JOINT INSTITUTE
交大密西根学院

Capstone Project: VMware#1-AR in Data Centers

Design Review #3

Group 21
Shuyi Zhou Liying Han Jinglei Xie
Chenyun Tao Yaxin Chen

Instructor
Mingjian Li
Sponsor & Mentor
Gavin Lu Yixing Jia

Team Introduction

Leader

Shuyi Zhou
Senior
ECE

Member

Chenyun Tao
Senior
ECE

Member

Liying Han
Senior
ECE

Member

Yaxin Chen
Senior
ECE

Member

Jinglei Xie
Senior
ECE

Contents

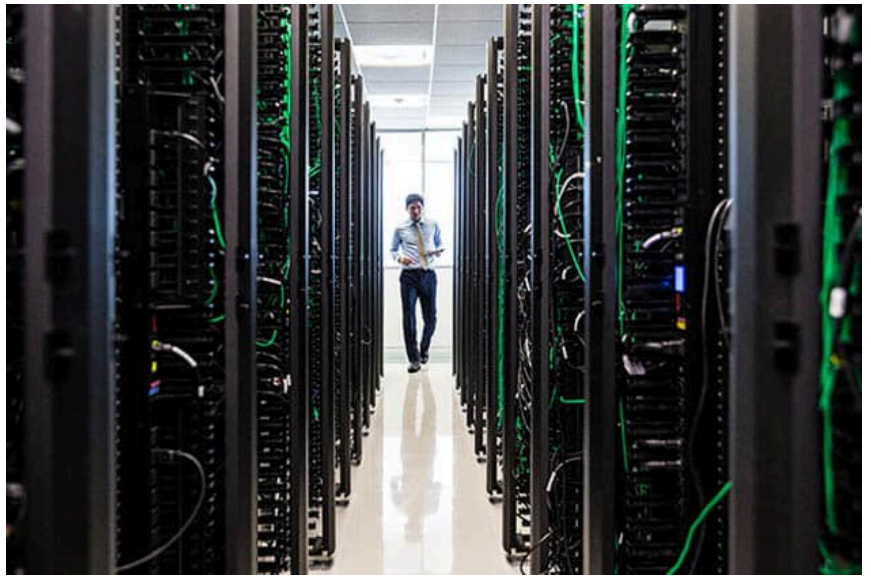
- Introduction
- Engineering Design Analysis
- Final Design Description
- Implementation & Current Progress
- Demonstration & Plan

Contents

Problems & Needs in Data Centers (DC)

Maintenance and audits

- Do not have **integrated** information system
  - Need an **integrated system** that involves all the information together
- Lack **user-friendly** instructions
  - Need a more **user-friendly** tool for instructions and information access

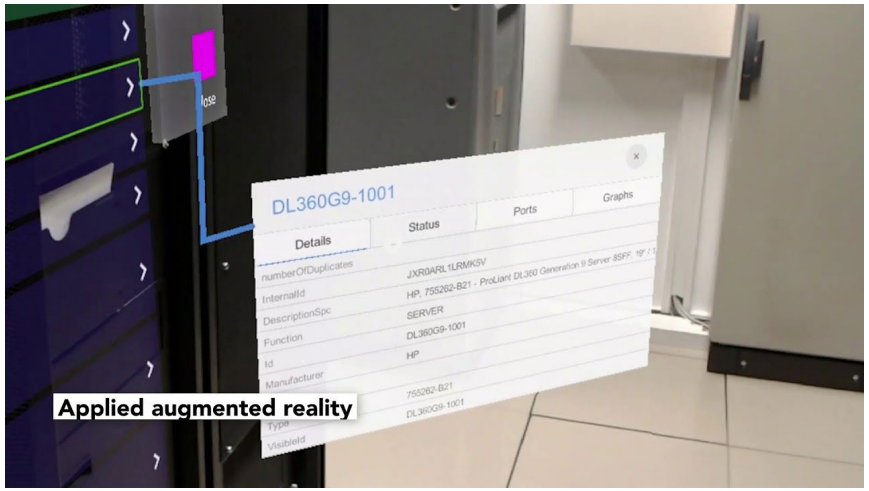


www.cisco.com

Project Goal

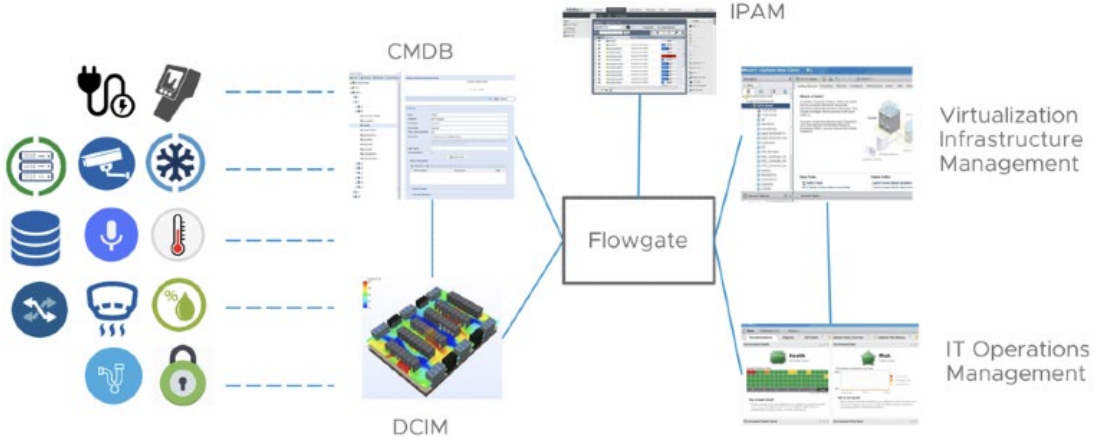
An Augmented Reality (AR) App
for aiding on-site DC maintenance & audit

- **Front-end: AR**
  - A user interface to display the information vividly



www.youtube.com/watch?v=1Pe028PjQhs

- **Back-end: Flowgate**
  - An integrated system containing all necessary information of DC



https://github.com/vmware/flowgate

Customer Requirements (CR) & Engineering Specifications (ES)

CR: Short Reaction Time

ES:

- Barcode localization & identification: < 0.55s [1]
- Database query complexity: O(log(n))
- AR image generation: < 0.1s [2]

CR: Portable Device

ES:

- Platform: Android 7.0+ / iOS 11.0+ [3]
- Light: >= 40lx [4]
- Software package size: < 110MB for Android / < 940MB for iOS [5]

[1] E. Ohbuchi, H. Hanaizumi and L. A. Hock, "Barcode readers using the camera device in mobile phones," *2004 International Conference on Cyberworlds*, Tokyo, Japan, 2004, pp. 260-265, doi: 10.1109/CW.2004.23.
[2] A. Baek, K. Lee, and H. Choi, "CPU and GPU parallel processing for mobile Augmented Reality." 2013.
[3] https://developers.google.com/ar/discover/supported-devices & https://developer.apple.com/documentation/arkit
[4] L. Blom, "Impact of light on augmented reality." Diva Portal. 2018.
[5] https://play.google.com/store/apps & https://www.apple.com/app-store

Customer Requirements (CR) & Engineering Specifications (ES)

CR: Information Correctness
ES:

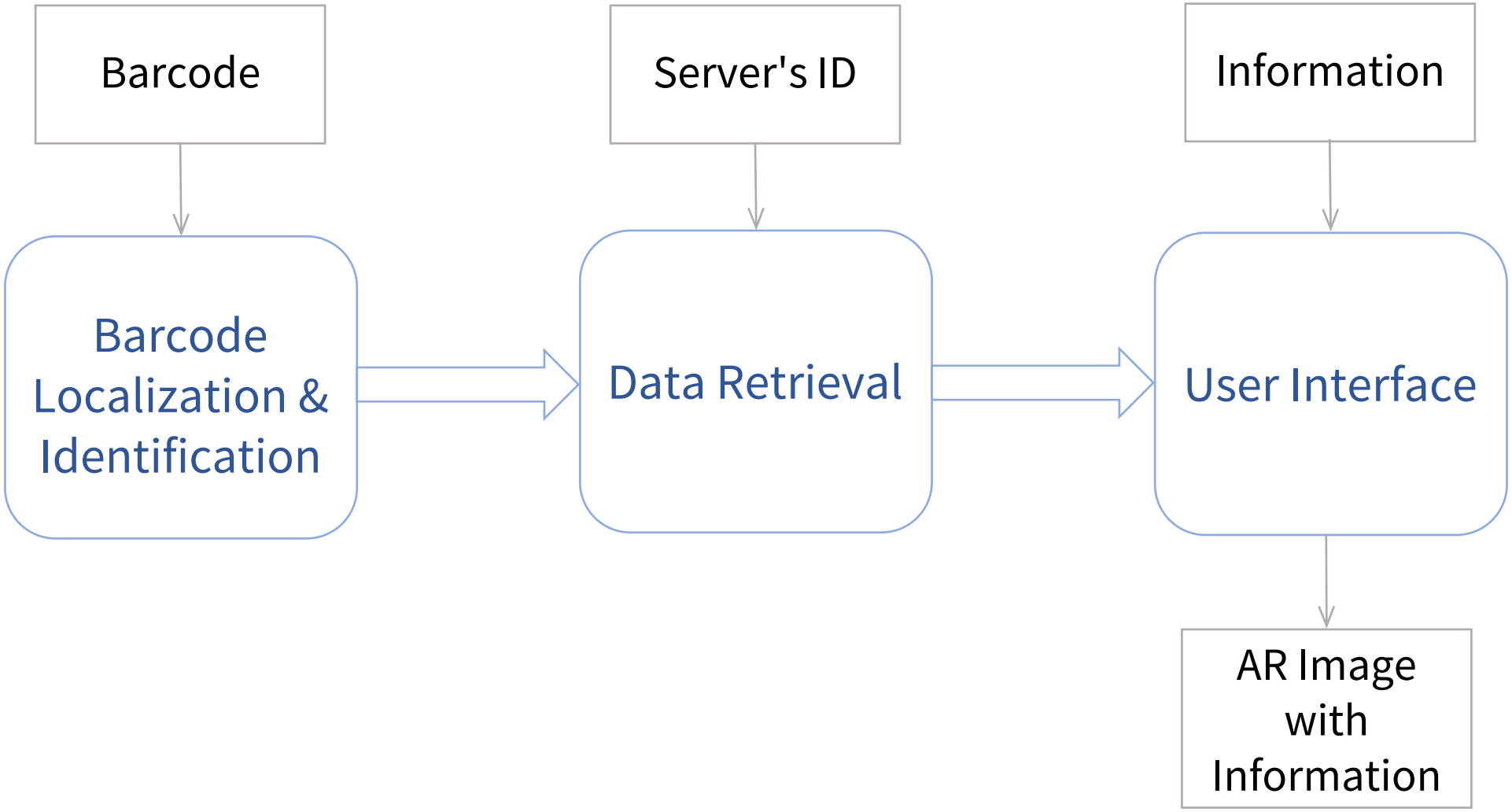
- Barcode localization correctness: > 90% [1]
- Data retrieval accuracy: > 99% [2]

CR: Comfortable Display
ES:

- Frame rate: > 15 frames/s [3]
- Sensible temperature of device: < 40 °C [4]

[1] O. Oktay et al., "Stratified Decision Forests for Accurate Anatomical Landmark Localization in Cardiac Images," in IEEE Transactions on Medical Imaging, vol. 36, no. 1, pp. 332-342, Jan. 2017, doi: 10.1109/TMI.2016.2597270.
[2] https://www.labce.com/spg650115_barcode_reading_and_accuracy.aspx
[3] A. Craig. Augmented Reality Hardware, pp. 69-124. 2013.
[4] https://support.apple.com/en-us/HT201678 & https://support.google.com/pixelphone/answer/9134668?hl=en

Concept Diagram



Contents

Barcode Localization & Identification

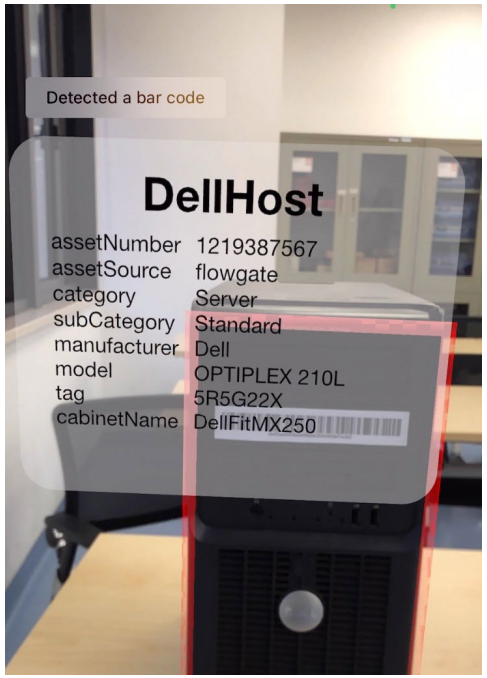
- **Barcode Localization & Identification**
  - Generated from ES & difficulty in implementation
- **How to choose software development kit?**

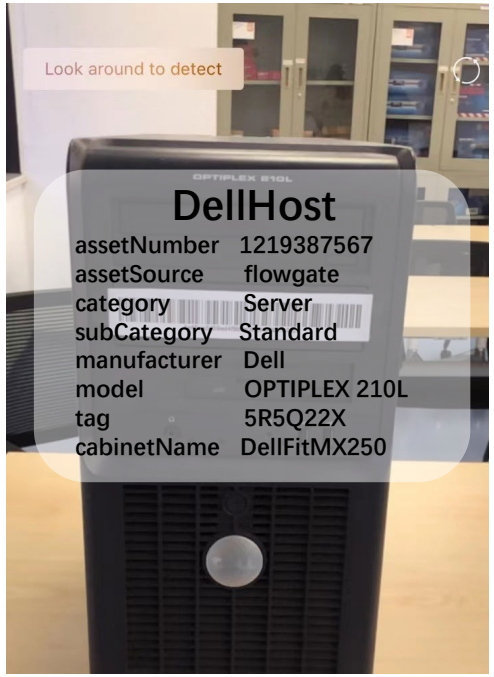
- **ZXing [1]**
  - Open source
  - Slow identification
  - Low accuracy
- **ML Kit**
  - Open source (Google)
  - Moderate speed
  - Moderate accuracy
- **Scandit**
  - Close source
  - Fast identification
  - High accuracy

[1] http://www.discoversdk.com/compare/scandit-_-barcode-scanner-sdk-vs-zxing

User Interface

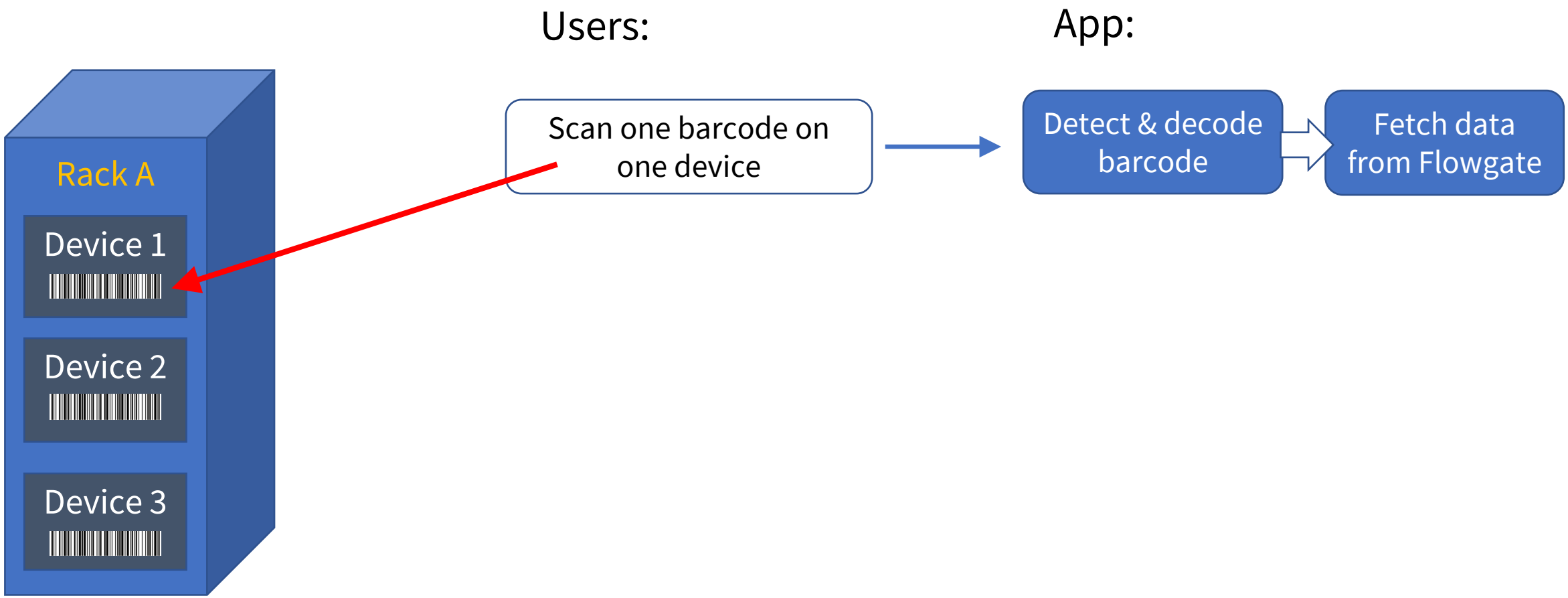
- **User Interface**
- Generated from CR & survey about existing similar softwares
- **How to display information?**
  - Display in AR (3-D coordinate)
  - Display on screen (2-D coordinate)



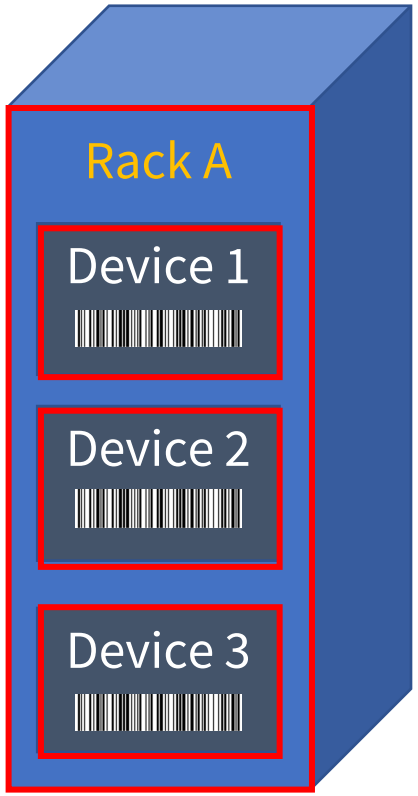


Contents

Final Design Workflow



Final Design Workflow



Users:

App:

Scan one barcode on one device

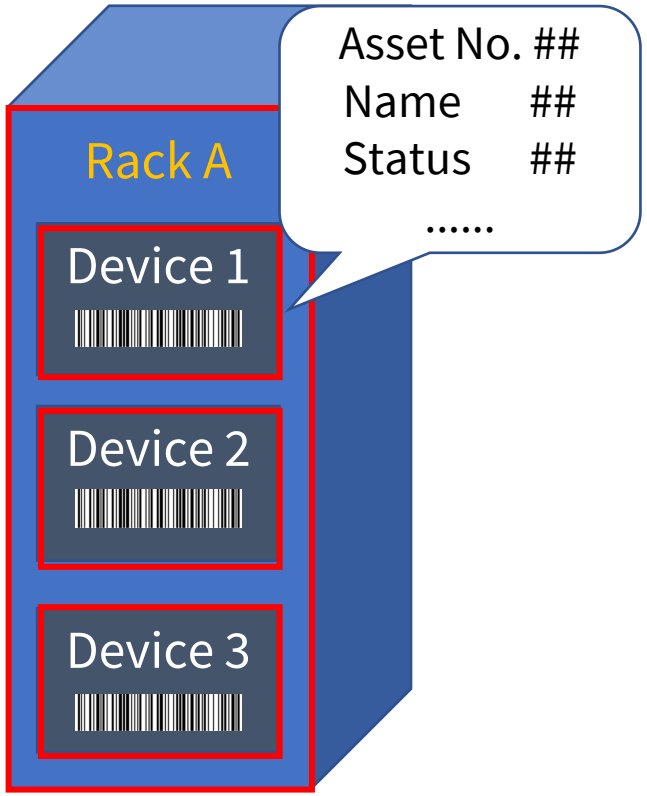
Step back to catch the whole view of rack

Detect & decode barcode

Fetch data from Flowgate

Show location of rack & device

Final Design Workflow



Users:

App:

Scan one barcode on one device

Step back to catch the whole view of rack

Step closer to maintain device with given info.

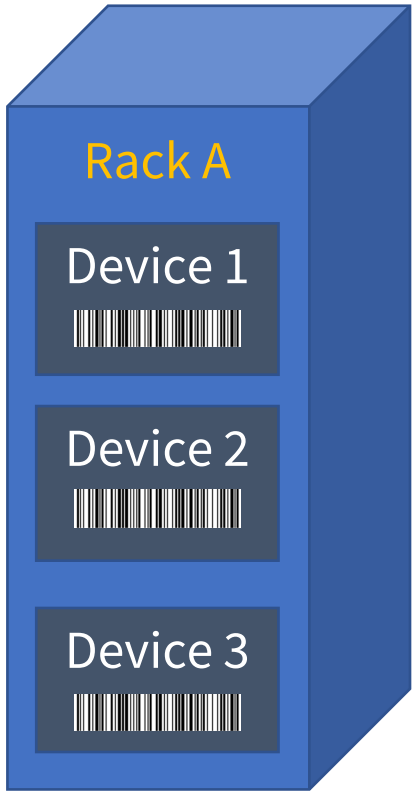
Detect & decode barcode

Fetch data from Flowgate

Show location of rack & device

Show data of device

Final Design Workflow



Users:

App:

Scan one barcode on one device

Step back to catch the whole view of rack

Step closer to maintain device with given info.

Finish.
Maintain another device

Detect & decode barcode

Fetch data from Flowgate

Show location of rack & device

Show data of device

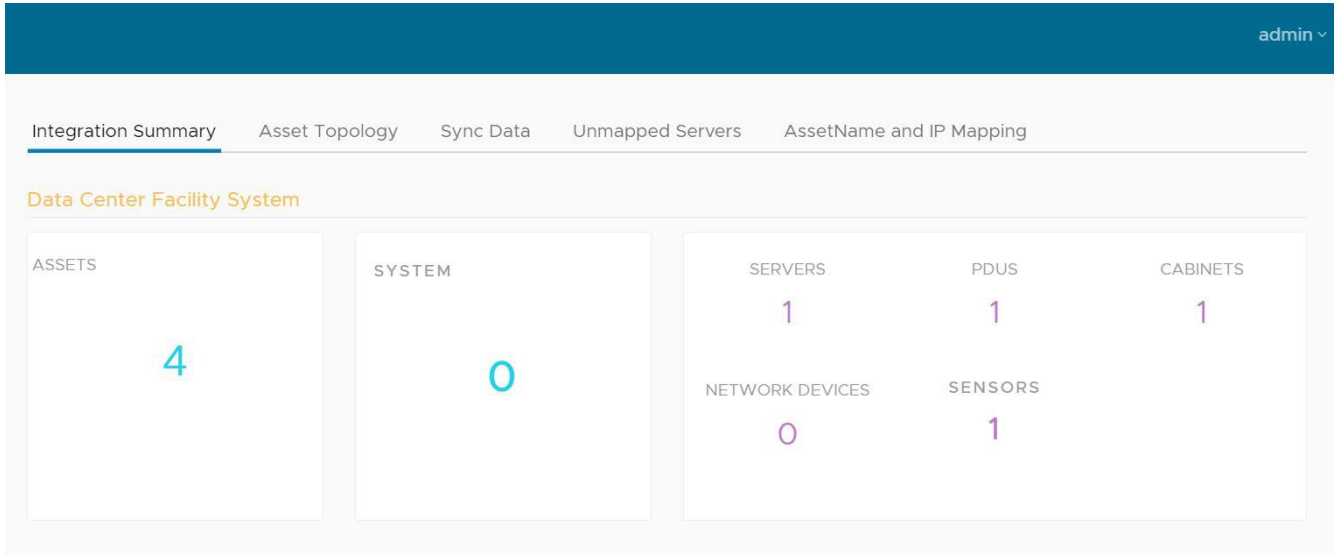
Initialize the app state

Contents

Steps of Implementation

Step 1
Configure backend server

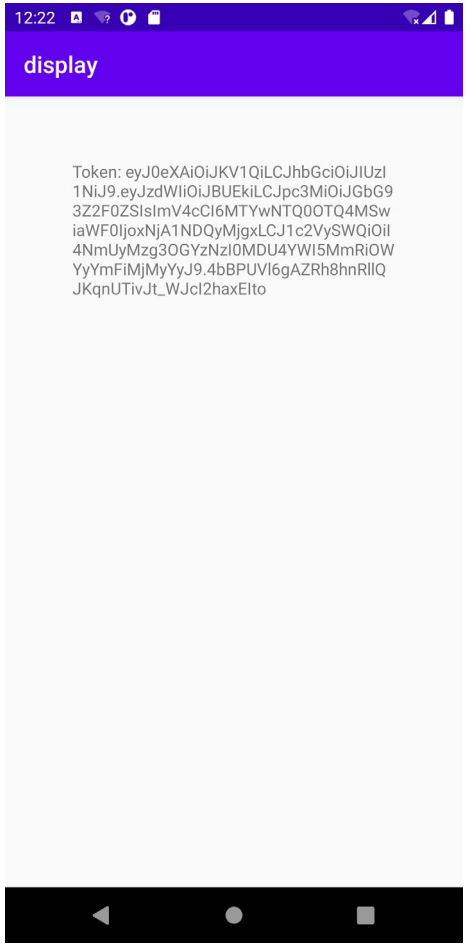
- Install flowgate server
- Create informations sets for data center assets



Steps of Implementation

Step 2-1
Implement data management functions using API

- Python (testing and asset mapping)
- Java (raw data retrieval in Android)
- Swift (raw data retrieval in IOS)



Steps of Implementation

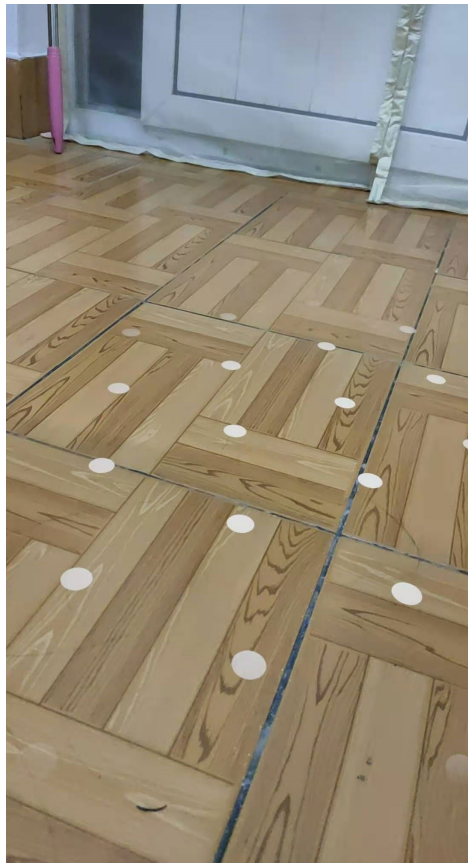
Step 2-2
Implement bar code scanning

- Read data strings from the bar codes

Steps of Implementation

Step 2-3
Implement simple AR apps (for testing)

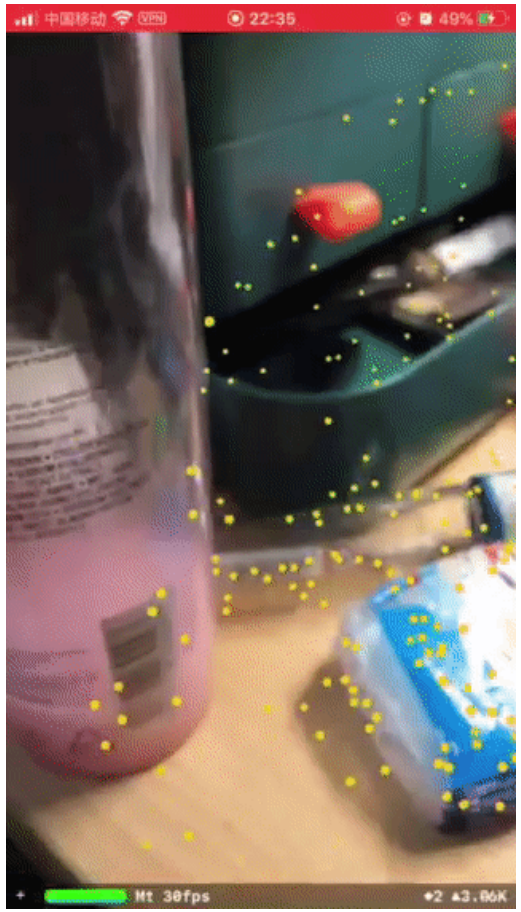
- Find a plane in the environment and place a textbox on it



Steps of Implementation

Step 3
Integrate AR with bar code scanning

- Scan the bar code and place the readings onto a 3D window



Steps of Implementation

Step 4
Integrate AR, bar code scanning and data retrieval

- Scan the bar code, retrive corresponding data from the backend server, and display the information onto a 3D window

Steps of Implementation

Step 5
Recognize devices in AR

- Recognize a device and mark it with a red rectangle in the AR interface

Steps of Implementation

Step 6
Recognize racks / cabinets in AR

Still in progress

- Recognize a rack / cabinet, mark it with a blue rectangle and display its corresponding information in the AR interface

Current performance

	Current performance	Requirement
Barcode localization & identification	≈ 1.5s (may occasionally have wrong reading in Android)	< 0.55s
Barcode localization correctness	≈ 98%	> 90%
AR image generation	< 0.1s	< 0.1s
Data retrieval accuracy	100%	> 99%
Frame rate	≈ 60 fps	> 15 fps
temperature	≈ 47 °C	< 40 °C

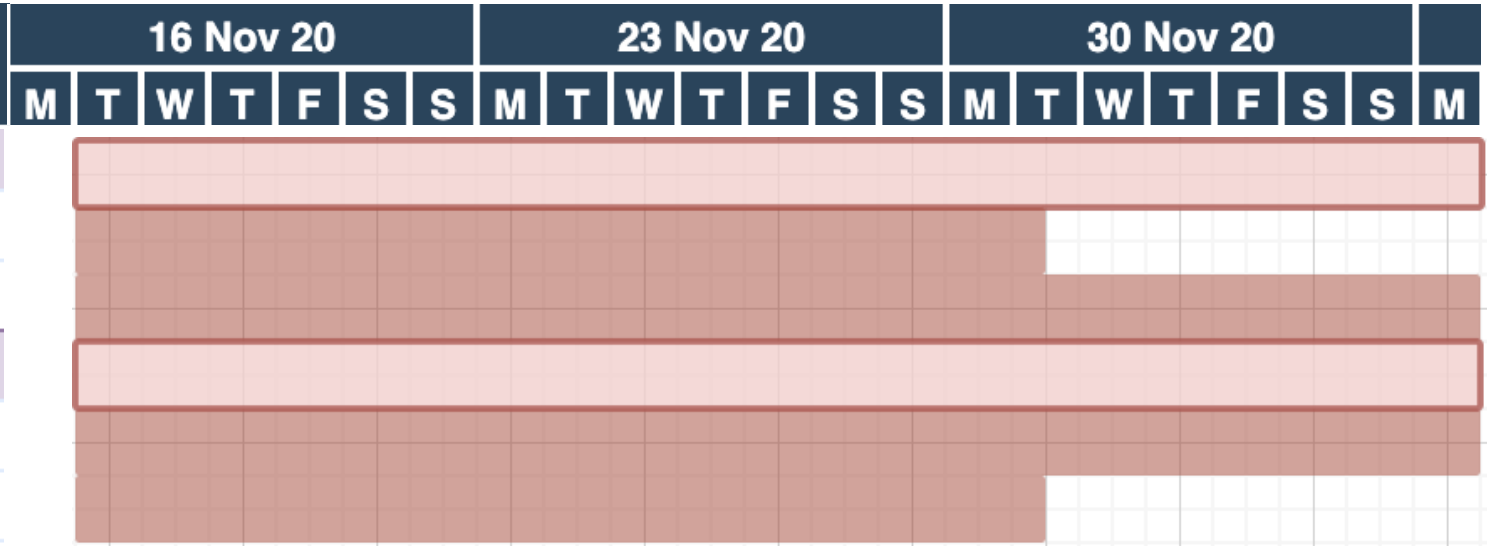
- CPU usgae: 116-134%
- Memory usage: ≈ 245MB

Contents

Future Plan

- **Manufacturing plan**

	Task Name
1.4	Good UX
1.4.1	Adjust the workflow
1.4.2	Be able to freeze the screen
1.5	Improve algorithm
1.5.1	Pause session to save CPU
1.5.2	Multiple cabinets/bar codes



Future Plan

- **Validation plan**
  - **test our program in real data center**
  - **multiple cabinets and multiple server bar code**


Joint Institute
Q & A