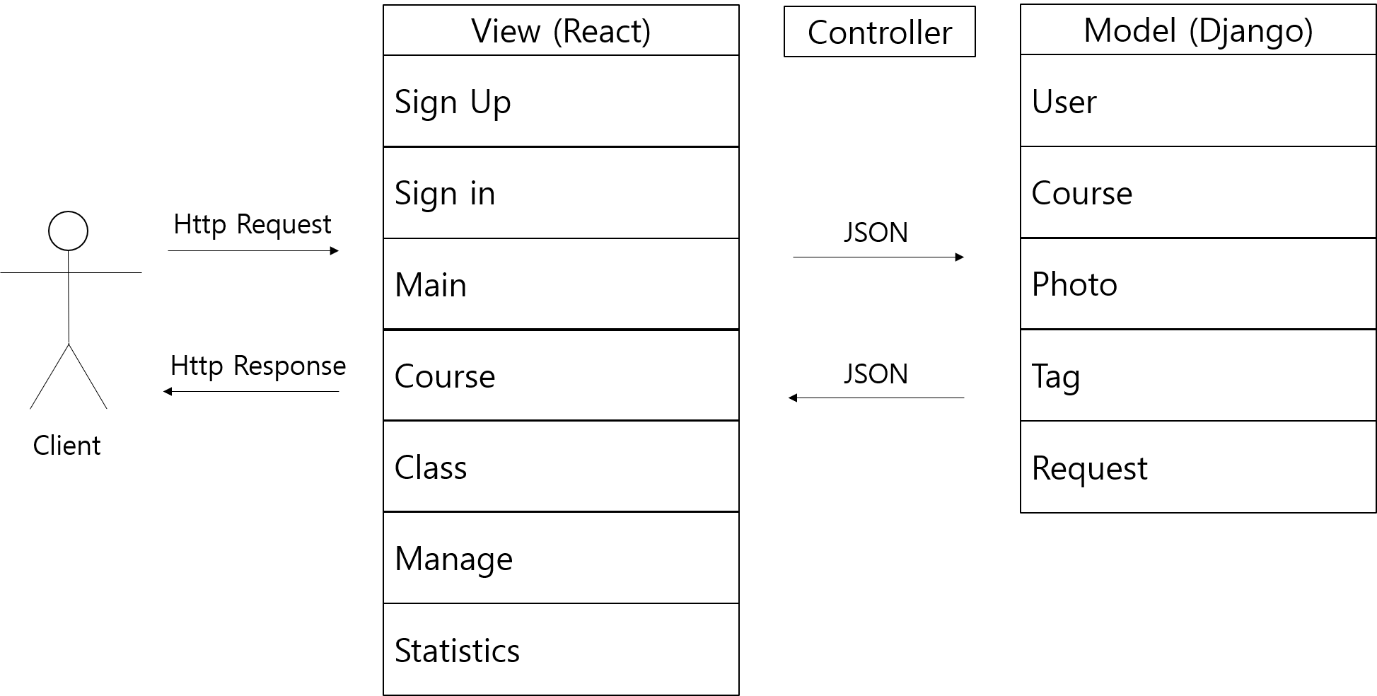
System Architecture

MVC

Here is MVC of Plain Plane.



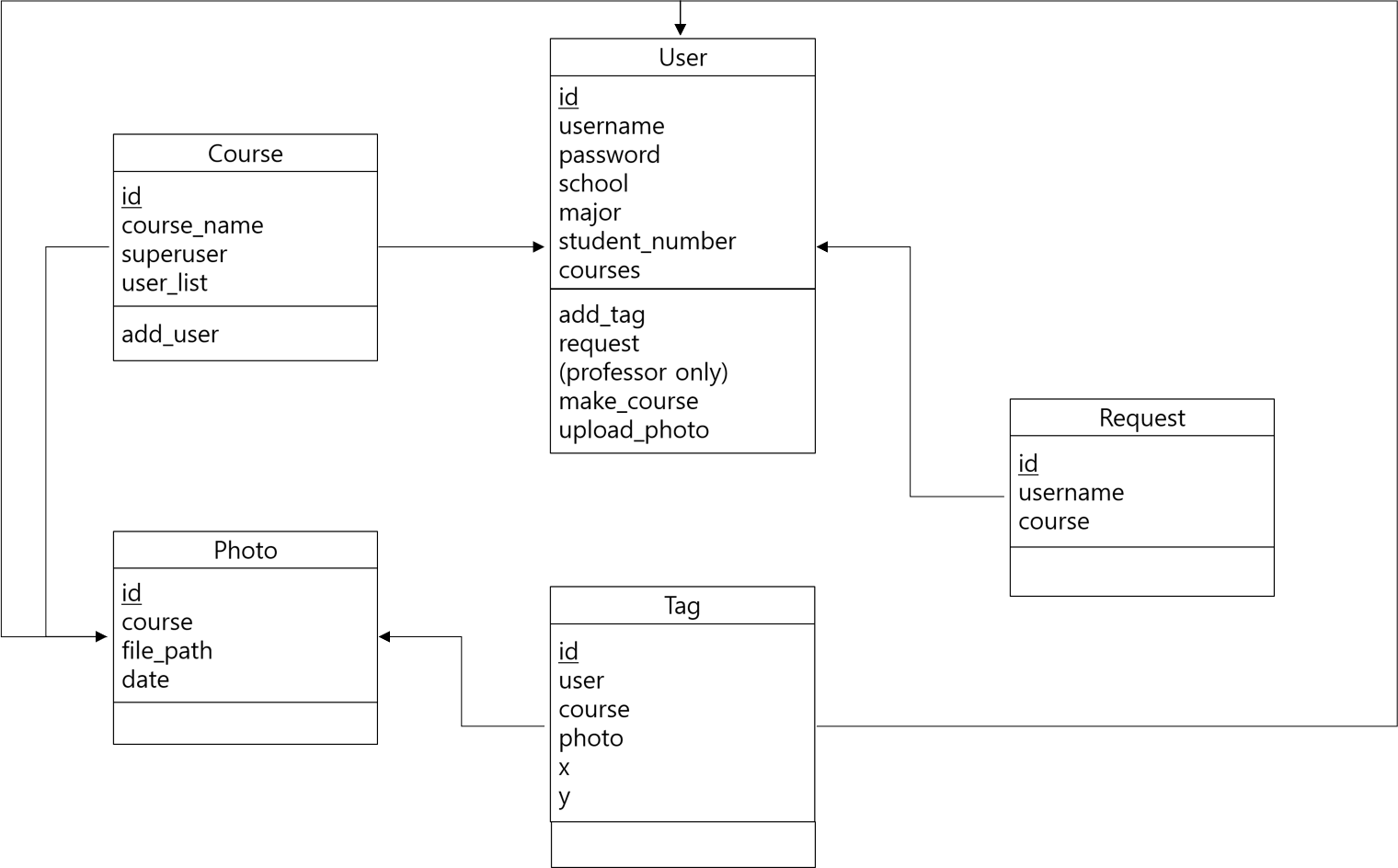
There are 7 views and 5 models. Controller will be described further in Controller part.

Model

Here is E-R (Entity Relationship) diagram for model design.

Rectangle stands for entity set, oval stands for relationship set. Entity attributes are listed inside entity rectangle, and underlined attribute indicates primary key.

Here is relation schema diagram based on E-R diagram. This is also the structure of Django Models the service will use. It also shows core methods of each model.



Rectangle stands for relation schema. Schema attributes are listed inside relation rectangle, followed by methods for Django models. Underlined attribute indicates primary key, and arrow represents foreign key constraints. Since Django saves whole object for foreign key, this diagram also follows the convention.

Backend Design

In the backend design, we use models which have been discussed in MVC architecture section.

RESTful API

Detailed specifications of RESTful APIs are as following:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Model** | **API** | **GET** | **POST** | **PUT** | **DELETE** |
| User | /signup | X | Create new user | X | X |
|  | /signin | X | Log in | X | X |
|  | /signout | Log out | X | X | X |
|  | /user/:id | Get specific user | X | X | X |
| Course | /course | X | Create new course (professor only) | X | delete |
|  | /course/:id | Get specified course | X | Modify specified course, add/delete user to course | X |
| Photo | /photo | Get photo list of course which the user belongs | Create new photo (professor only) | X | X |
|  | /photo/:id | Get specified photo | X | Modify specified Photo | delete |
| Tag | /tag | X | Create new photo (professor only) | X | X |
|  | /tag/:id | Get specified tag | X | Modify specified tag | delete |
| Request | /request | Get request list of course | Create new request | X | X |
|  | /request/:id | Get specified request | X | X | delete |