

Reservations – Documenting the Testing Process

#	Endpoint	Purpose	Input Data	Expected Output	Actual Output	Result
1	POST /api/v1/reservations/	Create a valid reservation	{ "user_id": "e0f1c2b3-4d5e-6789-0123-456789abcdef", "place_id": "123e4567-e89b-12d3-a456-426614174000", "start_date": "2025-10-16T14:00:00", "end_date": "2025-10-18T11:00:00", "price": 150.0, "discount": 10.0, "status": "pending", "payment_status": "unpaid" }	Status: 201, reservation returned with all the fields	Status: 201 Response matches expected	✓ Passed
2	POST /api/v1/reservations/	Missing required field <code>user_id</code>	{ "place_id": "...", "start_date": "...", "end_date": "...", "price": 150.0 }	Status: 400, error message	Status: 400, matches expected { "errors": { "user_id": "'user_id' is a required property" }, "message": "Input payload validation failed" }	✓ Passed
3	POST /api/v1/reservations/	End date before start date	{ "user_id": "e0f1c2b3-4d5e-6789-0123-456789abcdef", "place_id": "123e4567-e89b-12d3-a456-426614174000", "start_date": "2025-10-20T14:00:00", "end_date": "2025-10-19T11:00:00", "price": 150.0, "discount": 10.0, "status": "pending", "payment_status": "unpaid" }	Status: 400, error message	Status: 400, matches expected { "error": "Start date cannot be in the past" }	✓ Passed

4	GET /api/v1/reservations/{reservation_id}	Retrieve existing reservation	Valid reservation_id from created reservation	Status: 200, reservation data returned	Status: 200, matches expected	✓ Passed
5	GET /api/v1/reservations/	Retrieve all existing reservations	N/A	Status: 200, array of reservations	Status: 200, matches expected	✓ Passed
6	GET /api/v1/reservations/{reservation_id}/	Retrieve non-existent reservation	Invalid reservation_id	Status: 404, error message	Status 404, matches expected { "error": "Reservation not found" }	✓ Passed
7	GET /api/v1/reservations/	Retrieve all reservations when none exist	When reservation list is None	Status: 404, error message	Status 200, empty array []	✓ Passed
8	PUT /api/v1/reservations/{reservation_id}/	Update reservation dates and status	{ "start_date": "2025-10-17T14:00:00", "end_date": "2025-10-19T11:00:00", "status": completed }	Status: 200, reservation updated	Status: 200, matches expected	✓ Passed
9	PUT /api/v1/reservations/{reservation_id}/	Update with invalid end_date before start_date	{ "start_date": "2025-10-20T14:00:00", "end_date": "2025-10-19T11:00:00" }	Status: 400, error message	Status: 400, matches expected { "error": "Start date cannot be in the past" }	✓ Passed
10	PUT /api/v1/reservations/{reservation_id}/	Update non-updatable field: price	{ "price": 160.0 }	Status: 400, error message	Status: 400, matches expected { "error": "The following fields cannot be updated: price" }	✓ Passed

11	PUT /api/v1/reservations/{reservation_id}/	Update discount to a negative number	{ "discount": -10.0 }	Status: 400, error message	Status: 400, matches expected { "error": "Discount must be a positive number" }	✓ Passed
12	PUT /api/v1/reservations/{reservation_id}/	Update status to invalid value	{ "status": "unknown" }	Status: 400, error message	Status: 400, matches expected { "error": "Invalid status: must be one of ['pending', 'confirmed', 'cancelled', 'completed']" }	✓ Passed