Report on Pages and Routing in Next.js (App Router)

Date: June 30, 2025
Topic: Pages and Routing in Next.js (App Router)

Introduction

Today, I learned how routing in Next.js is built using the **file system** — there's no need to manually configure routes like in traditional React apps. The App Router makes routing **simple, intuitive, and scalable**.

File-Based Routing

Every file inside the app/ directory becomes a route.

Examples:
app/page.tsx → / (homepage)
app/about/page.tsx → /about
app/contact/page.tsx → /contact

It's simple and intuitive: just create folders and files — they automatically turn into pages!

Nested Routes

Nested folders = Nested routes.

Examples:
app/blog/page.tsx → /blog
app/blog/post/page.tsx → /blog/post

You can also have layout.tsx files inside nested folders to define shared UI across child pages.

Dynamic Routes

Dynamic routes are created using **square brackets** [param].

Example:
app/blog/[slug]/page.tsx → /blog/my-first-post

You can access the slug via params.slug.

Code Example:

```
export default function BlogPost({ params }) {
  return <h1>Blog Post: {params.slug}</h1>;
}
```

Catch-All Routes

Catch-all routes allow you to capture multiple path segments.

Example:
app/docs/[...slug]/page.tsx → /docs/a/b/c

Conclusion

Next.js App Router provides a **powerful and developer-friendly routing system** based entirely on the file system. It supports static, nested, dynamic, and catch-all routes with minimal configuration. This approach makes routing **clean, scalable, and easy to maintain**.