

Projekt Backlog - Erweiterte User Stories

Technical User Stories (for Development)

Epic: SCRUM-19 | **Status:** To Do

User Stories

Database Creation (SCRUM-20)

Titel: Datenbank erstellen

Beschreibung: Als Entwickler möchte ich eine Datenbank mit allen notwendigen Informationen erstellen, damit die Anwendung Pokemon-Daten persistent speichern kann.

Akzeptanzkriterien:

- Die Datenbank enthält Tabellen für Pokemon, Moves, Teams und Benutzer
- Alle Beziehungen zwischen Entitäten sind korrekt definiert
- Die Datenbank ist normalisiert (mind 3. NF) und performant
- Seed-Daten für Pokemon und Moves sind verfügbar

Schätzung: 4 Stunden

Priorität: 50 (Sehr wichtig)

Tasks:

- **SCRUM-43:** Develop Database Model (1,5 Stunden)
-

Domain Model Development (SCRUM-42)

Titel: Domain Model entwickeln **Beschreibung:** Als Entwickler möchte ich ein Domain Model erstellen, damit ich einfach API-Endpunkte entwickeln kann und eine klare Struktur habe.

Akzeptanzkriterien:

- Alle Geschäftsobjekte sind als C# Klassen implementiert
- Entitäten und Value Objects sind klar getrennt
- Aggregate Roots sind definiert und implementiert
- Das Model ist durch Diagramme visualisiert

Schätzung: 6 Stunden

Priorität: 40 (Wichtig)

Tasks:

- **SCRUM-46:** Visualize data using relational diagrams or graphics (1 Stunde)
- **SCRUM-47:** Split data into entities, value-objects (1,5 Stunden)
- **SCRUM-48:** Build useful Aggregates (2 Stunden)
- **SCRUM-49:** Implement as C# code (1,5 Stunden)

Disconnect Handling (SCRUM-21)

Titel: Verbindungsabbrüche behandeln

Beschreibung: Als Entwickler möchte ich Verbindungsabbrüche behandeln, damit Kämpfe nach einem Disconnect fortgesetzt werden können.

Akzeptanzkriterien:

- Spielstatus wird automatisch alle x Sekunden gespeichert
- Bei Verbindungsabbruch erscheint ein Wartebildschirm für max. 2 Minuten
- Automatische Wiederverbindung funktioniert korrekt
- Spieler können manuell reconnecten

Schätzung: 7,5 Stunden

Priorität: 30 (Mittel)

Tasks:

- **SCRUM-50:** Save game state on potential connection loss or regularly (3 Stunden)
- **SCRUM-51:** Implement waiting screen (1-2min) (2 Stunden)
- **SCRUM-52:** Either add reconnect button or do it automatically (2,5 Stunden)

Team Composition Validation (SCRUM-22)

Titel: Team-Zusammenstellung validieren

Beschreibung: Als Entwickler möchte ich Team-Zusammenstellungen validieren, damit Spieler keine ungültigen Teams verwenden können.

Akzeptanzkriterien:

- Keine doppelten Pokemon im Team
- Jedes Pokemon muss gültige Moves haben
- Fehlermeldungen sind benutzerfreundlich und spezifisch

Schätzung: 4 Stunden

Priorität: 35 (Wichtig)

Tasks:

- **SCRUM-53:** Define checks for validation (1,5 Stunden)
- **SCRUM-54:** Visually display what the user did wrong (2,5 Stunden)

Account Management

Epic: SCRUM-1 | **Status:** To Do

 User Stories

Account Creation (SCRUM-23)

Titel: Account erstellen

Beschreibung: Als neuer Spieler möchte ich einen Account erstellen, damit ich meinen Fortschritt speichern kann.

Akzeptanzkriterien:

- Benutzer kann sich mit E-Mail und Passwort registrieren
- E-Mail-Validierung ist implementiert
- Passwort-Anforderungen werden überprüft (min 8 characters)

Schätzung: 3 Stunden

Priorität: 45 (Sehr wichtig)

Tasks:

- **SCRUM-55:** Choose a framework that handles account creation/validation (1 Stunde)
 - **SCRUM-56:** Follow the frameworks documentation to implement it (2 Stunden)
-

Secure Login (SCRUM-24)

Titel: Sicherer Login

Beschreibung: Als wiederkehrender Spieler möchte ich mich sicher anmelden, damit ich auf meine gespeicherten Daten zugreifen kann.

Akzeptanzkriterien:

- Login mit E-Mail/Benutzername und Passwort funktioniert
- "Passwort vergessen" Funktion ist verfügbar

Schätzung: 2 Stunden

Priorität: 45 (Sehr wichtig)

Tasks:

- **SCRUM-57:** Implement secure login functionality (2 Stunden)
-

Profile Statistics (SCRUM-25)

Titel: Profil-Statistiken anzeigen

Beschreibung: Als Spieler möchte ich meine Profil-Statistiken einsehen, damit ich meinen Fortschritt verfolgen kann.

Akzeptanzkriterien:

- Anzeige von Wins/Losses Ratio
- Gesamtanzahl gespielter Kämpfe
- Lieblings-Pokemon werden angezeigt
- Statistiken sind grafisch aufbereitet

Schätzung: 5 Stunden

Priorität: 20 (Niedrig)

Tasks:

- **SCRUM-58:** Update backend to save history of all matches (2 Stunden)
 - **SCRUM-59:** Create endpoint which provides necessary statistics (1,5 Stunden)
 - **SCRUM-60:** Develop UI that displays statistics in a nice manner (1,5 Stunden)
-

Account Management (SCRUM-26)

Titel: Account verwalten

Beschreibung: Als Spieler möchte ich meinen Benutzernamen und mein Passwort ändern können, damit ich mein Konto pflegen kann.

Akzeptanzkriterien:

- Benutzername kann geändert werden (Eindeutigkeit wird geprüft)
- Passwort kann mit aktueller Passwort-Bestätigung geändert werden
- Änderungen werden sofort wirksam
- Bestätigungs-E-Mail bei wichtigen Änderungen

Schätzung: 3 Stunden

Priorität: 15 (Niedrig)

Tasks:

- **SCRUM-61:** Check docs of authentication provider (1 Stunde)
 - **SCRUM-62:** Enable or implement if necessary (2 Stunden)
-

Team Management

Epic: SCRUM-16 | **Status:** To Do

 User Stories

Team Building (SCRUM-27)

Titel: Team zusammenstellen

Beschreibung: Als Spieler möchte ich ein Team aus bis zu 6 Pokemon zusammenstellen, damit ich mich auf Kämpfe vorbereiten kann.

Akzeptanzkriterien:

- Alle verfügbaren Pokemon werden in einer Liste angezeigt
- Spieler kann 6 Pokemon auswählen
- Ausgewählte Pokemon werden visuell hervorgehoben
- Team kann gespeichert werden

Schätzung: 9 Stunden

Priorität: 40 (Wichtig)

Tasks:

- **SCRUM-63:** Create Endpoint that provides necessary information (5 Stunden)
 - **SCRUM-64:** Implement UI that displays Pokemon and allows selection (4 Stunden)
-

Pokemon Stats Viewing (SCRUM-29)

Titel: Pokemon-Stats anzeigen

Beschreibung: Als Spieler möchte ich detaillierte Stats jedes Pokemon einsehen, damit ich strategische Entscheidungen treffen kann.

Akzeptanzkriterien:

- Alle Basis-Stats (HP, Attack, Defense, etc.) werden angezeigt
- Pokemon-Typ und Fähigkeiten sind sichtbar
- Stats sind übersichtlich formatiert
- Detailansicht öffnet sich per Klick/Touch

Schätzung: 4 Stunden

Priorität: 35 (Wichtig)

Tasks:

- **SCRUM-65:** Add detailed information endpoint (1,5 Stunden)
 - **SCRUM-66:** Implement UI for Pokemon details (2,5 Stunden)
-

Move Customization (SCRUM-30)

Titel: Attacken anpassen

Beschreibung: Als Spieler möchte ich die Attacken meiner Pokemon anpassen, damit ich verschiedene Strategien entwickeln kann.

Akzeptanzkriterien:

- Jedes Pokemon kann bis zu 4 Attacken haben
- Nur erlaubte Attacken für das jeweilige Pokemon werden angezeigt
- Attacken können einzeln ausgetauscht werden
- Änderungen werden sofort gespeichert

Schätzung: 7,5 Stunden

Priorität: 30 (Mittel)

Tasks:

- **SCRUM-67:** API endpoint for possible moves (2,5 Stunden)
- **SCRUM-68:** UI for move selection (3 Stunden)
- **SCRUM-69:** Update team validation (2 Stunden)

Team Configuration Management (SCRUM-31)

Titel: Team-Konfigurationen verwalten

Beschreibung: Als Spieler möchte ich mehrere Team-Konfigurationen speichern, damit ich zwischen ihnen wechseln kann.

Akzeptanzkriterien:

- Mehrere Teams können gespeichert werden
- Gespeicherte Teams können geladen und bearbeitet werden

Schätzung: 9 Stunden

Priorität: 25 (Mittel)

Tasks:

- **SCRUM-70:** Update DB for team configurations (2 Stunden)
 - **SCRUM-71:** Backend endpoints for team management (3 Stunden)
 - **SCRUM-72:** Add save option to team creation (2 Stunden)
 - **SCRUM-73:** Team configuration management screen (2 Stunden)
-

Battle System

Epic: SCRUM-4 | **Status:** To Do

 User Stories

AI Battle (SCRUM-32)

Titel: KI-Kampf

Beschreibung: Als Spieler möchte ich gegen eine KI kämpfen, damit ich meine Strategie üben kann.

Akzeptanzkriterien:

- KI wählt intelligente Züge basierend auf Spielsituation
- KI hat verschiedene Schwierigkeitsgrade
- Kampf läuft rundenbasiert ab
- KI reagiert in angemessener Zeit (max. 3 Sekunden)

Schätzung: 10 Stunden

Priorität: 35 (Wichtig)

Tasks:

- **SCRUM-74:** Develop opponent model (10 Stunden)
-

Player vs Player Battle (SCRUM-33)

Titel: Spieler gegen Spieler

Beschreibung: Als Spieler möchte ich gegen andere Spieler kämpfen, damit ich meine Fähigkeiten testen kann.

Akzeptanzkriterien:

- Matchmaking funktioniert automatisch
- Echtzeit-Kommunikation zwischen Spielern
- Synchrone Züge beider Spieler
- Verbindungsabbrüche werden behandelt

Schätzung: 25 Stunden

Priorität: 30 (Mittel)

Tasks:

- **SCRUM-77:** Implement (SignalR?) for real-time communication (18 Stunden)
 - **SCRUM-78:** Frontend for online match search (7 Stunden)
-

Move Selection (SCRUM-34)

Titel: Zug-Auswahl

Beschreibung: Als Spieler möchte ich Züge während des Kampfes auswählen, damit ich meinen Gegner angreifen kann.

Akzeptanzkriterien:

- Alle verfügbaren Attacken werden angezeigt
- Attacken-Details (Schaden, Typ) sind sichtbar
- Zug wird nach Auswahl sofort übermittelt
- Schäden werden korrekt berechnet

Schätzung: 9 Stunden

Priorität: 45 (Sehr wichtig)

Tasks:

- **SCRUM-79:** Backend for turn-based combat (4 Stunden)
 - **SCRUM-80:** UI for move selection (3 Stunden)
 - **SCRUM-81:** Unit tests for damage calculations (2 Stunden)
-

Pokemon Switching (SCRUM-35)

Titel: Pokemon wechseln

Beschreibung: Als Spieler möchte ich während des Kampfes zwischen meinen Pokemon wechseln, damit ich meine Strategie anpassen kann.

Akzeptanzkriterien:

- Nur lebende Pokemon können eingewechselt werden

- Wechsel kostet eine Runde
- Aktuelles Pokemon wird visuell hervorgehoben

Schätzung: 3 Stunden

Priorität: 40 (Wichtig)

Tasks:

- **SCRUM-83:** Frontend for Pokemon switching (3 Stunden)
-

Battle Animations (SCRUM-36)

Titel: Kampf-Animationen

Beschreibung: Als Spieler möchte ich Kampf-Animationen sehen, damit ich dem Geschehen folgen kann.

Akzeptanzkriterien:

- Animationen für Angriffe sind vorhanden
- Pokemon-Sprites bewegen sich entsprechend
- Schadens-Anzeige ist animiert

Schätzung: 11 Stunden

Priorität: 15 (Niedrig)

Tasks:

- **SCRUM-84:** Source battle animations (6 Stunden)
 - **SCRUM-87:** Implement animations in frontend (5 Stunden)
-

Battle Forfeit (SCRUM-14)

Titel: Kampf aufgeben

Beschreibung: Als Spieler möchte ich einen Kampf aufgeben können, damit ich ein ungünstiges Match beenden kann.

Akzeptanzkriterien:

- Aufgeben-Button ist jederzeit verfügbar
- Bestätigungsdialog verhindert versehentliches Aufgeben
- Gegner wird sofort über Aufgabe informiert
- Kampf endet sofort mit entsprechendem Ergebnis

Schätzung: 2 Stunden

Priorität: 25 (Mittel)

Tasks:

- **SCRUM-88:** Add forfeit endpoint (1 Stunde)
 - **SCRUM-89:** Add forfeit button to UI (1 Stunde)
-

Social Features

Epic: SCRUM-17 | **Status:** To Do

User Stories

Friend Management (SCRUM-37)

Titel: Freunde verwalten

Beschreibung: Als Spieler möchte ich Freunde hinzufügen, damit ich einfach Kampfpartner finden kann.

Akzeptanzkriterien:

- Freunde können über Benutzername gesucht und hinzugefügt werden
- Freundschaftsanfragen müssen bestätigt werden
- Freundesliste zeigt Online-Status an
- Freunde können entfernt werden

Schätzung: 8 Stunden

Priorität: 20 (Niedrig)

Tasks:

- **SCRUM-90-93:** Complete friend system implementation (8 Stunden)
-

Profile Viewing (SCRUM-38)

Titel: Profile anderer Spieler anzeigen

Beschreibung: Als Spieler möchte ich Profile anderer Spieler einsehen, damit ich ihre Stats und Strategien kennenlernen kann.

Akzeptanzkriterien:

- Öffentliche Statistiken anderer Spieler sind sichtbar
- Kampfhistorie kann eingesehen werden (falls öffentlich)
- Profile sind über Freundesliste oder Suche erreichbar
- Datenschutz-Einstellungen werden respektiert

Schätzung: 6 Stunden

Priorität: 15 (Niedrig)

Tasks:

- **SCRUM-94:** Add stats endpoint (2,5 Stunden)
 - **SCRUM-95:** Update UI for profile viewing (3,5 Stunden)
-

Friend Challenges (SCRUM-39)

Titel: Freunde herausfordern

Beschreibung: Als Spieler möchte ich Kampf-Herausforderungen an Freunde senden, damit ich gegen sie

antreten kann.

Akzeptanzkriterien:

- Herausforderungen können an Online-Freunde gesendet werden
- Empfänger erhält Benachrichtigung über Herausforderung
- Herausforderungen können angenommen oder abgelehnt werden
- Bei Annahme startet sofort ein privater Kampf

Schätzung: 6 Stunden

Priorität: 18 (Niedrig)

Tasks:

- **SCRUM-97:** Implement challenge system (6 Stunden)
-

Battle History (SCRUM-40)

Titel: Kampfhistorie anzeigen

Beschreibung: Als Spieler möchte ich meine Kampfhistorie einsehen, damit ich meine Leistung überprüfen kann.

Akzeptanzkriterien:

- Chronologische Liste aller Kämpfe
- Details zu jedem Kampf (Gegner, Ergebnis, Datum, verwendetes Team)
- Filtermöglichkeiten (Sieg/Niederlage, Gegnertyp)
- Paginierung bei vielen Einträgen

Schätzung: 5 Stunden

Priorität: 22 (Niedrig)

Tasks:

- **SCRUM-98:** Update DB for battle history (1,5 Stunden)
 - **SCRUM-99:** Add GET endpoint for history (1,5 Stunden)
 - **SCRUM-100:** Update frontend for history view (2 Stunden)
-

Gesamtübersicht: 5 Epics | 20 User Stories | 58 Tasks