Deploy FastAPI on Deta

In this section you will learn how to easily deploy a **FastAPI** application on Deta using the free plan. 🎁

It will take you about **10 minutes**.

!!! info Deta is a **FastAPI** sponsor. 🥕

A basic FastAPI app

- Create a directory for your app, for example, `./fastapideta/` and enter into it.

FastAPI code

- Create a `main.py` file with:

```Python
from fastapi import FastAPI

app = FastAPI()


@app.get("/")
def read_root():
    return {"Hello": "World"}


@app.get("/items/{item_id}")
def read_item(item_id: int):
    return {"item_id": item_id}
```

Requirements

Now, in the same directory create a file `requirements.txt` with:

```
fastapi
```

!!! tip You don't need to install Uvicorn to deploy on Deta, although you would probably want to install it locally to test your app.

Directory structure

You will now have one directory `./fastapideta/` with two files:

```
.
└── main.py
└── requirements.txt
```

Create a free Deta account

Now create a free account on Deta, you just need an email and password.

You don't even need a credit card.

Install the CLI

Once you have your account, install the Deta CLI:

=== "Linux, macOS"

```
<div class="termy">

```console
$ curl -fsSL https://get.deta.dev/cli.sh | sh
```

</div>
```

=== "Windows PowerShell"

```
<div class="termy">

```console
$ iwr https://get.deta.dev/cli.ps1 -useb | iex
```

</div>
```

After installing it, open a new terminal so that the installed CLI is detected.

In a new terminal, confirm that it was correctly installed with:

```
$ deta --help

Deta command line interface for managing deta micros.
Complete documentation available at https://docs.deta.sh

Usage:
  deta [flags]
  deta [command]

Available Commands:
  auth        Change auth settings for a deta micro

...
```

!!! tip If you have problems installing the CLI, check the official Deta docs.

Login with the CLI

Now login to Deta from the CLI with:

```
$ deta login
```

```
Please, log in from the web page. Waiting..
Logged in successfully.
```

This will open a web browser and authenticate automatically.

Deploy with Deta

Next, deploy your application with the Deta CLI:

```
$ deta new

Successfully created a new micro

// Notice the "endpoint" 🔍

{
    "name": "fastapideta",
    "runtime": "python3.7",
    "endpoint": "https://qltnci.deta.dev",
    "visor": "enabled",
    "http_auth": "enabled"
}

Adding dependencies...


---> 100%


Successfully installed fastapi-0.61.1 pydantic-1.7.2 starlette-0.13.6
```

You will see a JSON message similar to:

```
{
        "name": "fastapideta",
        "runtime": "python3.7",
        "endpoint": "https://qltnci.deta.dev",
        "visor": "enabled",
        "http_auth": "enabled"
}
```

!!! tip Your deployment will have a different `"endpoint"` URL.

Check it

Now open your browser in your `endpoint` URL. In the example above it was `https://qltnci.deta.dev`, but yours will be different.

You will see the JSON response from your FastAPI app:

```
{
    "Hello": "World"
}
```

And now go to the `/docs` for your API, in the example above it would be `https://qltnci.deta.dev/docs`.

It will show your docs like:

Enable public access

By default, Deta will handle authentication using cookies for your account.

But once you are ready, you can make it public with:

```
$ deta auth disable

Successfully disabled http auth
```

Now you can share that URL with anyone and they will be able to access your API. 🚀

HTTPS

Congrats! You deployed your FastAPI app to Deta! 🎉 🍰

Also, notice that Deta correctly handles HTTPS for you, so you don't have to take care of that and can be sure that your clients will have a secure encrypted connection. ✅ 🔒

Check the Visor

From your docs UI (they will be in a URL like `https://qltnci.deta.dev/docs`) send a request to your *path operation* `/items/{item_id}`.

For example with ID `5`.

Now go to https://web.deta.sh.

You will see there's a section to the left called "Micros" with each of your apps.

You will see a tab with "Details", and also a tab "Visor", go to the tab "Visor".

In there you can inspect the recent requests sent to your app.

You can also edit them and re-play them.

Learn more

At some point, you will probably want to store some data for your app in a way that persists through time. For that you can use Deta Base, it also has a generous **free tier**.

You can also read more in the Deta Docs.

Deployment Concepts

Coming back to the concepts we discussed in Deployments Concepts{.internal-link target=_blank}, here's how each of them would be handled with Deta:

- **HTTPS**: Handled by Deta, they will give you a subdomain and handle HTTPS automatically.
- **Running on startup**: Handled by Deta, as part of their service.
- **Restarts**: Handled by Deta, as part of their service.
- **Replication**: Handled by Deta, as part of their service.
- **Memory**: Limit predefined by Deta, you could contact them to increase it.
- **Previous steps before starting**: Not directly supported, you could make it work with their Cron system or additional scripts.

!!! note Deta is designed to make it easy (and free) to deploy simple applications quickly.

```
It can simplify several use cases, but at the same time, it doesn't support others,
like using external databases (apart from Deta's own NoSQL database system), custom
virtual machines, etc.

You can read more details in the <a href="https://docs.deta.sh/docs/micros/about/"
class="external-link" target="_blank">Deta docs</a> to see if it's the right choice
for you.
```