**Contributing to SportAI Suite**

First off, thank you for considering contributing to SportAI Suite! 🎉

**Code of Conduct**

By participating in this project, you agree to abide by our Code of Conduct:

* Be respectful and inclusive
* Welcome newcomers and help them get started
* Focus on constructive criticism
* Show empathy towards other community members

**How Can I Contribute?**

**🐛 Reporting Bugs**

Before creating bug reports, please check existing issues to avoid duplicates.

**To submit a bug report:**

1. Use the issue template
2. Include detailed steps to reproduce
3. Include system information
4. Add screenshots if applicable
5. Describe expected vs actual behavior

**💡 Suggesting Enhancements**

Enhancement suggestions are tracked as GitHub issues.

**To suggest an enhancement:**

1. Use a clear and descriptive title
2. Provide a detailed description of the proposed functionality
3. Explain why this enhancement would be useful
4. List any alternative solutions you've considered

**🔧 Pull Requests**

1. **Fork the repository** and create your branch from develop
2. **Write clear commit messages** following conventional commits:
   * feat: New feature
   * fix: Bug fix
   * docs: Documentation changes
   * style: Code style changes
   * refactor: Code refactoring
   * test: Test additions or changes
   * chore: Build process or auxiliary tool changes
3. **Follow the coding standards:**
4. # Good
5. def calculate\_facility\_usage(facility\_id: str, date: datetime) -> float:
6. """Calculate facility usage percentage for a given date."""
7. pass
8. # Bad
9. def calc(f, d):
10. pass
11. **Write tests** for your changes:
12. def test\_calculate\_facility\_usage():
13. result = calculate\_facility\_usage("facility-1", datetime.now())
14. assert 0 <= result <= 100
15. **Update documentation** if needed
16. **Run the test suite:**
17. # Run all tests
18. pytest
19. # Run with coverage
20. pytest --cov=.
21. # Run specific test
22. pytest tests/test\_security.py
23. **Format your code:**
24. # Format with Black
25. black .
26. # Check with Flake8
27. flake8 .
28. # Type checking
29. mypy sportai\_main\_app\_file.py
30. **Submit the pull request** with:
    * Clear description of changes
    * Link to related issue (if applicable)
    * Screenshots for UI changes
    * Confirmation that tests pass

**Development Setup**

**Prerequisites**

* Python 3.8+
* Git
* Docker (optional)

**Local Development**

# Clone your fork

git clone https://github.com/yourusername/sportai-suite.git

cd sportai-suite

# Create virtual environment

python -m venv venv

source venv/bin/activate # Windows: venv\Scripts\activate

# Install in development mode

pip install -r requirements.txt

pip install -r requirements-dev.txt

# Create feature branch

git checkout -b feature/your-feature-name

# Make changes and test

streamlit run sportai\_main\_app\_file.py

# Run tests

pytest

# Commit changes

git add .

git commit -m "feat: add awesome feature"

# Push to your fork

git push origin feature/your-feature-name

**Project Structure**

sportai-suite/

├── sportai\_main\_app\_file.py # Main application (be careful!)

├── ai\_modules/ # AI components

│ └── \*.py # Add new AI modules here

├── modules/ # Feature modules

│ ├── facility\_management/ # Facility-related features

│ ├── membership\_management/ # Member features

│ └── ... # Other feature categories

├── tests/ # Test files

│ └── test\_\*.py # Test files must start with test\_

└── docs/ # Documentation

**Adding New Modules**

To add a new module:

1. Create module file in appropriate directory:

# modules/facility\_management/my\_new\_feature.py

def run():

"""Entry point for the module."""

import streamlit as st

st.header("My New Feature")

# Your implementation here

1. Add to module loader in main file (carefully!)
2. Write tests:

# tests/test\_my\_new\_feature.py

def test\_my\_new\_feature():

from modules.facility\_management import my\_new\_feature

assert hasattr(my\_new\_feature, 'run')

**Git Workflow**

We use Git Flow:

* main - Production-ready code
* develop - Development branch
* feature/\* - Feature branches
* hotfix/\* - Urgent fixes
* release/\* - Release preparation

**Review Process**

All submissions require review:

1. Automated tests must pass
2. Code review by maintainer
3. Documentation updated
4. No merge conflicts

**Community**

* **Discord**: [Join our community](https://discord.gg/sportai)
* **Forum**: [GitHub Discussions](https://github.com/yourusername/sportai-suite/discussions)
* **Twitter**: [@SportAISuite](https://twitter.com/SportAISuite)

**Recognition**

Contributors will be recognized in:

* README.md contributors section
* Release notes
* Annual contributor spotlight

**Questions?**

Feel free to:

* Open an issue for questions
* Join our Discord server
* Email: contributors@sportai.com

Thank you for making SportAI Suite better! 🏆