# Guidance for the Next AI Developer Team

This document summarises the mission, current state, and recommended practices for continuing development of the **Spectra App** project. It distils information from the project’s internal documents (brains, atlas, patch notes, handoffs) to help new contributors understand the scope of work and maintain the project’s standards.

## 1. Mission and Success Criteria

The project’s goal is to build a **research‑grade web application** that lets users upload spectra and compare them against archival references across the UV/Vis/IR range. According to the authoritative brief, the mission requires the system to:

* **Ingest heterogeneous formats** (CSV/TXT/FITS) and normalise them into a canonical internal model[[1]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L8-L17). The data model uses a baseline axis of vacuum‑nm wavelengths and supports multiple intensity modes (flux density, relative intensity, transmission, absorbance and optical depth)[[2]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L102-L111).
* **Fetch spectra and related products** from major archives (SIMBAD, MAST, SDSS) and atomic/molecular line lists[[3]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L12-L16).
* **Compare spectra fairly** using physically honest transforms: wavelength standard toggles, unit conversions, air↔vacuum corrections, resolution matching and velocity/frame corrections[[4]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L20-L23).
* **Provide A−B and A/B analyses** with numerical safeguards[[5]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L15-L16).
* **Export reproducible bundles** that include the current view and manifest with citations/DOIs[[6]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L16-L17).

Success criteria include fast plotting, minimal and accessible UI chrome, scientifically correct physics, pervasive provenance logging and a documentation tab explaining usage and data sources[[7]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L18-L25). The brief mandates Python 3.11, Streamlit (initially), Plotly/Altair for plotting, astrophysics libraries (Astropy, astroquery, specutils), and strict quality gates (pytest, ruff, black, mypy, pre‑commit)[[8]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L28-L38).

### Repository Structure and Continuity

The canonical repository structure is specified in the brief, with separate app, server, data, docs/static, atlas, brains, handoffs, PATCH\_NOTES, tools/verifiers, and tests folders[[9]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L43-L94). It also defines a **continuity system**: every run should update:

* **/atlas**: system design notes (e.g., architecture, data model, fetcher overview).
* **/brains**: run journal documenting context, changes, decisions, tests, follow‑ups, and checklists (e.g., brains/v1.0.0l\_\_assistant\_\_archive\_fetchers.md[[10]](https://github.com/brettadin/spectrasuite/blob/main/brains/v1.0.0l__assistant__archive_fetchers.md#L7-L16)).
* **PATCH\_NOTES/**: user‑facing change logs.
* **handoffs/**: high‑level summaries of each run with next steps, decisions and quick‑start notes[[11]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/HANDOFF_v1.0.0(l).md#L2-L34).

Always update these artifacts when you introduce features, fixes or documentation changes to maintain the historical record.

## 2. Current State (v1.0.0l / 1.0.0.dev12)

The latest handoff lists the following working components[[12]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/HANDOFF_v1.0.0(l).md#L8-L14):

* **Tabs:** Overlay (with axis and transform provenance captions, line overlays and export), Differential, Star Hub (SIMBAD resolver + new archive metadata), Line Atlas, and Docs[[13]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/HANDOFF_v1.0.0(l).md#L8-L13).
* **Ingestion:** ASCII/FITS loaders handle wavelength, wavenumber, frequency and energy axes with provenance logging[[14]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/HANDOFF_v1.0.0(l).md#L10-L11).
* **Archive fetchers:** Operational MAST and SDSS adapters return spectroscopic products with coverage/resolution metadata[[15]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/HANDOFF_v1.0.0(l).md#L4-L12). The brains log details the normalisation and decisions made when implementing these fetchers[[16]](https://github.com/brettadin/spectrasuite/blob/main/brains/v1.0.0l__assistant__archive_fetchers.md#L7-L17).

The handoff’s **next steps** include wiring the new Product outputs into the Star Hub UI, capturing mission‑specific citations/DOIs, and evaluating caching strategies[[17]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/HANDOFF_v1.0.0(l).md#L15-L18). Follow these priorities before adding new features.

## 3. Key Practices for Future Development

1. **Maintain the continuity system.** Every run should create or update the relevant brains entry (context, changes, decisions, tests, regressions and follow‑ups)[[18]](https://github.com/brettadin/spectrasuite/blob/main/brains/v1.0.0l__assistant__archive_fetchers.md#L18-L43), patch notes, handoff, and atlas sections. Use the verifiers in tools/verifiers to ensure these artifacts are updated.
2. **Document decisions and rationale.** When implementing features (e.g., fetchers, transforms, UI), record why you chose particular algorithms or conventions. This helps avoid regressions and guides future developers[[18]](https://github.com/brettadin/spectrasuite/blob/main/brains/v1.0.0l__assistant__archive_fetchers.md#L18-L43).
3. **Follow strict testing and linting.** Use pytest with hypothesis where appropriate, run ruff, black and mypy, and verify the UI contract and documentation using the provided scripts[[19]](https://github.com/brettadin/spectrasuite/blob/main/brains/v1.0.0l__assistant__archive_fetchers.md#L18-L28).
4. **Avoid merge artefacts.** Do not commit unresolved conflict markers (<<<<<<<, =======, >>>>>>>) or leftover scaffolding. Review diffs carefully before opening pull requests to prevent manual fixes and formatting errors.
5. **Implement features incrementally** and **update docs as you go.** For example, after adding the archive fetchers, the project updated atlas/fetchers\_overview.md and the relevant brains, patch notes and handoff files[[20]](https://github.com/brettadin/spectrasuite/blob/main/brains/v1.0.0l__assistant__archive_fetchers.md#L40-L43). Do the same for new transforms, UI components or data models.
6. **Preserve scientific honesty.** Ensure all transforms (unit conversions, air↔vacuum, resolution matching, velocity shifts) are reversible and recorded in provenance logs[[21]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L118-L147). Provide optional controls in the UI for users to choose transforms rather than silently applying them.
7. **Accessibility and performance.** Keep the UI minimal and keyboard‑friendly as mandated[[22]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L20-L25). Use Plotly’s WebGL or Altair for large traces and consider downsampling or asynchronous loading to maintain responsiveness.

## 4. Quick Launch and Clean Setup

Although the repository includes a README with basic setup instructions, future contributors should provide a **one‑click script** (e.g., launch.sh or Makefile) that:

1. **Creates a Python virtual environment** using python -m venv .venv.
2. **Installs dependencies** with pip install -e .[dev].
3. **Runs static checks and tests** (ruff check ., black --check ., mypy ., pytest -q), failing fast if any gate fails.
4. **Verifies continuity artifacts** by running the verifiers in tools/verifiers.
5. **Launches the Streamlit app** via streamlit run app/app\_patched.py.

Provide separate commands for a **quick smoke test** (install + run tests) and a **clean launch** (fresh venv + run app). Document these in the README and ensure they work on Windows, macOS and Linux. Consider using pipx or a Makefile for cross‑platform convenience.

## 5. Areas for Improvement and Outstanding Issues

* **Integrate archive fetchers into the UI:** The new MAST/SDSS adapters currently return Product metadata but are not yet wired into the Star Hub for users to load spectra. Implement this, then update the atlas, brains and patch notes[[17]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/HANDOFF_v1.0.0(l).md#L15-L18).
* **Add manual axis overrides and error‑bar display:** The overlay automatically detects axis families and units but misclassifications can occur. Provide UI controls to override detected axes and display uncertainty arrays (error bars) for FITS data.
* **Improve caching and network resilience:** Evaluate caching strategies and retry logic for archive calls to improve performance and robustness[[23]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/HANDOFF_v1.0.0(l).md#L17-L18).
* **Enrich documentation:** Populate docs/static with tutorials, FAQs and example workflows. Link to atlas and brains entries to help users understand the data model, transforms and provenance.
* **One‑click export and replay:** Ensure the export bundle reproduces the UI state and implement a replay function to reload sessions from manifests.

By following these guidelines, future AI developers can build on the existing foundation, maintain continuity and scientific rigor, and deliver a polished, reproducible spectral analysis tool.

[[1]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L8-L17) [[2]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L102-L111) [[3]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L12-L16) [[4]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L20-L23) [[5]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L15-L16) [[6]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L16-L17) [[7]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L18-L25) [[8]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L28-L38) [[9]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L43-L94) [[21]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L118-L147) [[22]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt#L20-L25) GitHub

[https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI Handoff Initial.txt](https://github.com/brettadin/spectrasuite/blob/main/handoffs/AI%20Handoff%20Initial.txt)

[[10]](https://github.com/brettadin/spectrasuite/blob/main/brains/v1.0.0l__assistant__archive_fetchers.md#L7-L16) [[16]](https://github.com/brettadin/spectrasuite/blob/main/brains/v1.0.0l__assistant__archive_fetchers.md#L7-L17) [[18]](https://github.com/brettadin/spectrasuite/blob/main/brains/v1.0.0l__assistant__archive_fetchers.md#L18-L43) [[19]](https://github.com/brettadin/spectrasuite/blob/main/brains/v1.0.0l__assistant__archive_fetchers.md#L18-L28) [[20]](https://github.com/brettadin/spectrasuite/blob/main/brains/v1.0.0l__assistant__archive_fetchers.md#L40-L43) GitHub

<https://github.com/brettadin/spectrasuite/blob/main/brains/v1.0.0l__assistant__archive_fetchers.md>

[[11]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/HANDOFF_v1.0.0(l).md#L2-L34) [[12]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/HANDOFF_v1.0.0(l).md#L8-L14) [[13]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/HANDOFF_v1.0.0(l).md#L8-L13) [[14]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/HANDOFF_v1.0.0(l).md#L10-L11) [[15]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/HANDOFF_v1.0.0(l).md#L4-L12) [[17]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/HANDOFF_v1.0.0(l).md#L15-L18) [[23]](https://github.com/brettadin/spectrasuite/blob/main/handoffs/HANDOFF_v1.0.0(l).md#L17-L18) GitHub

<https://github.com/brettadin/spectrasuite/blob/main/handoffs/HANDOFF_v1.0.0(l).md>