IET Bachelor Projects
By Dr. Sherin Abdelhamid

The projects are easier to be developed in Python (https://www.python.org/) with the Django REST framework (https://www.djangoproject.com/).

P2P Resource Discovery for IoT Environments

Design a P2P discovery protocol that IoT devices can use to find nearby IoT resources (e.g., other IoT devices), connected over WiFi. The protocol should consider the limited power budget and constrained computational capability of IoT environments. The discovery functionality should be designed using the successful REST approach (i.e. using RESTful APIs) for interoperability and heterogeneity purposes. Students can inspire their design by the CoAP (http://coap.technology/) protocol. Evaluation should be carried out to evaluate the overall design as well as to measure the power consumption profile.

TEST CASE: a student must walk into an unknown environment and be able to discover nearby resources (e.g., WiFi cameras, smart devices, rest bands, etc.)

Towards Dynamic Access Control in IoT Scenarios

Develop a dynamic access control mechanism to authorize people's access to surrounding devices (e.g., AC control, printer, IoT lock). The use privilege changes according to dynamic context. For example, user U can actuate AC temperature on office A but can only view temp in research lab B. However, if the professor of lab B coexist, then user U can change the temp. Another generic example is: student S can have full access to a course material, but video materials are made available only for students who were present in class. In all cases, the user/or student does not have to reveal his/her identity for authentication purposes. It should be enough for the user to proof that he/she possesses required credentials (example, a member of the research lab in the first case and registered student in the 2nd case) -- using a zero-knowledge protocol [1].

References:

[1] N. Kahani, Khalid Elgazzar, Jim Cordy, “Authentication and Access Control in e-Health Systems in the Cloud,” The 2nd IEEE International Conference on Big Data Security on Cloud, New York, USA, April 9-10, 2016.

Towards Dynamic Context Management for Smart Systems

Context is a key enabler to smart interactions in IoT environments, where context collection, modeling, management and dissemination lie at the core of any smart application. IoT devices (especially smartphones) capitalize on their sensing capabilities to a variety of real-time context information to support context-aware systems making better decisions. Such context includes device profile (OS platform, device capabilities, embedded sensors, etc.), user profile (e.g., role, activities), environmental context (e.g., location, available networks, etc.). This project aims to design and implement a context representation model that strikes a balance between expressiveness and complexity.

Resource Management for Community-based IoT

There is a growing interest in community-based IoT deployments, in which a variety of devices (e.g., smart vehicles, smartphones, dash cameras, traffic cameras, etc.) participate to fulfil on-demand sensing requests (e.g., traffic updates, air quality in a certain geographical region, etc.). When a sensing request is received by the system, it matches all applicable sensors, based on their capabilities, to resolve the request. A key attribute to the success of such a deployment is to have the required critical mass of participating devices ready to contribute. To enable this paradigm, we need to maintain an active list of all devices along with their sensing capability to be able to perform runtime matching. This project will develop a platform that provides this physical resource infrastructure to support resource allocation mechanisms. The platform will expose the registration functionality as a RESTful API (using Django) so that mobile users of any platform can use the same API and register their resources. The platform will be able to communicate with the registered device to read all applicable sensing capabilities of the device. The registration process will extract these capabilities and display the results to the mobile user to confirm. The user can just hit “accept” or choose to unregister some capabilities based on preferences. By the end of the registration process, the system will store the device profile including all sensing capabilities and possible user preferences over each one (such as I enable my dash cam to participate, if I am in the downtown area and out of rush periods).

Mobile Resource Selection for On-Demand Sensing Tasks

There is a growing interest in utilizing mobile resources (e.g., smart vehicles, smartphones, dash cameras, traffic cameras, etc.) to fulfil on-demand sensing requests (e.g., traffic updates, air quality in a certain geographical region, etc.). Such mobile sensing resources are monitored by a centralized system that handles the task-resource assignment. When a sensing request is received by the system, it matches all applicable sensors, based on their capabilities, to resolve the request. This project aims at developing an algorithm for selecting the best candidate devices “from too many” to fulfil a certain on-demand request. The algorithm should first match existing devices to request requirements based on the device capabilities. The result of this stage is a list of candidate devices. This list may be too large (over provisioning) or inadequate (cannot fulfil the request). If the list is too large, the algorithm should include a mechanism for selecting the best set from the large candidate set.