学園祭模擬店の食数管理システムの開発

5 年 B 組　中盛 楓也
指導教員　藤野 智美

１．要約

本校学園祭の模擬店では、食券を使用して商品の販売を行なっているが、食券の管理は手作業でカウントされており、ミスが発生したり、管理に長時間を要している。これらの問題点を解決すべく、本研究では食数管理ネットワークシステムの開発を目指した。

キーワード　ネットワーク、バーコード、食券管理

２．研究の背景と目的

本校学園祭の模擬店では、図１のように、顧客は本部にて現金で食券を購入し、生徒が運営する模擬店で食券と交換することで商品の購入が可能となる。各模擬店は１時間に一度、売り上げた食券を本部まで運び、台紙に食券を貼り付けて販売数をカウントし、各店舗の売り上げ数を報告する。本部では各模擬店の売上数を把握後、在食数を割り出す。学校各所に点在する模擬店からの往復移動時間、台紙貼り付け作業含めて約 20 分の作業である。これは本部が売れた数に応じて、客に売り切れ商品を表示したり、商品の不足による返金を減らすことを目的としている。

本校ではこの模擬店運営形態を 10 年以上続けているが、この一連の作業には、①労力が多い　②食券運搬や台紙貼り付け中の食券紛失等の手作業によるミスや重複使用のミスが発生する　③リアルタイム性に欠けるという 3 つの問題点がある。これらの問題点を解決すべく、本研究では食数管理システムの開発を目指した。

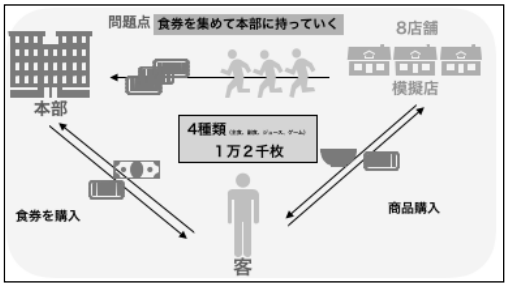

図１　システム導入前の模擬店販売の流れ

３．研究内容

研究目的で述べた3 つの問題点を解決するために、以下の内容を実現できるシステムを開発する。

・ネットワークによる食券購入数リアルタイム管理(問題点①、③の解決)
・食券重複使用チェック(問題点②の解決)

市販されている既存のシステムを導入する場合、専用のバーコードリーダーやシステムの購入費等で数十万円の導入コストがかかる。本研究では導入コストを抑えるために、PC やスマートフォンなどすでに学校や模擬店員が所持している多様なデバイスに対応させ、機材の追加購入の必要性をなくすことを目指した。

また今後の販売戦略に生かすために以下の機能も追加した。

・購入時間の記録

・購入データの書き出し

３．１　バーコードの設定

食券には、使用日、食券種類(主食、ジュースなど)、重複チェック用にそれぞれ違った固有の番号の 3 つの情報を記す必要がある。

食券の大きさやデータの読み取りやすさ及び、12,000 枚の食券を識別するために必要な情報量は少ないため、バーコード最低桁数の 8 桁の JIS8 を選択した。学校に授業で使用したバーコードリーダーがあったことや、食券の形やサイズを考えると、QR コードよりもバーコードが適していると判断した。

図 2 のように食券の裏側にそれぞれバーコード(JIS8)を印字した。なおバーコードは以下のルールで設定する。

【バーコードの設定方法】

1 桁目・・使用日

（2 日間のうち、どの日の食券か)

2 桁目・・食券種類

（主食＝1、ジュース＝2 など）

3～7 桁目・・食券固有番号

（重複チェック用にそれぞれ異なる）

図 2　作成した食券

３．２　ソフトウェアの開発

オフラインでも使用できるよう Web 形式ではなく、ソフト形式での開発を選択した。また、様々なデバイス種類に対応するため、Unity で開発を行い、ネットワークサーバーは Photon を利用した。

トラブルが起きていない通常運用時のアルゴリズムは図 3 のようになる。

研究当初はネットワークトラブル(wi-fi 環境の弱さ)は発生しないと想定していたため、最初にネットワークが健全な状態においてのみ正しく機能するアルゴリズムを組んだ。

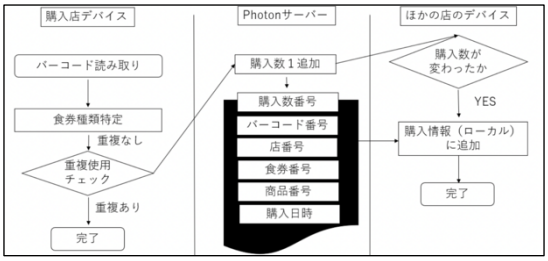

図 3　通常運用時のアルゴリズム

【アルゴリズム１(通常運用時)】

①ソフト起動時に店舗専用のバーコードでログインし、どの店舗が端末を使用しているかを設定する。

②商品販売を行った店舗が、客から受け取った食券のバーコードをバーコードリーダーで読み取る。

③バーコード 2 桁目から食券種類を特定し、店舗情報からどの商品（焼きそばなど）が購入されたか特定する。もし同一店舗で主食が 2 種類以上販売されており、食券種類だけで商品を特定できない場合、図 4 の通知画面が表示され、キーボードキー等でソフトの使用者に商品を決定してもらう。

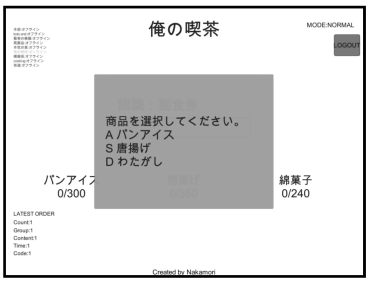

図 4 商品選択通知画面

④これまでの購入データからバーコード番号が一致しないか確認し、重複使用されていなかったら、[累計会計数(全端末での総数)][バーコード番号][店舗番号][商品番号][購入日時]をサーバーに送信する。ネットワークに参加している全端末は、サーバーに送られたこれらのデータをローカルでセーブし、サーバーにはデータを蓄積しない。その理由はネットワークが切れて、オフラインで使用する場合を想定したためである。

ネットワークに参加している全端末に購入データを送信したら会計は完了する。また、各店舗では図 5 のように、自店舗の商品の累計販売個数が表示され、本部では図 6 のように、全店舗の累計販売個数が表示される。

図 5 各店舗画面

図 6 本部画面

学園祭当日の運用を想定したシステムとして、2019 年 8 月 22 日に学園祭の当日の店舗の場所に PC を設置し、当日使用される 8 台で同時多発的に会計を試みた。結果的にトラブルは発生せず、適切に会計処理がなされた。

３．４　初日の運用とフィードバック

学園祭 1 日目に本システムを運用した結果、ネットワーク回線が弱い屋外の発団が、オフラインの状態で会計処理を行ったため、データが破損した。このトラブルによって 8 店舗(屋内 4、屋外 4)中 4 店舗が適切に運用でき、残りの 4 店舗では正常に運営できなかった。内訳として、成功した店舗はいずれも学校屋内のネットワーク回線が行き届いている場所であり、失敗した 4 店舗全てが屋外の店舗だった。以下発生した問題点である。

①LOGOUT ボタンを押すことでローカルに保存していたデータを PC 上のファイルに書き出してセーブするように設定していた。これはアプリケーションの動作を軽くするためである。そのため事前に模擬店部員全員に LOGOUT ボタンを押して再起動することを伝えていたが、当日 LOGOUT ボタンを押さずにウィンドウを閉じた発団があり、オフライン中だけでなく、オンライン状態の時のデータも破損してしまったというヒューマンエラーが発生した。

②１日目の運用結果から考察すると、本校の wi-fi は接続する場所やネットワーク使用人数によって、断続的に回線が途切れてしまうことがわかった。Photon の特性上、一度 PC のネットワーク回線が途切れてしまうと、たとえその後 PC がネットワークにつながったとしても、アプリケーションを再起動しないとサーバーに再び接続できない。これに対する対処法として、以下が挙げられた。

・問題点①のヒューマンエラーによるアプリケーション再起動におけるミスを解決する方法として、アプリケーションを全画面化し、ウィンドウを閉じるボタンを押せないようにすることで、意図しないアプリケーションの終了を防いだ。
・問題点②のネットワーク環境問題に対しては以下の種類のトラブルシューティングを開発した。

３．５　トラブルシューティング１：ネットワーク切断

ネットワークが切断され、オフライン使用時にデータが共有されずそのまま破損してしまった。この問題を解決するため以下のアルゴリズムを組んだ。
①今までの購入データと、オフライン中の自端末で会計した分のデータから重複使用チェックを行う。
②購入データはサーバーに送信せず、ローカルで保存しておく。
③ネットワークに再接続したら、オフライン時に保存したローカルデータをサーバーに送信し､3.6 の方法によりそれまでの他のデバイスの購入データを受け取る。

３．６　トラブルシューティング２：サーバーに途中参加した場合の想定

サーバーに途中参加したデバイスは、サーバーにはデータがないため、それ以前の他の店舗の購入データを持っていない。この問題を解決するため図 7 のアルゴリズムを組んだ。
①現在の累計会計数と自端末が持っている購入データの数を比べ、持っていないデータあった場合、そのデータの累計会計数番号をサーバーに送信して、他端末にデータを送信してもらうよう要請する。

②サーバーに送信された要請データを持っていた場合、そのデータをサーバーに送信する。
③要請したデバイスは、他の端末が上げた購入データをローカルに保存して要請を解除する。

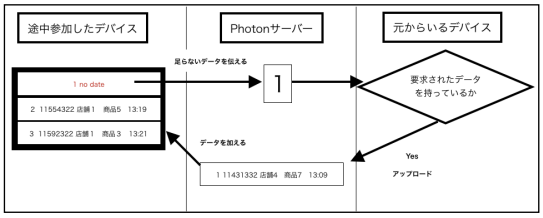

図 7 サーバー途中参加アルゴリズム

４．結果

3.5 および 3.6 のトラブルシューティングを施した学園祭 2 日目の本システムの運用結果は以下のようである。
① 1 日目のような会計データの破損などの問題はなく、適切にシステムを運用することができた。
②オフラインの端末がオンラインに復帰するのに時間がかかった店舗があり、リアルタイム性に欠けるという結果になった。
7 店舗(屋内 3、屋外 4)中 7 店舗全てで適切に運用でき、7 店舗中 4 店舗(屋内 3、屋外１)が常にリアルタイムでデータの送信に成功した。残りの 3 店舗(屋外 3)は最大約 1 時間のデータのディレイがあった。

また、2 日目のシステム運用で得られた、CSV 形式の購入データから売れた個数と時間の統計をとることで、来年以降の販売の参考にできる。

５．考察

3.5 および 3.6 の 8 月に実施したシステムテストおいて、ネットワークに関して問題が起こらなかったにも関わらず、学園祭当日はネットワークが弱かった。これには模擬店以外の学園祭発団のネットワーク使用など、テスト時とは異なるネットワーク使用状態になったことが考えられる。トラブルシューティンを導入したことでオフラインにも対応でき、考えられる不測の事態全てに対応できたと考えられる。実際に、学園祭 2 日目にはシステムは適切に運用できた。

来年度のシステム運用に向けて、オフライン中は音を鳴らしてアプリケーションの再起動を促すなど、ユーザーインターフェースの方向からの、リアルタイム性の問題について解決に努めたい。

６．今後の展望

本研究によって、ネットワークによる食券購入数リアルタイム管理・食券重複使用チェック・購入時間の記録・購入データの書き出しを達成することができた。これによって各店舗それぞれ 4 時間の作業時間削減と CSV データによる販売時間や販売量のデータを入手でき、来年度以降の販売戦略の参考データが取得できる。

研究方法で示したトラブルシューティングのように、本研究のシチュエーションのようにデータ数が少ない場合、全端末でデータをローカル保存することで、ネットワーク回線が悪い状況にも対応できる。

本研究で、「ネットワークによる小規模コミュニティでのデータ共有」,「デバイスの種類に捉われないソフト開発」,「オフライン状況下でのオンラインと同様の振る舞い」に成功した。本研究で開発したソフトを一般化することで、他学校やフリーマケット等の小規模マーケットや入場者管理システム等への汎用性が期待できる。また本システムのアルゴリズムを使用することで、リアルタイムで教師と生徒の問題提供や回答送信ソフトを作り、教育現場の生徒管理、ログの記録に大幅に貢献できる。このように、本研究は小規模コミュニティでのリアルタイム情報共有システムに展開でき、教育現場など様々なシチュエーションで実社会に応用できるものと考える。

７．参考文献

[1]Unity4 入門
出版社：SB クリエイティブ　2013 年 7 月 25 日出版　著：浅野 祐一、荒川 巧也 、森信虎
[2] PUN2 で始めるオンラインゲーム開発入門
https://connect.unity.com/p/pun2deshi-meruonraingemukai-fa-ru-men-sono1

８．謝辞

今回の研究を行うにあたり、顧問の藤野先生にご指導いただきました。この場を借りて、深く御礼申し上げます。