Cybersecurity Education with Generative AI: Creating Interactive Labs from Microelectronic Fundamentals to IoT Security Exploitation

Authors: Kushal Badal, Xiaohong Yuan, Huirong Fu, Darrin Hanna, Jason Gorski

North Carolina Agricultural and Technical State University

The Cybersecurity Education Challenge

- **months per lab module**
- **Materials obsolete by deployment**
- **High expertise & infrastructure costs**
- **Technology outpaces curriculum**

Traditional development cannot keep pace with evolving threats

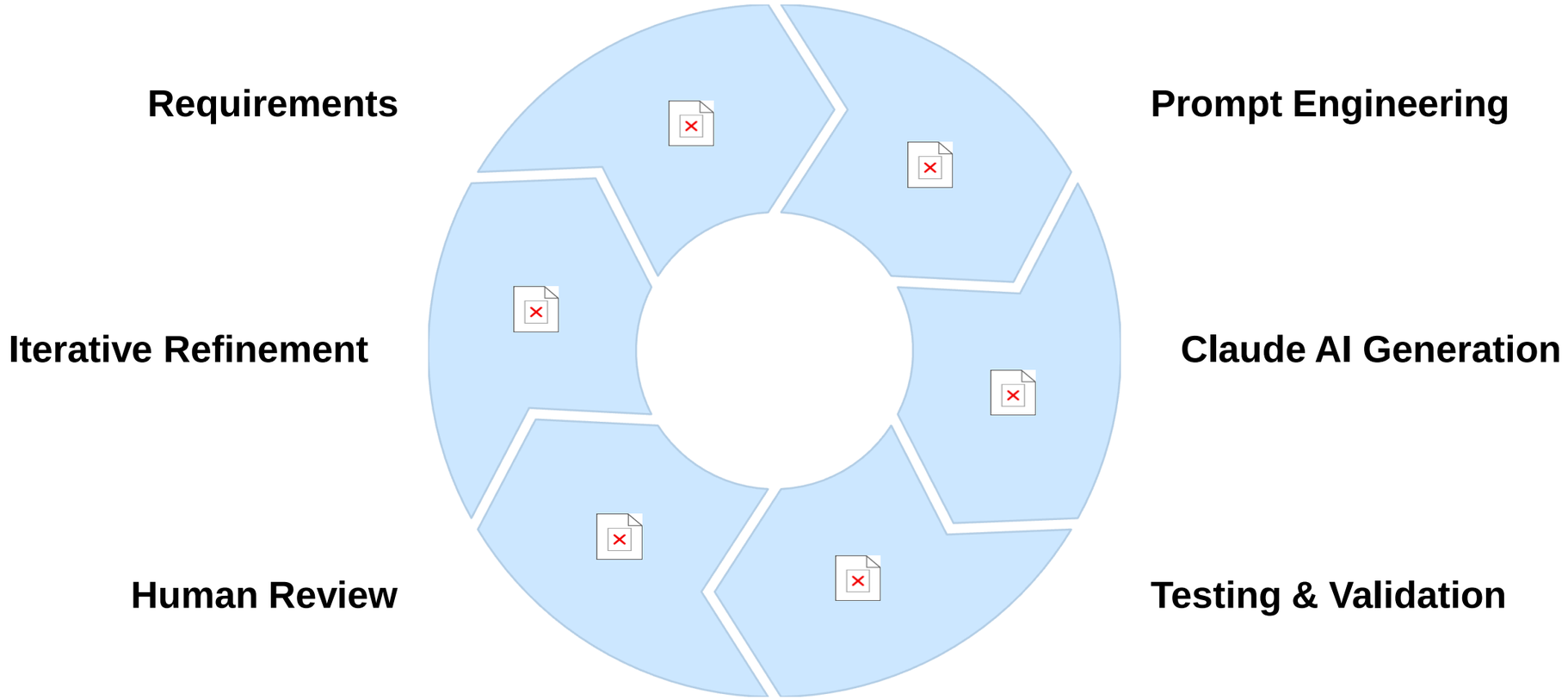
Our Solution - The Breakthrough

MONTHS → **21 DAYS**

AI	15,247	40	2025
Claude AI Opus 4	lines of code	K-12 students	GenCyber Camp

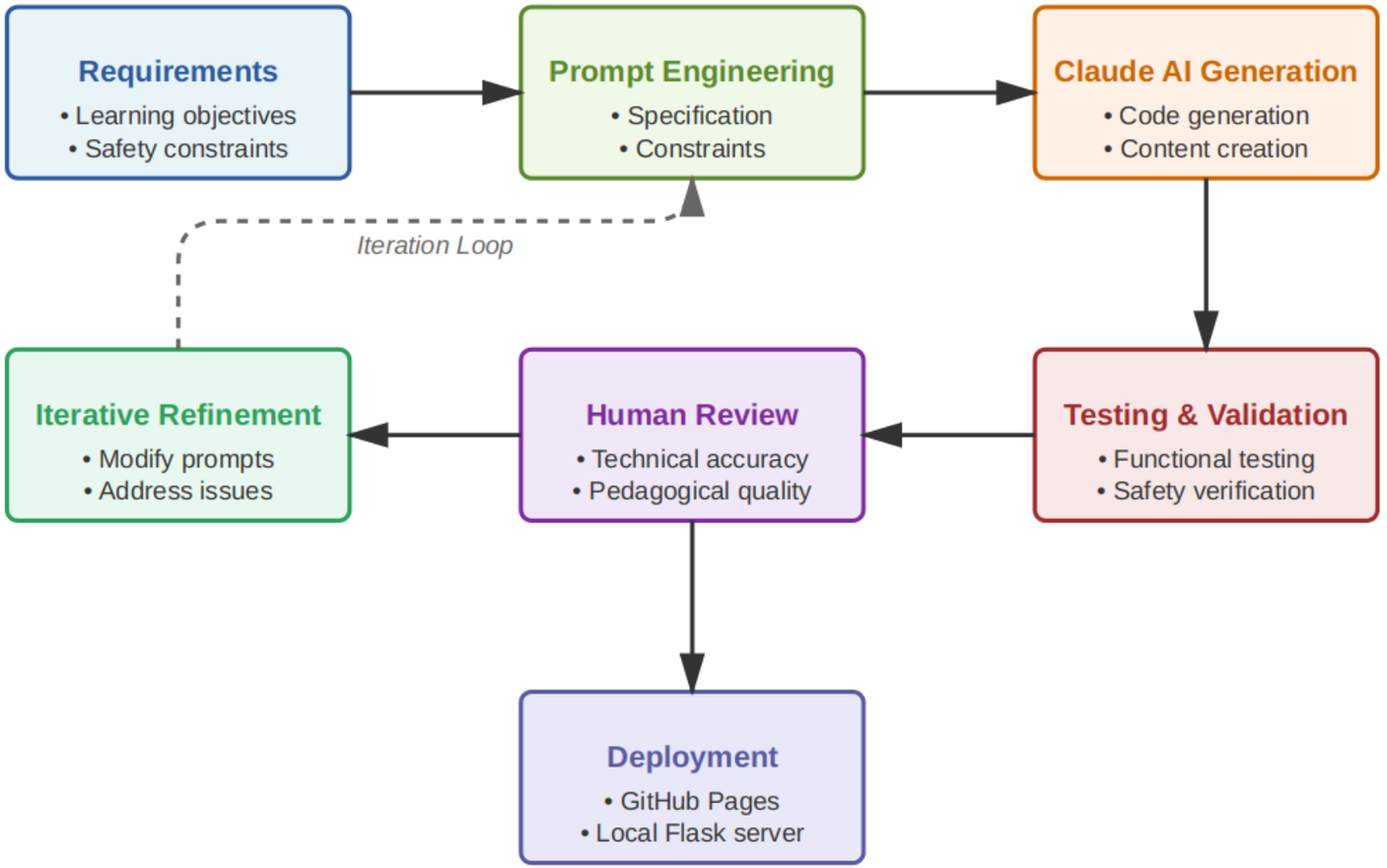
AI-Assisted Development: 80%+ Time Reduction

AI-Assisted Development Process



20 major prompts • ~30 refinements • 15% manual corrections

AI-Assisted Development Process



Complete Educational Platform

Interactive Content

- 12 HTML5 presentations
- Vacuum tubes → IoT evolution
- Gamified quizzes + leaderboards

Security Tools

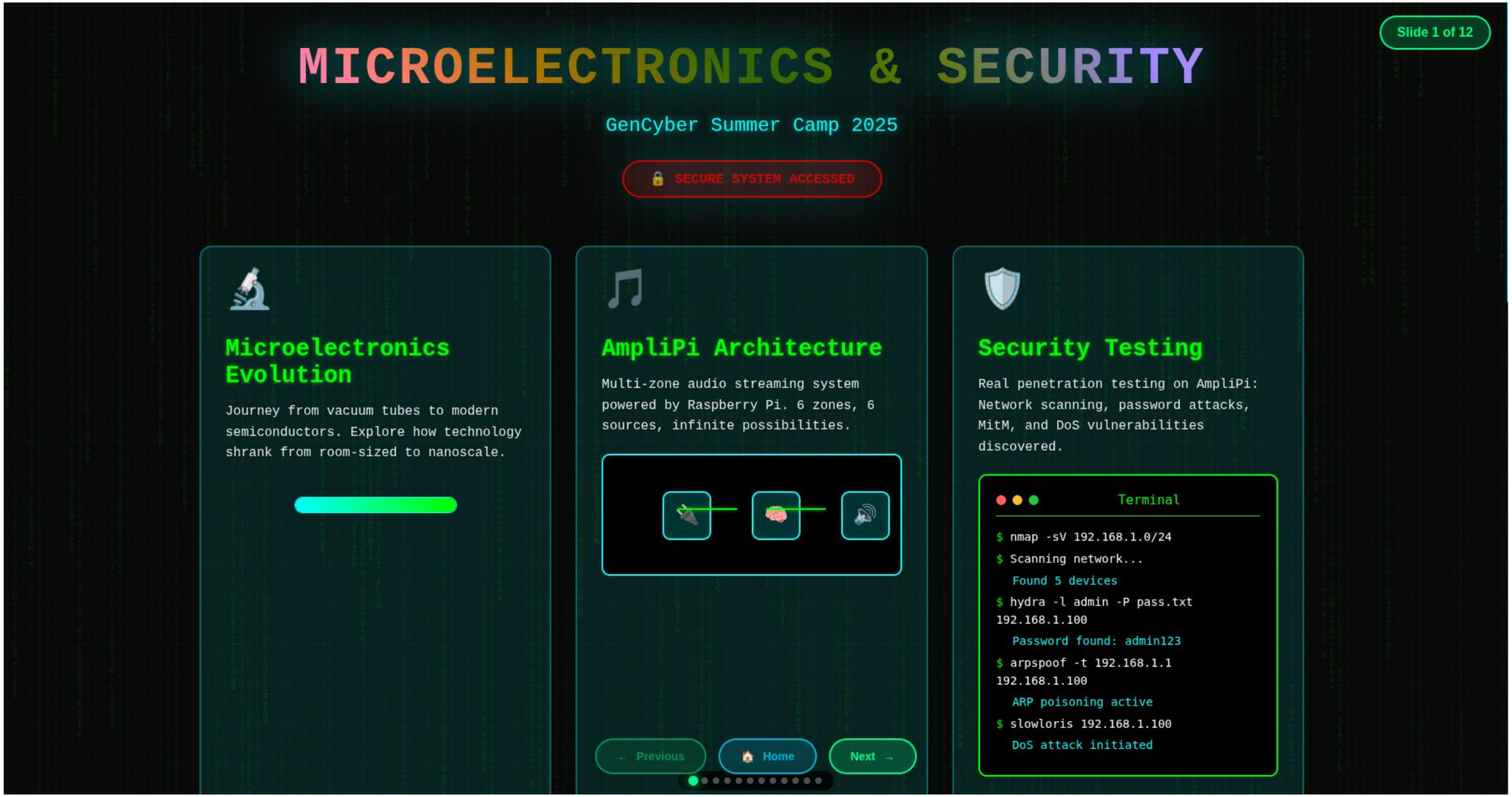
- SlowLoris DoS tool
- Dictionary password cracker
- Built-in safety controls

Real Target

- AmpliPi IoT audio system
- Open-source platform
- Genuine vulnerabilities

All materials open-source on GitHub

Complete Educational Platform



Teaching Through Visualization: Network Discovery

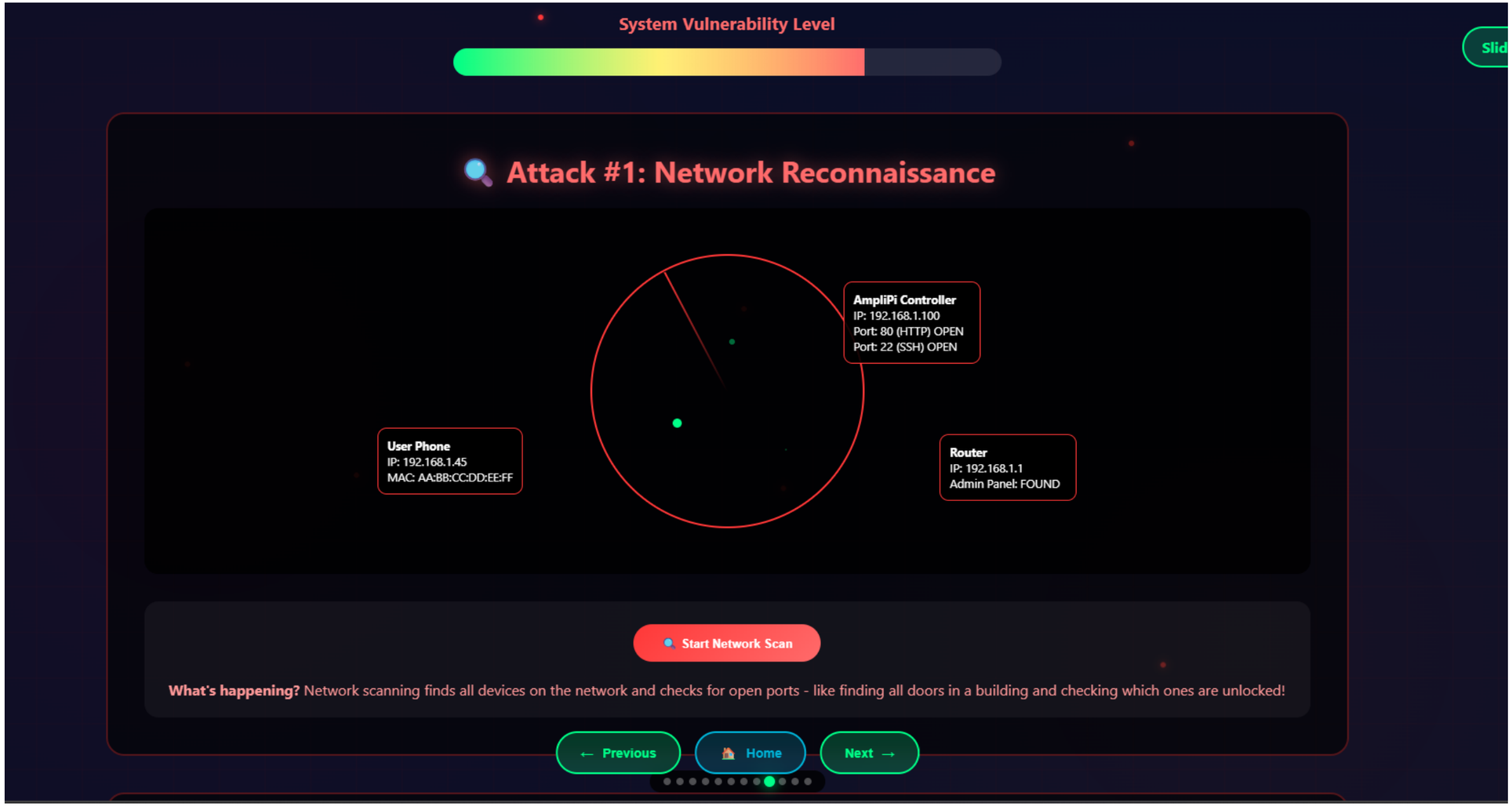
Key Features

- Progressive revelation - devices appear one by one
- Visual + audio feedback
- Learning by doing, not reading

Educational Value

- Students understand scanning process
- Immediate engagement
- Concepts become tangible

Network Discovery



Teaching DoS Attacks: The Restaurant Analogy

The Metaphor

Network connections = Restaurant tables

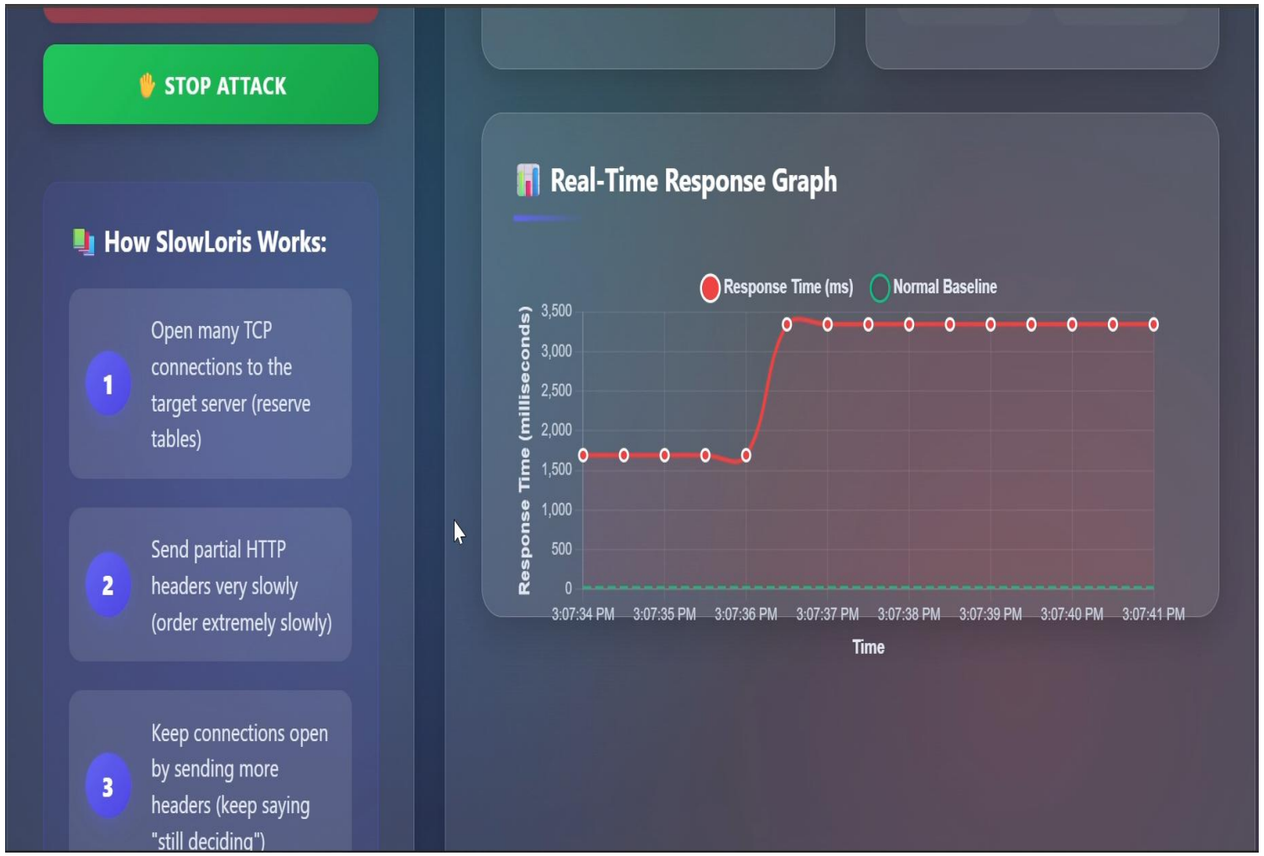
Partial HTTP requests = Customers holding tables

Never completing requests = Never ordering food

> "It's like someone sitting at all tables but never ordering food"

DoS Attacks


Understanding SlowLoris: The Restaurant Analogy
The Restaurant
Your web server is like a restaurant with limited tables (connection slots). It can only serve a certain number of customers at once.
The SlowLoris Attacker
The attacker sends many "customers" who sit at tables but order very slowly, never finishing their order and holding tables indefinitely.
The Attack Method
Each fake customer keeps saying "I'm still deciding..." every few seconds, preventing the waiter from giving the table to real customers.
The Impact
Eventually all tables are occupied by slow fake customers. Real customers can't get in - the restaurant appears "full" but no one is actually eating!
READY
Current Status
0
Active Connections
50ms
Response Time
0
Blocked Attempts
Attack Control Panel
Target IP (AmpliPi Device)
192.168.1.127
Target Port
80
Number of Connections (Fake Customers)
50
TEST CONNECTION
LAUNCH SLOWLORIS ATTACK
STOP ATTACK
How SlowLoris Works:
Live Results & Impact
Attack Visualization
Attacker
AmpliPi Server
NORMAL
Defense Mechanisms
Connection Timeout
Kick out slow customers after 30s
Rate Limiting
Max 20 tables per customer group
Reverse Proxy
Bouncer filters fake customers
All Defenses
Maximum restaurant security
Real-Time Response Graph


STOP ATTACK
How SlowLoris Works:
1
Open many TCP connections to the target server (reserve tables)
2
Send partial HTTP headers very slowly (order extremely slowly)
3
Keep connections open by sending more headers (keep saying "still deciding")
Real-Time Response Graph
Response Time (ms)
Normal Baseline
Response Time (milliseconds)
3,500
3,000
2,500
2,000
1,500
1,000
500
0
3:07:34 PM
3:07:35 PM
3:07:36 PM
3:07:37 PM
3:07:38 PM
3:07:39 PM
3:07:40 PM
3:07:41 PM
Time

Dictionary Attack & Password Strength Analysis

Attack Features

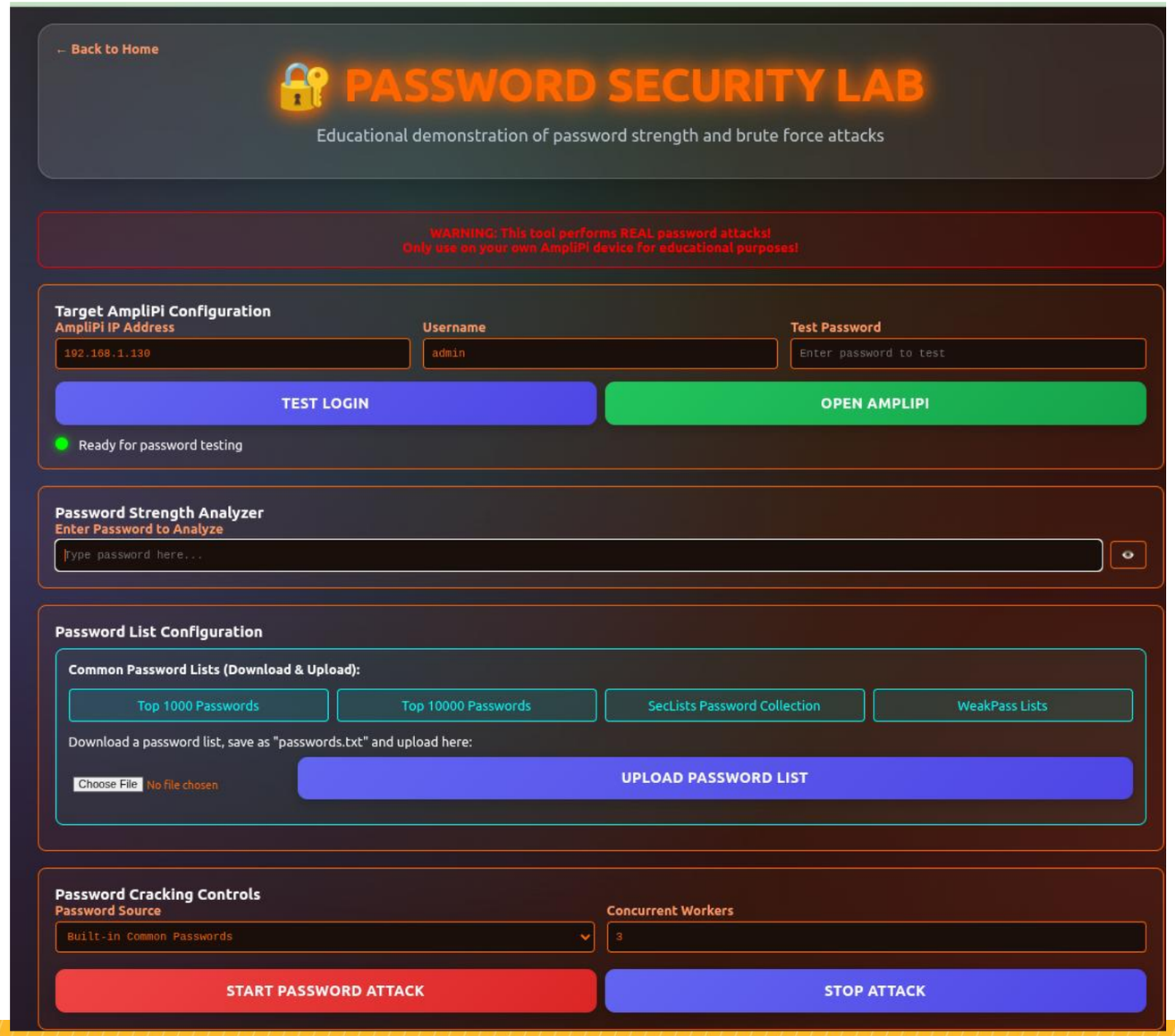
- Real-time attack progress
- Multiple authentication formats
- AmpliPi-specific handling

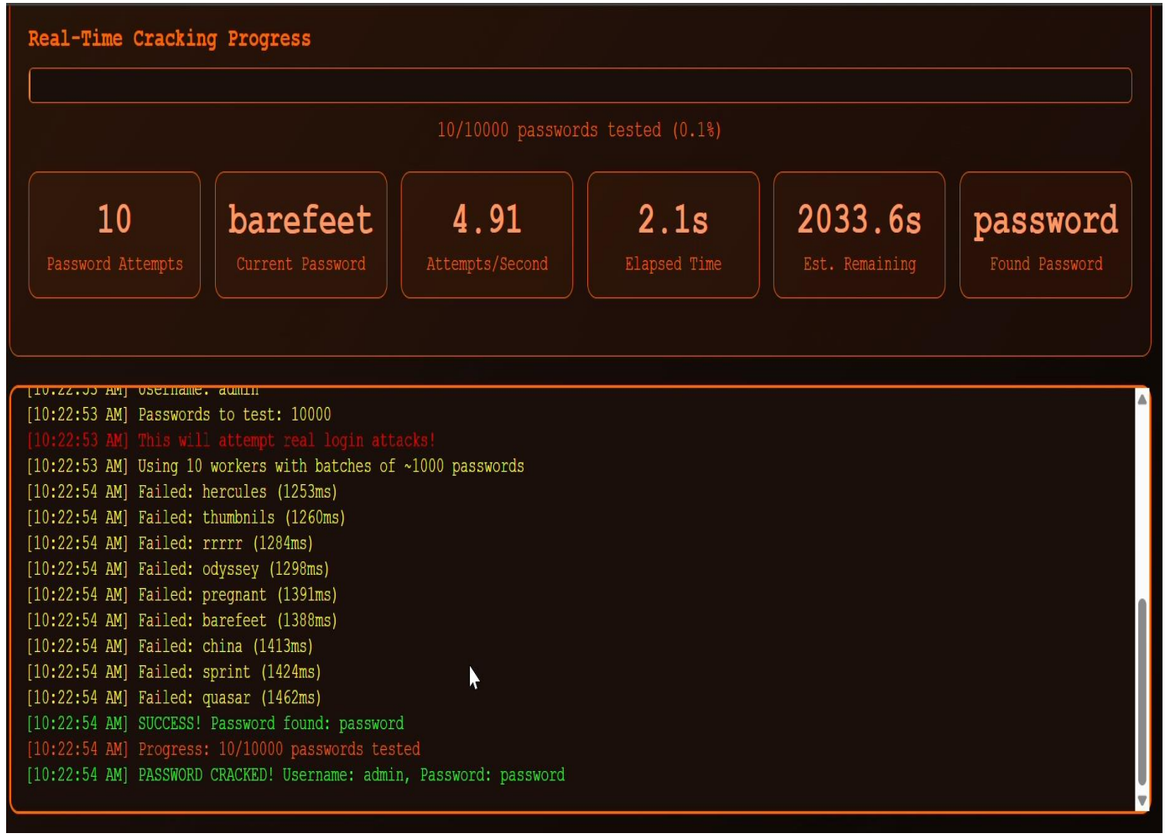
Educational Components

- Password strength analyzer
- Educational feedback
- Rate limiting: 3 workers max

All students successfully discovered weak default passwords

Dictionary Attack





Real-World IoT Target: AmpliPi Audio System

Open-source multi-room audio system

REST API with multiple endpoints

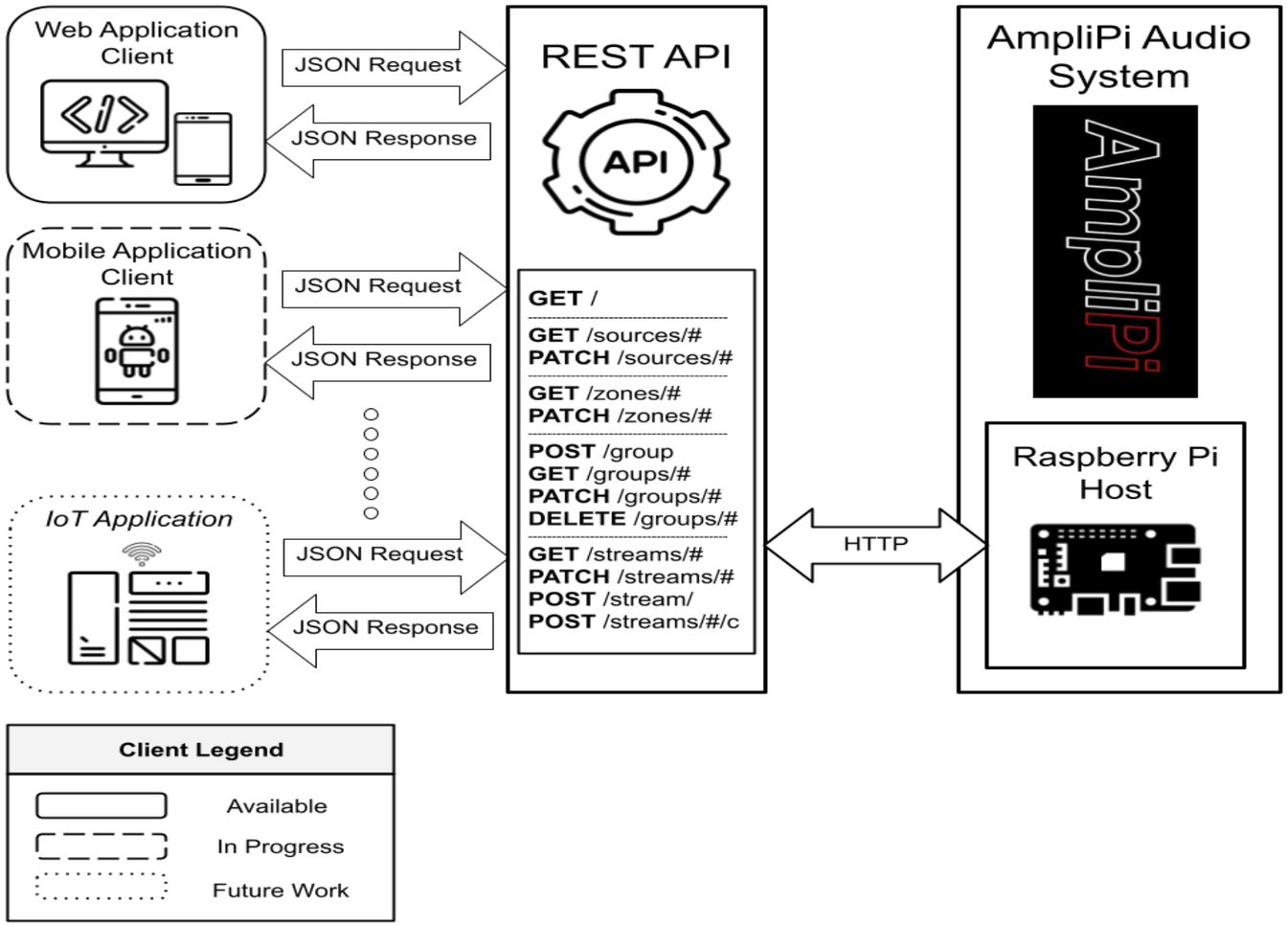
Runs on Raspberry Pi

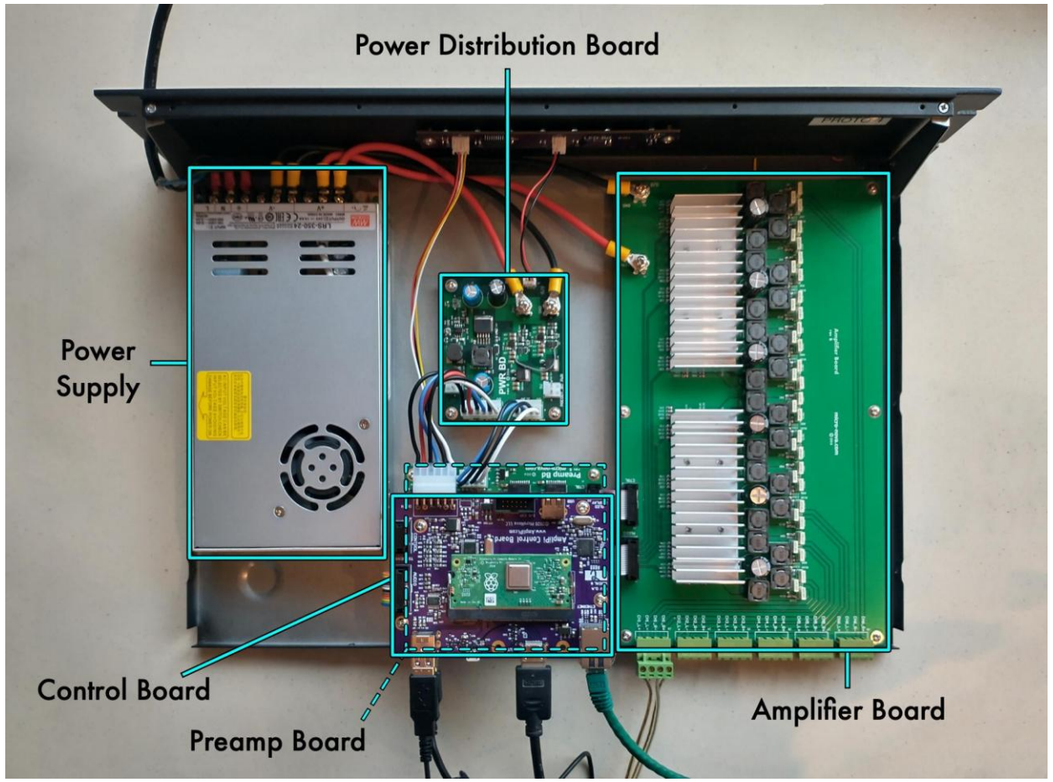
Representative of real IoT deployments

Attack Vectors Explored

- Authentication weaknesses
- Denial of service vulnerabilities
- API endpoint exploitation

AmpliPi Audio System





Real-World Deployment: GenCyber Summer Camp 2025

40
K-12 Students

2
Weeks, 2 Cohorts

1
Isolated Lab Network

Lab Environment

20	1	2	VLAN
Lab computers	Instructor machine (Flask server)	AmpliPi devices	192.168.1.0/24

All exercises supervised with safety controls

Quiz Result

A1 | Timestamp

	A	B	C	D	E	F	G	H	I	J	K
1	Timestamp	Name	Score	Completion Time (second	Percentage	Q1 Answer	Q2 Answer	Q3 Answer	Q4 Answer	Q5 Answer	Session ID
2	31/07/2025 09:13:46	Al[redacted]n	5	10.4	100%	c	b	b	c	c	2025-07-31T13
3	31/07/2025 09:13:58	[redacted]He	5	19.5	100%	c	b	b	c	c	2025-07-31T13
4	31/07/2025 09:14:02	K[redacted]ako-F	5	28.6	100%	c	b	b	c	c	2025-07-31T13
5	31/07/2025 09:13:57	Z[redacted]vis	4	13.1	80%	c	c	b	c	c	2025-07-31T13
6	31/07/2025 09:13:52	El[redacted]row	4	22	80%	c	b	b	c	a	2025-07-31T13
7	31/07/2025 09:14:25	K[redacted]parte	4	30	80%	c	b	b	c	-	2025-07-31T13
8	31/07/2025 09:14:48	J[redacted]n	4	33	80%	c	b	b	c	a	2025-07-31T13
9	31/07/2025 09:14:02	K[redacted]h	3	30	60%	c	b	b	-	-	2025-07-31T13
10	31/07/2025 09:14:01	ch[redacted]	2	30	40%	c	c	b	b	-	2025-07-31T13
11	31/07/2025 09:14:10	Ka[redacted]	2	30	40%	c	-	-	a	c	2025-07-31T13
12	31/07/2025 09:14:32	gav[redacted]son	2	30	40%	c	c	b	b	-	2025-07-31T13
13	31/07/2025 09:14:35	Je[redacted]Major	2	30	40%	a	c	b	c	-	2025-07-31T13
14	31/07/2025 09:14:13	Ju[redacted]thin	1	30	20%	b	a	b	-	-	2025-07-31T13
15	31/07/2025 09:14:01	lak[redacted]uri	0	30	0%	-	-	-	-	-	2025-07-31T13

Keeping Students Engaged: Gamification in Action

Real-time leaderboard

Google Sheets integration

Immediate feedback

Competitive element

100% participation • Sustained attention • Active competition

Student Achievements: Real Vulnerabilities Discovered

Dictionary Attack Success

✓ 100% of students discovered weak passwords

SlowLoris DoS Demonstration

✓ All students successfully executed DoS attacks

AI-Generated Code: Strengths & Limitations

✓ What Worked Well

- Functional code with minimal modifications
- Creative pedagogical approaches
- Rapid iteration capability
- Constitutional AI safety guardrails
- 85% code used without changes

⚠ Required Human Oversight

- Domain-specific edge cases (HTTP 302)
- Browser compatibility fixes
- Pedagogical appropriateness review
- Technical accuracy validation

AI as accelerator, humans ensure quality & safety

Challenges & Solutions

1

Technical Challenges

- Flask server: 20 user limit
- Authentication detection issues
- Browser inconsistencies

2

Educational Challenges

- Hardware dependency (AmpliPi devices)
- Network setup complexity (VLAN configuration)
- Instructor technical expertise required

3

Solutions Applied

- Iterative refinement process
- Manual validation protocols
- Clear safety boundaries

Our Contributions

1

Complete Methodology

- Prompt engineering strategies
- Iterative refinement process
- Safety control implementation

2

Functional Tools

- Working penetration testing demos
- Educational scaffolding
- Built-in safety mechanisms

3

Validated Results

- 40 student deployment
- Real vulnerability identification
- GenCyber camp success

4

Open Source

- All materials on GitHub
- Reproducible by educators
- Community contribution

All materials available at microelectronics2025.github.io

Key Takeaways & Future Directions

Key Insights

AI reduces development time 80%+

Human expertise remains essential

Effective learning outcomes achieved

Future Research

- Model comparison (GPT-4, Gemini)
- Expanded IoT target platforms
- Real-time AI-human co-creation

Thank You

Questions?

GitHub repository: microelectronics2025.github.io

All materials open-source and freely available

Contact: kbadal@aggies.ncat.edu, xhyuan.ncat.edu