# CampusM User Group 2021

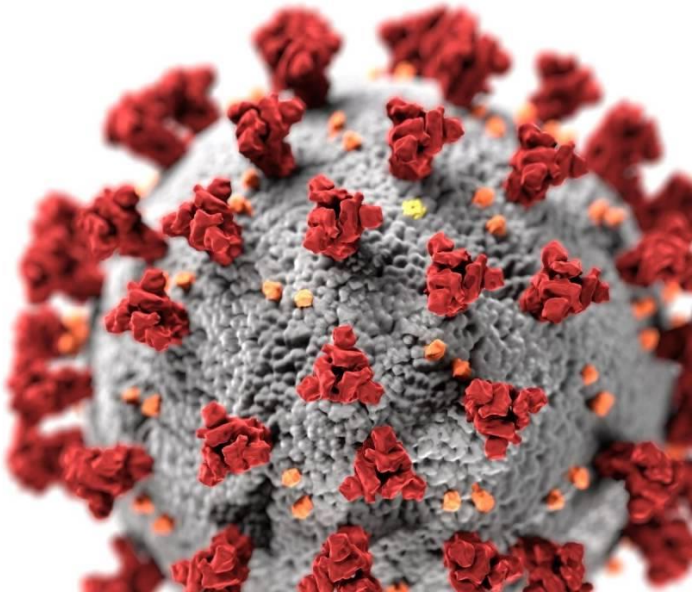
## Developing a Capacity Management Counter

Nishen Naidoo, Senior Applications Specialist
**Macquarie University**

# Introduction

## Background

- Jan 2020
  - Covid hits Australia
- Mar 2020
  - University shuts down – operates remotely
- Sep 2020
  - University opens with restrictions in place
  - Library is one of the first buildings to open with restrictions



# Restrictions

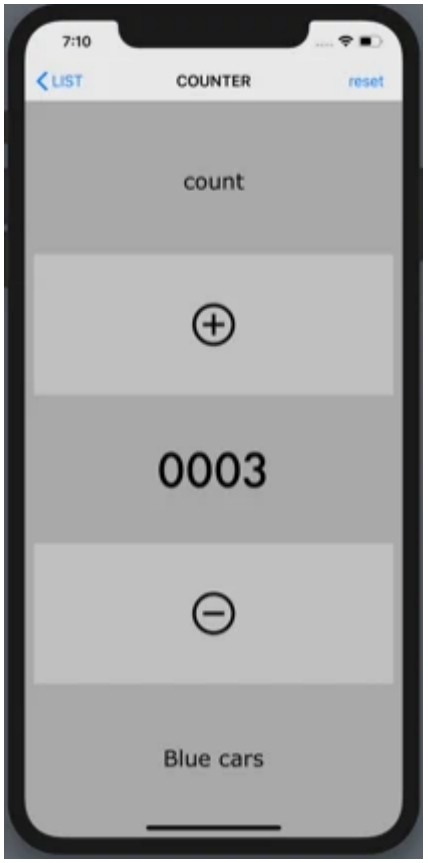
## Principles

- Restrict access to current staff and students only
- Social distancing guidelines
- Patrons tap and enter the library with their access cards
- Access is recorded for COVID-19 tracing purposes via the tap
- No tap to exit capability so unable to use tap to manage capacity

# Capacity Management

## Round One

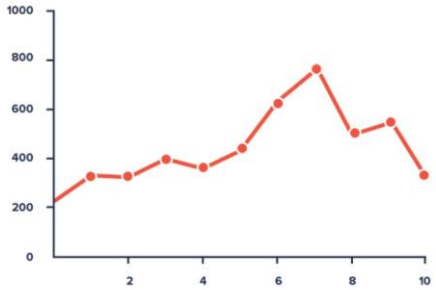
- Limit entry/exit to one door with tap sensor to manage traffic
- Security presence to ensure tap compliance and no tailgating
- Security provided with ‘off the shelf’ app to record entries and exits
- App has data export capabilities



# Capacity Management

## Data Management

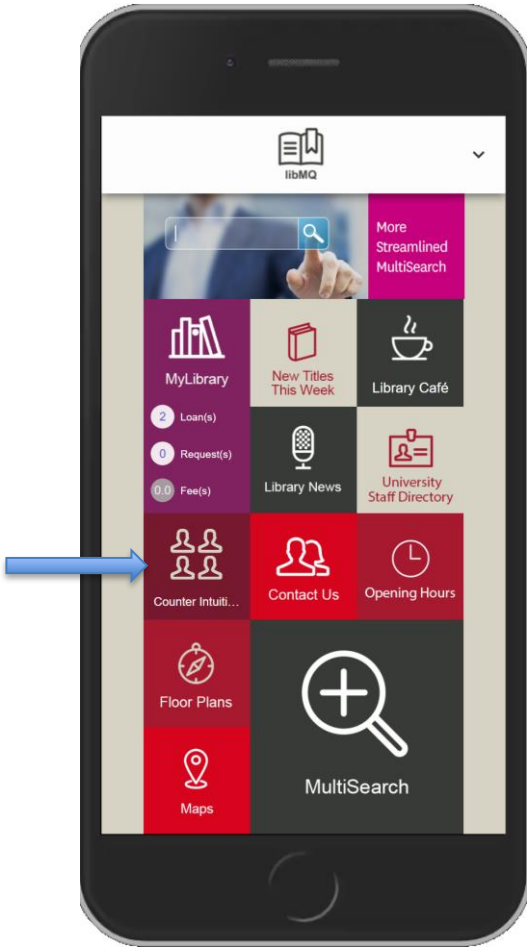
- Staff to export data each evening to Dropbox (app feature)
- Data issues arising:
  - Switching phones for battery reasons
  - Security staff unaware of process
  - Data exports being duplicated
  - Data exports missing
- Data obtained from Dropbox exports but staff have to massage the data and extract information
- Extrapolate missing data, deduplicate data – sometimes not possible
- Days before getting any actionable information




# Capacity Management

Surely there is an easier way!?

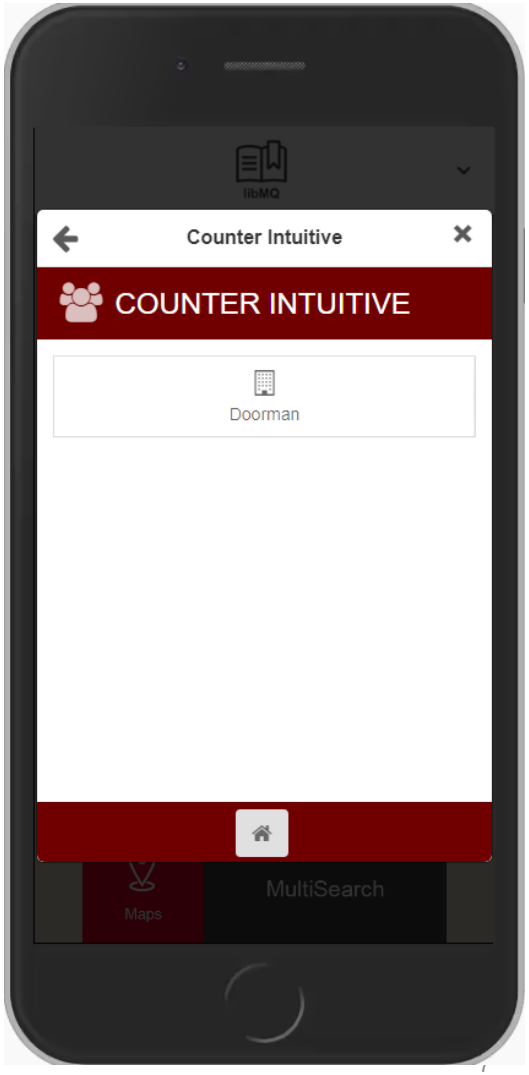
- Develop our own app?
  - Organisational processes are intensive and time consuming – probably not feasible
- What about CampusM? Could we leverage that infrastructure?
  - Already rolled out
  - Have development experience
  - Let's see what we can do…



# Capacity Management

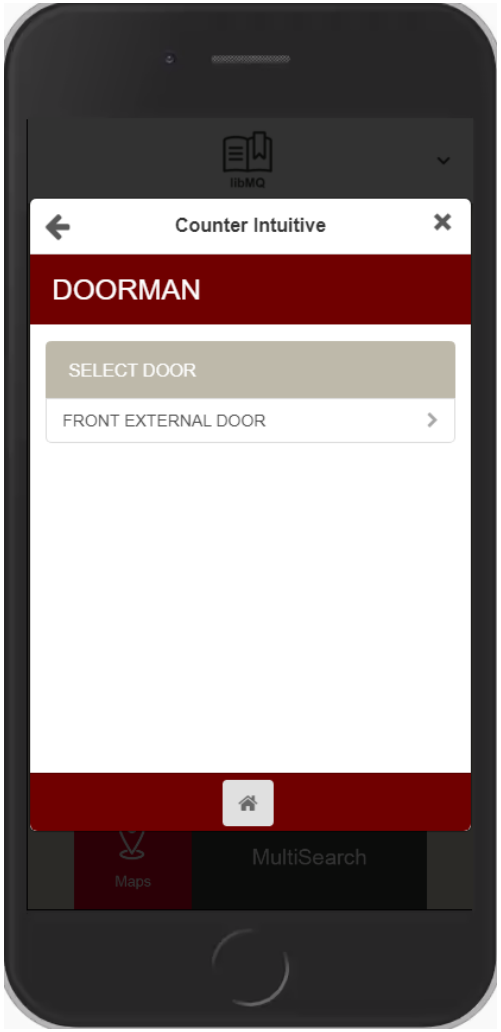
## Round Two – Counter Intuitive

- Developed an intuitive counter app on the CampusM platform using the campusM AEK - Counter Intuitive
- Initially thought about having a door counter option as well as a head count option
- Door count option - Doorman

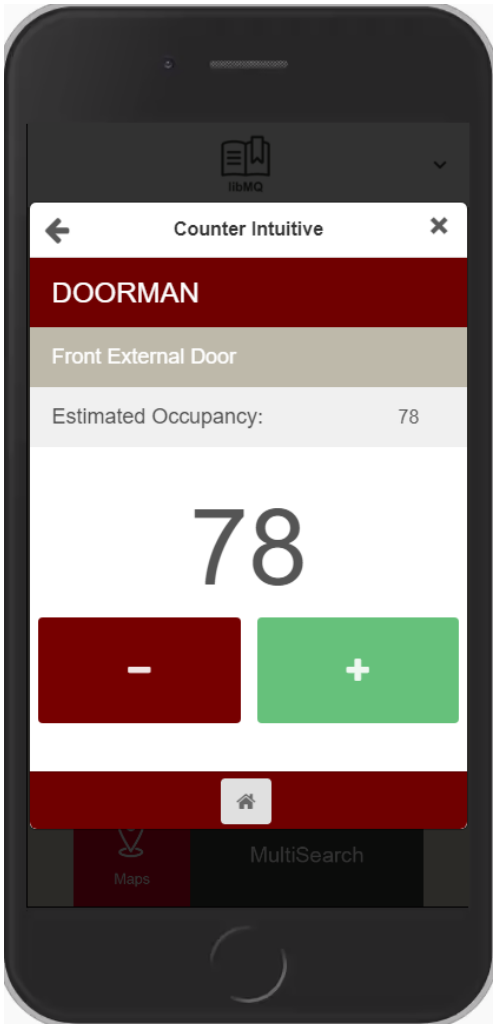


# Capacity Management

## Round Two – Multi Door

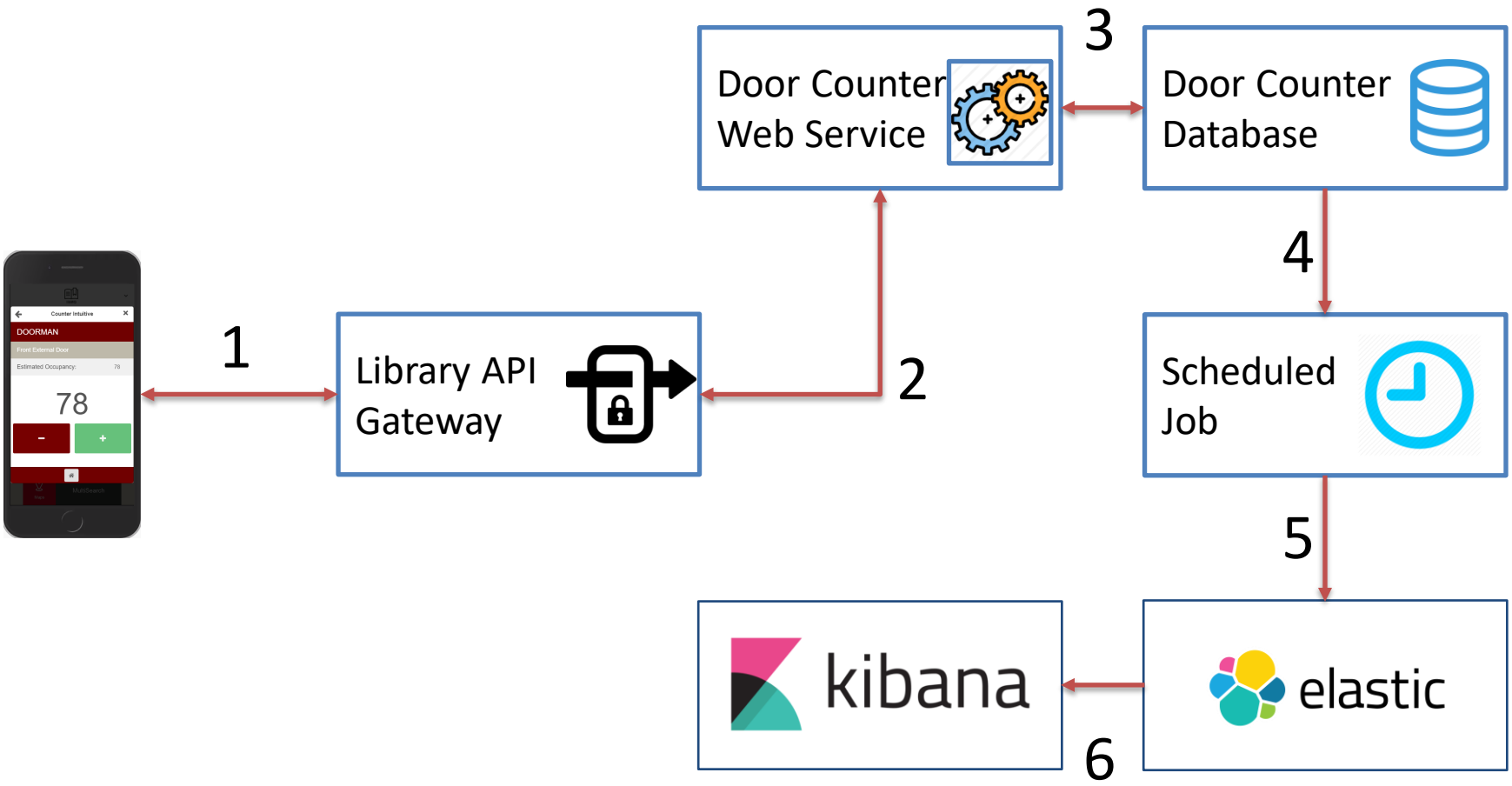



- Initially single door
- More doors in the future?
- Simple UI
- Application visible to Library Security group only
- Just under two weeks for solution



# Capacity Management

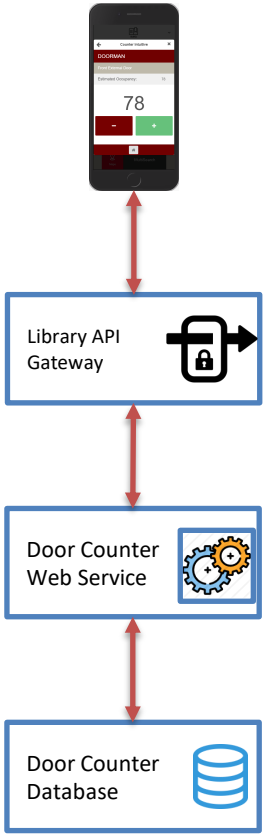
## How it works



# Capacity Management

## How it works – Saving the data

1. Every click on either + or – generates an event
2. Every event stored on device via AEK Storage library immediately
3. Every 20 seconds
   i. App checks local storage for events
   ii. If events found, then it packages them up and sends them to the Door Counter Service via the API Gateway
   iii. If successful, it clears the events that it sent (additional events could have been generated while it was sending, these are untouched and will be synced in the next cycle)
   iv. Door Counter Service saves the data in the Oracle Database backend
   v. Returns the latest combined count of all doors for the 'Estimated Occupancy'

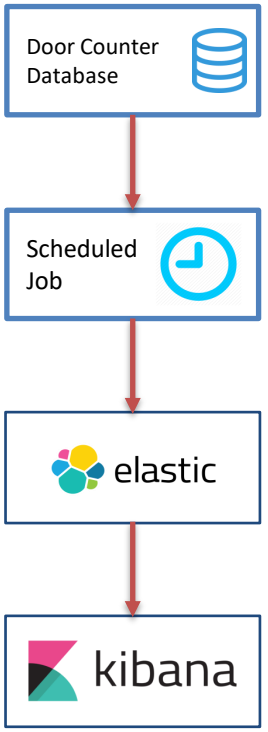



```
Sample event (JSON):
{
  "uuid": "8dc28b76-98a1-4630-9c3c-b7ad4174f3fb",
  "event_time: "2021-03-02T09:02:17+10:00",
  "change": 1,
  "user_id": "mq12345678",
  "door_id": 1
}
```

# Capacity Management

## How it works – Viewing the data

1. Scheduled Job runs every 5 minutes (configurable)
   i. Reads data from Database
   ii. Writes data to Elastic (UUID is ID – can't duplicate records)
2. Use Kibana to visualise the data and create realtime(ish) dashboards

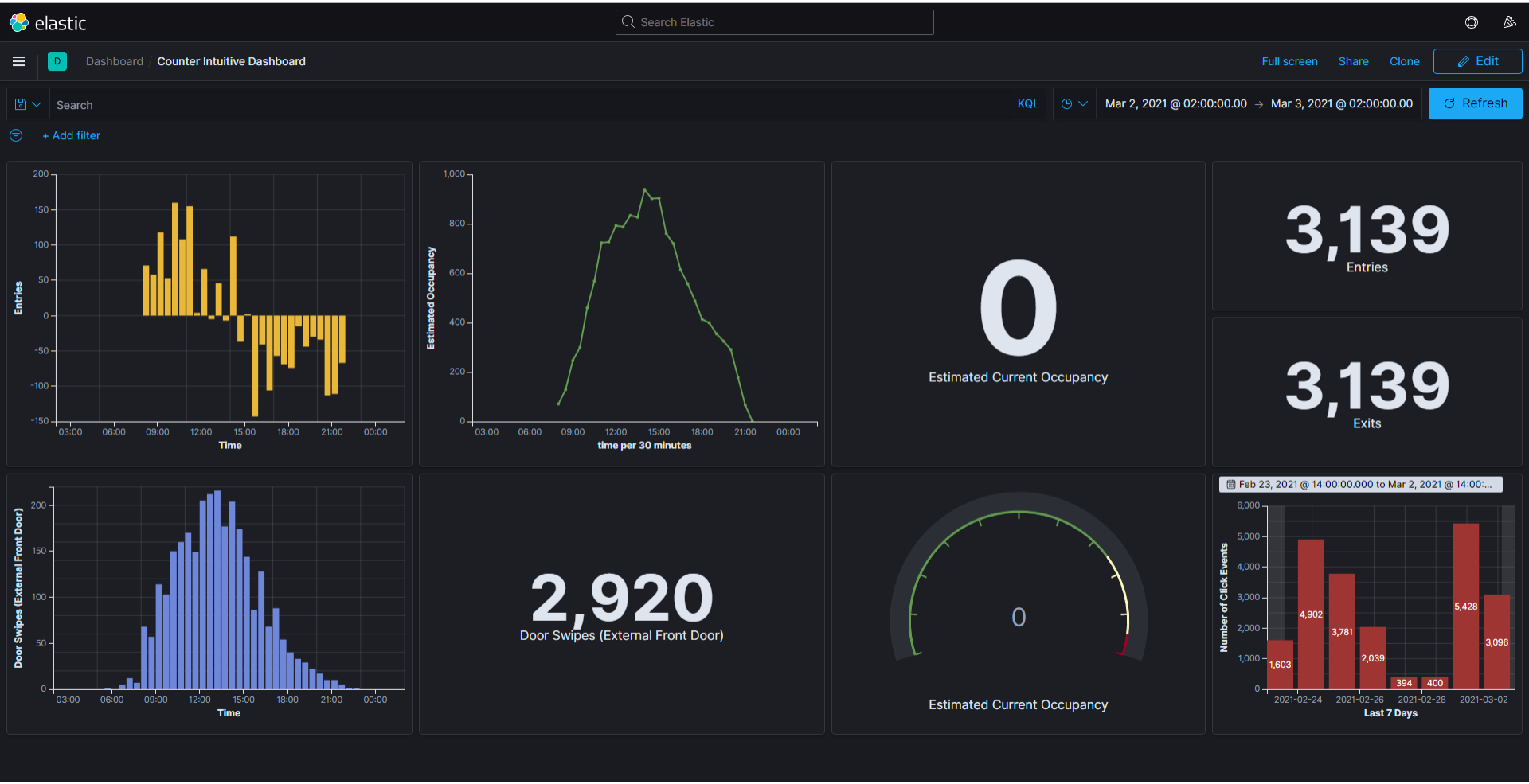


# Capacity Management

## How it works – End Result

# Capacity Management

## How it works – End Result



# The End

MACQUARIE University

Questions?

# Appendix A

Technology Stack

MACQUARIE University

| Component | Technology |
|---|---|
| Phone App | CampusM Platform + AEK |
| Library API Gateway | Kong API Gateway deployed on a Kubernetes Cluster |
| Door Counter Web Service | Python App:<br>- SQLAlchemy<br>- FalconAPI<br>- Gunicorn<br>Packaged as a Docker Container<br>Stored on GitHub Container Registry<br>Deployed on a Kubernetes Cluster |
| Door Counter Database | Hosted Oracle Database |
| Elasticsearch Cluster | Elasticsearch deployed on a Kubernetes Cluster |
| Kibana | Kibana Application deployed on a Kubernetes Cluster |