

About the reviewer

Akash Ranjan is a Python professional, with 6+ years of industry experience. He has developed, deployed, and managed large-scale enterprise applications. He has extensive experience in building APIs and designing microservice-based application architecture.

Table of Contents

Preface

Part 1: An Introduction to FastAPI

1

Getting Started with FastAPI

Technical Requirement	4	Package management	
Git basics	4	with pip	11
Installing Git	4	Installing pip	12
Git operations	5	Setting up Docker	13
Git branches	8	Dockerfile	14
Creating isolated development environments with Virtualenv	9	Building a simple FastAPI application	15
Creating a virtual environment	9	Summary	16
Activating and deactivating the virtual environment	9		

2

Routing in FastAPI

Technical requirements	18	Validating request bodies using Pydantic models	23
Understanding routing in FastAPI	18	Nested models	26
Routing example	18	Path and query parameters	27
Routing with the APIRouter class	19	Path parameters	27
		Query parameters	30

Request body	30	Building a simple CRUD app	37
FastAPI Automatic Docs	31	Summary	42

3

Response Models and Error Handling

Technical requirements	44	Status codes	44
Understanding responses in FastAPI	44	Building response models	45
What is a response header?	44	Error handling	48
What is a response body?	44	Summary	52

4

Templating in FastAPI

Technical requirements	53	Macros	57
Understanding Jinja	54	Template inheritance	58
Filters	54	Using Jinja templates in FastAPI	58
Using if statements	56	Summary	67
Loops	56		

Part 2: Building and Securing FastAPI Applications

5

Structuring FastAPI Applications

Technical requirements	72	Implementing the models	74
Structuring in FastAPI applications	72	Implementing routes	78
Building an event planner application	73	Summary	87

6

Connecting to a Database

Technical requirements	90	Document	104
Setting up SQLAlchemy	90	Initializing the database	106
Tables	90	CRUD operations	110
Rows	91	Create	110
Sessions	92	Read	110
Creating a database	92	Update	111
Creating events	97	Delete	112
Read events	99	routes/events.py	112
Update events	101	routes/users.py	114
Delete event	102	Summary	120
Setting up MongoDB	104		

7

Securing FastAPI Applications

Technical requirements	122	Handling user authentication	132
Authentication methods in FastAPI	122	Updating the application	133
Dependency injection	123	Updating the user sign-in route	133
Creating and using a dependency	123	Updating event routes	136
Securing the application with OAuth2 and JWT	124	Updating the event document class and routes	140
Hashing passwords	125	Configuring CORS	145
Creating and verifying access tokens	129	Summary	146