

# CMPS 356 – Fall 2022

## Web Applications Design and Development

### Lab 11

#### Next 13 and Mutations

##### Objective

1. Creating a root layout and nested layouts
2. Routing using directories and pages
3. Using loading interfaces, error handlers, and not-found pages
4. Serving JSON files through API endpoints

##### Prerequisites

1. Next 13: <https://beta.nextjs.org/docs>
2. Next 13 App Directory Playground: <https://app-dir.vercel.app>
3. Next API Endpoints: <https://nextjs.org/docs/api-routes/introduction>
4. NextUI: <https://nextui.org>
5. React Query: <https://tanstack.com/query/v4>

##### 1. New Features

1. Create a new directory 01-world-cup and a Next 13 application under it: “`npx create-next-app@latest --experimental-app`”.
  - 1.1. Install NextUI: “`npm i @nextui-org/react`” and provide it in the application.
  - 1.2. Install React Query: “`npm i @tanstack/react-query`” and provide it in the application.
2. Update the root layout to include a header, main content, and a footer.
  - 2.1. Define and use Header and Footer components.
  - 2.2. Create a Head component that receives a title property in `head.js`.
  - 2.3. Display the World Cup 2022 logo in the header with a link to the homepage.
  - 2.4. Add basic copyright information to the footer.
3. Add two endpoints: “`/api/teams`” and “`/api/matches`” for serving the team and match data, respectively, using the provided JSON files.
4. Display the World Cup 2022 logo in the center of the homepage.
5. Create a page for the groups and add a link to the header.
  - 5.1. Display the list of groups and corresponding teams in a responsive grid using cards.
  - 5.2. Display the name and flag for each team.
6. Create a page for the matches and add link to the header.
  - 6.1. Display the list of matches and corresponding teams in a responsive grid using cards.

- 6.2. Display the home/away teams, date/time, venue, and location for each match.
- 7. Create a page for the teams and add a link in the header.
  - 7.1. Display the list of teams in a responsive grid using cards.
  - 7.2. Create a sub-route for every team and list its matches.
  - 7.3. Add a not found handler for this sub-route.
- 8. Add loading interfaces and error handlers to all the routes.
- 9. Build the app using SSG, SSR, and ISR.
  - 9.1. SSG: Export a `generateStaticParams()` function for the paths.
  - 9.2. ISR: Export a `revalidate` constant.