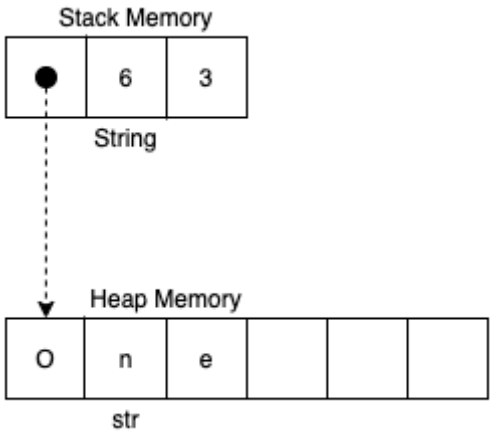
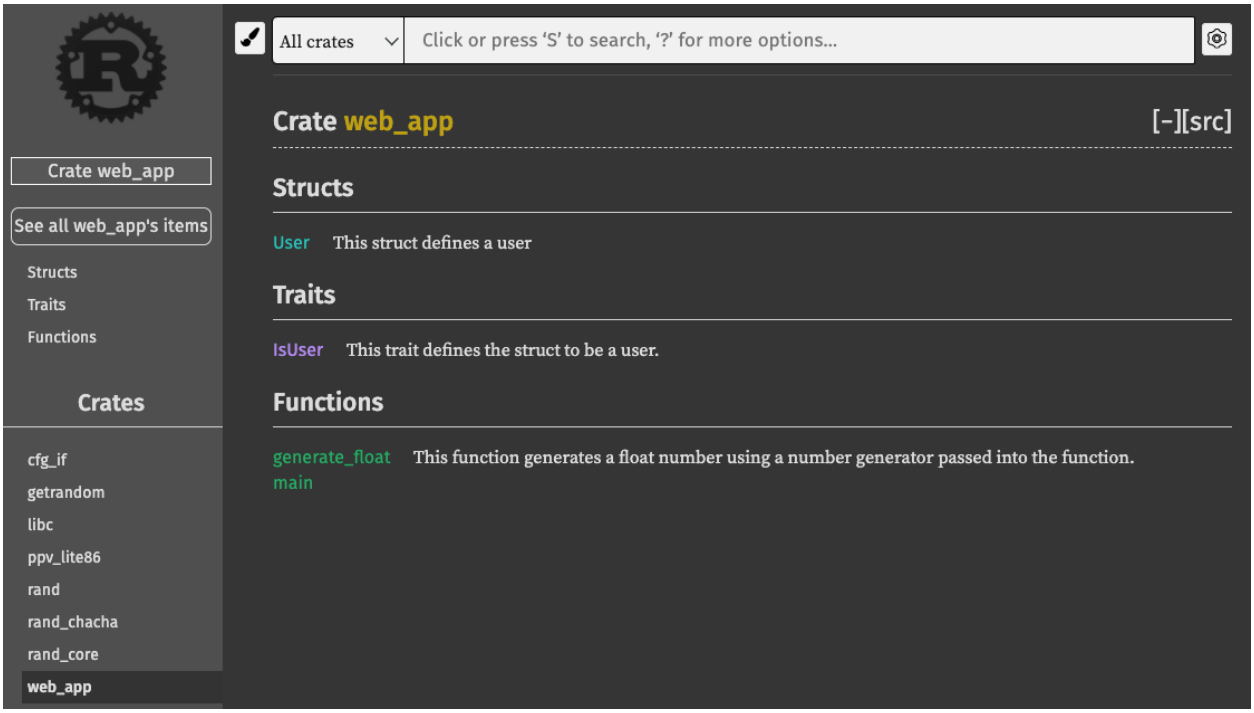
Chapter 1: Quick Introduction to Rust



Chapter 2: Designing Your Web Application in Rust



Struct web_app::User [-][src]

```
struct User {
    name: String,
    age: i8,
}
```

[-] This struct defines a user

Attributes

- name (String): the name of the user
- age (i8): the age of the user

Fields

```
name: String
age: i8
```

Trait Implementations

```
impl IsUser for User
```
[src]

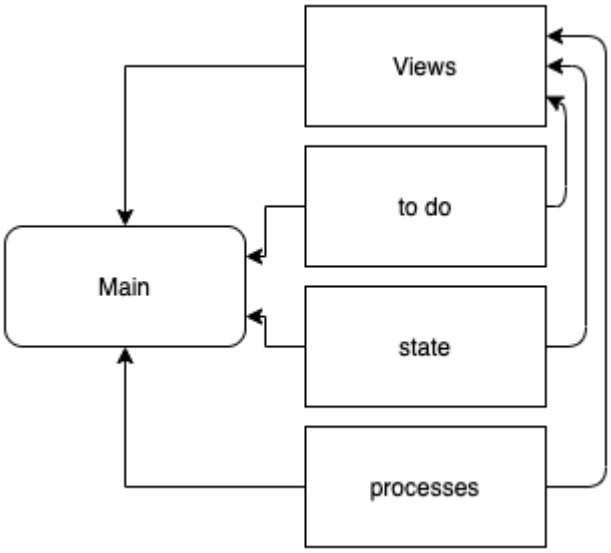
Auto Trait Implementations

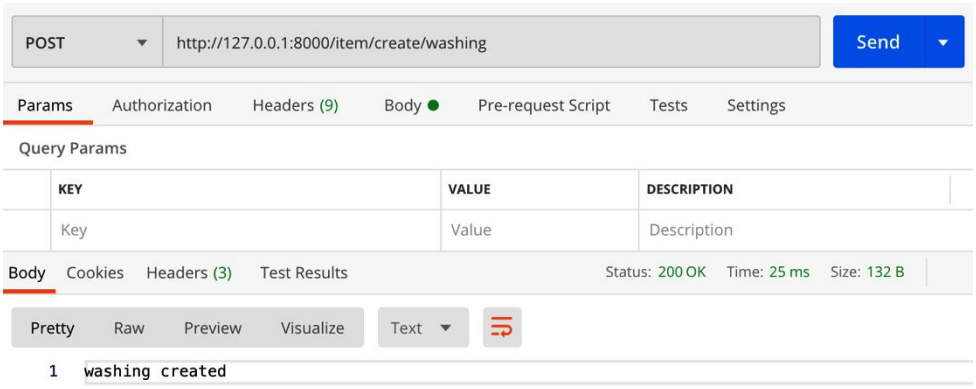
```
impl RefUnwindSafe for User
impl Send for User
impl Sync for User
impl Unpin for User
impl UnwindSafe for User
```

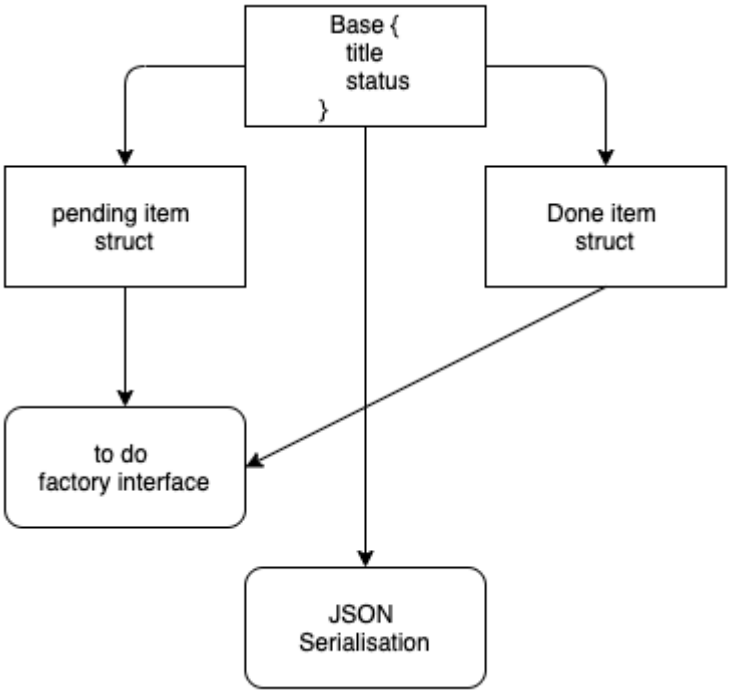
Chapter 3: Handling HTTP Requests

No images

Chapter 4: Processing HTTP Requests







PUT http://127.0.0.1:8000/item/edit Send

Params Authorization Headers (9) Body Pre-request Script Tests Settings

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

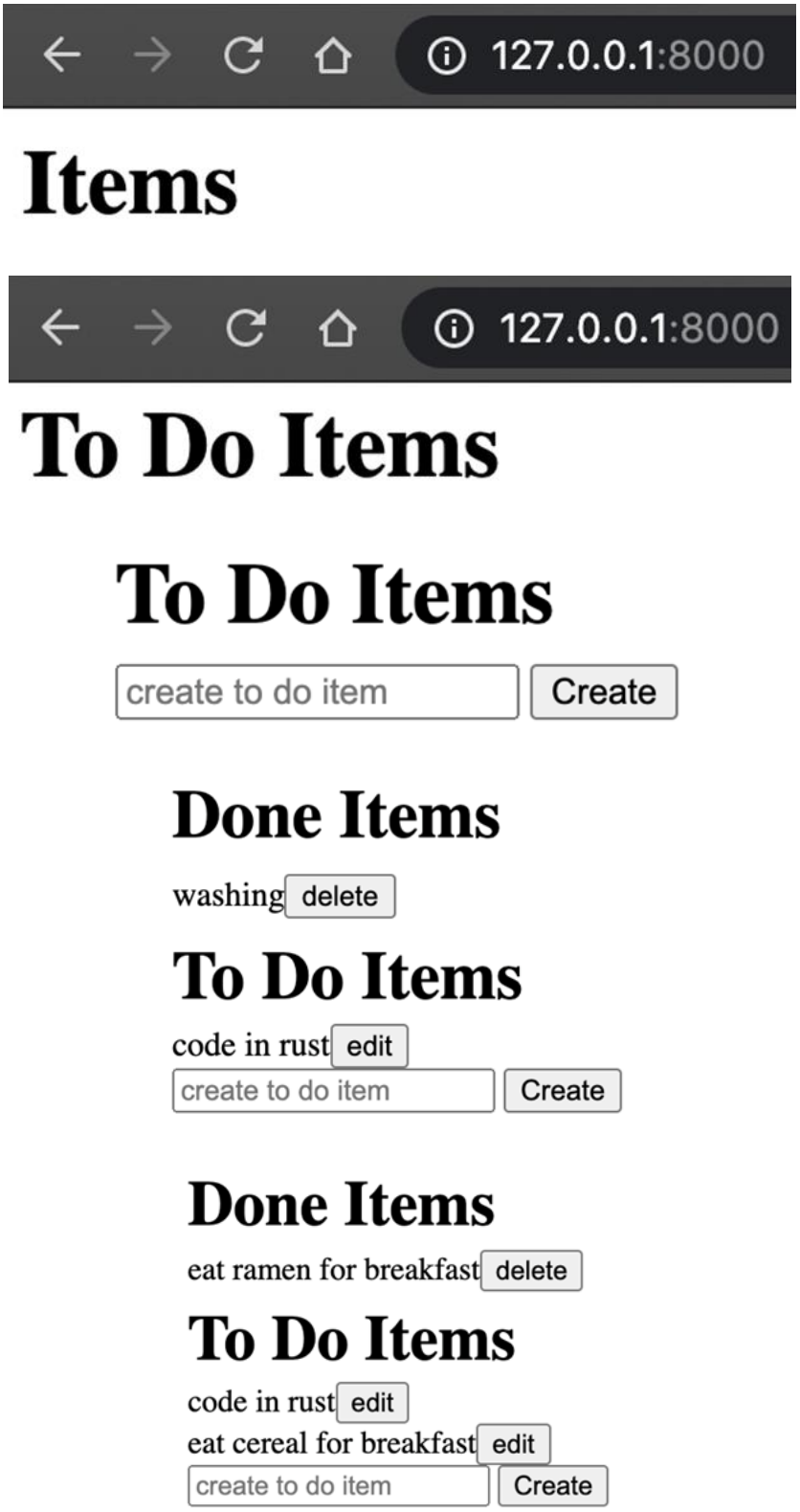
```
{
    "title": "washing",
    "status": "done"
}
```

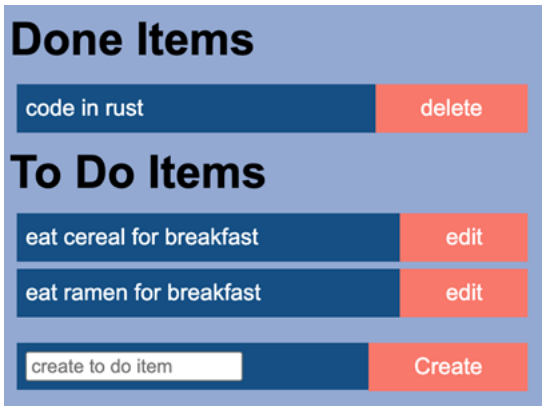
Body Cookies Headers (3) Test Results Status: 200 OK Time: 25 ms Size: 266 B

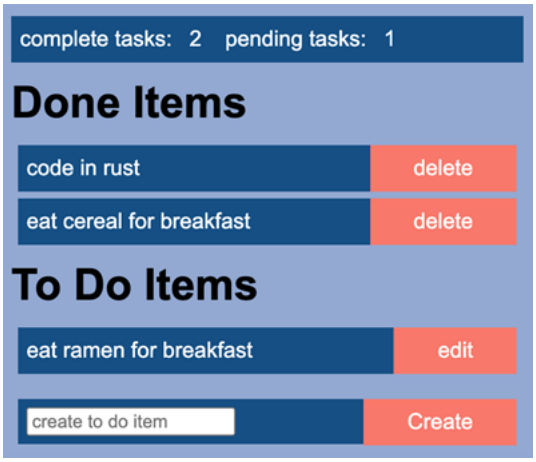
Pretty Raw Preview Visualize JSON


```
{
    "pending_items": [
        {
            "title": "code in rust",
            "status": "pending"
        }
    ],
    "done_items": [
        {
            "title": "washing",
            "status": "done"
        }
    ],
    "pending_item_count": 1,
    "done_item_count": 1
}
```

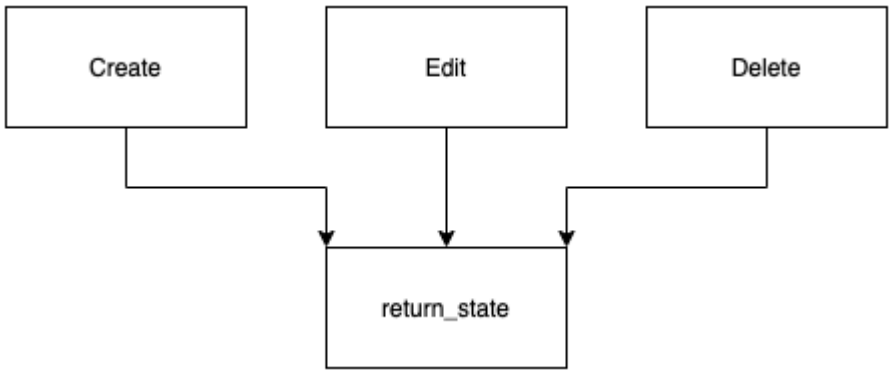
Chapter 5: Displaying Content in the Browser

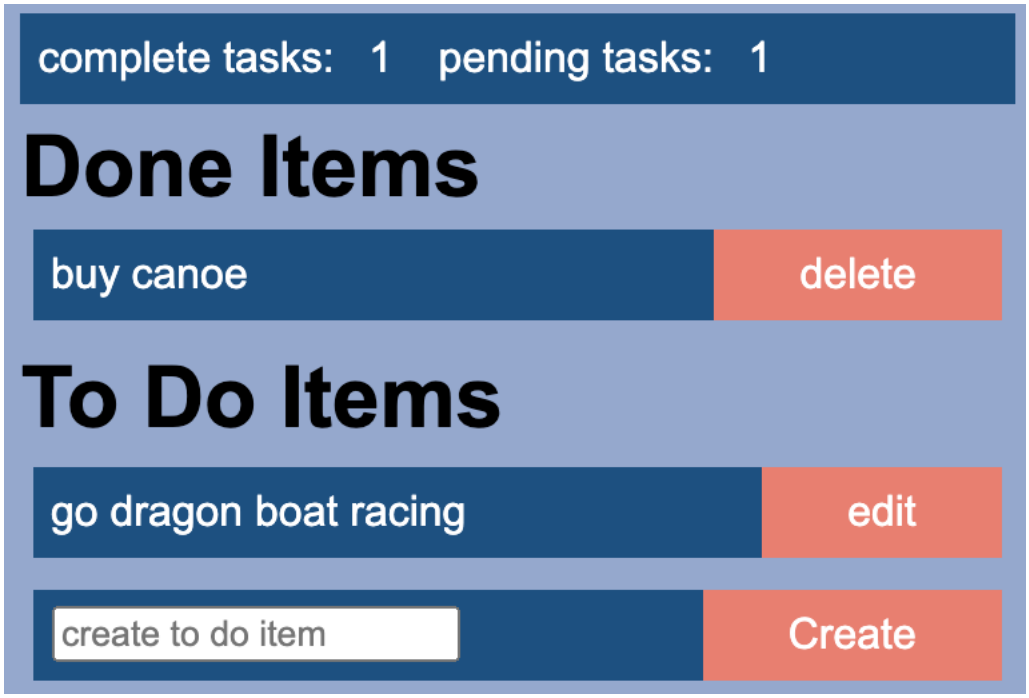



Done Items
code in rust
delete
To Do Items
eat cereal for breakfast
edit
eat ramen for breakfast
edit
create to do item
Create

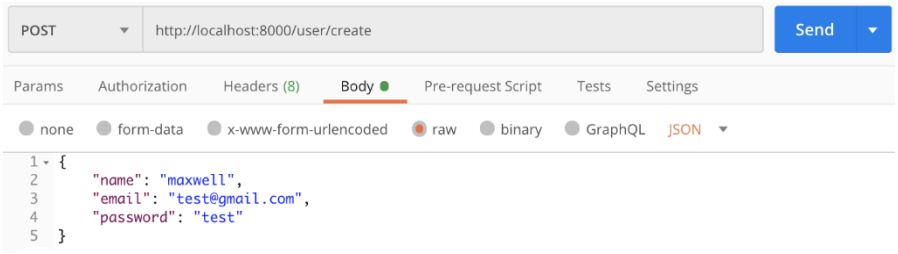

complete tasks: 2 pending tasks: 1
Done Items
code in rust
delete
eat cereal for breakfast
delete
To Do Items
eat ramen for breakfast
edit
create to do item
Create

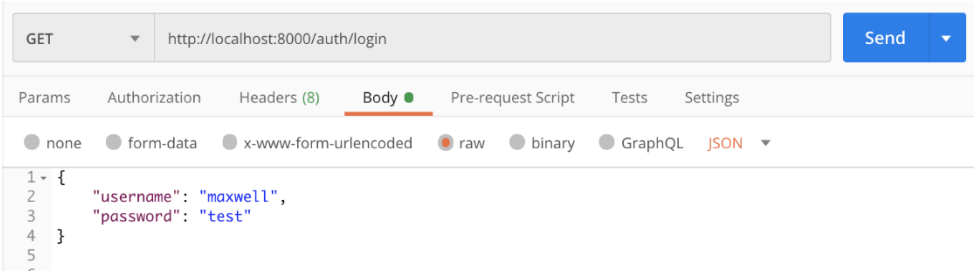
Chapter 6: Data Persistence with PostgreSQL

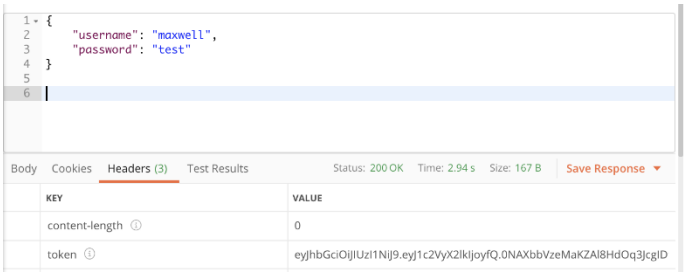




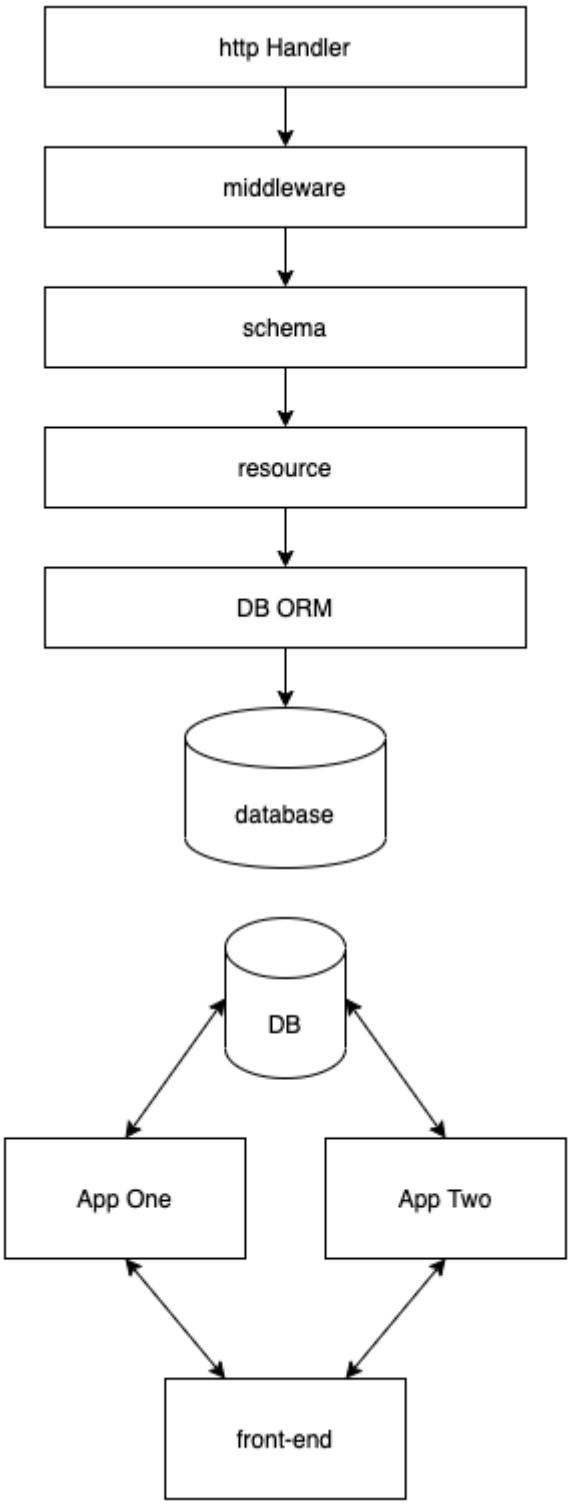
Chapter 7: Managing User Sessions

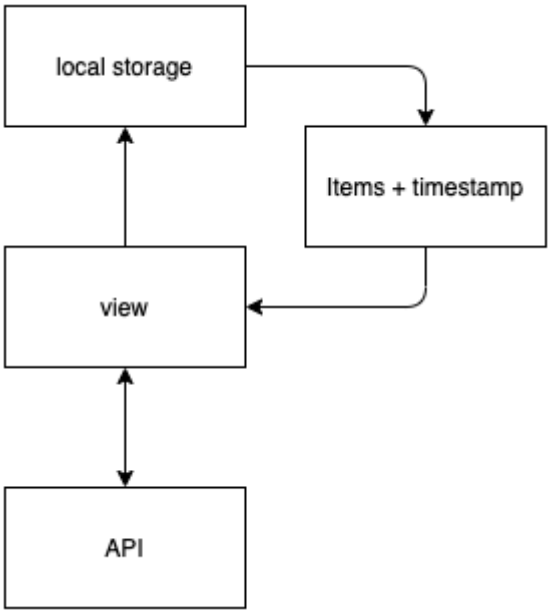




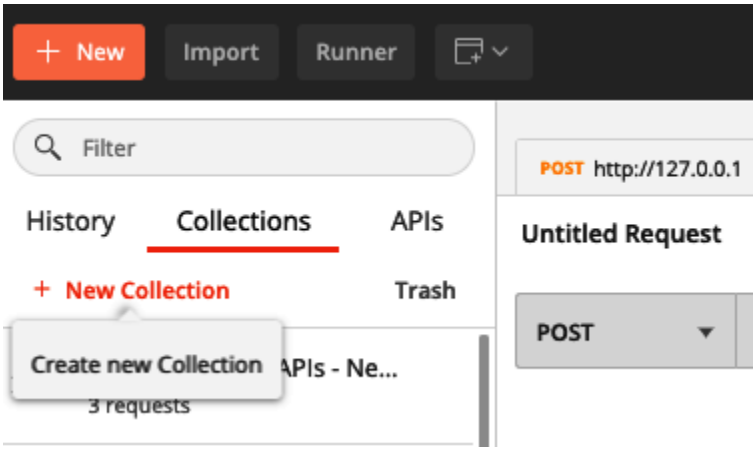


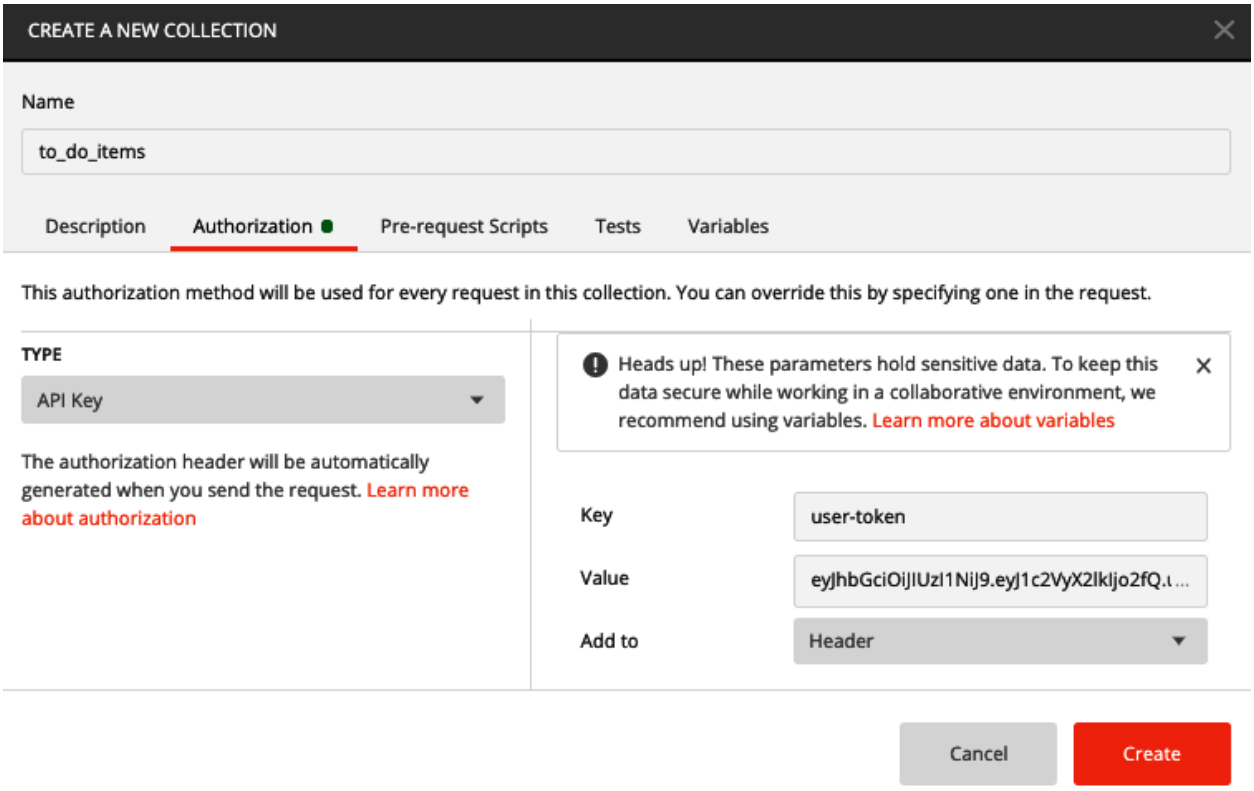
Chapter 8: Building RESTful Services

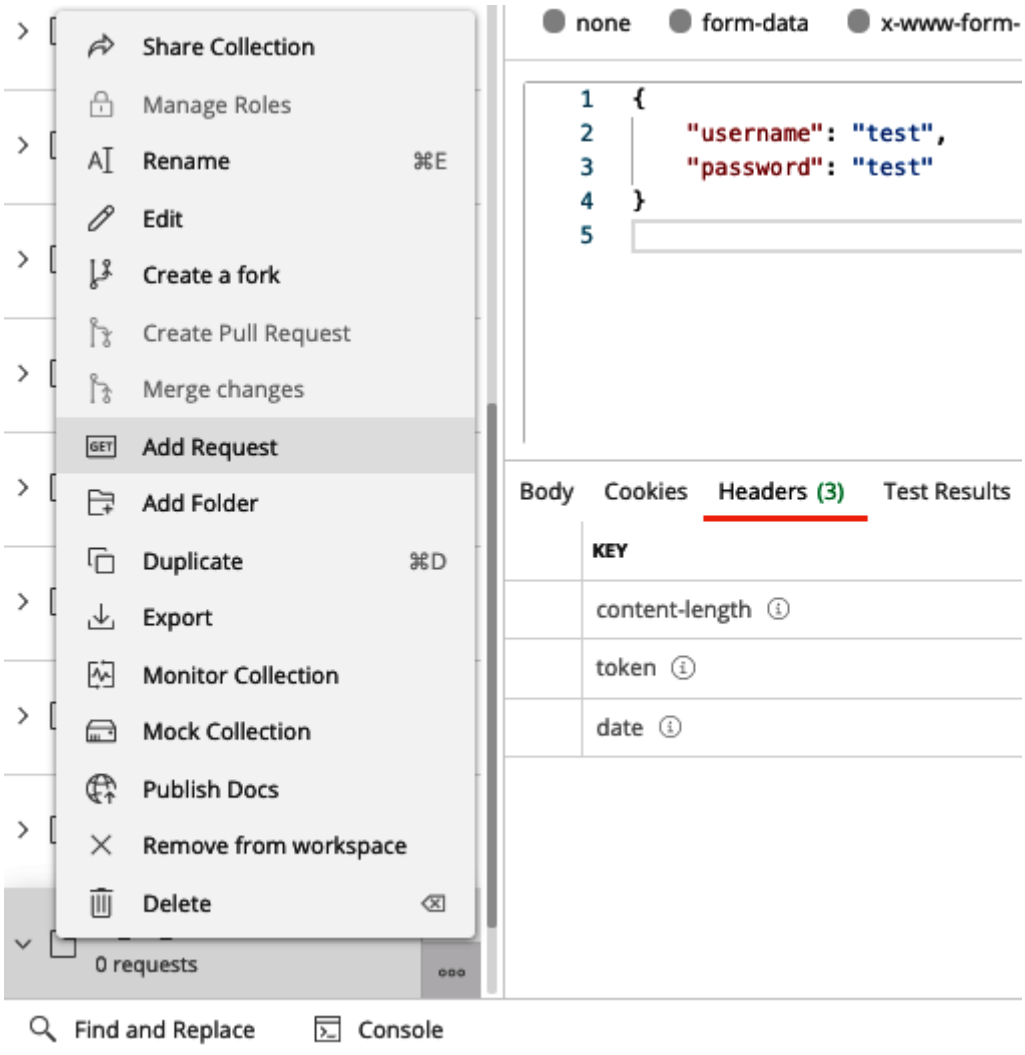


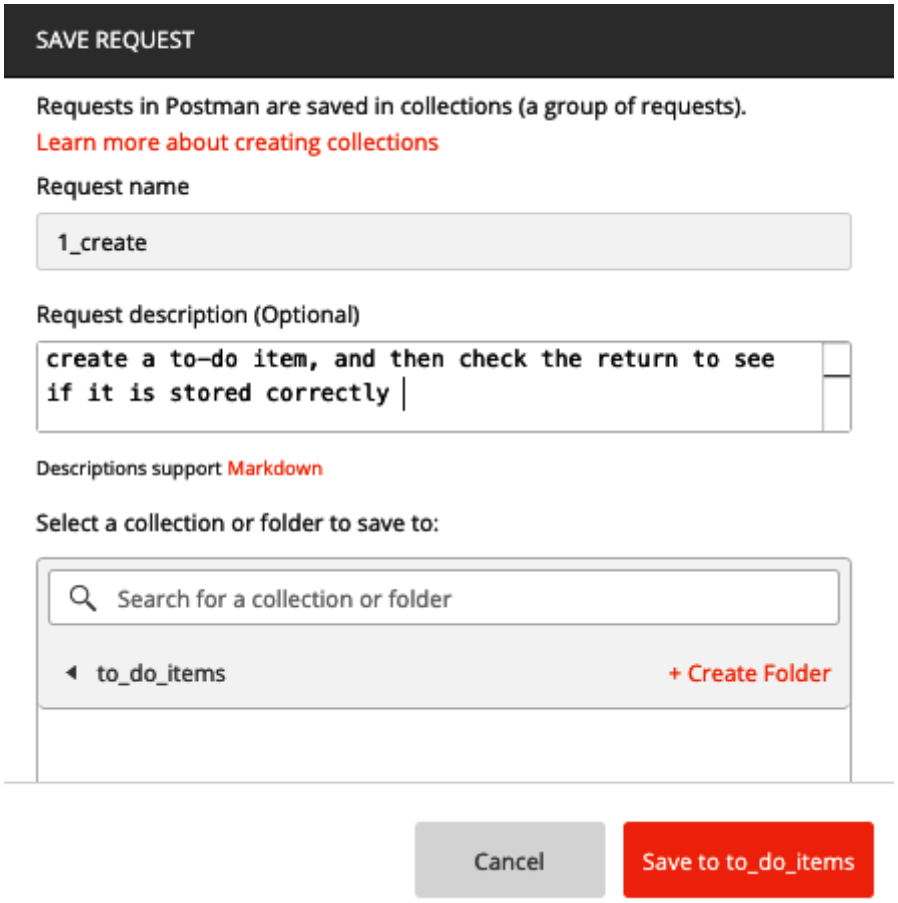

local storage
Items + timestamp
view
API

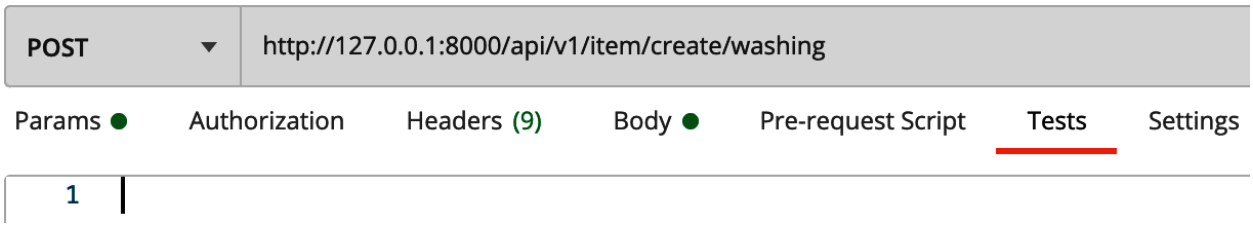
Chapter 9: Testing Our Application Endpoints and Component

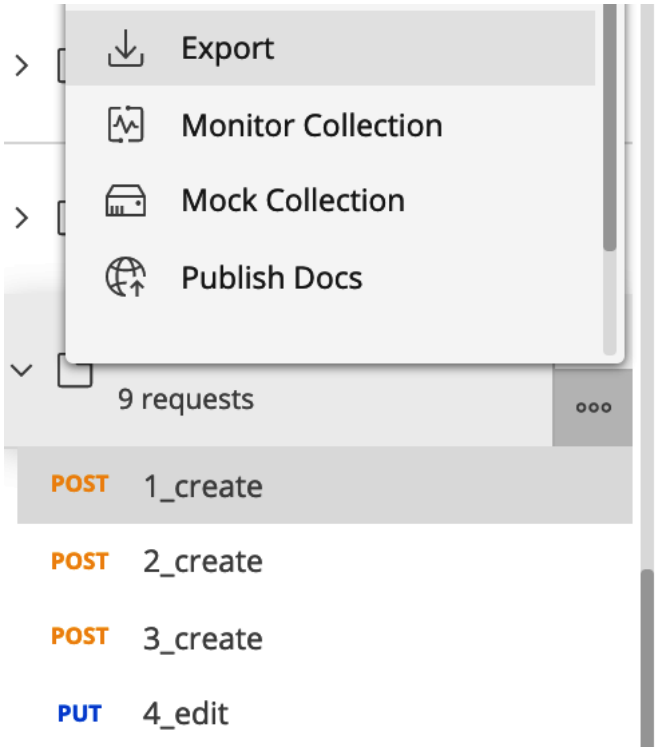





Share Collection
Manage Roles
Rename ⌘E
Edit
Create a fork
Create Pull Request
Merge changes
Add Request
Add Folder
Duplicate ⌘D
Export
Monitor Collection
Mock Collection
Publish Docs
Remove from workspace
Delete
0 requests
none
form-data
x-www-form-
{
"username": "test",
"password": "test"
}
Body
Cookies
Headers (3)
Test Results
KEY
content-length
token
date
Find and Replace
Console

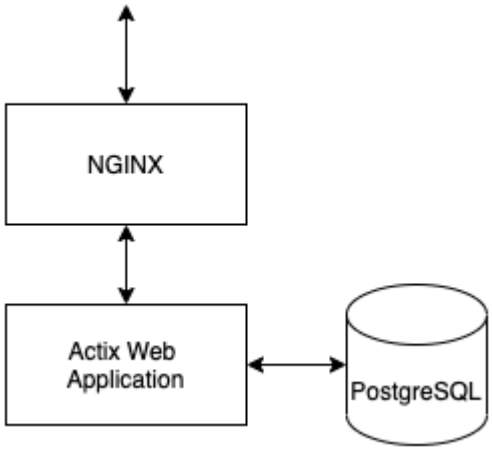

SAVE REQUEST
Requests in Postman are saved in collections (a group of requests).
Learn more about creating collections
Request name
1_create
Request description (Optional)
create a to-do item, and then check the return to see if it is stored correctly
Descriptions support Markdown
Select a collection or folder to save to:
Search for a collection or folder
to_do_items
+ Create Folder
Cancel
Save to to_do_items

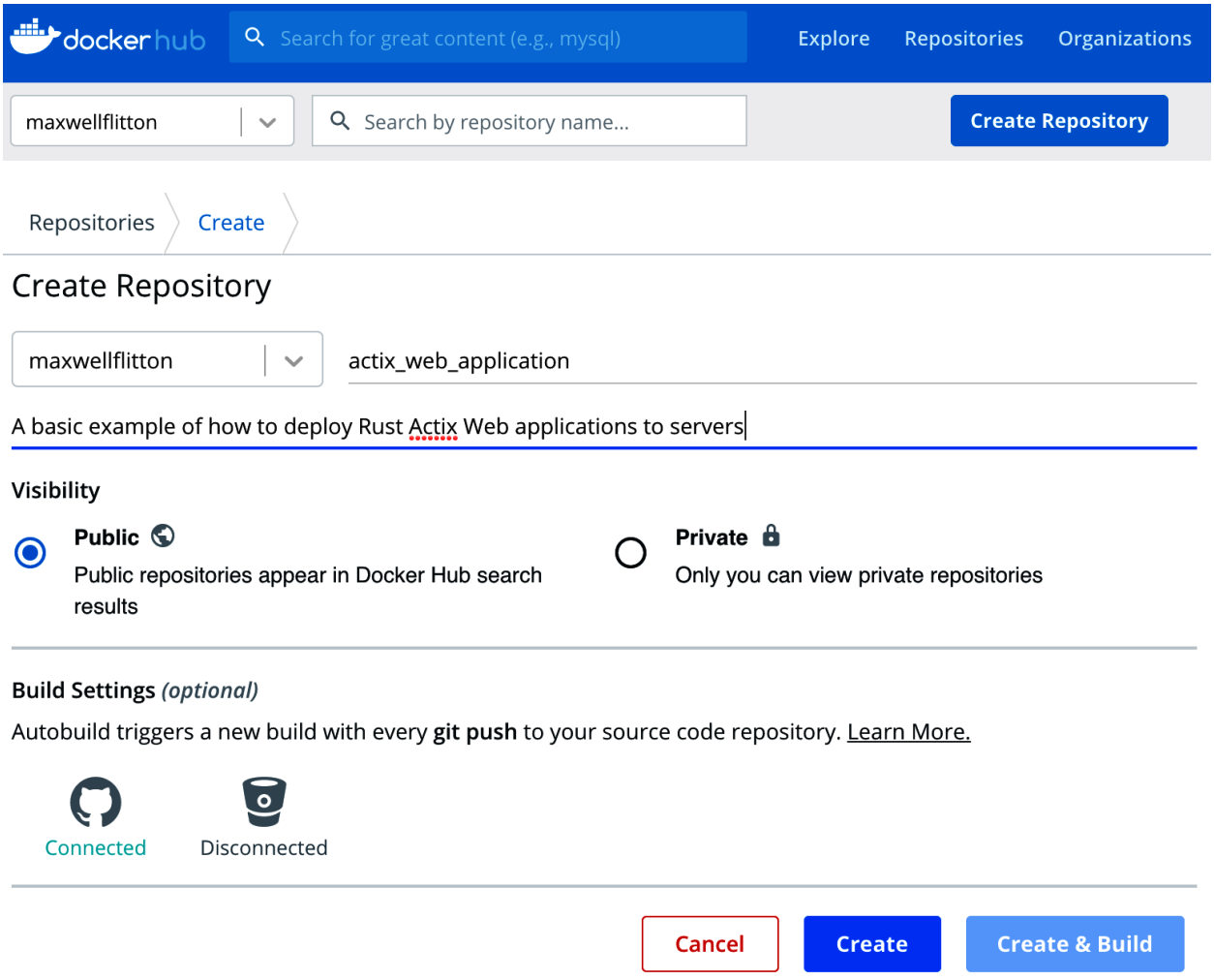

POST
http://127.0.0.1:8000/api/v1/item/create/washing
Params
Authorization
Headers (9)
Body
Pre-request Script
Tests
Settings
1



	executed	failed
iterations	1	0
requests	8	0
test-scripts	8	0
prerequest-scripts	0	0
assertions	64	0
total run duration: 775ms		
total data received: 1.05KB (approx)		
average response time: 64ms [min: 44ms, max: 115ms, s.d.: 21ms]		

Chapter 10: Deploying Our Application on AWS





maxwellflitton / **actix_web_application**

A basic example of how to deploy Rust Actix Web applications to servers

Last pushed: in 2 minutes

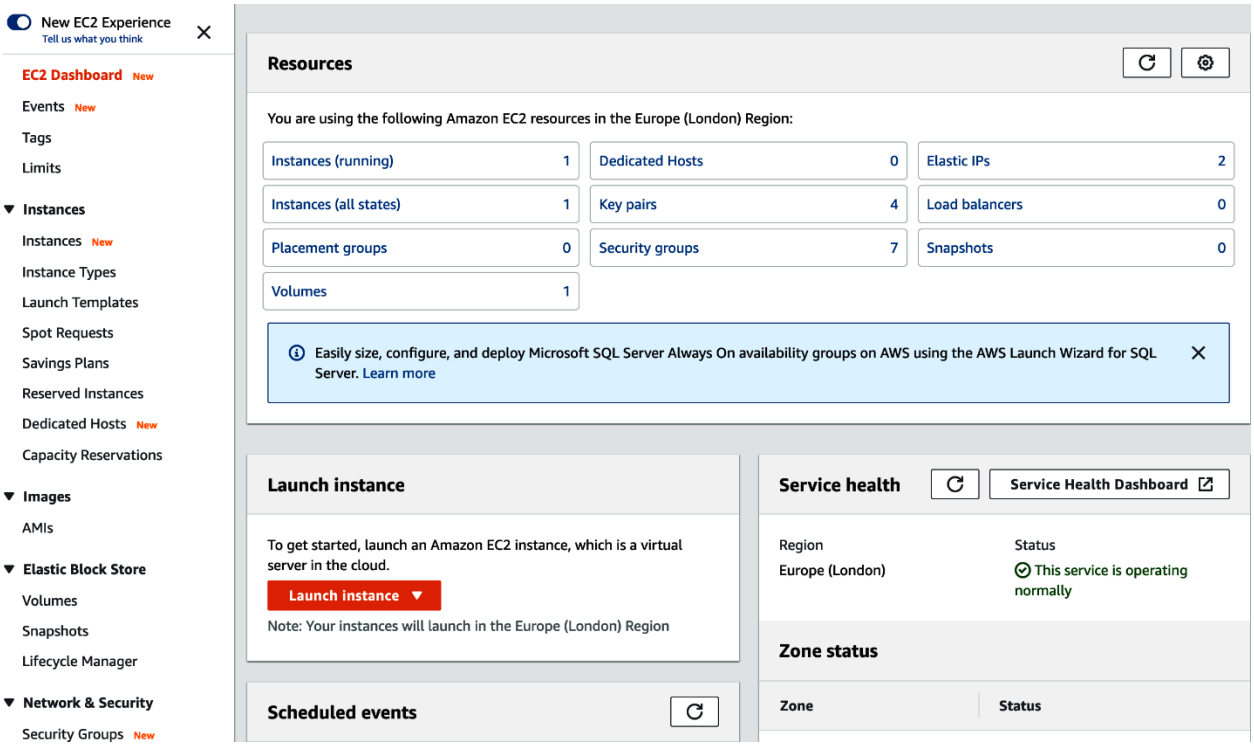
Tags and Scans

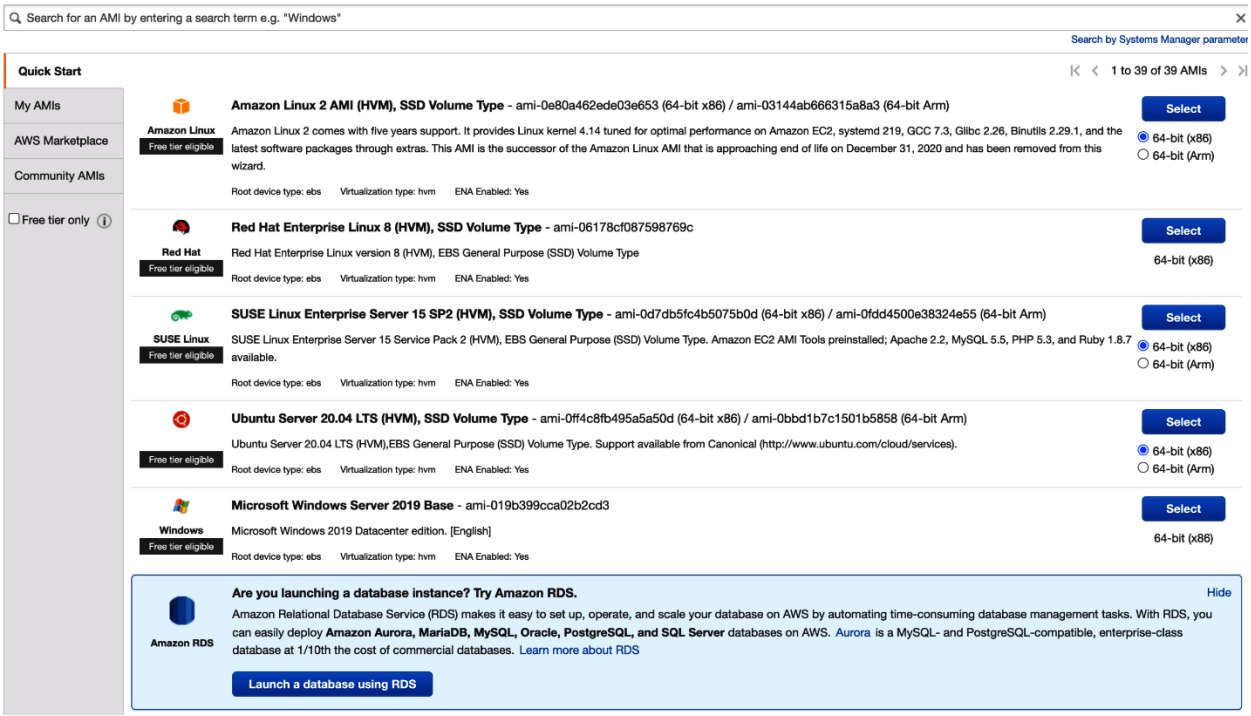
VULNERABILITY SCANNING - **DISABLED**
Enable

This repository contains 1 tag(s).

TAG	OS	PULLED	PUSHED
latest		in 2 minutes	in 2 minutes

See all





Filter by: All instance families | Current generation | Show/Hide Columns

Currently selected: t2.micro (- ECUs, 1 vCPUs, 2.5 GHz, -, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
☐	t2	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
☑	t2	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
☐	t2	t2.small	1	2	EBS only	-	Low to Moderate	Yes
☐	t2	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
☐	t2	t2.large	2	8	EBS only	-	Low to Moderate	Yes
☐	t2	t2.xlarge	4	16	EBS only	-	Moderate	Yes
☐	t2	t2.2xlarge	8	32	EBS only	-	Moderate	Yes
☐	t3	t3.nano	2	0.5	EBS only	Yes	Up to 5 Gigabit	Yes
☐	t3	t3.micro	2	1	EBS only	Yes	Up to 5 Gigabit	Yes
☐	t3	t3.small	2	2	EBS only	Yes	Up to 5 Gigabit	Yes
☐	t3	t3.medium	2	4	EBS only	Yes	Up to 5 Gigabit	Yes
☐	t3	t3.large	2	8	EBS only	Yes	Up to 5 Gigabit	Yes
☐	t3	t3.xlarge	4	16	EBS only	Yes	Up to 5 Gigabit	Yes
☐	t3	t3.2xlarge	8	32	EBS only	Yes	Up to 5 Gigabit	Yes
☐	t3a	t3a.nano	2	0.5	EBS only	Yes	Up to 5 Gigabit	Yes
☐	t3a	t3a.micro	2	1	EBS only	Yes	Up to 5 Gigabit	Yes
☐	t3a	t3a.small	2	2	EBS only	Yes	Up to 5 Gigabit	Yes
☐	t3a	t3a.medium	2	4	EBS only	Yes	Up to 5 Gigabit	Yes

Cancel | Previous | Review and Launch | Next: Configure Instance Details

Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI.

Create a new key pair

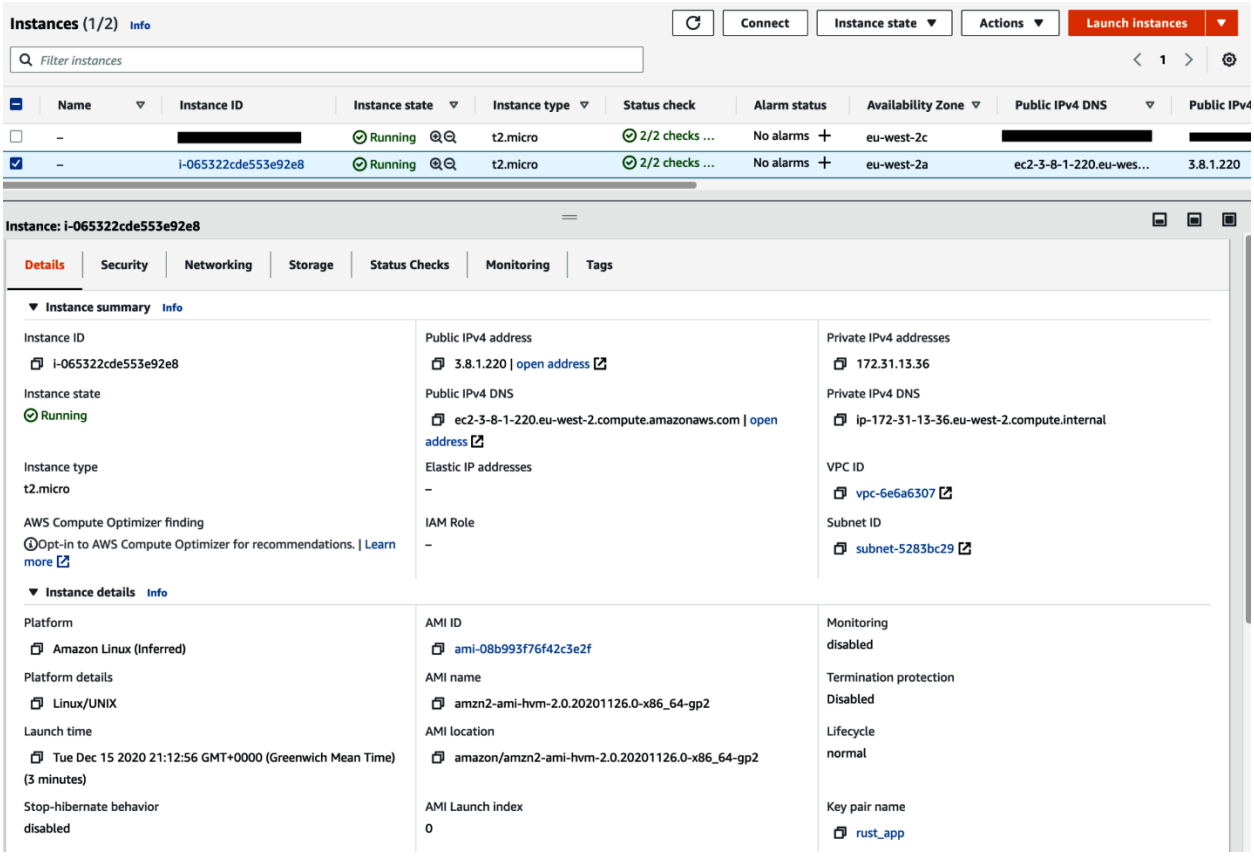
Key pair name

rust_app

Download Key Pair

You have to download the **private key file** (*.pem file) before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.

Cancel Launch Instances


Instances (1/2) Info
Connect
Instance state
Actions
Launch instances
Filter instances
1
Name
Instance ID
Instance state
Instance type
Status check
Alarm status
Availability Zone
Public IPv4 DNS
Public IPv4
–
Running
t2.micro
2/2 checks ...
No alarms
eu-west-2c
–
i-065322cde553e92e8
Running
t2.micro
2/2 checks ...
No alarms
eu-west-2a
ec2-3-8-1-220.eu-wes...
3.8.1.220
Instance: i-065322cde553e92e8
Details
Security
Networking
Storage
Status Checks
Monitoring
Tags
Instance summary Info
Instance ID
i-065322cde553e92e8
Public IPv4 address
3.8.1.220 | open address
Private IPv4 addresses
172.31.13.36
Instance state
Running
Public IPv4 DNS
ec2-3-8-1-220.eu-west-2.compute.amazonaws.com | open address
Private IPv4 DNS
ip-172-31-13-36.eu-west-2.compute.internal
Instance type
t2.micro
Elastic IP addresses
–
VPC ID
vpc-6e6a6307
AWS Compute Optimizer finding
Opt-in to AWS Compute Optimizer for recommendations. | Learn more
IAM Role
–
Subnet ID
subnet-5283bc29
Instance details Info
Platform
Amazon Linux (Inferred)
AMI ID
ami-08b993f76f42c3e2f
Monitoring
disabled
Platform details
Linux/UNIX
AMI name
amzn2-ami-hvm-2.0.20201126.0-x86_64-gp2
Termination protection
Disabled
Launch time
Tue Dec 15 2020 21:12:56 GMT+0000 (Greenwich Mean Time) (3 minutes)
AMI location
amazon/amzn2-ami-hvm-2.0.20201126.0-x86_64-gp2
Lifecycle
normal
Stop-hibernate behavior
disabled
AMI Launch index
0
Key pair name
rust_app

Details | **Security** | Networking | Storage | Status Checks | Monitoring | Tags

▼ **Security details**

IAM Role

–

Security groups

sg-0929f2dff51d8c1e3 (launch-wizard-1)

Owner ID

875019845375

▼ **Inbound rules**

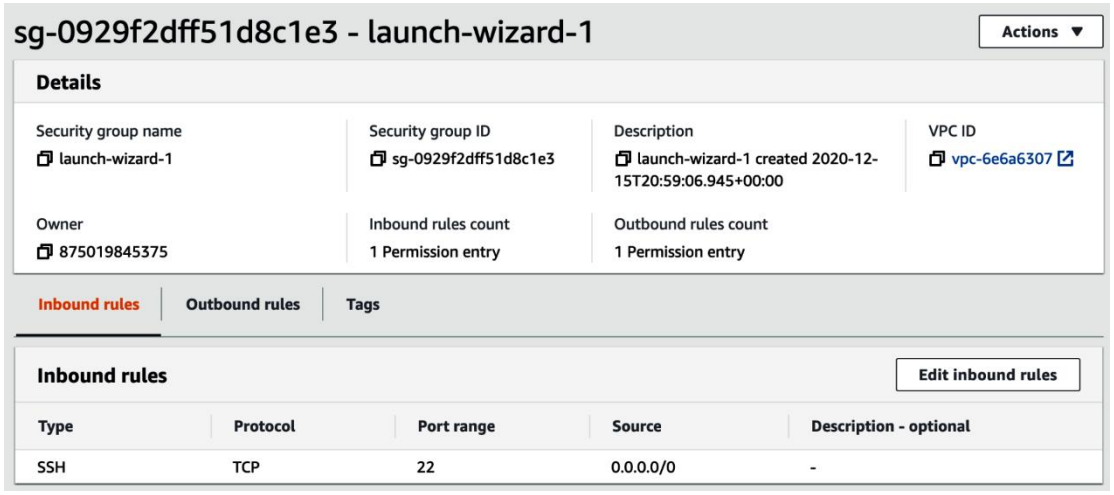
Filter rules

Port range	Protocol	Source	Security groups
All	All	0.0.0.0/0	launch-wizard-1

▼ **Outbound rules**

Filter rules

Port range	Protocol	Destination	Security groups
All	All	0.0.0.0/0	launch-wizard-1



Edit inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the instance.

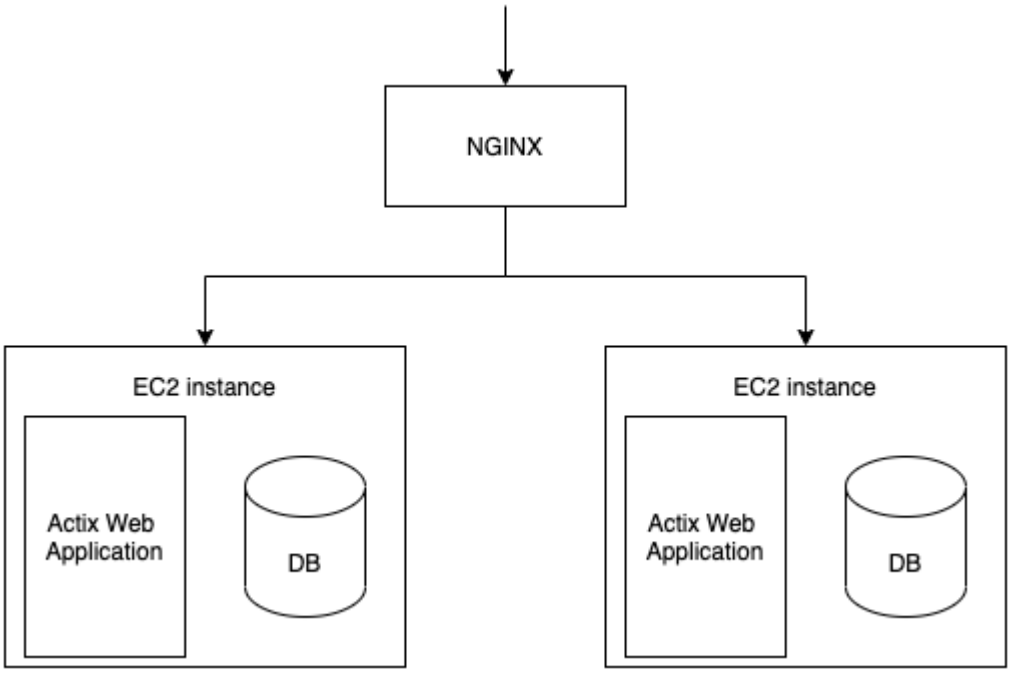
Inbound rules Info

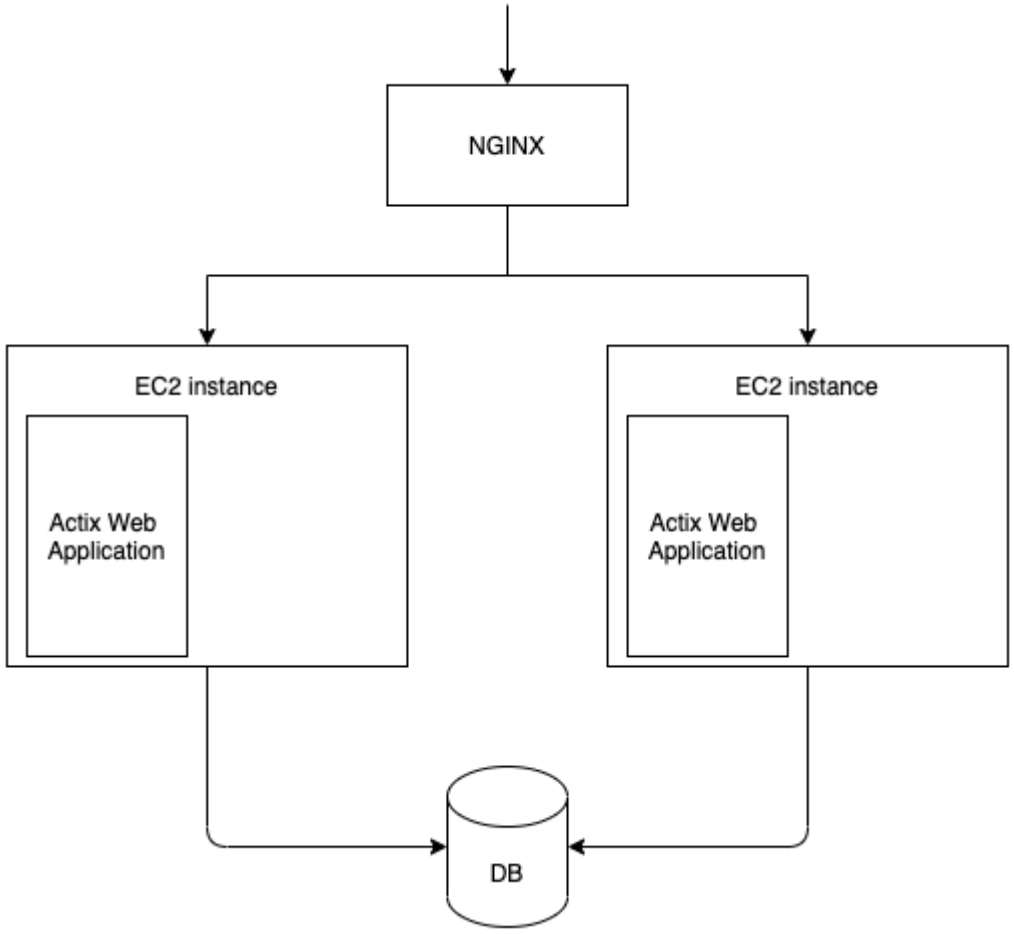
Type Info	Protocol Info	Port range Info	Source Info		Description - optional Info	
SSH	TCP	22	Custom	0.0.0.0/0		Delete
HTTP	TCP	80	Custom	0.0.0.0/0		Delete

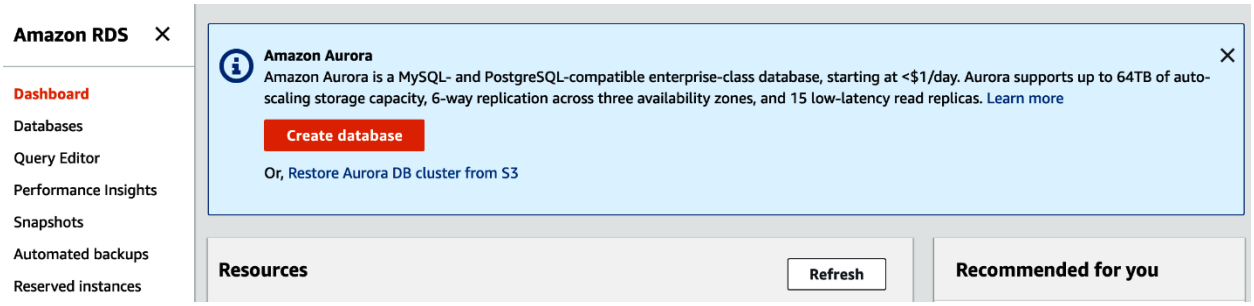
Add rule

NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

Cancel | Preview changes | Save rules

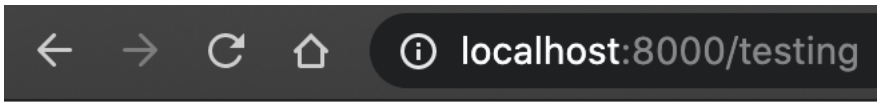



NGINX
EC2 instance
Actix Web Application
EC2 instance
Actix Web Application
DB


Amazon RDS
Dashboard
Databases
Query Editor
Performance Insights
Snapshots
Automated backups
Reserved instances
Amazon Aurora
Amazon Aurora is a MySQL- and PostgreSQL-compatible enterprise-class database, starting at <$1/day. Aurora supports up to 64TB of auto-scaling storage capacity, 6-way replication across three availability zones, and 15 low-latency read replicas. Learn more
Create database
Or, Restore Aurora DB cluster from S3
Resources
Refresh
Recommended for you

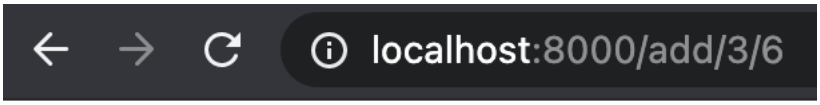
Chapter 11: Understanding Rocket Web Framework


```
Hello, 31 year old named maxwell!
```

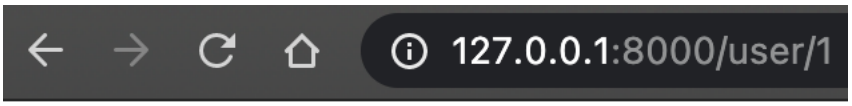



```
URL does not exist
```

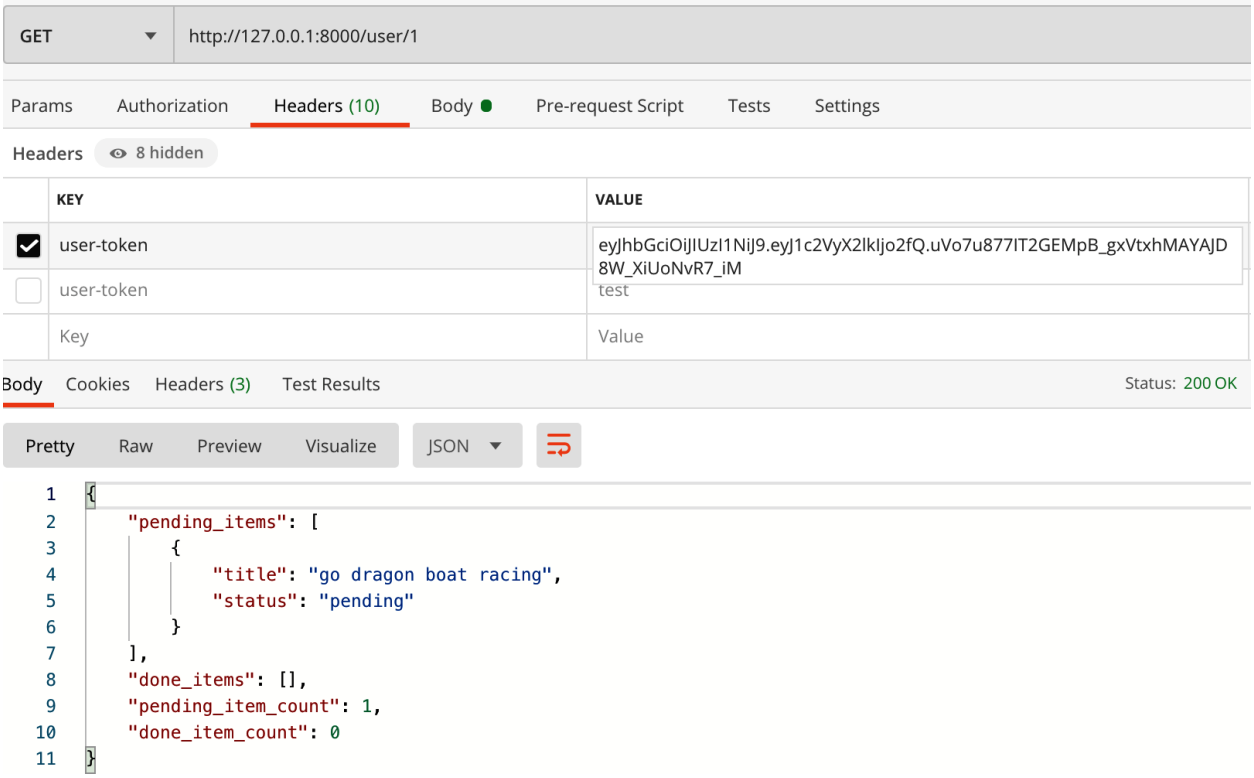
Appendix A: Understanding the Warp Framework



9



Missing request header "user-token"



POST http://127.0.0.1:8000/make

```
{
    "title": "eat dinner",
    "status": "pending"
}
```

Body Cookies Headers (3) Test Results Status: 200 OK

Pretty Raw Preview Visualize JSON


```
{
    "pending_items": [
        {
            "title": "this is a title",
            "status": "pending"
        },
        {
            "title": "cook dinner",
            "status": "pending"
        },
        {
            "title": "eat dinner",
            "status": "pending"
        }
    ],
    "done_items": [],
    "pending_item_count": 3,
    "done_item_count": 0
}
```