Coverity Connect MCP Server - 完成化プラン

🎯 現在の状況

✅ 完了済み

- メインサーバー実装 (`coverity_mcp_server.py`)
- 包括的なドキュメント
- pyproject.toml設定
- requirements.txt

📦 Phase 1: プロジェクト構造の完成

1.1 標準的なPythonパッケージ構造を作成

```
coverity_connect_mcp/
├── src/
│   └── coverity_mcp_server/
│       ├── __init__.py
│       ├── main.py (メインサーバー)
│       ├── config.py (設定管理)
│       ├── tools/ (MCP tools)
│       └── utils/ (ユーティリティ)
├── tests/
├── docs/
├── examples/
├── pyproject.toml
├── README.md
├── LICENSE
├── CHANGELOG.md
└── .github/workflows/
```

1.2 必要なファイル作成

- [ ] `src/coverity_mcp_server/__init__.py`
- [ ] `src/coverity_mcp_server/main.py` (現在のPythonファイルをリファクタリング)
- [ ] `src/coverity_mcp_server/config.py`
- [ ] `LICENSE` ファイル
- [ ] `CHANGELOG.md`
- [ ] `.gitignore`
- [ ] GitHub Actions workflow

🔧 Phase 2: コード品質向上

2.1 メインサーバーのリファクタリング

- [ ] モジュール分割 (tools, config, utils)
- [ ] 型アノテーション完善
- [ ] エラーハンドリング強化
- [ ] ログ機能追加

2.2 テスト実装

- [ ] 単体テスト (pytest)
- [ ] モックテスト (Coverity API)
- [ ] 統合テスト
- [ ] カバレッジ90%以上

🚢 Phase 3: デプロイメント準備

3.1 Docker化

- [ ] `Dockerfile`
- [ ] `docker-compose.yml`
- [ ] Multi-stage build
- [ ] セキュリティベストプラクティス

3.2 CI/CD パイプライン

- [ ] GitHub Actions
- [ ] テスト自動化
- [ ] コード品質チェック
- [ ] セキュリティスキャン
- [ ] PyPI自動デプロイ

📚 Phase 4: ドキュメント完成

4.1 詳細ドキュメント

- [ ] Installation Guide
- [ ] Configuration Reference
- [ ] API Documentation
- [ ] Troubleshooting Guide

4.2 Examples & Tutorials

- [ ] Basic usage examples
- [ ] Advanced workflows
- [ ] Integration examples

🚀 Phase 5: 公開準備

5.1 PyPI発行

- [ ] Test PyPI でテスト
- [ ] 本番PyPI発行
- [ ] バージョン管理戦略

5.2 GitHub リポジトリ

- [ ] リポジトリ作成
- [ ] Issue/PR テンプレート
- [ ] セキュリティポリシー
- [ ] Contributing guide

💡 推奨される次のアクション

1. **まず Project Structure を整理**
   - 適切なディレクトリ構造作成
   - メインファイルの分割
2. **次に Test & Quality**
   - 基本的なテスト実装
   - CI/CD パイプライン設定
3. **最後に Deployment**
   - Docker化
   - PyPI発行

どのフェーズから始めたいか教えてください！