

次世代携帯電話 P I D の提唱

猪狩一真、宗直喜、高久幸夫、平泉康志、山田淳

工学院大学 工学部 国際基礎工学科

1 はじめに

今の携帯電話は、電話・メールは当然のこと、音楽が聴け写真が撮れ、動画も撮影・視聴することができる。いつでもどこでも Web に接続することができ WWW 上の様々な情報を利用することが可能になり、1人1台所有することが当たり前になっている。そこで次世代の携帯電話、“誰もが必要とする機器”として本論文は「PID」(Personal Indispensable Device)を提唱する。

2 PID

PID はこの世に存在する膨大な情報から、その時・その場所で・その人に必要とされる情報を提供するため、あらゆる学生が必要とする携帯機器を学生の視点で提案するものである。

この PID を教室内での使用を想定した”Digital Interactive Class”、図書館内での”Digital Library”、鍵としての用途”Digital Gate Management”、食堂での”Digital Catering System”の4つの具体例を示す。

3 Digital Interactive Class

授業中に学生に PID、教師にパソコンを使用することでプリントの配布や、課題の回収を容易にする。方法としては PID とパソコンを無線 LAN などをつなぎ、プリントなどを送受信する。PID にすべてのデータを保存しておくのは無理なので、大きいデータは学内にある個人サーバ保存しておく。必要に応じて PID に受信し、閲覧する。

授業中

PID の画面は小さいため、資料などは見にくい。なので、PID をディスプレイ(電子ペーパー)などにつなぎ、表示する。課題を解く場合は一度、紙に解いてから PID でスキャニング(もしくはカメラで撮る)を行い、教師側のパソコンに送る。また、PID にキーボードとディスプレイをつなぎ、論文などを書く。また、選択式のアンケートや、課題、マークシートなどはディスプレイに問題を表示し、PID で回答を選択する方法がいいと思われる。

授業外

授業でもらった資料を紙としてほしい場合、PID から個人サーバに印刷を要求して学内にあるプリンターで印刷する。

これにより、資料の配布、回収の時間が削減されるが、紙媒体ではないので直接書き込めないのが難点である。解決方法のひとつとしてタブレットを使用して書き込む方法がある。

4 Digital Library

Digital Library とは、図書館に置かれている書物を電子化することによって、より効率よく図書館を利用するためのシステムのことである。これを利用するためには、パソコンとその周辺機器が無線 LAN 及び有線によって繋がれている環境が必要。

電子図書の貸し出し

P I D で自分が求めている図書の管理番号を図書館サーバに送信すると、ライブラリの中から図書が選り出され、P I D に返信される。容量の大きい図書は個人用のサーバに送ることもできる。

また、やりとりする図書も1冊全てではなく、必要なページ・部分だけを得ることもできる。

このときに OK を選択すると、その情報が図書館に備え付けられたプリンターでコピーされます。

5 Digital Gate Management

図書室や PC 室など情報管理が重要視される部屋の入退室は慎重かつ利用しやすくならなくてはならない。そこで入り口にゲートを設け、無線による PID の個人情報読み取りによってゲートを開閉するシステムを考案する。

無線認証なので PID をわざわざ取り出す必要はなく、ポケットやバッグに入れたまま、ゲートを通るだけで認証が可能である。

研究所などでは、密閉性を考え、非接触型 IC カードの技術を使用する。入室時のみ PID で開錠する。

教室では、ドアにゲートを設置し PID の生徒情報を読み取り、そこで出欠をとる。退室時にも PID を読み取り、最終出欠登録をします。

6 Digital Catering System

昼食時、多くの学生食堂では、学生が食堂のカウンタ、メニューの受け取り場所で大行列を作っている。これはトレイを取り、メニューを注文し、受け取り、カウンタで精算するのに多くの時間がかかるためである。さらに、空いている席を探すとなると10分以上かかってしまうことがある。そこで、「PID」を用いてもっと効率のよい食堂のシステム「Digital Catering System」を提案する。

我々が提案する「Digital Catering System」とは、利用者自身が注文・精算・配膳することで待ち時間を極力少なくできる食堂システムである。具体的には以下の要領で利用者は食堂を利用する。

利用者はPIDで食堂にアクセスし、混雑状況を把握する。その状況は食堂の職員が随時更新する。

利用者は席についたらPIDで注文・料金の精算をする。

調理場は注文を正常に受け付けたら受付番号を返信する。

調理が完了次第、その旨を該当のPIDに送信、利用者は料理を取りに行く。

場合によっては、食事後、利用者は感想等のアンケートに答えPIDで食堂に送信する。

アンケートに答えた利用者にはクーポンを発行し、次回の利用時に利用することができる。

このシステムにより、利用者は列に並ぶ必要がなくなり、料金清算の手間が省けるだけでなく、調理師が効率よく調理できることにより、料理の待ち時間も短縮することができるのである。

この「PIDによるDigital Catering System」をSpecCで実装すると次のようになる。

```
event Inquiry //混雑状況問い合わせイベント
event Order; //注文イベント
event Pay; //精算イベント
```

```
behavior PID{
    int student_number; //学籍番号
    void SendMenu()
    { //ここにメニュー送信処理を記述する}
    main() {
        notify Inquiry; //混雑状況問い合わせイベ
```

ント

```
        //混み具合をPIDに表示
        notify Order; //注文イベント発生
        SendMenu(); //メニュー情報を送信する
        wait Pay //精算イベント待機
        //料金が精算され次第
        //精算記録を受信する
        //精算記録をPIDに表示
        wait Cooked; //調理完了イベント待機
    }
};

behavior Cafeteria
void ReciveMenu()
{
    //ここにメニュー受付処理を記述する
}
void SendConditon()
{
    while(close){ //閉店するまで
        //混み具合を食堂の職員が確認
        //職員が端末で学内サーバに情報を送信
        //web上に情報が公開される
        //混み具合 = 今の混雑状況
        wait Inquiry; //混雑状況問い合わせイベント
        //PIDに混雑状況を送信する。
    }
}

main() {
    ReciveMenu(); //注文を受注する
    //メニューの情報を受信する
    notify Pay; //精算イベント発生
    //自動精算を行う
    //精算記録をPIDに送信
    notify Cooked; //調理完了イベント発生
}
};

behavior Register{
    void Payment()
    {
        //ここに料金精算の処理を記述する
    }
    main()
    {
        wait Pay; //精算イベント待機
        Payment(); //料金精算の処理
    }
};
```