

## タブレット端末を利用したスマートな投票システムの実現 Implementing a Smart Voting System Using Tablet PCs

藤江 翔太郎<sup>†</sup> 片山 真也<sup>†</sup> 白松 俊<sup>‡</sup> 大園 忠親<sup>‡</sup> 新谷 虎松<sup>‡</sup>

Shotaro Fujie Shinya Katayama Shun Shiramatsu Tadachika Ozono Toramatsu Shintani

### 1. はじめに

近年、タブレット端末が会議等の意思決定の場面で利用される機会が増えている。本研究では、投票および集計作業の効率化を目的として、タブレット端末を利用したスマートな投票システムを開発している。紙媒体で投票を行う場合には、投票環境や集計の手間が必要となるが、タブレット端末で投票システムを実現することにより、この問題を軽減できる。タブレット端末を用いた投票によって起こる秘匿性の課題の解決と、1人1票の投票を保証するための手法について述べる。また、本稿では、投票システムにおける、各種モジュールを説明する。

### 2. タブレット端末を利用したスマートな投票システム

本研究では、容易性、秘匿性、平等性、および、納得性の4点を満たす投票システムを実現するために、タブレット端末を導入した投票システムの実現を目指す。容易性とは、投票および集計作業の準備および実施が効率化されることを意味する。秘匿性とは、すべての投票者の投票内容等の情報が守秘されることを意味する。平等性とは、1人1票であることを意味する。納得性とは、投票者自身で投票が受理されたことを確認可能であることとする。

本システムは、容易性と納得性を満たすため、OS環境を選ばない、Webアプリケーションとして実現した。また、通信にSSLを用いることで、秘匿性を実現した。平等性を満たすには、先行研究として、パスワードを利用したBasic認証を用いた手法がある[1]。個人を認証することは、投票者に秘匿性に対する疑いを持たせること、および、投票における手間が容易性を下げる点が問題である。ここでは、タブレット端末に固有の端末IDを利用することにした。本システムでは、タブレット端末で動作する、投票アプリケーションを、ハイブリッドアプリケーションとして実現した。

#### 2.1 ハイブリッドアプリケーション

本システムの多くは、JavaScriptによるWebアプリケーションとして実現している。Webアプリケーションは、移植性が高く、投票環境の構築に有利であるが、JavaScriptは、取得できるクライアントの情報が限られている。そのため、Webアプリケーションから、タブレット端末の端末IDを取得することは困難である。いっぽうで、ネイティブアプリケーションは、クライアントの情報取得は容易であるが、タブレット端末のOSや機種ごとに投票アプリケー

ションを実装する必要がある、移植性が悪くなってしまう。本システムの投票アプリケーションでは、移植性が比較的困難なインタフェース部分、通信部分をWebアプリケーションとして実装する。タブレット端末の環境ごとに、処理の異なる、端末IDの取得部分をネイティブアプリケーションとして実装する。ネイティブアプリケーション内から、Webアプリケーション部分呼び出し、内部的に、データ通信をおこない、ひとつの投票アプリケーションとして実現する。ネイティブアプリケーションとWebアプリケーションの欠点を補った、ハイブリッドアプリケーションとして実現することで、1人1票の保証を実現した。また、ネイティブアプリケーションの内部にWebアプリケーション部分が含まれるため、通信部分のコードを外部から見られず、安全性が高まる。本システムでは、ネイティブアプリケーションを、Objective-Cを用いた、iOSアプリケーションとして実装した。Objective-CからJavaScriptを呼び出すために、UIWebViewクラスをつかい、WebページをiOSアプリケーション内部に埋め込んだ。ハイブリッドアプリケーションの構成を図1に示す。

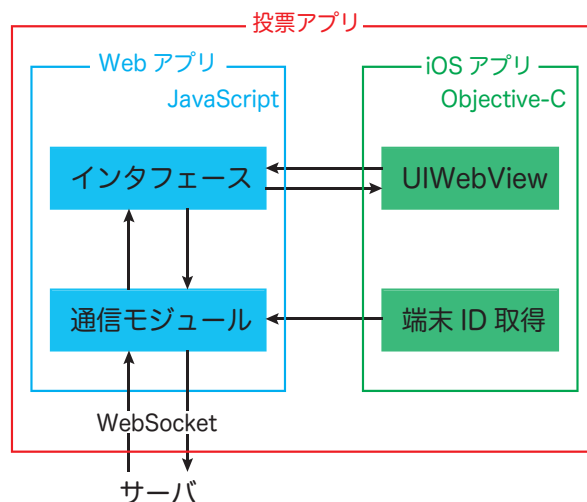


図1 ハイブリッドアプリケーション構成図

### 3. 実装

#### 3.1 システム概要

図1に、投票の管理、および、投票の状況を確認するための投票管理Webページを示す。本システムでは、投票の管理とは、投票の開始、終了を意味する。投票管理者は、図1の、左上のボタンを押すことによって、投票の開始、終了を管理する。開始ボタンを押すと、ページウィンドウが全画面表示になる。また、図1の投票管理Webページには、①投票タイトル、②投票ラベル、③投票数、④投票状況が表示されている。本システムにおける投票タイトルとは、投票における質問や題目を表し、投票タイトルに対

<sup>†</sup>名古屋工業大学 工学部 情報工学科

Dept. of Computer Science,  
Nagoya Institute of Technology

<sup>‡</sup>名古屋工業大学 大学院工学研究科 情報工学専攻

Dept. of Computer Science and Engineering,  
Graduate School of Engineering,  
Nagoya Institute of Technology

する選択肢を、投票ラベルとする。投票タイトル、投票ラベル、投票数は、図 3 左図に示す投票設定 Web ページで設定する。投票者には、投票番号が割り当てられている。投票番号とは、投票者につき 1 つ割り当てられるユニークな番号である。システムは、投票番号を用いて、投票を管理する。投票を行うと、図 1 の④のように、投票者自身の投票番号がハイライト表示され、投票者は、投票が正しく行われたことを確認でき、投票における納得性を満たした。投票状況の表示には、WebSocket によるサーバ側からのプッシュ配信を受け取ることで、リアルタイムに投票状況を確認できる。

①	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24
25	26	27	28	29	30
31	32	33	34	35	36

投票管理ページ

図 2 投票管理ページ

図 2 投票管理ページ

図 3 左図に投票設定 Web ページを示す。本システムでは、2 種類の投票を設定できる。ひとつは、賛成反対を問う簡潔な投票で、①「賛成反対・選択肢フォーム」で設定する。選択肢にラベルと点数を持たせた、アンケート形式の投票を、②「評価フォーム」から設定する。図 2 左図は、「評価フォーム」での設定例である。①、②で、投票タイトル、投票ラベルの設定をする。③「設定」で、投票者数を上限なく設定することができる。設定したデータは、サーバを介し、投票管理ページにプッシュ配信する。投票管理ページをリロードせず、設定項目をリアルタイムに反映でき、円滑な設定を行うことができる。④「結果」から、結果表示ページに移動する。結果表示ページでは、現在まで行ったすべての投票結果の集計が確認できる。集計したデータは、CSV ファイルとして書き出すことができ、投票データの解析を容易にする。

図 3 右図には、タブレット端末による、投票アプリケーション実行例を示す。画面遷移のない、1 ウィンドウで完結するインタフェースとした。図 1 の投票管理 Web ページによる投票開始と同時に、プッシュ配信によってウィンドウ上にパーツが表示される。リアルタイムに投票設定が反映され、投票者はただちに投票することができ、投票開始時のデータ再読込、といった操作を減らした。ウィンドウには、①自身の投票番号、②投票ラベル、③投票ボタンが表示される。投票者は、投票ボタンを押すことで、投票を行う。端末 ID により、タブレット端末とサーバの接続確立を行う。本システムには、投票開始から、投票終了までの間、ボタンを何度も押すことにより、そのたび、投票の変更が可能であるという特徴がある。本システムを初めて利用する投票者は、操作間違いによっておこる、本来の意思と異なった誤投票を防ぐことができる、という利点がある。また、投票管理者の投票終了操作と同時に、投票が

締め切られるため、不正な投票や、終了時の混乱も避けることができる。

すべての通信規格に WebSocket を採用している。サーバはサーバサイド JavaScript の Node で構築している。Node における WebSocket API を Socket.IO といい、通信は JSON 形式の文字列データでやりとりする。サーバでは、各データの送受信、データベースおよびログファイルへの書き出しを行っている。

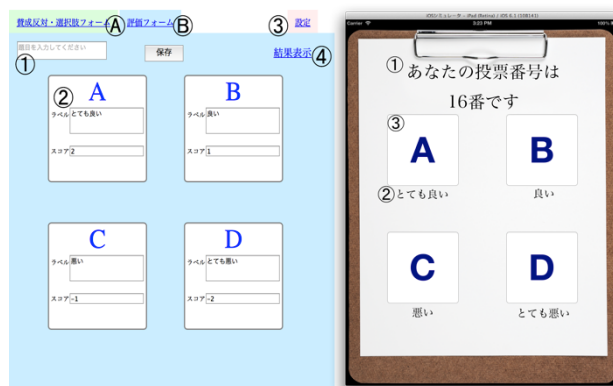


図 3 投票設定ページとタブレット端末による投票

#### 4. 考察

投票者はタブレット端末にインストールされた、投票アプリケーションを起動する。投票主催者が、投票管理ページで開始ボタンを押してから、投票者のタブレット端末に設定項目が、プッシュ配信され、画面に表示されるまでの時間を計測した。この時間をロード時間と呼ぶ。1 クライアントにおけるロード時間は数ミリ秒であった。通信するデータはすべて、バイト数の少ない文字列であるため、このような結果となった。従来の紙媒体による投票では、大規模な投票の環境構築に多大なコストを費やすが、本システムを用いることで、円滑に環境を構築できた。

本システムでは、インタフェース部分を Web アプリケーションで構築しているため、投票主催者が、自由にインタフェースや投票フォーマットの設定を追加できる機能の実現も可能である。

#### 5. おわりに

本稿では、従来の紙媒体による、投票の環境構築や、集計の効率化を目的とした、投票システムの実現について述べた。WebSocket によるプッシュ配信を用いることで、投票の環境構築の効率化、投票者の誤操作防止を実現した。

容易性、秘匿性、平等性、および、納得性の 4 点を満たす投票システムの実現手法として、ハイブリッドアプリケーションという実装方法を用いた。ハイブリッドアプリケーションでは、ネイティブアプリケーションからクライアントの情報を管理し、Web アプリケーションで通信およびインタフェースを構築した。

#### 参考文献

- [1] 中里 純二, 藤本 賢司, 菊池 浩明, “個人情報漏洩を防止する Web アンケートのセキュリティ強化”, 情報処理学会論文誌, Vol.46, No.8 (2005).