**ỦY BAN NHÂN DÂN THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC SÀI GÒN**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI THỰC HÀNH 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG**

**MÔN: KIỂM THỬ PHẦN MỀM**

**Sinh viên thực hiện:** 1. Lê Thị Lan Anh - 3122411006

2. Lê Thị Duyên – 3122411028

3. Hồ Kim Sen - 3122411176

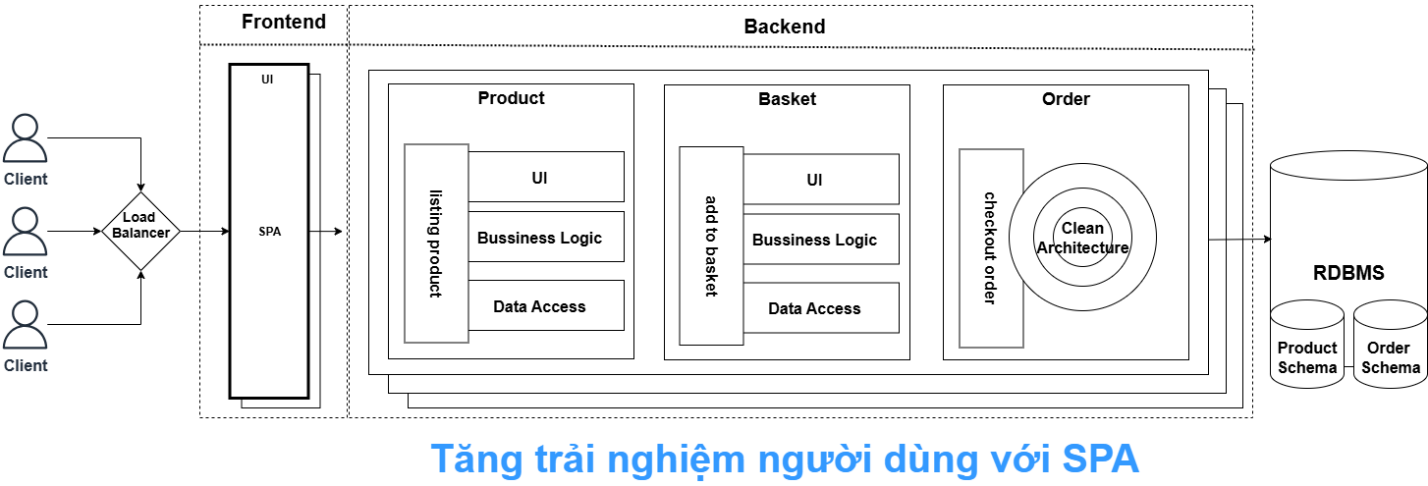
**Giảng viên phụ trách:** Đỗ Như Tài

**Thành phố Hồ Chí Minh, năm 2025**

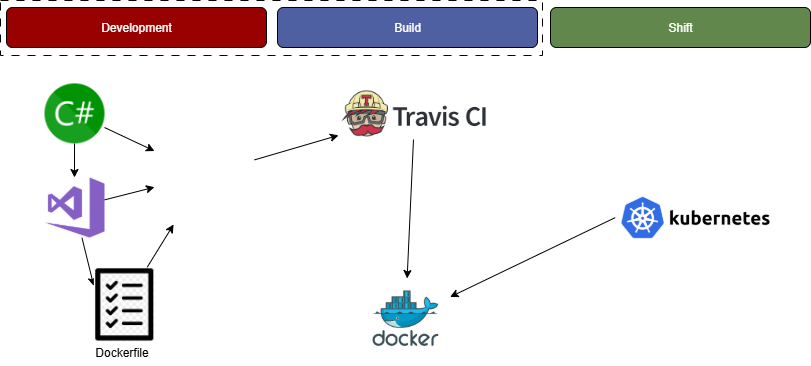
# **PHÂN CÔNG CÔNG VIỆC**

|  |  |  |
| --- | --- | --- |
| Thành viên | Nhiệm vụ | Mức độ hoàn thành |
| Lê Thị Lan Anh  lelananh02@gmail.com | Bài 4,5,8, BT ứng dụng | 100% |
| Lê Thị Mỹ Duyên  duyen210404@gmail.com | Bài 1,2,3,6,7 | 100% |
| Hồ Kim Sen  kimsen030504@gmail.com |  | 100% |

# **Bài 1:**



# **Bài 2:**



# **Bài 3:**

A screenshot of a computer

AI-generated content may be incorrect.

**Dô trang này: https://editor.swagger.io/**

openapi: 3.0.0

info:

  title: Coolstore services

  description: |

    Coolstore microservices project

    - Website

    - Send email to coolstore-microservices project

  version: "1.0.0"

servers:

  - url: HTTP

tags:

  - name: CartService

  - name: CatalogService

  - name: InventoryService

  - name: RatingService

paths:

  /cart/api/carts:

    post:

      tags: [CartService]

      summary: Create new cart

      responses:

        '200':

          description: Cart created

          content:

            application/json:

              schema:

                $ref: '#/components/schemas/Cart'

    put:

      tags: [CartService]

      summary: Update cart

      responses:

        '200':

          description: Cart updated

          content:

            application/json:

              schema:

                $ref: '#/components/schemas/Cart'

  /cart/api/carts/{cart\_id}:

    get:

      tags: [CartService]

      summary: Get cart by ID

      parameters:

        - name: cart\_id

          in: path

          required: true

          schema:

            type: string

      responses:

        '200':

          description: Cart details

          content:

            application/json:

              schema:

                $ref: '#/components/schemas/Cart'

  /cart/api/carts/{cart\_id}/checkout:

    put:

      tags: [CartService]

      summary: Checkout cart

      parameters:

        - name: cart\_id

          in: path

          required: true

          schema:

            type: string

      responses:

        '200':

          description: Checkout success

  /cart/api/carts/{cart\_id}/items/{product\_id}:

    delete:

      tags: [CartService]

      summary: Remove product from cart

      parameters:

        - name: cart\_id

          in: path

          required: true

          schema:

            type: string

        - name: product\_id

          in: path

          required: true

          schema:

            type: string

      responses:

        '204':

          description: Item deleted

  /catalog/api/products:

    post:

      tags: [CatalogService]

      summary: Create product

      responses:

        '200':

          description: Product created

          content:

            application/json:

              schema:

                $ref: '#/components/schemas/Product'

  /catalog/api/products/{current\_page}/{high\_price}:

    get:

      tags: [CatalogService]

      summary: Get products by price and page

      parameters:

        - name: current\_page

          in: path

          required: true

          schema:

            type: integer

        - name: high\_price

          in: path

          required: true

          schema:

            type: number

      responses:

        '200':

          description: List of products

          content:

            application/json:

              schema:

                type: array

                items:

                  $ref: '#/components/schemas/Product'

  /catalog/api/products/{product\_id}:

    get:

      tags: [CatalogService]

      summary: Get product by ID

      parameters:

        - name: product\_id

          in: path

          required: true

          schema:

            type: string

      responses:

        '200':

          description: Product details

          content:

            application/json:

              schema:

                $ref: '#/components/schemas/Product'

  /inventory/api/availabilities:

    get:

      tags: [InventoryService]

      summary: Get all availabilities

      responses:

        '200':

          description: Availability list

          content:

            application/json:

              schema:

                type: array

                items:

                  $ref: '#/components/schemas/Inventory'

  /inventory/api/availabilities/{id}:

    get:

      tags: [InventoryService]

      summary: Get availability by ID

      parameters:

        - name: id

          in: path

          required: true

          schema:

            type: string

      responses:

        '200':

          description: Availability details

          content:

            application/json:

              schema:

                $ref: '#/components/schemas/Inventory'

  /inventory/api/inventory/migrate:

    post:

      tags: [InventoryService]

      summary: Migrate inventory

      responses:

        '200':

          description: Migration success

  /rating/api/ratings:

    get:

      tags: [RatingService]

      summary: Get all ratings

      responses:

        '200':

          description: Ratings list

          content:

            application/json:

              schema:

                type: array

                items:

                  $ref: '#/components/schemas/Rating'

    post:

      tags: [RatingService]

      summary: Create rating

      responses:

        '200':

          description: Rating created

          content:

            application/json:

              schema:

                $ref: '#/components/schemas/Rating'

    put:

      tags: [RatingService]

      summary: Update rating

      responses:

        '200':

          description: Rating updated

          content:

            application/json:

              schema:

                $ref: '#/components/schemas/Rating'

  /rating/api/ratings/{product\_id}:

    get:

      tags: [RatingService]

      summary: Get rating by product ID

      parameters:

        - name: product\_id

          in: path

          required: true

          schema:

            type: string

      responses:

        '200':

          description: Rating details

          content:

            application/json:

              schema:

                $ref: '#/components/schemas/Rating'

components:

  schemas:

    Cart:

      type: object

      properties:

        cartId:

          type: string

        items:

          type: array

          items:

            $ref: '#/components/schemas/CartItem'

    CartItem:

      type: object

      properties:

        productId:

          type: string

        quantity:

          type: integer

    Product:

      type: object

      properties:

        productId:

          type: string

        name:

          type: string

        price:

          type: number

          format: float

    Inventory:

      type: object

      properties:

        productId:

          type: string

        stock:

          type: integer

    Rating:

      type: object

      properties:

        productId:

          type: string

        stars:

          type: integer

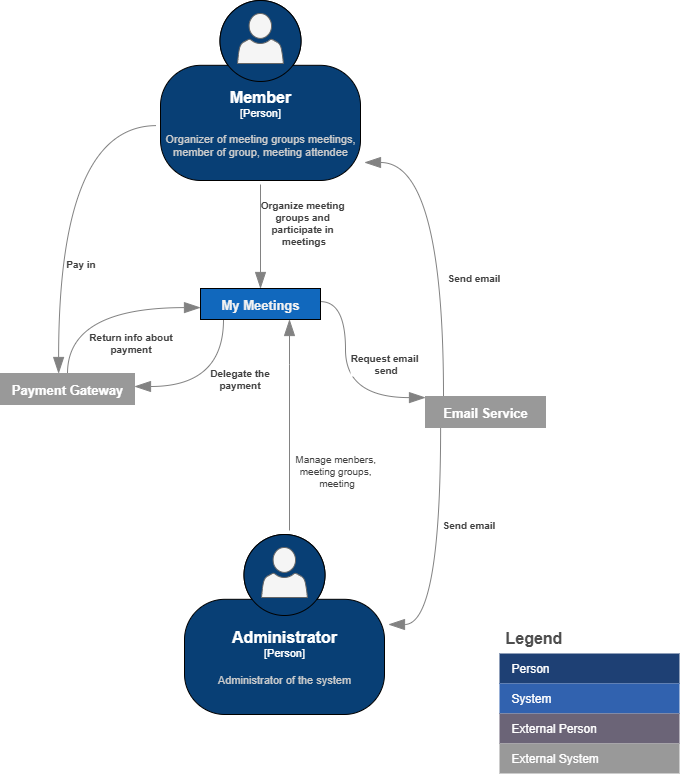
          minimum: 1

          maximum: 5

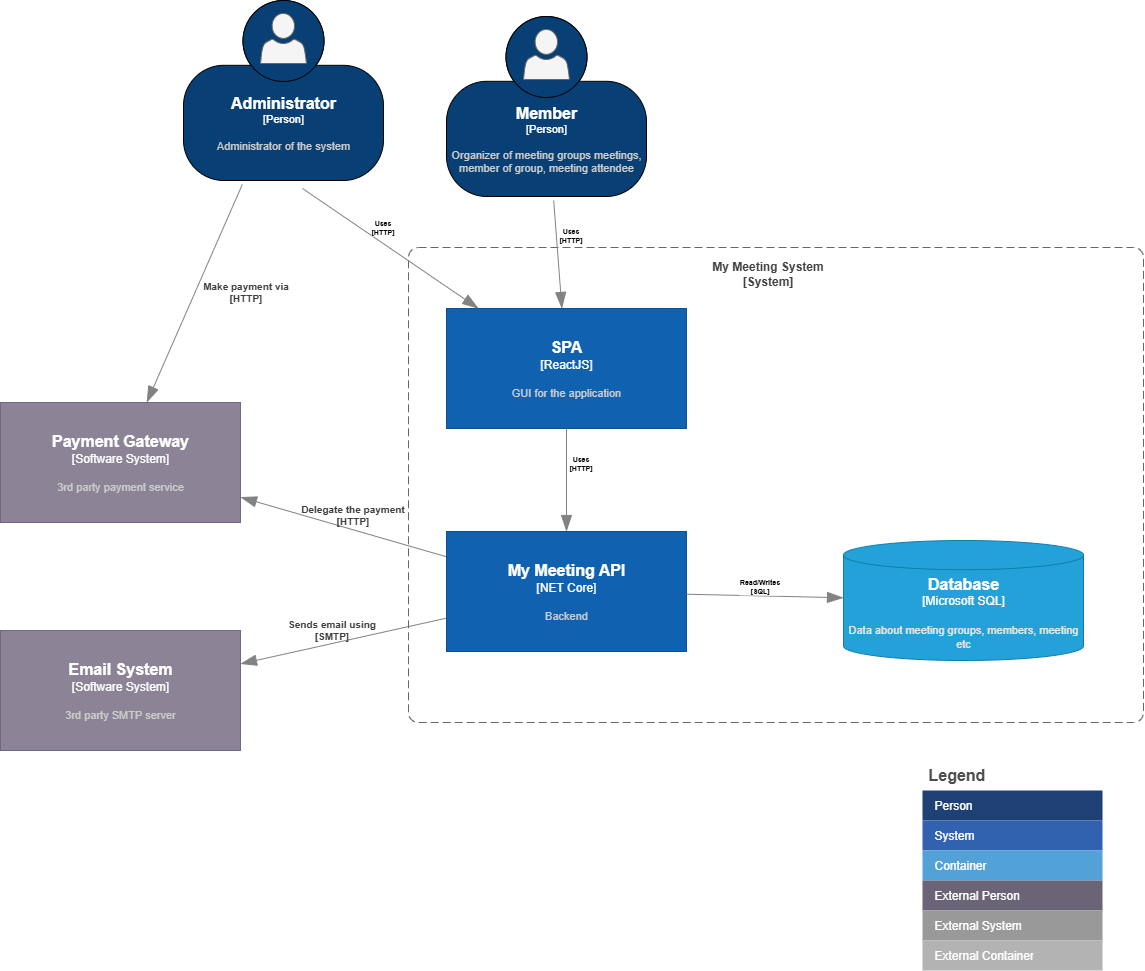
        comment:

          type: string

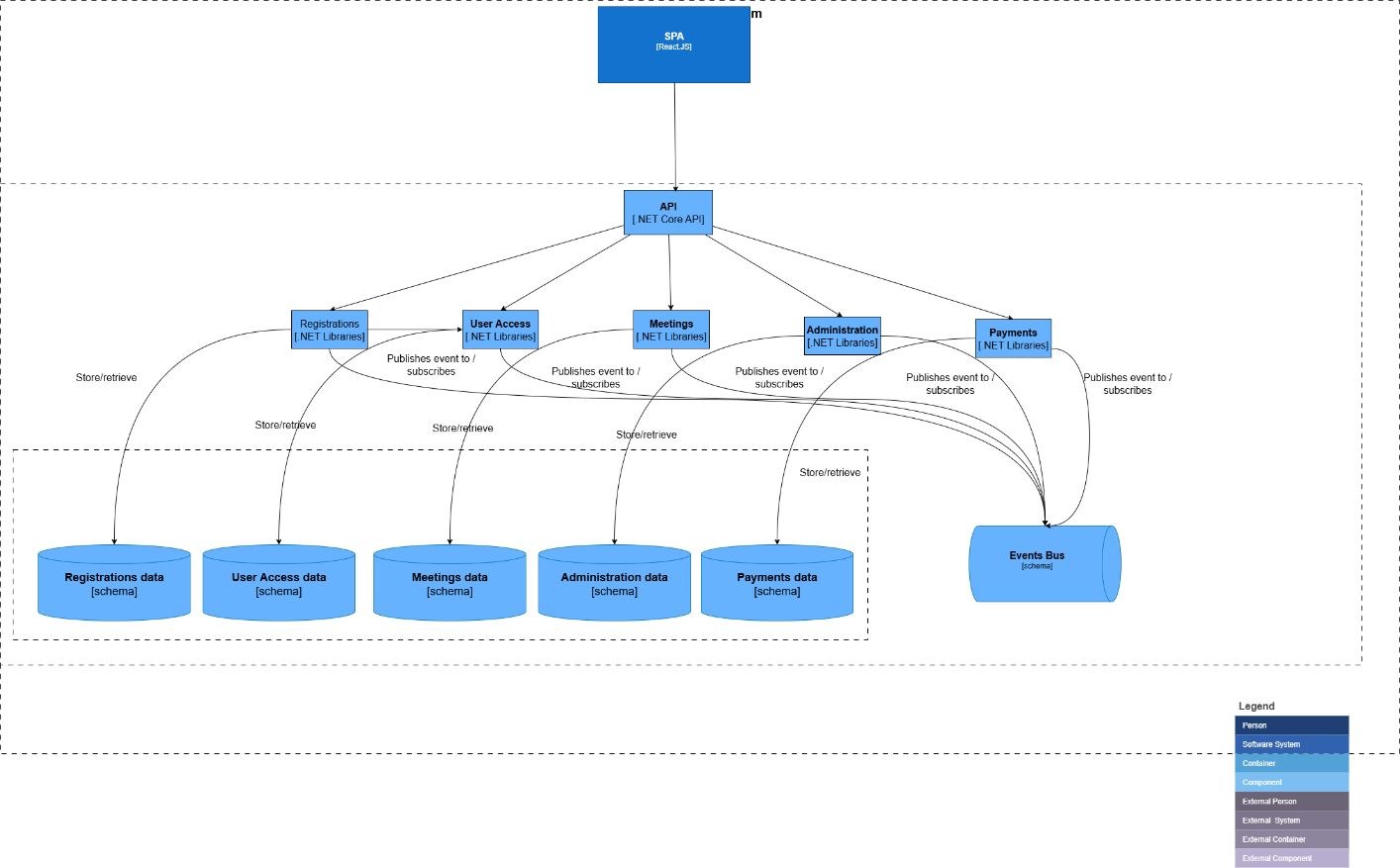
# **Bài 4:**



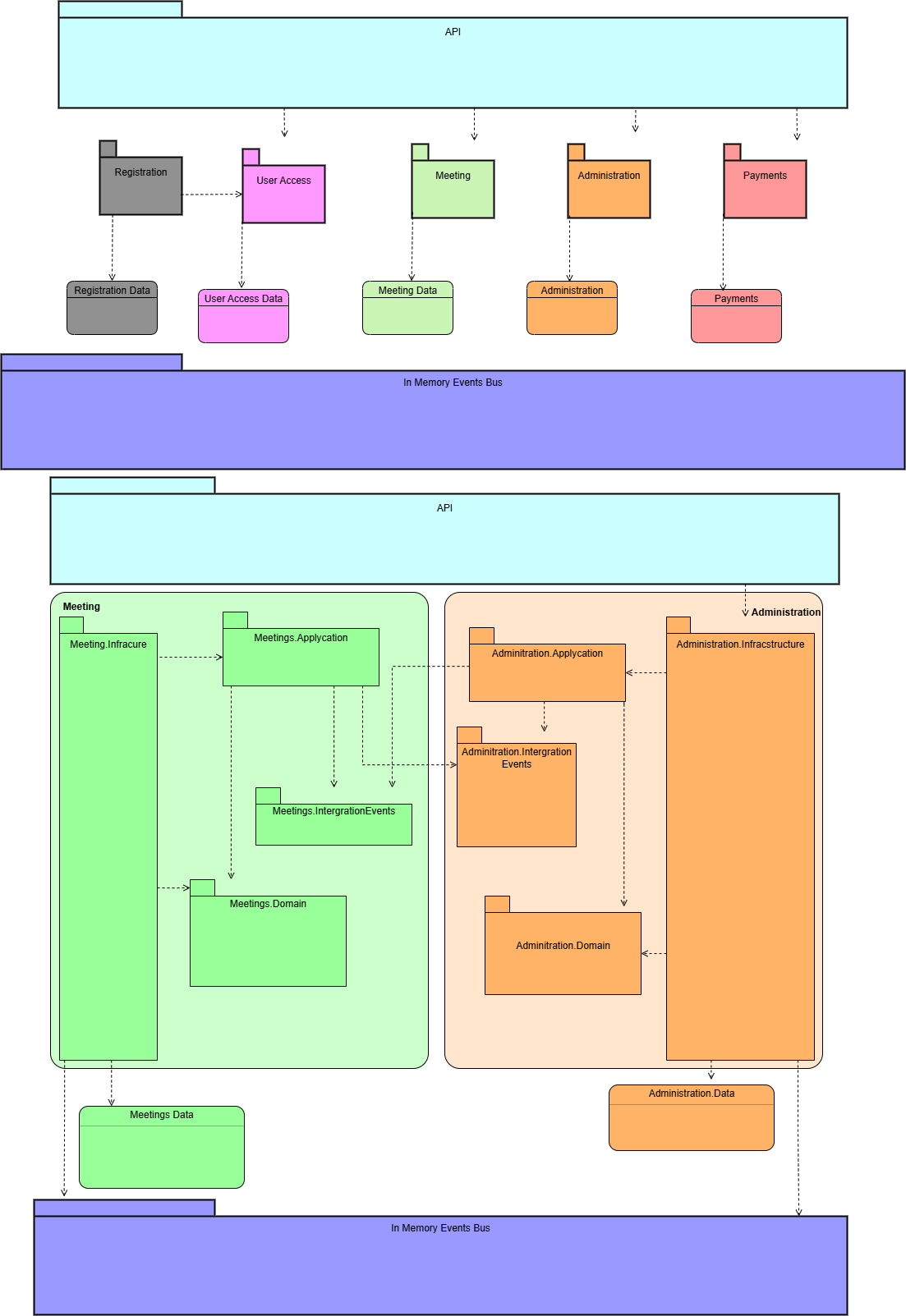
# **Bài 5:**



# **Bài 6:**



# **Bài 7:**



# **Bài 8:**



# **Bài tập ứng dụng:**

C1

A diagram of software system

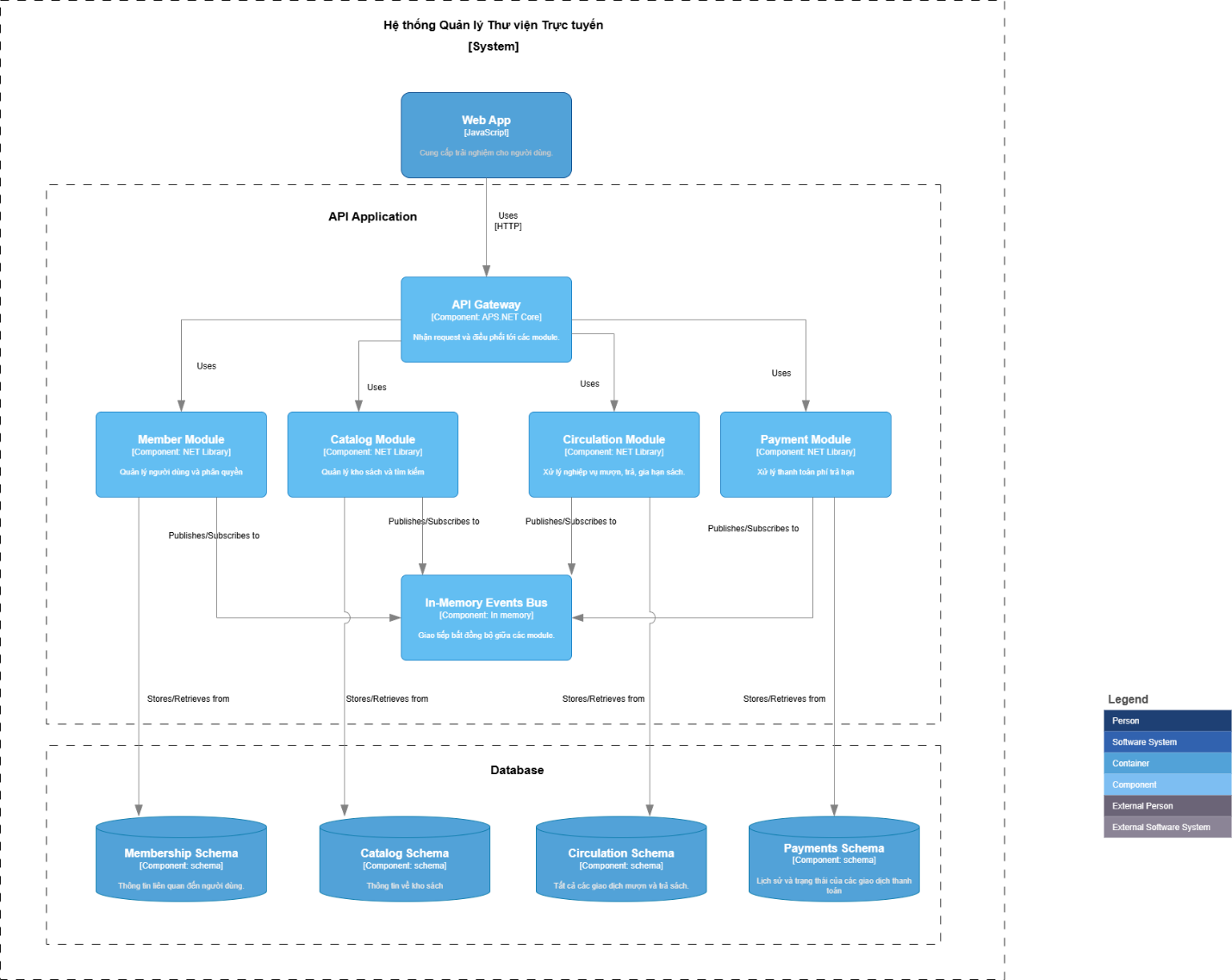
AI-generated content may be incorrect.

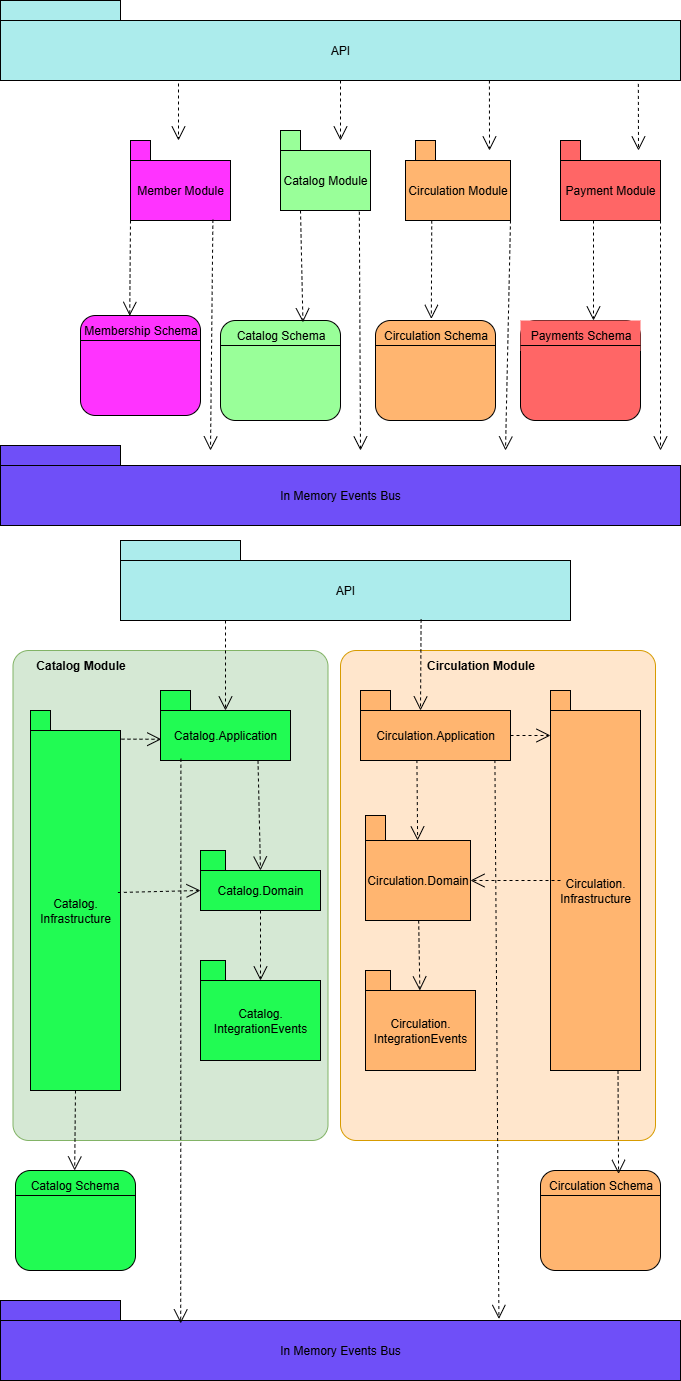
C2

A diagram of a application

AI-generated content may be incorrect.

C3





C4

A diagram of a software company

AI-generated content may be incorrect.