

IoTシステムを題材としたPBLの導入提案

細合 晋太郎^{1,a)} 石田 繁巳^{1,b)} 亀井 靖高^{1,c)} 大迫 周平^{1,d)} 井垣 宏^{2,e)} 鵜林 尚靖^{1,f)}
福田 晃^{1,g)}

概要: 近年, 多くの教育機関で Project Based Learning(PBL) による教育が取り入れられている. PBL による教育では, プロジェクトを進行するためのプロジェクトマネジメントの能力と, プロジェクトを実現するための技術的な能力を向上させるための教育が求められる. 本稿では, 本年度より取り入れた IoT システムを題材とした PBL の導入事例について紹介する.

キーワード: PBL, IoT, 組込みシステム

PBL for IoT System

HOSOAI SHINTARO^{1,a)} ISHIDA SHIGEMI^{1,b)} KAMEI YASUTAKA^{1,c)} OHSAKO SHUHEI^{1,d)}
IGAKI HIROSHI^{2,e)} UBAYASHI NAOYASU^{1,f)} FUKUDA AKIRA^{1,g)}

Abstract: Project-based Learning (PBL) is widely adopted for teaching team software development in many educational facilities. The PBL requires teachers to make students learn project management skills and product development skills. In this paper, we explain our PBL design about IoT system.

Keywords: PBL, IoT, Embedded System

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. 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))」^{*1} 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, グループワークを用いた短期集中合宿や分散 Project Based Learning(PBL) を実施し, 実践的な課題解決力を持った人材を育成するものである. 九州大学では, 名古屋大学と共に, 組込みシステム分野における実践的な教育を実施・普及する.

九州大学では, 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 のカリキュラムを取り入れている. 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. このうち, PBL 第一は共通のテーマを題材とし

¹ 九州大学大学院システム情報科学研究 院 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, Fukuoka, Japan

² 大阪大学

Osaka University, Osaka, Japan

a) hosoai@qito.kyushu-u.ac.jp

b) ishida@f.ait.kyushu-u.ac.jp

c) kamei@ait.kyushu-u.ac.jp

d) ohsako@qito.kyushu-u.ac.jp

e) igaki@ist.osaka-u.ac.jp

f) ubayashi@ait.kyushu-u.ac.jp

g) fukuda@f.ait.kyushu-u.ac.jp

^{*1} 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/>

て、PBL 第二、第三を進めるにあたり必要な PBL の知識や様々な技術の習得を目的とした基礎学習 PBL である。一方 PBL 第二・第三は学生が主体となりチームごとに異なるテーマを進める実践型 PBL である。

本年度より、PBL 第一の教材を改修し、組込みシステムに関する内容を取り入れた PBL を実施した。前年度までの PBL 第一の取り組みに関しては、[1] を参照頂きたい。

近年、組込みシステムはシステム単体で動作させるよりもインターネットに接続して新たなサービスを提供するものや、機器間の連携を取り入れたものが増えてきている。

Internet of Things(IoT)[2] は、Ashton の提唱した用語で、'もの' (Things) がインターネットに接続され、相互に情報交換や制御を行うものである。従来インターネットは人の相互情報交換が主たる目的であったが、IoT では'もの' が主体となりネットワークを形成する。このような IoT を実現するためには、組込みシステム、エンタープライズ・アプリケーション、ネットワークなど様々な技術要素を習得する必要がある。本コースでは、このような IoT を実現できる人材の育成に向けて、IoT システムの提案と実現を題材とした PBL の導入を行った。

PBL 第一では、PBL を進めるにあたり必要となるプロジェクトマネジメントやアイデア創発などの能力（以後、PBL スキル）と、プロジェクトを実現するために必要となる技術的な能力（技術スキル）の両方の向上を目指す。

本稿では、2 章で PBL スキルを向上させるためのカリキュラムについて、3 章で技術スキルを向上させるための教材や環境について述べる。4 章で現在の PBL 第一の状況について報告し、5 章で総括する。

2. カリキュラム

前章で上げた PBL スキルに対応する講義として、以下のようなカリキュラムを作成した。

- アイデア創発：ロジカルシンキング
- プロジェクトマネジメント：Scrum
- 開発プロセス：ICONIX

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。上記の講義を含む、PBL 第一の全体のスケジュールを表 1 に示す。

ロジカルシンキング

本コースでは、図 1 に示すような II 型の人材の育成を目指している。ロジカルシンキングでは、発想力だけでなくデザイン力も高めることを目的として、デザイン思考を取り入れた講義を実施した。ロジカルシンキングの講義は 2 回に亘り実施し、1 回目の講義ではロジカルシンキングやデザイン思考の基礎的な考え方を様々なワークを通じて習得できるものとした。2 回目の講義では、今回の題材を対象としたアイデア創発を行い、抽象的なアイデアから具体的なプロジェクトとなるように演習を行った。

表 1 PBL 第一スケジュール

Table 1 PBL Schedule

工程	日付	主要な内容	
ガイダンス	4 月 11 日	開発環境及び各種ツールの使い方、開発プロセス	
	4 月 16 日		
スプリント 1	4 月 18 日	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 23 日		
	4 月 25 日	ロジカルシンキング二回目	
	4 月 30 日	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 5 月 7 日		
スプリント 2	5 月 9 日	詳細設計についての講義、組込みシステム開発	
	5 月 14 日		
	5 月 16 日		
	5 月 21 日		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	エンタープライズシステム開発、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		
	5 月 30 日		
	6 月 4 日		
	6 月 6 日		IoT システム開発
振り返り	6 月 11 日	PBL 発表会練習 1	
	6 月 13 日		
	6 月 18 日		
	6 月 20 日		PBL 発表会練習 2
	6 月 25 日		
PBL 発表会	6 月 27 日	外部向け PBL 発表会	
スプリント 4 +振り返り	7 月 2 日	全体システム開発・修正	
	7 月 4 日		
	7 月 9 日		
	7 月 11 日		デモ検討
	7 月 16 日		スプリントレビュー
最終発表会	7 月 18 日	最終成果物ツールデモ(内部向け)	

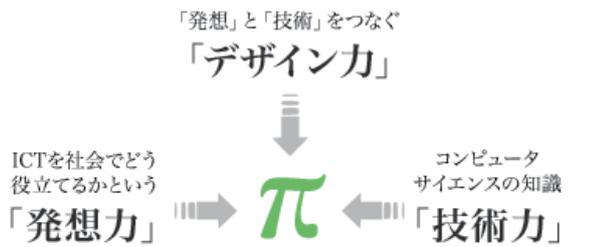


図 1 本コースの PBL の目指す人材像
Fig. 1 PBL Abilities

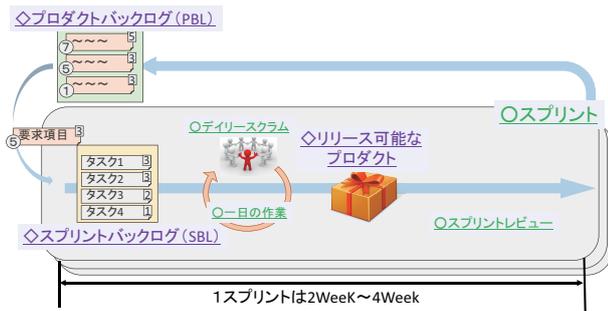


図 2 Scrum フレームワーク
Fig. 2 Scrum Framework

また、ロジカルシンキングではアイデア創発だけでなく PBL に必要となるチーム内での意見のやりとりやブレインストーミングの技法なども演習に取り入れた。

Scrum

プロジェクトマネジメントとして、今回はアジャイル開発の一つである Scrum[3] を用いた。講義では Scrum の基本的なフレームワークを学び、以後 PBL の中で学生が主体となり Scrum を運用できるよう指導を行った。

図 2 に Scrum の基本的なフレームワークを示す。

Scrum はプロダクトバックログと呼ばれる要件リストを元に活動を行う。開発の期間をスプリントと呼ばれる期間に区切り（通常 1~4 週間）、スプリントごとに小さな成果物を作成しながら徐々に製品品質を高めていく手法である。スプリント内では、スプリント計画ミーティングにて、そのスプリントで実現すべきプロダクトバックログ項目を決定し、時間見積もり可能なタスクに分割する。このタスクを並べたものはスプリントバックログと呼ばれる。スプリント期間内では、毎日作業を始める前に各自の担当タスクの状況確認やその日に行うタスクの宣言を行う（デイリースクラム）。以後スプリントが終了するまで繰り返す。

スプリントの終了前に、そのスプリントの成果物の確認を行うスプリントレビューを行う。スプリントレビューにより品質の確認を行うとともに、製品の方向性の確認や不備の洗い出しなどを行う。スプリントの最後では、そのスプリント自体の振り返りを行う。多くの場合 KPT 法が用いられ、そのスプリントで効果があり維持すべき項目 (Keep), 問題点 (Problem), 問題への対策や新たに取り

組むべき項目 (Try) の三点を洗い出す。KPT を通じチームのスプリントの運用自体を改善していく。

PBL 第一では、2 週間を 1 スプリントとして、Scrum を実施した。自発的にプロダクトバックログや KPT の改善が行われていることが確認できた。

ICONIX

Scrum においては、具体的な設計物や開発方法論は主体としていないため、開発方法論として ICONIX[4] を取り入れた。

ICONIX は UML を主体とした開発方法論で、ユースケースによる要求分析から、ロバストネス図によるロバストネス分析、その後シーケンス、クラス図と徐々に具体化していくことで要求から実装までの対応関係が理解しやすい。作業フェーズと図間の関係がわかりやすいため、設計の有用性が理解されやすいのではという意図で導入した。

ICONIX の実施は Scrum のスプリント内で、選択したプロダクトバックログ項目を対象に分析設計を行う。多くの場合アジャイル開発や Scrum ではコードを主体とした開発方法が多いが、設計経験の少ない学生が主体となった開発では、設計のフェーズを取り入れることで、チーム内で合意の取れた設計を行った上で実装作業を行うことで、破綻しにくいプロジェクトとなるようにした。

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 の活動の一つずつ追いつながらチュートリアル形式で実施した。第二スプリント以降は、各活動のタイミングだけ伝え基本的に学生が主体となりスプリントを回すように指示した。講義による座学と、チュートリアルスプリントで各図を順を追って作成した。

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。各スプリントのスプリント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。PBL 発表会では、多く参加者を募り、外部の企業の方や学生に対して発表を行う場を設けた。

3. 教材

PBL 第一では IoT システムの提案と実装を題材としている。IoT システムを一から作成するには PBL の時間は短すぎるため、基盤となる環境や教材を用意した。図 3 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エンタープライズシステムに分けられる。組込みボードには、提案するサービスに合わせてセンサとアクチュエータを接続し、情報の取得や返答を行う。サーバには組込みボードから送られたデータを用いるアプリケーションを配備する。図中の緑色の背景

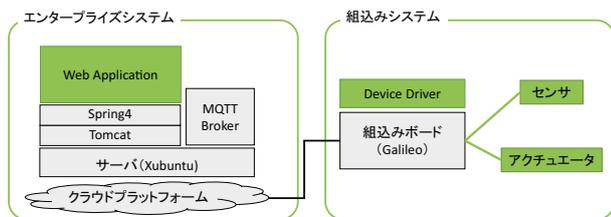


図 3 IoT 向け PBL 教材
Fig. 3 Educational Materials for IoT



図 4 Intel Galileo
Fig. 4 Intel Galileo

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

組み込みボードとして、Intel 社の Galileo^{*2} ボードを用いた。Galileo ボードは、Intel Quark SoC X1000 (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。図 4 に Galileo ボードを示す。

ピンサインは Arduino^{*3} と互換性があり、Linux の上位に Arduino 互換の環境も搭載している。このため、Arduino と同等に比較的容易に組み込みボードが扱えるとともに、必要であれば Linux のアプリケーションを追加して機能強化を行うことができる。また、Ethernet ポートも搭載されており、容易にネットワークに接続することができる。加えて、背面の Mini PCI Express ポートに WiFi モジュールを搭載することにより、WiFi 経由でネットワークに接続することもできる。

Spring4

Spring Framework4(以後 Spring4)^{*4} は Java のアプリケーションフレームワークで、Dependency Injection(DI) コンテナより、コンポーネントの管理を行うことができる。

^{*2} <http://www.intel.com/content/www/us/en/do-it-yourself/galileo-maker-quark-board.html>

^{*3} <http://arduino.cc/>

^{*4} <http://spring.io/>

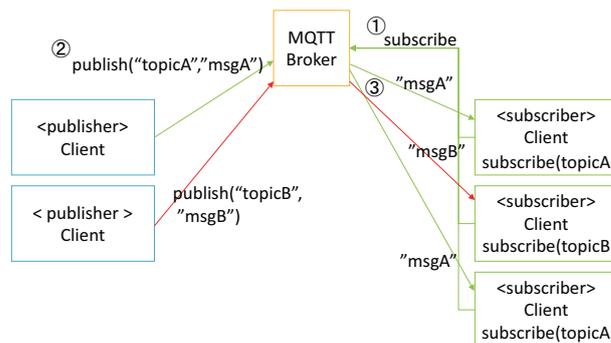


図 5 MQTT
Fig. 5 MQTT

特に Web アプリケーション向けに多くのコンポーネントが提供されており親和性が高い。

Spring4 では、従来 DI の制御に多くの XML 定義が必要であった部分が改善されており、アノテーション記述により多くの設定が可能となっている。

また、Spring4 より Spring Boot と呼ばれる簡易アプリケーション作成機構が取り入れられ、多くの機能が予め設定されており、最低限の記述のみでアプリケーションを構築することができる。

Spring Boot では Tomcat へのデプロイ作業無しで、組み込み Tomcat 上で簡易に作成中のアプリケーションを動作させ確認することが可能であるため、チーム開発では個々に開発とテストを行うことができる。個々に作成した成果物は最終的にはマージし、同サーバで動作している Tomcat 上にデプロイして運用する。

MQTT

Message Queuing Telemetry Transport(MQTT)^{*5} は IBM が提唱する IoT や M2M 向けの軽量プロトコルである。TCP/IP 上で動作する Publish / Subscribe モデルによるプロトコルで HTTP に比べ軽量である。

MQTT の通信方式について図 5 に示す。MQTT のネットワークは 1 つ 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 型であるため容易にスケールアップすることができるとともに、Publish,Subscribe 間の結合も小さくそれぞれ個別に作成しテストすることができる。

利用時は 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、WebSocket や HTTP の実装と比べ比較的容易に MQTT を用いたクライアントプログラムを作成することができる。

MQTT Broker には、MQTT の OSS 実装である

^{*5} <http://mqtt.org/>

表 2 センサとアクチュエータ
 Table 2 Sensors and Actuators

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 (縦設置)	QTR-1RC
フォトフレクタ・モジュール (横設置)	QTR-1A
3D コンパス (加速度・地磁気センサ)	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
ホイールポテンシオメータ 10k Ω	
トリマポテンシオメータ 10k Ω	
デジタルポテンシオメータ 10k Ω	MCP4131
プログラマブル赤外線リモコン受信チップ	SIS-2
1.2 インチ 8x8 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	

Mosquitto^{*5} を用いチームサーバ上に配備した。Galileo の OSS の MQTT ライブラリ Phao^{*6} を移植した。

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。表 2 に、センサ・アクチュエータの一覧を示す。

センサ・アクチュエータはロジカルシンキングの前の講義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た。

本来自由な発想を得るためには、事前にこのようなセンサ・アクチュエータを提示すると、アイデアが固定かされてしまうデメリットがあるが、現実的に実現可能なアイデアとするために、予め提示を行った。

センサ・アクチュエータのドライバの作成は、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 6 に前項の要素技術を用いた今回のシステムアーキテ

^{*5} <http://mosquitto.org/>

^{*6} <http://www.eclipse.org/paho/>

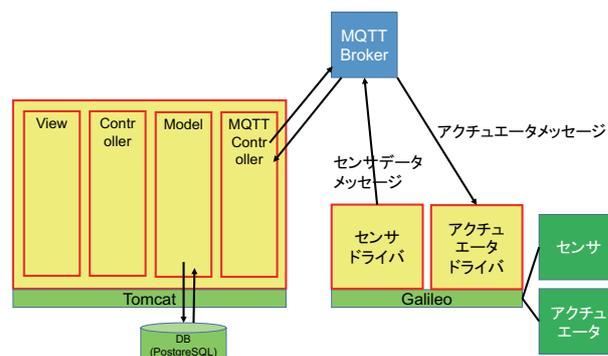


図 6 教材 IoT システムのアーキテクチャ
 Fig. 6 Architecture of Educational Materials

クチャを示す。

外界とのインタラクションはセンサ・アクチュエータを接続した組み込みボード Galileo で行い、取得したセンサ値を MQTT で送信する。またアクチュエータに応じたメッセージを購読する。

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と、予め用意する部分簡単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 7 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。VM による提供も検討したが、導入の容易性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 VM での提供も検

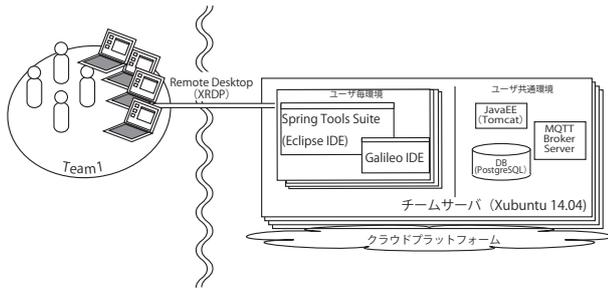


図 7 開発環境

Fig. 7 Development Environment

討している。

3.4 学習効果の評価

PBLによる学習効果の評価項目として、以下のようなものを考えている。PBLの場合、項目によっては作業担当者が明確にわからない場合もあるため、このような項目は、チームを対象とした。

チーム評価項目

- スプリントごとの Scrum のプロダクトバックログ項目の推移
- スプリントごとのスプリントバックログ項目の推移
- KPT 項目の推移

個人評価項目

- 担当者ごとのスプリントバックログ項目の推移
- 発表の積極性
- ソースコード量の推移
- コミット量、コミットコメント

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が搭載されており、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。

現在、PBL 第一は第 3 スプリントを終え、システム内の主要な機能の実装がほぼ確認できた状態である。今後、発表会に向けてシステムのブラッシュアップを行っていく。

5. おわりに

本稿では、本年度より取り入れた組込み分野向けの PBL の実施内容について紹介した。PBL において、プロジェクトを進める上で必要となるプロジェクトマネジメント力と、プロジェクトを実現するための技術力を両面から教育を行った。各種講義と演習を通じてプロジェクトマネジメント力を、教材と演習を通じて技術力の向上を図った。

本教材を用いた PBL は本年度が初回であるため、教育効果を評価するためのデータ今後、年度ごとのデータを取り、教材や講義内容の改善を行っていきたい。

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.