

1150722 請AI教我使用Firebase 研習講義：

<https://kingvincentwang.github.io/gses/Firebase/firebase0722.html>

*****不花一毛錢把Firebase學會*****

node.js下載點

<https://nodejs.org/zh-tw/download>

<https://notepad-plus-plus.org/downloads/v8.9.6.2/>

請幫我實作一個電子郵件認證功能, 使用html+Firebase,註冊即登入

請幫我修改為使用電子郵件註冊後須審核通過後才能登入

請幫我修改為使用google 註冊但需審核通過才能登入

`python -m http.server 8000`

<http://localhost:8000/>

請幫我將這個網頁部署至hosting

`Set-ExecutionPolicy RemoteSigned -Scope CurrentUse`

`npm install -g firebase-tools`

`firebase login`

`firebase init hosting`

```
PS C:\test> firebase init hosting

#####  ###  #####  #####  #####  ###  #####  #####
##      ##  ##  ##  ##  ##  ##  ##  ##  ##
#####  ##  #####  #####  #####  #####  #####  #####
##      ##  ##  ##  ##  ##  ##  ##  ##  ##
##      #####  ##  #####  #####  ##  ##  #####  #####

You're about to initialize a Firebase project in this directory:

  C:\test

Before we get started, keep in mind:

  * You are currently outside your home directory

✔Are you ready to proceed? Yes

=== Project Setup

First, let's associate this project directory with a Firebase project.
You can create multiple project aliases by running firebase use --add,

✔Please select an option: Use an existing project
✔Select a default Firebase project for this directory: lucky-58ec5 (lucky)

=== Hosting Setup

Your public directory is the folder (relative to your project directory) that
will contain Hosting assets to be uploaded with firebase deploy. If you
have a build process for your assets, use your build's output directory.

✔What do you want to use as your public directory? public
✔Configure as a single-page app (rewrite all urls to /index.html)? No
✔Set up automatic builds and deploys with GitHub? No
* Wrote public\404.html
* Wrote public\index.html

=== Agent Skills Setup

If you are using an AI coding agent, Firebase Agent Skills make it an expert at Firebase.
✔Would you like to install agent skills for Firebase? No

* Wrote configuration info to firebase.json
* Wrote project information to .firebaserc
* Wrote .gitignore

* Firebase initialization complete!
PS C:\test>
```

firebase deploy --only hosting

```
<!DOCTYPE html>
<html lang="zh-TW">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <title>Firebase 身分驗證與帳號審核展示</title>
  <script src="https://cdn.tailwindcss.com"></script>
  <link
href="https://fonts.googleapis.com/css2?family=Inter:wght@300;40
0;500;600;700&display=swap" rel="stylesheet">
  <style>
    body {
      font-family: 'Inter', sans-serif;
    }
  </style>
</head>
<body>
```

```

</style>
</head>
<body class="bg-slate-50 min-h-screen flex items-center
justify-center p-4">

  <div class="w-full max-w-md bg-white rounded-2xl shadow-xl
overflow-hidden border border-slate-100 transition-all
duration-300">

    <!-- 頂部標題列 -->
    <div class="bg-gradient-to-r from-indigo-600 to-violet-600 p-6
text-white text-center">
      <div class="inline-flex items-center justify-center w-12 h-12
bg-white/10 rounded-xl mb-3 backdrop-blur-sm">
        <svg class="w-6 h-6 text-white" fill="none"
stroke="currentColor" viewBox="0 0 24 24">
          <path stroke-linecap="round" stroke-linejoin="round"
stroke-width="2" d="M12 15v2m-6 4h12a2 2 0 02-2v-6a2 2 0
00-2-2H6a2 2 0 00-2 2v6a2 2 0 002 2zm10-10V7a4 4 0 00-8 0v4h8z"/>
        </svg>
      </div>
      <h1 class="text-xl font-bold tracking-tight"
id="header-title">會員驗證中心</h1>
      <p class="text-xs text-indigo-100 mt-1"
id="header-subtitle">需要管理員審核通過後方可登入</p>
    </div>

    <div class="p-6">
      <!-- 載入中動畫畫面 -->
      <div id="view-loading" class="flex flex-col items-center
justify-center py-10 space-y-3">
        <div class="w-8 h-8 border-4 border-indigo-600
border-t-transparent rounded-full animate-spin"></div>
        <p class="text-sm text-slate-500 font-medium">正在檢查登
入與審核狀態...</p>
      </div>

      <!-- 自訂錯誤訊息顯示區 (取代 alert) -->

```

```
<div id="error-message" class="hidden mb-4 p-3.5
bg-rose-50 border-l-4 border-rose-500 text-rose-700 text-sm
rounded-r-lg transition-all">
```

錯誤發生。

```
</div>
```

```
<!-- 自訂成功訊息顯示區 -->
```

```
<div id="success-message" class="hidden mb-4 p-3.5
bg-emerald-50 border-l-4 border-emerald-500 text-emerald-700
text-sm rounded-r-lg transition-all">
```

註冊成功！請等待管理員審核。

```
</div>
```

```
<!-- ===== 登入與註冊表單區塊
===== -->
```

```
<div id="view-auth" class="hidden">
```

```
<form id="auth-form" class="space-y-4">
```

```
<div>
```

```
<label class="block text-xs font-semibold
text-slate-600 mb-1">電子郵件 (Email)</label>
```

```
<input type="email" id="email" required
placeholder="user@example.com"
class="w-full px-3.5 py-2.5 bg-slate-50 border
border-slate-200 rounded-lg text-sm text-slate-800
focus:outline-none focus:ring-2 focus:ring-indigo-500
focus:bg-white transition-all">
```

```
</div>
```

```
<div>
```

```
<label class="block text-xs font-semibold
text-slate-600 mb-1">密碼 (Password)</label>
```

```
<input type="password" id="password" required
placeholder="至少 6 位數密碼" minlength="6"
class="w-full px-3.5 py-2.5 bg-slate-50 border
border-slate-200 rounded-lg text-sm text-slate-800
focus:outline-none focus:ring-2 focus:ring-indigo-500
focus:bg-white transition-all">
```

```
</div>
```

```
<button type="submit" id="submit-btn"
```

```
class="w-full mt-2 bg-indigo-600 hover:bg-indigo-700
active:bg-indigo-800 text-white font-medium py-2.5 px-4 rounded-lg
shadow-md hover:shadow-indigo-200 transition-all duration-200
flex items-center justify-center space-x-2">
```

```
<span id="btn-text">登入系統</span>
```

```
<div id="btn-spinner" class="hidden w-4 h-4 border-2
border-white border-t-transparent rounded-full
animate-spin"></div>
```

```
</button>
```

```
</form>
```

```
<div class="mt-6 text-center text-sm text-slate-600">
```

```
<span id="toggle-prompt">還沒有帳號嗎？</span>
```

```
<button id="toggle-mode-btn" type="button"
class="text-indigo-600 font-semibold hover:text-indigo-800
hover:underline transition-colors focus:outline-none ml-1">
```

```
立即註冊
```

```
</button>
```

```
</div>
```

```
<!-- 測試用：強制核准按鈕 (供開發測試使用) -->
```

```
<div class="mt-5 pt-4 border-t border-slate-100
text-center">
```

```
<button id="test-approve-btn" type="button"
class="hidden text-xs text-emerald-700 bg-emerald-50 border
border-emerald-200 hover:bg-emerald-100 px-3 py-1.5 rounded-md
font-medium transition-colors">
```

```
 開發者測試：一鍵核准此帳號
```

```
</button>
```

```
</div>
```

```
</div>
```

```
<!-- ===== 登入後的儀表板
===== -->
```

```
<div id="view-dashboard" class="hidden space-y-4">
```

```
<div class="p-4 bg-emerald-50 border border-emerald-100
rounded-xl text-center">
```

```
<div class="inline-flex items-center justify-center w-8
h-8 bg-emerald-100 rounded-full mb-1">
```

```
      <svg class="w-5 h-5 text-emerald-600" fill="none"
stroke="currentColor" viewBox="0 0 24 24">
```

```
        <path stroke-linecap="round"
stroke-linejoin="round" stroke-width="2" d="M5 13l4 4L19 7"/>
```

```
      </svg>
```

```
    </div>
```

```
    <p class="text-emerald-800 font-semibold text-sm">身分驗證通過 & 審核成功</p>
```

```
    <p class="text-xs text-emerald-600 mt-0.5">您已被管理員開通使用權限。</p>
```

```
  </div>
```

```
<div class="space-y-2 bg-slate-50 p-3.5 rounded-lg border
border-slate-100 text-xs text-slate-600">
```

```
  <div>
```

```
    <span class="font-medium text-slate-400 block">帳號
Email:</span>
```

```
    <span id="user-email-display" class="font-mono
text-slate-800 break-all"></span>
```

```
  </div>
```

```
  <div class="pt-2 border-t border-slate-200/60">
```

```
    <span class="font-medium text-slate-400
block">Firebase UID:</span>
```

```
    <span id="user-uid-display" class="font-mono
text-slate-500 text-[10px] break-all"></span>
```

```
  </div>
```

```
</div>
```

```
  <button id="logout-btn" type="button"
class="w-full bg-slate-100 hover:bg-slate-200
text-slate-700 font-medium py-2.5 px-4 rounded-lg transition-colors
duration-200 flex items-center justify-center space-x-1 text-sm">
```

```
    <span>安全登出</span>
```

```
  </button>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<script type="module">
```

```

import { initializeApp } from
"https://www.gstatic.com/firebasejs/11.6.1/firebase-app.js";
import {
  getAuth,
  signInWithEmailAndPassword,
  createUserWithEmailAndPassword,
  signOut,
  onAuthStateChanged
} from
"https://www.gstatic.com/firebasejs/11.6.1/firebase-auth.js";
import {
  getFirestore,
  doc,
  setDoc,
  getDoc,
  updateDoc
} from
"https://www.gstatic.com/firebasejs/11.6.1/firebase-firestore.js";

// ⚠ 請填入您的 Firebase 專案真實設定檔 (從專案設定 > 網頁應
用程式中取得)
const firebaseConfig = {
  apiKey: "AlzaSyAUSdutLSF619_2KhIs1sIN_KIkvhWHVQY",
  authDomain: "lucky-58ec5.firebaseio.com",
  projectId: "lucky-58ec5",
  storageBucket: "lucky-58ec5.firebaseio.com",
  messagingSenderId: "813489087951",
  appId: "1:813489087951:web:b75704828402e17e9a1eba"
};

const app = initializeApp(firebaseConfig);
const auth = getAuth(app);
const db = getFirestore(app);

// 取得環境 APP ID 確保資料隔離
const environmentAppId = typeof __app_id !== 'undefined' ?
__app_id : 'demo-auth-app';

const views = {
  loading: document.getElementById('view-loading'),

```

```

    auth: document.getElementById('view-auth'),
    dashboard: document.getElementById('view-dashboard')
  };

  const headerTitle = document.getElementById('header-title');
  const headerSubtitle =
document.getElementById('header-subtitle');
  const authForm = document.getElementById('auth-form');
  const emailInput = document.getElementById('email');
  const passwordInput = document.getElementById('password');
  const submitBtn = document.getElementById('submit-btn');
  const btnText = submitBtn.querySelector('#btn-text');
  const btnSpinner = document.getElementById('btn-spinner');
  const togglePrompt =
document.getElementById('toggle-prompt');
  const toggleBtn =
document.getElementById('toggle-mode-btn');
  const errorMsg = document.getElementById('error-message');
  const successMsg =
document.getElementById('success-message');
  const testApproveBtn =
document.getElementById('test-approve-btn');
  const userEmailDisplay =
document.getElementById('user-email-display');
  const userUidDisplay =
document.getElementById('user-uid-display');
  const logoutBtn = document.getElementById('logout-btn');

  let isLoginMode = true;

  const errorTranslations = {
    'auth/invalid-email': '電子郵件格式不正確。',
    'auth/user-not-found': '找不到此帳號，請檢查信箱或先進行註
冊。',
    'auth/wrong-password': '密碼錯誤，請重新輸入。',
    'auth/email-already-in-use': '此電子郵件已被註冊過。',
    'auth/weak-password': '密碼強度不足，請使用至少 6 個字元。',
    'auth/invalid-credential': '電子郵件或密碼不正確。',
    'auth/configuration-not-found': '請先在 Firestore 後台啟用「
Authentication > 電子郵件/密碼」登入方式。'
  };

```



```

};

function showView(viewName) {
    Object.keys(views).forEach(key =>
views[key].classList.add('hidden'));
    views[viewName].classList.remove('hidden');
}

function showError(message) {
    const translatedMessage = errorTranslations[message] || `發
生錯誤 (${message})`;
    errorMsg.textContent = translatedMessage;
    errorMsg.classList.remove('hidden');
    successMsg.classList.add('hidden');
}

function hideError() {
    errorMsg.classList.add('hidden');
}

function showSuccess(message) {
    successMsg.textContent = message;
    successMsg.classList.remove('hidden');
    errorMsg.classList.add('hidden');
}

function hideSuccess() {
    successMsg.classList.add('hidden');
}

function setButtonLoading(isLoading) {
    submitBtn.disabled = isLoading;
    if (isLoading) {
        btnSpinner.classList.remove('hidden');
        btnText.textContent = '處理中...';
        submitBtn.classList.add('opacity-80',
'cursor-not-allowed');
    } else {
        btnSpinner.classList.add('hidden');
    }
}

```

```

        btnText.textContent = isLoginMode ? '登入系統' : '註冊帳號
';
        submitBtn.classList.remove('opacity-80',
'cursor-not-allowed');
    }
}

function toggleAuthMode() {
    isLoginMode = !isLoginMode;
    hideError();
    hideSuccess();
    testApproveBtn.classList.add('hidden');

    if (isLoginMode) {
        headerTitle.textContent = '會員驗證中心';
        headerSubtitle.textContent = '需要管理員審核通過後方可登
入';

        btnText.textContent = '登入系統';
        togglePrompt.textContent = '還沒有帳號嗎?';
        toggleBtn.textContent = '立即註冊';
    } else {
        headerTitle.textContent = '建立新帳號';
        headerSubtitle.textContent = '註冊後將進入管理員審核流程
';

        btnText.textContent = '註冊帳號';
        togglePrompt.textContent = '已有帳號?';
        toggleBtn.textContent = '返回登入';
    }
}

toggleBtn.addEventListener('click', toggleAuthMode);

// 監聽登入狀態改變
onAuthStateChanged(auth, async (user) => {
    if (user) {
        showView('loading');
        try {
            // 檢查 Firestore 中的帳號審核狀態
            const statusRef = doc(db, 'artifacts', environmentAppId,
'users', user.uid, 'account', 'status');

```

```

const statusSnap = await getDoc(statusRef);

if (statusSnap.exists() && statusSnap.data().approved
=== true) {
    // 【審核通過】允許登入
    userEmailDisplay.textContent = user.email;
    userUidDisplay.textContent = user.uid;
    showView('dashboard');
    authForm.reset();
} else {
    // 【未審核或拒絕】強制登出
    await signOut(auth);
    showView('auth');
    showError("您的帳號尚未審核通過，請耐心等待管理員
核准。");

    // 顯示測試用核准按鈕
    testApproveBtn.dataset.uid = user.uid;
    testApproveBtn.classList.remove('hidden');
}
} catch (error) {
    console.error("檢查審核狀態錯誤:", error);
    await signOut(auth);
    showView('auth');
    showError("無法驗證審核狀態，請確認已啟用 Firestore 並
且設為測試模式。");
}
} else {
    showView('auth');
}
});

authForm.addEventListener('submit', async (e) => {
    e.preventDefault();
    hideError();
    hideSuccess();
    testApproveBtn.classList.add('hidden');
    setButtonLoading(true);

    const email = emailInput.value.trim();

```

```

const password = passwordInput.value;

try {
  if (isLoginMode) {
    // 執行登入 (onAuthStateChanged 會接手檢查審核狀態)
    await signInWithEmailAndPassword(auth, email,
password);
  } else {
    // 執行註冊
    const userCredential = await
createUserWithEmailAndPassword(auth, email, password);
    const user = userCredential.user;

    // 註冊後於 Firestore 記錄未審核狀態
    const statusRef = doc(db, 'artifacts', environmentAppld,
'users', user.uid, 'account', 'status');
    await setDoc(statusRef, {
      approved: false,
      email: email,
      createdAt: new Date().toISOString()
    });

    // 註冊完立刻強制登出
    await signOut(auth);

    // 切換回登入頁, 提醒等待審核
    isLoginMode = true;
    toggleAuthMode();
    showSuccess("註冊成功！帳號目前為「審核中」, 請等候管
理員核准後再登入。");
  }
} catch (error) {
  console.error("Auth Error:", error.code, error.message);
  showError(error.code);
} finally {
  setButtonLoading(false);
}
});

// 測試用「一鍵核准」功能

```

```

testApproveBtn.addEventListener('click', async () => {
  const uid = testApproveBtn.dataset.uid;
  if (!uid) return;

  try {
    testApproveBtn.textContent = '核准中...';
    const statusRef = doc(db, 'artifacts', environmentAppId,
'users', uid, 'account', 'status');
    await updateDoc(statusRef, { approved: true });

    hideError();
    showSuccess("測試成功：帳號已強制核准！您現在可以點擊
登入。");
    testApproveBtn.classList.add('hidden');
  } catch (error) {
    console.error("更新審核狀態失敗:", error);
    showError("核准失敗，請檢查 Firestore 是否已啟用並設為測
試模式。");
  } finally {
    testApproveBtn.textContent = '🔧 開發者測試：一鍵核准此
帳號';
  }
});

// 登出功能
logoutBtn.addEventListener('click', async () => {
  try {
    await signOut(auth);
  } catch (error) {
    showError("登出失敗，請稍後重試。");
  }
});
</script>
</body>
</html>

```

```

<!DOCTYPE html>
<html lang="zh-TW">

```

```

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <title>Firebase 身分驗證與帳號審核展示</title>
  <script src="https://cdn.tailwindcss.com"></script>
  <link
href="https://fonts.googleapis.com/css2?family=Inter:wght@300;40
0;500;600;700&display=swap" rel="stylesheet">
  <style>
    body {
      font-family: 'Inter', sans-serif;
    }
  </style>
</head>
<body class="bg-slate-50 min-h-screen flex items-center
justify-center p-4">

  <div class="w-full max-w-md bg-white rounded-2xl shadow-xl
overflow-hidden border border-slate-100 transition-all
duration-300">

    <!-- 頂部標題列 -->
    <div class="bg-gradient-to-r from-indigo-600 to-violet-600 p-6
text-white text-center">
      <div class="inline-flex items-center justify-center w-12 h-12
bg-white/10 rounded-xl mb-3 backdrop-blur-sm">
        <svg class="w-6 h-6 text-white" fill="currentColor"
viewBox="0 0 24 24">
          <path
d="M12.545,10.239v3.821h5.445c-0.712,2.315-2.647,3.972-5.445,3.97
2c-3.332,0-6.033-2.701-6.033-6.032s2.701-6.032,6.033-6.032c1.498,0,
2.866,0.549,3.921,1.453l2.814-2.814C17.503,2.988,15.139,2,12.545,2C
7.021,2,2.545,6.477,2.545,12s4.476,10,10,10c5.772,0,9.613-4.058,9.61
3-9.775c0-0.672-0.063-1.314-0.178-1.986H12.545z"/>
        </svg>
      </div>
      <h1 class="text-xl font-bold tracking-tight"
id="header-title">Google 身分驗證中心</h1>

```

```
<p class="text-xs text-indigo-100 mt-1"
id="header-subtitle">使用 Google 帳號登入，經管理員審核後即可使用
</p>
</div>
```

```
<div class="p-6">
  <!-- 載入中動畫畫面 -->
  <div id="view-loading" class="flex flex-col items-center
justify-center py-10 space-y-3">
    <div class="w-8 h-8 border-4 border-indigo-600
border-t-transparent rounded-full animate-spin"></div>
    <p class="text-sm text-slate-500 font-medium">正在檢查登
入與審核狀態...</p>
  </div>
```

```
<!-- 自訂錯誤訊息顯示區 -->
<div id="error-message" class="hidden mb-4 p-3.5
bg-rose-50 border-l-4 border-rose-500 text-rose-700 text-sm
rounded-r-lg transition-all">
  錯誤發生。
</div>
```

```
<!-- 自訂成功訊息顯示區 -->
<div id="success-message" class="hidden mb-4 p-3.5
bg-emerald-50 border-l-4 border-emerald-500 text-emerald-700
text-sm rounded-r-lg transition-all">
  驗證成功！
</div>
```

```
<!-- ===== Google 登入區塊
===== -->
<div id="view-auth" class="hidden space-y-4">
  <p class="text-xs text-slate-500 text-center mb-2">點擊下
方按鈕進行 Google 帳號驗證/註冊：</p>
```

```
<button id="google-login-btn" type="button"
class="w-full bg-white hover:bg-slate-50
active:bg-slate-100 text-slate-700 font-semibold py-3 px-4 border
border-slate-300 rounded-xl shadow-sm transition-all duration-200
flex items-center justify-center space-x-3">
```

```

<svg class="w-5 h-5" viewBox="0 0 24 24">
  <path fill="#4285F4" d="M22.56
12.25c0-.78-.07-1.53-.2-2.25H12v4.26h5.92c-.26 1.37-1.04 2.53-2.21
3.31v2.77h3.57c2.08-1.92 3.28-4.74 3.28-8.09z"/>
  <path fill="#34A853" d="M12 23c2.97 0 5.46-.98
7.28-2.66l-3.57-2.77c-.98.66-2.23 1.06-3.71 1.06-2.86
0-5.29-1.93-6.16-4.53H2.18v2.84C3.99 20.53 7.7 23 12 23z"/>
  <path fill="#FBBC05" d="M5.84
14.09c-.22-.66-.35-1.36-.35-2.09s.13-1.43.35-2.09V7.06H2.18C1.43
8.55 1 10.22 1 12s.43 3.45 1.18 4.94l2.85-2.22.81-.63z"/>
  <path fill="#EA4335" d="M12 5.38c1.62 0 3.06.56 4.21
1.64l3.15-3.15C17.45 2.09 14.97 1 12 1 7.7 1 3.99 3.47 2.18 7.06l3.66
2.84c.87-2.6 3.3-4.52 6.16-4.52z"/>

```

```
</svg>
```

```
<span>使用 Google 帳號繼續</span>
```

```
</button>
```

```
<!-- 測試用:強制核准按鈕 (供開發測試使用) -->
```

```
<div class="mt-5 pt-4 border-t border-slate-100
text-center">
```

```

  <button id="test-approve-btn" type="button"
class="hidden text-xs text-emerald-700 bg-emerald-50 border
border-emerald-200 hover:bg-emerald-100 px-3 py-1.5 rounded-md
font-medium transition-colors">

```

```

  <img alt="Wrench icon" data-bbox="272 608 305 630"/> 開發者測試:一鍵核准此帳號

```

```
</button>
```

```
</div>
```

```
</div>
```

```

<!-- ===== 登入後的儀表板
===== -->

```

```

  <div id="view-dashboard" class="hidden space-y-4">
    <div class="p-4 bg-emerald-50 border border-emerald-100
rounded-xl text-center">

```

```

      <div class="inline-flex items-center justify-center w-8
h-8 bg-emerald-100 rounded-full mb-1">

```

```

        <svg class="w-5 h-5 text-emerald-600" fill="none"
stroke="currentColor" viewBox="0 0 24 24">

```

```

          <path stroke-linecap="round"
stroke-linejoin="round" stroke-width="2" d="M5 13l4 4L19 7"/>

```



```
        </svg>
      </div>
      <p class="text-emerald-800 font-semibold text-sm">身分驗證通過 & 審核成功</p>
      <p class="text-xs text-emerald-600 mt-0.5">您已被管理員開通使用權限。</p>
    </div>

    <div class="flex items-center space-x-3 p-3 bg-slate-50 border border-slate-100 rounded-xl">
      <img id="user-avatar" src="" alt="Avatar" class="w-12 h-12 rounded-full border border-slate-200 object-cover">
      <div class="overflow-hidden">
        <h2 id="user-name-display" class="font-bold text-slate-800 text-sm truncate"></h2>
        <p id="user-email-display" class="text-xs text-slate-500 font-mono truncate"></p>
      </div>
    </div>

    <div class="space-y-1 bg-slate-50 p-3 rounded-lg border border-slate-100 text-xs text-slate-600">
      <span class="font-medium text-slate-400 block">Firebase UID:</span>
      <span id="user-uid-display" class="font-mono text-slate-500 text-[10px] break-all"></span>
    </div>

    <button id="logout-btn" type="button"
      class="w-full bg-slate-100 hover:bg-slate-200 text-slate-700 font-medium py-2.5 px-4 rounded-lg transition-colors duration-200 flex items-center justify-center space-x-1 text-sm">
      <span>安全登出</span>
    </button>
  </div>
</div>

<script type="module">
```

```

import { initializeApp } from
"https://www.gstatic.com/firebasejs/11.6.1/firebase-app.js";
import {
  getAuth,
  GoogleAuthProvider,
  signInWithPopup,
  signOut,
  onAuthStateChanged
} from
"https://www.gstatic.com/firebasejs/11.6.1/firebase-auth.js";
import {
  getFirestore,
  doc,
  setDoc,
  getDoc,
  updateDoc
} from
"https://www.gstatic.com/firebasejs/11.6.1/firebase-firestore.js";

// ⚠ 請填入您的 Firebase 專案真實設定檔
const firebaseConfig = {
  apiKey: "AIzaSyAUSdutLSF619_2KhIs1sIN_KIkvhWHVQY",
  authDomain: "lucky-58ec5.firebaseio.com",
  projectId: "lucky-58ec5",
  storageBucket: "lucky-58ec5.firebaseio.com",
  messagingSenderId: "813489087951",
  appId: "1:813489087951:web:b75704828402e17e9a1eba"
};

const app = initializeApp(firebaseConfig);
const auth = getAuth(app);
const db = getFirestore(app);
const googleProvider = new GoogleAuthProvider();

const environmentAppId = typeof __app_id !== 'undefined' ?
__app_id : 'demo-auth-app';

const views = {
  loading: document.getElementById('view-loading'),
  auth: document.getElementById('view-auth'),

```

```

    dashboard: document.getElementById('view-dashboard')
  };

  const googleLoginBtn =
document.getElementById('google-login-btn');
  const errorMsg = document.getElementById('error-message');
  const successMsg =
document.getElementById('success-message');
  const testApproveBtn =
document.getElementById('test-approve-btn');
  const userAvatar = document.getElementById('user-avatar');
  const userNameDisplay =
document.getElementById('user-name-display');
  const userEmailDisplay =
document.getElementById('user-email-display');
  const userUidDisplay =
document.getElementById('user-uid-display');
  const logoutBtn = document.getElementById('logout-btn');

  const errorTranslations = {
    'auth/popup-closed-by-user': '已取消 Google 登入視窗。',
    'auth/cancelled-popup-request': '登入請求已被取消。',
    'auth/operation-not-allowed': '請在 Firebase 後台啟用 Google
登入方式。',
    'auth/configuration-not-found': '請在 Firebase 控制台啟用「
Authentication > Sign-in method > Google」.'
  };

  function showView(viewName) {
    Object.keys(views).forEach(key =>
views[key].classList.add('hidden'));
    views[viewName].classList.remove('hidden');
  }

  function showError(message) {
    const translatedMessage = errorTranslations[message] || `發
生錯誤 (${message})`;
    errorMsg.textContent = translatedMessage;
    errorMsg.classList.remove('hidden');
    successMsg.classList.add('hidden');
  }

```

```

}

function hideError() {
  errorMsg.classList.add('hidden');
}

function showSuccess(message) {
  successMsg.textContent = message;
  successMsg.classList.remove('hidden');
  errorMsg.classList.add('hidden');
}

function hideSuccess() {
  successMsg.classList.add('hidden');
}

googleLoginBtn.addEventListener('click', async () => {
  hideError();
  hideSuccess();
  try {
    await signInWithPopup(auth, googleProvider);
  } catch (error) {
    console.error("Google 登入失敗:", error);
    showError(error.code);
  }
});

onAuthStateChanged(auth, async (user) => {
  if (user) {
    showView('loading');
    try {
      const statusRef = doc(db, 'artifacts', environmentAppld,
'users', user.uid, 'account', 'status');
      let statusSnap = await getDoc(statusRef);

      // 首次登入者: 自動在 Firestore 建立待審核紀錄
      if (!statusSnap.exists()) {
        await setDoc(statusRef, {
          approved: false,
          email: user.email,

```

```

        displayName: user.displayName || '未提供名稱',
        photoURL: user.photoURL || '',
        createdAt: new Date().toISOString()
    });
    statusSnap = await getDoc(statusRef);
}

if (statusSnap.exists() && statusSnap.data().approved
=== true) {
    // 【審核通過】顯示儀表板
    userAvatar.src = user.photoURL ||
'https://placeholder.co/100x100/indigo/white?text=User';
    userNameDisplay.textContent = user.displayName ||
'Google 使用者';
    userEmailDisplay.textContent = user.email;
    userIdDisplay.textContent = user.uid;
    showView('dashboard');
} else {
    // 【未審核】強制登出，並提醒使用者
    await signOut(auth);
    showView('auth');
    showError("您的 Google 帳號尚未審核通過，已提出申
請，請耐心等待管理員開通。");

    // 顯示測試核准按鈕
    testApproveBtn.dataset.uid = user.uid;
    testApproveBtn.classList.remove('hidden');
}
} catch (error) {
    console.error("檢查審核狀態錯誤:", error);
    await signOut(auth);
    showView('auth');
    showError("無法驗證審核狀態，請確認已啟用 Firestore 並
且設為測試模式。");
}
} else {
    showView('auth');
}
});

```

```

// 測試用「一鍵核准」功能
testApproveBtn.addEventListener('click', async () => {
  const uid = testApproveBtn.dataset.uid;
  if (!uid) return;

  try {
    testApproveBtn.textContent = '核准中...';
    const statusRef = doc(db, 'artifacts', environmentAppld,
'users', uid, 'account', 'status');
    await updateDoc(statusRef, { approved: true });

    hideError();
    showSuccess("測試成功：帳號已強制核准！請再次點擊「使用
Google 帳號繼續」。");
    testApproveBtn.classList.add('hidden');
  } catch (error) {
    console.error("更新審核狀態失敗:", error);
    showError("核准失敗，請檢查 Firestore 是否已啟用並設為測
試模式。");
  } finally {
    testApproveBtn.textContent = '🔧 開發者測試：一鍵核准此
帳號';
  }
});

// 登出功能
logoutBtn.addEventListener('click', async () => {
  try {
    await signOut(auth);
  } catch (error) {
    showError("登出失敗，請稍後重試。");
  }
});
</script>
</body>
</html>

```

研習回饋問卷：簽退用，請勿現在就填寫

<https://docs.google.com/forms/d/e/1FAIpQLSf-0W7yugdKSINCEp1zZ-MbVNUbVQvqoi9A8N6kCVA5cnRHeQ/viewform?pli=1>