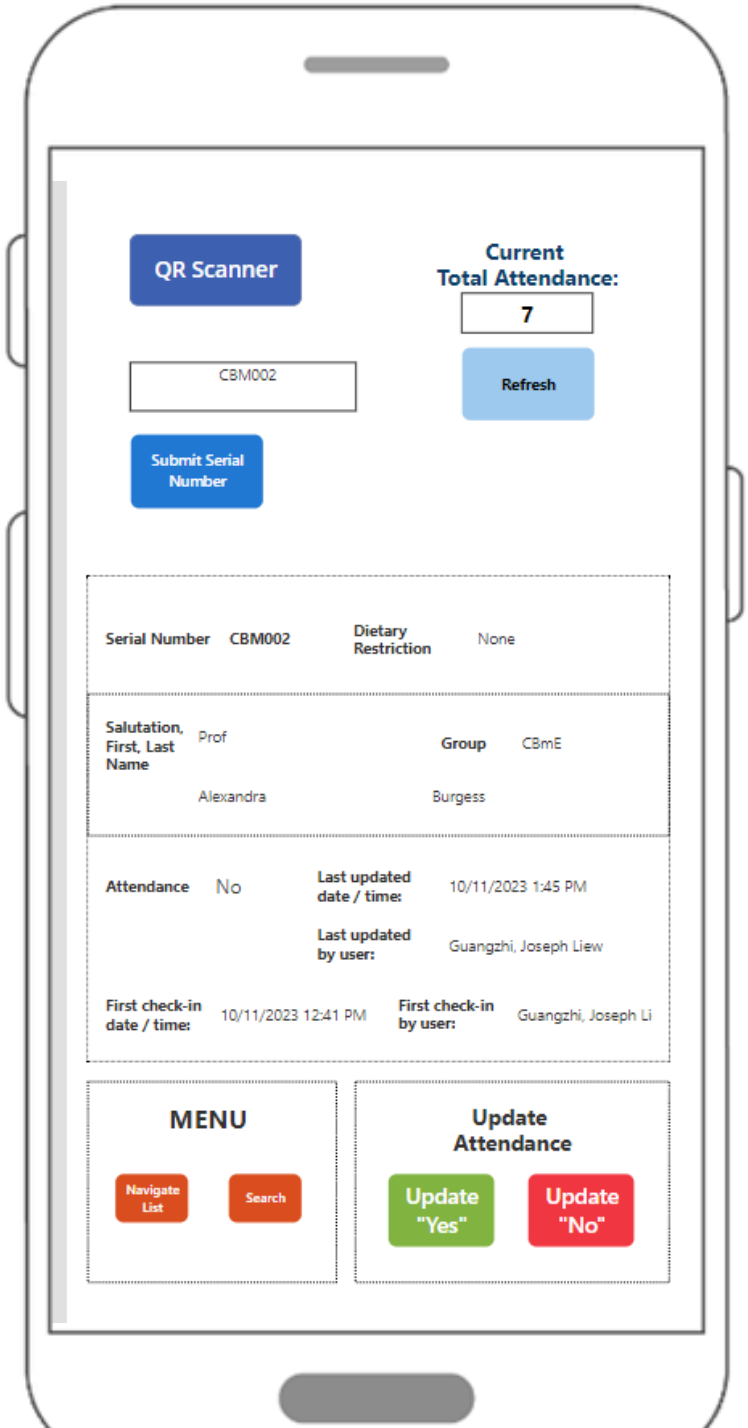
Software Portfolio

Mobile Phone App:

QR Code Scanner for Attendance-taking at Event Reception



Introduction

[Click here to see app demo video]

Use-case:

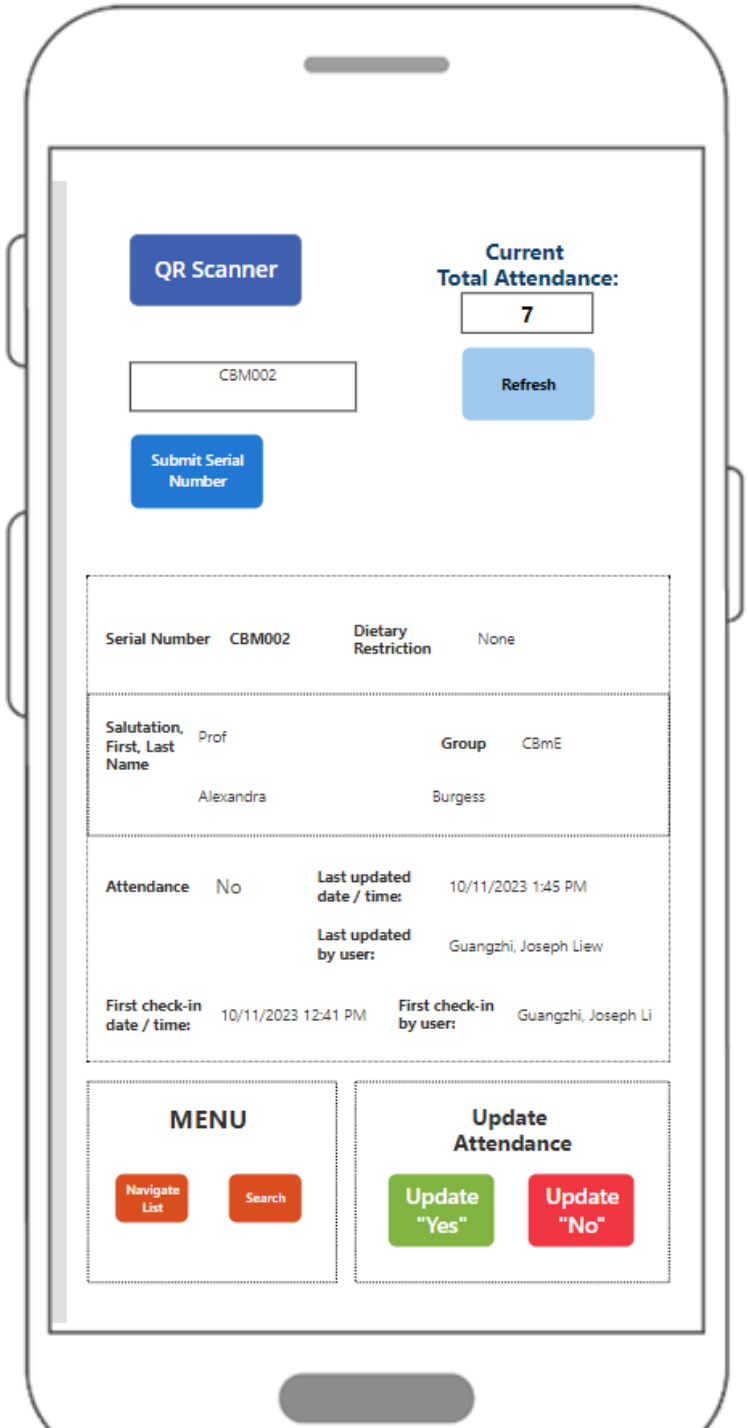
- Event reception staff who are checking-in participants and taking attendance

Core Features

- Scan participant's QR code to automatically take attendance and display participant info.
- Participant misplaced QR code? Search participant info and update attendance
- Edit participant attendance and info
- Real-time data consolidation
- Audit trail
- Fast performance

Software Design

- Object-oriented software design to overcome software and hardware limitation
- Client on-demand system to balance client and server load



Problem Statement

User's Problems

- During events, event organisers need to check-in and take attendance of participants at reception counter.
- Manual attendance tracking is time-consuming.
- Duplicate data entry can occur from multiple staff members checking in the same participant.
- No live update on total attendance.
- Staff lack participant information; unable to direct participant to seats.
- No record of attendee check-in time and audit trail.
- Difficulty updating participant information during the event, such as seating changes.

Technical Challenges

- **Limited infrastructure**

Available infrastructure	Limitation
Database: SharePoint Lists	• No support for server-side computation • Computation reliant on client device • Low cap for concurrent database calls
Client device: Personal mobile phone	• Personal device with low compute power
Software: PowerApps	• Limited development options. No OOP.

Solution

Limitation	Solution
• No support for server-side computation • Computation reliant on client device • Low cap for concurrent database calls	• Eliminate unnecessary database calls using client cache • Conserve client and server load. Use pull-based design to deliver updated data on demand.
• Personal device with low compute power	• Only cache data required for computation • Use modular design and cache data on client to avoid data duplication
• Limited development options. No OOP.	• Use object-structured design with global variables as cache

Object-Structured Design

- Split app to 3 core modules,
  matching 3 different user workflows
  - Query / update via QR code
  - Query / update via search
  - Query / update via data table navigation
- Each module has own environment and cache
  - Client only loads the required environment selected by user
  - Conserves client memory for necessary cache
- Never load entire database
  - Only data of individual participant that the user has selected will be loaded into cache and displayed for modification

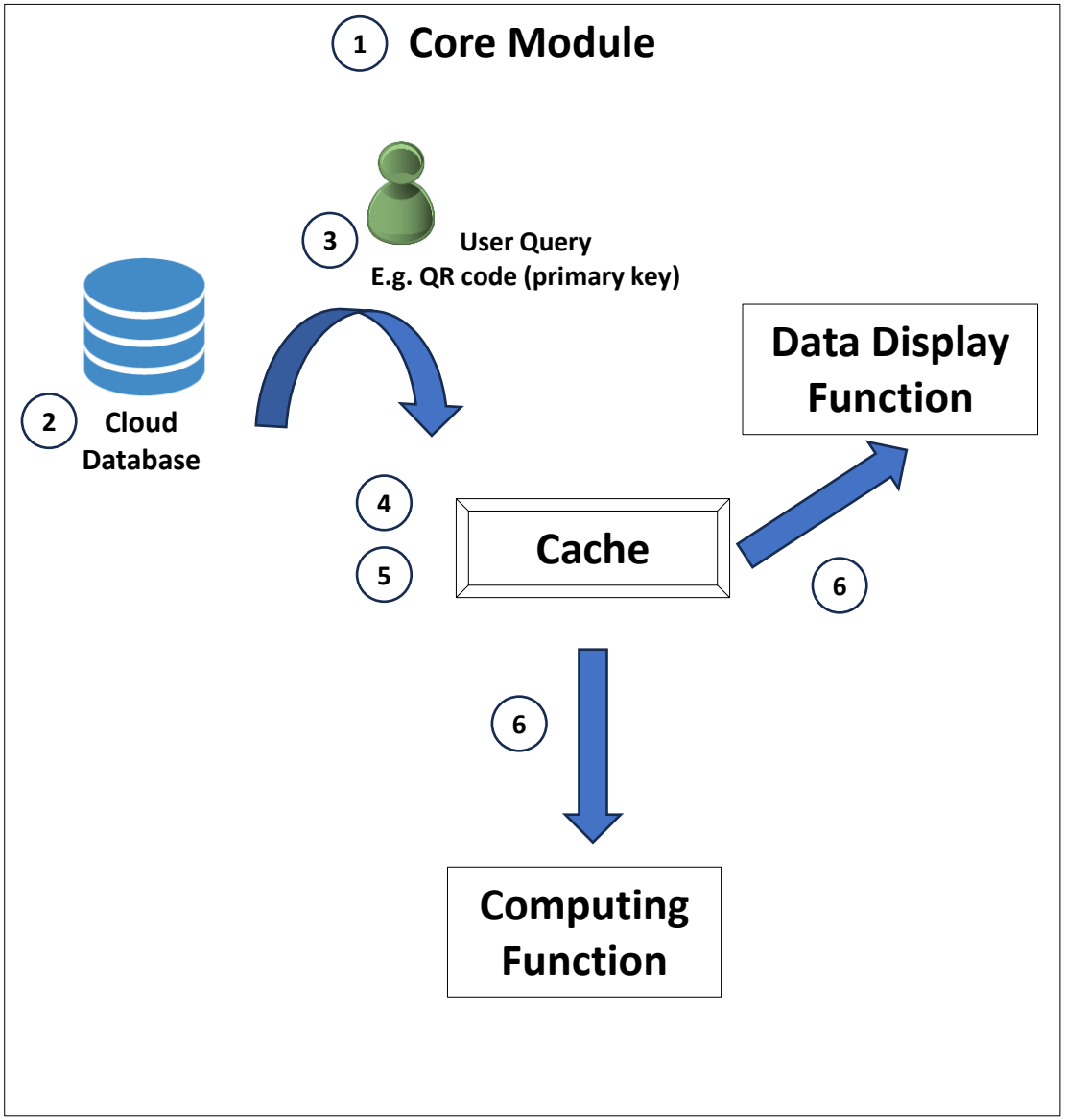
Pseudocode Flowchart

Object structured design allows efficient reuse of common cache for device computation

Data flow

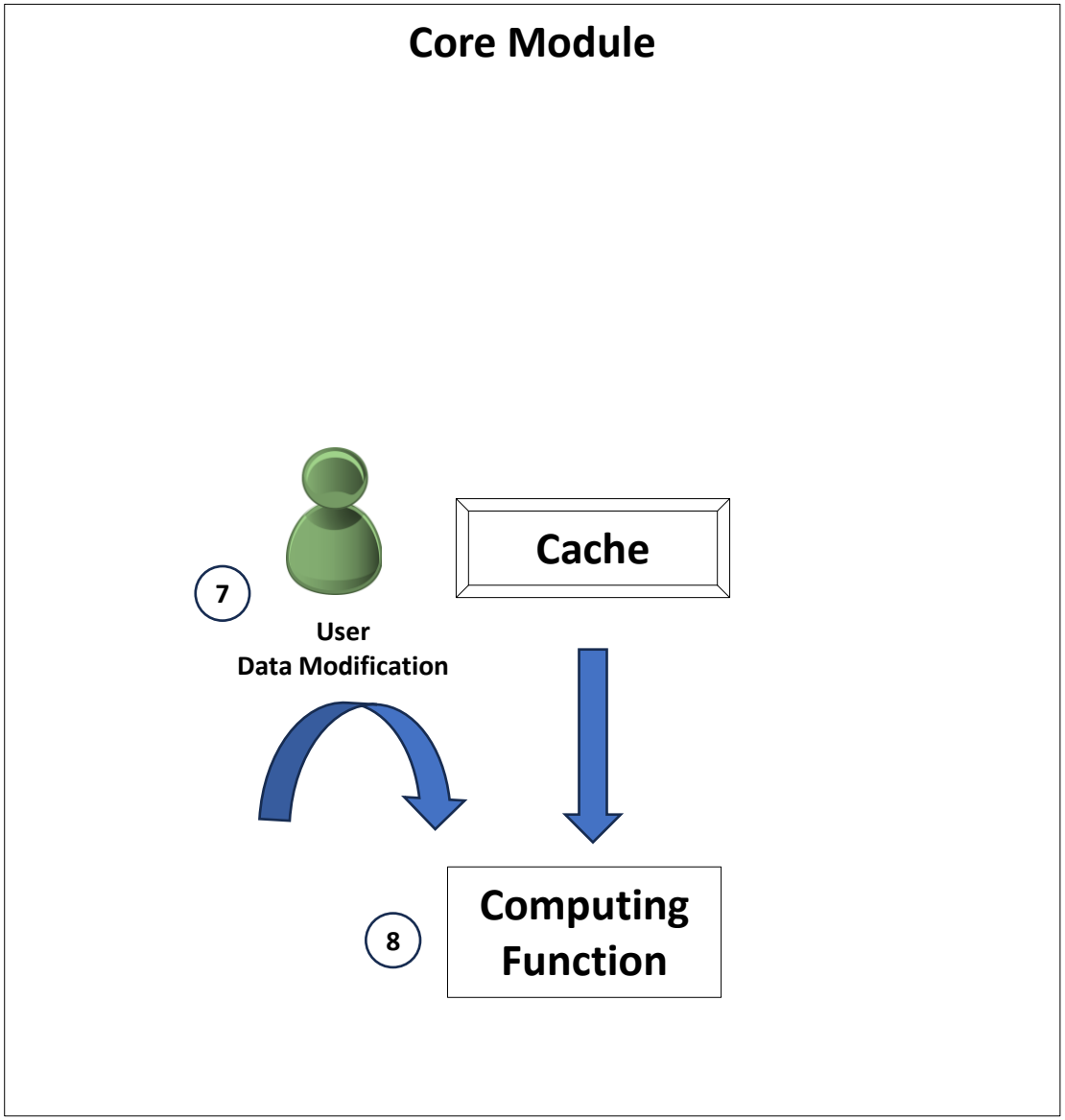
1. User selects desired **Core Module**
2. Data is stored in **Cloud Database**
3. Primary key inputted by user input translated into **User Query**
4. Device pulls query from **Cloud Database** and stores data in device's **Cache**
5. **Cache** provides data for current query instance, providing data required by different functions in the module.

   (Prevents duplicate database calls.)
6. Cache feeds data to **Data Display Function** for user to view data and **Computing Function** for user to modify data



Pseudocode Flowchart

7. If there is **User Data Modification**
8. Then **Computing Function** processes changes to cached data

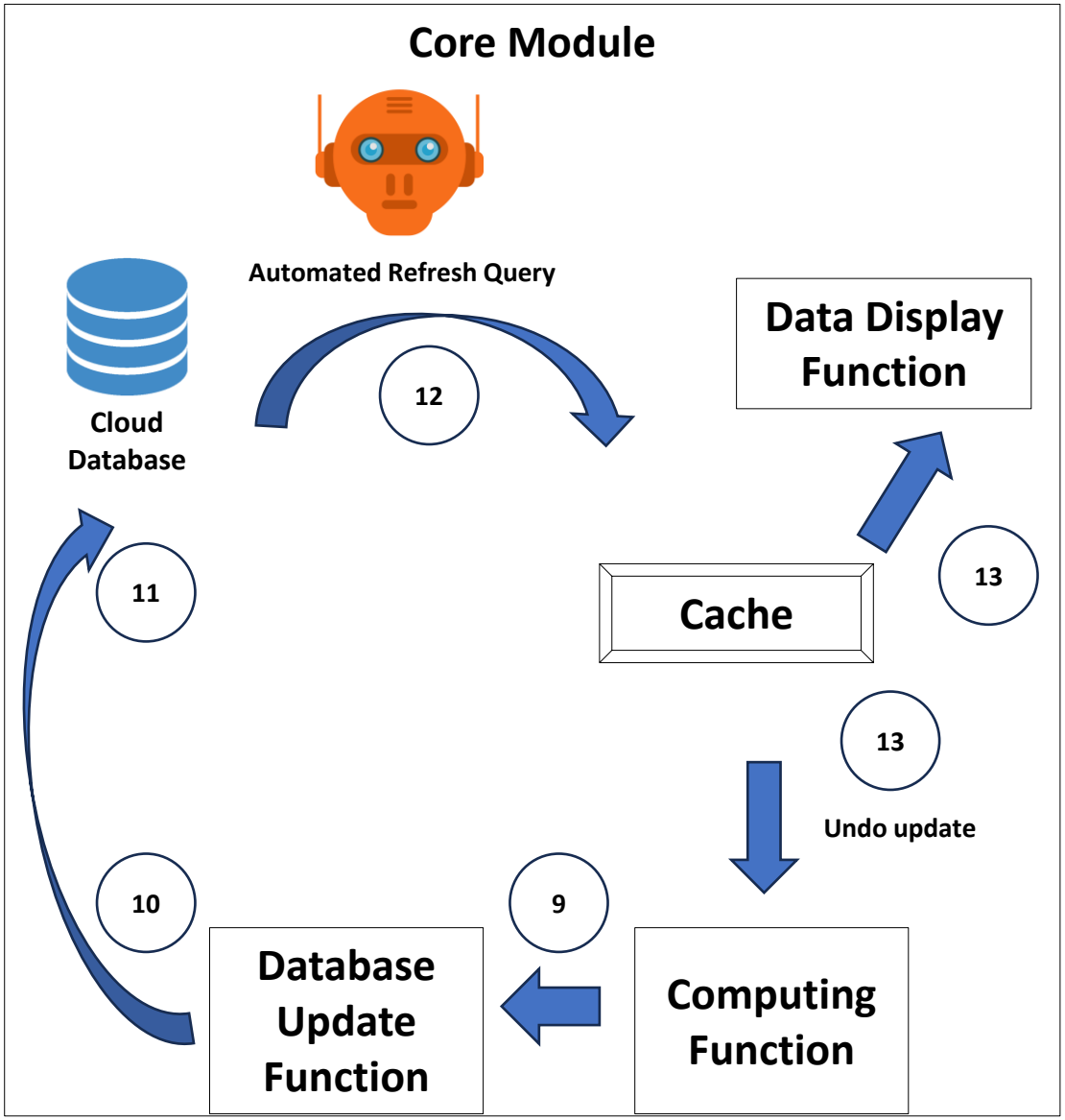


Pseudocode Flowchart

9. **Computing Function** selects only modified data from cache and push to **Database Update Function**
10. **Database Update Function** identifies primary key and patches changes to **Cloud Database**
11. In this way, all devices will have synced data when they refresh the same query from **Cloud Database**
12. In current device, **Automated Refresh Query** refreshes **Cache** data

    (This allows the cache to be hydrated by most recent update. The most recent update might be made from another device.)
13. **Cache** hydrates the data displayed in the **Data Display Function**.

    Or, if user wants to undo change, feeds data to the **Computing Function** to process the undo update.



Core Modules

[Click here to see app demo video]

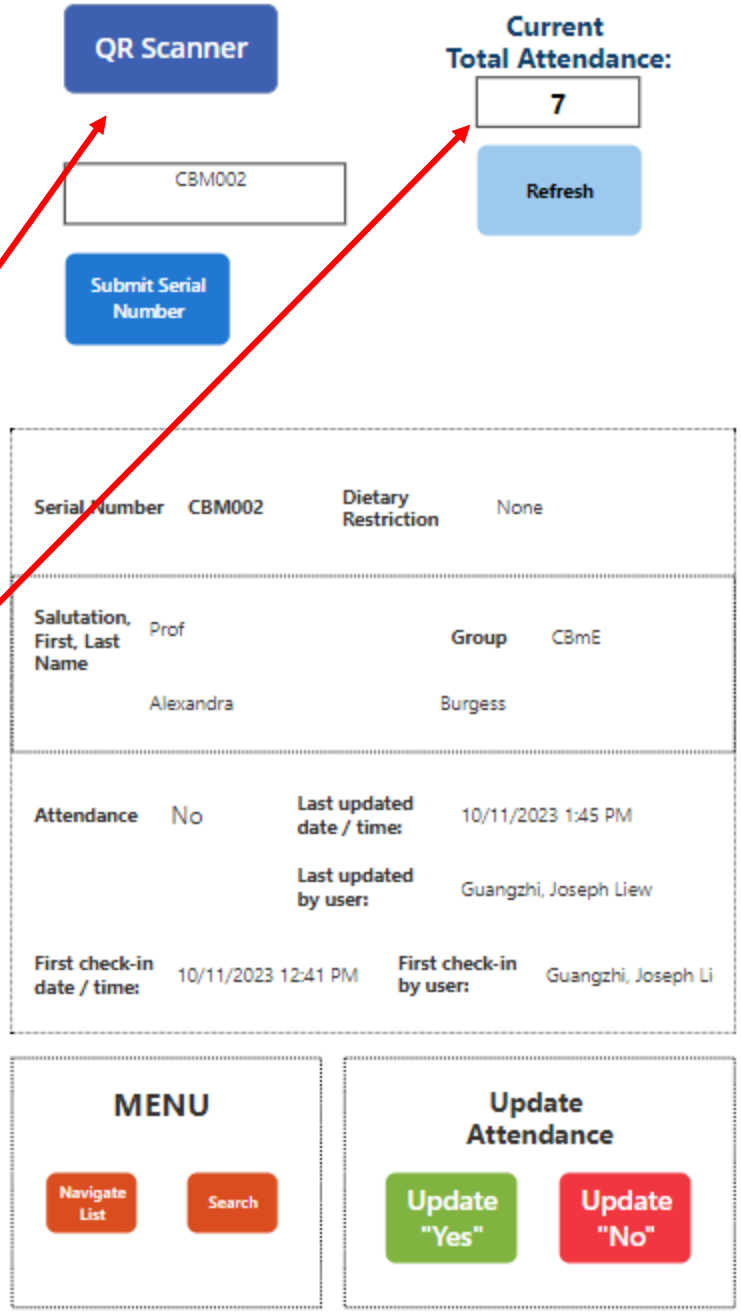
QR scanning for attendance taking and on-demand info retrieval

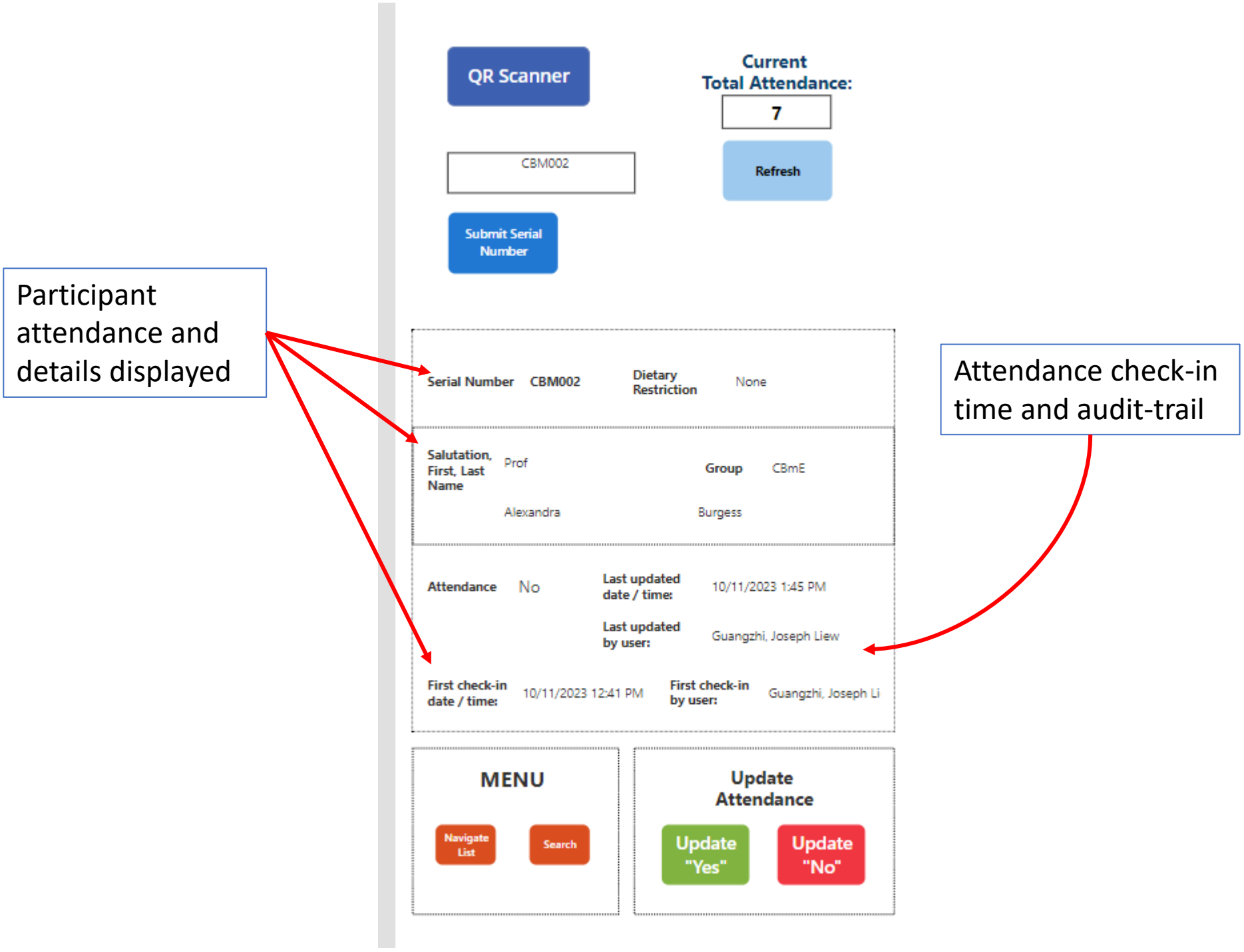
Press to use phone camera to scan QR code

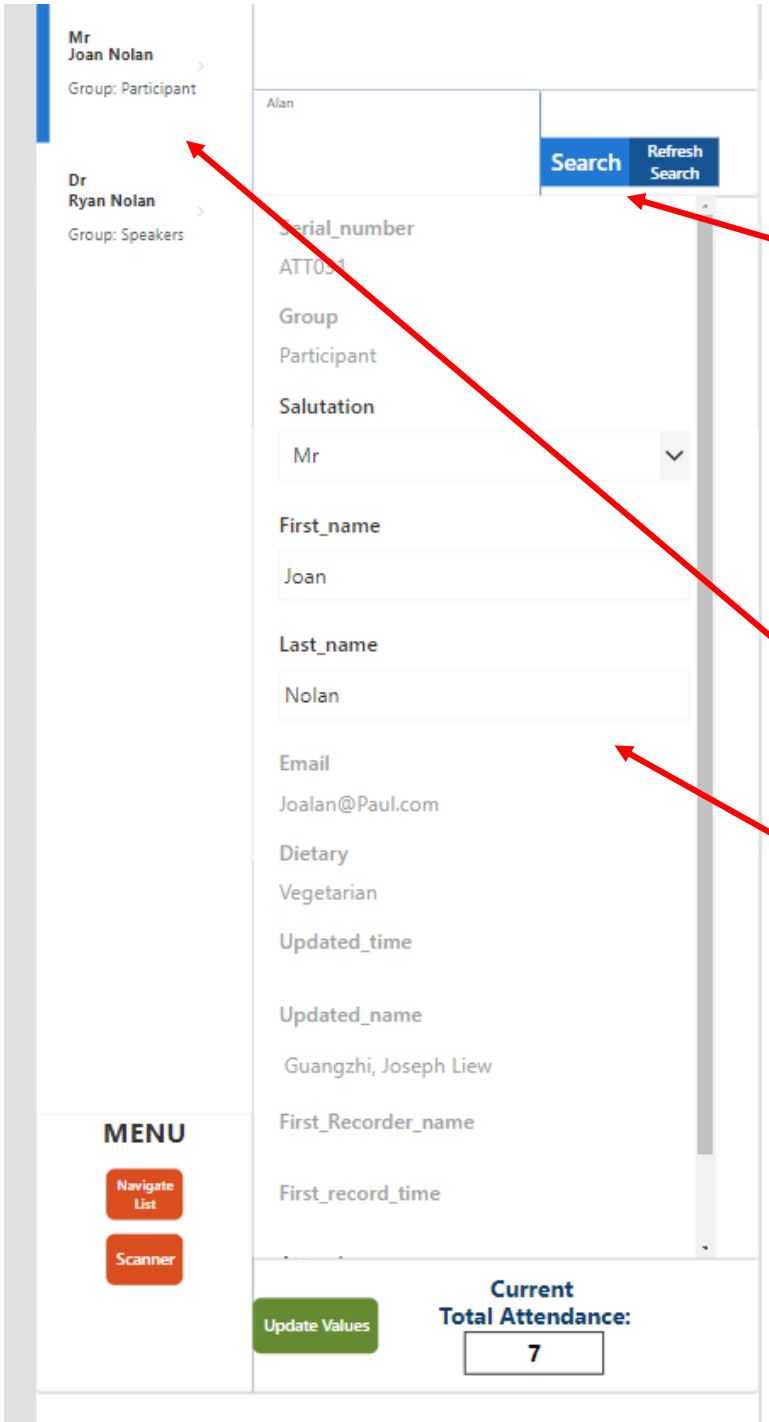
Attendance is automatically updated with no duplicates

Total attendees automatically refreshed when attendance is taken

Also refreshed on-demand when user selects refresh.




QR Scanner
Current Total Attendance:
7
CBM002
Refresh
Submit Serial Number
Participant attendance and details displayed
Serial Number CBM002
Dietary Restriction None
Salutation, First, Last Name Prof Alexandra Burgess
Group CBmE
Attendance No
Last updated date / time: 10/11/2023 1:45 PM
Last updated by user: Guangzhi, Joseph Liew
First check-in date / time: 10/11/2023 12:41 PM
First check-in by user: Guangzhi, Joseph Li
Attendance check-in time and audit-trail
MENU
Navigate List
Search
Update Attendance
Update "Yes"
Update "No"



On-demand Pull-based Search

Search box:
If participant misplaced QR code, easily find search the database to find participant.

User then selects participant to view data / update attendance.

Search results with minimum identifier data downloaded from cloud to device cache.

Full data of participant only queried from database when user selects a search result.

This is staged data hydration to cache.

Avoid server lag and device memory overload caused by unnecessary download of entire dataset.

On-demand Pull-based Navigation and Update

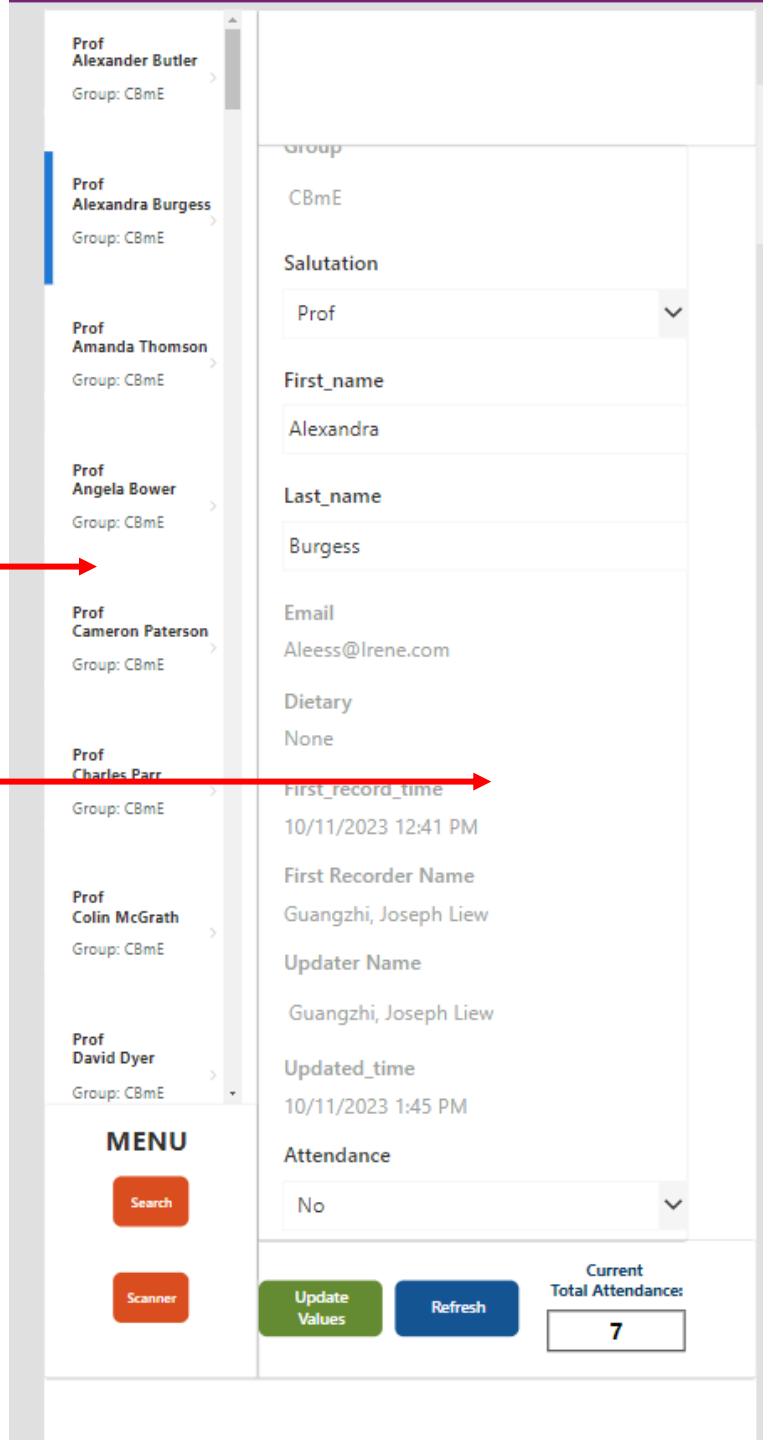
User can also select participant to view or update via table navigation

Minimum identifier data downloaded from cloud to device cache and shown on table.

Full participant data of participant only queried from database when user selects a participant.

This is staged data hydration to cache.

Avoid server lag and device memory overload caused by unnecessary download of entire dataset.



Results

Better event management

- Live update of total attendees enabled event organiser to move participants from reception to event venue at optimum time.

Reduced manpower

- Manpower at check-in reception reduced by half.
- Faster check-in: staff can process more check-ins in shorter period.
- In app search function eliminated the need to re-direct participant with no registration number or QR code a dedicated check-in counter.

Increased efficiency

- Attendance automatically consolidated into single database.
- Simplified logistics:
  - All-in-one phone app for update, search, info retrieval
  - Replaced computer station setup at reception