Title	Decentralized Orchestration of IoT in End-user Programming Environments
Student	Margarida Silva FEUP-MIEIC-2019-20
Supervisors	Sereno, Restivo
Scope of the work	The appearance of Internet-of-Things as the peak of ubiquitous computing and connectivity is the result of having Internet-connected devices everywhere capable of sensing and actuating in the real world. Developing IoT systems is challenging due to a set of particular characteristics of IoT systems, including heterogeneity, dynamic topologies and highly-distributed nature. One of the approaches for developing IoT-based systems has been visual programming. One of the most widespread visual programming solutions tailored for IoT is NodeRed. "NodeRed is an open-source flow-based development tool for visual programming developed originally by IBM for wiring together hardware devices, APIs and online services based on Node.js". However, NodeRed is no "silver-bullet" for IoT development and have several issues/gaps when compared with the development ecosystems of other languages. One of those issues is that NodeRed is a completely centralized system that doesn't leverage the computational resources available in the IoT devices. For example, if there's a "flow" computing the temperature average over the last 10 reads, even if the uController present in the temperature sensor is perfectly able to do this calculations, the sensor *will* communicate every reading back to NodeRed. This violates several principles of current IoT desiderate, including the notion of Local-First (that data and logic should reside locally and should work even when third-party services are offline). It also hinders several quality attributes of the system, including (but not limited to), resiliency, scalability and fault-tolerance. Put simply: even if the server crashed, I should have hot water in the morning.
Goals and expected results	The goal of this work is to develop a prototype (extending, or rewriting NodeRED), such that IoT devices are capable of communicating "computational capabilities" back to the orchestrator, and the orchestrator is able to compute "recipes" and send them to the nodes, leveraging their computational power and independency. As a secondary goal, several other challenges are to be tackled, viz: (i) inferring computational boundaries, (ii) detecting non-availability and using alternative computational resources, and (iii) exploring different alternatives of leveraging current

	IoT devices, including firmwares that are able to execute arbitrary programs written in Lua, Javascript, Python, etc., amongst others.
Innovative aspects	There is currently no solution in the market that answers the above desiderata.
Draft of work plan	First semester: – state of the art & problem analysis Second semester (full time): – 2 weeks: enumeration of the key research questions related with a decentralized orchestration of IoT; – 10 weeks: analysis, development and refinement of new and already existing tools and applications for a decentralized orchestration of IoT; (including 2 weeks writing of dissertation report). – 6 weeks – testing, validation, and refinement of the tools and applications (including 2 weeks writing of dissertation report).
References	* Ashton, Kevin. "That 'internet of things' thing." RFID journal 22, no. 7 (2009): 97-114. * Atzori, Luigi, Antonio Iera, and Giacomo Morabito. "The internet of things: A survey." Computer networks 54, no. 15 (2010): 2787-2805. * Node-RED, https://nodered.org/ * Tanimoto, Steven L. "A perspective on the evolution of live programming." In Proceedings of the 1st International Workshop on Live Programming, pp. 31-34. IEEE Press, 2013. * Dias, João Pedro, João Pascoal Faria, and Hugo Sereno Ferreira. "A Reactive and Model-Based Approach for Developing Internet-of-Things Systems." In 2018 11th International Conference on the Quality of Information and Communications Technology (QUATIC), pp. 276-281. IEEE, 2018.