Module 05

Purwadhika
Digital Technology School

AI Fullstack Software Development

Building and Deploying Serverless Applications

Job Connector Program

Outline

- Serverless Architecture
- Setting up the Environment
- Deployment on Vercel

Understanding Serverless Architecture

- Serverless doesn’t mean “no servers”, it means developers don’t have to manage them.
- The infrastructure, scaling, and maintenance are abstracted away by cloud providers.
- You focus purely on writing code, while the platform handles provisioning, scaling, and uptime.

Benefits of Going Serverless

- **Automatic Scaling:** Functions scale up or down based on demand.
- **Cost Efficiency**: Pay only for execution time, not idle servers.
- **Faster Deployment**: Deploy instantly with no DevOps overhead.
- **Developer Experience**: Focus on logic, not infrastructure.

Architecture Overview

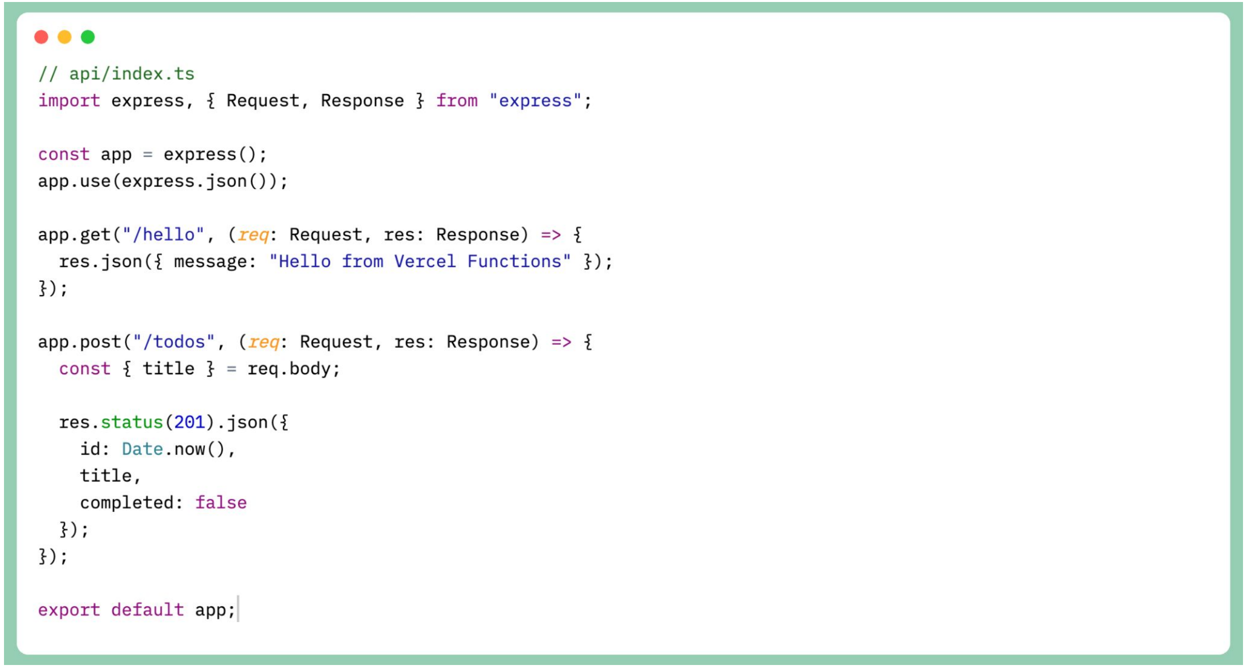
We’ll use:

- A serverless **Express API** written in **TypeScript**.
- **Supabase** as our cloud **PostgreSQL** database.
- **Vercel Functions** to host our API endpoints.

Vercel Functions

- Vercel Functions allow us to deploy backend logic without managing a server.
- Each API route runs as a serverless function, executed only when requested.
- When using Express, the Express app is wrapped and exported as a handler so Vercel can execute it per request.
- This approach is suitable for lightweight APIs, CRUD services, and microservices that need fast deployment and automatic scaling.

Serverless Function Example

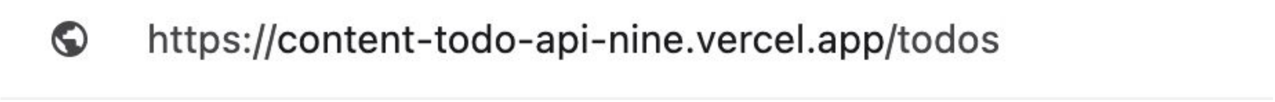


```
// api/index.ts
import express, { Request, Response } from "express";

const app = express();
app.use(express.json());

app.get("/hello", (req: Request, res: Response) => {
  res.json({ message: "Hello from Vercel Functions" });
});

app.post("/todos", (req: Request, res: Response) => {
  const { title } = req.body;

  res.status(201).json({
    id: Date.now(),
    title,
    completed: false
  });
});

export default app;
```

Vercel automatically treats this file as a serverless endpoint.

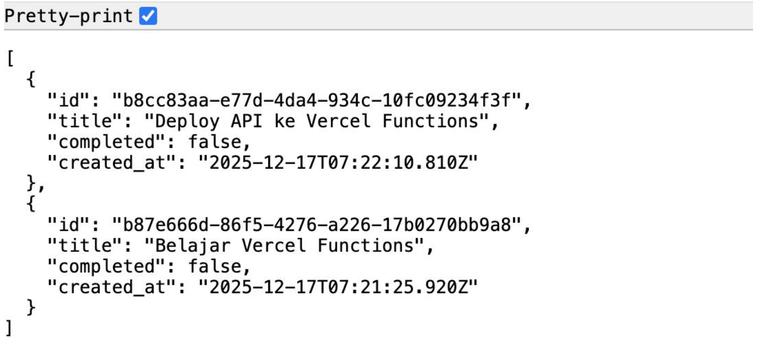
Testing the Deployed API

After deployment, Vercel gives you a live URL like:



https://content-todo-api-nine.vercel.app/todos

You can test endpoints with Postman or browser



Pretty-print ☑

```
[
  {
    "id": "b8cc83aa-e77d-4da4-934c-10fc09234f3f",
    "title": "Deploy API ke Vercel Functions",
    "completed": false,
    "created_at": "2025-12-17T07:22:10.810Z"
  },
  {
    "id": "b87e666d-86f5-4276-a226-17b0270bb9a8",
    "title": "Belajar Vercel Functions",
    "completed": false,
    "created_at": "2025-12-17T07:21:25.920Z"
  }
]
```

Key Deployment on Vercel Functions

- No need to call **app.listen()** — Vercel handles the server lifecycle
- Each request runs in an isolated environment (stateless execution)
- Cold starts may occur, but performance is optimized for short-lived tasks
- Ideal for REST APIs, webhooks, and simple backend services

Exercise

Enhance your existing **Blog App** by deploying it to a production environment using **Vercel** or **Render**, and integrating environment variables securely for **API** keys and database connections

Thank you