訂餐外送平台網頁服務

登入/驗證/註冊系統 (Login and Registration System)

2021.04~2021.05

大綱

- 簡報流程
  - 開發需求/規格說明
  - 開發方式說明
    - ✓開發流程架構
    - ✓輔助元件
  - 實務作品呈現
  - 需求/規格與開發功能對照
  - 開發心得分享

開發需求/規格說明

建構使用者登入系統

1)使用Hibernate建構前端與後端的連結

2)使用Spring Security做使用者存取資源的權限管理

3)使用Json Web Token做使用者權限的驗證

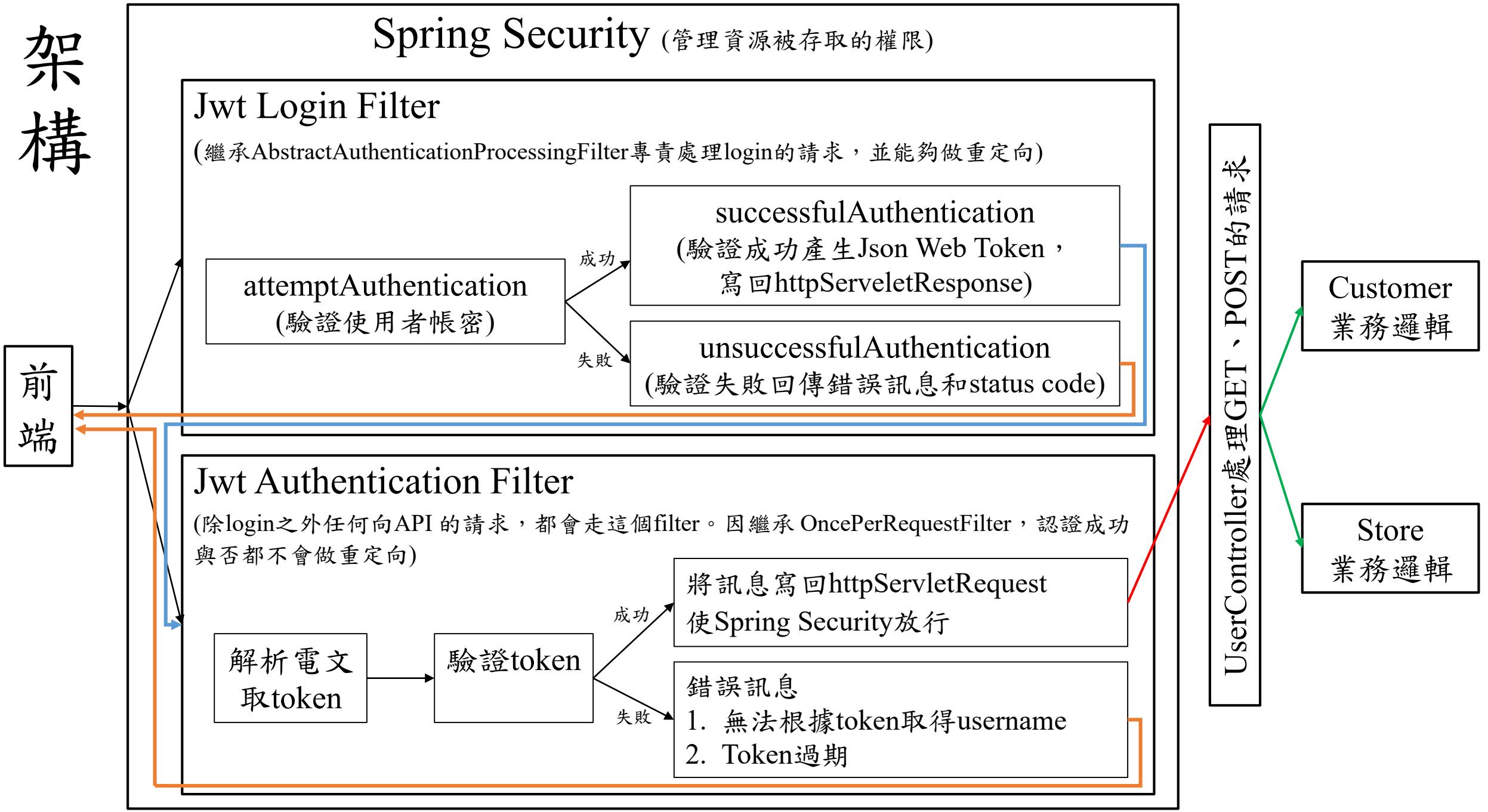
資料表名稱	*user_account*			
資料表說明	使用者帳號與密碼資料			
欄位名稱	資料型態	長度	必填	欄位說明
id	VARCHAR	255	Y	PK, 帳號id, UUID
social_id	VARCHAR	255	N	社群帳號id
name	VARCHAR	100	Y	使用者名稱
pwd	VARCHAR	255	Y	使用者密碼
role_id	int		Y	角色id
create_date	TIMESTAMP		Y	建立時間
modify_date	TIMESTAMP		Y	修改時間
active	smallint		Y	資料是否啟用的識別值-- 0:未啟用; 1:啟用. 預設值為1

資料表名稱	*user_role*			
資料表說明	用戶角色資料			
欄位名稱	資料型態	長度	必填	欄位說明
id	int	auto increment	Y	PK, 角色id. 1:店家; 2: 用戶
name	CHAR	100	Y	角色名稱

開發方式說明

登入/驗證/註冊系統 (Login and Registration System)

架構


Spring Security (管理資源被存取的權限)
Jwt Login Filter
(繼承AbstractAuthenticationProcessingFilter專責處理login的請求，並能夠做重定向)
attemptAuthentication
(驗證使用者帳密)
成功
失敗
successfulAuthentication
(驗證成功產生Json Web Token，
寫回httpServeletResponse)
unsuccessfulAuthentication
(驗證失敗回傳錯誤訊息和status code)
前端
Jwt Authentication Filter
(除login之外任何向API 的請求，都會走這個filter。因繼承 OncePerRequestFilter，認證成功與否都不會做重定向)
解析電文取token
驗證token
成功
失敗
將訊息寫回httpServletRequest
使Spring Security放行
錯誤訊息
1. 無法根據token取得username
2. Token過期
UserController處理GET、POST的請求
Customer
業務邏輯
Store
業務邏輯

輔助元件 1

- Model
  1. User
     Hibernate設定與Role為多對一的關係
  2. Role
     Hibernate設定與User為一對多的關係
  3. UserInfoVo
     用來作為使用者在前端輸入資訊的容器，供API做進一步的處理
  4. UserPrinciple
     作為儲存UserDetails的容器，供產生token時使用

資料表名稱	*user_account*			
資料表說明	使用者帳號與密碼資料			
欄位名稱	資料型態	長度	必填	欄位說明
id	VARCHAR	255	Y	PK, 帳號id, UUID
social_id	VARCHAR	255	N	社群帳號id
name	VARCHAR	100	Y	使用者名稱
pwd	VARCHAR	255	Y	使用者密碼
role_id	int		Y	角色id
create_date	TIMESTAMP		Y	建立時間
modify_date	TIMESTAMP		Y	修改時間
active	smallint		Y	資料是否啟用的識別值-- 0:未啟用; 1:啟用. 預設值為1

資料表名稱	*user_role*			
資料表說明	用戶角色資料			
欄位名稱	資料型態	長度	必填	欄位說明
id	int	auto increment	Y	PK, 角色id. 1:店家; 2: 用戶
name	CHAR	100	Y	角色名稱

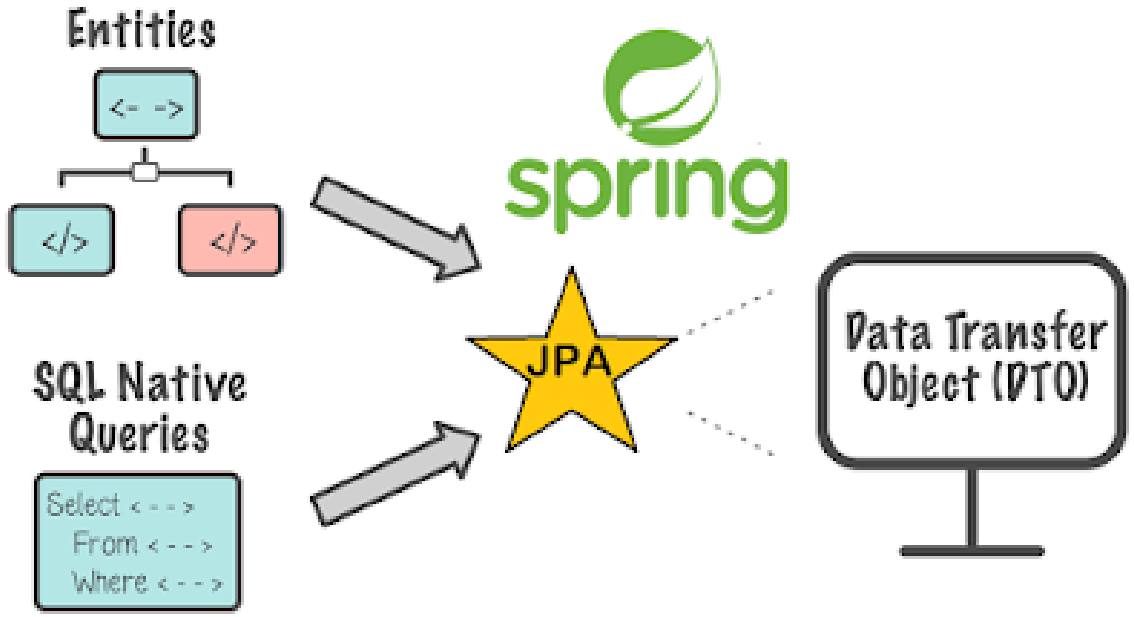
輔助元件 2

- Repository
  1. UserRepository

  繼承JpaRepository，能夠對User做新刪改查的DAO

  2. RoleRepository

  繼承JpaRepository，能夠對Role做新刪改查的DAO

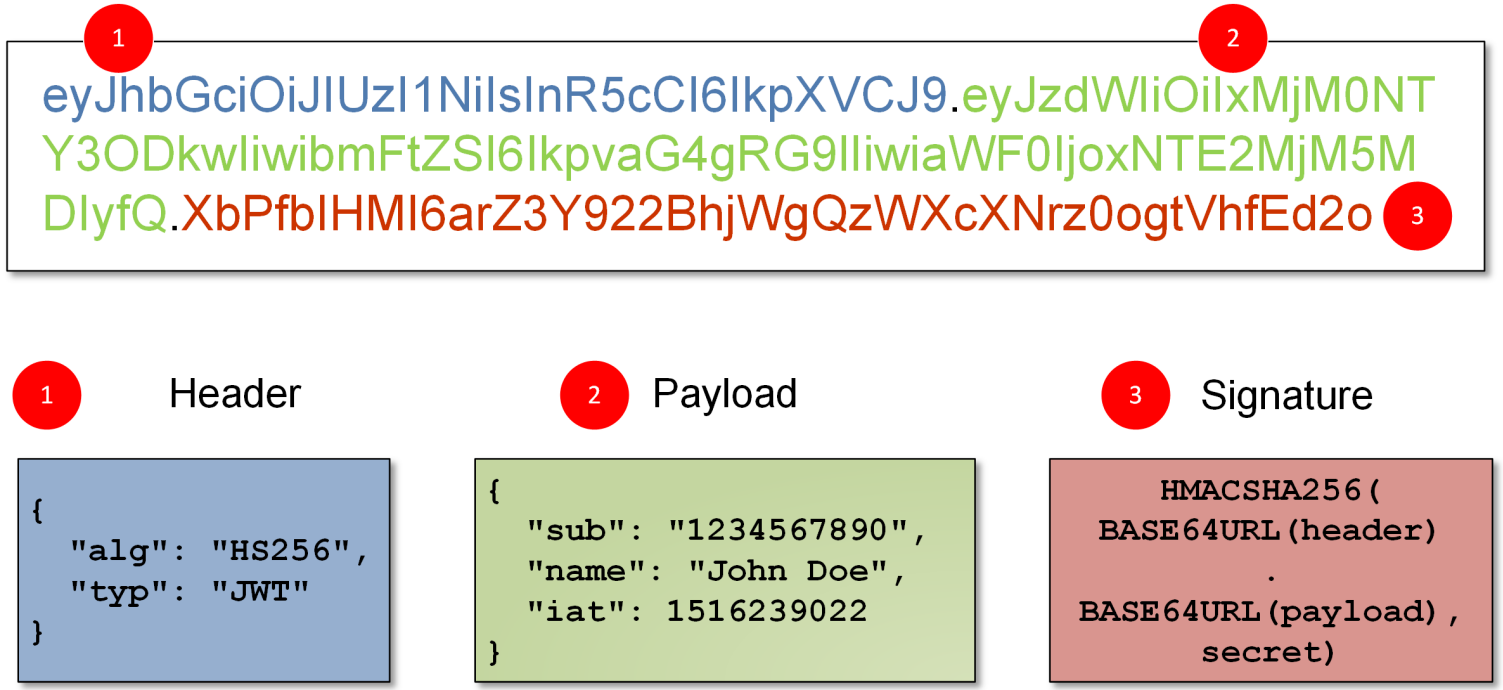


輔助元件 3

- Uitility

1. JWTUtility

   token相關的工具包，裡面包含產生、驗證、查找token資訊的方法

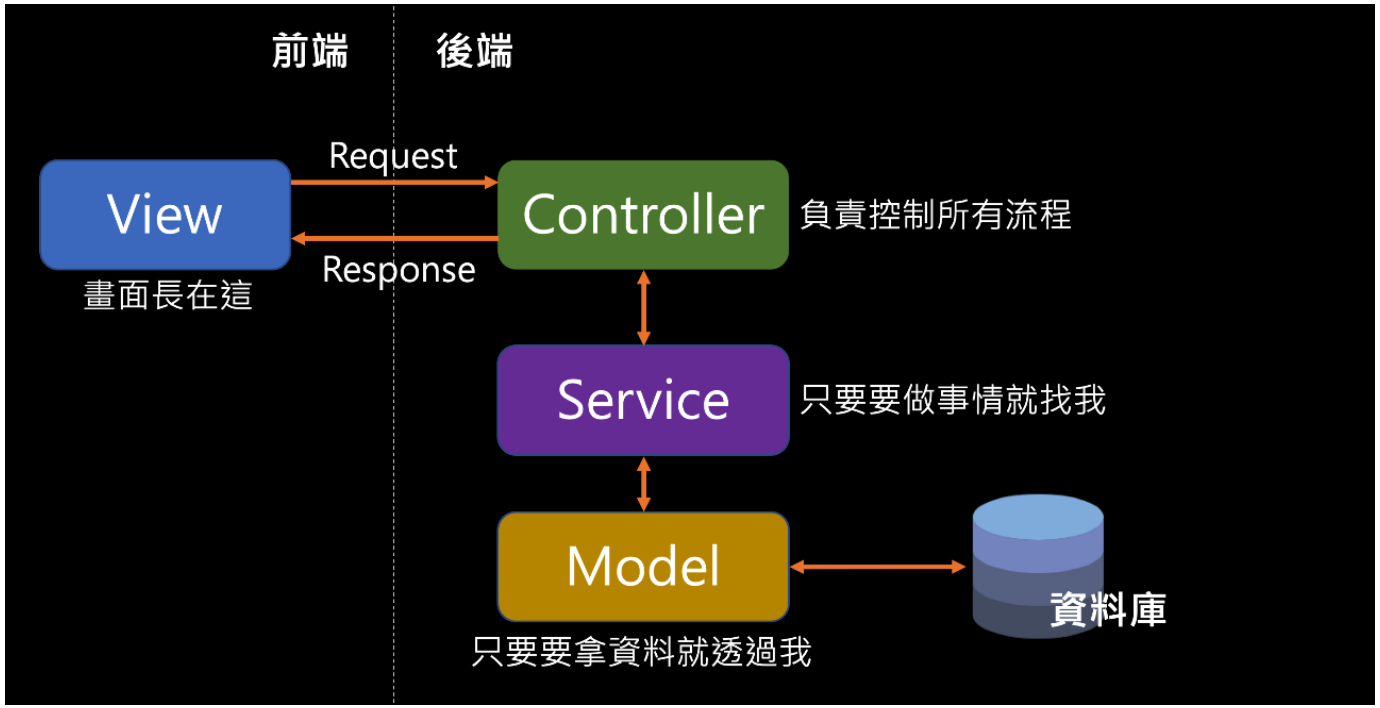


輔助元件 4

- Service

1. UserDetailsServiceImpl
   (實現UserDetailService.interface)
   實現儲存、查找使用者資訊的功能

2. TokenAuthenticationService
   呼叫JWTUtility的工具，製造、驗證、處理token



需求功能對照

建構使用者登入系統

1. 使用Hibernate建構前端與後端的連結 ✓
2. 使用Spring Security做系統登入時的資源權限管理 ✓
3. 使用Json Web Token做使用者權限的驗證 ✓✗

解決方案 1 ──Token自動展延

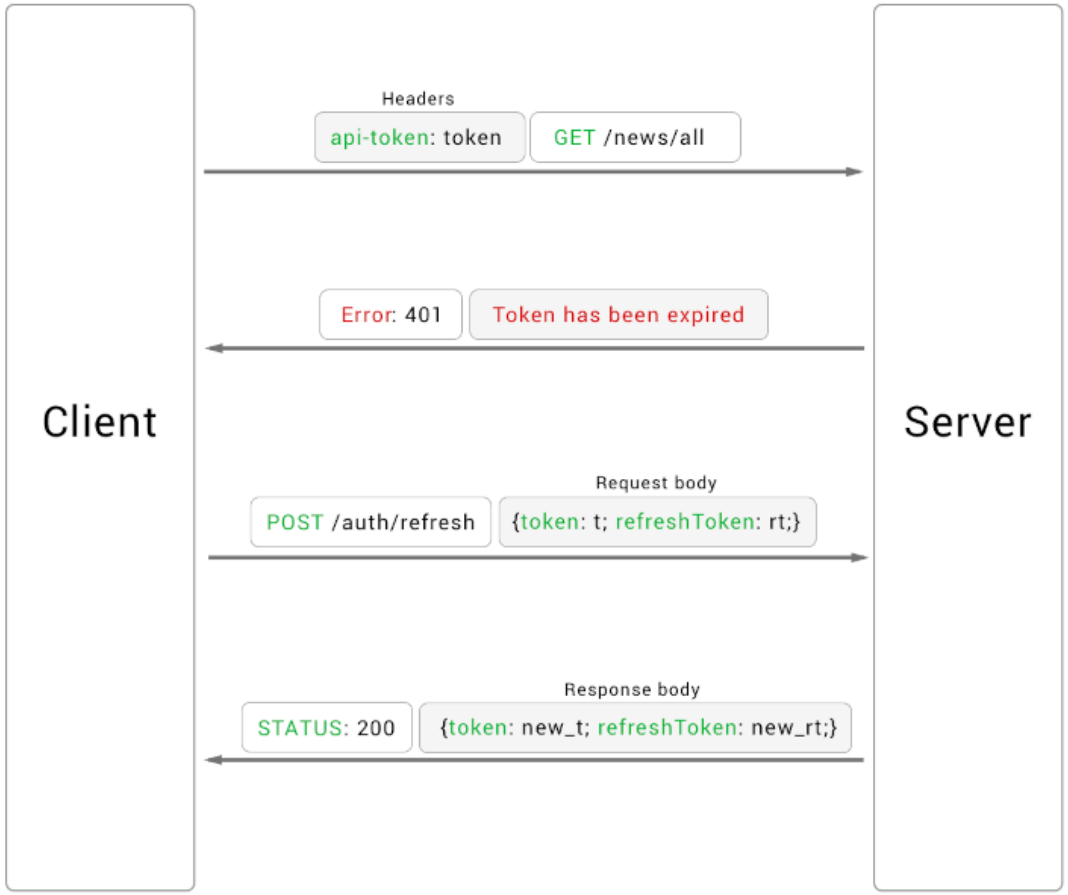
- 問題：

token到期後，使用者需要重新登入，無法自動為token的效期展延

- 解決方案1：

登入時給予兩組token (Access JWT token Refresh JWT token)。

當Access token過期，用Refresh token作驗證並產生新的兩組token。



解決方案 2 ──Token自動展延

- 解決方案 2：

在controller新增一個endpoint在token過期前展延其效期

- 例如: 設一個使用者最低必須活動時間30分鐘，用ifelse判斷如果使用者在最後30分鐘內有發送任何的request就把他丟到token展延的endpoint

Q&A