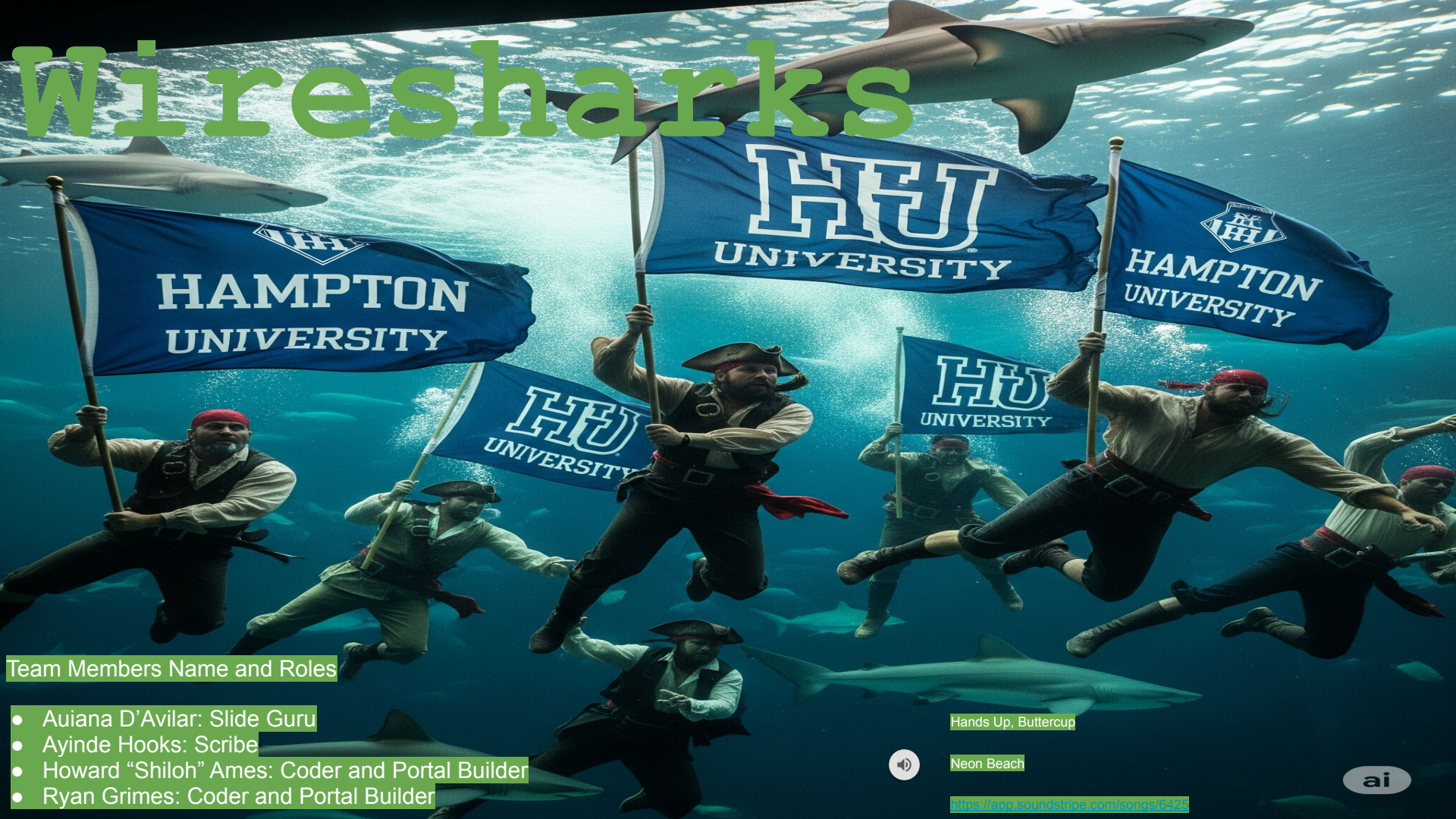


Wiresharks



Team Members Name and Roles

- Auiana D'Avilar: Slide Guru
- Ayinde Hooks: Scribe
- Howard "Shiloh" Ames: Coder and Portal Builder
- Ryan Grimes: Coder and Portal Builder

Hands Up, Buttercup

Neon Beach

<https://app.soundstripe.com/songs/R425>



Meet the Sharks

Ayinde Hooks

Ryan Grimes

Hampton University
Howard "Shiloh" Ames

Auiana D'Avilar

Scribe

Coder and Portal
Builder

Coder and
Portal Builder

Slide Guru

Why Are We Doing This?



Goal:

Evaluate how easy it is to reproduce existing papers from ICSE with Code Repos Artifacts Selection .

Key Questions We Ask:

- Did the repo include all necessary code?
- Was the dataset public and accessible?
- Were instructions up to date?
- Any issues with dependencies or hardware?
- How close were our results to theirs?

Score Card

Paper Availability	Availability of Code and Software	Availability of Datasets	Computer Requirements	GPU Requirements	Documentation Quality	Ease of Setup	Reproducibility of Results	Rating
<p>1: Unavailable/Impossible to find.</p> <p>2: Paywalled/Very hard to access.</p> <p>3: Available, but via obscure link.</p> <p>4: Open-access, direct link.</p> <p>5: Open-access, easily searchable.</p>	<p>1: No code/Private repository.</p> <p>2: Code available, but major parts missing/broken.</p> <p>3: Code available, incomplete/needs big fixes.</p> <p>4: Code available, mostly complete, minor issues.</p> <p>5: Code fully available, complete.</p>	<p>1: dataset missing.</p> <p>2: Mentioned, but completely inaccessible.</p> <p>3: Available, but very hard to find/access.</p> <p>4: Available, but metadata is poor/incomplete</p> <p>5: Fully accessible with complete metadata.</p>	<p>1: No info/Impossible to meet.</p> <p>2: Vague/Requires rare hardware.</p> <p>3: Specific, but hard to meet.</p> <p>4: Specific, but common hardware.</p> <p>5: Clear, common, and flexible hardware.</p>	<p>1: No info/Mandatory custom GPU.</p> <p>2: GPU required, vague/high-end specs.</p> <p>3: GPU optional, specific specs.</p> <p>4: GPU optional, common specs.</p> <p>5: No GPU required (CPU-only).</p>	<p>1: None or misleading.</p> <p>2: Very poor</p> <p>3: Basic or needs much interpretation.</p> <p>4: Clear but few details.</p> <p>5: Comprehensive & clear.</p>	<p>1: Cannot be run due to critical issues or missing parts.</p> <p>2: Can't run without major problems; needs expert help or significant workarounds.</p> <p>3: Can be run with some effort; requires troubleshooting or minor fixes.</p> <p>4: Runs well with minimal effort; minor adjustments might be needed.</p> <p>5: Runs perfectly by simply following the instructions; no issues.</p>	<p>1: Cannot be run due to critical issues or missing parts.</p> <p>2: Can't run without major problems; needs expert help or significant workarounds.</p> <p>3: Can be run with some effort; requires troubleshooting or minor fixes.</p> <p>4: Runs well with minimal effort; minor adjustments might be needed.</p> <p>5: Runs perfectly by simply following the instructions; no issues.</p>	<p>1 (Impossible): Cannot be run due to critical issues or missing parts.</p> <p>2 (Very Difficult): Can't run without major problems; needs expert help or significant workarounds.</p> <p>3 (Doable): Can be run with some effort; requires troubleshooting or minor fixes.</p> <p>4 (Mostly Smooth): Runs well with minimal effort; minor adjustments might be needed.</p> <p>5 (Plug and Play): Runs perfectly by simply following the instructions; no issues.</p>

Project 1: Business Flow Tampering (BFT) Detector

What it does:

Automatically finds ways to bypass website paywalls and ads.

Our Experience:

- Code was available, but instructions were very outdated.
- Major Python version and software conflicts prevented it from running.
- Hardware issues (Docker/virtualization) were a dead end.

Reproducibility Rating: 2/5 (Very Difficult)

Couldn't run the project to compare our results.

The background of the slide is a surreal underwater scene. Several pirates, dressed in traditional pirate attire like hats and boots, are swimming in the water. They are holding up blue and white flags that feature the Hampton University logo, which consists of the letters 'HU' in a stylized font. The pirates are also holding wooden oars. Surrounding the pirates are several large, realistic-looking sharks, some of which are swimming towards the viewer. The water is a deep blue-green color, and there are bubbles visible throughout the scene.

Project 2: Fairify

What it does:

Formally checks AI models for fairness (ensures similar people get similar outcomes).

Our Experience:

- Paper, Code, and Data were all available.
- Python version conflicts (needed older TensorFlow) required complex setup.
- **Critical Problem:** Script ran for hours, but NEVER produced any output files.

Reproducibility Rating: 3/5 (Doable)

We could run the project, but without results, we couldn't verify its claims.

The background of the slide is a surreal underwater scene. Several large sharks are swimming in the water. In the center, a group of pirates, dressed in traditional pirate attire like hats and vests, are holding up several blue and white flags. The flags feature the letters 'HU' and the words 'HAMPTON UNIVERSITY'. The scene is lit with a blue, watery glow, creating a mysterious and somewhat ominous atmosphere.

Project 3: Bad Snakes

What it does:

Evaluates tools designed to find malicious Python packages on PyPI.

Our Experience:

- Paper and Code were open-access and easily found.
- Python version and dependency conflicts also caused setup issues.
- **Major Problem:** The actual original datasets used in the paper were NOT available due to privacy.

Reproducibility Rating: 2/5 (Very Difficult)

Missing crucial data prevented us from verifying their quantitative results.

The background of the slide is a surreal underwater scene. Several large sharks are swimming in the water. In the center, a group of pirates, dressed in traditional pirate attire like hats and vests, are swimming and holding up blue and white flags. The flags feature the 'HU' logo of Hampton University. The scene is lit with a blueish-green light, creating a deep-sea atmosphere.

Common Challenges

- **Outdated Instructions:** Frequent source of setup failures.
- **Python Versioning:** Newer Python often breaks older project dependencies.
- **Missing Datasets:** A significant barrier to full reproduction.
- **Environment Specificity:** Unexpected hardware or system settings blocking progress.
- **Lack of Clear Output:** Scripts running but no results appearing.



Technology Used

Programming Languages: Python, JavaScript (for web tools)

Core Libraries/Frameworks: TensorFlow/Keras, Numpy, Z3, Poetry, Bandit, scikit-image (attempted)

Automation/Virtualization: Google Puppeteer, Docker (attempted for isolation), pyenv (for Python version management)

Environment: Apple Terminal, Kali Linux (WSL), VS Code, Github, and Docker

The background of the slide is a surreal underwater scene. Several pirates, dressed in traditional pirate attire like hats and vests, are swimming in the water. They are holding up various flags, some of which feature the Hampton University logo (a stylized 'HU' or 'H' with 'HAMPTON UNIVERSITY' text). Large, realistic-looking sharks are swimming around the pirates, creating a sense of danger and chaos. The scene is set in deep blue water with light rays filtering down from above.

Key Takeaways

Reproducing scientific software can be extremely challenging.

For better reproducibility, projects need:

- Clear, up-to-date documentation.
- Accessible datasets (or clear alternatives).
- Robust environment setup instructions (e.g., using Docker or explicit versioning).