**Web API container :**

**Docent only calls = rood**

# 1. Course Controller

**Methods: CRUD operations for Courses.**

**Components:**

* Course Service: Business logic related to courses.
* Course Repository: Interaction with the database for course data.

**End Points:**

* GET /courses: Retrieve all courses.
* POST /courses: Create a new course.
* GET /courses/{id}: Retrieve a specific course.
* PUT /courses/{id}: Update a specific course.
* DELETE /courses/{id}: Delete a specific course.

# 2. Assignment Controller

**Methods: CRUD operations for Assignments.**

**Components:**

* Assignment Service: Business logic related to assignments.
* Assignment Repository: Interaction with the database for assignment data.

**End Points:**

* GET /assignments: Retrieve all assignments.
* POST /assignments: Create a new assignment.
* GET /assignments/{id}: Retrieve a specific assignment.
* PUT /assignments/{id}: Update a specific assignment.
* DELETE /assignments/{id}: Delete a specific assignment.

# 3. Group Controller

**Methods: CRUD operations for Groups and managing chats and file uploads in groups.**

**Components:**

* Group Service: Business logic related to groups.
* Group Repository: Interaction with the database for group data.

**End Points:**

* GET /groups: Retrieve all groups.
* POST /groups: Create a new group.
* POST /groups/{id}/chat: Send a message in the group chat.

# 4. Message Controller

**Methods: Operations for sending and receiving private messages.**

**Components:**

* Messaging Service: Business logic related to private messaging.
* Messaging Repository: Interaction with the database for messaging data.

**End Points:**

* POST /messages: Send a new private message.

# 5. Portfolio Controller

**Methods: Operations for managing portfolios.**

**Components:**

* Portfolio Service: Business logic related to portfolios.
* Portfolio Repository: Interaction with the database for portfolio data.

**End Points:**

* GET /portfolios: Retrieve all portfolios.
* POST /portfolios: Upload a new portfolio.
* GET / portfolios /{me}: Get my own portfolio.
* GET /portfolios/{id}: Retrieve a specific portfolio.

# 6. User Controller

**Methods: Operations for user management.**

**Components:**

* User Service: Business logic related to user management.
* User Repository: Interaction with the database for user data.

**End Points:**

* GET /users: Retrieve all users.
* POST /users: Create a new user.
* GET /users/{id}: Retrieve a specific user.
* PUT /users/{me}: Update my own user.
* DELETE /users/{id}: Delete a specific user.