

# extranotes 20250614 2219 le storage

AMAZING! You're already using it and creating notes! That's the best validation.

But you're absolutely right - data security is CRITICAL for 10,000 notes. Let me think strategically about this:

## CURRENT SITUATION (Risk Assessment)

localStorage (what we have now):

- Works for testing
- Browser can clear it (disaster!)
- No backup mechanism
- Size limits (~5-10MB max)
- Not portable between devices

For 10k notes = DISASTER WAITING TO HAPPEN!

## PROPOSED SECURE STORAGE STRATEGY

Phase 1: IMMEDIATE SAFETY (do this NOW!)

Current localStorage → Export backup → File system storage

1. Export current notes to JSON file immediately
2. Migrate to Tauri file system storage
3. Auto-backup every hour to multiple locations

Phase 2: BULLETPROOF ARCHITECTURE

Primary: SQLite database (speed + reliability)

Backup 1: Plain text files (human readable)

Backup 2: JSON exports (data integrity)

Cloud: Optional sync to personal cloud

## RECOMMENDED ARCHITECTURE FOR 10K+ NOTES

4. Primary Storage: SQLite Database

```
// Tauri SQLite with full-text search
~/Library/Application Support/Extranuts/
└─ notes.db // Main SQLite database
```

```
├─ notes_backup.db // Daily backup
├─ exports/
├─ 2025-06-14.json
├─ plain_text/
├─ note_001.md
├─ note_002.md
```

Why SQLite?

- Handles millions of records
- ACID transactions (no corruption)
- Full-text search (instant even with 10k notes)
- Portable single file
- Industry standard (used by WhatsApp, etc.)

## 2. Triple Backup Strategy

Real-time: Auto-save to SQLite

Hourly: Export to JSON

Daily: Export to plain text files

Weekly: Full database backup

## 3. Future-Proof Formats

- JSON = machine readable, structured
- Markdown = human readable, universal
- SQLite = queryable, performant

MIGRATION PLAN (No Data Loss!)

Step 1: IMMEDIATE SAFETY (let's do this right now!)

```
// Export all current notes as backup
const backupData = {
  notes: notes(),
  categories: categories(),
  tags: tags(),
  timestamp: new Date().toISOString(),
  version: "1.0"
}
// Save to file via Tauri
```

Step 2: Enhanced Storage Layer

```
// New storage architecture
interface StorageEngine {
  primary: SQLiteStorage // Main database
```

```
backup: FileSystemBackup // Auto-backup
export: ExportManager // Human-readable exports
sync: CloudSync // Optional future feature
}
```

#### MY STRONG RECOMMENDATIONS:

##### Priority 1: IMPLEMENT NOW

1. Export backup function (save your current notes!)
2. Migrate to Tauri file system storage
3. Auto-backup mechanism

##### Priority 2: WITHIN A WEEK

4. SQLite integration for performance
5. Full-text search capability
6. Import/export to multiple formats

##### Priority 3: FUTURE-PROOFING

7. iCloud folder sync (optional)
8. Git integration for version history
9. Obsidian vault compatibility

#### IMMEDIATE ACTION PLAN

Can we implement the export backup RIGHT NOW? I can add:

10. "Export All Notes" button in header
11. Save to Downloads as JSON + Markdown files
12. Auto-backup every time you add/edit notes

Should I code this immediately to protect your existing notes?

This is critical - let's secure your data first, then we can implement the advanced storage later.

What do you think? Should I start with the immediate backup feature?