

Implementation Checklist

Phase 1: Setup (5 minutes)

- ☐ STEP 1: Create src/services/ folder



bash

```
mkdir src/services
```

Phase 2: Create Service Files (10 minutes)

- ☐ STEP 2: Create src/services/applicationService.js
 - Copy from artifact: "Refactored Hook with Service Layer"
 - Should be ~280 lines
- ☐ STEP 3: Create src/services/validationService.js
 - Copy from artifact: "Validation Service"
 - Should be ~115 lines

Phase 3: Update Core Files (10 minutes)

- ☐ STEP 4: Backup then replace src/hooks/useApplication.js



bash

```
cp src/hooks/useApplication.js src/hooks/useApplication.js.backup
```

- Copy from artifact: "Clean useApplication Hook (117 lines)"
- Should be ~117 lines
- ☐ STEP 5: Backup then update src/app/admin/applications/[id]/page.js



bash

```
cp src/app/admin/applications/[id]/page.js src/app/admin/applications/[id]/page.js.backup
```

- Copy from artifact: "Refactored Application Detail Page"
-

Phase 4: Create New Components (15 minutes)

- ☐ **STEP 6:** Create `src/components/admin/applications/StatusCard.jsx`
 - Copy `StatusCard` section from: "New Application Components"
- ☐ **STEP 7:** Create `src/components/admin/applications/InterviewScheduleCard.jsx`
 - Copy `InterviewScheduleCard` section from: "New Application Components"
- ☐ **STEP 8:** Create `src/components/admin/applications/NotesSection.jsx`
 - Copy `NotesSection` section from: "New Application Components"

Phase 5: Update Existing Components (10 minutes)

- ☐ **STEP 9:** Backup then update `FeedbackSection.jsx`



bash

```
cp src/components/admin/applications/FeedbackSection.jsx src/components/admin/applications/FeedbackSection.jsx.backup
```

- Copy `FeedbackSection` from: "New Application Components"
- ☐ **STEP 10:** Backup then update `ScoringSection.jsx`



bash

```
cp src/components/admin/applications/ScoringSection.jsx src/components/admin/applications/ScoringSection.jsx.backup
```

- Copy `ScoringSection` from: "New Application Components"
- ☐ **STEP 11:** Backup then update `ApplicationHeader.jsx`



bash

```
cp src/components/admin/applications/ApplicationHeader.jsx src/components/admin/applications/ApplicationHeader.jsx.backup
```

- Copy from artifact: "Updated Application Header"

Phase 6: Verification (5 minutes)

- ☐ **STEP 12:** Verify all files exist

New Files (5):



src/services/applicationService.js
src/services/validationService.js
src/components/admin/applications/StatusCard.jsx
src/components/admin/applications/InterviewScheduleCard.jsx
src/components/admin/applications/NotesSection.jsx

Updated Files (5):



src/hooks/useApplication.js
src/app/admin/applications/[id]/page.js
src/components/admin/applications/FeedbackSection.jsx
src/components/admin/applications/ScoringSection.jsx
src/components/admin/applications/ApplicationHeader.jsx

Phase 7: Testing (20 minutes)

- ☐ **STEP 13:** Start dev server



bash

`npm run dev`

Test Each Action:

- ☐ **Test 1:** Change status to "reviewed"
 - Timeline shows: "status_changed" event
- ☐ **Test 2:** Try to reject without reason
 - Should show error: "يرجى إدخال سبب الرفض"
- ☐ **Test 3:** Reject with reason
 - Timeline shows: "rejected" event with reason
- ☐ **Test 4:** Schedule interview
 - Timeline shows: "interview_scheduled" event
- ☐ **Test 5:** Reschedule interview
 - Timeline shows: "interview_rescheduled" with old→new
- ☐ **Test 6:** Add HR notes
 - Timeline shows: "notes_added" event
- ☐ **Test 7:** Update existing notes

- Timeline shows: "notes_updated" event
- ☐ **Test 8:** Add feedback
 - Timeline shows: "feedback_added" event
- ☐ **Test 9:** Add score
 - Timeline shows: "score_added" event
- ☐ **Test 10:** Complete interview
 - Timeline shows: "interview_completed" event

Quick Reference: What Each File Does

File	Purpose
applicationService.js	Business logic + API calls
validationService.js	Validation rules
useApplication.js	State management + UI logic
StatusCard.jsx	Status change UI
InterviewScheduleCard.jsx	Interview scheduling UI
NotesSection.jsx	Notes management UI
FeedbackSection.jsx	Feedback management UI
ScoringSection.jsx	Score management UI
ApplicationHeader.jsx	Header with edit button

Rollback Plan (If Something Goes Wrong)

If you need to revert:



bash

Restore useApplication.js

cp src/hooks/useApplication.js.backup src/hooks/useApplication.js

Restore page.js

cp src/app/admin/applications/[id]/page.js.backup src/app/admin/applications/[id]/page.js

Restore FeedbackSection.jsx

cp src/components/admin/applications/FeedbackSection.jsx.backup src/components/admin/applications/FeedbackSection.jsx

Restore ScoringSection.jsx

cp src/components/admin/applications/ScoringSection.jsx.backup src/components/admin/applications/ScoringSection.jsx

Restore ApplicationHeader.jsx

cp src/components/admin/applications/ApplicationHeader.jsx.backup src/components/admin/applications/ApplicationHeader.jsx

Remove new files

rm -rf src/services

rm src/components/admin/applications/StatusCard.jsx

rm src/components/admin/applications/InterviewScheduleCard.jsx

rm src/components/admin/applications/NotesSection.jsx



Estimated Total Time: ~1 hour

- Setup: 5 min
- Create services: 10 min
- Update core: 10 min
- Create components: 15 min
- Update components: 10 min
- Verification: 5 min
- Testing: 20 min

Success Criteria ✨

You're done when:

- ☐ No console errors
- ☐ All 10 test actions work
- ☐ Timeline shows correct events
- ☐ Each section has its own save button
- ☐ No duplicate timeline entries
- ☐ Edit mode toggles correctly

Congratulations!

Once all checkboxes are ☒, you have successfully:

- Reduced hook from 541 → 117 lines
- Separated concerns into 3 layers
- Made code maintainable and scalable
- Implemented independent action handlers
- Created clear timeline tracking