



寵物領養平台

109703023 林哲宇 資科四
109703057 劉軒宇 資科四
109703029 王睦閔 資科四

1. 題目簡介
2. 需求分析
3. 系統功能
4. ER Model
5. Relation Schema
6. 系統架構
7. Test
8. 心得、收穫與建議

題目簡介

題目簡介

寵物領養平台是一個專為希望送養或領養寵物的使用者而設計的線上平台。

- **寵物刊登服務：**

使用者可以在平台上刊登欲送養的寵物資訊，包括寵物的種類、品種、性別、年齡以及照片等，這些資訊幫助欲領養寵物的用戶了解寵物的背景與特徵。

- **寵物搜尋與查看：**

使用者可以通過平台查看被送養的寵物資訊，他們可以根據自己的需求和條件，如寵物的種類、性別、年齡等進行搜索，以找到符合他們需求的寵物。

- **寵物媒合**

使用者可以設定理想寵物條件，系統會媒合與使用者設定條件相符的寵物，並且寄信通知使用者。

這個平台旨在**促進寵物的再領養，提高送養和領養的便捷性與透明度**，同時幫助更多寵物找到一個溫馨的家庭。

需求分析

需求分析

使用者

- 使用者包含使用者名稱、電子郵件、密碼以及聯絡資訊等屬性。
- 使用者可以註冊、登入帳號
- 使用者可以瀏覽寵物資訊
- 使用者可以上架寵物資訊、編輯以及刪除寵物資訊。
- 使用者可以新增、編輯理想寵物條件。

寵物

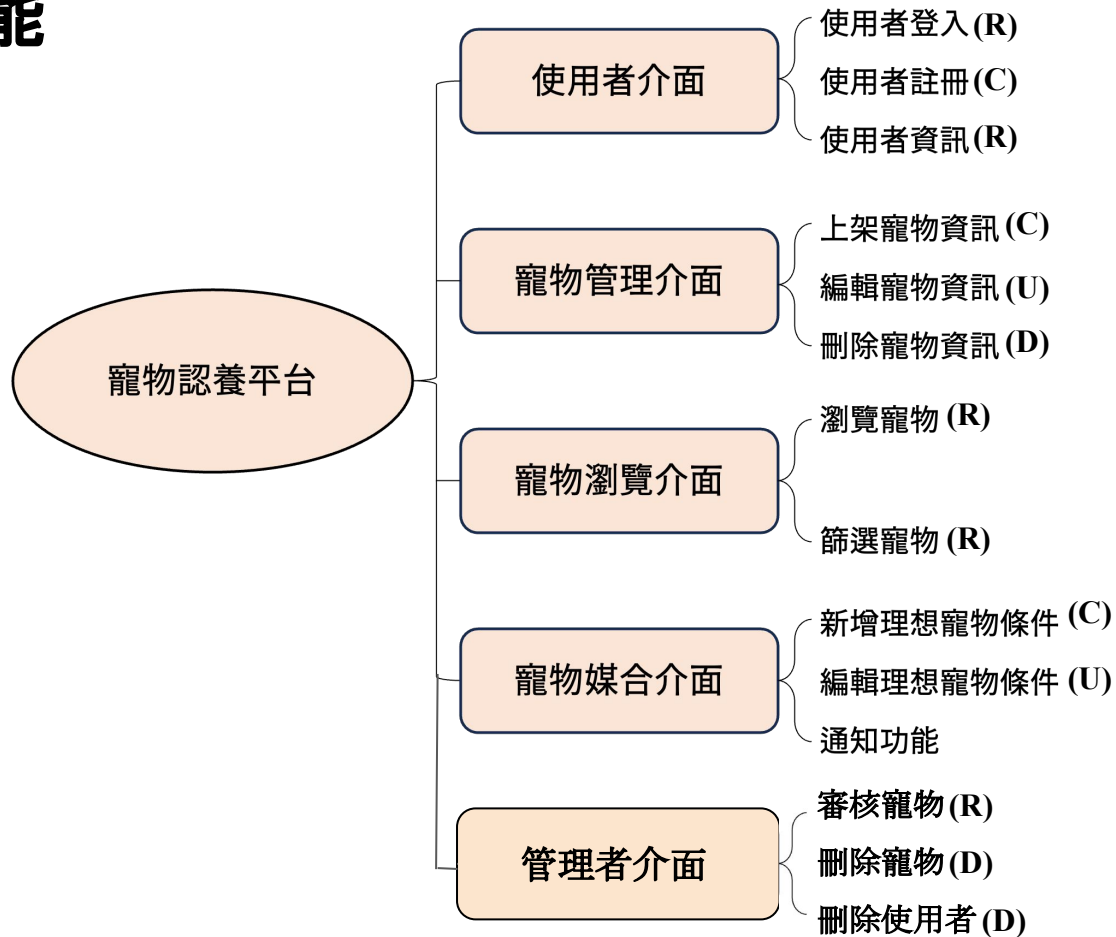
- 寵物包含寵物名稱、種類、品種、性別、大小、地區、年齡、毛色、是否結紮、照片以及其他資訊等屬性。

管理員

- 管理員包含管理員名稱、電子郵件、密碼等屬性。
- 管理員可以審核寵物資訊。
- 管理員可以刪除寵物、使用者。

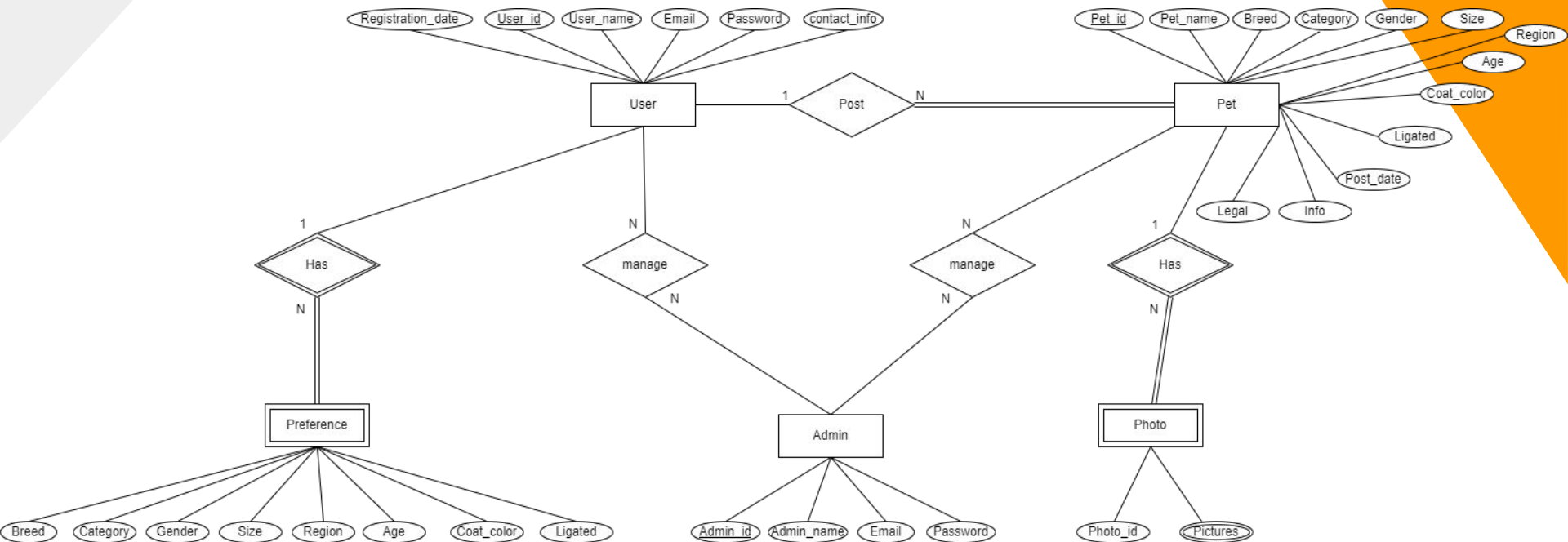
系統功能

系統功能



ER Model

ER model



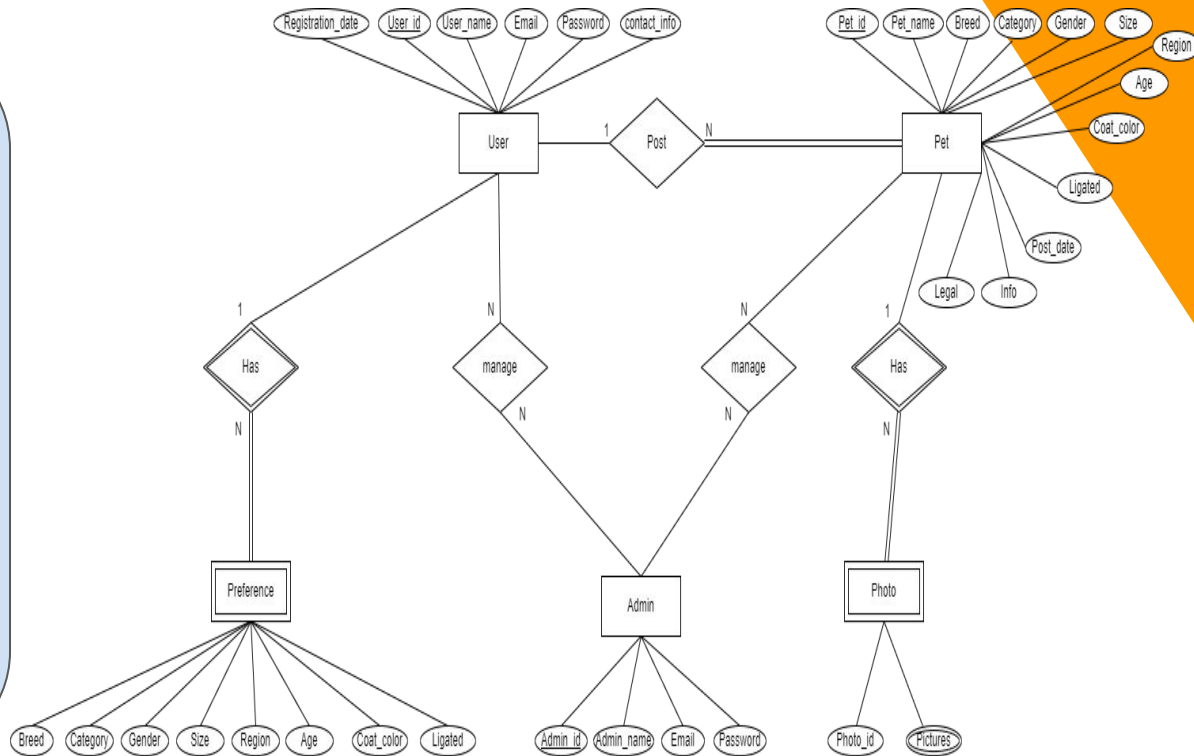
ER model

- 5 個 Entities

- 使用者
- 寵物
- 相片
- 理想寵物條件
- 管理員

- 5 個 Relationship

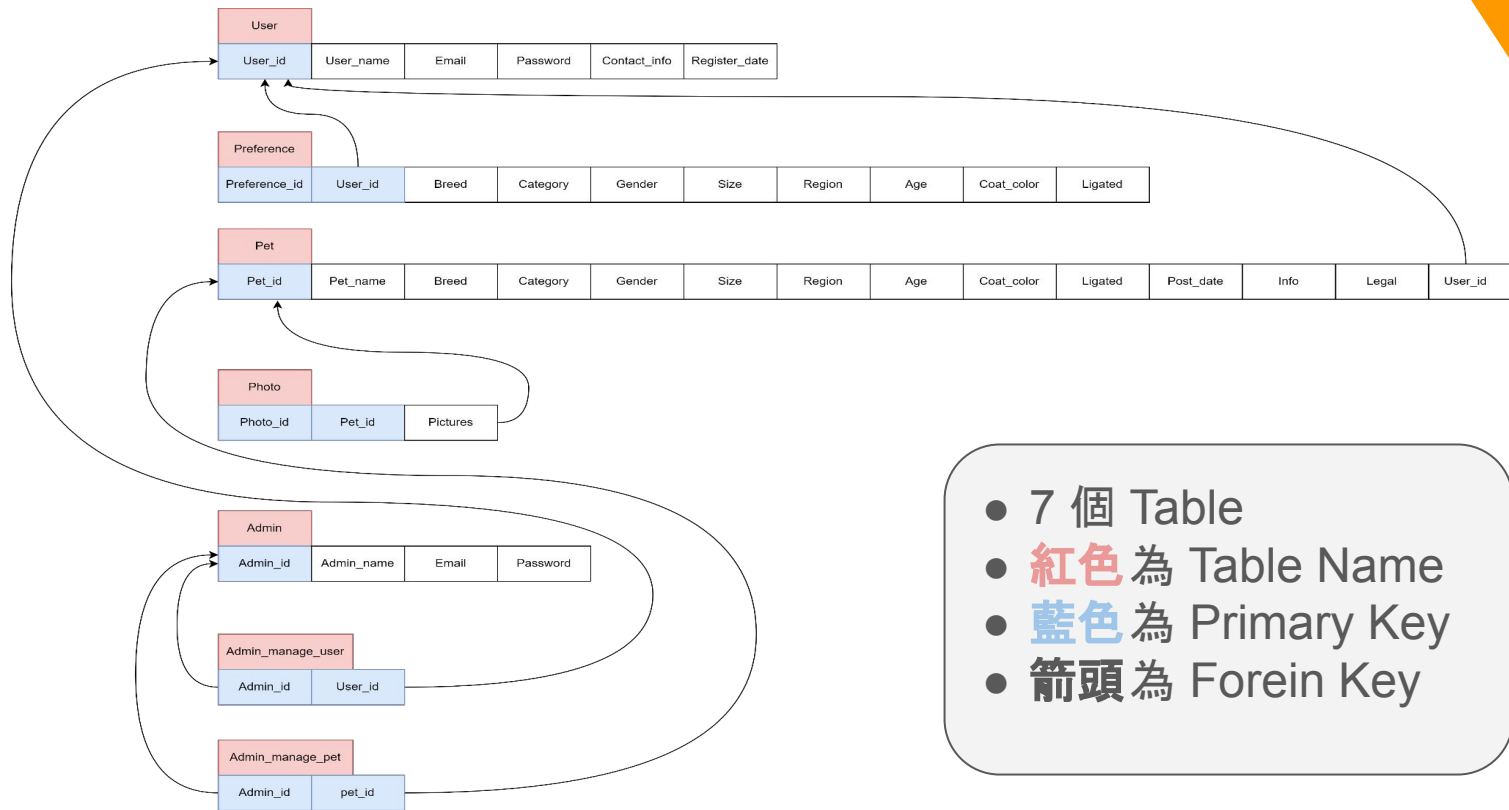
- 使用者 **上架** 寵物資訊
- 使用者 **新增** 理想寵物條件
- 寵物 **有** 多張照片
- 管理員 **管理** 寵物資訊
- 管理員 **管理** 使用者資訊





Relational Schema

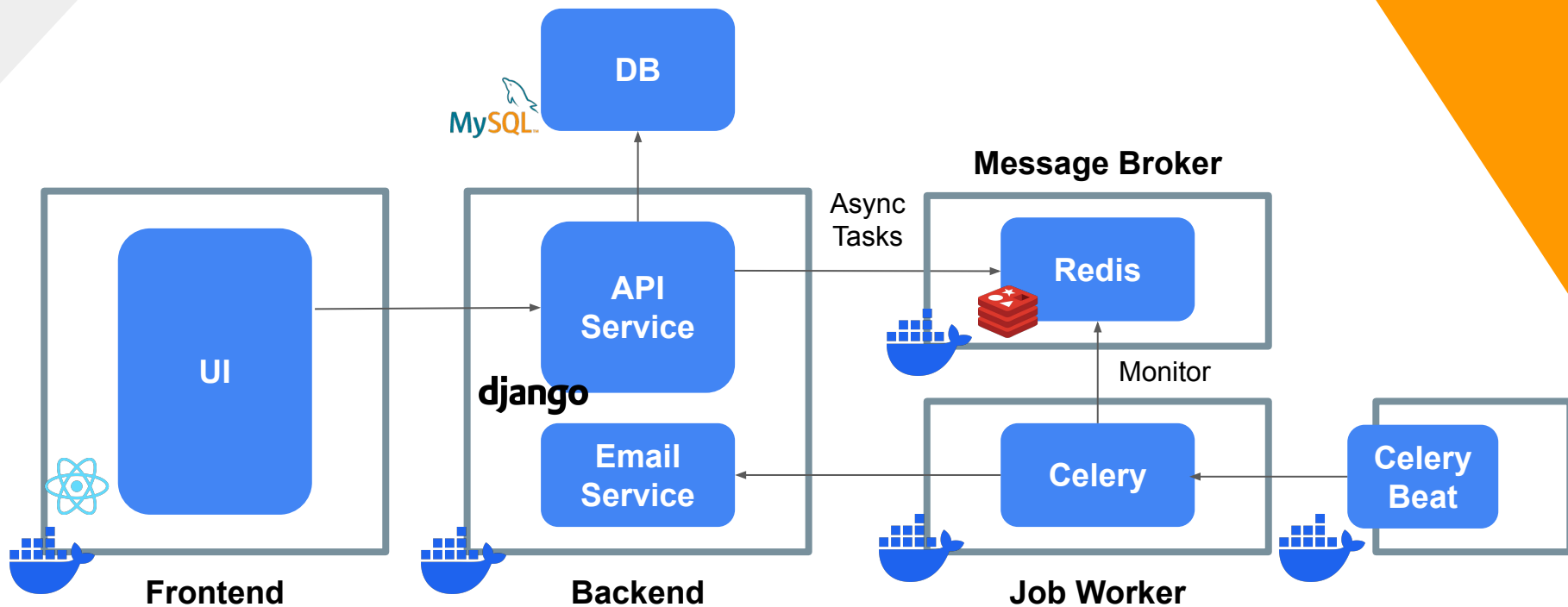
Relational Schma



- 7 個 Table
- 紅色為 Table Name
- 藍色為 Primary Key
- 箭頭為 Forein Key

系統架構

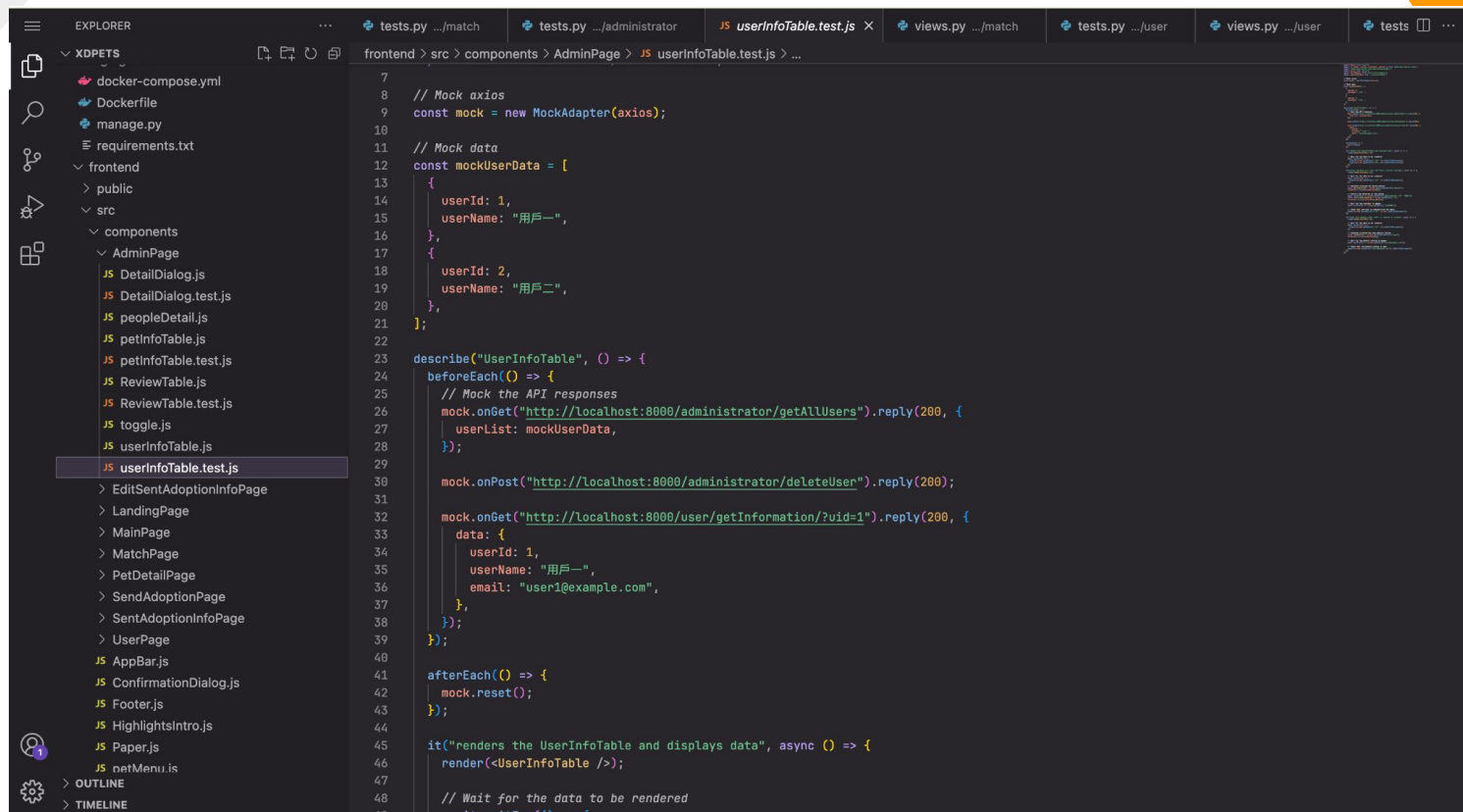
系統架構





Test

Testing: Frontend Test Case Example

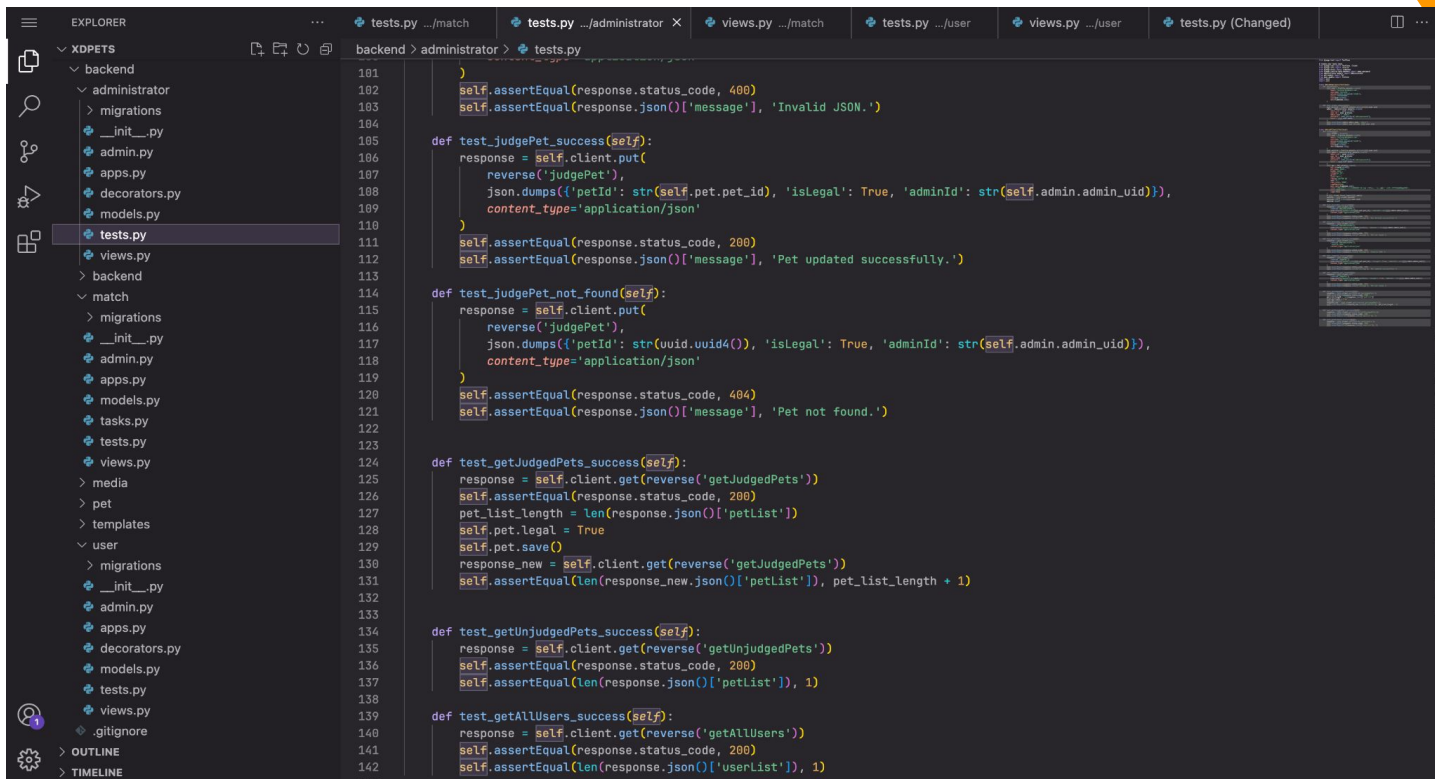


The image shows a screenshot of a code editor (VS Code) with a dark theme. The left sidebar displays the Explorer view, showing a file tree for a project named 'XDPEts'. The file tree includes a 'frontend' directory, which contains a 'src' directory, and a 'components' directory. The 'components' directory is expanded, showing a list of files including 'AdminPage', 'DetailDialog.js', 'DetailDialog.test.js', 'peopleDetail.js', 'petInfoTable.js', 'petInfoTable.test.js', 'ReviewTable.js', 'ReviewTable.test.js', 'toggle.js', 'userInfoTable.js', and 'userInfoTable.test.js'. The 'userInfoTable.test.js' file is selected and highlighted.

The main editor area displays the content of 'userInfoTable.test.js'. The code is written in JavaScript and uses Jest for testing. It includes mock data for users and mock API responses for the 'User Info Table' component. The code is as follows:

```
7 // Mock axios
8 const mock = new MockAdapter(axios);
9
10 // Mock data
11 const mockUserData = [
12   {
13     userId: 1,
14     userName: "用戶一",
15   },
16   {
17     userId: 2,
18     userName: "用戶二",
19   },
20 ];
21
22 describe("UserInfoTable", () => {
23   beforeEach(() => {
24     // Mock the API responses
25     mock.onGet("http://localhost:8080/administrator/getAllUsers").reply(200, {
26       userList: mockUserData,
27     });
28
29     mock.onPost("http://localhost:8080/administrator/deleteUser").reply(200);
30
31     mock.onGet("http://localhost:8080/user/getInformation?uid=1").reply(200, {
32       data: {
33         userId: 1,
34         userName: "用戶一",
35         email: "user1@example.com",
36       },
37     });
38   });
39
40   afterEach(() => {
41     mock.reset();
42   });
43
44   it("renders the UserInfoTable and displays data", async () => {
45     render(<UserInfoTable />);
46
47     // Wait for the data to be rendered
48     await waitFor(() => {
```

Testing: Backend Test Case Example



The screenshot displays a code editor with a file explorer on the left and a code editor on the right. The file explorer shows a project structure with folders like 'backend', 'administrator', 'match', 'media', 'pet', 'templates', and 'user'. The 'tests.py' file is selected under the 'administrator' folder. The code editor shows the following test cases:

```
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
```

```
self.assertEqual(response.status_code, 400)
self.assertEqual(response.json()['message'], 'Invalid JSON.')

def test_judgePet_success(self):
    response = self.client.put(
        reverse('judgePet'),
        json.dumps({'petId': str(self.pet.pet_id), 'isLegal': True, 'adminId': str(self.admin.admin_uid)}),
        content_type='application/json'
    )
    self.assertEqual(response.status_code, 200)
    self.assertEqual(response.json()['message'], 'Pet updated successfully.')

def test_judgePet_not_found(self):
    response = self.client.put(
        reverse('judgePet'),
        json.dumps({'petId': str(uuid.uuid4()), 'isLegal': True, 'adminId': str(self.admin.admin_uid)}),
        content_type='application/json'
    )
    self.assertEqual(response.status_code, 404)
    self.assertEqual(response.json()['message'], 'Pet not found.')

def test_getJudgedPets_success(self):
    response = self.client.get(reverse('getJudgedPets'))
    self.assertEqual(response.status_code, 200)
    pet_list_length = len(response.json()['petList'])
    self.pet.legal = True
    self.pet.save()
    response_new = self.client.get(reverse('getJudgedPets'))
    self.assertEqual(len(response_new.json()['petList']), pet_list_length + 1)

def test_getUnjudgedPets_success(self):
    response = self.client.get(reverse('getUnjudgedPets'))
    self.assertEqual(response.status_code, 200)
    self.assertEqual(len(response.json()['petList']), 1)

def test_getAllUsers_success(self):
    response = self.client.get(reverse('getAllUsers'))
    self.assertEqual(response.status_code, 200)
    self.assertEqual(len(response.json()['userList']), 1)
```



分工

分工

工人	分工	貢獻百分比
劉軒宇	前端設計、前端開發、前端測試、報告	33.3333333%
林哲宇	後端開發、後端測試、系統說明書、簡報	33.3333333%
王睦閔	資料庫設計(ER Model、Relational Schema)、後端開發、系統說明書、報告	33.3333333%

心得、收穫與建議

心得、收穫與建議

- FrontEnd
 - 頁面主要就照著前端的設計做，在串接時才發生比較多問題，而需要與後端進行比較多的溝通以完成頁面的相關功能。
- BackEnd
 - 當初確定好系統功能後，有撰寫 API 規格書，不過一開始 API 規格書並沒有寫好，會造成前端寫到一半發現缺少某些 API，需要跟後端溝通需要新增 API。
 - 熟悉 Django 的使用方式，如 migrate 會自動建立資料庫的 Table 而不用自己先建好。
- 了解到時間不能壓得太死，需要預留時間，避免造成遇到無法解的問題或是突發狀況而導致無法產出成品。



<http://35.201.245.108:3000/>

Demo

前端頁面:登入



Sign in

username *

Password *

☐ Remember me

SIGN IN

[Forgot password?](#)

[Don't have an account? Sign Up](#)

Copyright © 畢業小隊 2024.

前端頁面：新使用者註冊



Sign up

Username *

Email Address *

Password *

phone *



I want to receive inspiration, marketing promotions and updates via email.

SIGN UP

[Already have an account? Sign in](#)

Copyright © 畢業小隊 2024.

寵物認養

首頁 -> 寵物認養

此區的送養動物均由送養人刊登，所有送養人電話均經過簡訊驗證，如有任何領養、認養問題請直接與送養人聯繫。領養前請謹慎考慮，不隨意棄養。重要提醒！自即日起，查看領養資訊均需驗證手機號碼，且每次查看均會記錄相關登入資訊，請有他意者自重。本站紀錄之相關個人訊息均僅供檢察或司法警察機關調閱，不會提供給任何第三方使用

區域

不限區域

性別

不限性別

種類

不限種類

搜尋



皮蛋

種類: 貓
性別: 公
地區: 新北市 三區
品種: 米克斯



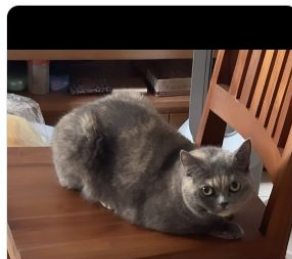
迪迪

種類: 狗
性別: 公
地區: 桃園市 桃園區
品種: 黑柴



六六

種類: 狗
性別: 公
地區: 台中市 太平區
品種: 柴犬



碰碰

種類: 貓
性別: 母
地區: 台中市 北屯區
品種: 雜色

前端頁面: 寵物認養

刊登送養

[首頁](#) > [刊登送養](#)

Send for Adoption

Pet Name *

Breed *

Type *

Gender *

Size *

Hair Color *

Age *

Ligation *

City *

Area *

Regulations for Sending Pets

刊登人使用由 畢業小隊 寵物領養平台(以下簡稱「畢業小隊」)所提供之送養資訊刊登服務(以下簡稱「本服務」)時，均受到本「刊登條款」規範。

1. 畢業小隊 要求所有刊登送養資訊及查看送養資訊者均須加入會員且強制驗證手機號碼，是為了保證所有查看資訊的人都確實的留下紀錄，請送養人注意，如將動物送養後有查覺異常或有任何虐待事項，請立即通報政府單位，我們將配合政府單位之發文要求提供相關紀錄資料。

前端頁面：刊登送養

由於所有送養訊息均須經過審核後才會公開，請確實檢查以上資料是否正確，送出後將無法修改內容。

Description *

0 / 1000 words

照片上傳 - 請務必上傳 (最多四張)



資訊送出後需經過審核，因此不會立即出現在列表中，請耐心等待審核勿重覆刊登

★ 近期有會員回報平台，有些領養人會留下假個資給送養人，為避免日後無法聯絡，請務必在送養當下確認領養人資料及聯絡方式是否正確！

★ 每一個生命都應被珍惜，社會上虐貓/犬、棄貓/犬等案件層出不窮，作為送養人請務必謹慎（如：領養人年紀、生活環境、經濟能力、家人是否同意、可否後續家訪、後續追蹤、協議書簽定...等。）

謹慎！希望所有送養的毛孩，我們都能找到溫暖的家。

2. 送養資訊不限刊登時間，因此將動物確定送養後請回網站關閉送養資訊，避免造成自身困擾。

3. 目前不開放剛出生或尚未開眼之動物送養。

4. 畢業小隊 僅為資訊交流平台，無法保證刊登內容之正確性，刊登人對其刊登內容及送養之行為應負完全之責任，刊登人應提供確實之刊登內容，並自負相關法律責任。

5. 畢業小隊 對刊登資訊保有最終篩選、修改、刪除及決定刊登與否之權利。

6. 畢業小隊 秉持以領養代替購買之精神，禁止以下類型之刊登內容：

a. 要求補貼、認養價、營養費、結紮費等一切任何額外費用或有實際販賣行為。

b. 捏造不實資訊。

c. 特意收集個人資料。

如有違反上述條款之情事，畢業小隊 可能將其進行會員停權或封鎖之處置。

7. 刊登人應確定刊登之內容未使用未經授權之文字、圖片、影音或任何形式之資訊。

8. 刊登人同意授權於本服務刊登之資訊內容無償提供第三人分享、轉載或推廣。

9. 資訊的審核時間約為 0-3 天，請耐心等待。




☐ 我已閱讀並同意本刊登條款

前端頁面：刊登送養

送養資訊

首頁 -> 會員中心 -> 送養資訊

新增送養

Actions	Name	Type	Breed	Gender	Age	Link
	pando	熊貓	米克斯	公	5	寵物連結🔗
	pandow	熊貓	王者	母	20	寵物連結🔗
	胖達	狗	貓熊	母	5	寵物連結🔗

前端頁面：送養資訊



pando

種類: 熊貓

品種: 米克斯

性別: 公

體型: 中型

毛色: 黑白色

年紀: 5

絕育: 已絕育

所在地: 台北市文山區

刊登日期: 2024-06-01T17:12:30.931Z

✕ 不需要簽署愛心認養切結書

✓ 需要同意接受後續追蹤

前端頁面：寵物資訊

媒合系統

首頁 -> 會員中心 -> 媒合介面

新增媒合

管理	狀態	種類	品種	地區	性別	年齡	體型	結紮	毛色
 	未通知	狗	柴犬	彰化縣	公	1	小	是	黑
 	未通知	貓	波斯貓	嘉義市	母	2	迷你	是	白

前端頁面：媒合系統

前端頁面：管理員介面 (審核寵物)

畢業小隊

寵物領養平台

管理員審核

Actions

☒ 通過 ☐ 刪除

本站所有資訊來源為政府資料開放平台及會
有任何疑問請與我們聯絡

© 2024 Copyright graduate.tw

詳細資訊

pet_id dd8a26bb-b262-410c-9352-2c8d4aac403f	pet_name 小八
breed 秋田犬	category 狗
gender 公	size 中
region 臺北市 文山區	age 1
coat_color 金色	ligated 已結紮
post_date 2024-06-13T15:21:17.968Z	info 小八出生在秋田縣大館市附近的一個農場，原名阿八（ハチ），他被上野英三郎帶到東京澀谷作為寵物飼養，八公每天都會前往澀谷站迎接主人回家。

關閉

查看使用者

Pet Name

小八

詳細資訊

page: 25 1-1 of 1

人權益由刊登者自行負擔相關責任。如

前端頁面：管理員介面 (管理寵物)

🐾 畢業小隊

寵物領養平台

會員中心

領養認證

刊登送養









管理員介面

首頁 -> 會員中心 -> 管理員介面

管理員審核

查看寵物

查看使用者

Actions	Status	User ID	Pet ID	Pet Name
 Delete	已上架	59d9730c-e77c-4ada-95a9-5c7116b471bc	03333366-6fa8-e616-84ef-42cd45dbcb89	皮蛋  詳細資訊
 Delete	已上架	59d9730c-e77c-4ada-95a9-5c7116b471bc	05538056-3551-908c-cec7-5055fe022493	迪迪  詳細資訊
 Delete	已上架	59d9730c-e77c-4ada-95a9-5c7116b471bc	0b6dc4a7-686f-ebc7-bfe6-99d9eaa94b1e	六六  詳細資訊
 Delete	已上架	59d9730c-e77c-4ada-95a9-5c7116b471bc	11671c34-80e8-dceb-0a1f-fb3d07db4b28	碰碰  詳細資訊

前端頁面：管理員介面 (管理使用者)



管理員審核

查看寵物

查看使用者

Actions	User ID	User Name	
Delete	0844d078-9054-469b-9daf-9f1d4b1d4cd5	jerry05	詳細資訊
Delete	09203c44-fe40-4dfb-978d-8a0e0c7deb02	大V天龍	詳細資訊
Delete	121baede-f121-4e76-affd-da457f69fe5d	jerry04	詳細資訊
Delete	1645e4a4-5173-49a1-90cc-256761e05533	潘島均2號	詳細資訊