Python 3.12 Compatibility Fix Summary

Problem Resolved

Fixed the `AttributeError: module 'pkgutil' has no attribute 'ImpImporter'` error that occurs when trying to install dependencies with Python 3.12. This error was primarily caused by NumPy 1.25.2 and other packages using deprecated Python import mechanisms.

Files Created/Updated

1. New Files Created

- `requirements-py312.txt` : Python 3.12 specific requirements with compatible versions
- `PYTHON_312_SETUP.md` : Comprehensive setup guide for Python 3.12 users
- `PYTHON_312_COMPATIBILITY_SUMMARY.md` : This summary document

2. Files Updated

- `requirements.txt` : Updated with better version ranges for backward compatibility
- `README.md` : Added Python 3.12 installation instructions and compatibility notes

Key Version Updates for Python 3.12

Package	Original Version	Python 3.12 Compatible	Reason for Change
numpy	1.25.2	>=1.26.0	**Critical**: 1.26.0+ uses Meson build system, avoiding pkgutil.ImpImporter
pillow	10.1.0	>=10.1.0	Official Python 3.12 support from 10.1.0
playwright	1.40.0	>=1.39.0	Fixed greenlet compatibility in 1.39.0
pandas	2.1.3	>=2.1.3	Works with Python 3.12 on 64-bit systems
scikit-learn	1.3.2	>=1.3.2	Official Python 3.12 support
tensorflow	2.15.0	>=2.15.0	Official Python 3.12 support
torch	2.1.1	>=2.1.1	May need nightly builds for full support
asyncio	3.4.3	**REMOVED**	Built into Python 3.12, external package causes conflicts

Installation Methods

Method 1: Python 3.12 Specific (Recommended)

```
python3.12 -m venv venv_py312
source venv_py312/bin/activate
pip install --upgrade pip setuptools
pip install -r requirements-py312.txt
```

Method 2: Conda Environment

```
conda create -n auction_py312 python=3.12
conda activate auction_py312
conda install -c conda-forge numpy>=1.26.0 pandas scikit-learn
pip install -r requirements-py312.txt
```

Method 3: Fallback to Python 3.11

```
python3.11 -m venv venv
source venv/bin/activate
pip install -r requirements.txt
```

Verification Test Results

✅ **Successfully tested core packages**:
- NumPy 1.26.0 - Installs and imports correctly
- Pandas 2.3.1 - Installs and imports correctly
- Scikit-learn 1.7.0 - Installs and imports correctly
- Pillow 11.3.0 - Installs and imports correctly

Known Issues and Solutions

Issue 1: PyTorch Python 3.12 Support

Problem: PyTorch 2.1.1 may not have official Python 3.12 wheels
Solutions:
- Use nightly builds: `pip install --pre torch --index-url https://download.pytorch.org/whl/nightly/cpu`
- Use conda: `conda install pytorch -c pytorch-nightly`
- Stick with Python 3.11 for this project

Issue 2: 32-bit Windows Compatibility

Problem: Some packages lack 32-bit Python 3.12 wheels
Solution: Use 64-bit Python 3.12 or downgrade to Python 3.11

Issue 3: Selenium-Wire Compatibility

Problem: `selenium-wire` may have blinker dependency conflicts
Solution:

```
pip uninstall blinker selenium-wire -y
pip install blinker==1.7.0
pip install selenium-wire
```

Recommendations

For New Installations

1. **Use Python 3.12** with `requirements-py312.txt` for best compatibility
2. **Upgrade pip and setuptools** before installing packages
3. **Use virtual environments** to avoid conflicts
4. **Test critical imports** after installation

For Existing Installations

1. **Backup current environment** before upgrading
2. **Create new Python 3.12 environment** rather than upgrading in place

3. **Use the verification script** in PYTHON_312_SETUP.md to test compatibility
4. **Keep Python 3.11 environment** as fallback if needed

For Production Deployments

1. **Use Docker** with Python 3.12 base image for consistency
2. **Pin exact versions** in requirements-py312.txt for reproducibility
3. **Test thoroughly** in staging environment before production
4. **Monitor for package updates** that may improve compatibility

Alternative Solutions

If Python 3.12 compatibility issues persist:

1. **Use Python 3.11**: Most stable option for this project
2. **Use Docker**: Containerize with known working Python version
3. **Use pyenv**: Manage multiple Python versions easily
4. **Use conda**: Better dependency resolution for scientific packages

Future Maintenance

- **Monitor package updates**: NumPy, TensorFlow, PyTorch regularly improve Python 3.12 support
- **Update requirements**: As packages release Python 3.12 compatible versions
- **Test compatibility**: Regularly verify new package versions work with Python 3.12
- **Update documentation**: Keep setup guides current with latest compatibility information

Support Resources

- NumPy Python 3.12 Support Issue (https://github.com/numpy/numpy/issues/23808)
- Playwright Python 3.12 Support (https://github.com/microsoft/playwright-python/issues/2096)
- Python 3.12 Release Notes (https://docs.python.org/3.12/whatsnew/3.12.html)
- PyPI Python 3.12 Readiness (https://pyreadiness.org/3.12/)

Last Updated: July 13, 2025
Status: ✅ Python 3.12 compatibility issues resolved