

## CRC Cards

**Class name:**

Template

**Description of responsibilities:**

- store name, type
- getter for name, type

**List of collaborators:**

User

**Layer of Clean Architecture to which the class belongs:**

Entity

**Class name:**

Game (subclass of the template class)

**Description of responsibilities:**

- store name, type, date, size
- getter for name, type, date, size

**List of collaborators:**

Template, Regular User

**Layer of Clean Architecture to which the class belongs:**

Entity

**Class name:**

User

**Description of responsibilities:**

- store name, email, sex
- store log in account number and password
- store a list of template/game the user create
- getter for name, email, sex

**List of collaborators:**

Template

**Layer of Clean Architecture to which the class belongs:**

Entity

**Class name:**

Admin User (subclass of the template class)

**Description of responsibilities:**

- edit the template
- add/remove the template

**List of collaborators:**

Template

**Layer of Clean Architecture to which the class belongs:**

Entity

**Class name:**

Regular User (subclass of the User class)

**Description of responsibilities:**

- create the template/game

**List of collaborators:**

Template, Game

**Layer of Clean Architecture to which the class belongs:**

Entity

**Class name:**

TemplateManager

**Description of responsibilities:**

- store all the templates
- getter all the templates
- accept requests from Users
- Initiate a new template

**List of collaborators:**

Template, User, Admin User

**Layer of Clean Architecture to which the class belongs:**

Use Case

**Class name:**

UserManager

**Description of responsibilities:**

- store and getter for a list of users

- initiate a new user with email and password
- add/remove a user
- edit information of user
- change the password with email
- check if the password is correct
- requests for login using password

**List of collaborators:**

User

**Layer of Clean Architecture to which the class belongs:**

Use Case

**Class name:**

GameManager (subclass of TemplateManager)

**Description of responsibilities:**

- store all the games
- getter all the games
- accept requests from Users
- Initiate a new game
- edit the information of a game

**List of collaborators:**

Game, Regular User

**Layer of Clean Architecture to which the class belongs:**

Use Case

**Class name:**

TemplateSystem

**Description of responsibilities:**

- receive requests from Main class
- send information to TemplateManager/GameManager

**List of collaborators:**

TemplateManager, GameManager

**Layer of Clean Architecture to which the class belongs:**

Controller

**Class name:**

UserSystem

**Description of responsibilities:**

- receive requests from Main class
- send information to UserManager

**List of collaborators:**

UserManager

**Layer of Clean Architecture to which the class belongs:**

Controller

**Class name:**

Main

**Description of responsibilities:**

- receive info from outside
- instantiate UserSystem
- instantiate TemplateSystem
- pass the information/requests to the systems/controllers

**List of collaborators:**

TemplateSystem UserSystem

**Layer of Clean Architecture to which the class belongs:**

Controller