2. TECHNOLOGIES OF INTERNET AND WEB OF THINGS

Ph.D., Prof. V.A. Tkachenko (NTU “KhPI”),
DrS, Prof. Y. V. Brezhniev (KhAI)

Contents

Abbreviations

6LoWPAN — IPv6 Low Power Wireless Personal Area Networks
AJAX — Asynchronous JavaScript and XML
ALIOT — Internet of Things: Emerging Curriculum for Industry and Human Applications
AMQP — Advanced Message Queuing Protocol
API — Application Programming Interface
BLE — Bluetooth Low Energy
BSON — Binary JavaScript Object Notation
CoAP — Constrained Application Protocol
DBMS — Database Management System
DDS — Data Distribution Service
DTLS — Datagram Transport Layer Security
ETSI — European Telecommunications Standards
IEEE — Institute Institute of Electrical and Electronics
IETF — Engineers Internet Engineering Task Force
IoE — Internet of Everything
IoP — Internet of People
IoT — Internet of Things
JMS — Java Message Service
jQuery — JavaScript Library
JS — JavaScript
JSON — JavaScript Object Notation
KhAI — Kharkiv Aviation Institute
KhPI — Kharkiv Polytechnic Institute
LoWPAN — Low-Power Wireless Personal Area Networks
LPWAN — Low-Power Wide-area Network
M2M — Machine-to-Machine
MEAN — MongoDB, Express.js, Angular.js, Node.js Mongo DB — Humongous Database
MQTT — Message Queue Telemetry Transport
MySQL — Free relational DBMS
NAU — National Aerospace University
NFC — Near field communication
Node.js — JavaScript run-time environment

NoSQL — Not Only SQL
NTU — National Technical University
OAuth 2.0 — Open Authorization Protocol
OCF — Open Connectivity Foundation
OGC — Open Geospatial Consortium
OpenSSL — Cryptographic Library (SSL/TLS) with Open Source
PaaS — Platform as a Service
PETRAS — Privacy, Ethics, Trust, Reliability, Acceptability and Security
PostgreSQL — Free object-relational DBMS
Pub/Sub — Publisher/Subscriber
REST — Representational State Transfer Radio
RFID — Frequency IDentification
SOA — Service-Oriented Architecture
SQL — Structured Query Language
SSE — Server Side Events
SSL — Secure Sockets Layer
TLS — Transport Layer Security
WLAN — Wireless Local Area Network
WoT — Web of Things
WPAN — Wireless Personal Area Network
WSN — Wireless Sensor Networks,
WSS — WebSocket Security
XML — eXtensible Markup Language
XMPP — Extensible Messaging and Presence Protocol

2.1 Internet and Web of Things

2.1.1 Conception of Web of Things

Development of technology M2M/IoT, information processing tools (Big Data) and decision-making (Cognitive Analytics) lead to changes in the technological, economic and social development models society. Areas of use of IoT are expanding in energy, transport, medicine, agriculture, housing, Smart City, Smart Home, etc. IoT focuses only on connecting physical objects to the network and their interaction with each other. Cisco introduced a new concept - Internet of Everything (IoE), which is based on the integration of people, things, data and processes. Thus, the next stage in the development of IoT / WoT is the Internet of all (IoE). In the future, the orbit of IoT will include technology deep machine learning, artificial intelligence, technology blockchain, robotics, etc. IoT is characterized by large changes in the infrastructure of the Internet and new communication models Smart Things or connections: "Thing-Thing", "Thing-User" and "Thing-Web Object". IoT Infrastructure consists of various networks of physical objects based on heterogeneous hardware and software platforms, protocol stacks, which are generally incompatible with each other. So the IoT is a collection of isolated physical networks that cannot communicate with each other via the Internet.

The concept of a WoT based on Web and its new technologies [1], enables the integration of all kinds of Smart Things and applications with which they interact. The concept WoT introduced such a notion as "Web Thing", which is a digital representation of a physical or virtual object that is accessible through Web API RESTful. One of the major development issues for this new concept is creating efficient hypermedia-enriched application programming interfaces (APIs) [2]. Web API RESTful or Web API built with consideration of the REST architecture for a virtual representation of the physical objects are identified by URL and use application layer protocols such as HTTP, WebSocket, CoAP, MQTT in JSON format, and TLS / DTLS cryptographic streaming protocols. Thus, the virtual equivalent of physical objects (Web Thing), which were assigned a URL via Web API, can communicate with each other or with applications by using application-level protocols and share data in text-based JSON. In addition to the software interface Web Thing can be equipped with custom interfaces to ensure

interoperability model “Thing- User”. The WoT reuses existing and well-known Web standards used in the programmable Web (e.g., REST, HTTP, JSON), semantic Web (e.g., JSON-LD, Microdata, etc.), the real-time Web (e.g., Web Sockets) and the social Web (e.g., OAuth or social networks). [3].

Thus, WoT provides the integration of Internet-connected physical devices of different producers on application level regardless of how they are connected on a network level and ensures the creation of a single global ecosystem of the IoT, which is open and compatible.

Currently WoT standardization has engaged WoT community and such international organizations for Standardization, as W3C (https://www.w3.org/WoT/), IETF (https://www.ietf.org/), ETSI (http://www.etsi.org/), OCF (https://openconnectivity.org/) and OGC (http://www.opengeospatial.org/), and supported by European research projects on the IoT, such as Sensei-IoT (http://www.sensei-iot.org/), SmartSantander (http://www.smartsantander.eu/) and IoT-A (https://iota.org/). WoT Interest Group published draft standards [4, 5, 6, 7, 8]. In addition to set out draft standards, Mozilla IoT community published its draft standards [9].

Together with the development of IoT technology and its network service WoT, there is an increasing need for specialists for development (software and hardware) and integration of technical solutions in the field of IoT, maintenance and operation of IoT networks. The problem of training (training and retraining) of IoT / WoT specialists is becoming urgent. The issue of training current and future engineers and researchers, technology application development and integration of modern IoT/WoT-solutions can be solved jointly by the companies that design and manufacture tools for IoT and institutions of higher education. For example, the aim of the project ALIOT [10] is integration of all available and prepared training programs, manuals and tools for the provision of training and advisory services in the field of systems based on the IoT for applications in different areas.

Companies involved into developing and manufacturing tools for IoT, are interested in higher-education-obtained professional skills creation and exploitation of IoT/WoT in a timely manner, in order to remain competitive in the field of development and production of IoT/WoT. In their turn higher education institutions are interested in teaching students the basics of the IoT/WoT design and operation to be competitive on the labor market in the field.

2.1.2 State of Art in Web of Things

It should be noted that companies that develop and produce tools for IoT and universities prepare specialists in the field of development and integration of modern IoT-solutions. For example, the company has created a University for Telit IoT [11]. The program Telit IoT University currently includes six courses, one of which is the IoT for Developers. In IoT University course [12], students look at User Interface and User Experience design strategies common to the industry and apply those strategies to building applications in ThingWorx using the Mash Up Builder. This course is focused on IoT-project, which does consider the WoT technologies. The PTC IoT Academic Program [1] consists of the ThingWorx™ application enablement platform in a PTC hosted environment where students and educators can build their own IoT applications. PTC works with corporate customers as well as market partners to ensure that students from all disciplines are better prepared to meet the needs of today's IoT world.

In the article [14] the information is presented on many bachelor's and master's programs on IoT. WoT technology is not considered. The IoT MSc program [15] is available at the Queen Mary University of London. MSc Internet of Things (Data) is currently available for one-year full-time study, two years' part-time study (Introduction to IOT, Enabling Communication Technologies for IOT). WoT technology is not considered. The article [16] presents the best universities that offer courses in the field of "Internet of Things", and study in detail what they offer their students. WoT technology is not considered.

The project "IoT Academy Samsung" [17] is organized on the base of Moscow Institute of Physics and Technology. In accordance with the experts of the research center of Samsung's teaching materials students will undergo a year-long training course on examining real case studies on Internet of things technologies in various industries and will be able to create their own IoT devices prototypes. WoT technology is not considered. The Cisco Internet of Things (IoT) [18] certifications and training are job-role-based programs designed to help meet the growing need for specialized talent. This education portfolio provides Internet Protocol (IP) networking expertise, with a focus on automation, manufacturing and energy and future expansion to

include equally transformative industries. WoT technologies are not considered.

National Aerospace University "KhAI" and other Ukrainian universities prepare specialists on programmable mobile systems and IoT [19]. IoT-based systems are developed and investigated as well. WoT technologies are not considered in the discipline of "Industrial Internet of Things (IoT). Lviv IT Cluster and National University "Lvivska polytechnica" have launched a Bachelor program "Internet of things" [20]. Goal is to prepare specialists in the field of designing elements and applications of IoT, WoT is not considered. The work [21] deals with possible reflection of the theme of the Internet of things (Internet of Things) and machine-to-machine (M2M) in higher education curriculum (programs), but it does not reflect the technology of WoT. The purpose of such a curriculum is to consider issues related to information and communication technologies used in IoT and M2M. The proposed course aimed at listeners acquainted with modern information technologies, which stand behind such directions as inter-machine interaction and Internet of things.

Currently the authors have not seen in open papers mentions of any existing training programs on teaching students about technology for Web Things application development. Practical guide "Building the Web of Things" [1] is a basic training manual, which presents key technologies and concepts necessary to build application-level IoT and Architecture WoT, as well as define the methodology of application development for the Web Things on JS/Node.JS. This manual is intended for trained professionals in the field of Web application development technologies at JS/Node.JS.

Step-by-step tutorial can help professionals use WoT and semantic network to develop applications of Semantic WoT [22], but does not solve the problem of student learning of all the necessary technologies for the development of WoT. The book "Web application development Framework" [23] is designed for inexperienced Web developers, it outlines the creation of user interfaces, discusses ways to develop the server-side application on Node.JS, and methods of cloud services usage for deploying Web applications. The book "Web development with Node and Express" [24] is intended for programmers who want to build Web applications (regular sites that embody REST application programming interfaces) using JavaScript, Node. JS and Express. The curriculum "Technology and development tools WoT applications" was prepared with consideration of all teaching materials,

that are prepared for programmers' learning on development technologies of advanced Web applications based on the JS/Node.JS, that will give the listeners the starting point for the WoT concept development and developing real applications Web Things.

2.1.3 Goals of the Web of Things course

The purpose of this work is to review IoT/WoT technology, curriculum (case-studies of WoT) and discus of structures tested in NTU "KhPI", the working curriculum of subjects "WoT application development technologies and tools", drawn up in the light of developing Web Thing API technologies proposed in the concept of WoT. This program offers technology development of WoT-applications, that is, the stack of MEAN technologies (Mongo, Express, Angular, Node) for development of WoT applications using JavaScript/Node.JS. Duration of training - one semester, the course is designed to train professionals (for master students) in "Computer Science", which is intended to form the students' theoretical knowledge and practical framework in design and exploitation of WoT-applications.

The task of the study course "Technology and tools for developing Web applications" is a theoretical and practical training of future specialists on such matters as:

– technologies for the application of markup languages, languages for description and programming in client Web applications;

– technology and tools for creating interactive Web interfaces;

technology application in Node.JS server applications;

– technology exchange messages between Web Apps in a mode of Real Time (Ajax, WebSockets) in XML messaging formats, JSON;

– technology for building applications with SOA architecture (architecture, REST);

– cloud computing technology and application deployment model for cloud platforms;

– architecture and technology of IoT;

– cloud platforms and services for the Web of Things;

– technology for application development based on Web based Things Raspberry Pi using the Node.js;

– security, privacy, and access control to the physical devices in the IoT/WoT.

The structure of this chapter is the following: Introduction; Survey of IoT/WoT Technologies; Structure of the training program "Technologies and tools for developing WoT applications"; Conclusions; References.

2.2 Survey of Internet and Web of Things technologies

Existing M2M-technologies allow machines to exchange information with each other. M2M is a subset of IoT. IoT is the Internet of People (IoP), extended by computing networks of physical items (Smart-Things), which can independently organize various connection models. IOT is a concept of the network infrastructure development (physical basis) online, in which "smart" things without human intervention are able:

- to connect to the network for remote interaction with other devices (Thing-Thing);
- to interact or interaction with autonomous or cloud data processing centers, or DATA centers (Thing-Web Objects) for data transmission, storage, processing, analysis and management decisions aimed at changing the environment Wednesday:
- to interact with user terminals (Thing-User) for the control and management of these devices [25].

The article [26] is the first to present the correlations among machine-to-machine (M2M), wireless sensor networks (WSNs), cyber-physical systems CPS and internet of things (IoT). The authors suggest that CPS is an evolution of M2M by the introduction of more intelligent and interactive operations, under the architecture of IoT.

Cisco believes [27] that The Internet of Everything is the next step in the evolution of smart objects-interconnected things in which the line between the physical object and digital information about that object is blurred. The WoT is a refinement of the IoT by integrating smart things not only into the Internet (network), but into the Web Architecture (application) [28].

2.2.1 IoT global network architecture

The papers provide overviews of the IoT: concepts, architectures, development technologies, physical devices, programming languages, protocols, and application [29, 30, 31, 32, 33]. The Internet of things consists of the networks of physical objects, the traditional network of the Internet and

various devices (Gateway, Border router, etc.) that connect these networks. Figure 2.1 presents the components of the IoT architecture, which consists of several computer networks of physical objects connected to the Internet.

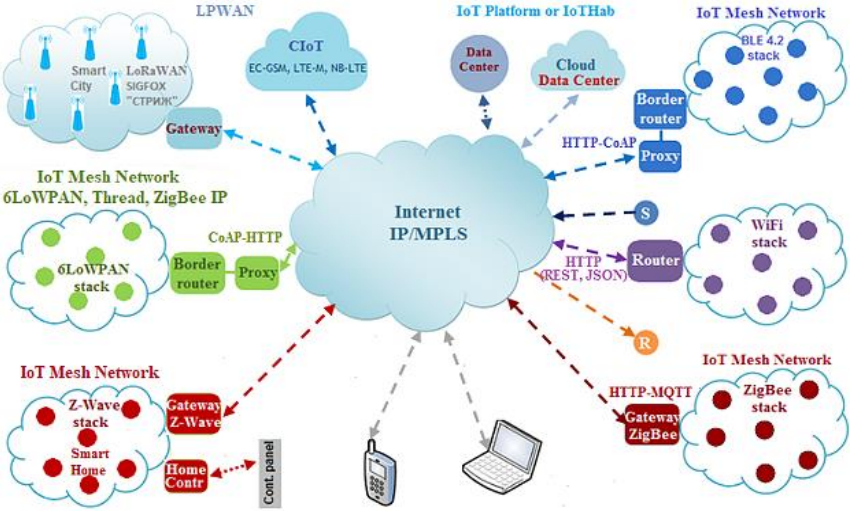

Figure 2.1. Components of IoT Architecture [25]

As seen of Figure 2.1, the network of Internet of things consists of: the computer networks of physical objects (Smart Objects), traditional IP Internet and various devices (Gateway, a Border router, Router), integrating these networks. It should be noted that Smart Objects are the sensors or actuators (sensors or actuators), equipped with a microcontroller with real-time operating system with a stack of protocols, memory, and communication device embedded into various objects, such as in electricity or gas meters, pressure sensors, vibration or temperature switches, etc. Smart Objects can be organized in computer network physical objects that can be connected via gateways (hubs or specialized IoT platform) to the traditional Internet.

In IoT there is not a single universal protocol for the integration of physical objects. Therefore, to create a network of physical devices, one shall acquire all the components of one manufacturer. As a result, the network of physical objects is fragmented and the provision of integration of physical devices connected to the Internet with incompatible protocol stacks is expensive.

Gateways are used to integrate the networks of IoT (for example, Z-Wave, ZigBee etc.), protocol stacks, which are incompatible with the TCP/IP stack of the Internet. Edge routers are used to integrate the Internet with networks of IoT, based on network protocol 6LoWPAN (IPv6 over Low-Power Wireless Personal Area Networks), where IPv6 is a version for wireless personal area sensor networks with low power consumption IEEE 802.15.4.

A proxy is used to harmonize protocols HTTP- CoAP. Technology-based network Thread, ZigBee, 6LoWPAN self-organizing nature are IP networks, and may not have an exit to external IP network using 6loWPAN protocol stack for the Organization of the work of autonomous networks and data transmission between the autonomous network nodes.

Wireless networks, used in LPWAN: IoT; WLAN and WPAN. LPWAN Technology. Key long-range LPWAN networks (Low-power Wide-area Network) technologies include: LoRaWAN; SIGFOX; Swift; CIoT (EC-GSM, LTE-M, NB-IoT). According to experts' estimates, more than 50% of IoT solutions would use LPWAN network. WLAN Technology. Medium-range technology, WLAN refers Wi-Fi (www.wi-fi.org) - a set of wireless standards IEEE 802.11, which can be used to build a wireless local area WLAN based network objects on the TCP/IP stack.

To build local wireless computer networks items, Wi-Fi Alliance has created a new IEEE 802.11 specification, which provides technology to build cellular networks. In addition, new standard Wi-Fi HaLow (IEEE 802.11 specification ah for the IoT) was created with low power consumption. Wireless personal area networks (WPAN). Key WPAN short-range wireless networks technologies: 6LoWPAN, Thread, ZigBee, Wireless IP, Z-Wave, EnOcean, RFID/NFC, BLE 4.2. Controllers and mini computers in the IoT. Today to manage the physical devices, the IoT uses controllers and mini computers: Arduino, Espruino, Tessel, Intel Edison and Galileo, Raspberry Pi, whose applications are created in c/C++, Java, JavaScript, Python, etc.

IoT application layer protocols. In the networks of physical objects, the interaction between components is done using the application layer protocols: DDS [34], CoAP, MQTT, XMPP, AMQP, JMS, REST/HTTP [35], etc. DDS is the core technology for Industrial IoT. CoAP Protocol (Constrained Application Protocol) - limited data transfer protocol similar to HTTP, but adapted to work with “smart” devices. MQTT protocols, XMPP, AMQP,

JMS - these messaging protocols are based on broker scheme: publish/subscribe.

Security considerations for IoT. Security of IoT must be addressed at all stages of the development cycle and operation of hardware and software, communication channels, protocols stack, cloud components etc. is currently given a lot of attention in the field of IoT security. In the paper [36], several security and privacy concerns related to IoT are mentioned. The protection of data and privacy of users has been identified as one of the key challenges in the IoT.

The survey presents Internet of Things with architecture and design goals. In addition, a review and analysis of security and confidentiality issues at different levels in the IoT was performed. It should be noted that for the security of the IoT, standards [37] and guidance [38] have been created that provide manufacturers of tools with a set of guidelines for improving IoT security.

The document "State-of-the-Art and Challenges for the Internet of Things Security" [37] can be used by implementers and authors of IoT specifications as a reference for details about security considerations while documenting their specific security challenges, threat models, and mitigations. The goal of guidance [38] is to help manufacturers build more secure products in the Internet of Things area. Royal Academy of Engineering (London) [39] is a leader in Cybersecurity of the IoT. The PETRAS Cybersecurity of the IoT Research Hub brings together nine leading UK universities.

The development of IoT depends on many factors: technology, low-power wireless networks; Smart Objects technology; the pace of 5G networks adoption; operating systems for microcontrollers sensors and actuators; widespread use of 6LoWPAN/IPv6 protocol stack; M2M technology; effective use of Cloud computing for IoT platforms; Misty technology computing (fog computing) and Software-Defined Networks; ensuring hardware and software cyber resilience.

2.2.2 Technology Web of Things

IoT focuses on the lower layers of the network stack, and the WoT service on the upper layers, application tiers. By using web technologies, protocols, programming languages and formats [40] such as REST, XML, JSON, MQTT, XMPP, Atom, WADL, Open ID and OAuth, the WoT has contributed to reducing the barriers for

common understanding and smooth interplay between heterogeneous real world devices, services and data.

The WoT concept, based on the Web and its new technologies [1], provides integration of all types of Smart Things and applications with which they interact. It is known that WoT uses standards applied in such technologies as programmable Web (HTTP, REST, JSON), semantic Web (JSON-LD, Microdata, etc.), Real-time Web (WebSockets) and social Web (OAuth or social networking APIs). The problems of the WoT architecture, development technologies, programming languages, APIs, application-level protocols based on RESTful principles are described in many articles [41, 42, 43, 44].

Thus, WoT provides integration of devices in the Internet. The WoT is a service similar to the IoT infrastructure service, World Wide Web, of the Internet infrastructure. WWW is a distributed information system based on the use of hypertext documents in HTML format, access and transfer of which are achieved using the HTTP application. WoT is an extended service Web.

By analogy with the Web architecture, the architecture of the WoT is the World Wide Web or distributed system of Web Things virtual resources (virtual representations of Things) that provide access to the physical objects, i.e. applications that are hosted on Smart Objects or intermediate IoT network devices through Web Thing API.

The essence of WoT is that Web Thing physical objects or intermediate gateway devices, given that they have their own URL (Web-address) and software interface with the RESTful Web API can communicate in text-based JSON both with each other and with applications based on SOA. To ensure interaction model "Thing- User" Web Things applications must have user interfaces. Due to limited resources, not all Web Things can offer their own Web API is RESTful, based on the concept of WoT.

For integration of Smart Objects in The Internet three different integration templates are provided:

- Direct Connectivity,
- Gateway Based Connectivity,
- Cloud Based Connectivity [1].

Implementation of Web Thing API on its own platform can be performed on the basis of a Web server that is hosted on the controller embedded in Things.

Web application for the Web Things can consist of frontend and backend, i.e. can be implemented as user interfaces for users to interact with Things via Web browsers (for example, site sensors: http://devices.webofthings.io/pi/) and mobile applications and interfaces (API) applications using the RESTful architecture for data exchange between devices. As a controller, you can apply, for example, single board computer Raspberry Pi based on Linux.

The latest version of the computer (Raspberry Pi 3 Model B) has a built-in support for Wi-Fi and Bluetooth 4.1. In addition, Raspberry Pi GPIO ports available for direct connection to devices (e.g. temperature sensors, displacement, etc.). To implement Web Server Node.js can be applied (for example, Node v 7.10.1).

To develop the server-side application it is advisable to choose the programming language JavaScript in Node.js. Software code of client side of the application is developed in HTML, CSS and JavaScript. For of data exchange between devices in application the interfaces of Client APIs and API Server, built with consideration of the REST architecture, are implemented.

Figure 2.2 presents Web API Thing, which can be placed either directly on the device itself, and intermediate Web Things gateway network or in cloud service.

In the case of implementation of a Web Thing API on an intermediate device such as a gateway, you can use the prototype gateway Things Gateway, Figure 2.3.

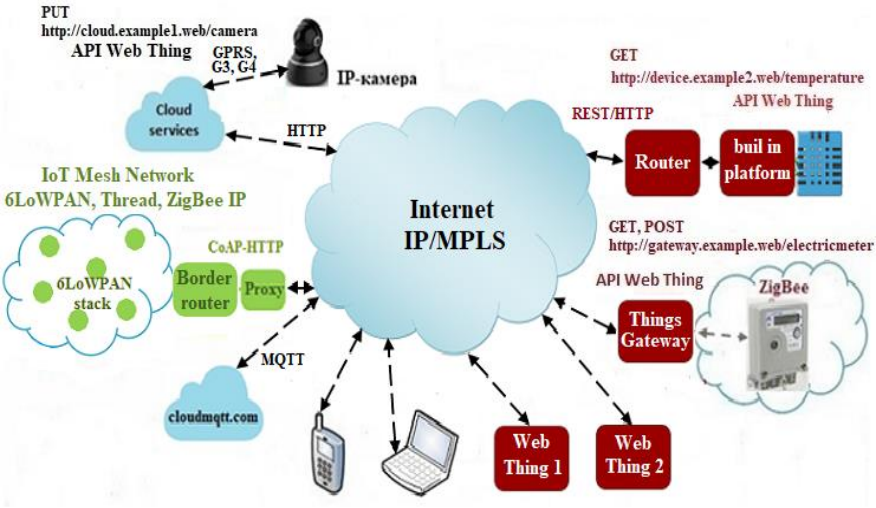

Figure 2.2. Components of WoT Architecture [45]

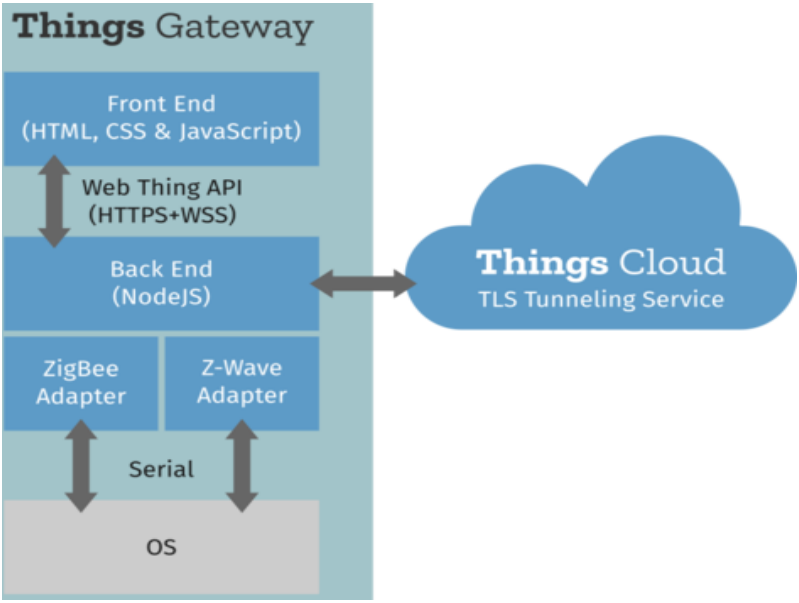

Figure 2.3. Things Gateway [46]

Things Gateway, Figure 2.3, is created by the developers of the Mozilla community in JavaScript using Node.js server platform, and is available as a ready to install on the Raspberry Pi board assemblies. If you implement a Web API Thing on cloud server, Web Thing Clients (Web devices or users) communicate to cloud-based server (cloud-based server analogous Things Gateway) by domain address of devices, which runs the application hosted on that server, and the application accesses devices, such as a camcorder, and manages them.

The EVRYTHNG Platform [47] is a cloud Platform-as-a-Service (PaaS) for storing, sharing, and analyzing data generated by physical objects. The Platform gives a unique and permanent digital identity (also known as ADIs) to each individual object and allows authorized applications and users to access it via REST and Pub/Sub (MQTT) APIs.

Security in WoT is provided by certificates, encryption and authentication. Cryptographic streaming protocols TLS/DTLS [48, 49]

are the basis of secure HTTP protocols (HTTPS), WebSocket (WSS), MQTT (MQTTS) and CoAP, which are used in WoT. To do this, you can install the OpenSSL library on the server (sudo apt-get install openssl). In addition, you can apply recommendations and methods for authorizing and controlling access to the server.

Authentication is one of the means of protecting WoT applications [50, 51, 52, 53]. You can set the API token with Node.js (install Node.js: node-oauth2-server). Authentication OAuth2 is designed to protect the Web API using a token-based authentication process. The token will be used to authenticate Smart Objects for each request to the server. You can use the OAuth 2.0 social media tools for WoT authentication.

2.3 Training on technologies and tools for developing WoT applications

All of the technologies outlined in the proposed themes are used in the development of Web Thing applications in line with the concept of WoT.

Thus, the training program proposed in Table 2.1 is designed to prepare future specialists develop real Web Thing applications.

Table 2.1. Curriculum structure

The topic	Content
Development tools for Web applications: IDE, browser, version control system.	At present, the only IDE for creating client and server applications on JS/Node.js is WebStorm [54]. But the own IDE (for JS development) could be built based on a text editor Sublime [55] with the plugins. In this program Git [56] is used as a version control system (VCS) of the Web application files, Git is used in many famous projects as VCS. In addition, familiarity with Git gives students the opportunity to explore GitHub, the largest Web service for hosting IT projects and their joint development.

HTML and XML technologies in client-side Web applications.	HTML document structure, logical languages HTML5 markup [57] and XML [58], technologies used in creating layouts or templates on HTML5 for Web sites, technology HTML-layout technology in editor Sublime Text.
CSS Technology And CSS3 in client Web applications and the use of the Bootstrap framework.	CSS [59] - language for describing the appearance of documents (style declaration, selector types, block and line elements, style preprocessors, CSS frameworks and Emmet LiveStyle). The layout technology of a web application with adaptive design based on Bootstrap [60] in the Sublime Text editor.
Technology and tools for creating interactive Web interfaces.	To develop interactive Web-interfaces, one must apply the basic triad HTML technologies, CSS and JavaScript [60, 61], which form the structure, style, and behavior of Web applications. One of the components of the triad technologies is: the JavaScript programming language (syntax, set of technologies for creating interactive Web applications with JavaScript).
Technology of using jQuery to create interactive Web interfaces.	jQuery [63] is a JavaScript-based library that contains ready-made JavaScript functions. jQuery manipulates the html elements of the document and uses the DOM to change its structure. There are two methods for connecting the jQuery library to the client application: local and remote connections. JQuery has a large number of third-party plug-ins with which make it possible to significantly improve the interface of the client side Web-applications or WoT.

The exchange of messages between Web Apps in a Real Time mode (Ajax).	Four network technologies for interacting Web Apps based on client-side JavaScript scenarios (AJAX, COMET, SSE, WebSocket). This topic is dedicated to AJAX [64] (Ajax technology and data transfer formats, ajax requests for "pure" JavaScript; ajax and jQuery).
Exchange of messages between web apps in Real Time mode (WebSocket).	WebSocket [65] – is a technology of asynchronous interaction between the Web client and the Web server. WebSocket is a protocol of full-duplex communication over a TCP connection, intended for the exchange of messages between a web client and a web server in real time. WebSockets have an API that can be used in web applications and is called the WebSockets API [66, 67].
Web servers and application servers.	It is proposed to consider the HTTP protocol, the client/server model, the architecture of the Web server, application server, as well as to form an idea about the technology of these tools. In addition to traditional Web servers, Node.js technology is considered [68], which enables to create event-driven servers using JavaScript.
DBMS. Technology and software for creating databases.	There are 6 data models: lists (flat) relational databases, hierarchical, network structures, object-oriented databases and document-oriented data model. Currently, they are the most widely used when designing a relational database model (MySQL, PostgreSQL, MSSQL Server). It should be noted that the most popular database management system for Node.js is currently the MongoDB [69], which is a NoSQL. MongoDB is a document-oriented management system (DBMS) open source software. MongoDB is a new approach to build databases without SQL queries, tables, foreign keys, etc. In MongoDB, JavaScript is used as the query language, the data is stored in the BSON format, i.e. binary JSON.

WebRTC - is a technology for creating Web communications applications.	WebRTC [70] is an open source technology for building peer-to-peer networks, which allows to send text and multimedia data directly between browsers. Signaling server is used only for setting up p2p connection between the two browsers. The WebRTC technology is implemented by three JavaScript APIs: RTCPeerConnection, Media Stream (getUserMedia), RTCDataChannel.
Cloud technologies-development tools for Web applications and messaging service in Real Time mode.	Cloud computing [71] is the delivery of computing services - servers, storage, databases, networking, software, analytics and more—over the Internet ("the cloud"). Companies offering these computing services are called cloud providers and typically charge for cloud computing services based on usage, similar to how you are billed for water or electricity at home.
Technologies for creating applica-tions with SOA architecture and use of SaaS with APIs.	A service-oriented architecture [72] is essentially a collection of services. These services communicate with each other. The communication can involve either simple data passing or two or more services coordinating some activity. Some means of connecting services to each other is needed.
IoT Technologies.	The Web of Things [29, 30, 31, 32, 33, 34, 35] is a refinement of the Internet of Things by integrating smart things not only into the Internet (network), but into the Web Architecture (application). In this section, it is necessary to consider: the IoT architecture, controllers and mini computers, the IoT platform, application programming languages, wireless network technologies, protocols, IoT security issues.

Protocols and technologies of creating Applications for Web Things based on Raspberry Pi 3 Model B using the Node.js platform.	The Semantic Web [1, 40]. The Web of Things [41, 42, 43, 44] is a high-level application protocol designed to maximize interoperability in the IoT. The WoT architecture stack is not composed of layers in the strict sense, but rather of levels that add extra functionality. Each layer helps to integrate Things to the Web even more intimately and hence making those devices more accessible for applications and humans. The following shall be considered in this section: Linux-based mini computers; versions of Raspbian for Raspberry Pi; implementation the Web Thing API for the Direct Connectivity integration template and for the Gateway Based Connectivity integration template.
Cloud platforms and services for the Web of Things.	Data Analytics. The Cloud Based Connectivity integration template allows Web platform to act as a gateway to implement API Web Thing on the staging device. EVRYTHNG platform [73] is a cloud-based platform-as-a-service (PaaS). The platform provides a unique and permanent identifier for each individual object and enables authorized applications and users to access it via the REST API and Pub/Sub (MQTT). It is proposed to consider the technology for implementing of the Web Thing API for the Cloud Based Connectivity integration template.
Security, privacy, and access control to the physical devices on the Internet.	Security in IoT [36, 37, 38] should be provided at different levels of the network. Security in WoT is provided by certificates, encryption and authentication [48, 49, 50, 51, 52, 53]. The security issues: protecting Web Thing (encryption, enable HTTPS, WSS and TLS on the server); authentication and access control; the use of social networking tools OAuth 2.0 for WoT-authentication.

NTU “KhPI Information Systems Department, trains students in the specialty “Computer Science” according to the curriculum presented in Table 2.1.

Figure 2.1 and Figure 2.2 shows the implementation materials of the web interface API Web Thing for the integration template Direct Connectivity. From the one shown in Figure 2.1 it follows that the HTTP and WebSocket servers are functioning. The Web Thing application installed in the Node.js environment on the Raspberry Pi 3 Model B provides data on the functioning of the motion sensors (PIR sensors), temperature and humidity (DH22) and the actuator (LED 1). Data on temperature, humidity and movement is constantly updated.

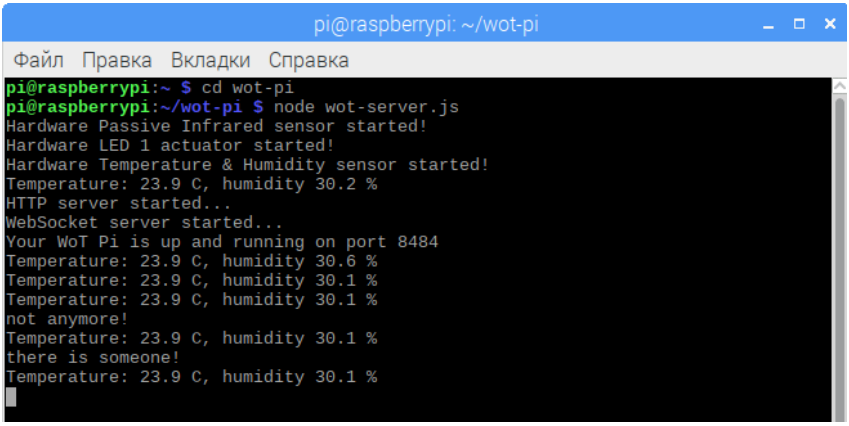

Figure 2.1. A screenshot of the application's operation Web Thing for the Direct Connectivity integration template

According to the Figure 2.2, on request you can view physical devices parameter values in a Web-browser, for example, the values of the temperature sensor.

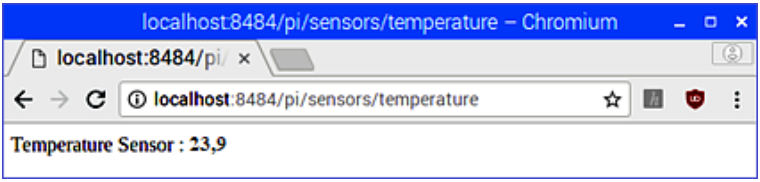

Figure 2.2. A screenshot displaying the values of the temperature sensor in the Web-browser

2.4 Work related analysis

This work reviews the existing technologies in the IoT / WoT, for which draft standards are being developed. At the 8th International Workshop on Web Objects (WoT 2017), it was noted that the REST architecture was the de facto basis for building the software interface of intelligent physical objects connected to the Internet. WoT, based on the Web and its new technologies, provides integration of all kinds of Smart Things and applications with which they interact, and transforms the world of physical objects into a distributed information system. Nowadays a creation of RESTful Web API for Web Thing on JS in Node.JS is preferred over other programming languages. In this regard, the proposed curriculum technologies are relevant and aim to prepare future professionals to develop real Web Thing applications. The technologies mentioned are tested in the teaching process of students.

The industrial IoT is part of IoT. The IoT is a network of computers, devices, and objects that collect and share the industrial data. It allows operating of industrial systems in more efficient and safe manner. Besides, industrial IoT is used for smart grid to improve efficiency of power generation and distribution. The many challenges and risks caused by industrial IoT shall be addressed by development of methods and tools. This will be done under the course that is to be developed by National aerospace university.

It should be noted that companies that develop and produce tools for IoT and universities prepare specialists in the field of development and integration of modern IoT-solutions. For example, the company has created a University for Telit IoT [11]. The program Telit IoT

University currently includes six courses, one of which is the IoT for Developers.

In IoT University course [12], students look at User Interface and User Experience design strategies common to the industry and apply those strategies to building applications in ThingWorx using the Mash Up Builder. This course is focused on IoT-project, which does consider the WoT technologies.

The PTC IoT Academic Program [13] consists of the ThingWorx™ application enablement platform in a PTC hosted environment where students and educators can build their own IoT applications.

In the article [14] the information is presented on many Bachelor's and Master's programs on IoT. The IoT MSc program [15] is available at the Queen Mary University of London. MSc Internet of Things (Data) is currently available for one-year full-time study, two years' part-time study (Introduction to IOT, Enabling Communication Technologies for IOT).

The article [16] presents the best universities that offer courses in the field of "Internet of Things", and study in detail what they offer their students. The project "IoT Academy Samsung" [17] is organized on the base of Moscow Institute of Physics and Technology.

Things (IoT) [18] certifications and training are job-role-based programs designed to help meet the growing need for specialized talent.

National Aerospace University "KhAI" and other Ukrainian universities prepare specialists on programmable mobile systems and IoT [19]. Lviv IT Cluster and National University "Lvivska polytechnica" have launched a Bachelor program "Internet of Things" [20].

Conclusions and questions

In this chapter we proposed and discussed the structure of the curriculum "Technology and development tools WoT applications" designed for training the development of modern web applications based on JS/Node.JS, which will give the listeners a starting point for mastering the WoT concept and developing Web Things applications.

Because in the future the orbit IoT/WoT will include the technologies of deep machine learning, artificial intelligence, technology blockchain and robotics, the curricula for training of future specialists in the field of IoT/WoT will be updated and filled with new content. In the future it is planned to deploy a specialty “Architecture and technology IoT/WoT” and propose the disciplines that will be taught within the framework of this specialty.

In order to better understand and assimilate the educational material that is presented in this section, we invite you to answer the following questions.

1. What technology is used to create interactive web applications using JavaScript?
2. What is the essence of jQuery technology?
3. What is the essence of a SQL database management system?
4. What is the essence of NoSQL (Not Only SQL) DBMS?
5. MongoDB DBMS Interaction Technologies with Web Applications.
6. What is WebSocket?
7. What is the essence of Ajax technology and data transfer formats?
8. What is the difference between WebSockets and REST?
9. What technology is used for web server and web application interaction?
10. What is the essence of Node.js technology, which allows you to create event-driven servers?
11. What are the main cloud computing models?
12. Web services and cloud computing.
13. What does the IoT architecture consist of?
14. What are the main technologies used in IoT/IIoT?
15. What are the main programming languages used to create applications that implement the API at the first level of IoT/IIoT?
16. What technologies are used to create client IoT applications for smart devices?
17. List the components of the IoT system architecture.
18. What are the main cloud computing used for IoT platforms?

19. Technologies of work with Big Data (MapReduce, Hadoop).
20. IoT/IIoT и интеллектуальный анализ данных (Data Mining).
21. 6LoWPAN protocol stack for IoT/IIoT.
22. Application Layer Protocols for IoT/IIoT.
23. With what networks does the interaction of sensors and actuators at the first level of IoT / IIoT?
24. RFID Technology Concepts.
25. WSN technologies and short-range and long-range networks protocols.
26. M2M technologies and protocols (DDS, LwM2M standards).
27. What are the main standards applied in IoT/IIoT?
28. Basics of WoT. Semantic web and microformats.
29. Stack of architecture Web of Things with different layers.
30. Patterns of Smart Objects Integration to the Internet.
31. Web Thing API implementation technologies for WoT integration patterns.
32. Security, privacy, access control to physical devices WoT
33. Integrating Web of Things and Semantic Web Technologies.
34. Semantic Web Technologies Standards (Semantic Sensor Network XG Final Report, RDF Schema).
35. What standards are used to develop WoT applications?

References

1. Dominique D. Guinard and Vlad M. Trifa, ‘Building the Web of Things’, Manning Publications.: United States, June 2016, 344 p.
2. Jaime A. Martins, Andriy Mazayev, Noélia Correia, ‘Hypermedia APIs for the Web of Things’, IEEE Journals & Magazines, Vol. 5, p.p. 20058 – 20067, September 2017.
3. Mihai Vlad Trifa, ‘Building Blocks for a Participatory Web of Things: Devices, Infrastructures, and Programming Frameworks’, A dissertation submitted to the ETH Zurich for the degree of Doctor of Science, 2011, 190 p.
4. Kazuo Kajimoto, Matthias Kovatsch, Uday Davuluru, ‘Web of Things (WoT) Architecture’. W3C First Public Working Draft, 14 September, 2017. Available at: https://w3c.github.io/wot-architecture/. [accessed May 3, 2018].
5. Sebastian Kaebisch, Takuki Kamiya, ‘Web of Things (WoT) Thing Description’. W3C Editor's Draft 27 April, 2018. Available at: https://w3c.github.io/wot-thing-description/. [accessed May 3, 2018].

6. Zoltan Kis, Kazuaki Nimura, Daniel Peintner, Johannes Hund, 'Web of Things (WoT) Scripting API'. W3C Editor's Draft 04 April, 2018. Available at: https://w3c.github.io/wot-scripting-api/. [accessed May 3, 2018].

7. Michael Koster, 'Web of Things (WoT) Protocol Binding', W3C Editor's Draft 04 April, 2018. Available at: https://w3c.github.io/wot-binding-templates/. [accessed May 3, 2018].

8. Vlad Trifa, Dominique Guinard, David Carrera, 'Web Thing Model', W3C Member Submission 24 August, 2015. Available at: https://www.w3.org/Submission/wot-model/. [accessed May 3, 2018].

9. Ben Francis, 'Web Thing API', Unofficial Draft 02 May, 2018. Available at: https://iot.mozilla.org/wot/. [accessed May 3, 2018].

10. The name of project is ALIOT, which is acronym from official name 'Internet of Things: Emerging Curriculum for Industry and Human Applications'. (n.d.). Available at: http://aliot.eu.org/. [accessed May 3, 2018].

11. Telit IoT University, 'Things-to-Apps Made Easy'. (n.d.). Available at: https://www.telit.com/iot-university/. [accessed May 3, 2018].

12. IoT University, 'IoT UI Development with ThingWorx: Course Summary, Course Milestones'. (n.d.). Available at: https://www.iotu.com/enrollment/student/iot-ui-development-with-thingworx. [accessed May 3, 2018].

13. IoT Academic Program, 'The PTC IoT Academic Program is a "passport" to the future for students, makers, and researchers'. (n.d.). Available at: https://www.ptc.com/-/media/Files/PDFs/Academic/iot_academic-program_EN.pdf?la=en&hash=6AAA287FED90220A9B4104D100CBA9BF94B1CF3B. [accessed May 3, 2018].

14. G. Kortuem, A. K. Bandara, N. Smith, M. Richards, M. Petre, 'Educating the internet-of-things generation', Computer, vol. 46, no. 2, pp. 53-61, Feb., 2013.

15. Queen Mary University of London, 'MSc Internet of Things - Queen Mary University of London'. (n.d.). Available at: https://www.qmul.ac.uk/postgraduate/taught/coursefinder/courses/173148.html. [accessed May 3, 2018].

16. IoT India Magazine, '10 Leading University Courses on IoT (Worldwide)', August, 2016. Available at: https://curiousdose.com/2016/08/10-leading-university-courses-iot-worldwide/. [accessed May 3, 2018].

17. MIPT, 'The IOT Academy of Samsung opens in MFTI'. (n.d.). Available at: https://mipt.ru/news/v_mfti_otkryvaetsya_iot_akademiya_samsung. [accessed May 3, 2018].

18. Cisco, ‘Internet of Things Specialists’. (n.d.). Available at: https://www.cisco.com/c/en/us/training-events/training-certifications/certifications/specialist/internet-of-things.html. [accessed May 3, 2018].

19. Department of Computer Systems, Networks and Cybersecurity, ‘Programmable mobile systems and Internet of Things’. (n.d.). Available at: https://csn.khai.edu/speciality/programovani-mobilni-sistemi-i-internet-rechej . [accessed May 3, 2018].

20. Internet of Things, ‘Bachelor program in Internet of Things’ (n.d.). Available at: http://iot.lviv.ua/. [accessed May 3, 2018].

21. Namiot D.E. ‘About the Internet of Things training programs’, International Journal of Open Information Technologies ISSN: 2307-8162 vol. 3, no. 5, pp. 35-38, 2015.

22. Amelie Gyrard, Pankesh Patel, Soumya Kanti Datta and Muhammad Intizar Ali, ‘Semantic Web Meets Internet of Things and Web of Things: [2nd Edition]’, WWW 2017 - 26th International World Wide Web Conference, Perth, Australia, pp. 917-920, Apr 2017.

23. Semmy Purewal, ‘Learning Web App Development’, O'Reilly Media, February 2014, 306 p.

24. Ethan Brown, ‘Web Development with Node and Express’, O'Reilly Media, July 2014, 336 p.

25. V. Tkachenko, ‘IoT - modern telecommunication technologies’, August, 2016. Available at: http://www.lessons-tva.info/articles/net/013.html. [accessed May 3, 2018].

26. Jiafu Wan, Min Chen, Feng Xia, Di Li, and Keliang Zhou, “From Machine-to-Machine Communications towards Cyber-Physical Systems”, Computer Science and Information Systems, Vol. 10, No. 3, pp. 1105-1128, 2013

27. M. Selinger, A. Sepulveda, and J. Buchan, ‘Education and the internet of everything: How ubiquitous connectedness can help transform pedagogy’, Cisco Consulting Service and Cisco EMEAR Education Team, Oct., 2013. Available at: https://www.cisco.com/c/dam/en_us/solutions/industries/docs/education/education_internet.pdf. [accessed May 3, 2018].

28. Dominique Guinard, ‘What is the Web of Things’? April 8, 2017. Available at: https://webofthings.org/2017/04/08/what-is-the-web-of-things/. [accessed May 3, 2018].

29. Fleisch, Elgar, ‘What is the Internet of Things’? An Economic Perspective. Economics, Management, and Financial Markets, 5 (2), pp. 125-157, ISSN 1842-3191, January, 2010.

30. Luigi Atzori, Antonio Iera, Giacomo Morabito, ‘The Internet of Things: A survey’, Computer Networks, Vol. 54, Issue 15, pp. 2787-2805, 28 October, 2010.

31. R. Parashar, A. Khan, and Neha, ‘A survey: The internet of things’, International Journal of Technical Research and Applications, vol. 4, Issue 3, pp. 251-257, May-June, 2016.

32. Pallavi Sethi and Smruti R. Sarangi, ‘Internet of Things: Architectures, Protocols, and Applications’, Journal of Electrical and Computer Engineering, Vol. 2017, Article ID 9324035, January, 2017, 25 p.

33. Jie Lin, Wei Yu, Nan Zhang, Xinyu Yang, Hanlin Zhang, Wei Zhao, ‘A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications’, IEEE Internet of Things Journal, Vol, 4, Issue 5, pp**,** 1125 – 1142, October, 2017.

34. DDS the proven data connectivity standard for the IoT, ‘What is DDS’? (n.d.). Available at: http://portals.omg.org/dds/what-is-dds-3. [accessed May 3, 2018].

35. Ala Al-Fuqaha, Mohsen Guizani, Mehdi Mohammadi, Mohammed Aledhari, Moussa Ayyash, ‘Internet of Things: A Survey on Enabling Technologies, Protocols and Applications’, IEEE Communications Surveys & Tutorials, Vol. 17(4): Fourthquarter 2015, pp. 2347 – 2376, November, 2015.

36. Sathish Kumar and Dhiren Patel, ‘A Survey on Internet of Things: Security and Privacy Issues’, International Journal of Computer Applications 90(11), pp. 20-26, March, 2014.

37. O. Garcia-Morchon, S. Kumar, M. Sethi, ‘State-of-the-Art and Challenges for the Internet of Things Security draft-irtf-t2trg-iot-seccons-14’, April, 21, 2018. Available at: https://datatracker.ietf.org/doc/draft-irtf-t2trg-iot-seccons/. [accessed May 3, 2018].

38. OWASP Internet of Things Project, ‘IoT Security Guidance’. (n.d.). Available at: https://www.owasp.org/index.php/IoT_Security_Guidance. [accessed May, 3, 2018].

39. Oxford e-Research Centre, ‘The Internet of Things: realising the potential of a trusted smart world’. (n.d.). Available at: http://www.oerc.ox.ac.uk/news/Centre-contribution-IoT-reports. [accessed May 3, 2018].

40. A. Kamilaris, S. Yumusak, and M. I. Ali, ‘WOTS2E: A search engine for a Semantic Web of Things’, 2016 IEEE 3rd World Forum on Internet of Things (WF-IoT), Reston, VA, USA, pp. 436-441, 2016.

41. Dominique Guinard, Vlad Trifa, and Erik Wilde, ‘A Resource Oriented Architecture for the Web of Things’. In Proc. of the 2nd International

Conference on the Internet of Things (IoT 2010), LNCS, Tokyo, Japan, Springer Berlin / Heidelberg, November, 2010.

42. Dominique Guinard, Vlad Trifa, Erik Wilde, ‘Architecting a Mashable Open World Wide Web of Things’, Technical Report No. 663, Department of Computer Science, ETH Zürich, February, 2010.

43. Yuchao Zhou, Suparna De, Wei Wang, Klaus Moessner, ‘Search Techniques for the Web of Things: A Taxonomy and Survey’, Sensors 16(5), 600; doi:10.3390/s16050600, Apr., 2016.

44. Andreas Kamilaris, Andreas Pitsillides, Francesc X. Prenafeta-Bold, Muhammad Intizar Ali, ‘A Web of Things based eco-system for urban computing - towards smarter cities’, Telecommunications (ICT) 2017 24th International Conference on, pp. 1-7, 2017.

45. V. Tkachenko, ‘Web of Things - IoT Network Service’, October 2017. Available at: http://www.lessons-tva.info/articles/net/014.html. [accessed May 3, 2018].

46. Ben Francis, “Building the Web of Things”, June, 2017. Available at: https://hacks.mozilla.org/2017/06/building-the-web-of-things. [accessed May 3, 2018].

47. EVRYTHNG Developer Hub, ‘Welcome’. (n.d.). Available at: https://developers.evrythng.com/docs. [accessed May 3, 2018].

48. S. Friedl, A. Popov, A. Langley, E. Stephan, 'Transport Layer Security (TLS) Application-Layer Protocol Negotiation Extension’, July, 2014. Available at: https://tools.ietf.org/html/rfc7301. [accessed May 3, 2018].

49. M. Peck, K. Igoe, ‘Suite B Profile for Datagram Transport Layer Security / Secure Real-time Transport Protocol (DTLS-SRTP) draft-peck-suiteb-dtls-srtp-02’, March, 2013. Available at: https://tools.ietf.org/html/draft-peck-suiteb-dtls-srtp-02. [accessed May 3, 2018].

50. Eugene Ferry, John O’Raw, Kevin Curran, ‘Security Evaluation of the OAuth 2.0 Framework’, Information Management and Computer Security, Vol. 23, Iss. 1, pp. 73, October, 2014.

51. OAuth Community Site, ‘OAuth 2.0’. (n.d.). Available at: https://oauth.net/2/. [accessed May 3, 2018].

52. T. Borgohain, A Borgohain, U Kumar, S Sanyal, ‘Authentication systems in Internet of Things’, arXiv preprint arXiv:1502.00870, 2015.

53. Elena Reshetova, Michael McCool, ‘Web of Things (WoT) Security and Privacy Considerations’, 14 December, 2017. Available at: https://www.w3.org/TR/wot-security/. [accessed May 3, 2018].

54. WebStorm, ‘The smartest JavaScript IDE. Powerful IDE for modern JavaScript development’. (n.d.). Available at: http://www.jetbrains.com/webstorm/. [accessed May 3, 2018].

55. Sublime text, ‘A sophisticated text editor for code, markup and prose’. (n.d.). Available at: https://www.sublimetext.com/. [accessed May 3, 2018].

56. Git, ‘Git-local-branching-on-the-cheap’. (n.d.). Available at: https://git-scm.com/. [accessed May 3, 2018].

57. Resources for developers, ‘HTML5 - Web developer guides’. (n.d.). Available at: https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/HTML5. [accessed May 3, 2018].

58. World Wide Web Consortium (W3C), ‘Extensible Markup Language (XML)’, October 2016. Available at: https://www.w3.org/XML/. [accessed May 3, 2018].

59. Resources for developers, ‘CSS3 - CSS: Cascading Style Sheets’. (n.d.). Available at: https://developer.mozilla.org/en-US/docs/Web/CSS/CSS3. [accessed May 3, 2018].

60. Bootstrap, ‘The most popular HTML, CSS, and JS library in the world’. (n.d.). Available at: https://getbootstrap.com/. [accessed May 3, 2018].

61. Resources for developers, ‘JavaScript’. (n.d.). Available at: https://developer.mozilla.org/en-US/docs/Web/JavaScript. [accessed May 3, 2018].

62. JavaScript, ‘Ready to try JavaScript’? (n.d.). Available at: https://www.javascript.com/. [accessed May 3, 2018].

63. jQuery, ‘What is jQuery’? (n.d.). Available at: https://jquery.com/. [accessed May 3, 2018].

64. Resources for developers, ‘Ajax’. (n.d.). Available at: https://developer.mozilla.org/en-US/docs/Web/Guide/AJAX. [accessed May 3, 2018].

65. Websocket.org - Powered by Kaazing, ‘About HTML5 WebSocket’. (n.d.). Available at: http://websocket.org/aboutwebsocket.html. [accessed May 3, 2018].

66. Socket.IO, ‘Socket.io 2.0 is here. Featuring the fastest and most reliable real-time engine’. (n.d.). Available at: https://socket.io/. [accessed May 3, 2018].

67. World Wide Web Consortium (W3C), ‘The WebSocket API’. (n.d.). Available at: https://www.w3.org/TR/websockets/. [accessed May 3, 2018].

68. Node.js®, ‘Node.js v8.11.1 Documentation’. (n.d.). Available at: https://nodejs.org/en/. [accessed May 3, 2018].

69. MongoDB for GIANT Ideas, ‘MongoDB Atlas. Database as a Service’. (n.d.). Available at: https://www.mongodb.com/. [accessed May 3, 2018].

70. Alan B. Johnston, Daniel C. Burnett, ‘WebRTC: APIs and RTCWEB Protocols of the HTML5 Real - Time Web’, St. Louis, USA: Digital Codex LLC, Smashwords Edition, 2013, 247 p.

71. Microsoft Azure, ‘What is cloud computing’? (n.d.). Available at: https://azure.microsoft.com/en-in/overview/what-is-cloud-computing/. [accessed May 3, 2018].

72. Service Architecture, ‘Service-Oriented Architecture (SOA) Definition’. (n.d.). Available at: https://www.service-architecture.com/articles/web-services/service-oriented_architecture_soa_definition.html. [accessed May 3, 2018].

73. EVERYTHNG Developer Hub, ‘Standard API Introduction’. (n.d.). Available at: https://developers.evrythng.com/v3.0/reference. [accessed May 3, 2018]