USC Viterbi
School of Engineering

EE 250L Distributed Systems for the Internet of Things

Units: 4

Term—Day—Time: Spring, Lectures: Tuesdays and Thursdays 9:30 to 10:50am, Lab: Wednesday 2 to 4:50pm

Location: TBA

Instructor: Bhaskar Krishnamachari
Office: RTH 410
Office Hours: Wednesdays 10-12, other times by appointment
Contact Info: bkrishna@usc.edu , 213-821-2528, emails are typically responded to within 48 hours.

Teaching Assistant: TBA
Office: TBA
Office Hours: TBA
Contact Info: TBA

Course Description
This is a 4-unit undergraduate course aimed at electrical engineering, computer science and computer engineering students with at least a Sophomore Standing. Students will learn how to design, build and evaluate hardware, operating systems, network and application layer protocols, as well as distributed system applications for the Internet of Things. Through hands-on lab activities, assignments, projects, as well as through guest lectures spanning research and practice, the course provides a comprehensive introduction to many relevant topics including printed circuit board design, real-time operating systems, programming microprocessors, signal processing, wireless communications, networks, control systems, publish-subscribe middleware, cloud computing, databases, machine learning and security.

Learning Objectives
By the end of the course, a student taking this class will be able to demonstrate:

- an understanding of how microprocessors, sensors, and radio hardware are integrated in an embedded hardware platform
- the ability to create circuit schematics and based on schematics create a printed circuit-board (PCB) design and layout and test a PCB's functionality
- an understanding of what real-time embedded operating systems do and the kind of programmer-support interfaces they provide
- an understanding of how low power wireless links behave in terms of metrics such as signal quality, error rates, throughput, as a function of link distance, channel and power settings.
- an understanding of how randomized medium access (MAC) protocols such as carrier sense multiple access allow multiple wireless devices to share a medium.
- an understanding of how to prolong the battery lifetime of an embedded device by using sleep cycling and asynchronous MAC protocols
- an understanding of how to process raw data / signals from sensors to minimize noise and to estimate useful parameters such as a device's location
- an understanding of the TCP/IP protocol suite and how to program one device to talk over the Internet to another using sockets.
- an understanding of how publish-subscribe protocols such as MQTT work, and how to programmatically use them to share messages in real time between distributed devices all over the Internet
- how to program mobile devices to access their sensors and communication capabilities.
- an understanding of how cryptographic encryption schemes and security protocols work to protect the confidentiality and integrity of data.
- an understanding how to store and process information on the cloud using databases and computing frameworks such as Hadoop or Google's Cloud Data Flow
- the ability to learn new emerging distributed system technologies emerging in the field of IoT and present them orally to classmates

Prerequisite(s): EE 109L, EE 155L

Recommended Preparation: hands-on experiences with DIY electronics, programming in a distributed, networked environment, experience with Linux.

Course Notes
Letter Grade. Web-Enhanced Course (Blackboard) - course notes, reader will be posted online. Course will include programming assignments to be turned in online via Github (http://github.com). There is a significant lab component to the course, where students will program embedded and cloud-based IoT platforms and applications and empirically evaluate not only functionality but also performance measures such as latency, energy, throughput.

Technological Proficiency and Hardware/Software Required
Students will need to have access to a sufficiently up-to-date computer for programming, the needed software and related instructions will be provided by the course instructor and teaching assistant. Additional hardware electronics and embedded processor supplies needed will be provided.

Required Readings and Supplementary Materials
Required readings (in the form of course notes developed and prepared by the instructor) and pointers to supplementary materials (research papers, tutorials and surveys) will be posted online on the course website.

There will be no required textbooks for the course, however the following two books are recommended texts:

Donald Norris, "The Internet of Things: Do-It-Yourself at Home Projects for Arduino, Raspberry Pi and BeagleBone Black", McGraw Hill Education, 2016.

Arshdeep Bahga and Vijay Madisetti, Internet of Things (A Hands-on-Approach), 2014.

Other supplementary materials that will be provided include the ContikiOS tutorials online at http://anrg.usc.edu/contiki/index.php/Contiki_tutorials , Android developer tutorials online at https://developer.android.c om/training/index.html , also relevant online materials from cloud computing providers such as AWS and Google Cloud.

Description and Assessment of Assignments

There will in-class lab activities (not assessed) nearly each week, along with a hands-on assignment (hardware demonstration or programming-based) to be done by each student individually that builds on the in-lab activity. Assignments will be assessed through grading within a week. The top 8 out of 10 total assignments will count towards the final grade.

There will be 2 projects to be done as group of two students. These projects will allow some opportunity for students to showcase their creative and innovative thinking.

There will be three exams in the class, that will test students' understanding of the key underlying concepts covered in class.

Each student will be asked to research, learn about and teach to their classmates in a roughly 10-15 minute long presentation some newly emerging IoT platform or technology. The goal of this activity is to help students learn to learn (an essential skill in today's rapidly changing world), and learn to communicate.

Further information on assessment strategies / rubrics is given in the assignment rubrics section.

Grading Breakdown

Assignment	**Points**	**% of Grade**
Homework	5x(best 8 of 10) = 40	40%
Project	10x2 = 20	20%
Exams	10x3 = 30	30%
Presentation	10	10%

Grading Scale
Course final grades will be determined using the following absolute scale. There will be no "curve"-based grading.

A	91-100
A-	89-90
B+	86-88
B	81-85
B-	79-80
C+	76-78
C	71-75
C-	69-70
D+	66-68
D	61-65
D-	59-60
F	58 and below

Assignment Rubrics
Any assignments or exam problems that require quantitative calculations will be evaluated primarily on the aptness and correctness of the analytical procedure used and the final answer. However curiosity and exploration will be encouraged -- even in case of incorrect answers significant partial credit may be provided in the face of evidence that the student attempted a different, unusual, novel but somehow justified line of approach in answering a question, even if it didn't result in a correct answer.

Assignments or project components involving programming will be graded on the correctness of the
code, style and documentation of the source code, as well as on any creative new features introduced by the student.

Projects will be evaluated on the whole on the basis of functionality, presentation, and evidence of creative originality. Projects will be assigned to groups of 2 students. The students will be asked to self report and assess each other's exact contributions to the project and this information may be used to distinguish their respective individual grades for the project.

In-class Presentations will be evaluated on the basis of organization of material, informativeness, and clarity.

For all work, evaluation will take into account positively any evidence of creative thinking such as use of order of magnitude estimates to simplify calculation, or evidence of self-initiated research to find information going beyond what was taught or presented in class.

Assignment Submission Policy
HW assignments and projects are due at 11pm the night before the second meeting day of the class in the corresponding week (see timetable above for weeks when each assignment and project is due).

Grading Timeline
Homeworks will typically be graded and returned within 1 week. Projects will be graded within 2 weeks. Presentations will be graded within one week of the conclusion of all presentations. Exams will be graded within 1 week.

Additional Policies
Late assignments will not be graded, however as only the best 8 HW count towards the grade, there is some buffer to mitigate student stress in this regard. It is expected that students attend all classes in person. Any absences must be cleared with the faculty member at least a week in advance or by presenting Doctor's note in case of health emergencies. Student must work with faculty member to make up for any missed classes as soon as possible after the absence.

Course Schedule: A Weekly Breakdown

	Topics/Daily Activities	**Readings and Homework**	**Deliverable/ Due Dates**
Week 1 Dates	Embedded Microprocessors and Operating Systems Lab activity: Hello World on ContikiOS	Chapter 1 of Course Reader: overview of embedded OS; Contiki Tutorials 1-3 HW 1 - Introduction to ContikiOS	
Week 2 Dates	Sensing, ADC, Serial Protocols Lab activity: ADC	Chapter 2 of Course Reader: ADC, interrupts, serial protocols Contiki Tutorials 4-6 HW 2 - wiring a sensor	HW 1 due
Week 3 Dates	Radio Communications, Wireless Standards Lab activity: radio strength and packet loss measurement	Part 1 of Chapter 3 of Course reader: Wireless communications HW 3 - wireless radio standards	HW 2 due
Week 4 Dates	Schematic Design, Printed Circuit Board Design and Layout Lab activity: schematic design and PCB layout	Chapter 4 of Course Reader: PCB Design **Project 1: PCB design to integrate sensor, Microprocessor, and radio**	HW 3 due
Week 5 Dates	Medium Access and Scheduling Lab activity: CSMA and multi-channel link scheduling **Invited Industry Talk (20 mins + Q&A)**	Chapter 3 of Course reader, Part 2: Medium Access and scheduling	

Week 6 Dates	Sleep Cycling and Energy Utilization Lab activity: measuring energy utilization for computation and communication	Chapter 3 of Course Reader, Part 3: Energy Efficiency HW 4 - medium access and sleep	**Exam 1**
Week 7 Dates	Network Programming with TCP/IP Sockets Lab activity: running servers and clients	Chapter 4 of Course Reader: multi-hop networks and TCP/IP, additional reading on Socket Programming HW 5 - socket programming	HW 4 due, Project 1 due (PCB design project)
Week 8 Dates	Sensor Signal Processing **Invited Research Talk (20 mins + Q&A)** Lab activity: low pass filtering, RF localization	Chapter 5 of Course Reader: sensor signal processing for denoising compression, localization HW 6 - sensor signal processing	HW 5 due
Week 9 Dates	Mobile Device Programming Lab activity: Android Hello World	Additional reading on Android programming HW 7: Android programming	HW 6 due
Week 10 Dates	Actuation and Control Lab activity: Controlling lamp with light sensor	Chapter 6 of Course Reader: control systems to close the loop between sensing and actuation	HW 7 due
Week 11 Dates	Publish-Subscribe Middleware Lab activity: “hello world” on MQTT	Chapter 7 of Course Reader: publish subscribe middleware **Project 2: Distributed**	**Exam 2**

		system design using pub-sub messaging	
Week 12 Dates	Security: data encryption, authentication, transport-layer security Lab activity: use of symmetric and public key cryptography **Invited Industry Talk (20 mins + Q&A)**	Chapter 8 of Course Reader: Security HW 8: authentication and encryption for pubsub	in-class presentations
Week 13 Dates	Relational and Non-relational Databases Lab activity: "hello world" on SQL	Chapter 9 of Course Reader: Data Management HW 9: SQL queries	HW 8 due, in-class presentations
Week 14 Dates	Cloud Computing Lab activity: "hello world", simple computation on cloud server	Chapter 10 of Course Reader, part 1: Cloud Computing HW 10: Cloud computing	HW 9 due, in-class presentations
Week 15 Dates	IoT Cloud Platforms Lab activity: voice recognition on IoT platform **Invited Research Talk (20 mins + Q&A)**	Chapter 10 of Course Reader, part 2: Data Analytics and IoT Platforms	Project 2 due, HW 10 due, in class presentations
FINAL Date		**Final exam**	Date: For the date and time of the final for this class, consult the USC *Schedule of Classes* at www.usc.edu/soc.

Statement on Academic Conduct and Support Systems

Academic Conduct

Plagiarism – presenting someone else's ideas as your own, either verbatim or recast in your own words – is a serious academic offense with serious consequences. Please familiarize yourself with the discussion of plagiarism in *SCampus* in Part B, Section 11, "Behavior Violating University Standards" https://policy.usc.edu/student/scampus/part-b. Other forms of academic dishonesty are equally unacceptable. See additional information in *SCampus* and university policies on scientific misconduct, http://policy.usc.edu/scientific-misconduct.

Discrimination, sexual assault, intimate partner violence, stalking, and harassment are prohibited by the university. You are encouraged to report all incidents to the *Office of Equity and Diversity/Title IX Office* http://equity.usc.edu and/or to the *Department of Public Safety* http://dps.usc.edu. This is important for the health and safety of the whole USC community. Faculty and staff must report any information regarding an incident to the Title IX Coordinator who will provide outreach and information to the affected party. The sexual assault resource center webpage http://sarc.usc.edu fully describes reporting options. Relationship and Sexual Violence Services https://engemannshc.usc.edu/rsvp provides 24/7 confidential support.

Support Systems

A number of USC's schools provide support for students who need help with scholarly writing. Check with your advisor or program staff to find out more. Students whose primary language is not English should check with the *American Language Institute* http://ali.usc.edu, which sponsors courses and workshops specifically for international graduate students. *The Office of Disability Services and Programs* http://dsp.usc.edu provides certification for students with disabilities and helps arrange the relevant accommodations. If an officially declared emergency makes travel to campus infeasible, *USC Emergency Information* http://emergency.usc.edu will provide safety and other updates, including ways in which instruction will be continued by means of Blackboard, teleconferencing, and other technology.