HACKSTREET BOYS

YARI PETTIS ZION PEASE DAVE BROWN EJAY AGUIRRE JULIAN TOLBERT

SGX3 CODING
INSTITUTE

Future Design
penguinmusi
https://pixabay.com/music/future-bas

Project Overview & Team Roles

Julian – GitHub & Documentation Lead

Sets up repo, manages README.md, folder structure, and code organization.

Ejay – Poster & Presentation Lead

Designs project poster and final slide deck; supports portal content & layout. Will be working on Flask as well.

Zion – Paper Analyst

Selects target papers, evaluates reproducibility criteria, leads scorecard writing.

Dave – Code Runner

Attempts to reproduce paper results, logs code, dataset, and hardware issues

Yari – Communications & Submission Manager

Coordinates daily check-ins, manages submission proof, team info, and final review.

5 Day Plan

Days	Focus	Key Outputs
Monday	Kickoff & Setup	Repo setup, roles assigned, paper shortlist
Tuesday	Paper Deep Dive & Planning	Paper selected, access tested, plan slides
Wednesday	Scorecard Development & Testing	JSON/CSV file, initial portal layout, graph
Thursday	Portal Build & Poster Finalization	Site live, poster PDF, submission proof
Friday	Final Presentation & Deliverables Wrap-up	Slides PDF, final push to GitHub, rehearsal

HackStreet Boys: Project Progress

Priorities:

Our current priority is creating the datasets from the papers and a proper rubric.

We have decided to use a python script to read the scores in so that we can use the data to plot that information.

While also starting to build our website/github to host our information and the actual score cards that we are building.

HackStreet Boys

Project Plan

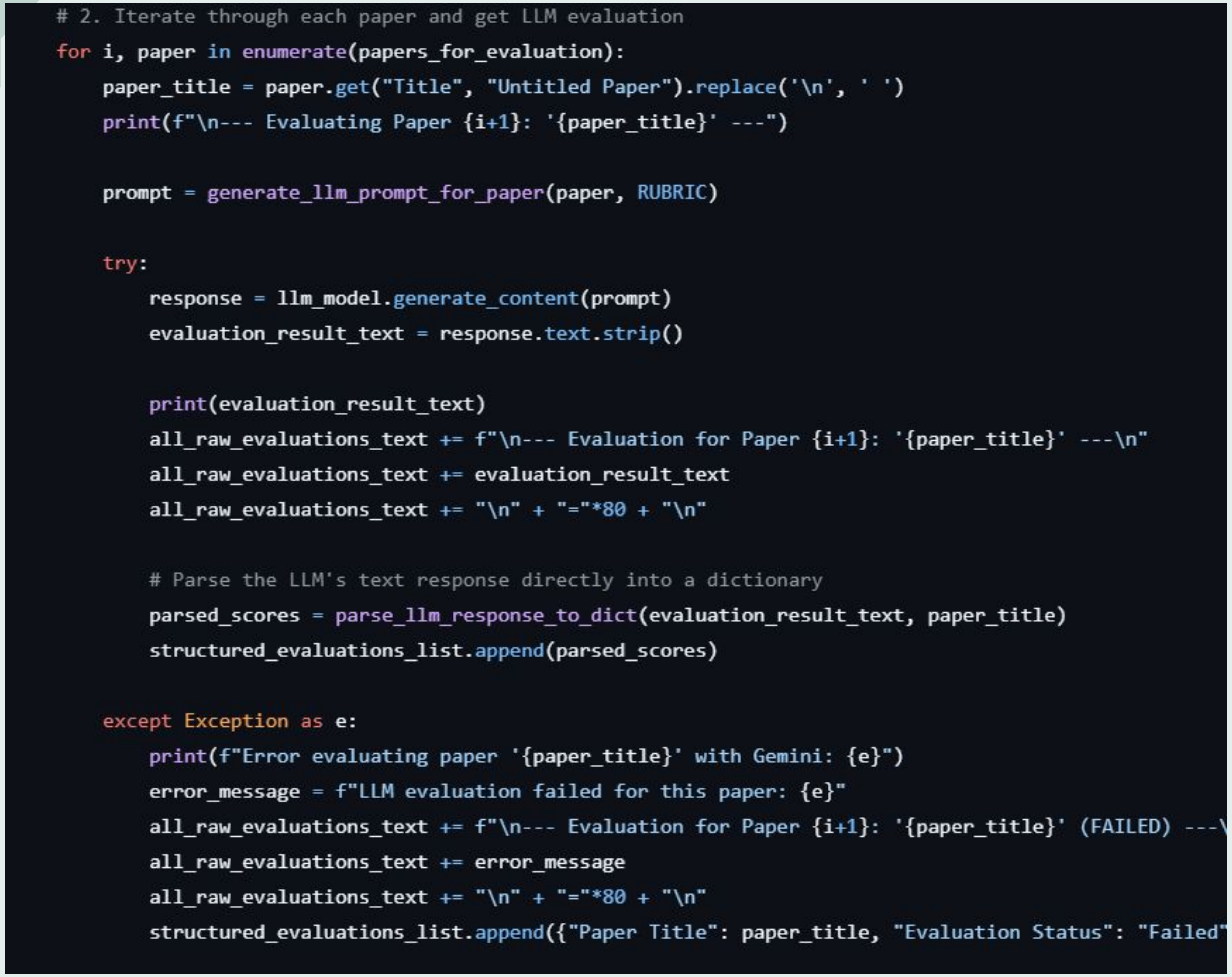
Update!

The project plan before is still in progress but we have started to move somethings to be done in tandem with other goals, like the beginning to code our flask and properly setting up the information held in papers/websites. While also including more flexibility due to time zones and some skill levels.

Technology/Problems

We have been using AI like manus, chatgpt and gemini prompting it to give us ideas for rubric and helping debug our code. While also applying our skills in virtual environments like colab to test out our python code.

We created a python script that scrapes a specified number papers, feeds them into Gemini AI, which then provides feedback on some scoring metrics and stores them in a csv file.



```
# 2. Iterate through each paper and get LLM evaluation
for i, paper in enumerate(papers_for_evaluation):
    paper_title = paper.get("Title", "Untitled Paper").replace('\n', ' ')
    print(f"\n--- Evaluating Paper {i+1}: '{paper_title}' ---")

    prompt = generate_llm_prompt_for_paper(paper, RUBRIC)

    try:
        response = llm_model.generate_content(prompt)
        evaluation_result_text = response.text.strip()

        print(evaluation_result_text)
        all_raw_evaluations_text += f"\n--- Evaluation for Paper {i+1}: '{paper_title}' ---\n"
        all_raw_evaluations_text += evaluation_result_text
        all_raw_evaluations_text += "\n" + "="*80 + "\n"

        # Parse the LLM's text response directly into a dictionary
        parsed_scores = parse_llm_response_to_dict(evaluation_result_text, paper_title)
        structured_evaluations_list.append(parsed_scores)

    except Exception as e:
        print(f"Error evaluating paper '{paper_title}' with Gemini: {e}")
        error_message = f"LLM evaluation failed for this paper: {e}"
        all_raw_evaluations_text += f"\n--- Evaluation for Paper {i+1}: '{paper_title}' (FAILED) ---\
        all_raw_evaluations_text += error_message
        all_raw_evaluations_text += "\n" + "="*80 + "\n"
        structured_evaluations_list.append({"Paper Title": paper_title, "Evaluation Status": "Failed"
```

HackStreet Boys

We were able to create a python script to evaluate papers and put them into a dataframe to be graphed. We also have started to get our flask app up and running including our team/background information

HackStreet Boys

Rubric:

This rubric assesses scientific presentations based on **six key criteria**, each scored from 1 (Poor) to 4 (Excellent):

1. **Accessibility**
   Evaluates whether the resource is open and free or requires payment/login.

2. **Computer Requirements**
   Measures how reasonable and clearly specified the hardware/software needs are.

3. **Reproducibility of Results**
   Assesses how easily others can reproduce the project using provided steps and data.

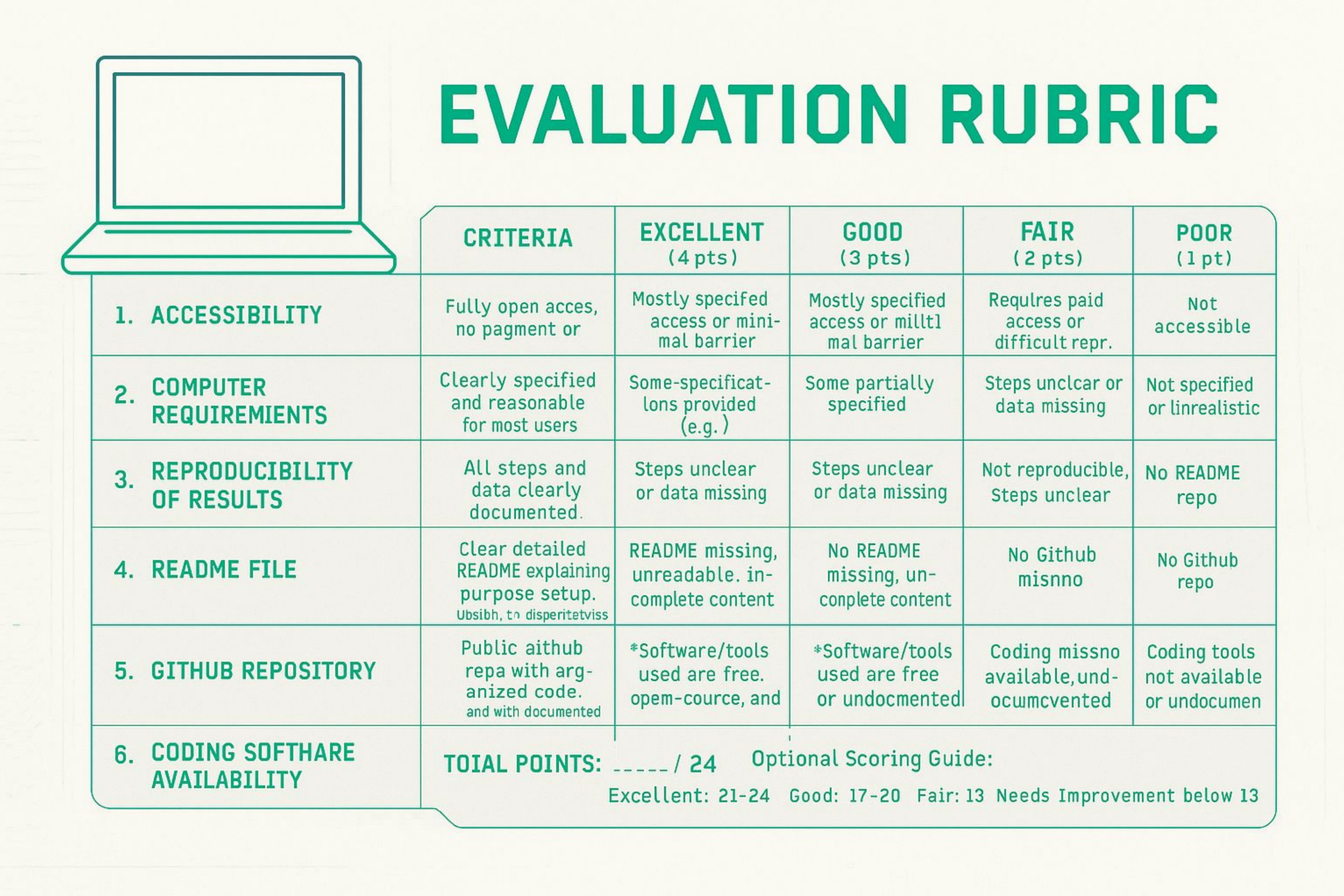
4. **README File**
   Looks at the presence and quality of documentation explaining the project's purpose, setup, and usage.

5. **GitHub Repository**
   Checks for an organized, public repo with code and version control.

6. **Coding Software Availability**
   Rates how accessible and well-documented the required coding tools are.

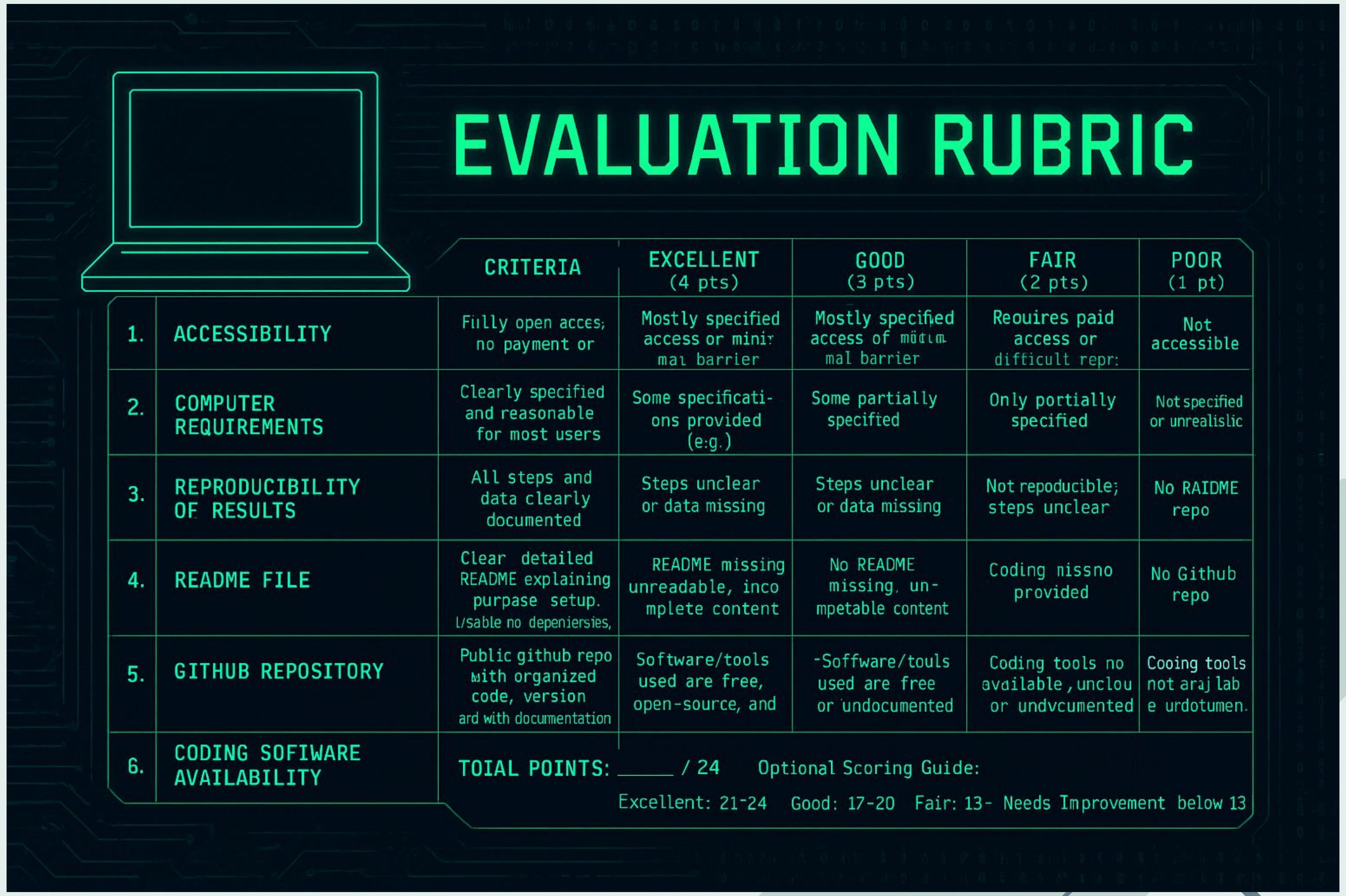
Total Score: /24

- **Excellent**: 21–24
- **Good**: 17–20
- **Fair**: 13–16
- **Needs Improvement**: Below 13



EVALUATION RUBRIC

	CRITERIA	EXCELLENT (4 pts)	GOOD (3 pts)	FAIR (2 pts)	POOR (1 pt)
1. ACCESSIBILITY	Fully open acces, no pagment or	Mostly specifed access or minimal barrier	Mostly specified access or milltl mal barrier	Requlres paid access or difficult repr.	Not accessible
2. COMPUTER REQUIREMIENTS	Clearly specified and reasonable for most users	Some-specificatlons provided (e.g.)	Some partially specified	Steps unclcar or data missing	Not specified or linrealistic
3. REPRODUCIBILITY OF RESULTS	All steps and data clearly documented.	Steps unclear or data missing	Steps unclear or data missing	Not reproducible, Steps unclear	No README repo
4. README FILE	Clear detailed README explaining purpose setup. Ubsibh, tn disperitetviss	README missing, unreadable. incomplete content	No README missing, unconplete content	No Github misnno	No Github repo
5. GITHUB REPOSITORY	Public aithub repa with arganized code. and with documented	*Software/tools used are free. opem-cource, and	*Software/tools used are free or undocmented	Coding missno available, undocumcvented	Coding tools not available or undocumen
6. CODING SOFTHARE AVAILABILITY	TOIAL POINTS: _____ / 24 Optional Scoring Guide: Excellent: 21-24 Good: 17-20 Fair: 13 Needs Improvement below 13				



EVALUATION RUBRIC

	CRITERIA	EXCELLENT (4 pts)	GOOD (3 pts)	FAIR (2 pts)	POOR (1 pt)
1. ACCESSIBILITY	Filly open acces; no payment or	Mostly specified access or mini mal barrier	Mostly specified access of mödım mal barrier	Reouires paid access or difficult repr:	Not accessible
2. COMPUTER REQUIREMENTS	Clearly specified and reasonable for most users	Some specifications provided (e:g.)	Some partially specified	Only portially specified	Not specified or unrealislic
3. REPRODUCIBILITY OF RESULTS	All steps and data clearly documented	Steps unclear or data missing	Steps unclear or data missing	Not repoducible; steps unclear	No RAIDME repo
4. README FILE	Clear detailed README explaining purpase setup. l/sable no depeniersies,	README missing unreadable, incomplete content	No README missing, unmpetable content	Coding nissno provided	No Github repo
5. GITHUB REPOSITORY	Public github repo with organized code, version ard with documentation	Software/tools used are free, open-source, and	-Soffware/touls used are free or undocumented	Coding tools no available, unclou or undvcumented	Cooing tools not araj lab e urdotumen.
6. CODING SOFIWARE AVAILABILITY	TOIAL POINTS: _____ / 24 Optional Scoring Guide: Excellent: 21-24 Good: 17-20 Fair: 13- Needs Improvement below 13				

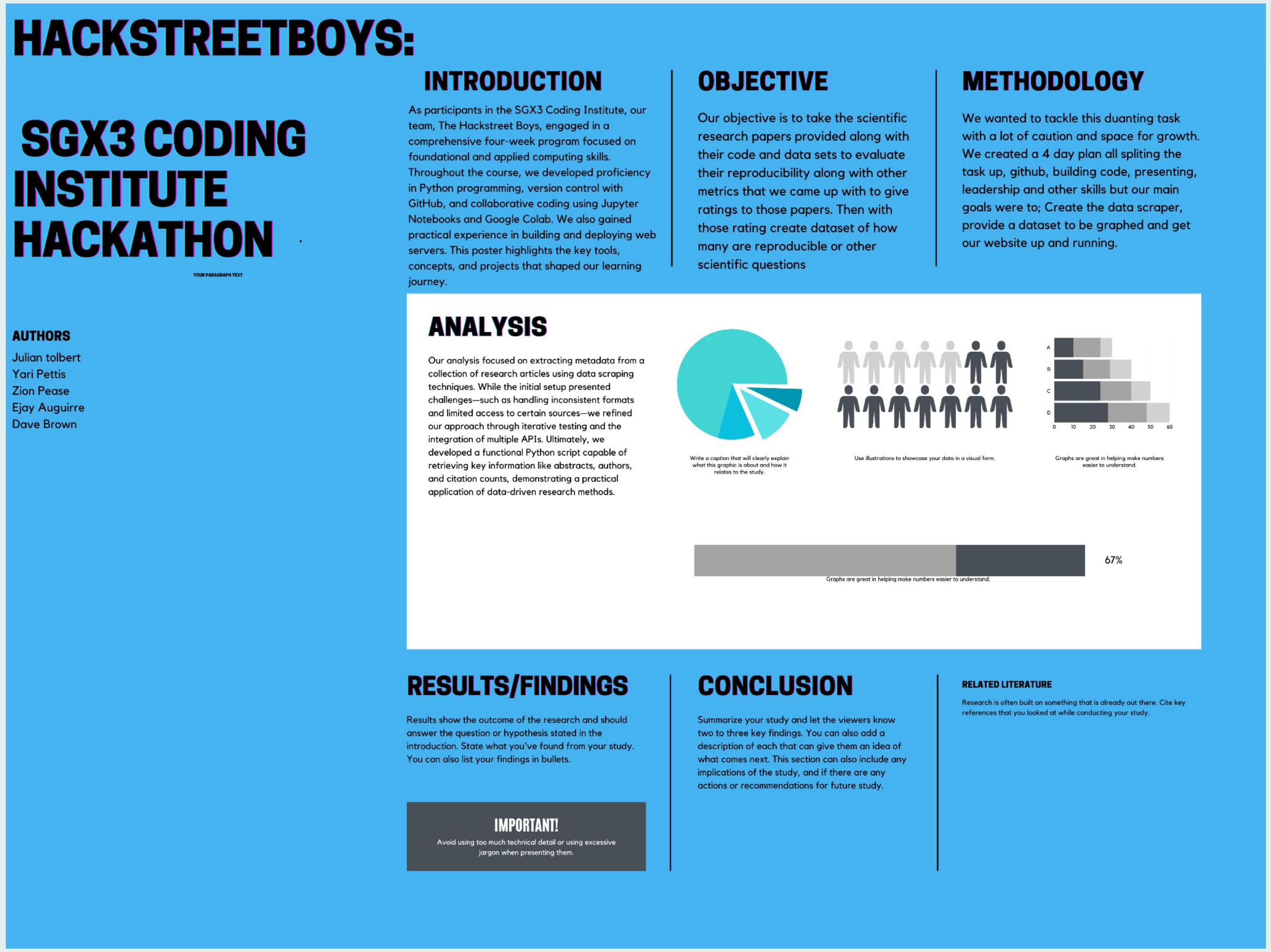
Website Demo

HackStreet Boys - Day 4: Update

We selected a design template that best fit our project goals and began editing our poster to customize it with the necessary information of our website, the data gleaned and visuals that match our website.

We also have been working on and completed the SC24 paper scrapping and have updated our code to read the pdf's from those papers since they were not directly on the SC24 website.

We have had some issues with using gemini's limited AI API and integrating our flask app onto the server



HACKSTREETBOYS:

SGX3 CODING INSTITUTE HACKATHON

YOUR PARAGRAPH TEXT

AUTHORS

Julian tolbert
Yari Pettis
Zion Pease
Ejay Auguirre
Dave Brown

INTRODUCTION

As participants in the SGX3 Coding Institute, our team, The Hackstreet Boys, engaged in a comprehensive four-week program focused on foundational and applied computing skills. Throughout the course, we developed proficiency in Python programming, version control with GitHub, and collaborative coding using Jupyter Notebooks and Google Colab. We also gained practical experience in building and deploying web servers. This poster highlights the key tools, concepts, and projects that shaped our learning journey.

OBJECTIVE

Our objective is to take the scientific research papers provided along with their code and data sets to evaluate their reproducibility along with other metrics that we came up with to give ratings to those papers. Then with those rating create dataset of how many are reproducible or other scientific questions

METHODOLOGY

We wanted to tackle this duanting task with a lot of caution and space for growth. We created a 4 day plan all spliting the task up, github, building code, presenting, leadership and other skills but our main goals were to; Create the data scraper, provide a dataset to be graphed and get our website up and running.

ANALYSIS

Our analysis focused on extracting metadata from a collection of research articles using data scraping techniques. While the initial setup presented challenges—such as handling inconsistent formats and limited access to certain sources—we refined our approach through iterative testing and the integration of multiple APIs. Ultimately, we developed a functional Python script capable of retrieving key information like abstracts, authors, and citation counts, demonstrating a practical application of data-driven research methods.

Write a caption that will clearly explain what this graphic is about and how it relates to the study.

Use illustrations to showcase your data in a visual form.

Graphs are great in helping make numbers easier to understand.

Graphs are great in helping make numbers easier to understand.

RESULTS/FINDINGS

Results show the outcome of the research and should answer the question or hypothesis stated in the introduction. State what you've found from your study. You can also list your findings in bullets.

IMPORTANT!

Avoid using too much technical detail or using excessive jargon when presenting them.

CONCLUSION

Summarize your study and let the viewers know two to three key findings. You can also add a description of each that can give them an idea of what comes next. This section can also include any implications of the study, and if there are any actions or recommendations for future study.

RELATED LITERATURE

Research is often built on something that is already out there. Cite key references that you looked at while conducting your study.