

# Enhanced Soldier Report System - Deployment Guide

## Prerequisites

### System Requirements

- **Operating System:** Windows 10+, macOS 10.14+, or Linux (Ubuntu 18.04+)
- **Python:** 3.8 or higher
- **Memory:** 4GB RAM minimum, 8GB recommended
- **Storage:** 1GB free space minimum
- **Display:** GUI requires display/X11 forwarding for remote deployments

### Python Dependencies

```
bash

pandas>=1.5.0
numpy>=1.20.0
pytest>=7.0.0
pytest-cov>=4.0.0
```

## Installation Methods

### Method 1: Standard Installation

```
bash

# Clone or extract the project
cd enhanced_soldier_report_system_refactored

# Run installation script
chmod +x deployment/scripts/install.sh
./deployment/scripts/install.sh
```

### Method 2: Manual Installation

```
bash
```

```
# Create virtual environment
python3 -m venv venv
source venv/bin/activate # On Windows: venv\Scripts\activate

# Install dependencies
pip install -r requirements.txt

# Create directories
mkdir -p logs reports config

# Copy configuration
cp config/default_settings.json config/settings.json
```

## Method 3: Docker Deployment

```
bash

# Build Docker image
docker build -t soldier-report-system deployment/docker/

# Run container (requires X11 forwarding for GUI)
docker run -it \
  -v /tmp/.X11-unix:/tmp/.X11-unix \
  -e DISPLAY=$DISPLAY \
  -v $(pwd)/reports:/app/reports \
  soldier-report-system
```

## Configuration

### Configuration File Structure

The system uses JSON configuration files located in `config/`:

```
json
```

```
{
  "column_mapping": {
    "callsign": "Callsign",
    "heartrate": "Heart_Rate"
  },
  "performance": {
    "activity_weight": 0.3,
    "casualty_penalties": {
      "WOUNDED": 10,
      "KIA": 20
    }
  },
  "reporting": {
    "output_directory": "reports/enhanced",
    "max_concurrent_reports": 4
  }
}
```

## Environment Variables

- `PYTHONPATH`: Set to `/path/to/src` for proper imports
- `LOG_LEVEL`: DEBUG, INFO, WARNING, ERROR (default: INFO)
- `CONFIG_PATH`: Path to configuration file (default: config/default\_settings.json)

## Running the Application

### Command Line

```
bash

# Activate virtual environment
source venv/bin/activate

# Start application
python src/main.py

# With custom config
CONFIG_PATH=config/custom_settings.json python src/main.py
```

## Testing

```
bash
```

```
# Run all tests
./deployment/scripts/run_tests.sh

# Run specific test categories
python -m pytest tests/unit/ -v
python -m pytest tests/integration/ -v

# With coverage
python -m pytest tests/ -v --cov=src --cov-report=html
```

## Architecture Overview

### Event-Driven Design

- **Event Bus:** Central communication hub
- **Services:** Independent, loosely-coupled components
- **GUI:** Reactive interface responding to events
- **Configuration:** External, runtime-configurable settings

### Key Components

1. **Data Loader:** CSV file processing and validation
2. **Analysis Engine:** Coordinates statistical analysis
3. **Safety Analyzer:** Medical and safety assessments
4. **Performance Scorer:** Configurable scoring algorithms
5. **Report Generator:** HTML report creation
6. **GUI:** Tkinter-based user interface

## Troubleshooting

### Common Issues

#### GUI Not Starting

```
bash
```

```
# Check display
```

```
echo $DISPLAY
```

```
# Install tkinter (Linux)
```

```
sudo apt-get install python3-tk
```

```
# For remote sessions
```

```
ssh -X user@host
```

## Import Errors

```
bash
```

```
# Set Python path
```

```
export PYTHONPATH=/path/to/enhanced_soldier_report_system_refactored/src
```

```
# Or add to .bashrc
```

```
echo 'export PYTHONPATH=/path/to/project/src' >> ~/.bashrc
```

## Permission Issues

```
bash
```

```
# Make scripts executable
```

```
chmod +x deployment/scripts/*.sh
```

```
# Fix file permissions
```

```
chmod -R 755 src/
```

## Log Files

- **Application logs:** logs/soldier\_report\_YYYYMMDD.log
- **Error logs:** logs/errors\_YYYYMMDD.log
- **Test logs:** test\_reports/

## Performance Tuning

- **Event Bus Workers:** Adjust max\_workers in configuration
- **Memory Usage:** Set chunk\_size for large datasets
- **Concurrent Reports:** Tune max\_concurrent\_reports

## Maintenance

## Regular Tasks

- **Log Rotation:** Logs rotate automatically at 10MB
- **Test Execution:** Run tests before deployments
- **Configuration Backup:** Version control settings files
- **Performance Monitoring:** Review event bus statistics

## Updates

1. Backup current configuration
2. Update source code
3. Run migration scripts if needed
4. Execute full test suite
5. Deploy with monitoring

## Security Considerations

### File Permissions

- Configuration files: 640 (owner read/write, group read)
- Log files: 644 (owner read/write, others read)
- Scripts: 755 (executable)

### Data Protection

- Reports contain sensitive military data
- Implement access controls as needed
- Consider encryption for sensitive deployments
- Audit log access and modifications

## Support

### Getting Help

1. Check log files for error details
2. Run diagnostics: `python src/main.py --debug`
3. Review test output for component issues
4. Consult architecture documentation

## Development

- **Adding Features:** Follow event-driven patterns
- **New Analysis Types:** Extend analysis engine
- **Custom Reports:** Create new report templates
- **UI Components:** Add to gui/components/