# Chapel Checking System RAD and User Manual

Jonathan Manos
Travis Pullen

**Link:** https://docs.google.com/presentation/d/11H6eyJmzBasDiV0p89yt1-F5T5EPhE3-mXhh_2p2sWA/edit?usp=sharing

# Costs of the Current System

- Expensive
- Outdated - over 10 years
- Slow - checkout
- Time-consuming - upload process
- Faulty - crashes
- Password - Single generic password

# Purpose of the New System

- increase the productivity of checkouts at chapel events
- decrease the costs of credit checking equipment
- decrease the work needed to upload information

# Objectives of the New System

1. Sign-in for chapel checker by tapping card to authenticate user
2. Select an event from the current time slot to give credits for
3. Quick processing of students – tap, beep, and go
4. Indication of successful scan with nice beep and code displayed
5. Non-Credit button for late students to block credit for the student
6. Done button to end credit checking for the specific event
7. Automatic syncing and backup of stored student attendance for the chapel event

# Proposed System

- Windows 10 tablet/laptops

- RFID USB plug-in

# Functional Requirements

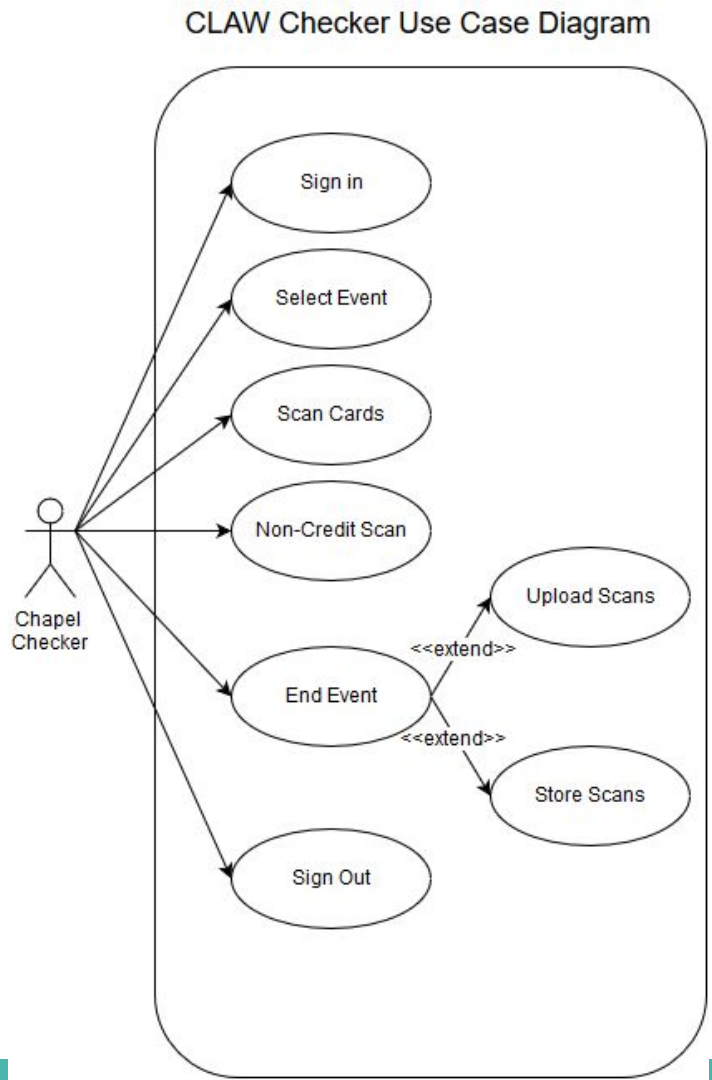
- The chapel checker must be able to log-in to the system in order to access the events page, as well as the scanning window.
- The students should be able to tap their card to the RFID scanner and have the information read correctly.
- After the scanning of the card there should be both a visual and audio confirmation that the card was read.
- The chapel checker should be able to access the blacklist to scan an id
- The “done” button should be able to be accessed by the checker and should also save the file of ids locally
- The files should also be set up to upload automatically to the server

# Nonfunctional Requirements

- Usability - Easy to learn
- Reliability - Reads and stores student information without much error
- Performance - Responsive, around a second to process an ID
- Supportability - Application code will be in possession of CTS
- Implementation - Windows 10 Devices, USB port
- Interface - Access to student ID database and Go.Gordon server
- Packaging - Installed on Windows 10 devices by us and CTS
- Legal - No known liability issues or licensing fees
- Interface - Gordon colors, pleasant scan feedback noise

# Use Case Diagram

Actor: Chapel Checker

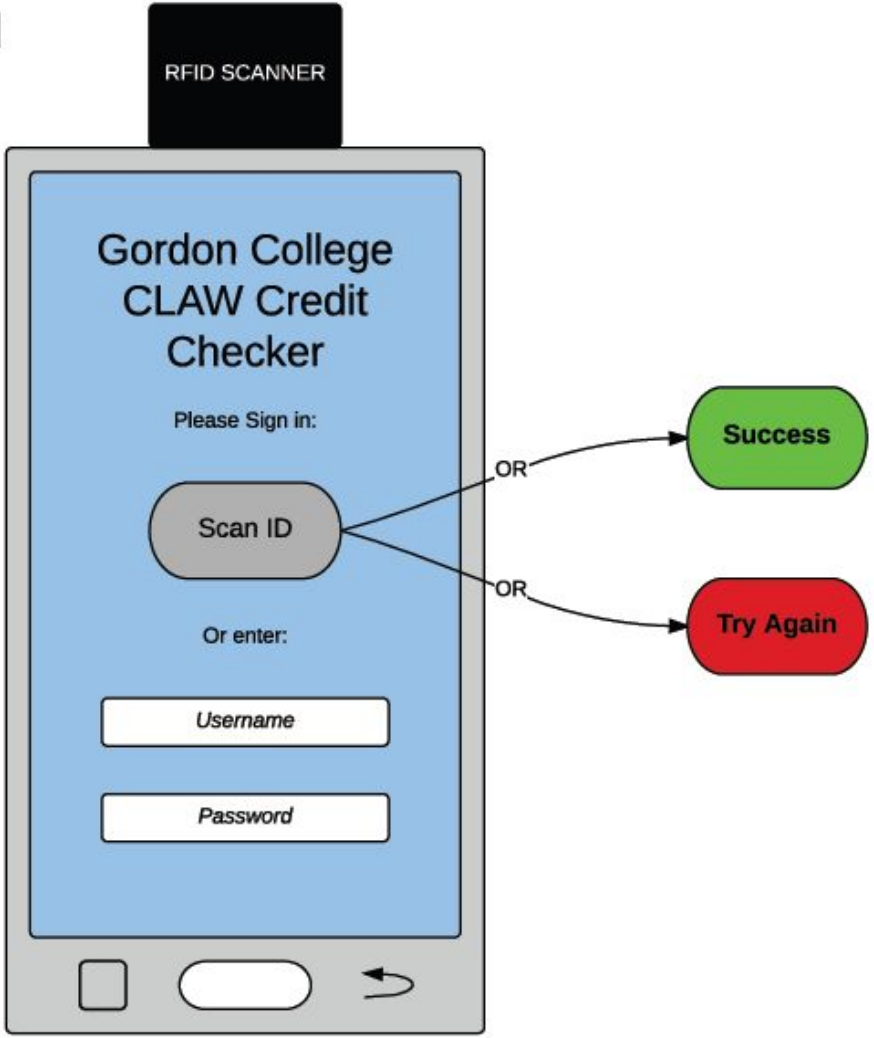


# Scenarios

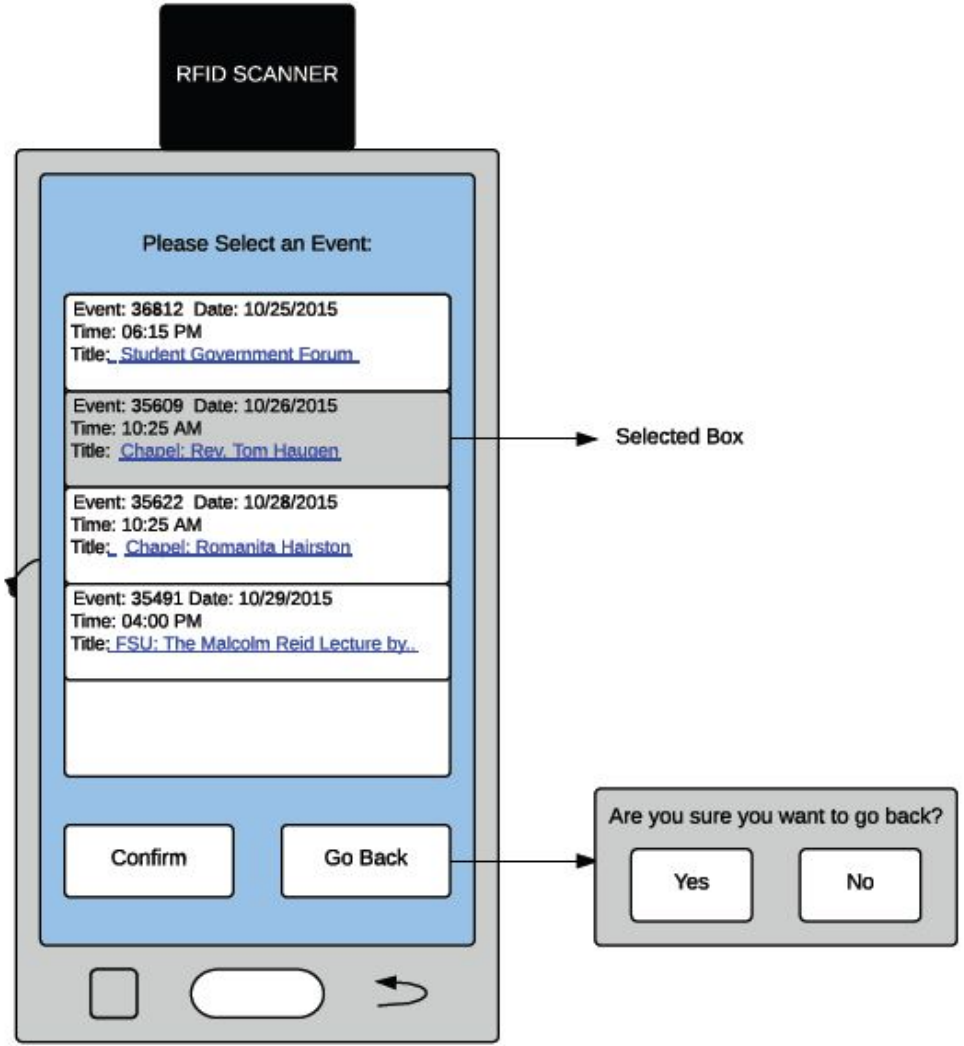
- Sign-in
- Select Event
- Scan Cards
- Non-Credit Scan
- End Event
  - Upload to Server
  - Local Store
- Sign Out

# User Manual Slides

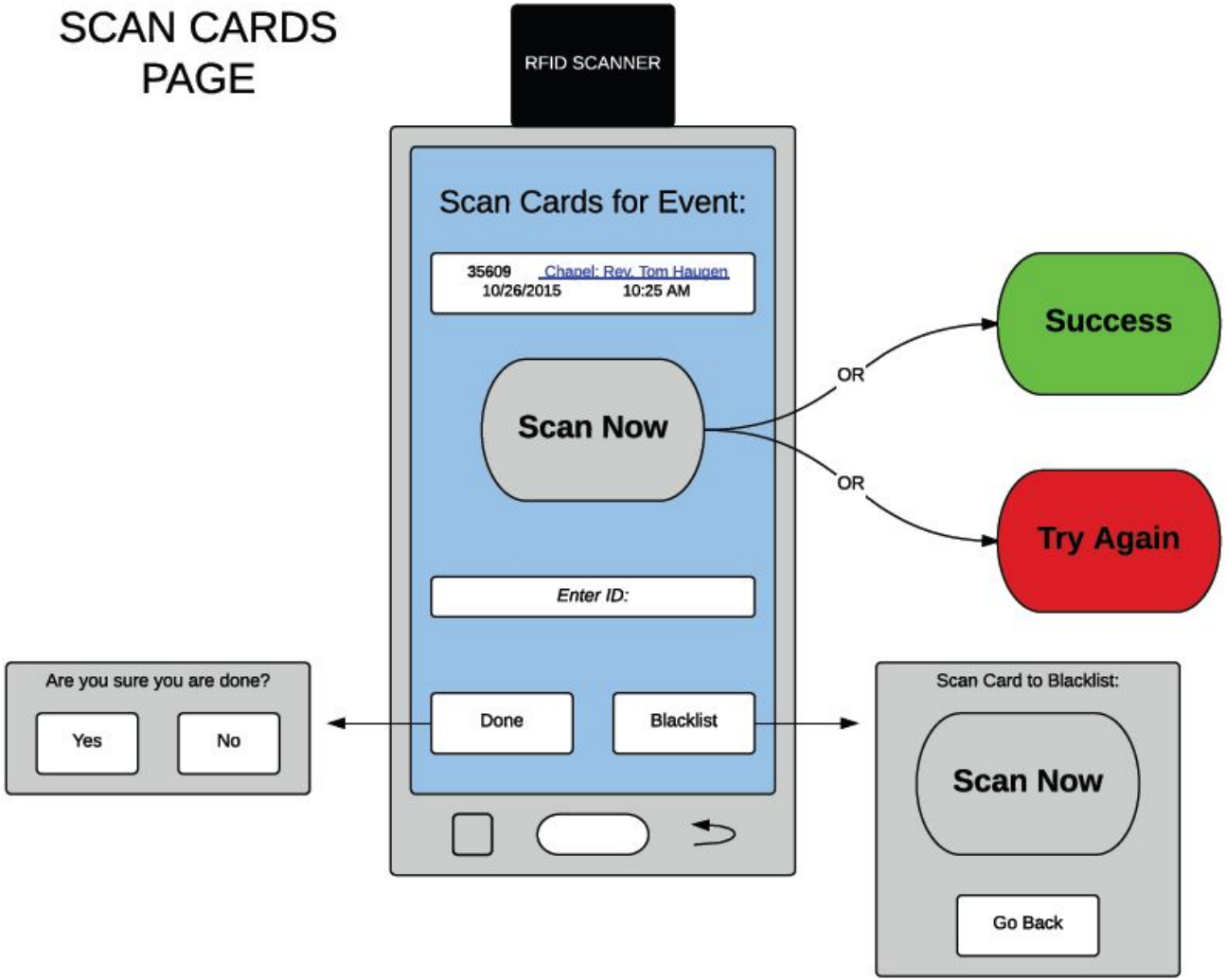
# CHECKER SIGN IN PAGE



# SELECT EVENT PAGE



# SCAN CARDS PAGE



# RESULTS PAGE