Thank you for your interest in improving SpotAxis! We welcome contributions of all kinds–from bug reports and documentation tweaks to code enhancements and shiny new features. By collaborating, we make SpotAxis stronger and more useful for everyone

Table of Contents

Who Can Contribute?
Everyone is welcome to contribute to SpotAxis, including but not limited to:

- **End users** who discover bugs or have ideas for new features
- **Developers** who want to improve code quality, add modules, or refactor existing functionality
- **Testers and QA engineers** who can reproduce issues, write tests, and improve reliability
- **Technical writers and designers** who can enhance documentation, examples, or UI/UX guidance

- **Community maintainers** who triage issues, review pull requests, and welcome newcomers

No prior experience with Git or Applicant Tracking Systems is required—your enthusiasm and fresh perspectives are highly valued.

How to Contribute

1. **Review the Code of Conduct:** Please read and adhere to our Code of Conduct to maintain a welcoming environment for all contributors.

2. **Search Existing Issues:** Before opening a new issue, search both open and closed issues to avoid duplicates.

3. **Open a New Issue:** If your concern isn't already reported, open an issue using GitHub's issue form and fill out all required fields—this helps us help you faster.

4. **Fork and Clone the Repository:** This keeps your work organized and isolated.

5. **Create a Branch:** Branch from `main` with a descriptive name (e.g., `feat/new-auth-flow` or `fix/login-bug`).

6. **Implement Your Changes:** Follow existing code style, add or update tests, and ensure new code is covered by tests.

7. **Run Tests and Linters:** Install dependencies (`npm install` or `pip install -r requirements.txt`), run the test suite (`npm test` or `pytest`), and fix any linting errors (ESLint, Prettier, Flake8).

8. **Commit and Push:** Use clear, imperative-style commit messages (see below), reference the related issue number, then push your branch to your fork.

9. **Open a Pull Request:** Target the `main` branch of `Assystant/SpotAxis`, use our PR template, link the issue (if any), summarize your changes, and include screenshots or recordings for UI updates.

10. **Respond to Feedback:** Maintain an open dialogue; address review comments promptly and update your PR until it's ready to merge.

Do's and Don'ts

Do

- Follow the Code of Conduct at all times
- Search for existing issues before creating new ones
- Write focused, self-contained pull requests with a single purpose
- Include tests for new features or bug fixes
- Keep commit messages clear, concise, and in imperative mood
- Run the full test suite and linters locally before submitting

Don't

- Submit large, monolithic PRs without prior discussion
- Combine formatting changes with functional changes in the same commit
- Push code that fails tests or linting checks
- Include sensitive information (passwords, API keys) in issues or code
- Ignore or dismiss reviewer feedback—collaboration is iterative

---

Code Style

Consistency is the most important thing in SpotAxis. Always match the existing style, formatting, and naming conventions of the file you're touching and the project as a whole—this keeps reviews focused on functionality rather than superficial tweaks.

For example, if private properties are prefixed with an underscore (`_userData`), please continue that pattern for any new properties you add. If methods use camelCase (`thisIsMyNewMethod`), don't switch to snake_case (`this_is_my_new_method`). When in doubt, skim nearby code or ask in a PR comment for guidance.

Whenever possible, style and formatting will be enforced automatically with our configured linter, so be sure to run it locally before committing.

Commit Message Guidelines

1. **Subject line** in imperative mood, ≤ 50 characters (e.g., “Add timeout to login”)
2. **Blank line** separating subject from body
3. **Body** explains *why* the change was needed; wrap lines at 72 characters
4. **No period** at end of subject line
5. **Reference** the issue number when applicable

Bug & Feature Request Guidelines

Before filing or reviewing a request:

- Confirm you’re on the latest SpotAxis release
- Search existing issues to avoid duplicates
- Fill out all required fields in GitHub’s issue form for clarity and context
- We’ll label issues as “bug,” “enhancement,” or “documentation,” and may ask for additional details (environment, reproduction steps)

Reporting Security Issues

For security vulnerabilities, please **do not** open a public issue.

Community & Support

Join our Discord community for real-time chat, questions, and camaraderie:
Discord: https://discord.gg/AVAMpXsA

Roadmap

Our detailed roadmap is **coming soon**—stay tuned for upcoming features and priorities!

Acknowledgments

This document draws inspiration from:

- Jesses Squires' CONTRIBUTING.md template
- Mozilla Science's "Wrangling Web Contributions" workshop
- GitHub Open Source Guides

Your contributions keep SpotAxis moving forward—thank you for helping us build a better platform!