

# Project Specification



## TwentyOne

### Resource Description

Ressource	Description	Relationship
player	User of the application	has many game-sessions
game-session	Instance of a Blackjack game	has many cards, many moves, one game-result and one player
move	Represents a player's action	belongs to game-session
card	Model of a dealt Card	belongs to game-session
game-result	Result of game-session	belongs to game-session

### User Stories

#### User Story 1 – Static page

As an interested visitor, I want to read the rules of Blackjack and how the app works, so that I can understand the game before I start playing.

**Affected Resources:**

none

**Planned Implementation:**

Homepage with a static explanation of game rules, visual illustrations, and a start button.

## User Story 2 – View Game Status (GET)

**Story:**

As a player, I want to view the current state of the game, so that I can see my cards, the dealer's visible card, and my total.

**Affected Resources:**

`game-session`, `card`

**Planned Implementation:**

GET `/game-sessions/{id}` → Displays the current hand (player and dealer), totals, and game phase.

## User Story 3 – View Game Result (GET with Relationship)

**Story:**

As a player, I want to see the game result after a game session ended, so that I know the outcome of a session.

**Affected Resources:**

`game-result`, `game-session`

**Planned Implementation:**

GET `/game-sessions/{id}/game-results` → Displays outcome of a game session.

## User Story 4 – Player Profile

**Story:**

As a player, I want to manage my profile, so that I can create, view, update, or delete my account.

**Affected Resources:**

`player`, `game-session`, `game-result`

**Planned Implementation:**

- GET `/players/{id}` → View player details
- GET `/players/{id}/game-session?outcome={win/loss/push}` → View all game-sessions (with game-result) of a player with given outcome.
- POST `/players` → Create a new player profile
- PUT `/players/{id}` → Update player details
- DELETE `/players/{id}` → Delete player profile

## User Story 5 – Game Session

### Story:

As a player, I want to control my game session, so that I can start a new game, monitor its progress, make changes if needed, or end it when I'm done.

### Affected Resources:

`game-session`, `card`

### Planned Implementation:

- GET `/game-sessions/{id}` → Displays the current hand (player and dealer), totals, and game phase (User Story 2).
- POST `/game-sessions` → Start a new game session
- PUT `/game-sessions/{id}` → Modify game-session details (e.g., change activity state)
- DELETE `/game-sessions/{id}` → End the game session and clear state

## User Story 6 – Moves

### Story:

As a player, I want to view, create, modify, or delete moves during a game session, so that I can play blackjack.

### Affected Resources:

`move`, `game-session`

### Planned Implementation:

- GET `/game-sessions/{id}/moves` → View all moves of a game-session
- POST `/game-sessions/{id}/moves` → Create a new move for a game-session (e.g., "hit", "stand")
- PUT `/moves/{id}` → Update an existing move before it's resolved (timer)
- DELETE `/moves/{id}` → Remove a move (e.g., invalid or misclick) before it's resolved (timer)