後端API 接口交付

如何使用


```
<html>
  ▶ <head> ⋯ </head>
  ▼ <body> == $0
      <a href="views/public.html">Click Me</a>
      <!-- <img src="https://firebasestorage.googleapis.com/v0/b/project-
      20240429.appspot.com/o/1715219412279-6.png?alt=media&token=92074505-e5c8-4e3d-9751-
      5087b2427b45" alt=""> -->
      <input type="file" id="photo">
      <br>
      <img src id="image" alt>
      <button id="upload" onclick="upload()">Upload Image</button>
      <script src="https://www.gstatic.com/firebasejs/7.7.0/firebase-app.js"></script>
      <script src="https://www.gstatic.com/firebasejs/7.7.0/firebase-storage.js"></script>
      <script src="controller/upload.js"> </script>
  </body>
</html>
```

- Code : https://github.com/AdventrousAstronaut/web-app

- 引入資料庫資源

- <script src="https://www.gstatic.com/firebasejs/5.0.0/firebase-app.js"></script>

-   <script src="https://www.gstatic.com/firebasejs/5.0.0/firebase-database.js"></script>

- 引入 javascript

- <script src = "/public/controller/insert.js"></script>

有關故事 的物件

- <mark>交付</mark>:
- formatStories(soundList, textList, translationTextList)
- formatItem(title, language，imageURL，stories，id)
- 填入項目生成物件
- Item 的產生詳看 =>

```
/ Item
  / title 標題
  / language 語種
  / image 圖片
  / signature 生成的用戶ID
  / stories [list of Paragragh Object Story]
```

```
// Example usage:
const soundList = ["song1.mp4", "song2.mp4"];
const textList = ["Text 1", "Text 2"];
const translationtextList = ["Text 1", "Text 2"];
const stories = formatStories(soundList, textList,
translationtextList);
const imageURL = "../static/image/1.png";
const ID = "id"
const item = formatItem("Title", "Language", imageURL,
stories, ID);
const token = "-NxL96fO7UBm_LN8vDws";
```

物件的操作

- publishStories(token, item)
- updateStories(id, token, item)
- newStories(id, item)
- getPublishedStories()
- getMyStories(id)

- 使用情境:
- NewStories()
- 用戶生成故事 => "務必" 將新故事的id 以存於 前端 這樣才可能實現獲取此時 觀看的故事的id (可用 hidden 存)
- updateStories(id, token, item)
- 用戶想要更新過去的故事
- getMyStories(id)
- 獲取用戶的故事 render在頁面上
- getPublishedStories()
- 獲取公開的故事
- publishStories(token, item)
- 發布故事

回傳的內容 (透過 . Access)

類別
Class: Paragragh < Story < Item | user

```
/ Item
  / title 標題
  / language 語種
  / image 圖片
  / signature 生成的用戶ID
    / stories [list of Paragragh Object Story]

/ Paragragh
  / soundtrack 音檔
  / text 文字
  / translation 翻譯
```

```javascript
getMyStories("id")
.then(stories => {
        console.log(stories);
})
```

insert.js:122

{-NxL96fO7UBm_LN8vDws: {…}, -NxLHr_qzd_a4_hZsRXe: {…}, -NxLHuzn1K7DCgXtRMWK: {…}, -NxLHv
4IEaV_xqPLB-72: {…}, -NxLIKl_pWJZOVGY8iCs: {…}, …}
  ▶ -NxL96fO7UBm_LN8vDws: {image: '../static/image/1.png', language: 'Language', signature
  ▶ -NxLHr_qzd_a4_hZsRXe: {image: '../static/image/1.png', language: 'Language', signature
  ▶ -NxLHuzn1K7DCgXtRMWK: {image: '../static/image/1.png', language: 'Language', signature
  ▶ -NxLHv4IEaV_xqPLB-72: {image: '../static/image/1.png', language: 'Language', signature
  ▶ -NxLIKl_pWJZOVGY8iCs: {image: '../static/image/1.png', language: 'Language', signature
  ▶ -NxLIKldxaa9i94JslVT: {image: '../static/image/1.png', language: 'Language', signature
  ▶ -NxLINbyq8sm-u1y_RVz: {image: '../static/image/1.png', language: 'Language', signature
  ▶ -NxLIO0pKdGzOGLcwr8G: {image: '../static/image/1.png', language: 'Language', signature
  ▶ -NxLIOGaLN3Yq4puQLOl: {image: '../static/image/1.png', language: 'Language', signature
  ▶ -NxLIOGk8490tnxXlm07: {image: '../static/image/1.png', language: 'Language', signature
  ▶ [[Prototype]]: Object

用戶資料 => userStatus.js

- <mark>交付:</mark>
- getUser(email) return Object user
- isValid(email, loginPwd) return boolean
- register(email, pwd) return Promise
- How to use promise type =>
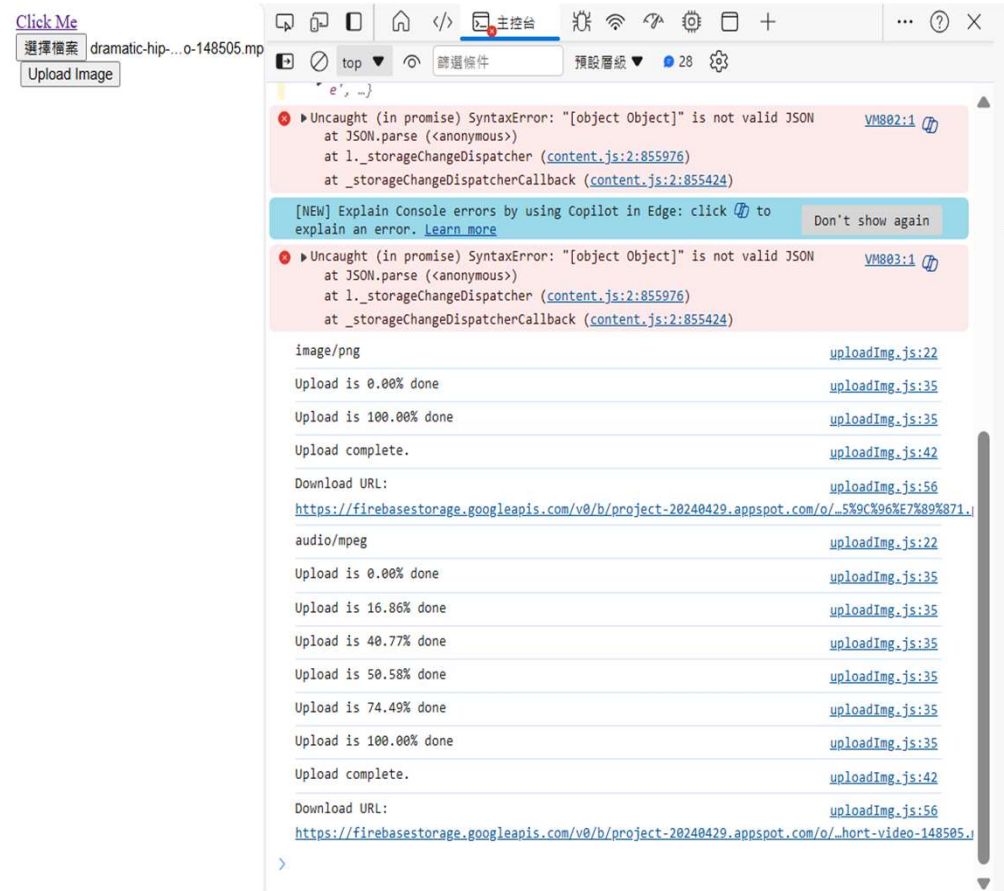
```
class User {
    constructor(email, pwd) {
        this.username = "Anon",
        this.email = email;
        this.pwd = pwd;
        this.userID = Date.now().toString();
        this.avatarURL = "/public/assets/image/
        defaultAvatar.jpeg",
        this.intro = "I have something to say...";
    }
}
```

```
register(mail, password)
.then(() => {
    console.log("User registered successfully!");
});
```

交付: 關於 Audio 與 Image => upload.js

- 交付 upload(file)
- 填入 file 就可以獲得 url 訪問資源
- 用途: 請按照資源 填入 img 等項目資訊

有關 Image 與 audio 的 使用

- 不需要從資料庫撈 <mark>可以直接透過 url 獲取資源</mark>
- 例子 目前可以去 https://project-20240429.web.app/ 測試
- https://firebasestorage.googleapis.com/v0/b/project-20240429.appspot.com/o/images%2F1715224662363-%E5%9C%96%E7%89%871.png?alt=media&token=85f9f646-efa5-479b-97bd-513fd1d28b72
- https://firebasestorage.googleapis.com/v0/b/project-20240429.appspot.com/o/audio%2F1715224683561-dramatic-hip-hop-music-background-jazz-music-for-short-video-148505.mp3?alt=media&token=d60a4312-3dac-4939-ae0e-2eb0e568dc4b

https://project-20240429.web.app/
測試解釋

- 上傳圖片 (可能需要一點時間)
- 按右鍵 > 檢查 > console
- URL 即為圖片的 url 可以在任何地方打開 以url 方式存儲在 Item 中
- Audio 與 Image 態同樣適用
- 測試遠端打開: