Mint Vision Application Architecture

30 January 2020

Created by Peter Reid

1 Introduction

This document outlines the Architecture of the Mint II Vision Product

The Mint II Vision Product combines a Universal Windows application with Azure Cognitive computing, storage and processing to leverage the best of both worlds. The full architecture of the system is described below.

All the components built below are Mint's Intellectual Property. For components that reside in Azure, such as the CosmosDB and Cognitive Services, these are Microsoft's Services but the manner in which our code consumes these services is our Intellectual Property.

2 Current Architecture

The diagram below represents the current architecture of the application.

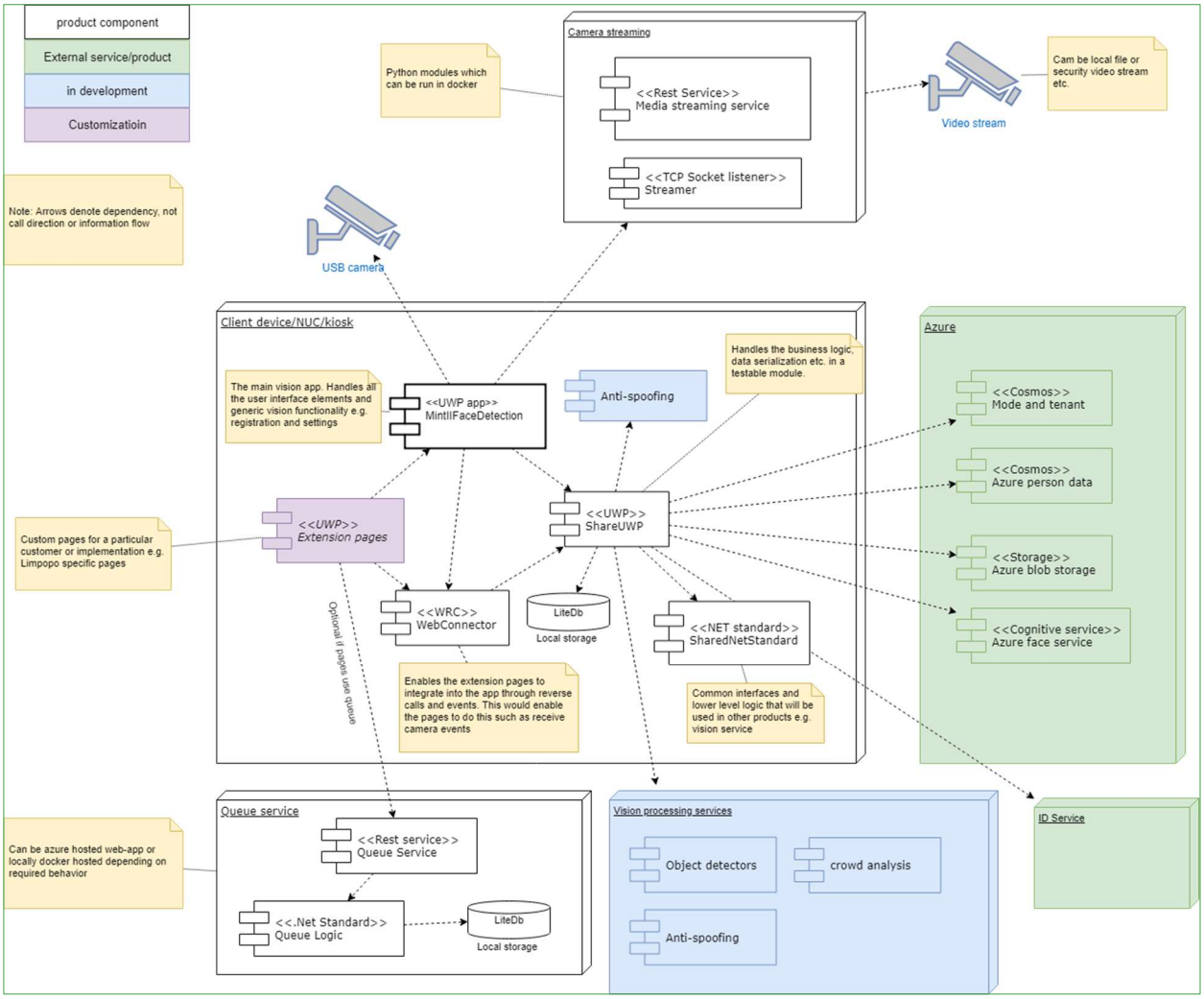


Figure 1: Mint II Vision Application Architecture

The individual components are described below.

2.1 Client Device

The Client Device is a Universal Windows Application. This application runs in Kiosk Mode on any Windows 10 Device, including embedded devices such as Raspberry Pi's.

The application has several components that are noteworthy:

- The application can run in online or offline mode
- The application can bind to streaming or usb cameras
- The application has a local storage solution that is then synchronized to the cloud
- The application can embed custom pages such as web pages, which are used to extend the application into several different use-cases

2.2 Azure

The application relies on several technology layers in Azure. Each layer is indicated below.

2.2.1 Storage

2.2.1.1 Images

Images that are captured of each face event are stored as images in Azure. This means that minutes after a capture event, the images and audit entries are stored in the cloud and retrievable from anyhere in the world.

2.2.1.2 Data

Data for the application, such as users, audit trails and settings, are stored in Azure CosmosDB. They are stored as documents and synchronized regularly to the devices themselves.

Cognitive Services

The application uses Microsoft Cognitive Services for

- Face Detection
- Face Recognition
- Similar Face Matching

2.2.1.3 IoT Hub

The IoT Hub is used to co-ordinate messages to and from the devices. For example, messages can be sent to each device to force an upgrade to a newer version of the application. We also send a “HeartBeat” message to the IoT Hub to facilitate detection of when a device is down.

2.2.1.4 Cognitive Services

The Vision Application uses Cognitive Services to do face recognition actions. In addition, the OCR service is used for processing Drivers’ licenses and ID documents.

2.3 Camera Streaming Service

In order to consume the stream from network cameras, a separate streaming server is deployed with the application. This is a service that is deployed using containers either on-premises, independently in Azure or within a Kubernetes cluster. The service consumes streaming camera in a multitude of formats, applies any necessary image correction, and then publishes the stream to the application in order to be consumed by the system.

2.4 Vision Processing Service

The Vision Processing Service is a REST-based, container-distributed service that performs Image Processing services such as Object Detection, Object Recognition, and similar machine-learning-based services. This processing engine is external to the application so that powerful GPU-based servers can process the images.

2.5 Queue Service

Hosted external to the application (often as a service in Azure) is the Queue Management service. The Queue Management Service allows for people queueing, such as at a branch at a bank or other such use cases.

2.6 Extension Services

The application is designed to be extended to several different use-cases. The extension part of the application combines two pluggable items:

- Custom Pages. For example, pages that serve content for touchless purchasing in a canteen differ vastly to pages that dispense medication
- Custom integration. For example in a university setting the application connects to the Student Database to synchronize student details
- Custom Triggers. When using the application for example as an Access Control mechanism, the triggers open or close doors. When in a canteen, the triggers can cause vending machines to dispense, etc.

3 Roadmap Architecture

The below architecture represents the roadmap architecture for the application and platform. It is important to note that, at any particular time, the actual architecture will be sometime between the as-is and roadmap architectures as new features and components are being added every sprint.

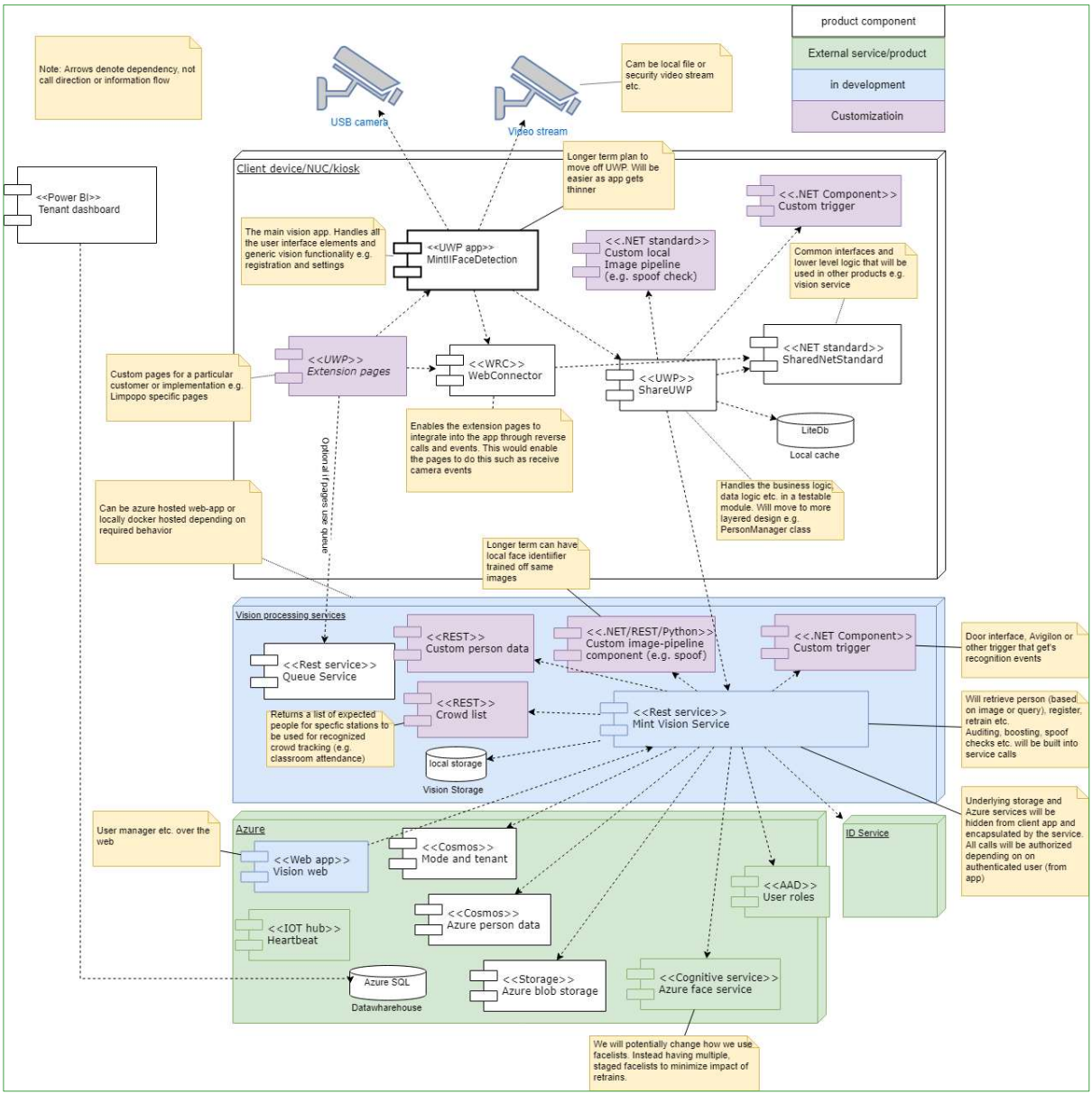


Figure 2: Roadmap Architecture

The new sections are described below.

3.1 Edge Device

3.1.1.1 UWP Extension Pages

UWP Extension Pages are embedded in the application, and represent Single Page Applications that can be loaded into he application once a face is recognised.

3.1.1.2 Custom Trigger Component

When triggering hardware, such as doors or coffee machines, each trigger is slightly different. The externalised trigger mechanism allows for any trigger to listen to events on the application and trigger them as and when they occur.

3.1.1.3 Custom Local Image Pipeline

The application uses the execution of ONNX models to perform custom vision processing. The custom local image pipeline is a pluggable pipeline that uses the image processing engine to detect and classify objects and sequences.

3.1.1.4 Web Connector

The web connector serves as a bridge between the face dectection routines and the Custom Extension Pages.

3.2 Service

3.2.1.1 Custom Person Data

Each person stored in the application can have extension fields, such as “Student Number” or “Patient Number”, stored against their profile. These fields can be stored either as extension fields on the application themselves, or fetched from an integration layer.

3.2.1.2 Crowd List

In Crowd Tracking scenarios, the crowd list allows for querying of current and new faces, and other settings such as scan time and persisted face time.

3.2.1.3 Boost List

The application uses “Boost Lists”, which takes advantage of known-audience heuristics to boost the confidence of certain faces. These lists are maintained and called from the Boost List service.

3.2.1.4 Custom image Pipeline

When processing images for purposes such as Object Detection or sequence evaluation, the pipeline allows for injection of models that perform this analysis.

3.2.1.5 Vision Web

Some administrative features, such as user management, are exposed through the Vision Web API.

3.3 Queue Service

The Queue Service performs queueing of people, objects (for example on a conveyor belt, or medicine) and exposes this as an API to the end-user.

3.4 Azure

We use the following services within Azure:

- Cosmos
- Azure Blob
- Azure SQL
- AAD
- Cognitive Services
- IoT Hub
- Azure Functions

3.5 External Services

3.5.1.1 ID Service

We validate South African ID's through a third-party service. Although this is implemented in a pluggable architecture, at present there is a single provider that we use.

3.5.1.2 Reporting

All reporting in the system is implemented through PowerBI, using our Cosmos and SQL databases as a data source.

4 User Flow

The following flows describe how the application is used in an example application. Note that this is a sample use-case; the application is highly extensible and so can be used in many different ways besides patient registration.

4.1.1 User Enrolment

When a new user arrives at the system and is not recognized, the system prompts the patient to register.

4.1.1.1 Image Capture

The system takes several pictures of the user for enrolment

4.1.1.2 ID Card

The user holds up their ID Card, and the system extracts their ID Number, first name and last name. They are also optionally prompted to enter their cell-phone number, so that they can receive SMS notifications of their progress.

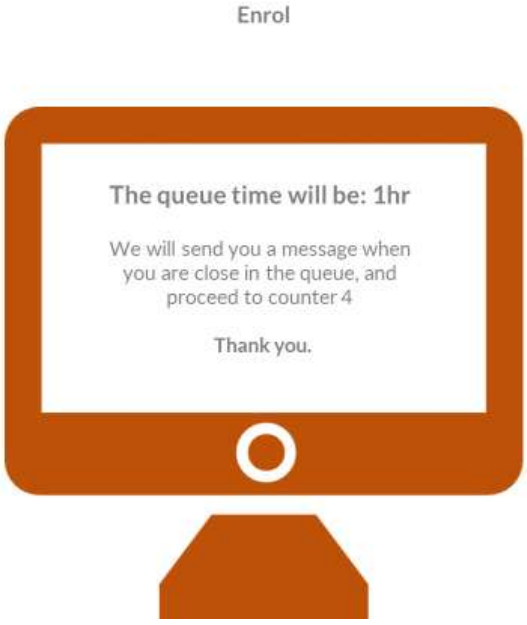
4.1.1.3 Prompts for the Queue (if Queue Management is enabled)

When the user first registers, they are asked what the purpose of their visit is. This is so that they can be directed to the correct queue.

4.1.2 User Greeting

If the user is recognized on arrival, they are asked to confirm their presence and the reason they are at the facility.

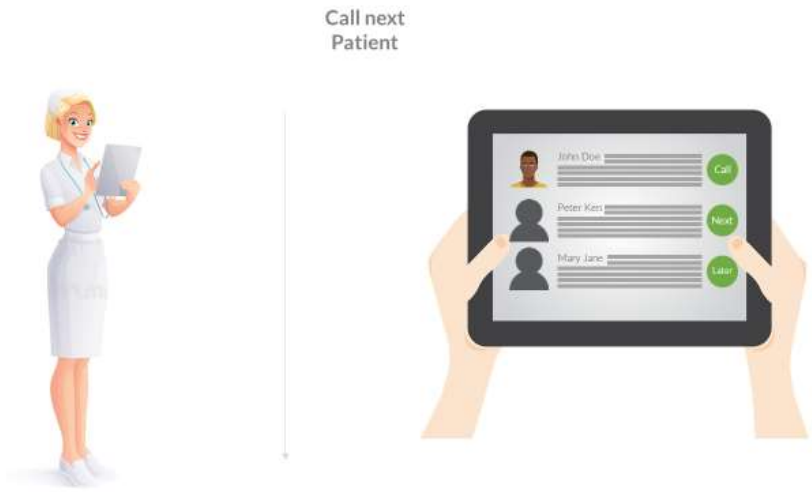
4.1.2.1 Queue Prompts



The system prompts them regarding the queue time, and optionally notifies them that they will be sent an SMS when they are close to the front of the queue

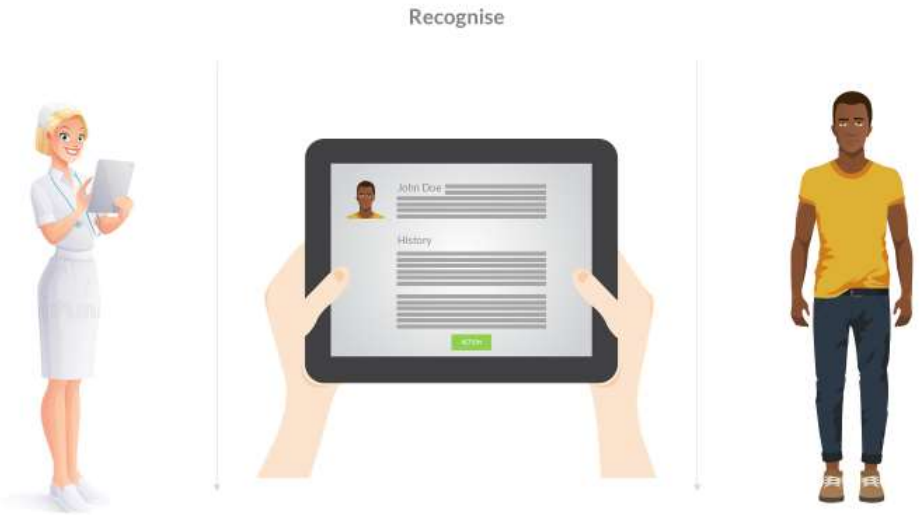
4.1.3 At Station Greeting

4.1.3.1 At station



At each station, the station manager is shown a list of the current queue, with pictures of each user, and can call the next user in the queue. In this example a clinician is seeing a list of patients and is calling the next patient.

4.1.3.2 Recognition



As the user walks up to the station, they are recognized by the cameras and their history is displayed to the station manager. This will include the history of a user that has visited multiple locations (where this system is installed) over the last few days or weeks.

4.1.3.3 Capture of actions

The station manager performs whichever action is necessary for the use rand captures a free-text description of what they did in the patient history.

5 Registration Process (Screenshots)

When a user must be registered on the system, the following process is followed to register them.

5.1 Recognition

As per the normal flow, the user stands in front of the camera. The camera will attempt to recognise the individual.

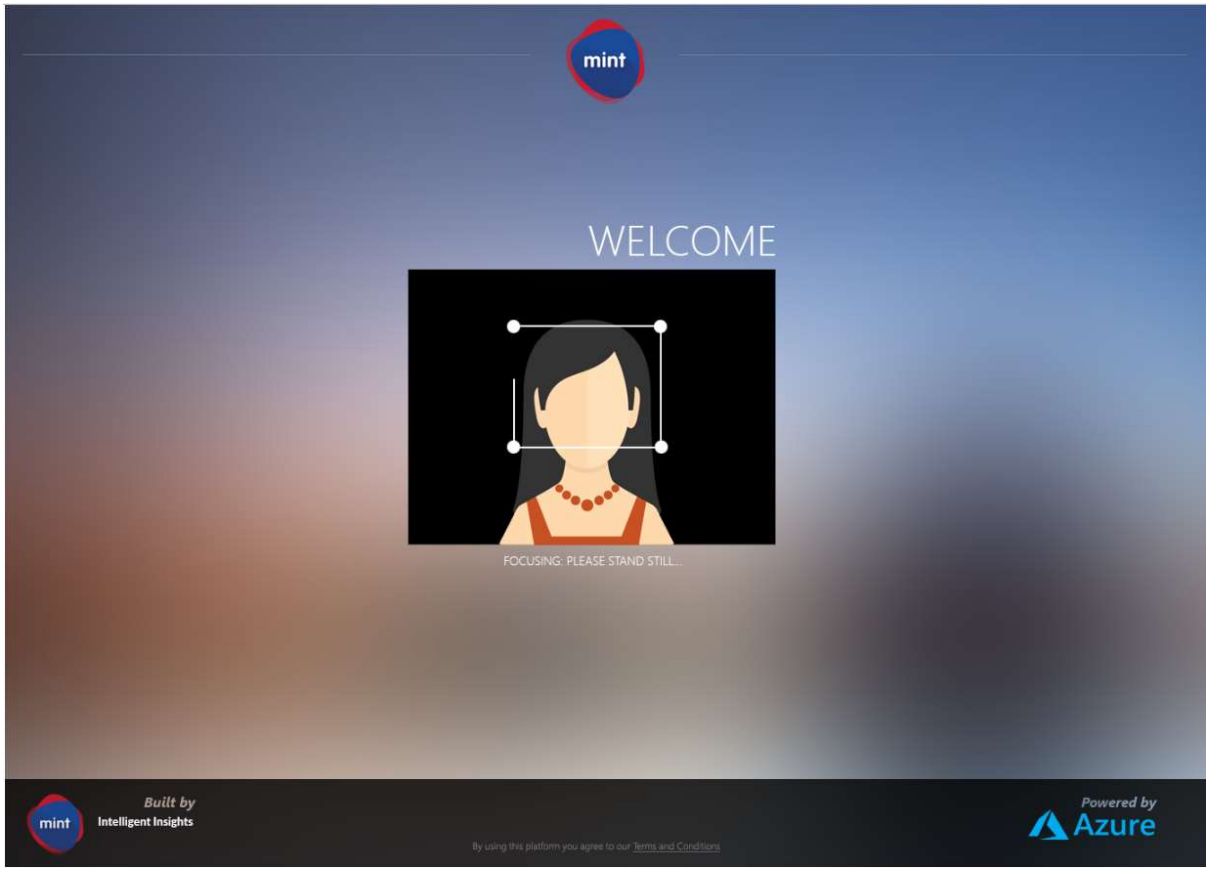


Figure 3: Capturing The Image

5.2 Prompt for Recognition

If the user is not recognised, they are prompted to register on the system.

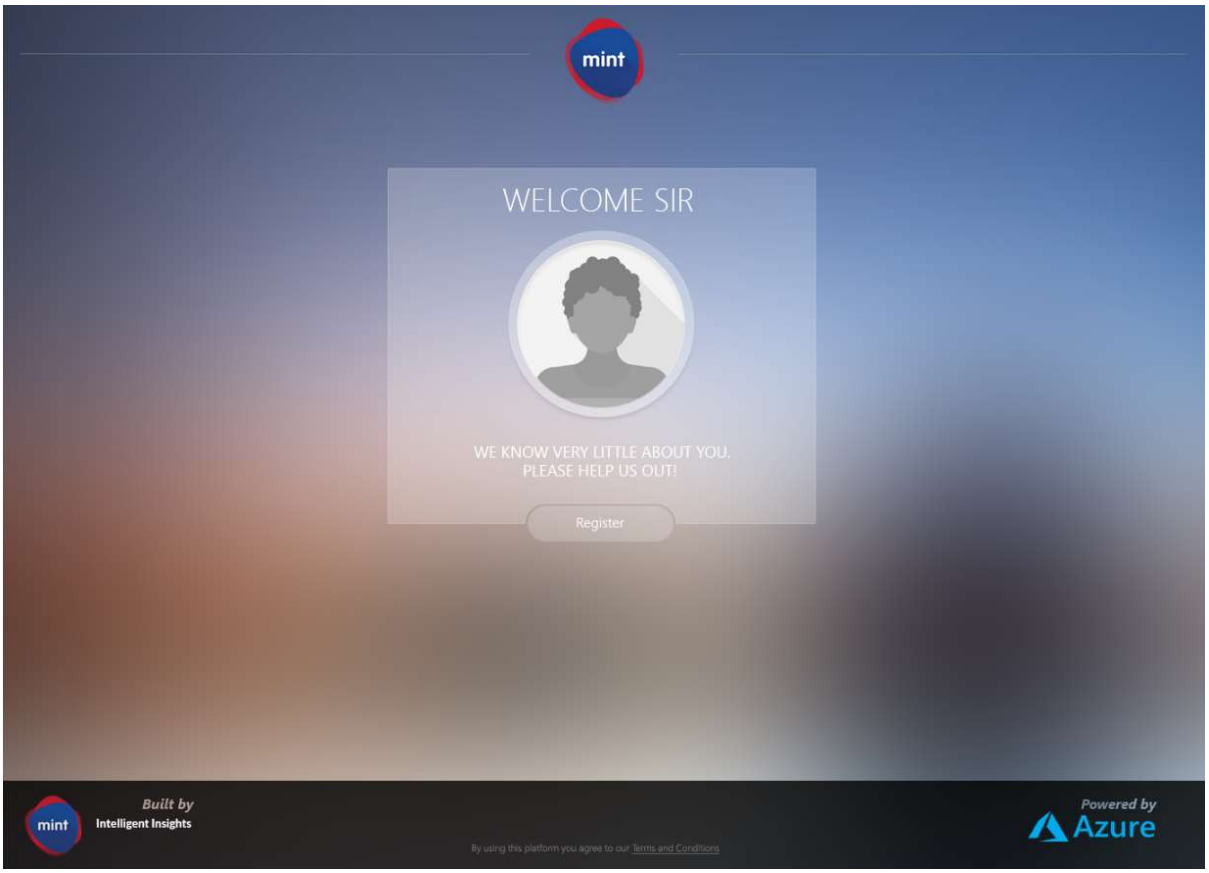


Figure 4: Prompt for registration

5.3 Scan Drivers or ID

If the system is configured to allow it, the user is prompted to scan their drivers license, ID Card or ID Document.

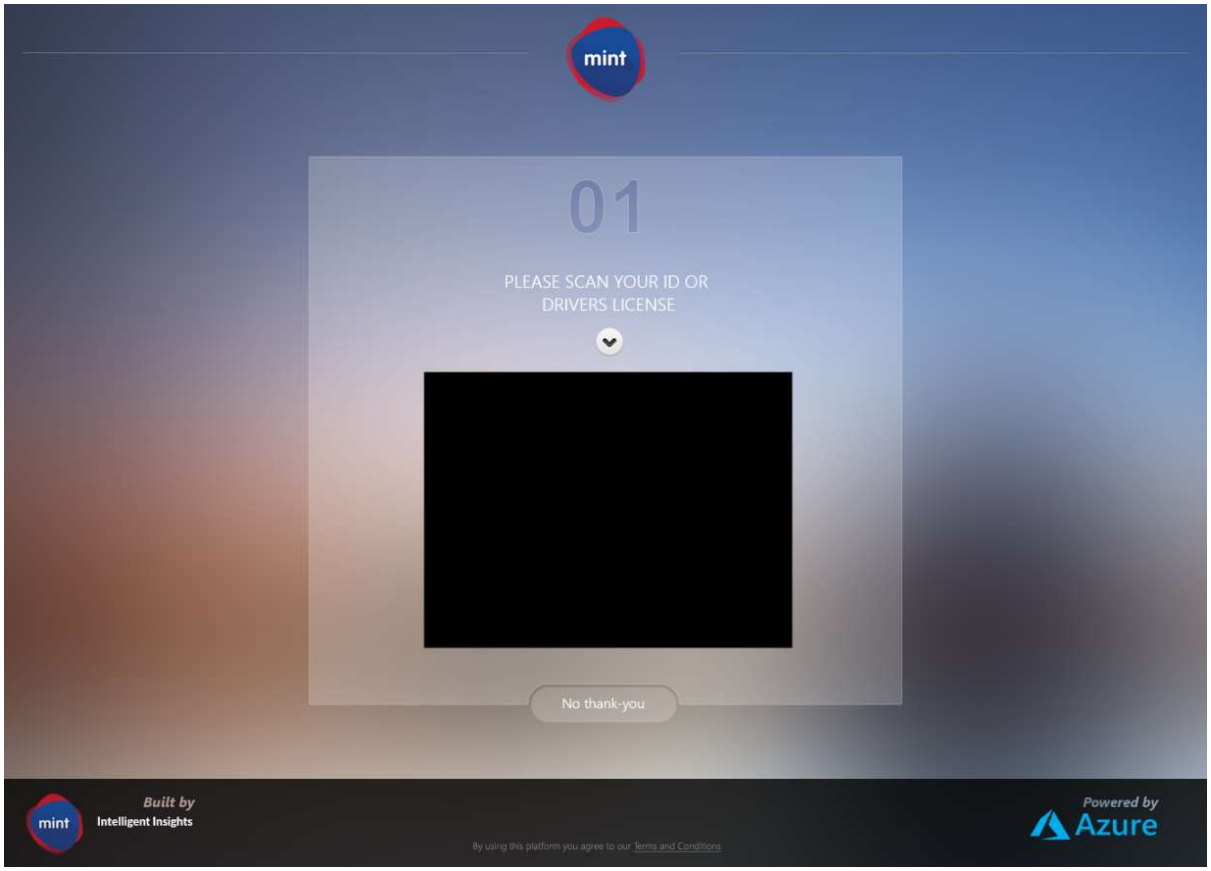


Figure 5: Scan Drivers or ID Document

5.4 Enter ID Number

If the user does not have their documents with them, or does not wish to scan their documents, they can click “No, thank-you” and they are prompted to enter their ID Number manually.

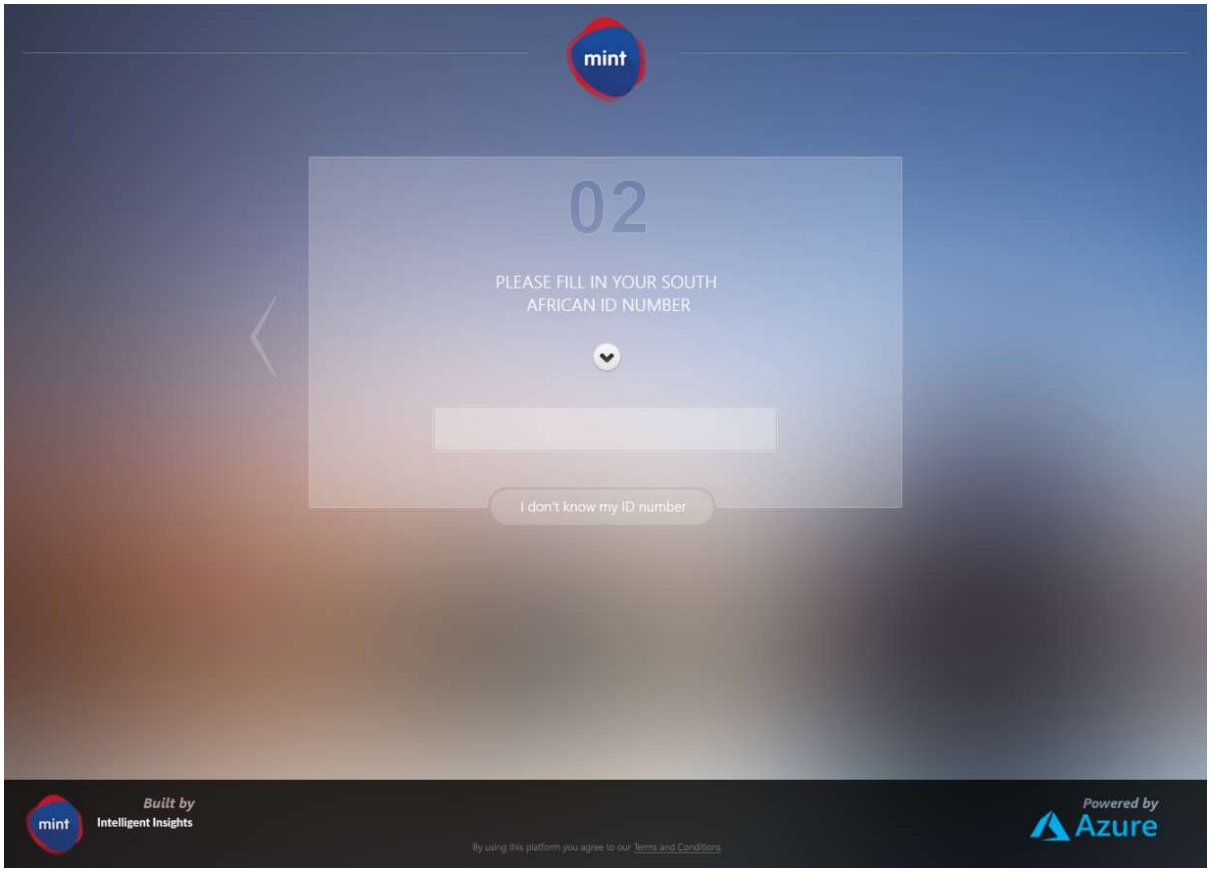


Figure 6: Enter ID Number

5.5 Manually Enter Information

If they user does not have an ID number, or is unwilling to enter their ID number, they can click “I don’t know my ID Number” and they are prompted to enter their details manually. At this stage, they can enter any details that have been configured to be required, such as Name, Surname, as well as extended fields such as Patient Number, Doctor Number or Student Number and email address.

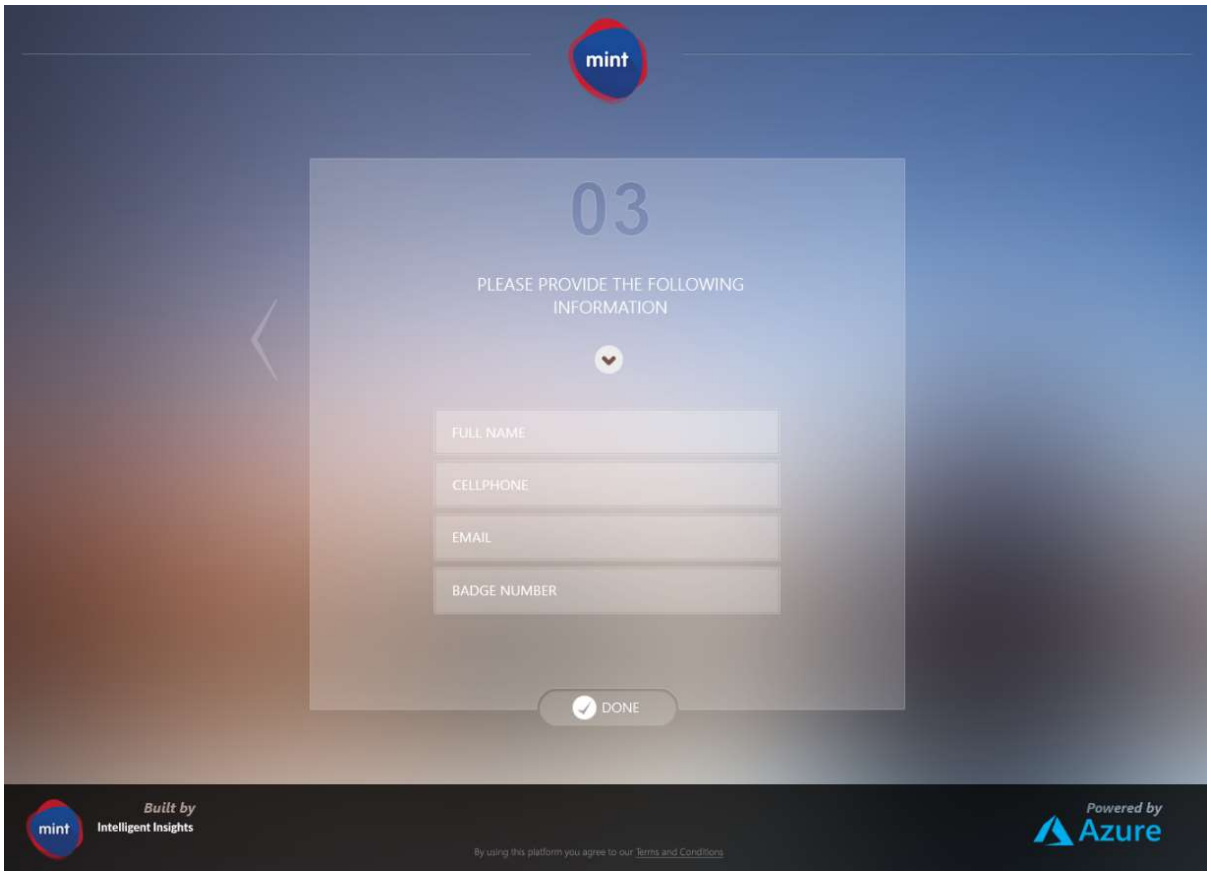


Figure 7: Enter details manually

5.6 Prepare to Capture Images

The user is prompted that they are about to have pictures taken. They are given 5 seconds to prepare, although they can trigger the process immediately by clicking “Do it now”.

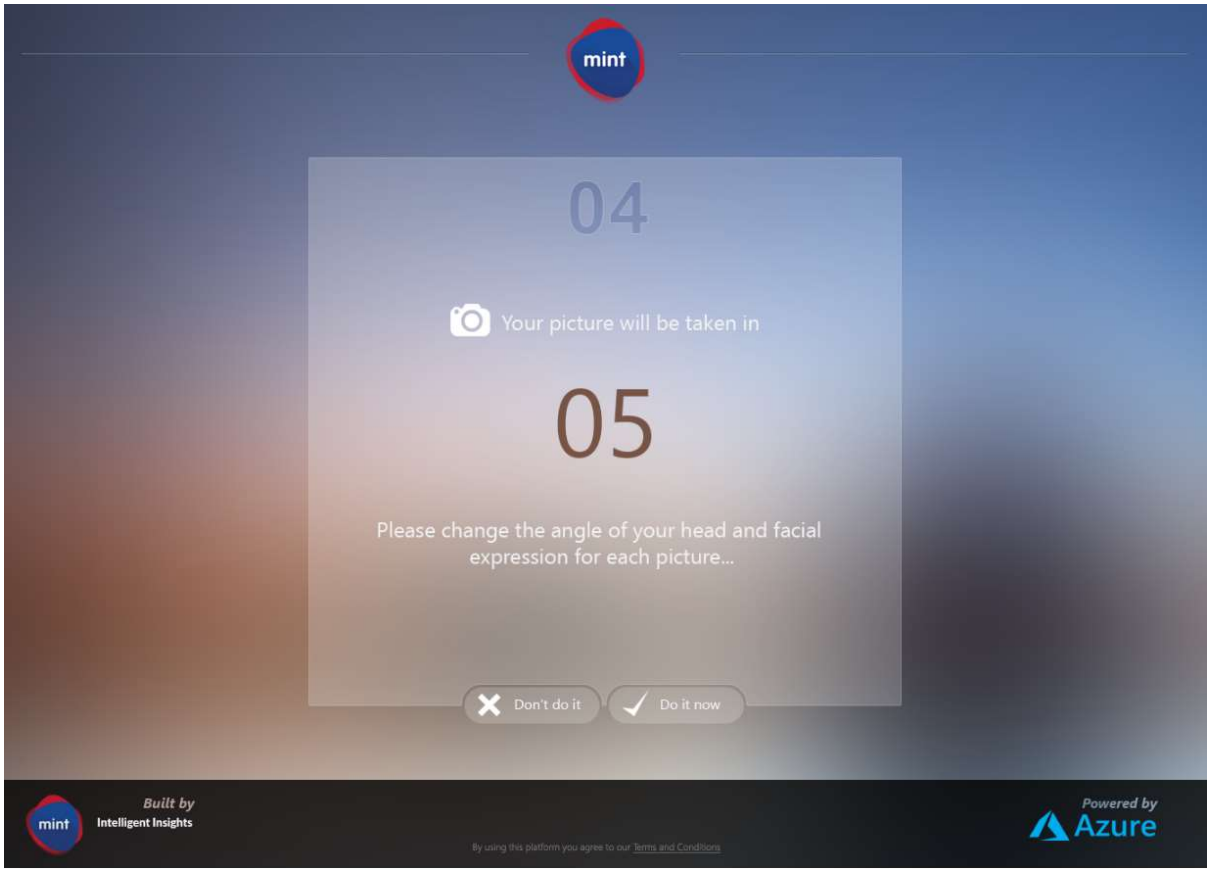


Figure 8: Prepare for images

5.7 Capture Images

The system captures a number of faces, and trains them into the system. Once all of the faces are captured, they are prompted to return to the standard recognition page.

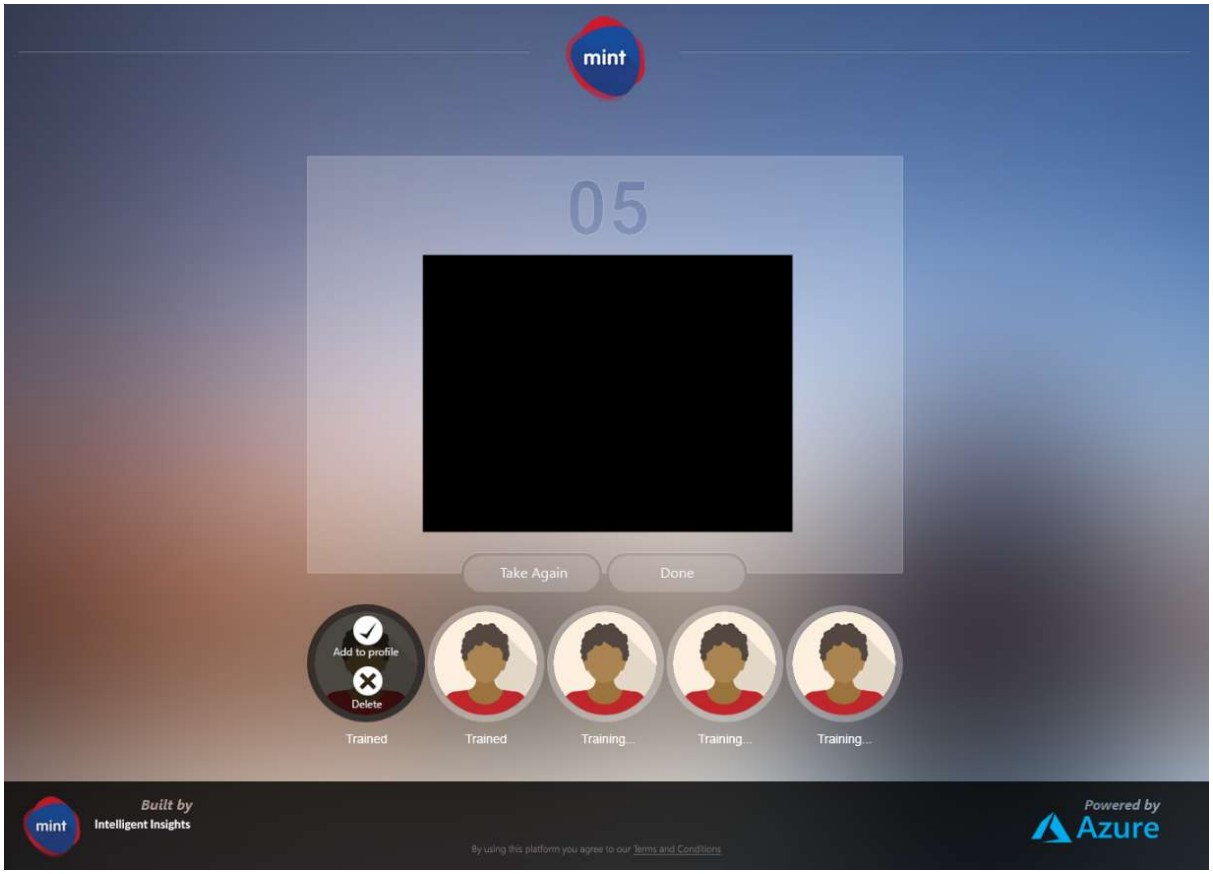


Figure 9: Capture Images

6 Sample Reports and Dashboards

6.1 Reports and Dashboards

Standard Crowd-Tracking reports are available out-of-the-box. Extension reports are available to be customised on request.

(Note: Some elements on the *example* dashboard that relate to the full user journey, as illustrated below, are only applicable in some use-cases.

6.1.1 Sample Crowd Tracking Report (1)

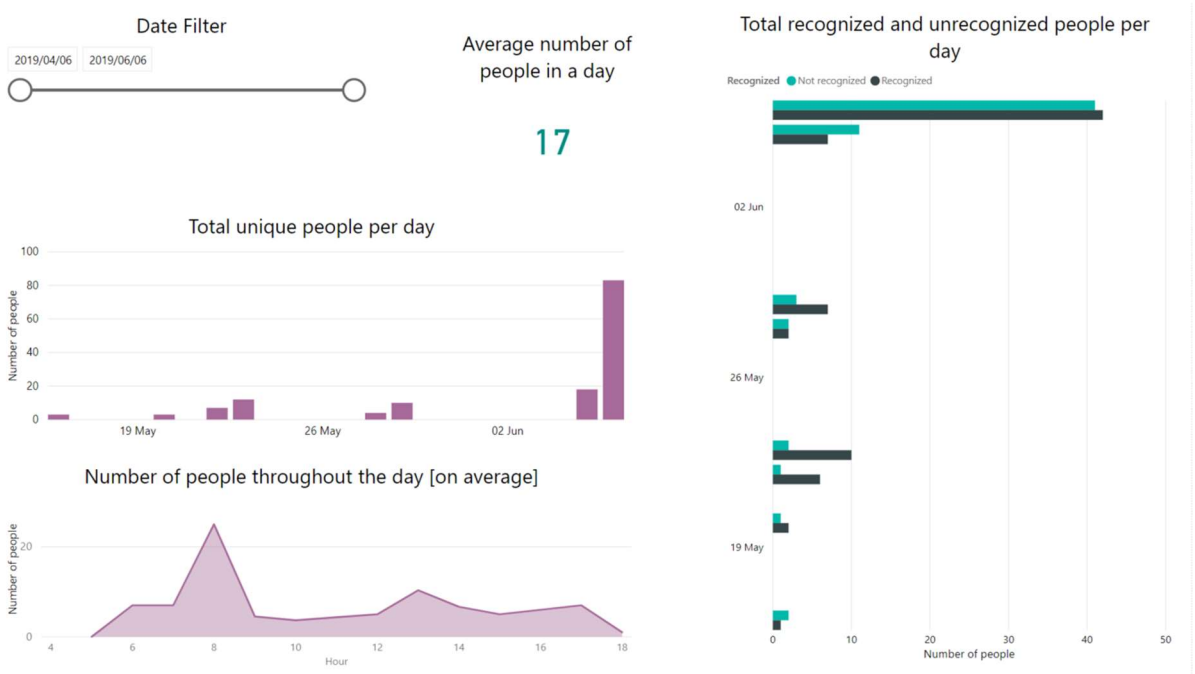


Figure 10: Sample Crowd Tracking Report

6.1.2 Sample Crowd Tracking Report (2)

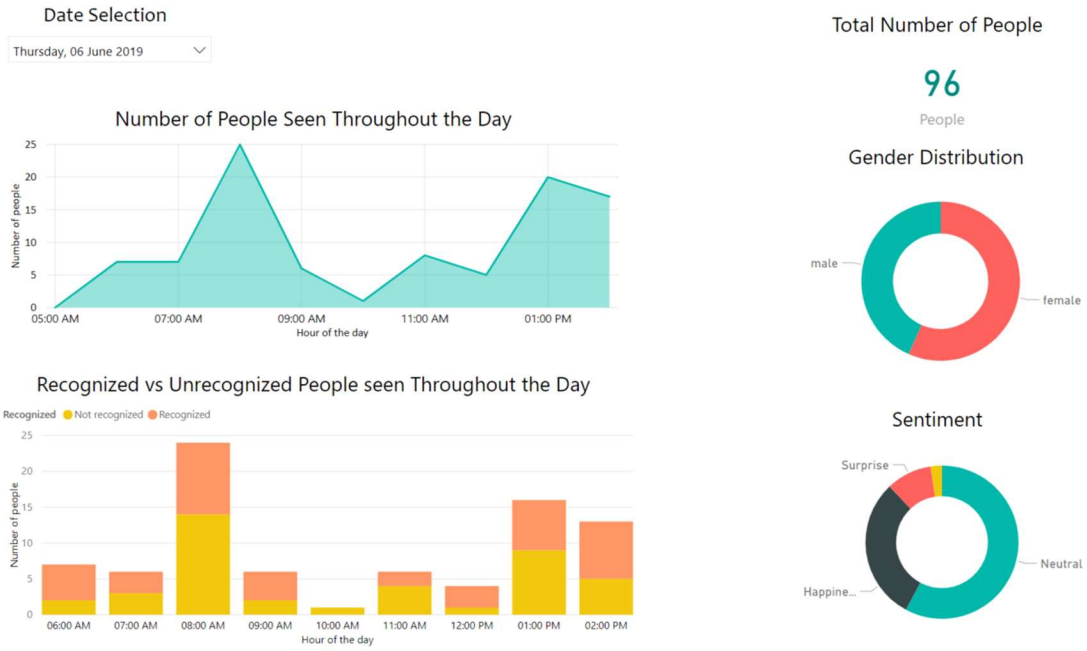


Figure 11: Sample Crowd Tracking Report

6.1.3 Sample Queue Report

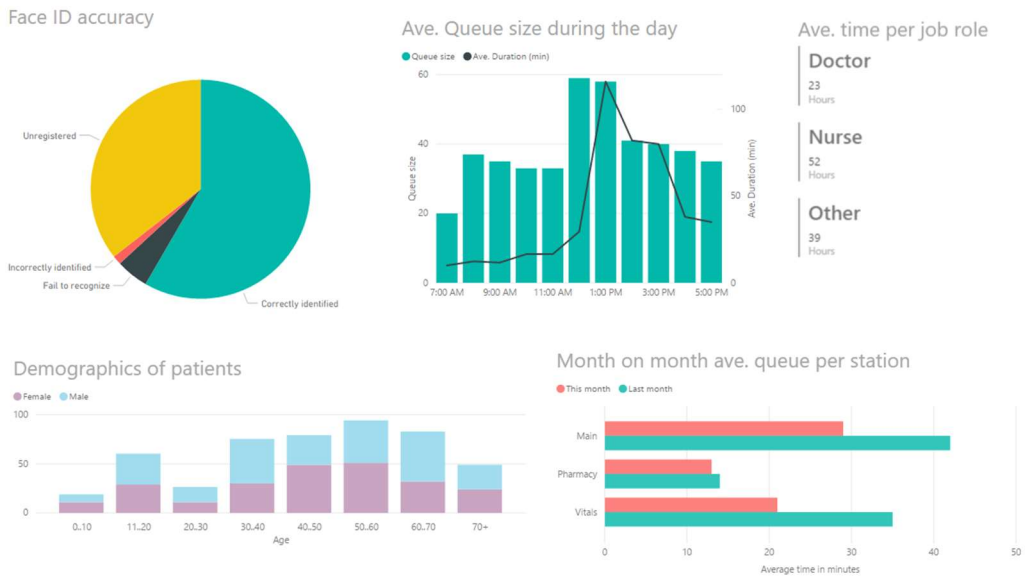


Figure 12: Sample Queue Report