

部室入退室管理班

吉田享平 *藤原浩一 †濱田和幸 ‡
廣田公大朗 §宇根岬 ¶山岡聖弥 ||
木村将人 ** 鈴木雅隆 †芹澤拓也 ‡‡

2018/2/5

* 情報理工学部 1 回生
† 情報理工学部 1 回生
‡ 情報理工学部 1 回生
§ 情報理工学部 1 回生
¶ 情報理工学部 1 回生
|| 情報理工学部 1 回生
** 情報理工学部 1 回生
†† 情報理工学部 2 回生
‡‡ 理工学部 1 回生

目次

1	はじめに	2
2	概要	2
3	実装	3
3.1	読取ソフトの実装	3
3.2	サーバーソフトの実装	4
4	反省点と改善点	5
4.1	アプリの反省点	6
4.2	プロジェクトの反省点	6
4.3	アプリの改善点	7
4.4	プロジェクトの改善点	7
5	今後の展望	7
6	おわりに	8
7	参考文献	9

1 はじめに

文責：吉田享平

入退室の管理をすると会員のモチベーションが上がり、会内全体の活発な活動につながるのではないか。このような思いから本プロジェクトは発足した。

現在、部室を監視するシステムが存在するが、諸事情で正常に機能しないこともある。その現状を打開する目的も本プロジェクトは持ち合わせている。

2 概要

文責：吉田享平

本プロジェクトのアプリケーション群（以下、本アプリ）は情報読み取りソフトウェア（以下、読取ソフト）、集約サーバ（以下、サーバ）、Web アプリケーションに分けられる（図1）。

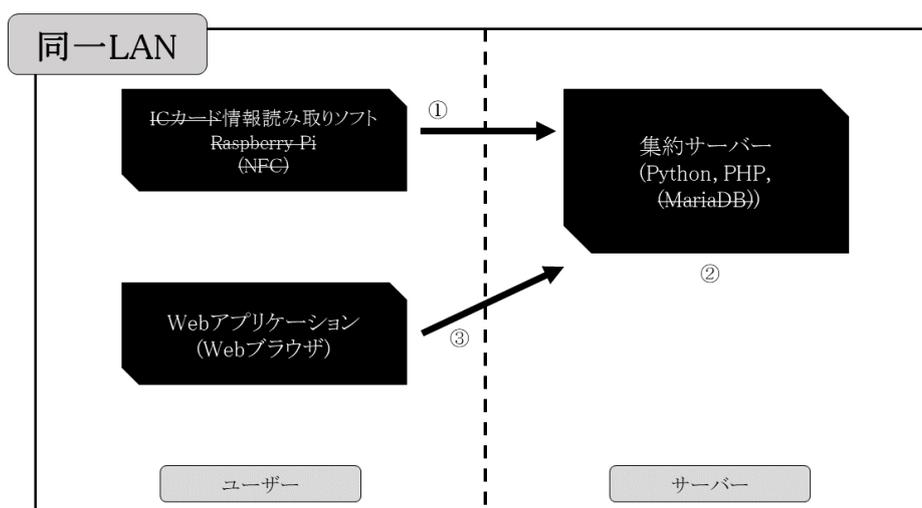


図1 概要図

読取ソフトは、ユーザから情報を受け取り、サーバに情報を転送する（図1①）。本プロジェクトでは、この入力を NFC 読取装置（図2）によって IC カード（学生証）をタッチすることで実現した。



図2 NFC 読取装置

サーバは入力された情報の蓄積および加工を行っている（図1②）。データのやりとりはファイルを通じて行っている。

webアプリケーションはユーザが他の人の情報を含め、入退室情報を得るためのソフトウェアである。サーバからデータを取得し（図1③）、可視化したものである。

3 実装

3.1 読取ソフトの実装

文責：吉田享平，濱田和幸

3.1.1 環境

以下の環境を想定した。

- ハード：Raspberry Pi 3
- OS：Raspbian
- 読取装置：SONY RC-S320
- 開発言語：Python 3.5.5

しかし、Raspberry Pi は用意できなかったため、仮想環境としてLinux環境で開発を行った。

3.1.2 実装

このソフトは、大まかに①ICカード情報の読取、②サーバ送信時のフォーマットの整形、③サーバへの送信と3つの処理に分けられる。

①ICカード情報の読取

ここでは、NFC 読取装置を利用し学生証をかざすと生徒番号を取得するように実装した。ユーザが任意のタイミングで読み込めるよう、無限ループの仕様となっている。しかし、アプリ実行時のエラーを回避するため、ループ内処理 1 回につき 3 秒のスリープ処理を設けている。また、カード読み取りにはシェルを利用している。

```
1 #ICカード読み取り
2 def card_read(data):
3     read_command = "felica_dump|_head_-n9|_tail_-n1|_used's/#_card_IDm_//'"
4     while(1):
5         #シェルの実行
6         read_flag = subprocess.check_output(read_command,shell=True)
7         read_flag = read_flag.replace("\n","")
8         if(len(read_flag)):
9             if(read_flag.find("can't") == -1 :disappointed:
10                if(data != read_flag or data == ""):
11                    return read_flag
```

Listing 1 カード情報読み取り

②サーバ送信時のフォーマットの整形

サーバに送るデータのフォーマットは以下である。

```
日付,学生番号
```

Listing 2 データ送信フォーマット

なお、日付はシェルを利用し取得している。プログラム内では string 型で扱っている。

③サーバへの送信

『paramiko』というモジュールを利用し、ssh 通信でコマンドを送ることによってサーバへデータを送信している。

```
1 #ssh接続
2 def exec_ssh(ip_, port_, user_, pass_):
3     ssh = paramiko.SSHClient()
4     ssh.set_missing_host_key_policy(paramiko.AutoAddPolicy())
5     ssh.connect(ip_, port=port_, username=user_, password=pass_)
6     ssh_session = ssh.get_transport().open_session()
7
8     return [ ssh, ssh_session ]
```

Listing 3 モジュールによる ssh 通信の確立

```
stdin, stdout, stderr = ssh.exec_command('DATA=$(date);echo_{DATA},'+readed_number+'>/var/www/html/data/aaa')
```

Listing 4 ソースコードから ssh 通信でコマンドを送信

3.2 サーバーソフトの実装

文責：藤原浩一

3.2.1 サーバー環境

サーバー環境は以下の環境を構築した。

- 構築環境：Hyper-V(Windows Server 仮想環境)
- OS：Cent OS 7
- 開発言語：PHP (X) 系, Python 3.5.5
- データベース：MariaDB

3.2.2 サーバー側の処理概要

サーバー側では、読み取り機から送られてきたデータをもとに主に三つの処理を行っている。

- データ受信後のデータ成形
- データベースより個別データの呼び出し
- ユーザーへの情報提供

3.2.3 データ受信後のデータ成形

読み取り機で取得された学生証番号データは、scp を使いサーバーに転送される。その後、サーバー側でデータと共にデータが送られてきた時間を取得し各種ファイルにログ保存を行う。

今回は作業の関係上、後の Web アプリケーションでの処理を行いやすいように、「全ログデータ」と「直近 10 件のログデータ」の二種類を作成することになっている。

3.2.4 データベースより個別データの呼び出し

ログファイルには学生証番号と入退室の記録が記録されているが、後の Web アプリケーションで学生証番号を表示しても誰のものか判断がつかない。

そこで今回は MariaDB のデータベースソフトを用いて、学生証番号から氏名を呼び出せるように構築した。登録しているユーザーに関してはこれにて名前を表示することが可能となった。

3.2.5 ユーザーへの情報提供

今回、PHP を用いた Web アプリケーションをユーザーに提供することで、部室に誰がいつ入退出したのかを確認できるログ表示ページを作成した。

ユーザーは指定したアドレスにアクセスすることによって、部室の入退出履歴直近 10 件を確認することができる。データベースに登録しているユーザーに関しては、データベースより氏名を呼び出し表示することも可能になっている。登録されていないユーザーに関しては「未登録ユーザー」として表示されるようにした。

未登録ユーザーは以下の SQL 文を利用して登録の有無を確認している。

```
SELECT (*) FROM user WHERE id = "学生証番号";
```

この SQL 文の結果が「0」であれば、そのユーザーの学生証番号は未登録であると確認できる。

4 反省点と改善点

文責：吉田享平

ここではアプリとプロジェクトについて、2面からの反省点及び改善点を述べる。

4.1 アプリの反省点

プロジェクト発足当初、先輩方をはじめ多くの方が web アプリケーションに期待を寄せてくださっていた。企画段階では入室状況をアバターで表示しようとしていた。その案に対し、「ログインボーナスが欲しい（あれば面白いのではないか）」「レベル（入室頻度）によって着ることのできる服を変えられるようにしてほしい」など多くの良い意見をいただいた。それにもかかわらず、それらの機能のみならずアバター機能さえ実装に至らなかったというのは大きな反省点である。

原因については後述するプロジェクトの反省点と同様である。

4.2 プロジェクトの反省点

プロジェクトの反省点について、班員から貴重な意見を頂いた。

● 意見 1

文責：廣田公大朗

プロジェクト運営に関して

プロジェクトが正式に決定した例会後の顔合わせ以来、一度もメンバーが召集され会議が行われなまま追い込み合宿を迎えたことは大きな問題だ。再三、リーダーに対してメンバーの招集及び会議等の開催を求めたが改善は見られなかった。リーダーの責任は大きいと考えるが、リーダーに頼るだけでなくメンバーが自主的にプロジェクトを進めるなど、すべきであった。

最終的には完成したが、開発は追い込み合宿中に行われプロジェクトではなく事実上ハッカソンとなってしまう。

● 意見 2

文責：山岡聖弥

毎週の活動という意味では、プロジェクト活動としてふさわしくない活動状態であったことが反省点である。

確かに突貫作業で成果発表会には間に合わせる事ができた。しかし、もし毎週活動していたならば、余裕を持って作り上げることができただけでなく、より良い成果物を完成させることができたのではないかと考える。

このプロジェクトの進捗が毎週の活動で出ていなかったことは週報より明確であった。

次回以降はこのような状態にならぬよう、部員一同で各プロジェクトがきちんと成り立っているかを監視していくことが望ましい。

以上より、以下の反省点に整理できる。

- 開発合宿以外での活動が一切行われなかった。
- リーダーによる本プロジェクトの進捗管理が十分に行われなかった。

原因として、プロジェクトリーダーの管理能力の欠如が大きい。この点については、プロジェクト企画段階で

進捗管理を大きな目標としていたのに関わらずこのような事態に発展したことは大きく反省するべきである。また班員全員、特にプロジェクトリーダーの予定に余裕がなかったことも原因の1つである。企画段階で予定していた初回の全体ミーティングは何度か日程の調整を行ったが、結局開かれることはなかった。これが1つの要因となり、開発が合宿に集中してしまった。

4.3 アプリの改善点

1番の改善点はセキュリティの問題である。現行のシステムでは会内の人間によりすぐにクラッキングされる事態になることは明白である。早急に措置を講じるべきである。また、前述したように web アプリケーションの表示があまりにも簡素なものになっているので、ビジュアライズを進めていきたい。

4.4 プロジェクトの改善点

やはり進捗管理の徹底、プロジェクトメンバー全員が活動できる場の提供・日程調整が大きな改善点になるだろう。このプロジェクトが継続されるようならば、最重要課題として取り組みたい。

5 今後の展望

文責：藤原浩一

今後の展望に関して、後期の期間中に完成できなかった点を一番に考えると、まず RCC の部室内でサービス提供できるようなレベルまでの完成を目指したい。また、現在データベース上には学生証番号と名前データは登録されていないが、今後入部時に自動的に登録できるようなサービスも提供できれば良いと考えている。

現在の Web アプリケーションの動作イメージは図3のようになっている。

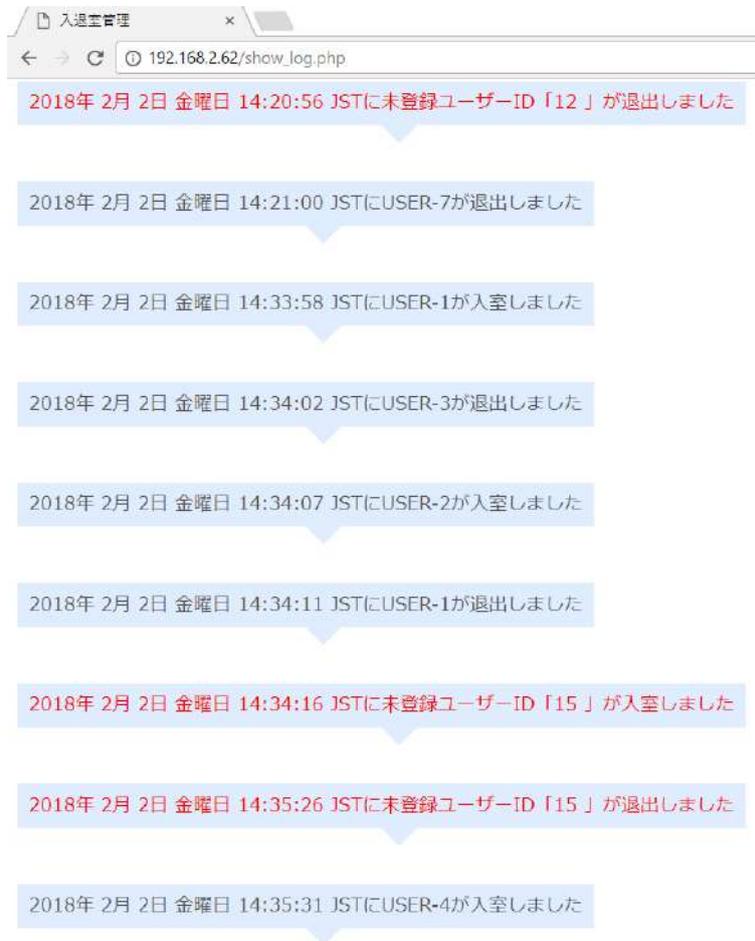


図3 Web アプリケーションイメージ

このように入退室のログが見れるのみとなっており、今部室に誰がいるのかを直感的に知ることができないので、今後のアップデートで一覧を表示できるようにしたい。

最終的な目標としては部室内 LAN のみならず RCC の公式ホームページからログ参照を行えるようにすることである。セキュリティに関しては、LDAP の認証を用いることで内部メンバーのみ閲覧できるように対策を講じたい。

6 おわりに

文責：吉田享平

「なんとか動くものが作れた。」と内心ほっとしている。開発合宿以外活動していなかったのにも関わらず、開発合宿に参加してくださったメンバーには心から感謝している。また、開発合宿に参加できなかったもののリモートで文章執筆をしてくださったメンバーや最後まで温かく見守ってくれたメンバーにも謝意を伝えたい。プロジェクトがなかなか進展しなかったのは、紛れもなくこの執筆者でありプロジェクトリーダーである私である。私の力量のなさに対し、本当に申し訳ないと感じている。

併せてこの活動でしか RCC の活動がないメンバーにはさらにお詫びを申し上げたい。当初はプログラミング講習も行いたいとしていたが実現はしなかった。なにかプログラミングについて不明な点があれば、頼りない私の知っている範囲でしか答えられないが質問を遠慮せずしてほしい。

最後になったが、いつかこのプロジェクトが本当の意味で終了し、部室において稼働することを願って報告書としたい。

7 参考文献

- PHP でファイルを開いて読み込む - Qiita
<https://qiita.com/tadsan/items/bbc23ee596d55159f044>
- ファイルから 1 行ずつ読み込む - fgets()
<https://webkaru.net/php/function-fgets/>
- CSS で作る！吹き出しデザインのサンプル 19 選
<https://saruwakakun.com/html-css/reference/speech-bubble>
- ディレクトリ内の特定ファイルをソートして先頭の n 件取得したい
<http://sisidovski.hatenablog.com/entry/2014/05/20/020529>
- 【Python 入門】 split 関数で文字列の分割の仕方 -Qiita
<https://qiita.com/Morio/items/38701038ad098dd5dc3d>