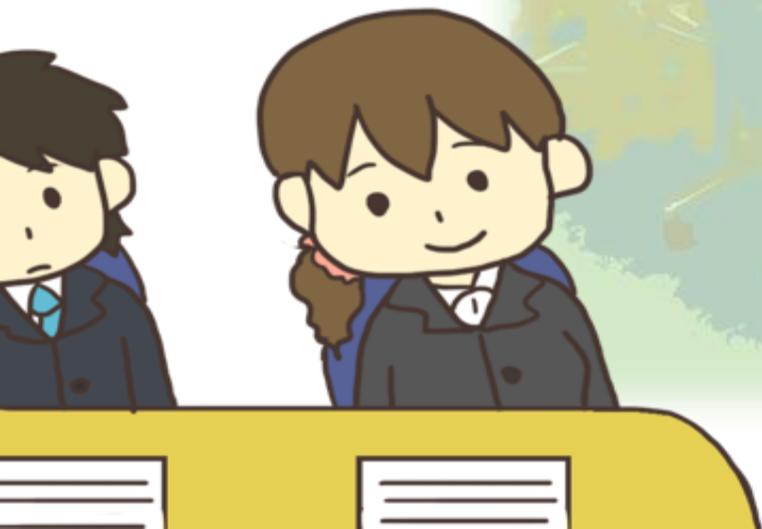


# Postkey

—会議室らくらくシェアシステム—



自由部門  
登録番号 20024



# はじめに

会議室や講義室を使いたい…。  
しかし部屋の利用に関して以下の問題があります。

部屋の利用者



部屋の管理者



鍵を取りに行くのがめんどくさい！  
(手間の問題)

会議室の予約は早い者勝ちだから、使いたい部屋が使えない  
(優先度の問題)

空いている部屋があるのはもったいない！  
(有効利用の問題)

複数の鍵を管理するのは紛失の可能性が増える！  
(安全性の問題)

問題解決には…

鍵のやり取りの**手間を省けて**、  
部屋の**有効利用**ができるシェアシステムが必要！！

そこで私達は！！

会議室らくらくシェアシステム

**Postkey** を提案します！！

# Postkey とは？

postKeyとは、会議室などの部屋のシェアを手助けするシステムです  
具体的にはスマホ1つで**部屋の予約**、**鍵の開閉**を行うことにより、  
部屋を借りるときの面倒なやり取りを省くことができます。

※スマホはスマートフォンを意味します

ターゲット

学生・教員など

管理者

利用者

## 1. 準備

Webで以下を登録

- ・アカウント
- ・鍵の解錠用デバイス  
(以下postKeyデバイス)

## 2. 申請

システムが優先度を考慮し  
自動的に部屋を割り振ります

## 3. 利用

利用者の履歴をWebで  
閲覧することができます

## 1. 準備

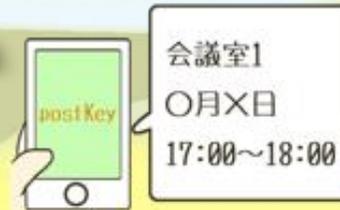
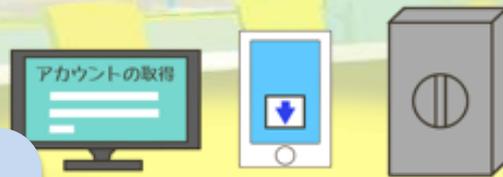
- ・スマホアプリのインストール
- ・管理者によるアカウントの発行

## 2. 申請

アプリから部屋利用の申請をします

## 3. 利用

申請が認められた場合、スマホを  
持って部屋の前に行くだけで部屋の  
鍵が開いて利用することができます



# 1.準備

The screenshot shows the 'postKey デバイス登録画面' (postKey Device Registration Page). On the left is a sidebar menu with options like 'デバイス一覧', 'デバイス追加', and 'クライアントユーザ発行'. The main area contains a form with fields for '部屋名' (Room Name), 'postKeyシリアル番号' (postKey Serial Number), 'ジャンル' (Genre), '収容人数' (Capacity), '開始時間' (Start Time), and '終了時間' (End Time). A green '作成' (Create) button is at the bottom.

管理者はまずアカウントを新規登録します。  
その後、以下を管理者が登録します。

## デバイス登録

- ・ postKeyデバイスのシリアル番号
- ・ 部屋名、ジャンル、収容人数、利用可能時間

## ユーザ発行

- ・ 利用者のユーザ名、メールアドレス、**ランク決め**

ランクには「教員」と「学生」があり、  
申請が被った場合、学生よりも教員の方が優先されます。



## あとは…

利用者はアプリをインストールしたら、  
postKeyからのメールに記載されたユーザ名と  
パスワードでログインをします。

これで準備はOK!

## 2.申請



### 申請方法

- ① **部屋ジャンル**を選択
  - ✓ 会議室
  - ✓ 講義室
  - ✓ 多目的室
- ② **利用人数**を選択
  - ✓ ~10人
  - ✓ ~20人
  - ✓ ~30人
  - ✓ 30人~
- ③ 部屋使用の**開始時間**を選択
  - ✓ ○月○日○曜日○時○分
- ④ 部屋使用の**終了時間**を選択
  - ✓ ○月○日○曜日○時○分
- ④ 真ん中の**申請ボタン**を押し、申請完了

前日の18時までに  
予約すること！

朝、登校・通勤時にアプリで申請  
の許可/不許可を通知！

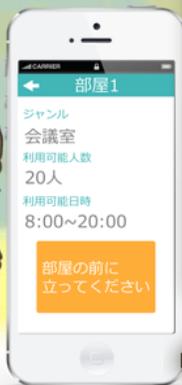
### 管理者は…

優先度を考慮してシステムが自動的に部屋を割り振るので、申請が来る度に許可/不許可の決定をする必要はありません。

## 3.利用



ガチャ



利用者がアプリを起動し  
**部屋の情報**を表示して  
**10m以内**に近づくと  
解錠できます。  
解錠は**10秒間**だけ有効で、  
**10秒経過後は自動的に**  
施錠されます。

### 管理者は…

部屋ごとに利用した人や  
時間のログが見られます。

会議室1  
○月×日：Aさん  
△月□日：Bさん

# 解決方法

同じランクの利用者が重複した場合  
ポイント制により優先度を決定し、  
部屋を割り振ります。  
優先されなかった利用者には、  
同じジャンルの別の空き部屋を  
かわりに提案します。



**優先度の問題  
有効利用の問題**  
を解決

スマホを部屋の前に  
持って行くだけで鍵が開き、  
部屋の利用が可能です。



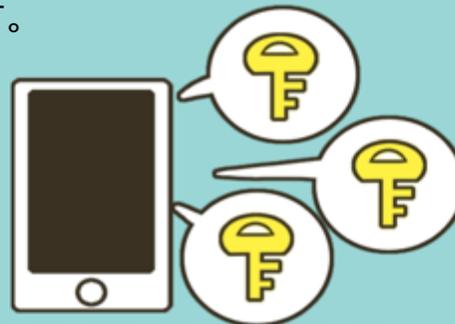
**手間の問題**  
を解決

もしスマホを紛失してしまっても、  
アプリのアカウントで鍵の開閉を  
行うので、ログインされなければ  
第三者に部屋利用の権限を乱用さ  
れる心配はありません。



**安全性の問題**  
を解決

複数の部屋を利用する場合でも、  
スマホを1つ持っていればどの  
部屋の鍵も開けることができます。  
鍵をいくつも持つ必要がなくなり  
ます。



# 部屋の決定

## ポイント制の導入

もし同じランクでユーザ同士の申請が重なってしまった場合、ポイントが高いユーザほど優先度があがります！

### マイナスポイント

#### \* 部屋不使用時のペナルティ

予約をしたのに部屋を使わなかった場合  
ポイントがマイナスされます

#### \* 累計時間

累計使用時間が多いとポイントがマイナスされます

#### \* 使用回数

使用回数が多いとポイントがマイナスされます



使用時間、回数が少ない人を  
優先することで皆が  
**部屋を公平に**利用できます！！



### プラスポイント

#### \* ログインボーナス

1日の初回ログイン時にポイントが加算されます

#### \* ゴールデンタイム制

みんなが使わない時間に部屋を利用すると  
ポイントが加算されます

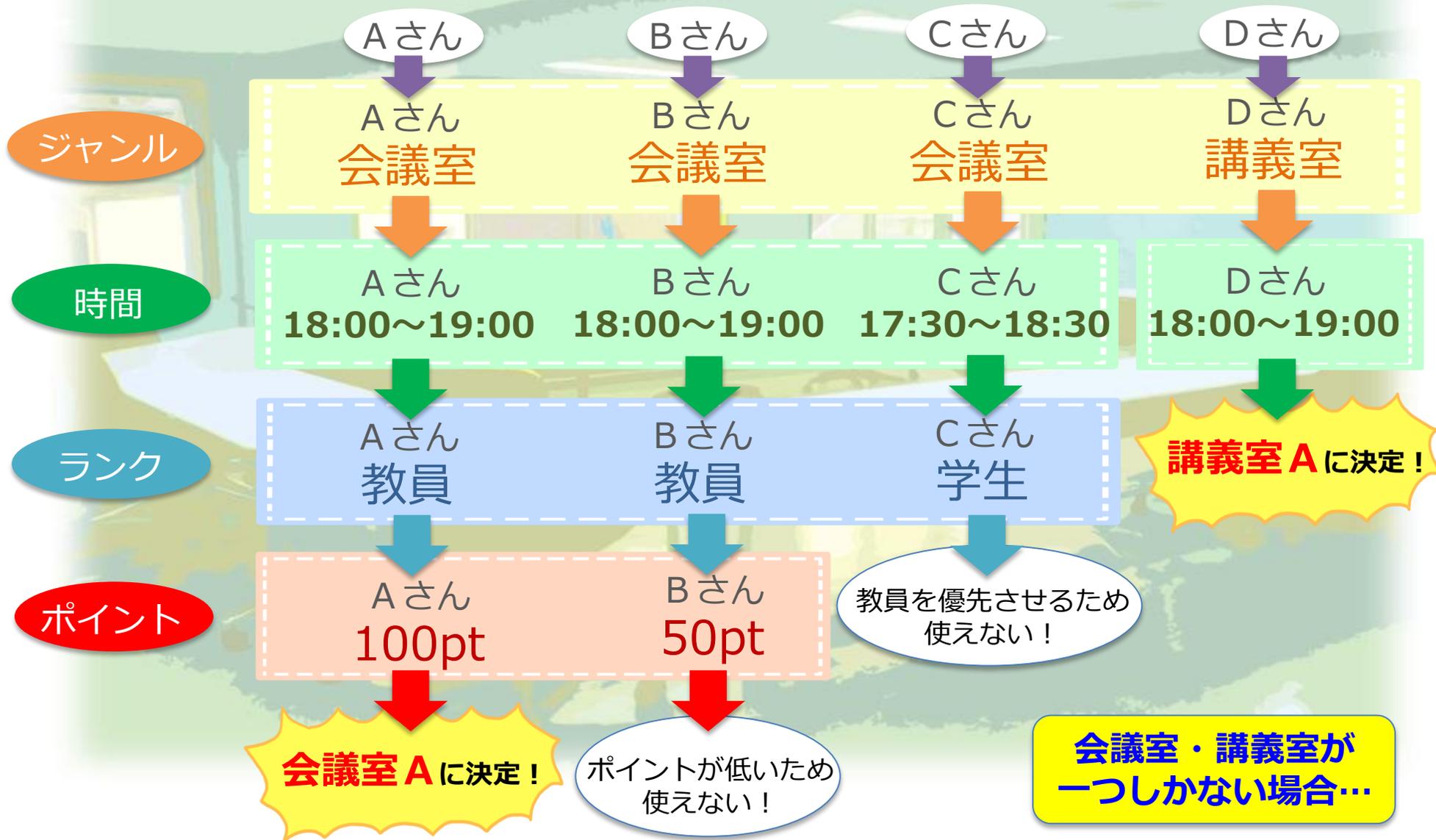


**次回自分の希望時間に  
使いやすくなります！**



# 実現の可能性 優先度の決定

4人の利用申請があった場合の部屋割り当ての手順

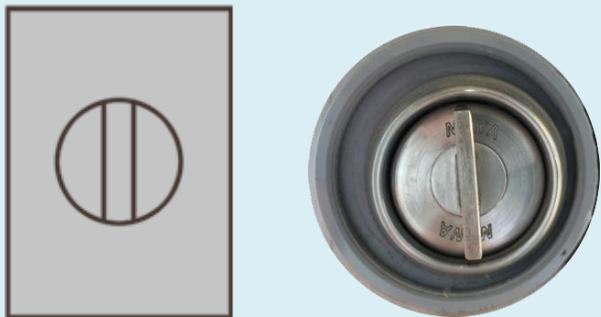


# 鍵開閉の実現方法

postKeyデバイスが対応する  
鍵の種類は…

「シリンダー付サムターン」

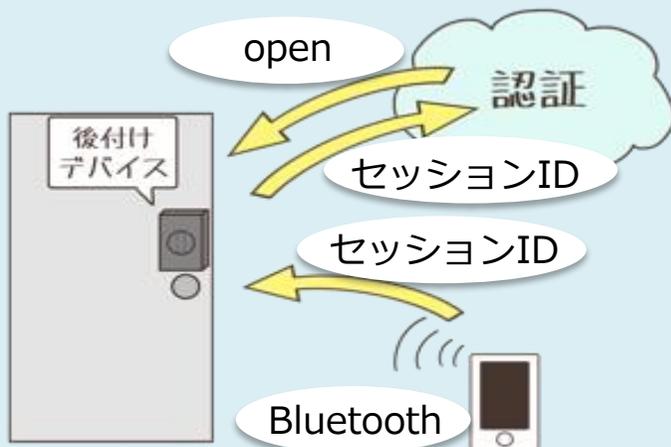
1. postKeyデバイスを設置します。



2. ドアの鍵に取り付けられたpostKeyデバイスはサムターンと連結して、鍵の操作を行います。



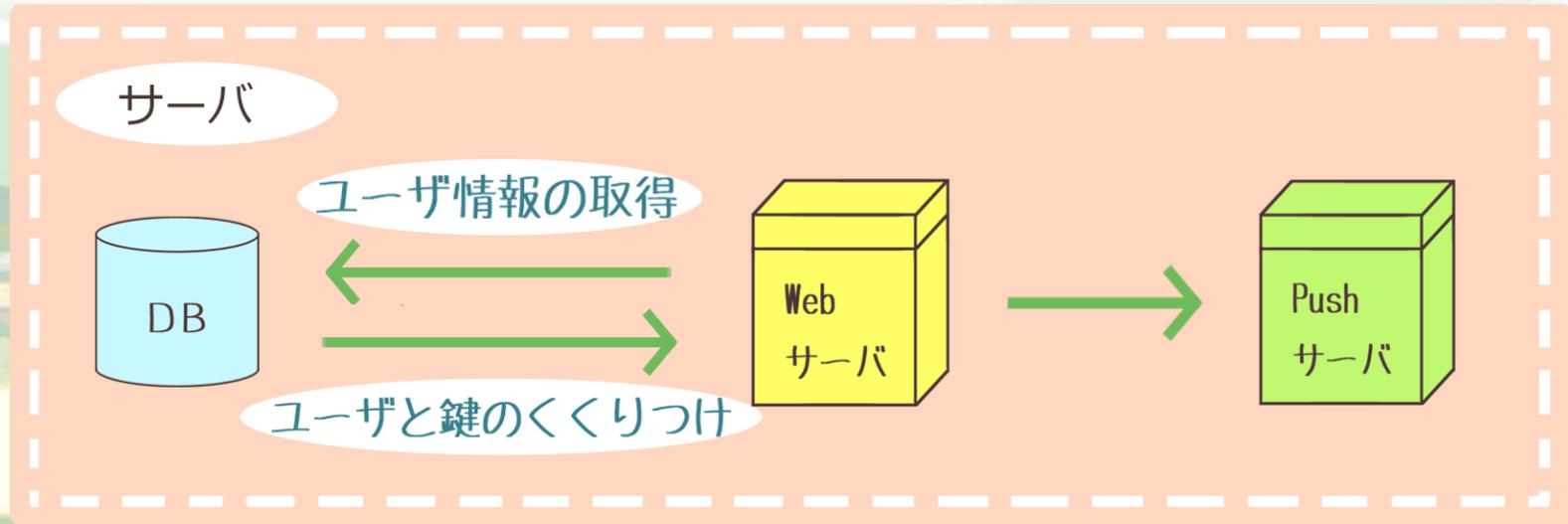
3. 申請が承認された部屋の利用者がスマホを持って部屋に近づくと、スマホとpostKeyデバイスが通信し、サーバを介して部屋利用の認証を行います。



4. 認証がなされると、postKeyデバイスによって部屋の鍵が開きます。



# システム構成



# postKeyの独創的な部分

本システムでは、ユーザを**アカウント**で管理します。  
アカウントの情報を参照して、**部屋の利用権限の優先度が高い順に部屋を割り振ることが可能**となります。  
また、部屋を開けるためには利用申請を出す必要があり、**申請が承認された利用者**にのみ**申請した希望時間だけ部屋を開ける権限**が与えられます。これにより、カードキーシステムのようにいつでもだれでも部屋に入ることはできません。

## 類似品との比較

|  | 部屋の申請 | 鍵との連携 | 鍵が第三者に漏れない   | 時間による入室管理    | 部屋利用の優先管理 |
|--|-------|-------|--------------|--------------|-----------|
| ChoiceRESERVE<br>予約システム  | ○     | ×     | —<br>鍵との連携なし | ○            | ×         |
| カードキー<br>部屋管理<br>システム  | ×     | ○     | ×            | △<br>繰り返しは可能 | ×         |
| <br>本システム | ○     | ○     | ○            | ○            | ○         |

# 開発環境

## サーバ

php,HTML,  
javascript  
Windows  
Azure(Ubuntu  
14.\* LTS)

## アプリ

objective-c  
Xcode( OS X )  
iOS 7  
iPhone5s

## デバイス

arduino IDE  
arduino  
その他 arduino通信  
モジュールなど  
Bluetooth LE

# 開発日程

|              | 5月 | 6月 | 7月 | 8月 | 9月 | 10月 |
|--------------|----|----|----|----|----|-----|
| 設計           | ←→ |    |    |    |    |     |
| サーバ          | ←→ |    |    |    |    |     |
| アプリ          |    | ←→ |    |    |    |     |
| デバイス         |    | ←→ |    |    |    |     |
| バグの修正<br>テスト |    |    |    |    | ←→ |     |