Diagramele UML

Tema: Aplicatie web de planificare Grupa 12

a evenimentelor sportive

[1. Diagrama de activitate 3](#_Toc187695413)

[2. Diagrama de stare 4](#_Toc187695414)

[3. Diagrama de clasa 5](#_Toc187695415)

[4. Diagrama de deployment 6](#_Toc187695416)

[5. Diagrama caurilor de utilizre 7](#_Toc187695417)

# Diagrama de activitate

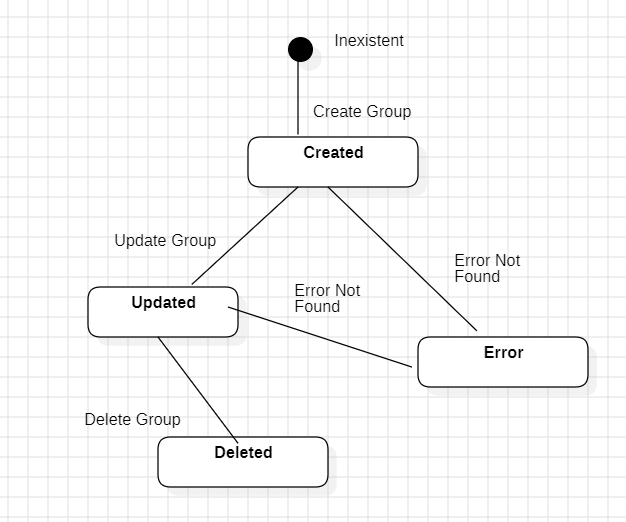
A diagram of a group

Description automatically generated

Această diagramă ilustrează fluxul principal de execuție al aplicației. Am utilizat-o pentru a descrie pașii din cadrul procesului de creare a unui Group:

* Start: Utilizatorul inițiază procesul printr-o cerere HTTP POST.
* ConvertDTO: Datele primite sunt transformate dintr-un obiect GroupDTO într-un obiect Group.
* AddToService: Obiectul Group este trimis către GroupService pentru procesare.
* CheckSuccess: Se verifică dacă operațiunea a fost reușită.
* ReturnOk sau ReturnNotFound: Sistemul răspunde cu succes sau eroare, în funcție de rezultat.

# Diagrama de stare



Aceasta descrie ciclul de viață al unui obiect Group, incluzând tranzițiile dintre stări:

* Inexistent: Starea inițială, în care grupul nu există.
* Created: Grupul a fost creat.
* Updated: Grupul a fost actualizat ulterior.
* Deleted: Grupul a fost șters.
* Error: Pot apărea erori, cum ar fi încercarea de a accesa un grup inexistent.

# Diagrama de clasa

A diagram of a group

Description automatically generated

Aceasta prezintă structura logică a aplicației și relațiile dintre clasele legate de Group:

* Group: Modelul central, care conține proprietăți precum Id și GroupCreator.
* GroupDTO: Obiect utilizat pentru transferul datelor.
* GroupService: Gestionează logica de business.
* GroupRepo: Accesează baza de date.
* GroupController: Răspunde la cererile utilizatorilor.

# Diagrama de deployment

A diagram of a service

Description automatically generated

Această diagramă arată modul în care componentele aplicației sunt distribuite în mediul de execuție:

* Client: Trimite cereri către server (browser sau client HTTP).
* WebServer: Rulează aplicația ASP.NET Core și găzduiește GroupController.
* ServiceLayer: Include GroupService pentru logica de business.
* Repository: Accesează baza de date folosind GroupRepo.
* Database: Stochează datele despre grupuri.

# Diagrama caurilor de utilizre

A diagram of a group

Description automatically generated

Prezintă scenariile principale de utilizare ale funcționalității Group:

* Create Group: Adminul poate crea un nou grup.
* View Groups: Utilizatorii pot vedea lista grupurilor.
* Update Group: Adminul poate actualiza datele unui grup.
* Delete Group: Adminul poate șterge un grup.