

## EVIDENCIAS EXAMEN FINAL:

### "Se debe soportar las operaciones:

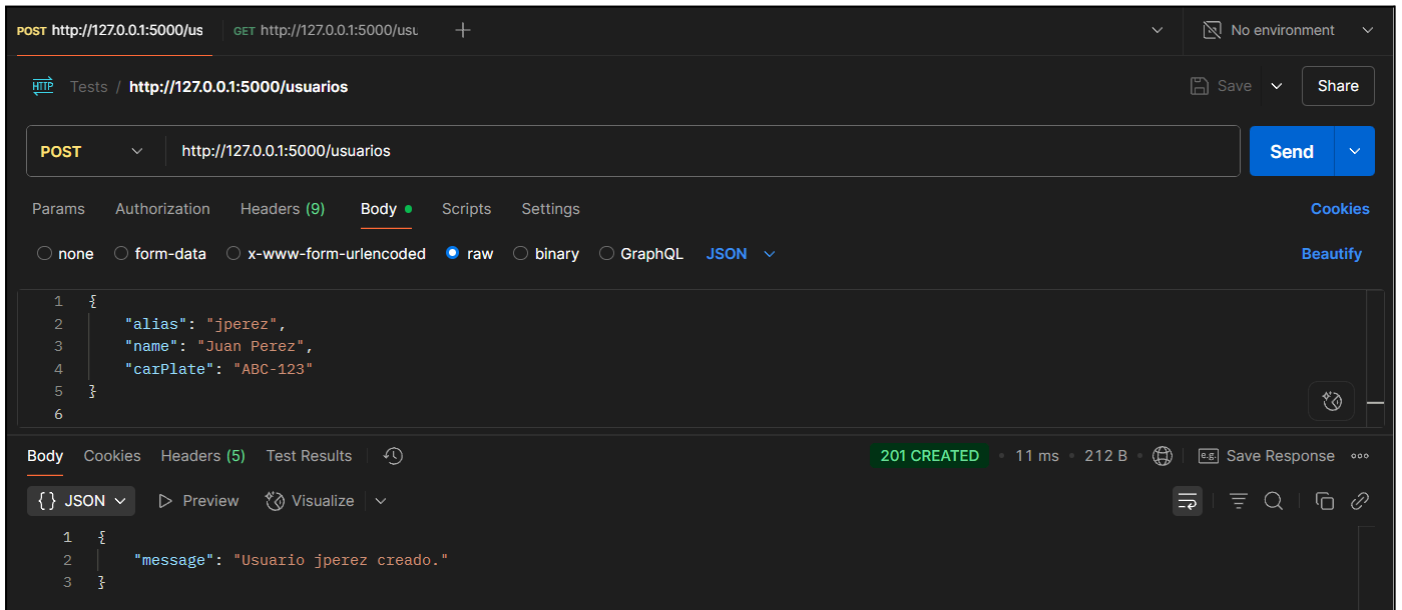
- Usuarios: Creación y listado de usuarios."

### 1. Crear usuarios (POST /usuarios)

**Enunciado:** "Usuarios: Creación"

Body JSON:

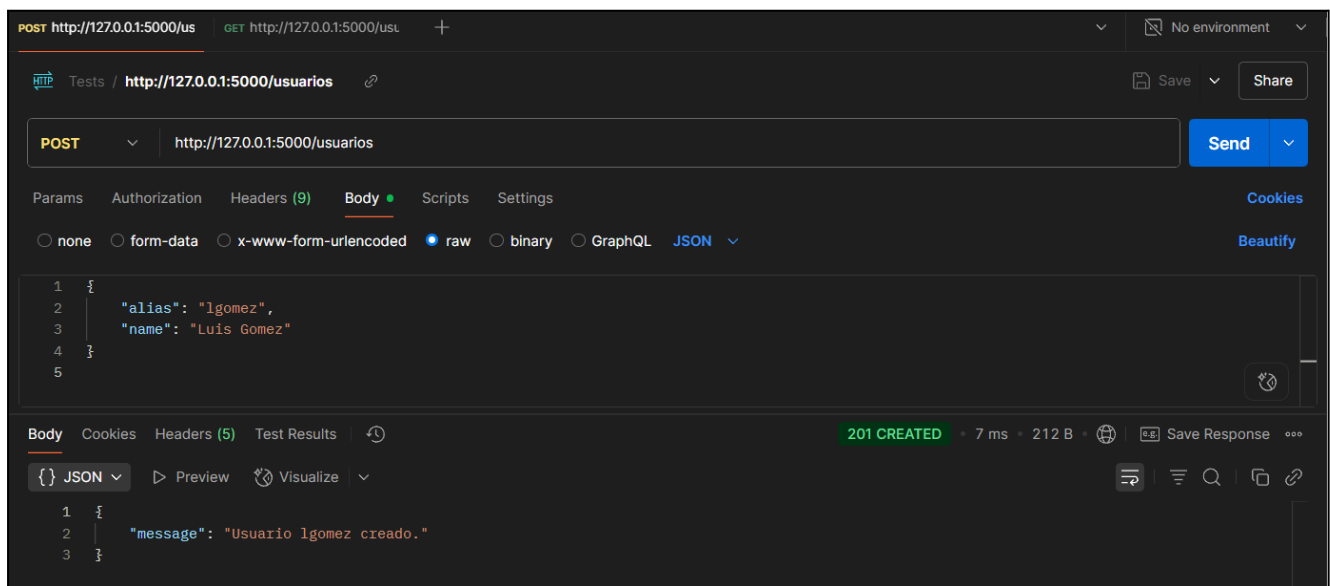
```
{  
  "alias": "jperez",  
  "name": "Juan Perez",  
  "carPlate": "ABC-123"  
}
```



**Usuario sin auto (solo participante):**

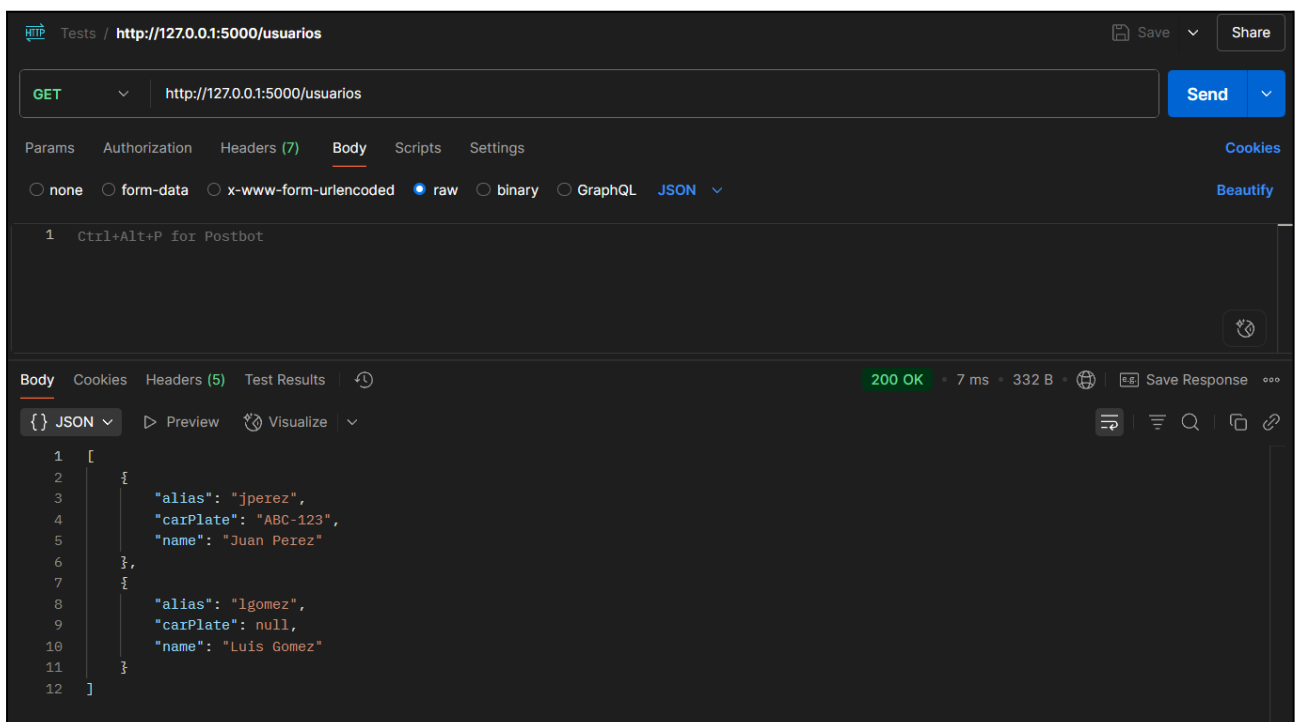
**Body JSON:**

```
{  
  
  "alias": "lgomez",  
  
  "name": "Luis Gomez"  
}
```



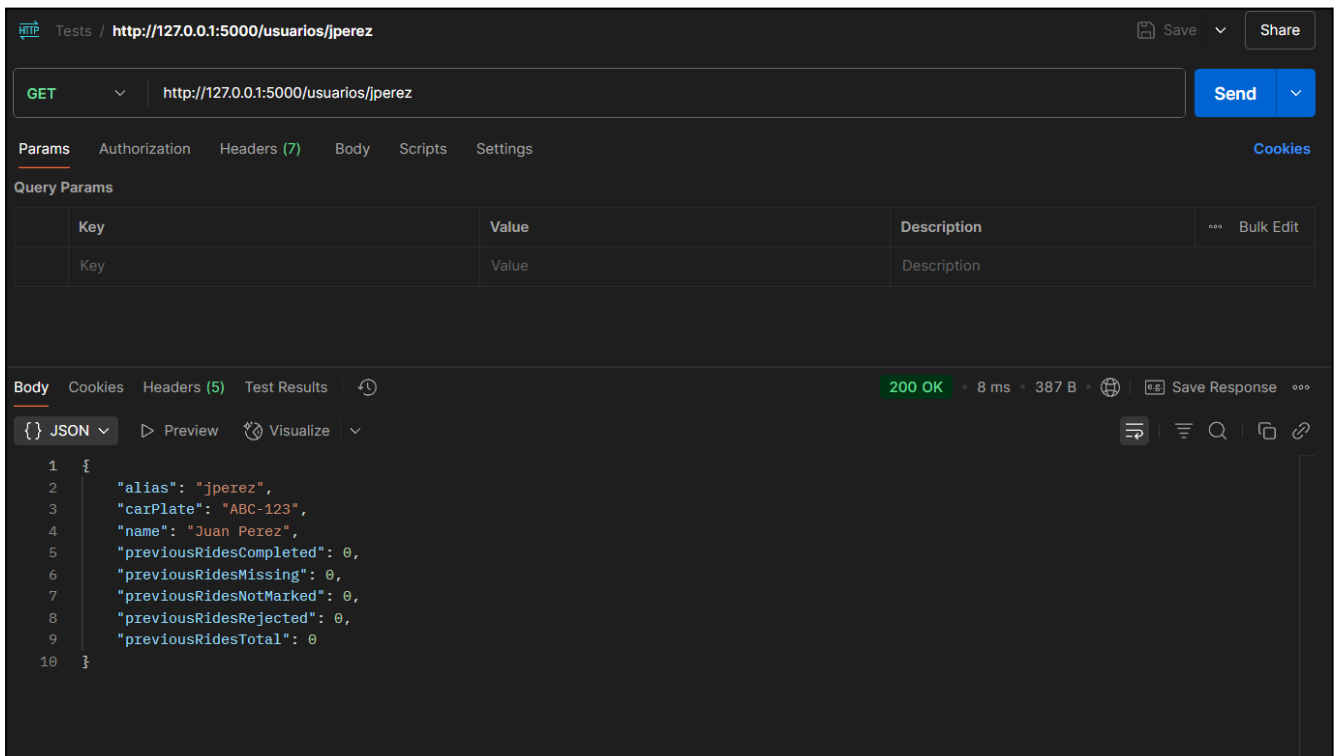
## 2. Listar usuarios (GET /usuarios)

**Enunciado:** “Usuarios: listado de usuarios.”



### 3. Ver datos de usuario (GET /usuarios/{alias})

**Enunciado:** “Ver mi ride (mostrar ranking y participaciones)”



The screenshot shows a REST client interface with the following details:

- URL:** `http://127.0.0.1:5000/usuarios/jperez`
- Method:** `GET`
- Status:** `200 OK` (8 ms, 387 B)
- Response Body (JSON):**

```
{  "alias": "jperez",  "carPlate": "ABC-123",  "name": "Juan Perez",  "previousRidesCompleted": 0,  "previousRidesMissing": 0,  "previousRidesNotMarked": 0,  "previousRidesRejected": 0,  "previousRidesTotal": 0}
```

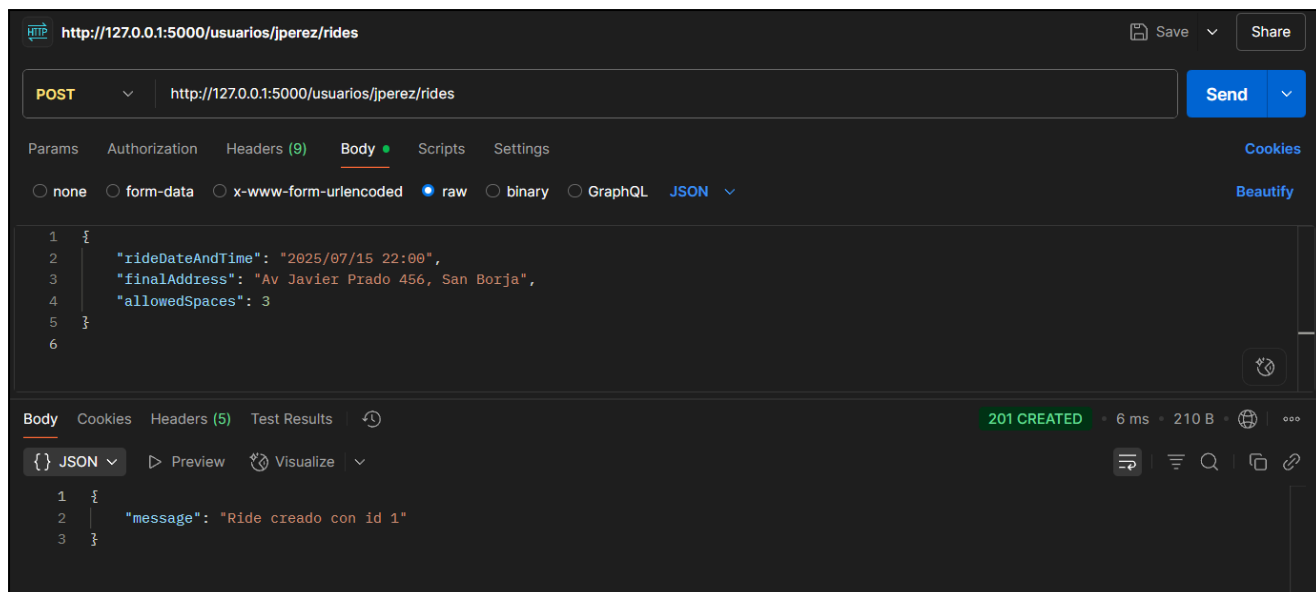
## Según el enunciado

**"- Creación de Rides: Un usuario conductor crea un ride con una hora de salida de la universidad, cuantos asientos tiene disponibles y su destino."**

### 4. Crear ride (POST /usuarios/{alias}/rides)

Body JSON:

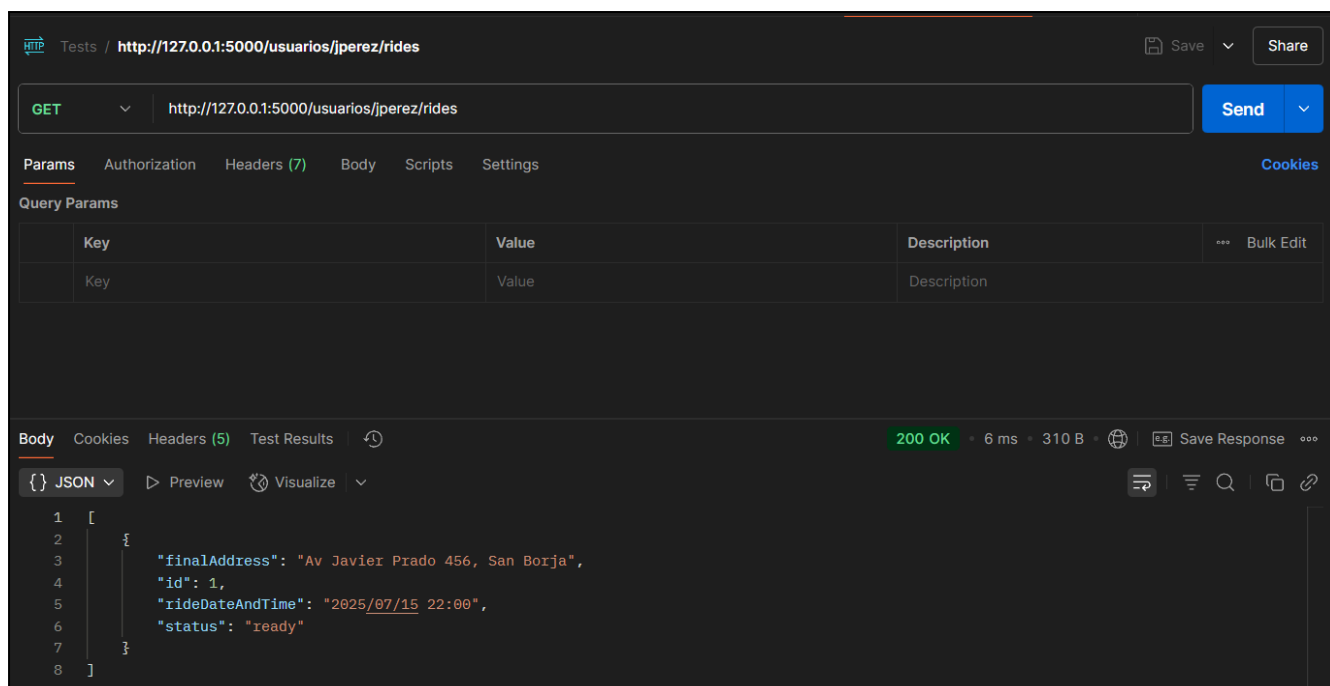
```
{  "rideDateAndTime": "2025/07/15 22:00",  "finalAddress": "Av Javier Prado 456, San Borja",  "allowedSpaces": 3}
```



## Según el enunciado

### "- Listado de Rides Activos"

#### 5. Listar rides creados por el usuario (GET /usuarios/{alias}/rides)



## 6. Ver datos completos del ride (GET /usuarios/{alias}/rides/{rideid})

Enunciado: “Ver mi ride (mostrar participantes con stats y destinos)”

Tests / <http://127.0.0.1:5000/usuarios/jperez/rides/1> Save Share

GET <http://127.0.0.1:5000/usuarios/jperez/rides/1> Send

Params Authorization Headers (7) Body Scripts Settings Cookies

Query Params

Key	Value	Description	Bulk Edit
Key	Value	Description	

Body Cookies Headers (5) Test Results 200 OK • 6 ms • 366 B Save Response

JSON Preview Visualize

```
1 {
2   "ride": {
3     "driver": "jperez",
4     "finalAddress": "Av Javier Prado 456, San Borja",
5     "id": 1,
6     "participants": [],
7     "rideDateAndTime": "2025/07/15 22:00",
8     "status": "ready"
9   }
10 }
```

## Según el enunciado

"- Solicitar unirse a un ride indicando destino y número de asientos."

## 7. Solicitar unirse (POST /usuarios/{alias}/rides/{rideid}/requestToJoin/{alias})

Tests / <http://127.0.0.1:5000/usuarios/jperez/rides/1/requestToJoin/lgomez> Save Share

POST <http://127.0.0.1:5000/usuarios/jperez/rides/1/requestToJoin/lgomez> Send

Params Authorization Headers (9) Body Scripts Settings Cookies Beautify

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON

```
1 {
2   "destination": "Av Aramburú 245, Surquillo",
3   "occupiedSpaces": 1
4 }
5
```

Body Cookies Headers (5) Test Results 200 OK • 10 ms • 223 B Save Response

JSON Preview Visualize

```
1 {
2   "message": "lgomez solicitó unirse al ride 1."
3 }
```

## Según el enunciado

"- Aceptar la solicitud de unirse a un ride."

### 8. Aceptar participante (POST /usuarios/{alias}/rides/{rideid}/accept/{participant})

The screenshot shows a REST client interface with the following details:

- Method:** POST
- URL:** http://127.0.0.1:5000/usuarios/jperez/rides/1/accept/lgoomez
- Status:** 200 OK
- Response Body (JSON):**

```
{  "message": "lgomez aceptado."}
```

### 9. Revisar cómo cambió (GET /usuarios/jperez/rides/1)

Espera: participants ahora tiene a lgomez con:

The screenshot shows a REST client interface with the following details:

- Method:** GET
- URL:** http://127.0.0.1:5000/usuarios/jperez/rides/1
- Status:** 200 OK
- Response Body (JSON):**

```
{  "ride": {    "driver": "jperez",    "finalAddress": "Av Javier Prado 456, San Borja",    "id": 1,    "participants": [      {        "confirmation": true,        "destination": "Av Aramburú 245, Surquillo",        "occupiedSpaces": 1,        "participant": {          "alias": "lgomez",          "previousRidesCompleted": 0,          "previousRidesMissing": 0,          "previousRidesNotMarked": 0,          "previousRidesRejected": 0,          "previousRidesTotal": 0        }      }    ],    "status": "confirmed"  },  "rideDateAndTime": "2025/07/15 22:00",  "status": "ready"}
```

# Según el enunciado

"- Iniciar un ride (indicar riderParticipants inprogress)"

## 10. Iniciar ride (POST /usuarios/{alias}/rides/{rideid}/start)

The screenshot shows a REST client interface with the following details:

- URL:** `http://127.0.0.1:5000/usuarios/jperez/rides/1/start`
- Method:** `POST`
- Response:** `200 OK` with a response time of 6 ms and a body size of 201 B.
- Body:** A JSON object: 

```
{ 1: { 2: "message": "Ride 1 iniciado." 3: }
```

**Espera:** ahora status = inprogress y lgomez status = inprogress.

The screenshot shows a REST client interface with the following details:

- URL:** `http://127.0.0.1:5000/usuarios/jperez/rides/1`
- Method:** `GET`
- Response:** `200 OK` with a response time of 6 ms and a body size of 792 B.
- Body:** A JSON object representing ride details: 

```
1: { 2: "ride": { 3: "driver": "jperez", 4: "finalAddress": "Av Javier Prado 456, San Borja", 5: "id": 1, 6: "participants": [ 7: { 8: "confirmation": true, 9: "destination": "Av Aramburú 245, Surquillo", 10: "occupiedSpaces": 1, 11: "participant": { 12: "alias": "lgomez", 13: "previousRidesCompleted": 0, 14: "previousRidesMissing": 0, 15: "previousRidesNotMarked": 0, 16: "previousRidesRejected": 0, 17: "previousRidesTotal": 0 18: }, 19: "status": "inprogress" 20: }, 21: ], 22: "rideDateAndTime": "2025/07/15 22:00", 23: "status": "inprogress" 24: } 25: }
```

## Según el enunciado

"- Bajar del ride (lo hace el participante)."

### 11. Bajarse del ride (POST /usuarios/{alias}/rides/{rideid}/unloadParticipant)

The screenshot shows a REST client interface with the following details:

- URL:** `http://127.0.0.1:5000/usuarios/lgomez/rides/1/unloadParticipant`
- Method:** POST
- Response Status:** 200 OK
- Response Time:** 6 ms
- Response Size:** 216 B
- Response Body (JSON):**

```
{  "message": "lgomez se bajó del ride 1."}
```

## Según el enunciado

"- Terminar el ride (lo hace el conductor)."

### 12. Terminar ride (POST /usuarios/{alias}/rides/{rideid}/end)

The screenshot shows a REST client interface with the following details:

- URL:** `http://127.0.0.1:5000/usuarios/jperez/rides/1/end`
- Method:** POST
- Response Status:** 200 OK
- Response Time:** 9 ms
- Response Size:** 202 B
- Response Body (JSON):**

```
{  "message": "Ride 1 terminado."}
```



# Según el enunciado

"Ver mi ride (mostrar stats)"

## 13. Consultar stats (GET /usuarios/lgoomez)

The screenshot shows a REST client interface with the following details:

- URL:** `http://127.0.0.1:5000/usuarios/lgoomez`
- Method:** `GET`
- Response Status:** `200 OK` (6 ms, 382 B)
- Response Body (JSON):**

```
{  "alias": "lgomez",  "carPlate": null,  "name": "Luis Gomez",  "previousRidesCompleted": 1,  "previousRidesMissing": 0,  "previousRidesNotMarked": 0,  "previousRidesRejected": 0,  "previousRidesTotal": 2}
```

**Espera:** previousRidesTotal incrementó a 1, previousRidesCompleted incrementó a 1.

## TEST COVERAGE:

**python -m coverage run -m unittest discover tests**

**python -m coverage report**

```
(venv) PS C:\Users\luisr\ExFinalIng\examenfinaling> python -m coverage run -m unittest discover tests
.....
-----
Ran 6 tests in 0.002s

OK
(venv) PS C:\Users\luisr\ExFinalIng\examenfinaling> python -m coverage report
Name                               Stmts  Miss  Cover
-----
models.py                          33      0  100%
tests\test_app.py                   34      1   97%
-----
TOTAL                               67      1   99%
(venv) PS C:\Users\luisr\ExFinalIng\examenfinaling> 
```

**Repositorio GITHUB:**

<https://github.com/fabianorifer/examenfinaling>