

# IoT を題材とした PBL の実施と分析

細合 晋太郎 石田 繁巳 亀井 靖高 大迫 周平

井垣 宏 鷗林 尚靖 福田 晃

本学では、本年度より取り入れた IoT システムを題材とした Project Based Learning(PBL) の導入を行った。IoT を取り入れた PBL では、Web アプリケーション、組込みシステムと幅広い技術を扱うため、学生が自身のアイデアを形にすることに注力出来るよう教材や開発環境の準備が重要となる。本稿では、PBL の導入事例の紹介と、アンケートによる授業の評価を行う。

In this year, we brought in Project Based Learning(PBL) about IoT system. PBL about IoT System needs a wide range of technology Web applications, and embedded systems. For this reason, Preparation of teaching materials and the development environment is important to the students to create their own ideas. In this paper, we explain our PBL design about IoT system and analysis based on the questionnaire results of the PBL.

## 1 はじめに

近年、情報技術を活用して社会の様々な課題を解決できる人材の育成が重要な課題となっている。我々はこのような課題に対して、文部科学省 情報技術人材育成のための実践教育ネットワーク形成事業の支援のもとで、九州大学を含む全国 15 大学（申請代表校：大阪大学）は、教育プログラム「分野・地域を超えた実践的情報教育協働ネットワーク（Education Network for Practical Information Technologies (enPiT)）」<sup>†1</sup> を平成 24 年に開始した。

enPiT は、クラウドコンピューティング、セキュリティ、組込みシステム、ビジネス・アプリケーション

の 4 分野を対象に、複数の大学が連携して実践的な課題解決力を持った人材育成を行う教育プログラムである。特に九州大学では、名古屋大学と共に、組込みシステム分野 (enPiT-Emb) における実践的な教育を実施・普及を行っている。

組込みシステムでは、センサ、アクチュエータ、プロセッサといった様々なデバイス、ソフトウェアを組み合わせた複合的なシステムの開発が求められる。特に昨今では、Internet of Things(IoT) [2] と呼ばれる新しい概念が提唱されており、モノ同士をインターネット経由で連携した、新しいサービス、システムの開発が行われるようになりつつある。

九州大学では、このような組込みシステム開発の現状にもとづき、大学院システム情報科学府 情報知能工学専攻 社会情報システムコース (QITO) の学生を対象に enPiT-Emb のカリキュラムを進めている。本コースでは修士一年前期に PBL 第一、修士一年後期に PBL 第二、修士二年前期に PBL 第三と計一年半に渡る PBL を実施している。これらの PBL は組込みシステム、特に IoT を題材としており、PBL 第一は以降の PBL を進めるにあたって必要な基礎技術や知識を修得するための基礎学習型 PBL、第二、第

### A Case Study of IoT-based PBL

Shintaro Hosoi, Shigemi Ishida, Yasutaka Kamei, Shuhei Ohsako, Naoyasu Ubayashi, Akira Fukuda, 九州大学大学院システム情報科学研究所システム情報科学府 高度 ICT 人材教育開発センター, QUTE:Kyushu University Research Center for Advanced Information and Communication Technology Education, Kyushu University.

Hiroshi Igaki, 大阪大学, Osaka University.

<sup>†1</sup> 分野・地域を超えた実践的情報教育協働ネットワーク, <http://www.enpit.jp/>

三は学生主体の実践型 PBL として実施される。

本稿では、2014 年度に実施した PBL 第一を対象として、IoT を題材とした PBL のカリキュラム設計及び実施結果について報告する。以降では、2 章で PBL 第一を含む我々のカリキュラムについて、3 章で IoT 教育に必要な教材や環境について述べる。4 章で PBL 第一で提案された学生テーマを紹介し、5 章で現在の PBL 第一の状況と授業評価アンケートの結果について詳述し、6 章でまとめとする。

## 2 PBL 第一カリキュラム

PBL 第一では、チームで組込みシステムを開発するために必要な様々な技術・知識の習得を目的とし、組込みシステム実現に求められる実装技術や開発環境だけでなく、ロジカルシンキングによるアイデアの創発、システム設計技術や開発プロセスについての講義・演習を実施した。実施にあたっては受講生 15 名を 3~4 名の 4 チームに分けた。

PBL 第一は水曜日に 2 コマ、金曜日に 3 コマの週 5 コマで実施している。PBL 第一のカリキュラムを表 1 に示す。

我々のカリキュラムでは、ソフトウェア開発プロセスとして Scrum [3] と呼ばれるアジャイルソフトウェア開発フレームワークを教育している。Scrum では、計画・開発・レビュー・振り返りという一連の流れをスプリントと呼ばれる一定期間ごとの繰り返し中に実施する。そのため PBL 第一においても、約 2 週間を 1 スプリントにとし、その期間中にシステムの案出し、開発計画、IoT システムの開発、レビュー・振り返りというスケジュールを設定した。さらに各スプリントでは、受講生による開発以外に、チームでのソフトウェア開発に必要な種々の技術や知識に関する講義・演習を行うことで、プロダクト/プロセス両面において必要な技術・知識を段階的に習得できるようカリキュラムを設計した。以降では、各スプリントの実施内容について説明する。

### 2.1 第 1 チュートリアルスプリント

第一スプリントでは、Scrum についての理解度が十分ではないため、Scrum の流れを一つずつ追いな

表 1 PBL 第一スケジュール

工程	日付	主要な内容
ガイダンス	4 月 11 日 (金)	開発環境及び各種ツールの使い方、開発プロセス (Scrum)
	4 月 16 日 (水)	
スプリント 1	4 月 18 日 (金)	ロジカルシンキング 1 回目、スプリントチュートリアル振り返り+計画
	4 月 23 日 (水)	設計 (ユースケース図, ロバストネス図, シーケンス図等) の講義
	4 月 25 日 (金)	ロジカルシンキング二回目
	4 月 30 日 (水)	Scrum 演習 (プロダクトバックログ, スプリント計画, スプリントバックログ)
	5 月 2 日 (金) 5 月 7 日 (水)	
スプリント 2	5 月 9 日 (金)	詳細設計についての講義, 組込みシステム開発
	5 月 14 日 (水)	
	5 月 16 日 (金)	
	5 月 21 日 (水)	スプリントレビュー, 振り返り, 次スプリントのための計画
スプリント 3	5 月 23 日 (金)	Web アプリケーション開発, スプリントレビュー, 振り返り, 次スプリントのための計画
	5 月 28 日 (水)	
	5 月 30 日 (金)	
	6 月 4 日 (水)	
スプリント 4	6 月 6 日 (金)	IoT システム開発
	6 月 11 日 (水)	
	6 月 13 日 (金)	PBL 発表会練習 1
	6 月 18 日 (水)	
	6 月 20 日 (金)	PBL 発表会練習 2
6 月 25 日 (水)		
PBL 発表会	6 月 27 日 (金)	外部向け PBL 発表会
スプリント 5	7 月 2 日 (水)	全体システム開発・修正
	7 月 4 日 (金)	
	7 月 9 日 (水)	
	7 月 11 日 (金)	デモ検討
	7 月 16 日 (水)	スプリントレビュー
最終発表会	7 月 18 日 (金)	最終成果物ツールデモ (内部向け)

がらチームごとに作成するシステムのアイデア出しを行った。アイデア出しを行う際には、本 PBL で受講生が利用可能なセンサ群、アクチュエータ群とサーバの情報を提示した（利用したセンサ・アクチュエータについては後述する）。さらにこのスプリントでは、ICONIX プロセス [4] と呼ばれる UML を用いたシステム設計手法にもとづいて、受講生のアイデアを設計に落としこむ方法について講義・演習を行った。また、アイデア出しを有効に行うための手段として、ロジカルシンキングの授業においてアイデアの創発や具体化といった手法についての講義を行った。

なお、このスプリントでは組込みシステムを実際に開発するところまでではできなかったため、アイデア及び概要設計についてのみレビューを行った。

## 2.2 第2～第5スプリント

第2スプリントでは、第1スプリントの成果から詳細設計を作成し、一部の機能を実際に組込みシステム開発するところまでを実施した。受講生らが提案した内容はセンサやアクチュエータだけでなく、インターネット上の Web サービスを組み合わせた非常に多様な連携を必要とするものであった。しかしながら、一度にすべての内容をカバーすることは困難であったため、ここではその提案の中のセンサやアクチュエータを利用する部分のみを対象とし、教員による技術指導にもとづいて、受講生チームが各自の組込みシステムのプロトタイプ開発を行った。

第3スプリントでは、Web アプリケーション開発手法に関する講義・演習を行った。IoT システムでは、センサ値を Web アプリケーションから呼び出したり、アクチュエータを既存の Web サービスと連携して操作することが多い。そのためこのスプリントでは、受講生らの提案内容のうち Web アプリケーション部分を対象として開発を行うよう教員が誘導した。

第4スプリントでは、第1～第3スプリントでの講義・演習内容及び受講生らの開発成果を組み合わせ、どのようにして IoT システムとして完成させればよいかの講義・演習を行った。受講生らはこのスプリントでの講義にもとづいて、Web アプリケーションと組込みシステムを連携させる基本的な IoT シス

テムの開発を行った。さらにこのスプリントでは中間発表として、九州大学内外のゲストを対象として開発中の IoT システムをデモベースで紹介する PBL 発表会を実施した。

第5スプリントは最終成果物を完成させるスプリントとして、7/18 の最終ツールデモ大会に向けた開発・レビューを実施した。

## 2.3 レビューと発表

PBL 第一では、コミュニケーション力と表現力の向上を目的に、発表の場を多く設けている。各スプリントのレビューでは成果物のデモと提案テーマに関するプレゼンテーションを行う。この際、教員に加え修士二年の先輩にも参加して貰った。また、6/27 の PBL 発表会には外部の企業 16 名、外部の大学等から 6 名、九州大学教員・学生 42 名が参加する中で開催された。

## 3 教材設計

PBL 第一では IoT システムの提案と実装を題材としている。本 PBL では、IoT システムの基盤を教員から受講生に教材として提示した。図 1 に教材の全体構成を示す。

教材は大きく組込みシステムと Web アプリケーションに分けられる。さらに組込みシステムは組込みボードとセンサ、アクチュエータから構成されており、組込みボードにセンサとアクチュエータを接続して利用する。Web アプリケーションはクラウドプラットフォーム上のサーバに構築されており、組込みボードと MQTT Broker を通じて通信を行う。図中の緑色の背景の要素が学生が主に開発する項目である。その他の部分は予め利用できるように準備を行った。以降、使用機材や要素技術について述べる。

### 3.1 要素技術と使用機材

#### Galileo

組込みボードとして、Intel 社の Galileo<sup>†2</sup> ボードを用いた。Galileo ボードは、Intel Quark SoC X1000

<sup>†2</sup> <http://www.intel.com/content/www/us/en/do-it-yourself/galileo-maker-quark-board.html>

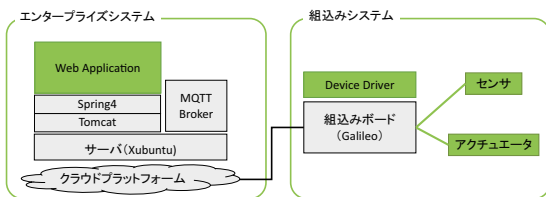


図 1 IoT 向け PBL 教材

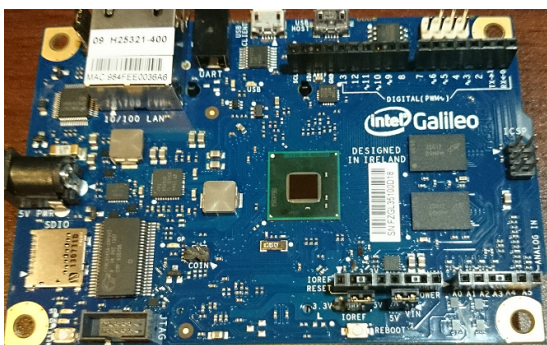


図 2 Intel Galileo

(400MHz) を搭載した高機能な組み込みボードで OS として軽量 Linux を搭載している。また、Linux 上には Arduino 環境が搭載されており、センサやアクチュエータを利用する際は Arduino 上にデバイスドライバとアプリケーションを実装する。

図 2 に Galileo ボードを示す。

Galileo には Ethernet ポートだけでなく、Wifi モジュールの搭載も可能となっており、容易にネットワークに接続することができる。

### Spring4

Spring Framework4(以後 Spring4)<sup>†4</sup> は Java 言語を用いたアプリケーションフレームワークの 1 つである。特に Spring Boot と呼ばれる機構を用いることで、最低限の記述のみで Web アプリケーションを簡単に作成することができる。チーム開発では、各自の開発環境に Spring Framework を導入し、各自の開発環境で開発・テストを行う。個々に作成した成果物は最終的にはマージされ、サーバで動作している Tomcat 上にデプロイされる。

†4 <http://spring.io/>

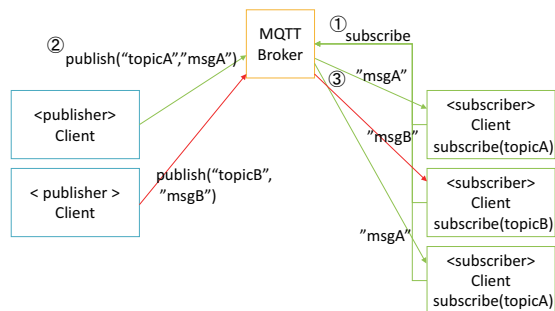


図 3 MQTT

### MQTT

Message Queuing Telemetry Transport(MQTT)<sup>†5</sup> は IBM による IoT 向けの軽量プロトコルである。TCP/IP 上で動作し、Publish / Subscribe モデルにもとづいており、HTTP に比べ軽量である。

MQTT の通信方式について図 3 に示す。MQTT のネットワークは 1 つの MQTT Broker と複数の MQTT Client からなる。Client は Publisher または Subscriber (または両方) となり、特定のトピックに向け Publisher から送信 (Publish) されたメッセージが、そのトピックを購読 (Subscribe) している Subscriber に配信される。Publish/Subscribe 型であるためテストが個別に実施でき、スケールアップも容易である。

MQTT 利用においては MQTT Broker のアドレスとトピックを指定し、メッセージを送信・受信するのみであるため、HTTP の実装と比べ比較的容易にクライアントプログラムを作成することができる。

本 PBL では、MQTT Broker に MQTT の OSS 実装である Mosquitto<sup>†5</sup> を用いチームサーバ上に配備した。また、Web アプリと Galileo 上では OSS の MQTT ライブラリ Phao<sup>†6</sup> を用いてクライアントプログラムの実装を行えるようにした。

### センサとアクチュエータ

実世界の様々な情報を取得したり、反応を返すために多くのセンサ・アクチュエータを準備した。図 2

†5 <http://mqtt.org/>

†5 <http://mosquitto.org/>

†6 <http://www.eclipse.org/paho/>

表 2 センサとアクチュエータ

名称	型番
静電容量キーパッド	MRP121
静電容量式タッチセンサ	AT42QT1010
気圧センサ (気温センサ付き)	BMP180
フォトフレクタ・モジュール	QTR-1RC
3D コンパス (加速度・地磁気)	LSM303D
超音波距離センサ	HC-SR04
温度センサ	HTU21D
三軸ジャイロモジュール	ITG-3200
温度センサ (I2C)	TMP006
GROVE - 光センサ	
GROVE - 電流センサ	
GROVE - 水分センサ	
感圧センサ (小型)	
感圧センサ (円形)	
SoftPot 接触位置センサ	
心拍センサ	A.P. Shield 05
ホイールポテンショメータ 10k Ω	
トリマポテンショメータ 10k Ω	
デジタルポテンショメータ 10k Ω	MCP4131
赤外線リモコン受信チップ	SIS-2
2色 LED マトリックス (I2C)	
サーボモータ (大)	
サーボモータ (小)	
LilyPad バイブレータ	
リモコン送受信キット	IRKit
Max Power 赤外線 LED キット	

に、センサ・アクチュエータの一覧を示す。

センサ・アクチュエータは第1チュートリアルスプリントの段階で自由に選べるようにし、アイデア創発の手助けとなるようにしている。

センサ・アクチュエータのドライバの実装は、データシートの読み方等を指導した上で、基本的に学生が主体となって行っている。

### 3.2 アーキテクチャ

図4に前項の要素技術を用いた今回のシステムアーキテクチャを示す。

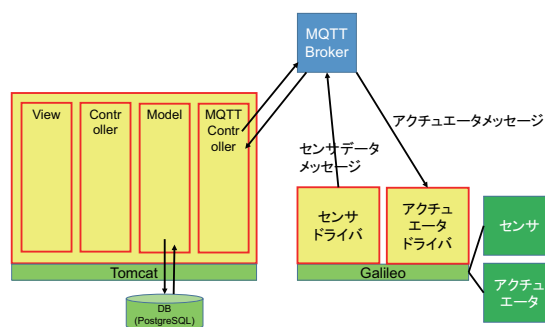


図 4 教材 IoT システムのアーキテクチャ

センサ・アクチュエータを接続した組み込みボード Galileo は MQTT を用いてセンサ値を送信する。同時に MQTT で送信されてきたメッセージにもとづいてアクチュエータを操作する。

クラウド上に構築したサーバ上には、開発環境と、Tomcat, MQTT Broker が配備されている。MQTT のメッセージは MQTT Broker に送られ、そこから Tomcat 上のアプリケーションに配信される。Tomcat 上の Web アプリケーションは MVC アーキテクチャに加え組み込み機器との連携を行うコントローラを追加した形とした (MQTT Controller)。このパッケージではセンサ・アクチュエータに応じた IoT コントローラを作成し、MQTT メッセージの送受を行う。

Model パッケージでは、センサ・アクチュエータに応じたモデルと、DB に保存するモデルを配置する。View/Controller パッケージでは、Web アプリケーションに対応するコントローラとビューを配置する。View は Thymleaf を用いた HTML テンプレートにより実現する。学生の実装する部分については簡易なサンプルプログラムを提供しており、これを参考に各チームのサービスに応じたクラスを作成するよう指導した。MQTT の通信部や DB 操作、ログイン機能などは予め作成したものを提供した。

### 3.3 開発環境

開発環境はクラウド上にサーバを作成し、DaaS 環境として提供した。図5に PBL で用いた DaaS 開発環境を示す。

本コースの学生はそれぞれ個別にノート PC が配

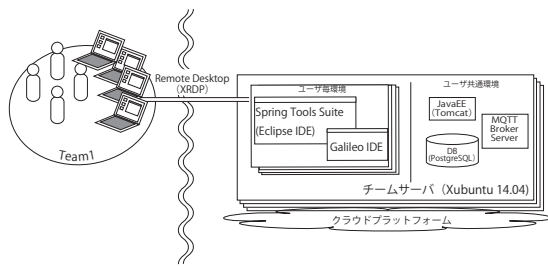


図5 開発環境

布されており、環境が様々であるため、同一の開発環境を構築することが難しい。学生にとって導入が容易であることやサポートの容易性、活動の監視のしやすさから、DaaSを選択した。しかしながらネットワーク環境に依存してしまうため、本校のネットワークポリシーにより、学外からの利用が難しいことや、同時アクセスによる負荷の問題など、運用上の問題も多く、今後は仮想マシンを受講生の環境で動作させる手法での提供も検討している。

#### 4 学生提案テーマ

現在 QITO コースの学生 15 名、4 チームが本 PBL に取り組んでいる。各チームが取り組んでいるテーマを以下に示す。

**自動購買システム (ICLA)** オフィスや研究室などでは、有志により置き菓子や飲み物といった小規模な無人販売を行っている所も多い。このようなシステムでは金銭の管理や購買履歴の把握といったことが難しい。本システムでは、コインセンサと重量センサを用いて、簡易な自動販売機システムを構築する。コインセンサによる投入金額の把握と、重量センサによりケース内の金額の把握を行う、また、購入履歴は Web サービスを介して保存・閲覧ができる。

**メール自動返信システム** 携帯電話など携帯することを忘れた場合、送信者には、受信者に届いていないことが確認できない。本システムでは、受信者が携帯電話を携帯していなかった場合に、受信したメールに自動返信を行う。室内のドアノブに取り付けた接触センサにより、ユーザの位置を把握するとともに、WiFi で携帯電話の位置を

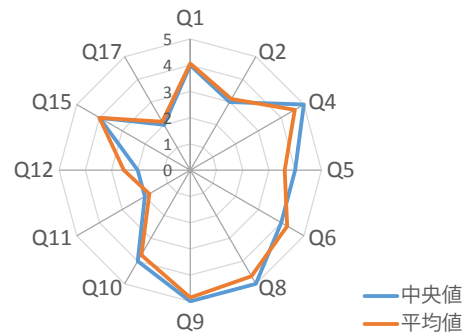


図6 アンケート結果 (レーダーチャート)

検知し、携帯していないと判断した際に自動返信を行う。

#### 携帯忘れ通知感知システム (I'LL BE BACK)

近年、携帯電話には多くの個人情報と搭載されており、紛失した場合の損失は非常に大きい。本システムは、携帯電話と WiFi で接続し、携帯電話の位置と使用者の位置関係を把握する。携帯電話と一定以上離れるとバイブレーションで通知することで、携帯電話の置き忘れを防止する。

#### 天気情報付き植物栽培補助システム (PWIS)

植物の育成には多くの時間とノウハウが必要となる。本システムでは、プランターに取り付けた水分、温度、湿度、気圧、照度といったセンサ情報により、植物の現在状況を取得し、保存する。オンラインからは天気予報情報を取得し、植物の現在状況に応じて適切な水分管理をユーザに促すシステムである。育成初心者へのサポートとともに、熟練者のノウハウのデータ化と蓄積にも活用できる。

#### 5 授業評価

7/18 の最終成果物ツールでも終了後に、授業評価アンケートを実施した。アンケートの設問のうち選択式の項目を表 3 に示す。

アンケート結果の平均値及び中央値をレーダーチャートで示したものを図 6 に示す。座学及び演習の有益さ、IoT というテーマ設定、演習室のファシリティ、教員のサポート体制や総合評価といった観点については非常に高い評価が得られた。また、チャート

表 3 アンケート結果（選択設問）

質問項目		1	2	3	4	5	
Q1. 座学の有益さ	有益ではない	0	0	3	8	4	有益だった
Q2. 座学の難易度	易しい	0	3	7	5	0	難しい
Q4. 演習の有益さ	有益ではない	0	0	0	6	9	有益だった
Q5. 演習課題の難易度	易しい	0	0	6	9	0	難しい
Q6. センサなどを用いる IoT テーマには意欲的に取り組めたか？	取り組めなかった	0	0	1	9	5	取り組めた
Q8. 教員の演習指導やサポート	不足	0	0	0	5	10	適切
Q9. 教員に関する総合評価	悪い	0	0	0	2	12	良い
Q10. 開発環境導入の容易さ	易しい	0	2	2	9	2	難しい
Q11. DaaS 環境の操作レスポンス	不快	5	8	2	0	0	快適
Q12. 開発環境（STS, Galileo IDE, Linux 環境等）の使いやすさ	使いづらい	2	7	3	2	1	使いやすい
Q15. 演習室環境	使いづらい	0	2	2	5	6	使いやすい
Q17. ネットワーク環境	不便	6	4	2	3	0	快適

には示していないが座学の時間配分についても適切であったという回答が最も多かった。このことより、IoT ベースの PBL 実施において、そのテーマ設定、教員が提供した座学や演習等の技術指導は受講生にとって有益であったと判断できる。

一方で、演習課題の難易度、開発環境導入の容易さの値は低くなっており、特に DaaS 環境の操作レスポンス、開発環境の使いやすさ、ネットワーク品質については低い値となった。本 PBL においては、開発環境導入は原則として DaaS を用いて実施していたが、Galileo との連携のための開発環境等の一部は受講生によっては各自で自身のノート PC 上に導入していた。また、PBL を実施した演習室の無線 LAN 環境の不具合の影響で、ネットワークの品質が悪くなり、同時にネットワーク越しにアクセスして利用する DaaS 環境の品質（操作レスポンス）も悪化することとなった。Galileo IDE 等の組み込みシステム周りの開発環境については、まだ新しい技術でもあることから、受講生が Linux 等に慣れていないことから、使いやすさの評価が低くなったものと考えられる。今後、ネットワーク品質については大学側と連携し、改善を進めていきたい。

その他に、開発環境や追加に必要な座学・演習があ

るかを自由記述で聞いたところ、ほとんどの受講生が現状で満足している一方で、開発環境の使い方についての座学や演習があると望ましいと回答している受講生がいた。次年度のカリキュラム設計に向けて、検討していきたい。

## 6 おわりに

本稿では、IoT をテーマとした PBL の実施内容について紹介した。IoT システムは、センサ・アクチュエータといった所謂従来の組み込みシステムだけでなく、Web サービス、アプリケーション等多様なモノ同士の連携が必要となる。大学の限られた授業時間内で IoT システムの指導を行うため、チームでの開発プロセス、センサ・アクチュエータ、Web アプリケーション、といった内容を座学・演習等で段階的に指導していった。結果として、チームごとの多様な内容の IoT システムを期間内に完成させ、デモンストレーションを兼ねた発表会を実施することが可能となった。

今後は、不満の多かった開発環境やネットワーク環境を中心にカリキュラム・教材の改善を進めていきたい。

謝辞 九州大学 PEARL プロジェクトは、「分野・地

域を越えた実践的情報教育協働ネットワーク」の補助金により文部科学省 情報技術人材育成のための実践教育ネットワーク形成事業の一環として実施したものである。

#### 参考文献

- [1] 細合晋太郎, 亀井靖高, 大迫周平, 井垣 宏, 鶴林尚靖, 福田 晃, “PBL への DaaS 開発環境の導入事例”, 電子情報通信学会技術研究報告. KBSE, 知能ソフトウェア工学 113(160), 103-108, 2013-07-18, 一般社団法人電子情報通信学会
- [2] Kevin Ashton, “That ‘Internet of Things’ Thing.”, RFID Journal, 22 July 2009. Retrieved 8 April 2011.
- [3] Schwaber, K. and Sutherland, J.: The Scrum Guide, <https://www.scrum.org/Scrum-Guides> (2013)
- [4] Doug Rosenberg, 三河 淳一, 船木 健児, 佐藤 竜一, “ユースケース駆動開発実践ガイド”, 翔泳社, October, 2007.