

# Petfolio – Week 3 Progress

Pet CRUD + Photo Upload

## This Week's Goals

- Implement Pet CRUD UI (list, create, edit, delete)
- Add photo upload with Firebase Storage
- Enforce owner-only access with Firestore/Storage rules
- Ship beginner-friendly, commented code

# What We Shipped

## UI & UX

- Pet Dashboard with avatars and summary
- Create/Edit Pet form (prefilled when editing)
- Photo picker with preview, progress states, errors
- Delete with confirmation dialog

## Data & Architecture

- PetsRepository (Firestore + Storage)
- Riverpod `PetListNotifier` streaming user's pets
- Real-time list updates on create/update/delete

## Security

- Firestore rules: owner-only create/read/update/delete for `pets`
- Storage rules: owner-only writes; reads allowed via auth or download token

## Technical Notes

- Web-specific fixes:
  - Image preview uses `MemoryImage` on web, `FileImage` on mobile/desktop
  - Storage upload uses bytes on web, file on mobile/desktop
  - CORS: configured Cloud Storage CORS with a fixed dev origin (<http://localhost:5555>)
- Removed temporary Firestore smoke-test write
- Cache-busted `NetworkImage` to avoid stale 403s

## Demo Path

1. Run web on fixed port: `flutter run -d chrome --web-port 5555`
2. Sign in → Home
3. Add Pet (with or without photo)
4. Edit Pet (change name/species/photo)
5. Delete Pet → disappears from list



## Current Status vs Plan

- Pet CRUD UI: Complete
- Photo Upload: Complete (Firebase Storage)
- Enhanced Pet Fields: Basic fields done; extend next week
- Rules (Firestore/Storage): Published and verified

## Known/Resolved Issues

- Web CORS blocked images → Fixed with bucket CORS config
- Index requirement on Firestore → Avoided by removing orderBy temporarily; can add index later
- Edit button created new pet → Fixed by passing `Pet` into edit route





## Next (Week 4 Preview)

- Care Plan model (diet, feeding, meds)
- Care Plan UI + validation
- Local notifications for reminders
- Optional: restore list ordering with Firestore composite index

## Appendix: References

- See Week 1 deck for project overview
- See Week 2 deck for Auth details