**Admin Login, Booking, and Unit Test Cases Documentation**

**1. Admin Login Flow**

**1.1 Authentication Process**

Admins authenticate using their credentials (email and password). The login flow follows these steps:

1. Admin enters their email and password.
2. The system verifies that both fields are provided.
3. If the provided email matches the stored admin email (from environment variables), authentication proceeds.
4. The system searches for the admin user in the database.
5. If the user exists, the system compares the provided password with the stored hashed password.
6. Upon a successful match, a secure token (JWT) is generated and stored as a cookie.
7. The system responds with a successful login message.

**1.2 Error Handling**

* If any field is missing, a 422 status is returned with a message: Please fill all fields.
* If the email does not match the stored admin email, a 400 status is returned with Invalid Credentials.
* If the user is not found, a 400 status is returned with Invalid Credentials.
* If the password does not match, a 400 status is returned with Invalid Credentials.
* If an internal error occurs, a 500 status is returned.

**1.3 Security Best Practices**

* **Password Hashing:** Admin passwords are securely hashed using bcrypt.
* **Token-based Authentication:** A JWT is generated upon successful login to maintain session security.
* **Secure Cookies:** The JWT is stored in an HTTP-only cookie for security.
* **Input Validation:** Required fields are validated before processing.

**2. Admin Booking Flow**

**2.1 Booking List Retrieval**

Admins can retrieve a list of all bookings using the AdminBookingsController. Steps include:

1. Admin sends a request with their email.
2. If email is missing, a 422 status is returned with No booking for this Email.
3. If email does not match the stored admin email, a 400 status is returned with Unauthorized.
4. If bookings exist, they are fetched and returned as a structured list.
5. If no bookings exist, a 400 status is returned with No Bookings yet.
6. If an internal error occurs, a 500 status is returned.

**2.2 Admin Ticket Booking**

Admins can book tickets using the AdminBookTicketController.

1. Admin submits user details and selected adventures.
2. If required details are missing, a 422 status is returned with Please fill all fields.
3. If no visitors are selected, a 422 status is returned with Select at least one Visitor.
4. A unique booking number is generated.
5. The booking is saved in the database.
6. If successful, a 201 status is returned with Booking successfully.
7. If an internal error occurs, a 500 status is returned.

**2.3 Admin Ticket Editing**

Admins can modify booking details using the AdminEditTicketController.

1. Admin submits updated user and booking details.
2. If required fields are missing, a 400 status is returned.
3. If no visitors are selected, a 400 status is returned.
4. The system updates the booking details in the database.
5. A 200 status is returned with Booking Update Successful.
6. If an internal error occurs, a 500 status is returned.

**2.4 Admin Ticket Deletion**

Admins can delete a specific booking adventure using the AdminDeleteBookingController.

1. Admin submits the booking details.
2. The system retrieves the booking and removes the specified adventure.
3. The booking is updated and saved.
4. A 200 status is returned with Booking Delete Successful.
5. If an internal error occurs, a 500 status is returned.

**3. Unit Test Cases**

**3.1 Admin Signup Tests**

* **Missing Fields:** Ensures a 422 response if any field is missing.
* **Short Password:** Ensures a 422 response if the password is too short.
* **Password Mismatch:** Ensures a 422 response if passwords do not match.
* **Existing User:** Ensures a 400 response if the admin already exists.

**3.2 Admin Login Tests**

* **Missing Fields:** Ensures a 422 response if email or password is missing.
* **Non-existent User:** Ensures a 400 response if the admin is not found.
* **Incorrect Password:** Ensures a 400 response if password does not match.

**3.3 Forgot Password Tests**

* **Missing Fields:** Ensures a 422 response if any field is missing.
* **Non-existent User:** Ensures a 400 response if the admin email is not found.
* **Password Mismatch:** Ensures a 422 response if passwords do not match.

**3.4 Logout Tests**

* **Invalid Token Handling:** Ensures a 500 response if an invalid token is used.

**3.5 Booking Tests**

* **Missing User Details:** Ensures a 422 response if required fields are missing.
* **No Visitors Selected:** Ensures a 422 response if no visitors are added.
* **Unauthorized Booking Access:** Ensures a 400 response if non-admin email is used.
* **No Bookings Found:** Ensures a 400 response if no bookings exist.