COSMOS Educational Toolkit

Abhi Adhikari

Department of Electrical and Computer Engineering, Columbia University

https://cosmos-lab.org/outreach

May 2023

*The COSMOS testbed design and deployment is joint work with the COSMOS team (www.cosmos-lab.org).



















COSMOS Educational Toolkit

- Hardware components:
 - Processing Units: Intel NUC, Raspberry PI
 - Software Defined Radio: ADALM Pluto SDR, RTL-SDR
 - IoT Nodes: Arduino/Micro:bit, XBEE/BLE, sensors
- Software components:
 - Web front-end interface: HTML, CSS and Javascript
 - Web back-end server: Python
 - SDR Software: GNU Radio
 - IoT management: influxdB and Chronograf



Fig: Hardware components of the 'COSMOS Educational Toolkit'



















Program Overview (1/3)

- a. Lecture and lab phase: The participants are introduced in fundamental and some advanced concepts in wireless communications and networking
 - \$8K stipend for summer
- **b. Design phase:** The participants conduct research on potential educational NGSS STEM lessons with a hands-on wireless labs using the 'COSMOS Educational Toolkit'



Fig: Participants attend an instructor-led lesson using the 'COSMOS Educational Toolkit'



Fig: Participants conduct research using sensors



















Program Overview (2/3)

- c. Development phase: teachers co-develop with the researchers their best ideas on how to use the wireless labs for NGSS-aligned STEM lessons
- d. Implementation phase: teachers and students use the developed lessons in the class during the school year



Fig: Teachers present their developed NGSS lesson plans at Silicon Harlem

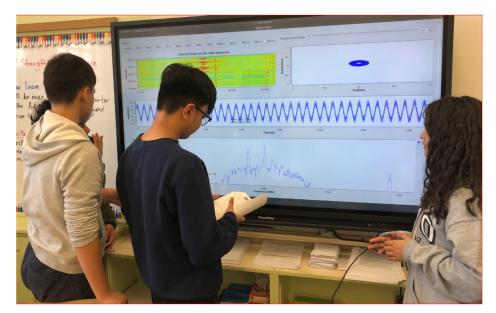


Fig: Middle school students using the 'COSMOS Educational Toolkit' in a Mathematics Class



















Program Overview (3/3)

e. Feedback phase: teachers provide feedback in order to improve the NGSS STEM lessons and develop new ones





Fig: Teachers present the PD Program and the developed NGSS Lesson Plans at the NE-ASTE Conference 2018



















Conclusion

https://cosmos-lab.org/outreach

























