Memoir.ai — Engineering Runbook

Operational Procedures for Engineering & Platform Teams

Document Version: 1.0

Status: Operational Reference

Owner: Engineering Operations

Last Updated: YYYY-MM-DD

---

1. Purpose

This runbook provides operational procedures for engineering teams responsible for maintaining, deploying, debugging, and recovering Memoir.ai systems. It ensures consistent responses to operational events and platform changes.

---

2. Scope

Covers:

• Development and production environments

• Deployment and rollback operations

• Vault and database recovery

• AI pipeline operations

• Performance troubleshooting

• Logging and monitoring response

• Incident response actions

Does not cover end-user workflows except where operational impact occurs.

---

3. System Architecture Overview

Memoir.ai consists of:

• Electron desktop application shell

• Local encrypted SQLCipher database

• AI processing workers

• Optional Supabase metadata synchronization

• Stripe-based billing and entitlement validation

All sensitive content remains local to user vaults.

---

4. Engineering Responsibilities

Backend Engineers:

• Maintain ingestion and AI pipelines.

• Ensure database integrity and query performance.

• Validate IPC and job execution security.

Frontend Engineers:

• Maintain UI state correctness.

• Handle loading/error states safely.

• Ensure accessibility and performance compliance.

DevOps Engineers:

• Maintain CI/CD reliability.

• Package, sign, and distribute builds.

• Manage release and rollback procedures.

Security Engineers:

• Audit encryption and data handling.

• Review access control mechanisms.

• Monitor dependency vulnerabilities.

---

5. Development Environment Setup

Requirements:

• Node.js v20+

• Package manager configured

• Electron builder installed

• SQLCipher-compatible environment

Setup Flow:

1. Clone repository.

2. Install dependencies.

3. Configure environment variables.

4. Start development environment.

5. Verify vault creation and encryption flow.

---

6. Deployment Procedure

Standard deployment steps:

1. Verify tests pass.

2. Perform dependency security checks.

3. Build UI and backend bundles.

4. Package Electron application.

5. Sign binaries.

6. Publish release artifacts.

7. Trigger update distribution.

Always verify application launch and vault creation post-build.

---

7. Release Gates

Release promotion requires:

• Test suite passes

• No critical vulnerabilities

• Performance targets met

• Manual product review completed

Failure blocks deployment.

---

8. Rollback Procedure

If deployment fails:

1. Pause auto-update channel.

2. Revert release pointer to previous version.

3. Notify users if downgrade required.

4. Validate restored version stability.

Database compatibility must be maintained across versions.

---

9. Database & Vault Recovery

Vault Corruption Response:

1. Attempt integrity check.

2. Trigger index rebuild.

3. Restore from latest backup if needed.

Never modify encrypted DB files manually.

---

10. AI Pipeline Operations

Snapshot Generation Flow:

• Evidence extraction

• Prompt construction

• Local model inference

• Hallucination guard validation

• Citation linking

• Version storage

Failures should retry inference or flag snapshot as incomplete.

---

11. Logging & Diagnostics

Log Levels:

• FATAL: Application crash or DB failure

• WARN: Import or processing issues

• INFO: Job lifecycle events

• DEBUG: Development-only IPC tracing

Logs must not include personal content.

---

12. Monitoring & Alerts

Monitor:

• Disk space

• Worker memory usage

• Database latency

• Job execution times

Alert users for resource exhaustion risks.

---

13. Incident Response

Operational incidents:

1. Identify issue scope.

2. Isolate faulty components.

3. Preserve logs.

4. Apply mitigation.

5. Publish patch if required.

Security incidents:

• Confirm breach possibility.

• Advise vault export and wipe procedures.

• Patch vulnerabilities immediately.

---

14. Performance Troubleshooting

Common issues:

• Slow timeline queries: rebuild indexes.

• Import stalls: verify parsing workers.

• Memory spikes: restart background workers.

• Slow search: confirm indexing completion.

---

15. Secrets & Key Handling

Engineering rules:

- Never store passphrases.
- Never log encryption keys.
- Rotate internal tokens regularly.
- Restrict build secrets to CI environments.

---

16. Backup Recommendations

Recommended procedures:

- Automatic vault backups enabled.
- Users reminded to export periodically.
- Backups stored on separate media.

---

17. Support Interaction Rules

Engineering and support must:

- Never request vault passphrases.
- Never request raw vault content.
- Use sanitized diagnostics only.

---

18. Operational Best Practices

Recommended practices:

- Test migrations locally before release.
- Monitor performance regressions.
- Audit dependencies regularly.
- Validate entitlements gating.

---

19. Conclusion

This runbook standardizes engineering responses and operational execution for Memoir.ai. Adhering to these procedures ensures privacy preservation, reliability, and safe platform evolution.