YOUR LOGO

Presenter: Manas ranjan

Introduction to Testing in React Development

Key Concepts and Best Practices

Contents

Welcome & Introduction	01
Overview of Tools: Jest & VS Code	02
Setting Up the Environment	03
Writing Unit Tests with Jest	04
Snapshot Testing in Jest	05
Debugging Tests in VS Code	06
Best Practices & Tips	07
Q&A and Conclusion	08

01

Welcome & Introduction

Session Overview

Importance of Testing

Testing is a crucial part of building reliable, maintainable applications. Helps catch bugs early and improves developer confidence.

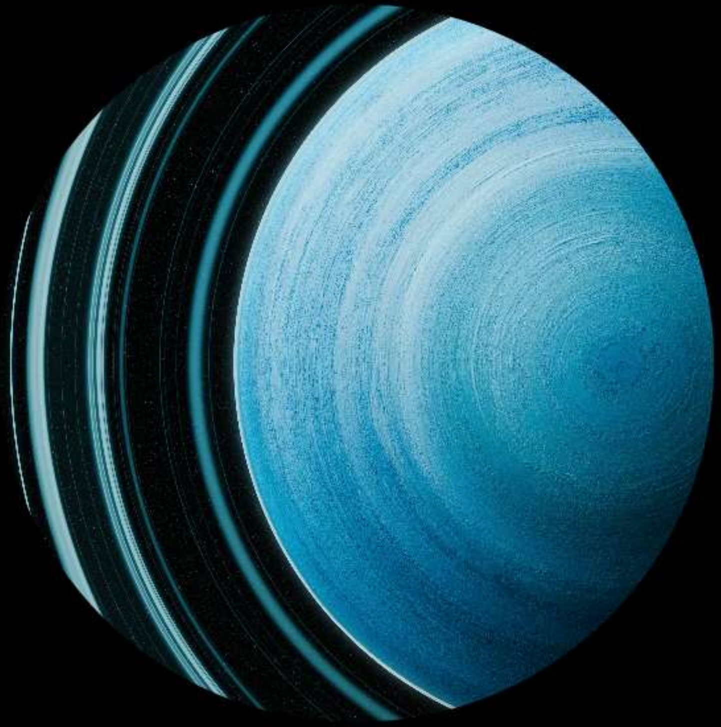
Key Topics

Importance of testing in React
Tools like Jest and VS Code
Writing unit and snapshot tests
Debugging using VS Code
Best practices and live tips

Importance of Testing in Modern React Development

Why Testing Matters

Ensures application functionality and performance.
Makes refactoring safer and faster.
Encourages modular, clean code.
Facilitates team collaboration and scalability.
Critical in CI/CD pipelines for continuous quality checks.



02

Overview of Tools: Jest & VS Code

Tools Overview

Jest Framework

Jest: JavaScript testing framework by Meta.
Supports unit, integration, and snapshot testing.
Built-in assertion library and test runner.

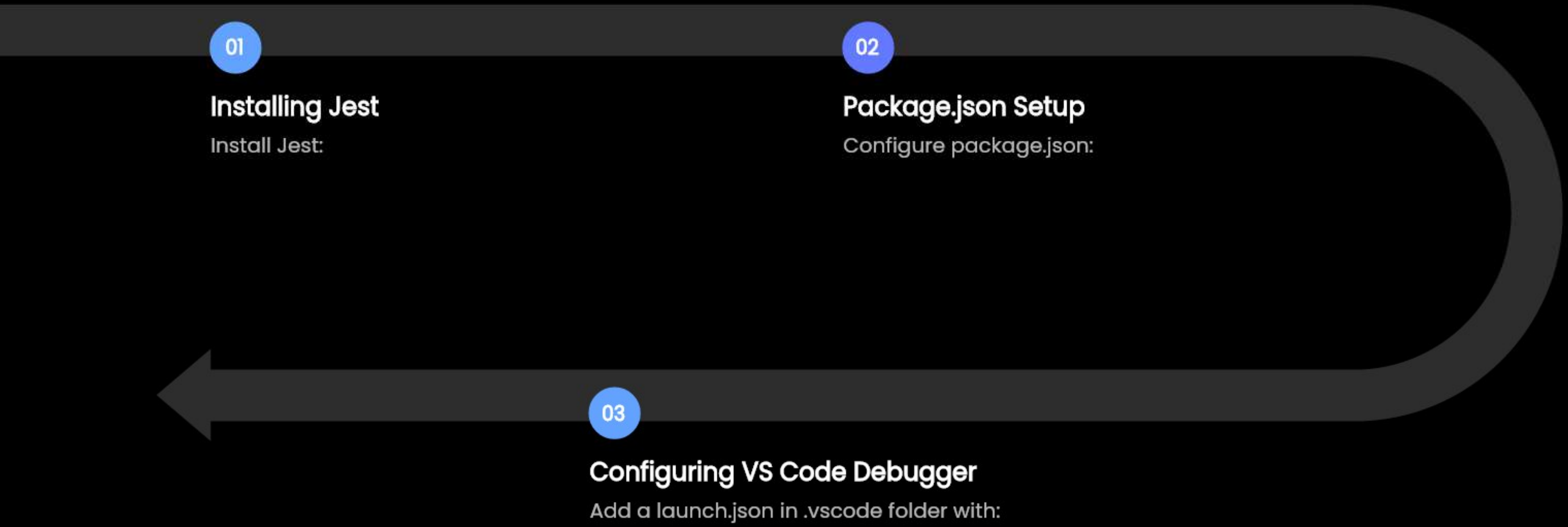
Visual Studio Code

Lightweight and powerful code editor.
Great support for debugging JavaScript and Jest tests.
Extensions like "Jest" and "Debugger for Chrome" boost productivity.

03

Setting Up the Environment

Environment Setup



Installing Jest

Install Jest:

Package.json Setup

Configure package.json:

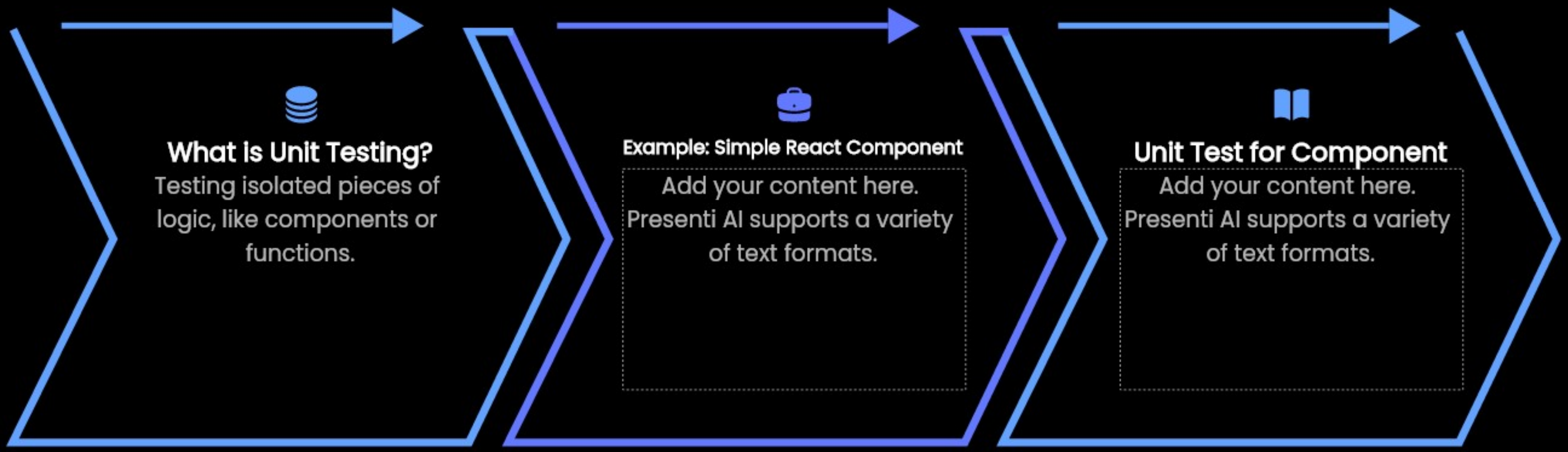
Configuring VS Code Debugger

Add a launch.json in .vscode folder with:

04

Writing Unit Tests with Jest

Unit Testing Basics



What is Unit Testing?

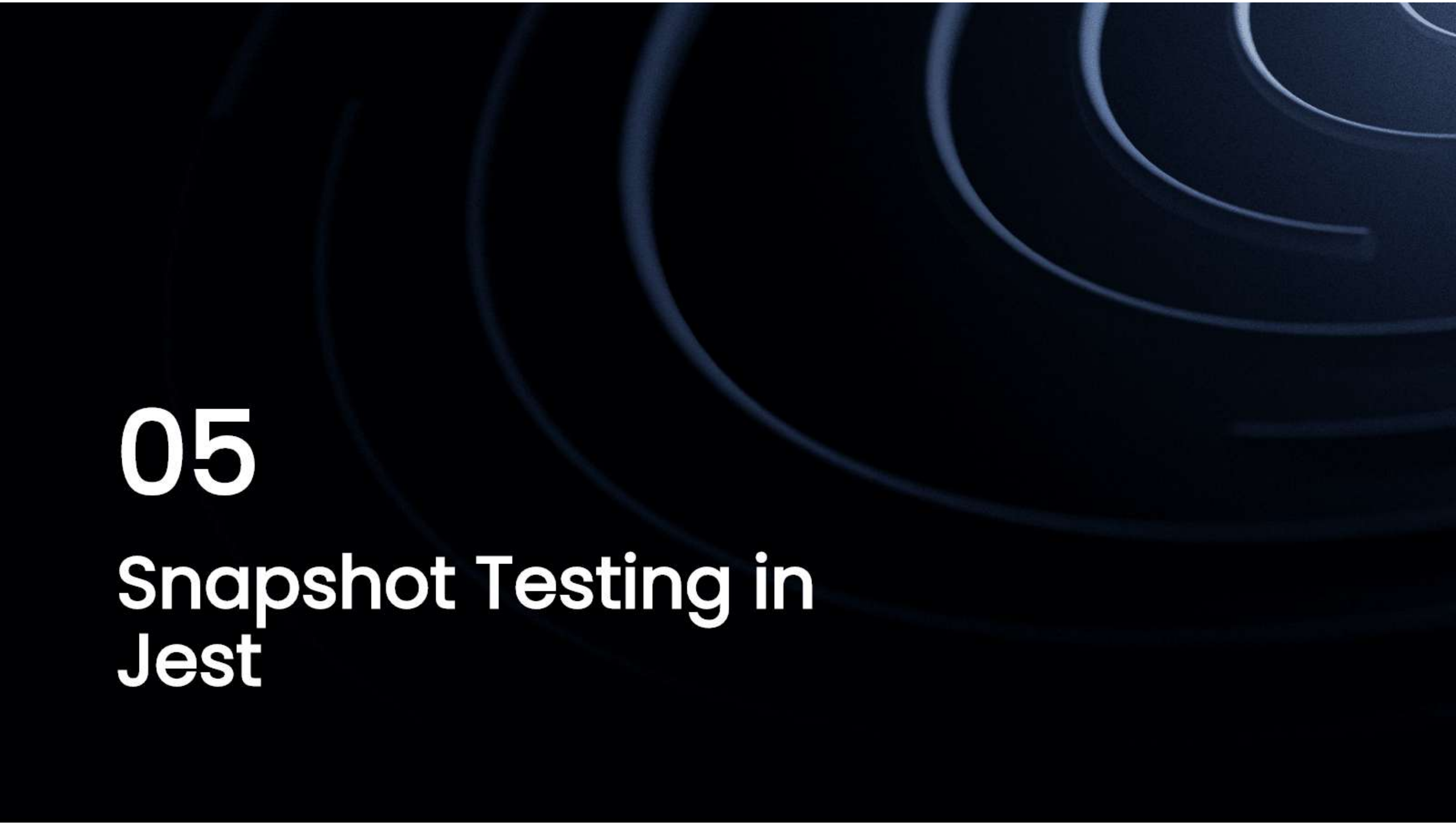
Testing isolated pieces of logic, like components or functions.

Example: Simple React Component

Add your content here. Presenti AI supports a variety of text formats.

Unit Test for Component

Add your content here. Presenti AI supports a variety of text formats.



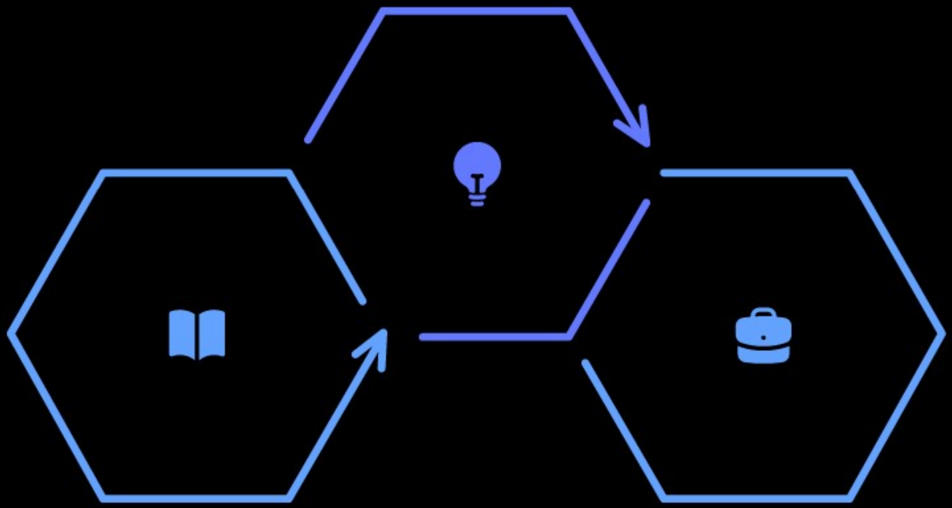
05

Snapshot Testing in Jest

Snapshot Testing

Snapshot Test Example

Add your content here.
Presenti AI supports a variety
of text formats.



What is Snapshot Testing?

Captures rendered output and compares on future runs.
Useful for detecting unexpected UI changes.

Tips for Snapshot Testing

Commit snapshots only when UI changes are intentional.

06

Debugging Tests in VS Code

Debugging with VS Code

Set Breakpoints

Open your test file.
Click on the left gutter to set breakpoints.

Run Jest in Debug Mode

Open the Run & Debug tab in VS Code.
Select the “Jest Tests” configuration.
Start debugging—VS Code will pause at breakpoints.

Inspecting Code Execution

Use the Debug Console.
Hover over variables for values.
Step through code execution line by line.

07

Best Practices & Tips

Testing Best Practices

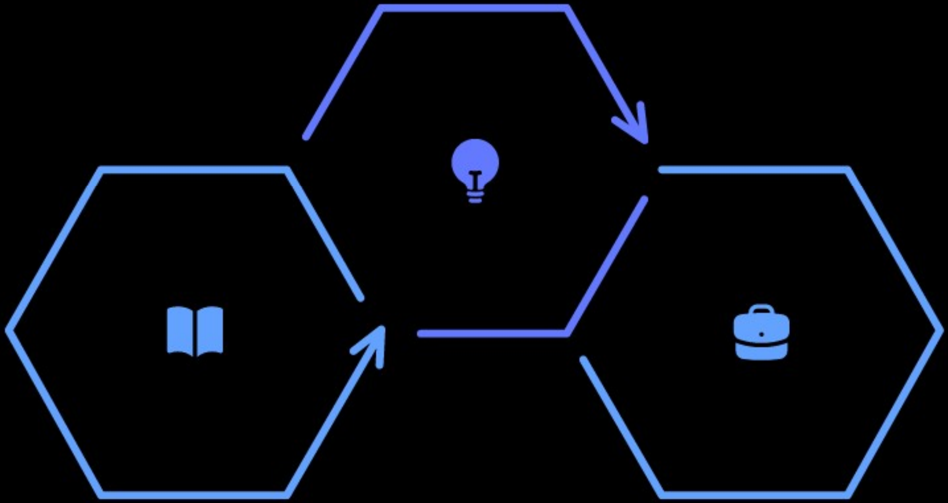
General Best Practices

Test behavior, not implementation.
Keep tests isolated and fast.
Use descriptive test names.
Mock external dependencies.

User Interaction Testing

Prefer screen.getByRole or getByText for testing user interactions.

Debugging Tips



Add console.log() for temporary insights.

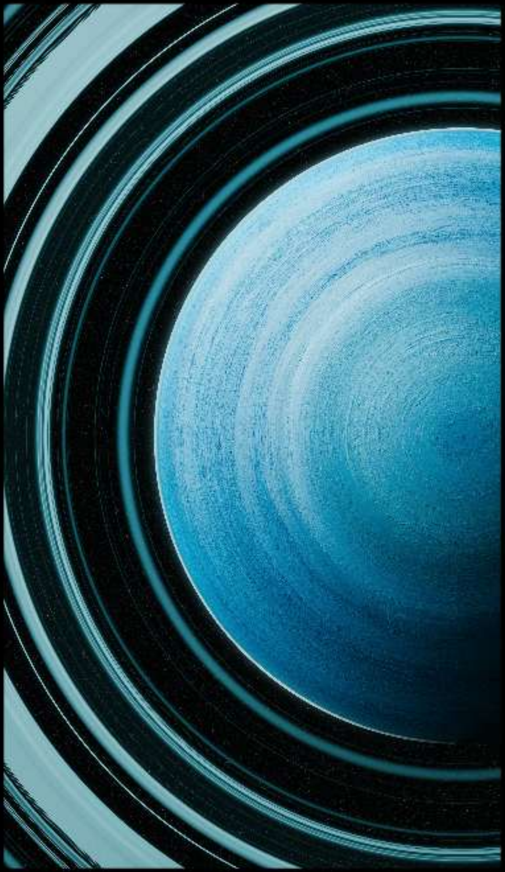
Use --watch mode for faster feedback.

Use .only or .skip to focus on specific tests.

08

Q&A and Conclusion

Wrap-up & Key Takeaways

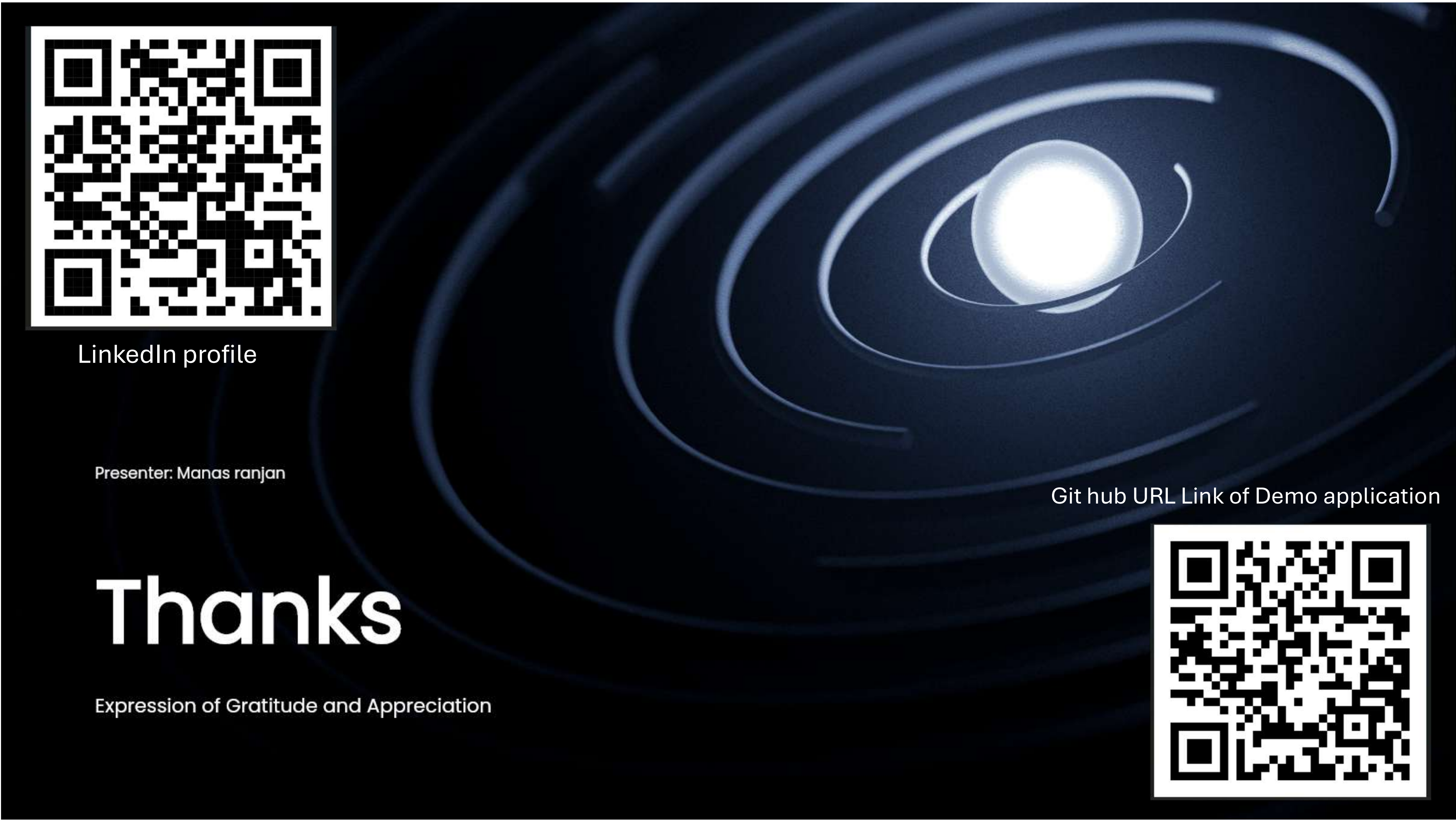


What We Covered

Importance of testing React apps
Tools: Jest and VS Code
Unit & snapshot tests
Debugging inside VS Code
Best practices to follow

Questions and Closing

Questions?
Happy to answer!
Thank You!Stay confident with testing, and keep your code quality high!



LinkedIn profile

Presenter: Manas ranjan

Git hub URL Link of Demo application

Thanks

Expression of Gratitude and Appreciation