

Feature Specification: 貓狗領養平台 — 規格更新

Feature Branch: 002-title-description-project

Created: 2025-10-09

Status: Draft

Input: 使用者提交之平台願景、任務、目標與功能需求（詳見提交內容）

User Scenarios & Testing (mandatory)

此規格以「讓更多人以領養代替購買」為核心目標，優先支援會員瀏覽與提出領養申請、飼主發佈送養、以及管理員的審核流程。

User Story 1 - 瀏覽列表並搜尋/篩選動物 (Priority: P1)

描述：任何訪客或已登入會員可以在「可領養動物」頁面瀏覽列表，套用篩選與排序（物種、年齡、性別、縣市、關鍵字），並使用分頁或無限滾動查看結果。

Why this priority: 使用者尋找目標動物是平台的第一步，直接影響流量轉換與申請數量。

Independent Test: 未登入訪客或已登入會員在列表頁能夠使用至少 4 個不同篩選條件並得到符合條件的結果，分頁或滾動載入下一頁內容。

Acceptance Scenarios:

- Given:** 使用者進入列表頁，**When:** 選擇物種 = DOG 且 縣市 = Taipei，**Then:** 顯示所有符合條件的狗狗，且結果數量與分頁資訊正確。
- Given:** 使用者使用關鍵字搜尋（例如 "米克斯"），**When:** 點選搜尋，**Then:** 顯示包含關鍵字的動物卡片，關鍵字高亮或在結果摘要中顯示。
- Given:** 使用者啟用排序（例如依上架時間），**When:** 點選排序選單，**Then:** 清單以正確順序更新，且分頁正常運作。

Edge cases:

- 無符合條件結果時顯示友善提示與推薦（例如熱門動物）。
- 篩選參數組合過多或不合理時仍可回傳空結果但不崩潰。

User Story 2 - 動物詳情檢視 (Priority: P1)

描述：從列表進入個別動物詳情頁，包含多張圖片、描述、醫療紀錄摘要、飼主或收容所聯絡資訊（非敏感）、以及「提出申請」或「聯絡飼主」的動作。

Independent Test: 點選任一動物卡片能看到 3 張以上的圖片（若存在）、基本資料、醫療紀錄摘要與申請按鈕；點擊圖片可開啟圖片檢視器。

Acceptance Scenarios:

- Given:** 動物有醫療紀錄，**When:** 進入詳情頁，**Then:** 顯示最近三筆醫療紀錄摘要並提供「查看全部醫療紀錄」連結。
- Given:** 動物為送養狀態 PUBLISHED，**When:** 登入會員點擊「提出申請」，**Then:** 彈出申請表單。

Edge cases:

- 私有或受限制的聯絡資訊不直接顯示；需透過系統訊息或在申請通過後揭露。
 - 圖片缺失時顯示預設占位圖與替代文字。
-

User Story 3 - 提出領養申請 (Priority: P1)

描述：已登入的會員能在動物詳情頁提交領養申請，填寫必要欄位（聯絡方式、申請理由、可到訪時間等），並立即在「我的申請」看到該筆申請（狀態 PENDING）。

Independent Test: 已登入會員提交表單後系統回傳成功，並在 會員 -> 我的申請 列表看到新紀錄，狀態為 PENDING。

Acceptance Scenarios:

1. **Given:** 會員填寫完必填欄位， **When:** 點擊送出， **Then:** 申請建立（HTTP 201），並寄送 in-app 與 email 通知給送養者與管理員。
2. **Given:** 欄位驗證失敗（缺少必填）， **When:** 點擊送出， **Then:** 表單顯示錯誤並阻止提交。

Edge cases:

- 使用者在提交期間網路中斷：系統應提供重試或草稿快取機制。
 - 同一會員重複提交多次：系統應提示已有相似未審核申請。
-

User Story 4 - 飼主發佈/管理送養 (Owner Rehome) (Priority: P1)

描述：飼主（必須為已驗證會員）可建立送養貼文，包含至少一張圖片、描述、健康與醫療摘要；可在會員頁面編輯或下架該送養；管理員審核後變為公開 PUBLISHED。

Independent Test: 飼主建立送養後紀錄狀態為 SUBMITTED；管理員在後台批准後狀態變更為 PUBLISHED，前台可見。

Acceptance Scenarios:

1. **Given:** 飼主完成發布且上傳圖片， **When:** 提交， **Then:** 產生 SUBMITTED 紀錄並通知管理員。
2. **Given:** 飼主需要修改公開資訊， **When:** 在個人送養列表選擇編輯， **Then:** 可更新文字與圖片（圖片需重新上傳或標記保留）。

Edge cases:

- 若送養被拒，飼主會收到拒絕理由並可重新提交。
 - 若送養中已有正在審核的領養申請，飼主下架或刪除時需提示後續影響。
-

User Story 5 - 飼主審核 (收養者選擇) (Priority: P2)

描述：當有會員對飼主的已通過送養動物提出領養申請時，飼主可在自己的送養管理頁檢視所有申請紀錄、查看申請者資料摘要與訊息、並將其中一筆標記為 "同意面談" 或直接標記為 "接受" 或 "拒絕"（此操作會更新申請狀態並觸發通知）。單一已通過送養在任何時間只允許一筆處於審核中。

Independent Test: 飼主在送養管理頁成功對某筆申請做狀態更新（例如設為 UNDER_REVIEW 或 APPROVED），且更新在前台會員申請頁面同步顯示。

Acceptance Scenarios:

1. **Given:** 多筆申請存在，**When:** 飼主選擇一筆標記為 UNDER_REVIEW，**Then:** 其他申請自動維持 PENDING/WAITING，直到該結果為 REJECTED 才能啟動下一筆審核。
2. **Given:** 飼主接受申請，**When:** 點選 ACCEPT，**Then:** 申請狀態變為 APPROVED 並發送通知給申請者與管理員。

Edge cases:

- 飼主誤操作：提供確認對話框與操作回滾窗口（soft undo）在短時間內。
 - 若申請者被封鎖或已刪除帳號，顯示適當錯誤與處理流程。
-

User Story 6 - 管理員後台審核與資料管理 (Priority: P1)

描述：管理員在後台可以全面查看申請、動物、會員與收容所資料；對送養/領養申請進行審核（標記通過/拒絕/退回），新增或修改動物資料與醫療紀錄，並能執行刪除或恢復（soft-delete）操作。

Independent Test: 管理員能在後台將 SUBMITTED 的送養標為 PUBLISHED 並檢視變更日誌；能為動物新增醫療紀錄；能恢復被標為 soft-deleted 的動物。

Acceptance Scenarios:

1. **Given:** 一筆送養為 SUBMITTED，**When:** 管理員審核通過，**Then:** 變更為 PUBLISHED，並在 audit log 中記錄操作人與時間。
2. **Given:** 管理員刪除一筆動物，**When:** 刪除，**Then:** 標記為 soft-deleted，並能在一定期限內恢復。

Edge cases:

- 審核中同時有多位管理員操作：後端需使用適當鎖定或版本控制以避免 race condition。
 - 對於敏感資訊的操作需有更高權限提示且紀錄審計。
-

User Story 7 - 我的申請與資料管理 (Member self-service) (Priority: P2)

描述：會員可在個人頁面查看與管理自己提交的領養與送養申請，編輯可編輯的草稿、查看審核歷程、下載申請表、以及接收與查看通知與管理者或送養者的訊息紀錄。

Independent Test: 會員登入後在「我的申請」看到所有歷史紀錄；可下載某筆申請的 PDF 摘要。

Acceptance Scenarios:

1. **Given:** 會員有多筆申請，**When:** 進入我的申請頁，**Then:** 列表按時間排序並可 filter（PENDING/APPROVED/REJECTED）。
2. **Given:** 某筆申請為 PENDING，**When:** 會員點選撤回，**Then:** 申請狀態變為 WITHDRAWN 並通知相關方。

Edge cases:

- 會員嘗試編輯已進入審核程序的申請：系統應阻止並提供替代操作（例如補件或聯絡送養者）。
-

User Story 8 - 通知與通訊 (in-app + email, Priority: P2)

描述：系統在申請狀態變更、審核結果、留言或管理員操作時發送通知；會員可在個人設定選擇通知偏好（in-app、email、SMS）。

Independent Test: 當管理員在後台將申請標記為 APPROVED，申請者在 1 分鐘內收到 in-app 與 email 通知（若偏好設為 email）。

Acceptance Scenarios:

1. **Given:** 申請狀態變更， **When:** 變更發生， **Then:** 系統排程發送通知並在使用者的通知中心顯示。
2. **Given:** 使用者停用 email 通知， **When:** 事件發生， **Then:** 只產生 in-app 通知，不寄送 email。

Edge cases:

- 郵件被退回或丟失：系統需有重試機制與失敗記錄。
 - 通知頻率過高時應合併或節流（digests）。
-

User Story 9 - 簡易 in-app messaging (Priority: P3)

描述：會員可透過平台向送養者發送簡短訊息（文字 + 圖片附件限制），以便約定面談或追問細節；訊息記錄顯示於會員與送養者的對話紀錄中，管理員可於必要時查看或干預。

Independent Test: 會員 A 向飼主 B 發送訊息並在 30 秒內顯示於 B 的訊息中心；若 B 未讀，系統在 24 小時後提醒。

Acceptance Scenarios:

1. **Given:** 會員 B 有未讀訊息， **When:** B 登入， **Then:** 在訊息中心看到新訊息預覽與未讀計數。
2. **Given:** 訊息包含附件超過限制， **When:** 上傳， **Then:** 拒絕並提示最大大小與允許格式。

Edge cases:

- 若任一方被停權，訊息應該被凍結並僅供管理員查看。
-

User Story 10 - 動物醫療紀錄管理 (Priority: P1)

描述：管理員、收容所工作人員與飼主皆可為動物新增、編輯或附加醫療紀錄（recordType、date、provider、details、attachments），但所有新增或修改的紀錄初始為 unverified 狀態；系統須保留審核歷程（verified, verifiedBy, verifiedAt）與變更日誌，管理員可驗證（verify）或拒絕該紀錄。醫療紀錄的摘要會在動物詳情頁展示，完整紀錄僅顯示給有權限的使用者或在經過審核後公開。

Why this priority：醫療紀錄直接影響領養者對動物健康與適配性的判斷，為配對可靠性與平台信任的關鍵要素。

Independent Test：

- A. 飼主新增一則醫療紀錄並上傳附件，紀錄於系統顯示為 unverified 且出現在該動物的「待驗證醫療紀錄」清單中；管理員登入後能看待遇驗證項目並執行 Verify 操作，Verify 後該紀錄於動物詳情摘要顯示為已驗證。

Acceptance Scenarios：

1. **Given:** 飼主為已驗證會員， **When:** 上傳某次疫苗紀錄並填寫 recordType、date、provider、details， **Then:** 系統建立 MedicalRecord (status: unverified)，並在 audit log 中記錄 actor 與 timestamp。
2. **Given:** 管理員在後台審核該紀錄， **When:** 點選 Verify， **Then:** 記錄變為 verified=true，verifiedBy=adminId，verifiedAt=時間，並觸發 in-app 通知給紀錄建立者。

3. **Given:** 未經驗證的醫療紀錄，**When:** 一般公眾檢視動物詳情，**Then:** 只顯示醫療紀錄摘要或遮罩關鍵詳細資訊；經驗證後，則顯示更完整內容。
4. **Given:** 上傳的附件格式或大小不符，**When:** 使用者上傳，**Then:** 前端阻擋上傳並顯示錯誤（允許的格式與最大大小限制在規格中說明）。

Edge cases：

- 若多方（飼主與收容所）針對同一事件上傳不同版本的醫療紀錄，系統需保留來源與時間戳並允許管理員合併或選擇可信版本。
- 當醫療紀錄被標記為欺詐或不實，管理員可標示為 **disputed**，並將該紀錄隱藏或加入調查流程；同時保留審計記錄以供追溯。
- 若紀錄被刪除或修改，系統應採 **soft-delete** 與版本控制（保留歷史版本以遵守審計要求）。

Mapping to FRs：此 user story 直接對應 **FR-008**（醫療紀錄的新增、檢視、更新與審核）與 **FR-007/FR-010**（審核歷程與個資保護）。

Requirements (*mandatory*)

Functional Requirements

- **FR-001:** 系統 **MUST** 允許使用者註冊、登入與登出，會員資料可在個人頁面檢視與更新（生日於首次註冊可選填，之後可補填）。
- **FR-002:** 系統 **MUST** 允許會員瀏覽「可領養動物」清單，並支援基本篩選（物種、年齡範圍、性別、縣市）與分頁顯示。
- **FR-003:** 系統 **MUST** 在動物詳情頁顯示：名稱、物種、年齡、性別、描述、圖片集（支援多張）、收容所或飼主資訊及醫療紀錄摘要（若有）。
- **FR-004:** 系統 **MUST** 允許會員針對動物提出領養申請，並在會員介面列出該會員的所有申請紀錄（含狀態、提交時間、審核時間、審核備註）。
- **FR-005:** 系統 **MUST** 允許會員發佈送養申請（包含至少一張圖片與描述等必填欄位），送養初始狀態為 **SUBMITTED**，等待管理員審核。
- **FR-006:** 系統 **MUST** 允許管理員在後台檢視/審核/更新/刪除動物資料與所有申請，並可為收容所新增動物資料。
- **FR-007:** 對於每筆領養申請，系統必須記錄完整的審核歷程（誰在何時將狀態變更為何，並能儲存審核備註）。
- **FR-008:** 系統 **MUST** 支援動物之醫療紀錄的新增、檢視與更新（治療、健檢、疫苗）。管理員、收容所工作人員與飼主皆可新增或編輯醫療紀錄，但所有新增或修改的紀錄須包含審核標記（**verified**: boolean）並保留審核歷程（**verifiedBy**、**verifiedAt**）。建議必填欄位：**recordType**、**date**、**provider**、**details**；可接受附件 **attachments**（例如醫療證明照片）。
- **FR-009:** 系統 **MUST** 支援通知機制（例如：申請狀態變更、審核結果、管理員通知），通知型式（**email/SMS/in-app**）為選項化，預設為 **in-app** 與 **email**。

安全與隱私相關（非實作細節）

- **FR-010:** 系統 **MUST** 對會員個資（**email**、**電話**、**身份資訊**）採最低必要原則儲存，並提供管理員與使用者能夠查詢與匯出其個人資料之申請紀錄（符合法規要求）。

Key Entities (*include if feature involves data*)

- **Member:** 識別使用者帳號與基本資料（**username**, **email**, **phoneNumber**, **firstName**, **lastName**, **role**[**member**|**owner**|**admin**], **profile fields**）。

- **Animal:** 動物項目 (id, name, species[CAT|DOG], age, sex, description, images[], status[PUBLISHED|SUBMITTED|RETIRED], shelterId?, ownerId?) 。
- **Application:** 領養/送養申請 (id, applicantId, animalId, type[ADOPTION|REHOME], status[PENDING|UNDER_REVIEW|APPROVED|REJECTED], submittedAt, reviewedAt, reviewNotes, assigneeId) 。
- **Shelter:** 收容所資訊 (id, name, contact, address) 。
- **MedicalRecord:** 動物醫療紀錄 (id, animalId, recordType[TREATMENT|CHECKUP|VACCINE], date, details, provider, notes) 。
- **AuditLog:** 事件紀錄 (actorId, action, targetType, targetId, timestamp, details) —用於追蹤審核歷程與資料變更。

Execution Flows for Functional Requirements

為每個 Functional Requirement 提供可執行的流程藍圖 (Preconditions, Steps, Postconditions, Error handling, API example) 。開發者與 QA 可直接將其轉換為 API contract 與測試腳本。

FR-001: 使用者註冊 / 登入 / 登出

- Preconditions: 無 ; email 必須唯一 。
- Steps (註冊): 1) POST /auth/register {email,password,username} 2) server 驗證輸入並建立未驗證帳號 3) server 發送驗證信 4) 使用者點驗證連結 => GET /auth/verify?token=... => 帳號狀態 = VERIFIED 。
- Steps (登入): 1) POST /auth/login {email,password} 2) server 驗證憑證 3) 回傳 access token + refresh token 。
- Postconditions: 已驗證會員可使用受限功能 (發佈送養、提交申請) 。
- Error handling: 重複 email => 409 ; 格式錯誤 => 400 ; 憑證錯誤 => 401 。
- API example: POST /auth/register, POST /auth/login, POST /auth/logout (invalidate token)

FR-002: 瀏覽可領養動物清單 (篩選 / 分頁)

- Preconditions: 有 PUBLISHED 的 animal 資料 。
- Steps: 1) GET /animals?species=DOG&city=Taipei&page=1&size=20 2) server 回傳 items + pagination metadata 3) client render 。
- Postconditions: 列表與分頁顯示正確 。
- Error handling: 不合法參數 => 400 ; 超大 page size => 413 或自動調整回最大值 。
- API example: GET /animals

FR-003: 動物詳情 (含圖片集與醫療紀錄摘要)

- Preconditions: animalId 存在 , 且請求者有權查看 (公開或已授權) 。
- Steps: 1) GET /animals/{id} 2) server 回傳 animal + images[] + medicalRecordSummaries[] 3) client 顯示詳情與圖片檢視器 。
- Postconditions: 使用者能查看詳情並觸發下一步 (提出申請) 。
- Error handling: animal 不存在 => 404 ; 存取被拒 => 403 ; 圖片載入錯誤以 placeholder 顯示 。
- API example: GET /animals/{id}

FR-004: 提出領養申請

- Preconditions: requestor 為已登入會員 , animal 可申請狀態 (PUBLISHED) 。
- Steps: 1) GET 詳情顯示表單 2) POST /applications {applicantId,animalId,answers,attachments?} 3) server 建立 application {status:PENDING,submittedAt} 並通知 owner/admin 。

- Postconditions: 申請顯示於會員「我的申請」；觸發審核工作項。
- Error handling: 欄位驗證失敗 => 400；重複申請（相同 applicantId+animalId 且未結案）=> 409。
- API example: POST /applications

FR-005: 發佈送養（飼主）

- Preconditions: 使用者為已驗證會員並擁有 owner 權限或想要發佈的 animal 資料。
- Steps: 1) POST /rehomes {ownerId,animal:{...},images[],medicalSummary} 2) server 建立 record status=SUBMITTED 3) 通知管理員待審核。
- Postconditions: 飼主看到 SUBMITTED；管理員在後台處理。
- Error handling: 圖片上傳失敗 => 422；少必填欄位 => 400。
- API example: POST /rehomes

FR-006: 管理員檢視 / 審核 / 更新 / 刪除

- Preconditions: requestor role = admin。
- Steps (審核): 1) GET /admin/rehomes?status=SUBMITTED 2) admin GET /admin/rehomes/{id} 3) POST /admin/rehomes/{id}/review {action: APPROVE|REJECT,notes} 4) server 更新狀態、寫入 auditLog、通知相關方。
- Steps (刪除): DELETE /admin/animals/{id} => soft-delete 並記錄 audit。
- Postconditions: 變更反映前台，auditLog 可查。
- Error handling: 權限不足 => 403；版本衝突 => 409。
- API example: POST /admin/rehomes/{id}/review, DELETE /admin/animals/{id}

FR-007: 記錄完整審核歷程（Audit）

- Preconditions: 所有改變申請與動物狀態的操作均會觸發 audit entry。
- Steps: 在每個狀態變更的 API 中，同步或透過可靠隊列寫入 AuditLog {actorId,action,targetType,targetId,before,after,timestamp}；提供 GET /admin/audit?targetType=application&targetId=...
- Postconditions: audit 可查且不可被普通帳號刪除。
- Error handling: 若寫入失敗，將事件入 queue 並回傳 202（接受但延後處理）。
- API example: GET /admin/audit?targetType=...

FR-008: 動物醫療紀錄的新增 / 檢視 / 更新 / 審核

- Preconditions: requestor role in (admin,shelter,owner) 可新增；公眾僅可查看已驗證摘要。
- Steps (新增): POST /animals/{id}/medical-records {recordType,date,provider,details,attachments?} => 建立 status=unverified 並記 audit；管理員於 /admin/medical-records 審核並 POST /medical-records/{id}/verify {verified:true|false,notes}
- Postconditions: verified=true 的紀錄顯示於詳情摘要；unverified 僅出現在待驗證清單。
- Error handling: 附件格式/大小錯誤 => 422；衝突/重複 => 409。
- API example: POST /animals/{id}/medical-records, POST /medical-records/{id}/verify

FR-009: 通知機制（in-app + email）

- Preconditions: 使用者有通知偏好設定（預設 in-app+email）。
- Steps: 1) 事件觸發（application status change 等）2) server push event to notification queue 3) worker 發送 in-app 存入 DB，並發 email（外部 provider）4) 更新通知狀態並重試失敗項目。
- Postconditions: 使用者在通知中心看到事件；email 若失敗被重試並記錄。

- Error handling: 外部 provider 失敗回傳 502/503 => 依重試策略處理並在超過次數後記錄 failed。
- API example: GET /notifications, POST /notifications/{id}/mark-read

FR-010: 個資保護與查詢/匯出 (Data Privacy)

- Preconditions: 使用者為 data owner 或 admin ; 需通過驗證流程。
- Steps (匯出): POST /data/export {userId,requesterId} => server 驗證權限、生成資料包 (含申請紀錄、個資) , 並提供下載連結 (暫時可用於 24hr) 。
- Steps (刪除): POST /data/delete {userId,requesterId} => server 執行資料匿名化或刪除流程 , 並記錄於 auditLog ; 如涉及備份 , 標記待清理或列入刪除政策。
- Postconditions: 匯出檔案可供下載 ; 刪除請求後主要可見資料被移除或匿名化 ; audit 記錄保存。
- Error handling: 權限不足 => 403 ; 長時間作業建議回傳 202 並提供查詢 job status 的 endpoint (GET /jobs/{id}) 。
- API example: POST /data/export, POST /data/delete, GET /jobs/{id}