**1. View Personal Gacha Collection**

* **API Endpoint**: localhost:8080/gacha/my\_gacha\_list
* **Flow**:
  + Player sends a GET request to the API Gateway at /my\_gacha\_list.
  + The API Gateway forwards the request to the Player microservice.
  + The Player microservice sends a GET request to the Players DB manager at /players/1.
  + The Players DB manager retrieves the player information from the Players DB and returns it to the Player microservice.
  + The Player microservice returns the gacha list to the API Gateway.

**2. Access Details of Specific Gacha in Personal Collection**

* **API Endpoint**: localhost:8080/gacha/1/1
* **Flow**:
  + Player sends a GET request to the API Gateway at /gacha/1/1.
  + The API Gateway forwards the request to the Player microservice.
  + The Player microservice sends a GET request to the Players DB manager at /players/1.
  + The Players DB manager retrieves the player information from the Players DB and returns it to the Player microservice.
  + The response contains the list of the player’s gachas. If gacha 1 is in the list, the Player microservice sends a GET request to the Gacha microservice at /gachas/1.
  + The Gacha microservice retrieves the gacha 1 information from the Gachas DB and returns it to the Player microservice.
  + The Player microservice returns the gacha 1 information to the API Gateway.

**3. Purchase In-Game Currency**

* **API Endpoint**: localhost:8080/users/player/real\_money\_transaction
* **Flow**:
  + Player sends a POST request to the API Gateway at /real\_money\_transaction with the user ID and amount.
  + The API Gateway forwards the request to the Player microservice.
  + The Player microservice sends a POST request to the Players DB manager at /update\_balance/PLAYER with the user ID and amount.
  + The Players DB manager updates the user’s balance in the Players DB and returns a success response.
  + The Player microservice sends a POST request to the Transactions microservice to record the transaction.
  + The Transactions microservice records the transaction in the Transactions DB and returns a success response to the Player microservice.
  + The Player microservice returns a success response to the API Gateway.

**4. Set Up Auctions for Personal Gacha Items**

* **API Endpoint**: localhost:8080/auctions/add
* **Flow**:
  + Player sends a POST request to the API Gateway at /set\_auction with the gacha ID and auction details.
  + The API Gateway forwards the request to the Auction microservice.
  + The Auction microservice sends a GET request to the Gacha microservice to verify the gacha ownership.
  + The Gacha microservice verifies the ownership from the Gachas DB and returns the result to the Auction microservice.
  + The Auction microservice creates the auction in the Auctions DB and returns a success response to the API Gateway.