

VANTI^Q

混雑検出アプリケーション開発

課題概要

- これまでに学習した内容を使いながら本資料を参考に混雑検出アプリケーションの実装を行ってください。
- 詳細な手順はありません。アプリケーションの実装に必要なTypeやSourceなどリソースは全てご自身で追加してください。
- 応用的な課題であり、これまでに使っていないActivity PatternやVAILでのプログラミングが必要になりますのでリファレンスを調べながら取り組んでください。
 - App Builderリファレンス
 - <https://dev.vantiq.co.jp/docs/system/api/index.html>
 - VAILリファレンス
 - <https://dev.vantiq.co.jp/docs/system/rules/index.html>

本アプリケーションのシナリオ

1. ビルのそれぞれのフロアの混雑を、フロアごとのデバイス数を元に検出する
2. 各フロアにWiFi APが5台ずつ存在し、それらがフロア内のデバイスを検出する
3. WiFi APは検出した全デバイスの情報（配列データ）と自身のMACアドレスをPOSTする
4. VANTI QはPOSTされたデータをもとに混雑と、ウォッチリストに登録されたデバイスの検出を行う

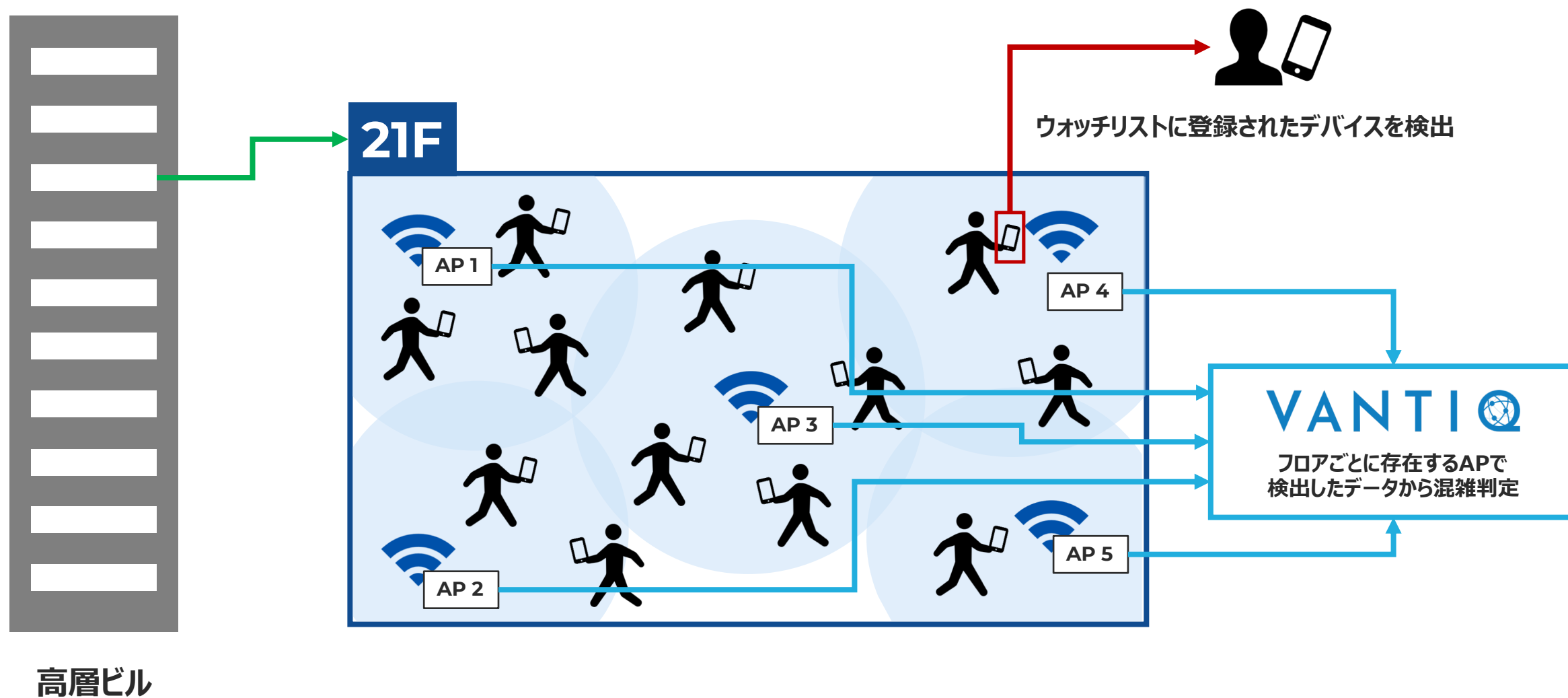
※ウォッチリストは、PCR検査で陽性だった、2週間以内に発熱があったなど、ビルに来ては行けない人のデバイスが登録されているリストを指します

※ 2と3を再現するためのデータジェネレータが配布されます

実装するアプリケーションの機能

1. フロアごとに混雑率を計算し混雑を検出する
2. ウォッチリストに登録されたデバイスがフロア内に侵入したことを検出する
3. 検出された場合、通知・他システムへの連携を行う

1フロアあたりの動作イメージ



VANTIQで実装する処理の流れ（概要）

1. WiFi AP（データジェネレータ）からPOSTされるデータを受信する
2. データをPOSTしたWiFi APのMACアドレスに合わせてフロア情報を付与する
3. 検出されたデバイスの配列データをデバイス単位に分割し処理を並列化する
4. 不要なパラメータを削除し、データを最適化する
5. デバイスがウォッチリストに存在しているか判定する
6. ウォッチリストに存在していたら任意のサービスに通知する
7. デバイスごとのデータをフロア単位のストリームに分割する
8. 直近3分間のデバイス数をフロアごとにカウントし、フロアごとの許容人数に合わせて混雑率を計算する
9. 混雑率50%を超えたら混雑と判定する
10. 混雑が発生したことを任意のサービスに通知する

実装のヒント

- Activity Pattern
 - Filter
 - 単純なフィルタリングから、閾値にProcedureを設定することで柔軟に判定を行える
 - AccumulateState
 - 次々に受信するイベントをトラッキングしてカウントをしたりできるため、Procedureの書き方次第で一定期間に受信したイベント数を使った計算など自由度高く実装することができる
 - Statistics
 - イベントの統計処理が行える
 - SplitByGroup
 - グループごとにストリームを分割できる

※全てのActivity Patternを使用しなければならないわけではありません