HACKHPC @ADMI26
JUNE 22-26, 2026

https://hackhpc.github.io/admi26

Team Project Plan

June 23, 2026 - Morning Session

ADMI THE ASSOCIATION OF COMPUTER SCIENCE DEPARTMENTS AT MINORITY INSTITUTIONS
NSF
SGX3 Extend. Expand. Exemplify.
TACC TEXAS ADVANCED COMPUTING CENTER
projectEUREKA!
HackHPC Hackathons and Codeathons

HACKHPC @ADMI26
JUNE 22-26, 2026

Goals For the Next Check-in

Your team needs to add at least two (2) slides to the shared deck with:

- **Project Theme and Goals**: The first slide must clearly define the team's chosen project theme and the goals they want to accomplish
- **Team Member Roles**: The plan must outline the roles each team member will play, specifically based on completing the required project deliverables
- **Proposed Technologies**: Teams need to list the technologies they are considering using for their project, such as Jupyter Notebooks or Google Colab
- **Project Milestones**: The plan must establish a minimum of seven project milestones, which act as primary challenges the team must meet to progress
- **Tasks per Milestone**: Each milestone must include a minimum of three specific tasks that need to be accomplished, such as defining data collection methods or finding a seed Wikipedia article
- **Include a Gantt chart of your project timeline**

Project Plan Guide

[Note: If your slide deck is made in Canva it must be exported to PPTX and then imported to the shared slide deck]

HACKHPC
@ADMI26
JUNE 22-26, 2026

Duplicate this slide to customize

Example Team Name

Team Theme Song (inserted on this slide from Insert menu so it will play)

https://hackhpc.github.io/admi26

HACKHPC @ADMI26
JUNE 22-26, 2026

Example Team Name

- Team Members Names
  - Member 1
  - Member 2
  - Member 3
  - Member 4
- Origin Tracking

Duplicate this slide to customize

Team Zoom Virtual Background

Team Theme Song Name and URL

HACKHPC @ADMI26
JUNE 22-26, 2026

Project Theme and Goals

Duplicate this slide to customize

https://hackhpc.github.io/admi26

HACKHPC @ADMI26
JUNE 22-26, 2026

Project Plan

1. Roles for Team Members based on project Deliverables
2. Minimum of seven (7) project milestones
3. Minimum of three (3) tasks per milestone
   a. Example: Seed Wikipedia article to begin the knowledge tree
   b. Example: How you will get the data (Web scrape for details)
   c. Example: Will you fine-tune a model or use RAG?
4. What technologies you are considering using?

Duplicate this slide to customize

https://hackhpc.github.io/admi26

The Hacking Tribunal

Future Event Prediction

HACKHPC
@ADMI26
JUNE 22-26, 2026

https://hackhpc.github.io/admi26

theHackingTribunal:~$

Future Event Prediction

Madison Conley
Computer Science, *MVSU*

Nadia Rapheal
Computer Science, *WSSU*

Josiah Johnson
Computer Science, *WSSU*

Usir Jackson
Computer Science, *Claflin*

THE HACKING TRIBUNAL
PREDICTING THE FUTURE THROUGH AI

Project Theme:

FutureVision: Predicting the Future of Biology through Knowledge Evolution

Biology has always been a critical field to **understanding and improving humankind.** As we evolve, so does the study of life, with studies macroscopic, microscopic, and everywhere in between. Being able to **predict the next-generation breakthroughs** can only happen when we understand the pillars upon it stands now, and the **foundational innovations it has gone through.**

Our theme in this hackathon is to **predict the future of biological studies** by **analyzing trends of innovation found in the past.** Through this process, we will create forecasted projections upon what we believe is the future of biology in developments, breakthroughs, and emerging fields.

Project Goals:

Develop an AI-assisted forecasting platform that uses encyclopedic knowledge of past trends in biology to produce forecasted emerging concepts and innovation opportunities.

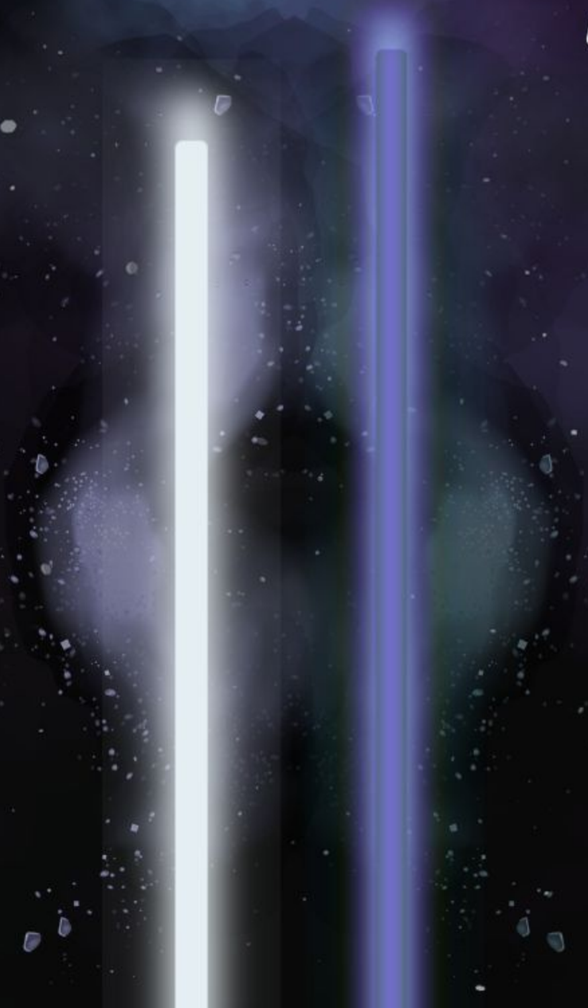
- Build a **biological knowledge graph** based on pages from Wikipedia and WikiData
- Identify **historical patterns** of scientific development (when they occurred, what they impacted, where it has lead)
- Detect **convergence** between biological disciplines (when they meet, at what point in history, and why)
- Generate evidence-based future **biological trend forecasts.**
- Demonstrate HPC techniques for **graph processing and analysis**
- Deliver **fully documented** deliverables

Roles:

Usir - Data Collection

Data + Knowledge Graph Lead

- identifying **strong articles** from Wikipedia and WikiData
- **extracting** biological concepts, categories, related articles, and WikiData properties
- converting information into **nodes** (content) and **edges** (connections)
- ensuring **high data quality** (no duplicates, good relationships)



Nadia - Historical Analysis

Graph Analytics & Prediction Lead

- identify important and growing concepts and **important nodes** for past and future
- detecting **convergence** at the intersection of large concepts
- creating **scoring systems** for breakthroughs and keywords
- discover which **trends and concepts** are relevant to the projection

Roles:

Madison - AI Assisted Interpretation

AI Research & Insight Lead

- **utilizing LLMs** to summarize graph findings and generate insight reports
- **verifying** historical trends and forecast explanations with **confidence levels**
- generating insights in **natural language** sentences with multiple data input

Josiah - Visualization, Project Management

Visualization Development, Project Manager

- building **interactive graphs**, timelines, and forecast displays
- creating the **demo interface** (website, dashboard, etc)
- **preparing** the readme, poster, presentation, and demo
- ensuring the **project is organized** and tasks are delegated

Project Milestones and Task Breakdown I:

[1] Project Foundation and Repository Setup

- Create **GitHub repository**, populate README
- Set up **Google Colab** notebooks
- Create a shared config cell in notebooks with Wikipedia APIs for all to use

[2] Biological Seed Dataset Creation

- Select 5-10 Wikipedia articles to use for **starting nodes**
- Use WikiAPI to pull text, wikilinks, and categories recursively to **depth 3**
- Use "instance of", "part of", and other **relations**
- Feed through **token parser** to produce concept-relation pairs

[3] Knowledge Graph Construction

- Represent each biological concept as a **graph node** with important metadata
- Create **directed edges** from wikilinks and WikiData semantic relationships, add weight
- Store completed graph in **NetworkX** (maybe in interactive graph?)

Project Milestones and Task Breakdown II:

[4] Historical Evolution Analysis

- Pull Wikipedia API's **"first-edit" timestamp** for birthdate
- Build a timeline dataset, for each node record **first year of appearance** and **expansion**
- Compute graph centrality metrics (degree, betweenness, closeness?) across **historical snapshots**
- Identify **historical** innovation pathways (i.e. DNA to PCR to CRISPR)

[5] Convergence Detection System

- Apply community detection to segment graph into biological **sub-field clusters**
- Track inter-cluster edge growth, when **cross-links** start accelerating then flag it
- **Rank convergence hotspots** by rate of cross-cluster edge addition, save best as key findings
- Prompt LLM with top-5 convergence pair subgraph to input for **natural language** output

[6] AI Forecasting Engine - RAG Approach

- Adopt **RAG** to use the knowledge graph and Wiki article text for corpus
- Design **forecasting prompt**, with subgraph of top-converging nodes and growth metrics
- Create new keyword list, **concepts under-represented** but predicted to grow anyways
- Produce **Future Trend Report**, top 10 predicted breakthroughs (thoughts, reasoning, score)

Project Milestones and Task Breakdown III:

[7] Visualization and User Experience

- Build an **interactive knowledge graph explorer,** with nodes filterable by sub field cluster, time period, and convergence score, with hover-card details on each concept
- Create a **dashboard** showing concept emergence
- Design a convergence **heatmap** and a radar chart of predicted breakthrough zones

[8] Final Submission Package

- Finish updating **complete README file** with all goals, system architecture, environment setup, and reproduction guide
- Design **project poster** covering important information as outlined
- **Complete commenting** and annotation in Colab notebooks
- **Practice demo** with live-visualization and key findings before submit

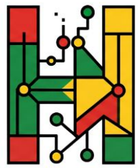
Technologies Used:

- Wikipedia API
- Python
- Pandas
- NetworkX
- HuggingFace
- JSON
- PyVis
- GitHub
- Google Colab
- Matplotlib

Gantt Chart Timeline

4-Day Hackathon Gantt Chart (Condensed)

Workstream	Owner	Day 1	Day 2	Day 3	Day 4
Project Setup & Architecture	Lead/Integration	■			
Data Collection & JSON Pipeline	Data Engineer	■	■		
Knowledge Graph Construction	Graph/HPC Engineer	■	■		
Historical Evolution Analysis	Forecasting Engineer	■	■		
Convergence Detection	Forecasting Engineer		■	■	
Forecasting Engine	Forecasting Engineer			■	
HPC Optimization & Benchmarking	Graph/HPC Engineer		■	■	
Visualization & Dashboard	Forecasting Engineer			■	■
Documentation & Repository	Lead/Integration	■	■	■	■
Poster, Demo & Presentation	Entire Team				■



HACKHPC
@ADMI26
JUNE 22-26, 2026

Origin Zero

BeatKitchen - Cold Streets (87 BPM) Rap Instrume

https://hackhpc.github.io/admi26

HACKHPC @ADMI26
JUNE 22-26, 2026

Origin Zero

Team Members Names

- Emmanuel Akwasi Opoku
- Prince Osei
- Anthony Nwafor
- Amelia Hunter

Origin Tracking

BeatKitchen - Cold Streets (87 BPM) Rap Instrume

https://hackhpc.github.io/admi26

THEME AND GOALS

Tracing the root causes of health disparities in underrepresented communities — through Wikipedia, one origin at a time.

THEME

Where did it start?

Health disparities aren't random — they have documented historical origins

Wikipedia is the knowledge source that connects the dots

We surface those connections through graph analysis

GOALS

What we're building

Live Wikipedia crawler that maps disparities to historical origins

PageRank algorithm to identify true root causes

Interactive visual origin chain with plain English narrative

TECHNOLOGIES

Our stack

Python	wikipedia-api
NetworkX	NumPy
Hugging Face	D3.js
HTML / CSS	GitHub Pages

ORIGIN ZERO ROLES

Each member owns one part of the pipeline — we build in parallel and hand off cleanly.

ANYHONY NWAFOR

Frontend & Visualization

- Builds the interactive origin chain display using D3.js
- Designs the search bar and user flow judges interact with
- Connects backend output to the browser in real time

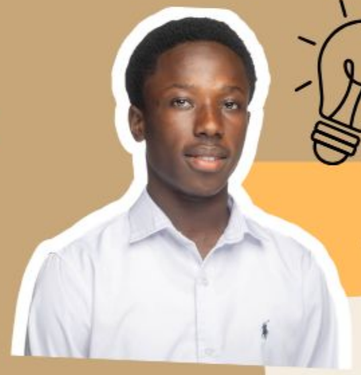
THE FACE

EMMANUEL OPOKU

Narrative Generator

- Loads Hugging Face facebook/bart-large-cnn model locally
- Fine-tunes prompt input using ranked nodes from Person 2
- Generates plain English origin story displayed on the frontend

The Voice



PRINCE OSEI

Graph & PageRank Analyst

- Builds the knowledge graph using NetworkX
- Runs PageRank algorithm to identify and rank root causes
- Exports the ranked chronological origin chain as JSON

The Brain

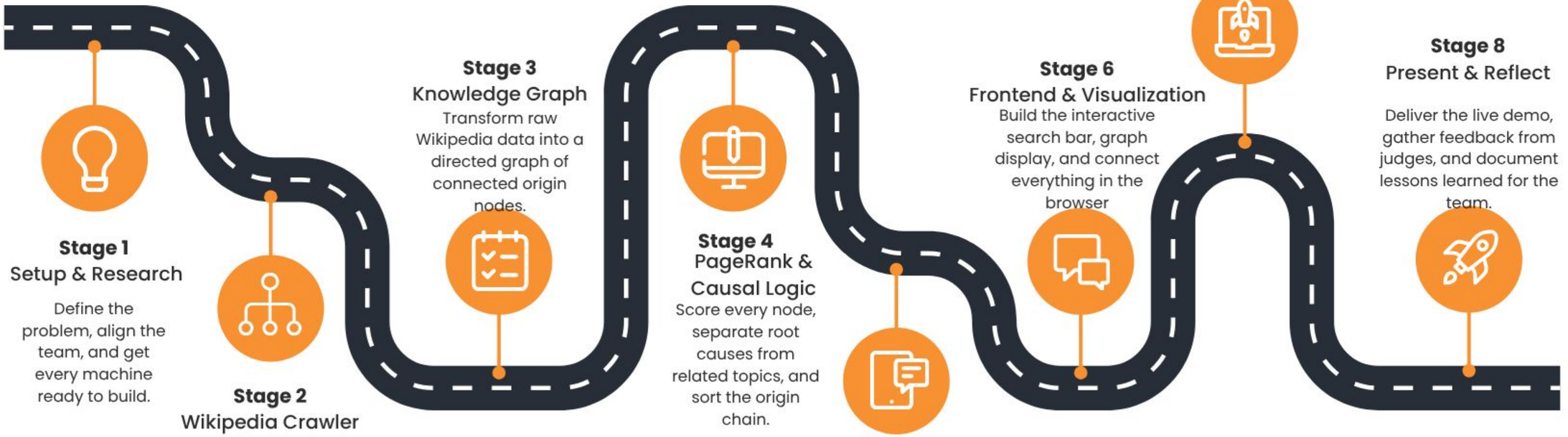
AMELIA HUNTER

Wikipedia Data Crawler

- Seeds the Wikipedia article to begin the knowledge tree
- Makes live API calls to pull articles, links, and categories
- Outputs clean JSON data for the rest of the team to consume

The Engine

Origin Zero — From Idea to Demo



Stage 1
Setup & Research

Define the problem, align the team, and get every machine ready to build.

Stage 2
Wikipedia Crawler

Seed the knowledge tree and pull live Wikipedia articles, links, and categories via API.

Stage 3
Knowledge Graph

Transform raw Wikipedia data into a directed graph of connected origin nodes.

Stage 4
PageRank & Causal Logic

Score every node, separate root causes from related topics, and sort the origin chain.

Stage 5
Narrative Generator

Feed the ranked chain into Hugging Face and generate a plain English origin story.

Stage 6
Frontend & Visualization

Build the interactive search bar, graph display, and connect everything in the browser

Stage 7
Demo & Pitch Ready

Run the full pipeline end to end, fix final bugs, and practice the 5 minute pitch.

Stage 8
Present & Reflect

Deliver the live demo, gather feedback from judges, and document lessons learned for the team.

ORIGIN ZERO ROLES

Days: Tues, Wed, Thur

TUESDAY

WEDNESDAY

THURSDAY

AMELIA HUNTER
Wikipedia Data Crawler
THE ENGINE

Seed Art. → API Crawl → Output JSON

CLEAN JSON DATA HANDOFF

PRINCE OSEI
Graph & PageRank Analyst
THE BRAIN

Build Network Graph → Compute PageRank → Rank Chain

RANKED CHRONOLOGICAL DATA HANDOFF

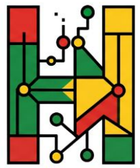
EMMANUEL OPOKU
Narrative Generator
THE VOICE

Load Models → Fine-Tune Prompts → Generate Story

STORY TEXT STREAMING HANDOFF

ANTHONY NWAFOR
Frontend & Visualization
THE FACE

D3.js Viz Design → User Flow Code → Connect Backend



HACKHPC
@ADMI26
JUNE 22-26, 2026

WinMaxxers

Winmaxxers:
https://drive.google.com/file/d/1RSo9XUcA_7guj58Akhidq8IYweRKV9hm/view?usp=drive_link

https://hackhpc.github.io/admi26


CHAMPION
CONQUER
VICTOR
VICTORY
SUCCESS
TRIUMPH
WINNER
WInMaxxers

HACKHPC @ADMI26
JUNE 22-26, 2026

WinMaxxors

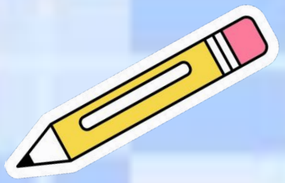
- Team Members Names
  - Quintin Ampofo
  - Cole Scott
  - Damone Washington
  - Darell Barnes

- Future Event Prediction

https://drive.google.com/file/d/1RSo9XUcA_7guj58Akhidq8lYweRKV9hm/view?usp=sharing

GOALS

- Wikipedia-only forecasting — predicting emerging concepts using only Wikipedia structure
- Knowledge graph signals — pages as nodes, links/categories as relationships=
- Explainable modeling — simple, transparent scoring instead of black-box ML
- Historical growth patterns — using past snapshots to detect rising concepts



Themes

- Our project explores whether the structure of Wikipedia itself can reveal tomorrow's important ideas. By treating Wikipedia as a living knowledge graph—where pages are nodes and links, categories, and references form the connections—we analyze how concepts grow and evolve over time.

Darell Barnes

Data Cleaning & Prep

- Responsible for Wikipedia data sourcing, including simulated historical snapshots.
- Collect Wikipedia data and simulate historical snapshots
- Extract links categories references for graph construction

Cole Scott

Modeling & Forecasting

- Designs the simple, explainable future-importance model
- Design heuristic model using transparent weighted features
- Compute future importance_scores and rank emerging concepts

TEAMWORK MAKES THE DREAM WORK!

Damone Washington

Graph & Scoring

- Builds the knowledge graph and computes all structural features
- Build Wikipedia graph from pages, links, and categories
- Compute graph features like PageRank, degree, and growth

Quintin Ampofo

, Visualization

- Owns the front-end, visuals, and storytelling.
- Build notebook structure with clear charts and explanations
- Create graph visualizations and craft the final story

HACKHPC @ADMI26
JUNE 22-26, 2026

WinMaxxers Road to Glory

Milestone 1 — Project Definition and Repository Setup

 # This is the initial environment setup and Ideation

- Create a Github repository to track the project
- Define the scope of our project
- Define key data points to extract
- Assign team roles.

Milestone 2 — Data Acquisition

 # Data cleaning and preparation for analysis

- Select Wikipedia snapshots to acquire
- obtain them via API calls or Web-scraping and Parsing
- Extract key data points, metadata, and links

Milestone 3 — Knowledge Graph Construction

 # Setting up a Knowledge Graph so that obscure or ambiguous relations can be found and converted to a feature (eg, clusters)

- Create node (wikipedia articles) and edge (article-to-article links)
- Build a graph using Networksx
- Validate graph integrity and connectivity

Milestone 4 — Feature Engineering

 # Extracting the features from the graph

- Extract Features from graph such as in-degree, out-degree, and page rank
- select ML algorithm to train model.
- Normalize these values to train a simple prediction model

Milestone 5 — Scoring Model

- Align pages across snapshots.
- Generate labels using future growth measurements.
- Create train/test splits based on time periods.
- Train baseline models.

Milestone 6 — Validation

***#** at this stage we essentially start to build the narrative / story*

- Identify emerging concepts.
- Analyze why concepts grow.
- Study community evolution and cluster growth.
- Discover future knowledge hubs.

Milestone 7 — Polish

- Build interactive graph explorer.
- Build timeline controls and time-travel view.
- Create prediction dashboard.
- Add search and concept exploration tools.
- Develop final presentation deck.
- Conduct demo rehearsals and judging preparation.

.

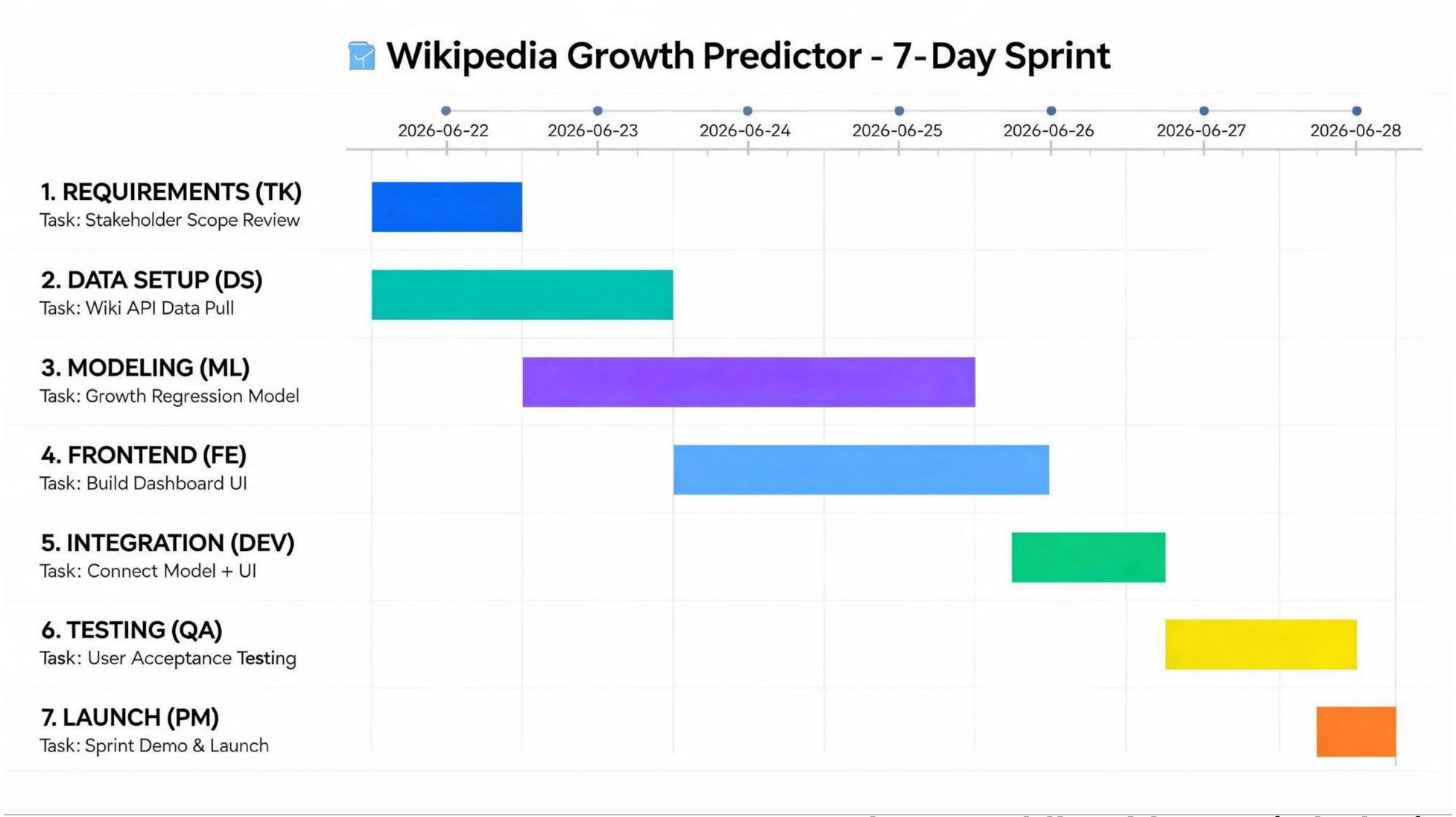
https://hackhpc.github.io/admi26

The Project Journey

**Timeline incorporates a 1.5x buffer to account for API rate-limiting and data cleansing overhead.*

HACKHPC @ADMI26
JUNE 22-26, 2026

Gantt Chart



HACKHPC @ADMI26
JUNE 22-26, 2026

WinMaxxers Technologies

🔷 *Data Source*

Wikipedia (MediaWiki API, Wikidata)
Historical page revisions & link structures

🌐 *Graph Modeling*

Wikipedia as a knowledge graph
(Nodes = pages | Edges = links)
Category + reference-based connections

📊 *Feature Engineering*

Centrality metrics (PageRank, degree, betweenness)
Link & citation growth trends
Category expansion over time

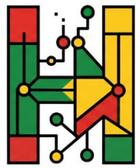
🧠 *Analytics & Tools*

Python (Jupyter Notebook)
NetworkX – graph analysis
Pandas – data processing
Explainable scoring model (weighted features → probability)

🚀 *Forecasting Approach*

Historical snapshot comparison (2015 → 2026)
Identify emerging vs. established concepts
Transparent probability-based ranking

https://hackhpc.github.io/admi26



HACKHPC
@ADMI26
JUNE 22-26, 2026

Capsule Corp CoderZ

https://hackhpc.github.io/admi26

HACKHPC @ADMI26
JUNE 22-26, 2026

Capsule Corp CoderZ

- Team Members Names
  - Javonte Carter
  - Joshua Chaney
  - Sydney Riddick
  - Chance Walker

Saiyan Spark

- Alternative Timeline Analysis

https://drive.google.com/file/d/1kQI7C_UtOGds_t3YNHLJpYXlbjNZM3yY/view?usp=sharing


CAPSULE
CORP.
CAPSULE CORP
CODER

Team Introduction:

Javonte Carter

FAMU

Junior Dev

Joshua Chaney

MVSU

Project Manager & Junior Dev

Sydney Riddick

ECSU

Creative & Graphic Designer

Chance Walker

WSSU

Project Manager

Project Theme:

The entire main timeline hinges on baby Goku falling down a ravine and suffering a massive head injury that erased his violent, innate Saiyan programming. If that accident never happened, Earth's history would be incredibly dark.

The Conquest of Earth: Instead of being raised as a kind-hearted martial artist, Goku would have grown up as the ruthless Saiyan "Kakarot." He likely would have killed Grandpa Gohan as a child or transformed into a Great Ape on a full moon and decimated the planet's population.

The Reunion: When Raditz finally arrives on Earth in Age 761, he won't find a peaceful man protecting the planet. He would find a barren, conquered world with Kakarot waiting for him.

The Future: Kakarot would happily join Raditz, Vegeta, and Nappa. Together, the four Saiyans would continue wiping out civilizations for Frieza. However, without the intense martial arts training and rivalries Goku experienced on Earth, his power level would likely remain incredibly low, meaning they would eventually be wiped out by Frieza anyway once they outlived their usefulness.

STATUS: CONQUERED

MILESTONES:

Milestone 1:
- Project Planning
- Choose the final project concept
- Define the main game paths
- Assign team member roles

Milestone 2:
- Story and Timeline Design
- Write the opening story
- Create at least two main timeline branches
- Write possible endings for each path

Milestone 3:
- Data Collection
- Seed the knowledge tree with a Wikipedia or Dragon Ball Wiki article
- Collect information about Goku, Raditz, Saiyans, Vegeta, Nappa, and Frieza
- Organize lore data into JSON or documents

Milestone 4:
- Game Logic Design
- Create a decision tree for player choices
- Decide how choices affect the ending
- Build a simple scoring system such as loyalty, destruction, rebellion, or power level

Milestone 5:
Front-End Development
Build the homepage
Create choice buttons
Design character cards and timeline screens

Milestone 6:
- Back-End / AI Integration
- Connect game choices to stored story branches
- Add AI-generated responses if possible
- Build a RAG system to answer lore-based questions

Milestone 7:
- Testing and Debugging
- Test every story path
- Fix broken buttons, missing endings, or incorrect branches
- Make sure the game is easy to understand

Milestone 8:
- Final Presentation
- Create demo slides
- Explain the alternate timeline theme
- Show the game, technologies, milestones, and team roles

STATUS: CONQUERED

MILESTONE TIMELINE:

CAPSULE CORP CODERZ

5-DAY ACCELERATED TIMELINE

SPRINT PHASES

PHASE	MON (Day 1)	TUE (Day 2)	WED (Day 3)	THU (Day 4)	FRI (Day 5)
PM & Data	Create Team Name & Logo; Curate Theme	Assign Roles	M1 & M3: Setup		
Narrative		M2 & G4: Story / Prologue	G5: Mid/Late Sagas		
UI / UX		G3: Base UI & Avatars	M5: App Layout		
Back-End / AI		G1 & G2: Engine/Combat	M4: Decision Trees	M6 & G6: LLM + Yemma	
QA & Launch				M7: Beta Testing	M8 & G7: Final Pitch

DAILY EXECUTION BY ROLE

DAY	CHANCE (PM)	JOSHUA (PM/DEV) </>	JAVONTE (DEV)	SYDNEY (DESIGNER)
MON	Create team name & logo; curate project theme. Assign roles & set project vision.	Create team name & logo; curate theme. Assign roles & set project vision.	Create team name & logo; curate theme. Assign roles & set project vision.	Create team name & logo; curate theme. Assign roles & set project vision.
TUE	Setup Git; define 3 main story endings (M1, M2). Track 7 prologue scripts (G4).	Scrape wiki data into JSON format (M3). Integrate JSON lore to backend.	Build core "Age-up" Python state loop (G1). Code stat matrix & combat math (G2).	Design DBZ color palette; mock UI layout. Draw Baby/Child Kakarot dynamic portraits (G3).
WED	Define conditions for late-game paths (G5).	Design Gemini LLM prompt structures.	Build Great Ape flags & mid-game trees (M4).	UI assets for Gradio/Streamlit; Adult portraits.
THU	Draft pitch slide structure.	Implement RAG pipeline (M6).	Code King Yemma Rebirth loop (G6).	Finalize Frontend UI panel layout (M5).
FRI	Lead QA beta testing; finalize Pitch Deck (M8).	Stress test API connections; fix crashes.	Balance power levels; fix dead-ends (M7, G7).	Polish final slides in Canva.

GOALS

Prototype the Core "Age-Up" Engine & Stat Matrix
Build the foundational text-simulator framework. Successful completion means a player can tap an "Age Up" button, watch their age increase, and see their core attributes (Ki Level, Malice, Infamy, and Health) dynamically update based on basic placeholder events.

Integrate the Text-Based Combat & Great Ape Systems
Develop the turn-based combat algorithm that calculates victories based on Ki levels and chosen techniques. Program the global "Full Moon" calendar event, ensuring a high-Malice Kakarot automatically triggers a Great Ape rampage that alters the world state.

Deploy the UI/UX Interface and Dynamic Portrait System
Move the game from a raw text build into a polished visual state. Design a clean, high-contrast, mobile-friendly interface utilizing Dragon Ball inspired color schemes, and implement evolving character portraits that shift based on Kakarot's age and active transformations.

Fully Script and Branch the 7 Prologue Paths (First 12 years)
Write and program all text options, choices, and RNG events for the childhood phase across all seven distinct origins (e.g., Crane School, Red Ribbon Army, Kami's Lookout). Players must be able to reach age 12 with vastly different stats and faction reputations.

Build out the Mid-to-Late Game "What If" Sagas (After 13 years)
Expand the narrative architecture to handle massive world-altering choices. Goal achievement requires mapping out how the world reacts to an evil Kakarot—whether he is conquering Earth for King Piccolo, ruling it via the Red Ribbon Army, or preparing to fight his own Saiyan heritage.

Code the "King Yemma" Legacy & Rebirth System
Implement the loop for player death. When Kakarot dies, the game must transition to a check-in screen at King Yemma's desk, tally the player's achievements, and reward them with "Saiyan Pride" points used to unlock rarer origin paths for subsequent playthroughs.

Execute Balance Tuning, Closed Beta, and Final Deployment
Run extensive automated testing simulations to ensure power levels scale correctly across all 7 paths (preventing soft-locks against late-game bosses like Frieza). Wipe out game-breaking narrative dead-ends during a closed beta phase before deploying the stable 1.0 build to app stores.

Technology Used:

- Google Gemini
- GitHub
- ChatGPT
- Canva
- Jupyter
- Claude