

FanDuel NFL Depth Chart – Mani

A small, production-style implementation of an NFL Depth Chart with a clean domain core, a minimal ASP.NET Core API, and both unit & integration tests.

- Add/remove players at a position and depth index
- Get backups for a player at a position
- Print/get the full depth chart
- Designed for easy extension to other sports/teams

Tech Stack

- .NET 8
- ASP.NET Core Web API (minimal hosting)
- xUnit + FluentAssertions
- Microsoft.AspNetCore.Mvc.Testing for integration tests
- (Optional) reportgenerator for coverage HTML

Repo Layout

src

DepthChart

DepthChart.csproj

DepthChartService.cs (Player, PlayerNumber, Service)

DepthChart Api

DepthChart.Api.csproj

Program.cs

Tests

DepthChart.Tests

DepthChartServiceTests.cs

DepthChart.IntegrationTests

DepthChartApiTests.cs

.github/workflows/dotnet.yml

Prerequisites

- .NET SDK 8 (check with `dotnet --version`)
- Windows/Mac/Linux terminal
- (Optional) curl or Postman for API calls

Quick Start

From the repo root:

Restore + build

```
PS C:\Users\Vikki\Desktop\Mani task\mani-fanduel-task> dotnet build .\src\DepthChart\
```

```
PS C:\Users\Vikki\Desktop\Mani task\mani-fanduel-task> dotnet restore .\src\DepthChart\
```

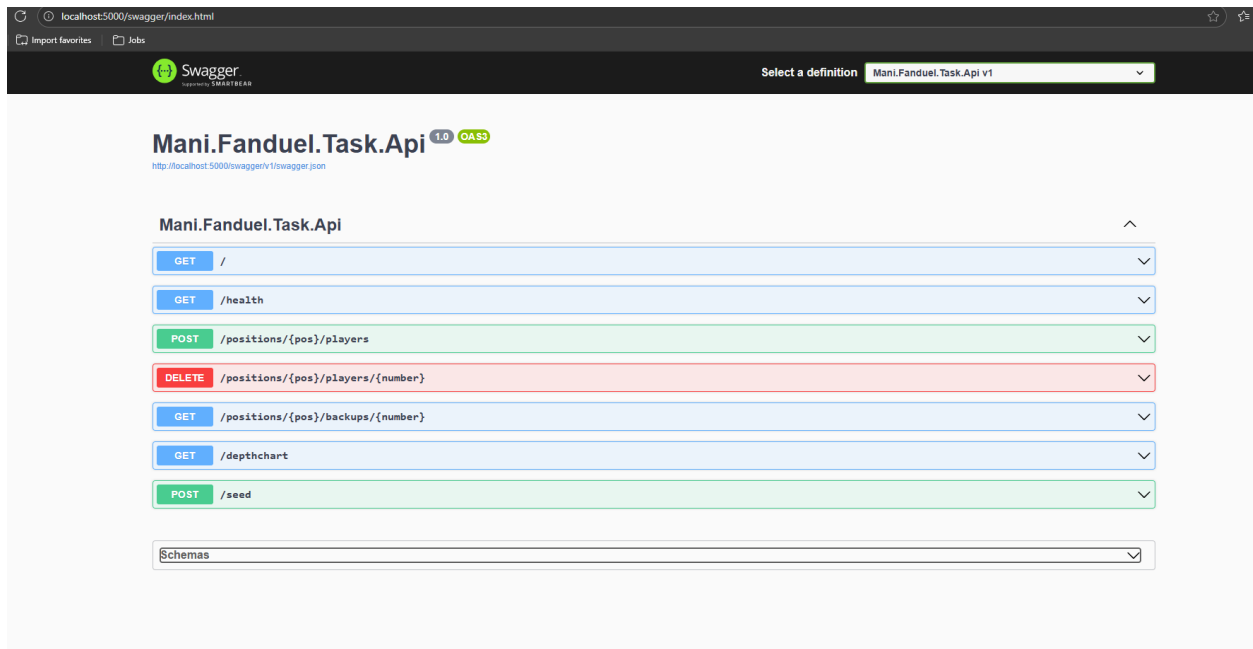
Run API (HTTP :5000, HTTPS :5001)

`dotnet run --project .\src\DepthChart.Api`

```
PS C:\Users\Vikki\Desktop\Mani task\mani-fanduel-task> dotnet run --project .\src\DepthChart.Api\
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: http://localhost:5000
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Production
info: Microsoft.Hosting.Lifetime[0]
      Content root path: C:\Users\Vikki\Desktop\Mani task\mani-fanduel-task\src\DepthChart.Api
```

```
PS C:\Users\Vikki\Desktop\Mani task\mani-fanduel-task\src> dotnet run --project "DepthChart.Api" --urls "http://localhost:5000;https://localhost:5001"
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: http://localhost:5000
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: https://localhost:5001
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Production
info: Microsoft.Hosting.Lifetime[0]
      Content root path: C:\Users\Vikki\Desktop\Mani task\mani-fanduel-task\src\DepthChart.Api
```

Open Swagger UI: <http://localhost:5001/swagger>



API Usage

Seed

POST /seed

Seeds a small Tampa Bay QB/LWR chart for demos/tests.

Add/Upsert Player

POST /positions/{position}/players?position_depth={index}

Body (JSON): { "number": 12, "name": "Tom Brady" }

- Inserts at index (shifts existing), or appends if index is > count.
- Re-adding an existing player moves them to the new index.

Remove Player

DELETE /positions/{position}/players/{number}

Removes the player if present. Returns removed player(s).

Get Backups

GET /positions/{position}/backups/{number}

Returns players listed behind the given player at that position.

Full Depth Chart

GET /depthchart

Returns entire chart as a map { position: [players...] }.

Example cURL (after seeding)

Backups for Brady (12) at QB

curl http://localhost:5000/positions/QB/backups/12

Full chart

curl http://localhost:5000/depthchart

Testing

```
PS C:\Users\Vikki\Desktop\Mani task\mani-fanduel-task> dotnet test .\tests\DepthChart.Tests\DepthChart.Tests.csproj -v n
```

```
Test run for C:\Users\Vikki\Desktop\Mani task\mani-fanduel-task\tests\DepthChart.Tests\bin\Debug\net8.0\DepthChart.Tests.dll (.NETCoreApp,Version=v8.0)
VSTest version 17.11.1 (x64)

Starting test execution, please wait...
A total of 1 test files matched the specified pattern.
Data collection : Unable to find a datacollector with friendly name 'XPlat Code Coverage'.
Data collection : Could not find data collector 'XPlat Code Coverage'
[xUnit.net 00:00:00.00] xUnit.net VSTest Adapter v2.8.1+ce9211e970 (64-bit .NET 8.0.19)
[xUnit.net 00:00:00.06] Discovering: DepthChart.Tests
[xUnit.net 00:00:00.09] Discovered: DepthChart.Tests
[xUnit.net 00:00:00.09] Starting: DepthChart.Tests
[xUnit.net 00:00:00.15] Finished: DepthChart.Tests
    Passed Mani.Fanduel.Task.Tests.DepthChartServiceTests.Insert_And_Shift_Behavior [20 ms]

Test Run Successful.
Total tests: 1
    Passed: 1
Total time: 0.8025 Seconds
1>Done Building Project "C:\Users\Vikki\Desktop\Mani task\mani-fanduel-task\tests\DepthChart.Tests\DepthChart.Tests.csproj" (VSTest target(s)).

Build succeeded.
    0 Warning(s)
    0 Error(s)

Time Elapsed 00:00:02.00
```

```
PS C:\Users\Vikki\Desktop\Mani task\mani-fanduel-task> dotnet test .\tests\DepthChart.IntegrationTests\DepthChart.IntegrationTests.csproj -v n
```

```
[xUnit.net 00:00:03.06] Finished: DepthChart.IntegrationTests
    Passed Mani.Fanduel.Task.IntegrationTests.DepthChartApiTests.Seed_Then_GetBackups_Works [1 s]

Test Run Successful.
Total tests: 1
    Passed: 1
Total time: 5.3340 Seconds
1>Done Building Project "C:\Users\Vikki\Desktop\Mani task\mani-fanduel-task\tests\DepthChart.IntegrationTests\DepthChart.IntegrationTests.csproj" (VSTest target(s)).

Build succeeded.
    0 Warning(s)
    0 Error(s)
```

Design Notes

- Domain-first: DepthChartService encapsulates the rules:
 - Add at index → shift to the right
 - Add with large index → append
 - Re-adding an existing player moves them instead of duplicating
 - Backups = players after the given player at that position
- Immutability-lite: PlayerNumber wraps the jersey number to avoid passing raw ints everywhere and to keep equality consistent.
- Separation of concerns: API is a slim layer that converts HTTP to domain calls; domain has no web dependencies.
- Testable: Service unit tests cover core behaviors; integration tests validate API endpoints & JSON contracts.

Assumptions

- Positions are case-insensitive strings (e.g., QB, LWR) with no strict enumeration to keep it flexible for new sports.
- A player is uniquely identified by number within a position for this exercise scope.
- Concurrency & persistence are intentionally out-of-scope (in-memory store) but addressed under *Extensibility*.

Troubleshooting

- 404 at GET /
That's expected; use <https://localhost:5001/swagger> or the endpoints above.
- Windows dev cert prompt
If HTTPS fails locally:
- `dotnet dev-certs https --trust`