

スマート端末から利用可能な TA 業務支援システムの開発と評価

林裕之¹ 橋本圭介¹ 古賀雅伸¹

概要: 近年, スマートフォンやタブレット端末, ウェアラブルデバイスのようなスマート端末の急速な普及に伴い, 教育現場におけるスマート端末の導入が進んでいる. 本研究ではスマート端末の TA(ティーチングアシスタント) 業務支援への応用を考える. TA とは大学や短大, 高専における多人数講義で学生が抱える多様な学習上の問題に指導教員のみで対応することが困難な場合に授業補助者として配置される大学院生などを指す. TA は講義中絶えず講義室内を歩き回りながら, 学生の出欠や課題のチェック, 学生からの質問への個別対応, 学生の学習状況の把握などの業務を行う. そこで, 本研究ではこれらの業務の効率化を図るため, 持ち運びが容易なスマート端末から利用可能な TA 業務支援システムを開発する. 本システムはスマート端末から利用可能なため, 歩き回りながら業務を行う TA でも利用でき, タッチ入力による直観的な操作で出欠や課題のチェックを素早く行うことができる. また, 学生の情報をサーバーで一括管理することでリアルタイムに情報を共有できるため, 講義室内に散らばった状態でもスマート端末を介して TA 間でスムーズに情報を共有することができ, 業務の効率化が期待できる.

キーワード: ティーチングアシスタント, スマート端末, 教育

1. はじめに

近年, スマートフォンやタブレット端末, ウェアラブルデバイスのようなスマート端末の急速な普及に伴い, 教育現場におけるスマート端末の導入が進んでいる. 実際, 文部科学省が 2011 年に発表した「教育の情報化ビジョン」によると, 2020 年までにすべての学校で 1 人 1 台のタブレットの配備やデジタル教科書・デジタル教材の普及促進, ネットワーク環境の構築などを目標として掲げている [1]. また, 国内だけでなく海外でも教育機関へのタブレット端末の導入が進んでいる.

本研究ではスマート端末の TA(ティーチングアシスタント) 業務支援への応用を考える. TA 制度とは大学院生に学部生等に対する助言や実験, 演習等の補助業務を行わせ, 大学教育の充実と大学院生のトレーニングの機会提供を図るとともに, これに対する手当ての支給により, 大学院生の処遇の改善の一助とすることを目的とした制度である [2]. そして, この補助業務を行う大学院生を TA と呼び, TA は講義中絶えず講義室内を歩き回りながら, 学生の出席や課題のチェック, 学生からの質問への個別対応, 学生の学習状況の把握などの業務を行う. 昨今は情報化の

進展に伴い, プログラミング演習のような PC を利用した多人数講義が増加しているため, TA の必要性が高まっている. なぜなら, PC を用いた演習は授業内容だけでなくハードウェア・ソフトウェアの両方から多岐に渡り, 学生が抱える多様な学習上の問題に対して教員がきめ細かく対応することが困難だからである [3][4]. しかし, 受講している学生数に対して TA の数が不足し, 学生が TA の対応に不満を持ち, TA の負担が大きくなることも少なくない. 加えて, 出席や課題の提出情報を授業後に集計する大変が大変である, 挙手した順番がわからないため, どの学生から対応するべきかわからないなどの課題がある.

これらの課題を解決しようと様々な研究が行われている [5][6]. [5] では学生からの質問の管理や保存, [6] では出席管理についてのシステムや手法が提案されているが, 演習課題のチェックについては触れられていない.

そこで, 本研究では学生の出席や演習課題をチェックした情報, および学生からの質問を管理する, スマート端末から利用可能な TA 業務支援システムを開発する. 本システムはスマート端末から利用可能なため, 歩き回りながら業務を行う TA でも利用でき, タッチ入力による直観的な操作で出席や課題のチェックを素早く行うことができる. また, 学生の出席や課題の提出状況, 質疑応答や課題提出の順番などの情報をサーバーで一括管理することでリア

¹ 九州工業大学
Kyushu Institute of Technology



図 1 紙によるチェック作業
Fig. 1 Check work by paper



図 2 挙手による呼び出し
Fig. 2 Call by a show of hands

リアルタイムに情報を共有できる。そのため、講義室内に散らばった状態でもスマート端末を介して TA 間でスムーズに情報を共有することができ、業務の効率化や上記の課題の解決を期待できる。

2. TA 業務の課題

従来、TA が出席や演習課題のチェック作業を行うとき、紙媒体を用いることが多かったが、TA は複数人で業務を行うことが多いため、紙を用いて作業を行うと各 TA が自分のチェック結果のみを持つことになる(図 1)。そのため、講義中に他の TA のチェック結果を把握できず、学生の出席や提出状況を把握することができない。加えて、TA は講義後にチェック結果を集計し、データを手作業で表計算ソフトなどに打ち込む必要があり、その負担は小さくなく、ミスが混入する可能性もある。

また、学生が質疑応答や課題を提出したいとき、挙手によって TA を呼び出す必要があったが、TA は挙手の順番を正確に把握することができない。そのため、つい近くの学生や目立つ学生から対応する傾向があり、挙手する人数が多いと遠くの学生や目立たない学生はいつまでも対応してもらえない。結果として、対応に不満を感じる学生が多くなる(図 2)。

そこで、上記の TA 業務の課題を解決するために TA 業

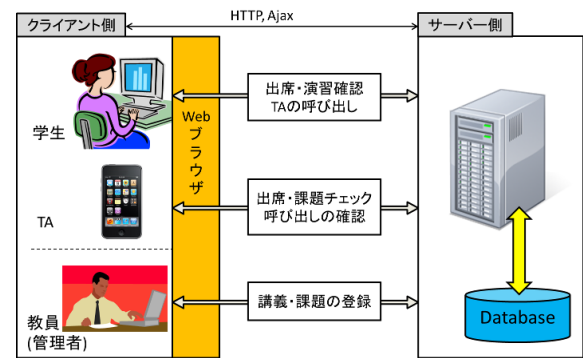


図 3 TASKit のシステム構成
Fig. 3 System configuration of TASKit

務支援システム TASKit を開発した。

3. TA 業務支援システム TASKit

3.1 概要

TASKit とは、前述の課題を解決することにより、TA 業務の効率化を図ることを目的とした Web アプリケーションである [7]。主な特徴として、Web ブラウザがインストールされた端末であれば利用可能、データベースによる情報の一括管理などが挙げられる。また、本システムは TA だけでなく、TA の呼び出しや出席や課題の提出状況の参照のため学生や教員も利用するため、各ロールごとに機能を分割し、それぞれの要求を満たすようにシステムを構築している。

3.2 システム構成

前章で述べた TASKit のシステム構成を図 3 に示す。TA は学生の質問に応じたり、演習課題のチェックを行ったりするため、絶えず講義室内を移動し、立ちながら業務を行う。また、学生はプログラミング演習などをデスクトップ PC やノート PC を用いて作業を行う。そのため、TA はスマート端末の Web ブラウザを通して操作を行い、学生と教員はデスクトップ PC やノート PC の Web ブラウザを通して操作を行うことを想定している。

TA がチェックした出席や課題、学生からの TA の呼び出し、教員が作成した課題などの情報は HTTP 通信により Web サーバーへと送信され、データベースに格納される。また、Ajax による非同期通信を用いることで、他の端末によってデータが更新されてもブラウザのリロードなしに更新され、常にリアルタイムにデータを確認できる。

3.3 ロールと機能

TASKit では各アカウントに対して、教員・学生・TA というロールを割り当て、各ロールごとに UI や機能が異なる。以下では TASKit について、教員・学生・TA それぞれの利用方法について述べる。

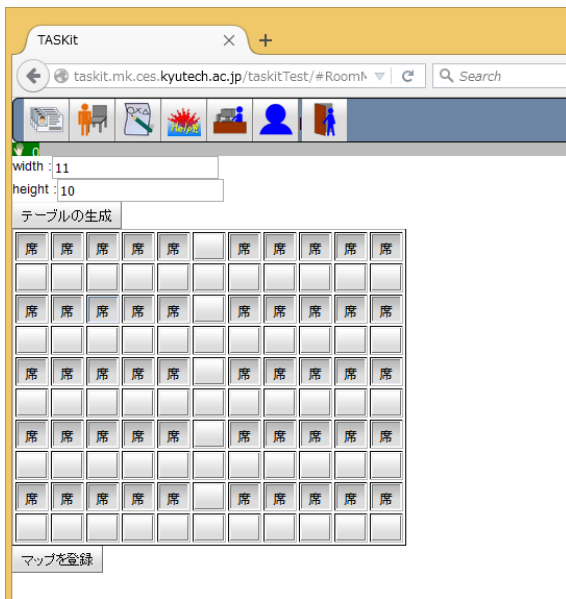


図 4 座席表の作成画面

Fig. 4 Screen of creating a seating chart

3.3.1 教員

教員のロールでは講義における演習課題，学生，TA の登録や，講義室の座席配置情報の設定，学生の座席設定ができる。

教員用の機能の例として，図 4 に講義室の座席配置情報の設定画面，図 5 に全学生の成績情報を参照する画面を示す。図 4 では「width」と「height」に数字を入力し，「テーブルの作成」ボタンを押すことで入力した数のセルのテーブルを作成することができる。各セルはトグルボタンになっており，選択することで「席」と表示され，そのセルを座席に指定することができる。図 5 では画面左端の列から順に「学生」が各学生のユーザー ID，大文字の数字が何回目の講義かを表し，各回の出席状況が表示されている。そして，括弧書きの数字が課題番号を表し，各課題の提出状況が表示されている。また，教員は全学生の出席や課題の提出状況に関するデータを CSV 形式でダウンロードすることができる。

3.3.2 学生

学生のロールでは講義室内で自身が着席している席の登録，講義における出席と課題の提出状況の確認，質疑応答や課題を提出するための TA の呼び出しができる。

学生は TA に自身が着席している座席を知らせるため，授業の始めに着席した座席を登録する必要がある。座席設定画面を図 6 に示す。図 6 では講義室の席の配置が表示されており，「空」が空席，「☆」が自分が登録している席，「満」が他の人が登録している席，「-」が通路を示している。学生は「空」の座席を選択することで自分の席として登録できる。

各学生は講義における自身の出席と課題の提出状況をログイン直後のページで確認することができる。確認画面を

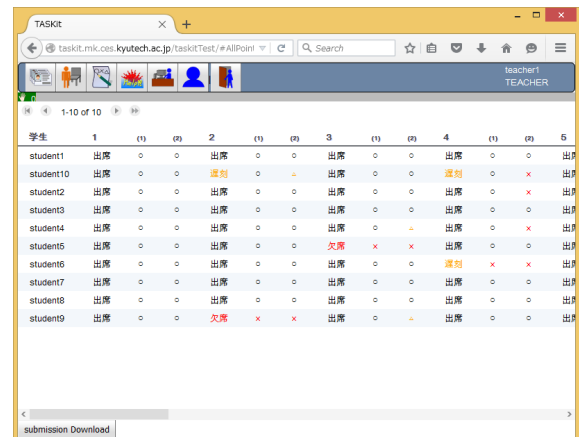


図 5 学生の成績一覧

Fig. 5 Screen of student performance list

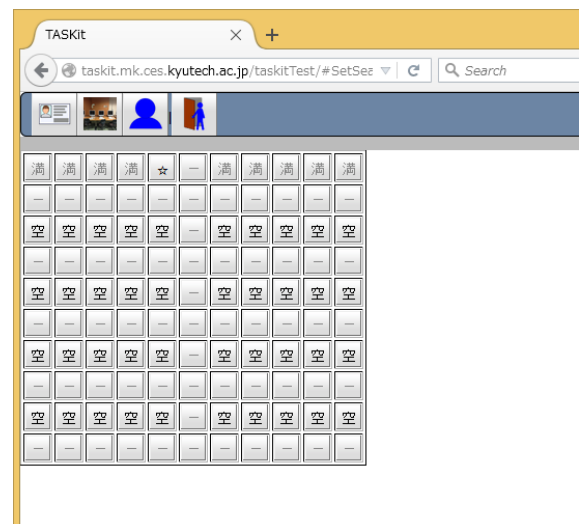


図 6 座席登録画面

Fig. 6 Seat registration screen

図 7 に示す。画面左端の列から順に「回」は何回目の講義なのか，「出席種別」は各回の出席状況，それ以降は演習課題の番号を表しており，課題を提出していれば採点結果に合わせて「○」や「△」などの記号が表示される。

演習課題が未提出の場合は「提出」と書かれたボタンが表示され，このボタンを押すことで呼び出しを TA に通知することができる。また，画面右の「Help!!」と書かれたヘルプボタンを押すことで質問の呼び出しを TA に通知することができる（以下，このボタンを押下することを「ヘルプコール」と呼ぶ）。学生が TA を呼び出している状態を図 8 に示す。図 8 は学生が演習課題 8-1 のチェックを希望しており，現在 4 番目で順番待ちであることを示している。このように学生は自分が現在何番目に待っているか常にリアルタイムに知ることができる。

3.3.3 TA

TA のロールでは呼び出している学生一覧の確認，各学生が着席している座席の確認，すべての学生の出席や課題

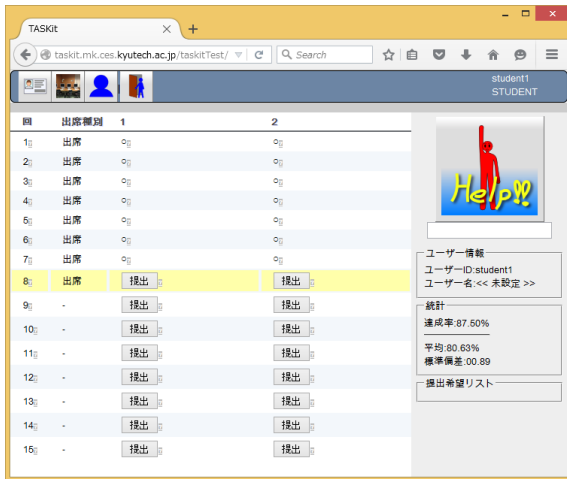


図 7 確認画面

Fig. 7 Confirmation screen

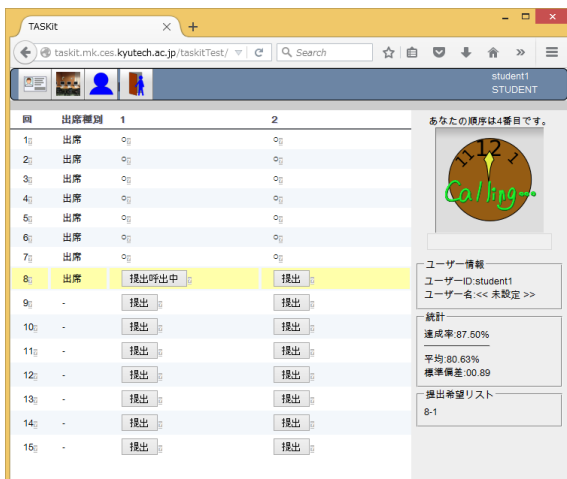


図 8 TA を呼び出している画面

Fig. 8 Screen of the state that is calling TA

のチェックができる。

TA は学生からヘルプコール、あるいは課題の提出の呼び出し要求が送信されると画面上に通知される。図 9 に呼び出し中の学生一覧を表示している画面を示す。画面左端の列から順に「日付」は呼び出し要求が送信されたタイムスタンプ、「呼び出しユーザー」は呼び出している学生のユーザー ID、「メッセージ」はヘルプコールであること、「対応中」は対応している TA のユーザー ID、「提出希望リスト」は提出したい課題名を表示している。また、「場所」ボタンを押すことで学生の座席を確認できる。図 9 の画面右側の枠内において、赤色の四角部分が呼び出している学生の位置を表している。

呼び出している学生一覧表から対応したい学生の行を選択することで、その学生の情報を入力する画面に遷移することができる。図 10 にその入力画面を示す。この画面では学生が提出を希望している演習課題の横に電話マークが表示される。また、情報を入力する際、チェック項目は

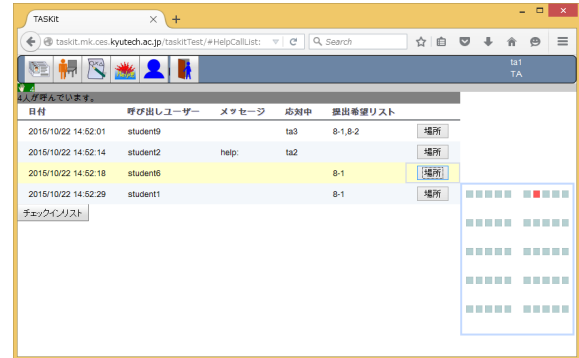


図 9 呼び出している学生一覧と学生が着席している座席の確認

Fig. 9 Screen of the student list that is calling TA and confirmation of seat student seated

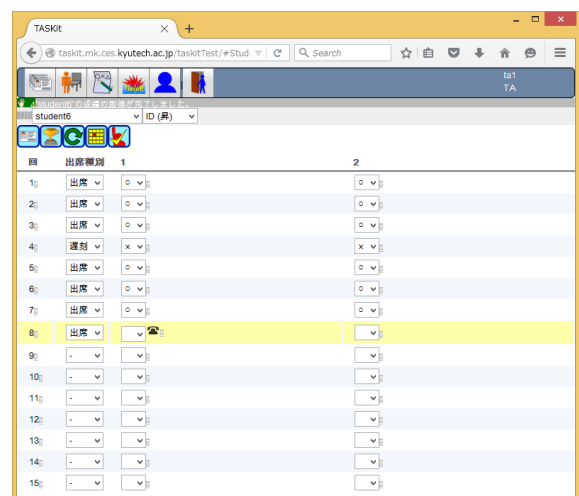


図 10 学生の情報を入力する画面

Fig. 10 Screen to enter the student information

プルダウンメニューになっているため、各チェック作業は 2 タップ (クリック) で終わることができる。そのため、TA は学生の席まで移動した後、タッチ入力による直観的な操作で課題のチェックを素早く行うことができる。また、チェックした内容は即座にサーバー上のデータベースに格納され、Ajax による非同期通信により、ブラウザのロードなしにデータが更新され、学生や TA は常にリアルタイムにデータを共有できる。

4. 評価

本システムは 2010 年度後期から九州工業大学情報工学部システム創成情報工学科の複数の PC を利用した多人数講義で利用されている。表 1 に過去 3 年の利用状況、図 11 に実際に TASKit を利用している授業の風景を示す。

評価では 2014, 2015 年度に TASKit を利用して TA 業務を行ったことのある大学院生 13 名を対象に Web アンケート調査を実施し、その結果を用いて本システムの評価を行った。評価内容としては、SUS(System Usability Scale) を利用してユーザビリティの数値的な評価を行った。また、

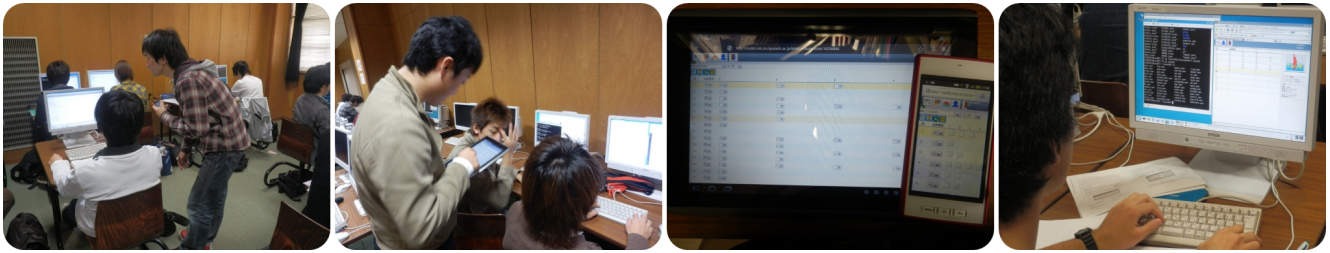


図 11 実際の利用風景

Fig. 11 Actual use landscape

表 1 過去 3 年の利用状況

Table 1 Usage of the past three years

導入年度	授業名	学生数	TA 数
2012	データ構造とアルゴリズム	92	4
	プログラミング応用 S	84	5
	プログラム設計	91	4
2013	データ構造とアルゴリズム	84	4
	システム制御コンピューティング	110	3
	シミュレーション物理	96	3
	プログラム設計	89	4
	プログラミング応用 S	81	5
2014	データ構造とアルゴリズム	87	5
	システム制御コンピューティング	94	4
	シミュレーション物理	90	4
	プログラミング S	87	4
	物作りプロジェクト	87	3

それに合わせてアンケート参加者に本システムを利用することにより TA 業務が効率的になったと思うかどうか、そして本システムを利用して感じた良い点・悪い点について回答して頂いた。

4.1 SUS を用いた評価

SUS を利用してユーザビリティの数値的な評価を行う。SUS とは Jhon Brooke 氏により 1986 年に開発され、ユーザビリティの評価に広く利用されている方法の一つである [8]。回答者に 10 個の質問を行い、1(全く同意しない)から 5(非常に同意する)の 5 段階で回答してもらう。その際、奇数番目をポジティブな質問、偶数番目をネガティブな質問とする。回答を集計後、奇数番目の項目は回答番号から値を 1 引き、偶数番目の項目は 5 から回答番号を引くことにより、すべての項目を 0 から 4 で評価し、足し合わせた合計値を 2.5 倍して 0 から 100 の数値 (以下、この数値を SUS スコアと呼ぶ) でユーザビリティを評価する。SUS スコアを計算するためにアンケート参加者に質問した項目を表 2 に示す。

今回のアンケート調査で得られた回答を基に SUS スコアを計算した結果を図 12 に示す。全体の平均値が 75.192、利用科目数が 2 科目以下の人 (10 名) の平均値が 72.25、3 科目以上の人 (3 名) の平均値が 85 であった。

[8] によると、[9] や [10] で行われた SUS スコアの解析結果を基に、Bangor 氏らは SUS スコアが 50 より下を許容範囲外、50 から 70 までを許容範囲の限界、70 より上を許容範囲として、ユーザビリティを評価することを提案している。今回の調査結果をこの範囲に照らし合わせてみると、平均値は利用回数の違いに関係なく 70 を超えているため、本システムのユーザビリティは許容範囲に収まっていることがわかる。全体と 2 科目以下の平均値における 95%信頼区間の下限値が 70 を下回っているため、本システムのユーザビリティが許容範囲であると断言できないが、少なくとも許容範囲の限界に収まっていると評価できる。さらに、SUS スコアを下げている原因を解明するため、各項目における評価の平均値を計算し、値が 2.8 (=SUS スコアを 70 とするために必要な各項目の平均値) を下回っている質問項目をピックアップしてみたところ、質問番号 2,5,8,10 の項目で下回っていることがわかった。このことから、UI のデザインや操作性に問題があるのではないかと考えられる。

また、[11] によると、評価対象のシステムの利用頻度が高いほど SUS スコアが高くなる傾向があると報告されている。2 科目以下と 3 科目以上の平均値における 95%信頼区間で若干の重なりが確認できるので、welch 法を用いて t 検定を行ったところ 0.0223 (< 0.05) となったため、両者の平均値に有意差があると判断した。したがって、本システムにおいても利用回数が多いほど SUS スコアが高くなる傾向があると言える。

4.2 質問とコメントによる評価

アンケート参加者に本システムを利用することにより TA 業務が効率的になったと思うかどうか、「はい」、「いいえ」、「変わらない」で回答してもらったところ、すべての参加者が「はい」と回答したため、本システムには有用性があると判断した。

また、アンケート参加者に本システムを利用して感じた良い点・悪い点に関するコメントをしてもらった。表 3 にいくつかのコメントを示す。挙手の順番や課題の提出状況などがすぐに確認できるようになったため、TA 業務を効率的にこなせるようになったというコメントが寄せられる一方で、前節で考察したように、UI のデザインや操

表 2 質問項目
Table 2 Question items

質問番号	質問内容
1	TA 業務の際、TASKit を頻繁に利用したいと思う。
2	TASKit を利用するには事前に詳しい説明が必要となるほど複雑に感じた。
3	TASKit は容易に使いこなすことができると思った。
4	TASKit を利用するのに専門家のサポートが必要だと感じる。
5	TASKit のコンテンツやナビゲーションは十分に統一感があると感じた。
6	TASKit の UI には一貫性のないところが多々あったと感じた。
7	たいていの人は、TASKit の利用方法をすぐに理解できると思う。
8	TASKit を利用していても操作しづらいと感じた。
9	TA 業務の際、TASKit を十分に利用できる自信がある。
10	TASKit を利用し始める前に知っておくべきことが多くあると思う。

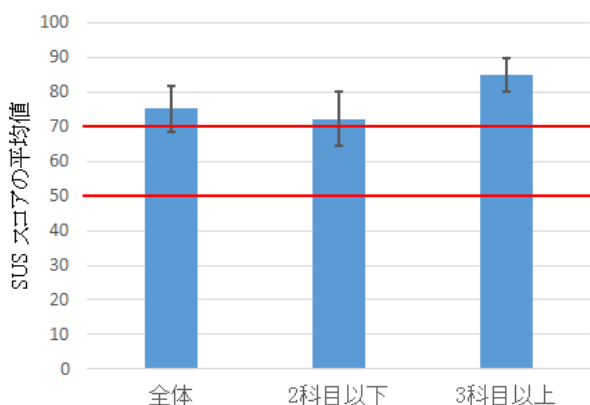


図 12 SUS スコアの平均値 (エラーバーは 95%信頼区間を表す)

Fig. 12 The average of SUS score (Error bars represent 95% confidence interval)

表 3 良い点・悪い点に関するコメント

Table 3 Comments about the good and bad point

	コメント
良い点	挙手した順番がわかる 受講生の課題提出順に不公平がない 課題のチェックが円滑に進む 集計作業をしなくてよい 格段に TA の負担が減った
悪い点	レイアウトがあまり良くない箇所がある 強いて言うならばデザインが悪い タブレットだと使いやすくスマホだと使いにくい 呼び出し音による通知等がないのでわかりづらい 通知等がないので画面を見続ける必要がある

作性に関する不満が多くみられた。また、Web アプリケーションの特性上、通知機能の実装はブラウザによって異なるため、本システムでは未実装となっており、通知機能に関する不満も多くみられた。

5. おわりに

本研究では学生の出席や演習課題をチェックした情報、および学生からの質問を管理する、スマート端末から利用

可能な TA 業務支援システムを開発した。

アンケート調査により、SUS を用いて本システムが最低限のユーザビリティは備えているという評価を得ることができた。その他にも、本システムの有用性や TA 業務の効率化ができていていることが示唆された。

今後の予定として、アンケート調査によって見えてきた UI のデザインや操作性の問題点について改良していく必要がある。また、通知機能の実装なども挙げられる。

参考文献

- [1] 文部科学省：教育の情報化ビジョン (online), 入手先 http://www.mext.go.jp/b_menu/houdou/23/04/_icsFiles/afiedfile/2011/04/28/1305484_01_1.pdf (2015.10.20).
- [2] 文部科学省：ティーチング・アシスタント (TA) について (online), 入手先 http://www.mext.go.jp/b_menu/shingi/chukyo/chukyo4/003/gijiroku/07011713/001/002.htm (2015.10.20).
- [3] 西澤泰彦：多人数講義における問題点と教育方法, 名古屋高等教育研究センター, Vol.6, pp. 45-57 (2006).
- [4] 加藤利康, 石川孝：プログラミング演習支援システムにおける学習状況把握機能の提案, 情報処理学会研究報告. コンピュータと教育研究会報告, Vol. 2013-CE-120, No.2, pp.1-8 (2013).
- [5] 井上真由美, 大即洋子, 中川正樹：PDA を利用したティーチングアシスタントによる演習授業支援システム, 情報処理学会全国大会講演論文集, Vol.67 (2005).
- [6] 呉 鞠, 葛 崎偉：多人数情報処理教育講義における出席について, 教育実践総合センター研究紀要, Vol.17, pp.1-9 (2004).
- [7] TASKit, <http://www.taskit.mk.ces.kyutech.ac.jp>.
- [8] Thomas, T. and William, A.: *Measuring the User Experience, Second Edition: Collecting, Analyzing, and Presenting Usability Metrics*, Morgan Kaufmann (2013).
- [9] Bangor, A., Kortum, P. and Miller, J.: Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale, *Journal of Usability Studies*, Vol.4, No.3, pp.114-123 (2009).
- [10] Tullis, T. S.: *SUS Scores from 129 conditions in 50 Studies*(online), 入手先 <http://measuringux.com/SUS-scores.xls> (2013.3.30).
- [11] McLellan, S., Muddimer, A. and Peres, S. C.: The effect of experience on system usability scale ratings, *Journal of Usability Studies*, Vol.7, No.2, pp.56-67 (2012).