

Ziele:

- Sicherstellen, dass das Spiel auf verschiedenen Geräten und Browsern korrekt funktioniert.
- Überprüfung der Sicherheit, insbesondere der Datenübertragungen.

Aufgaben:

1. Funktionstests durchführen für alle Features.
2. Performance-Tests für die Webseite und die Datenbank.
3. Sicherheitstests, um SQL-Injection und XSS-Angriffe zu verhindern.

6 Deployment und Wartung

Ziele:

- Veröffentlichung der Anwendung auf einem Live-Server.
- Planung für die Wartung und mögliche Updates.

Aufgaben:

1. Auswählen eines Hosting-Anbieters und Konfigurieren des Live-Servers.
2. Deployment der Anwendung und der Datenbank auf dem Live-Server.
3. Einrichten eines VCS (z.B. Git) für Versionskontrolle.
4. Überwachung der Anwendung auf Fehler und Performance-Probleme.

7 Authentifizierung und Sicherheit

Ziele:

- Implementierung einer sicheren Benutzerauthentifizierung und -autorisation.
- Schutz der Benutzerdaten und -interaktionen innerhalb der Anwendung.

Aufgaben:

1. Implementierung des Registrierungsprozesses für neue Benutzer:
    - Entwicklung eines sicheren Registrierungs-Endpoints.
    - Verwendung von Passwort-Hashing mit modernen Sicherheitsstandards.

2. Einrichtung des Login-Prozesses unter Verwendung von JSON Web Tokens (JWT):
    - Integration der Firebase/php-jwt-Bibliothek zur Token-Erzeugung und -Verifizierung.
    - Erstellung von Token nach erfolgreicher Authentifizierung und Rückgabe an den Client.
    - Validierung des Tokens bei jedem authentifizierungspflichtigen Request.
3. Implementierung des Logout-Mechanismus:
    - Anleitung des Clients, das gespeicherte Token zu löschen.
    - Optional: Einrichtung einer Token-Blacklist auf dem Server zur Invalidierung von Tokens.
4. Sicherstellung der Sicherheit durch HTTPS für alle Kommunikationswege.
5. Durchführung von Sicherheitstests, um die Wirksamkeit der implementierten Maßnahmen zu überprüfen.

8 Changelog und Dokumentation der Entwicklung

Einträge:

- **2024-04-19:**
    - Projektinitialisierung und Setup der Entwicklungsumgebung.
    - Einrichtung von Apache, MySQL, und PHP auf einem Raspberry Pi 4B.
    - Beginn der Erstellung der Datenbank und der notwendigen Tabellen für das Spiel.
    - Einrichtung und Konfiguration von SSL für sichere Datenübertragungen.
    - Erste Schritte zur Konfiguration von Apache und PHP.
- **2024-04-20:**
    - Implementierung des Login-Endpunkts unter Verwendung von JWT für die Authentifizierung.
    - Identifizierung und Lösung von Problemen mit der Apache-Konfiguration.
    - Erste Tests der Authentifizierungsfunktionen durchgeführt.
- **2024-04-20:**
    - Implementierung des Registrierungs-Endpunkts und Integration des sicheren Passwort-Hashing.
    - Implementierung des Logout-Mechanismus, der das Löschen des Tokens auf der Client-Seite umfasst.

Dokumentation:
Die Entwicklungsdokumentation umfasst technische Spezifikationen, Code-Kommentare und Setup-Anleitungen, um eine konsistente Wartung und zukünftige Erweiterungen zu gewährleisten. Eine detaillierte Dokumentation der Authentifizierungsprozesse wurde ebenfalls hinzugefügt, um die Sicherheitsmechanismen und den Datenschutz innerhalb der Anwendung zu erläutern.