user taps <login> button

Knockly

username / email

password

remember me ?

login

Use face ID

Forgot password ?

New user / No account ?

If <remember me> is checked on, the session is extended to 7 days and refreshed every time the app is opened. If a user is inactive for more than a week, the session is signed off and the user needs to login again.

When the user is not logged in, the app attempts to login via the system's face ID automatically. The face ID used is the same as the system and triggers the system's facial recognition, if configured on the system. If not, an error is raised and face ID login is unavailable. If there is a change to the system face ID, user is logged out.

user taps the <forgot password> button

login | Forgot password

example@gmail.com

Enter the username or email of the account you forgot the password of, and went to change

Confirm

user successfully pressed <confirm>

login | Change Password

New password : *

••••••••• medium

must be at least :
- 8 characters
- one letter (a-z/A-Z), number (0-9) and special characters (@_-!?)
- no spaces

Confirm new password : *

••••••••• match

Confirm

The password is dynamically checked for its strength as the user enters it. It has five bands - very weak, weak, medium, strong and very strong. Any password below <medium> band is not accepted and requires re-entering it. A <very weak> password is one that does not follows every requirement, <weak> password only follows the character count, <medium> follows only exactly the requirement with one number and special character, follows the requirement with a mix of valid letters/numbers/special characters and 8 ≤ characters < 12; finally <very strong> is a password over 12 characters.

The <visible> icon is a visibility toggle for the password to switch between an encrypted and plain text appearance. This should NOT affect the backend hashing of passwords within the database.

opens link from auth mail

user tapped the <login> button

user pressed the <create account> button

login | Create an account

Email : *

example@gmail.com

Username (must be unique) : *

example69 ✓ not taken

Dynamically checks for uniqueness for instant feedback while typing

Password : *

••••••••• weak

must be at least :
- 8 characters
- one letter (a-z/A-Z), number (0-9) and special character (@_-!?)
- no spaces

When the user re-types their password, it is dynamically checked to see if it matches the entered password and shows whether it is a match or not.

Confirm password : *

••••••••• match

This is the name displayed to the user in any of their own menus.

Display name (optional) :

example

Create Account

user taps <create account> with valid details

user taps the <confirm> button

Authentication Required

An email has been sent to the entered email account with the next steps to verify your account. Please complete the verification first.

Go back to login screen

Users can only move on from the <authenticate> screen to their next operation through a valid link sent to the email provided. That link should only last for 1 hour (as a session handler). The only other way to exit is through the <login> button on the screen that ignores any previous operations.

opens link from mail, successfully opening a new user session

user pressed the <logout> button

user successfully logged in through credentials or face ID

If a display name is configured, display that instead of username

logout | Welcome <username> | settings

<doorbell #1>

add doorbell

text field shows configured doorbell name

text field updates to configured account name

redirects to user related settings

logout | Welcome <username> | settings

<doorbell #1>

add doorbell

The dotted lines represent the hypothetical design format for the user's doorbell list page, not visible on user end. If the number of doorbells exceed the page, the scrollbar appears on the side, but has the visibility hidden otherwise.

user presses the <user home> button

user presses the <doorbell #1> button

user selects <user settings>

user selects <confirm>, <cancel> OR <user home> in <user settings>

Both these pages are the same "screen" with different backside toggles enabled

redirects to doorbell related settings

* Change to user settings and move doorbell settings to the tab; keep in mind the flows would also need changing compared to this design doc

user home | <doorbell #1> | settings

Tap on the feed to expand it

Live ⊙ | <time stamp> | door status closed

Open | Close

^ Tap to expand ^

Doorbell Options

the door status is a read-only color coded text field that displays whether the door is open or closed (binary)

The time stamp shows the full date, time & time zone in the <dd/mm/yyyy : hh:mm:ss - utc ±x> format

The live feed displayed on this screen is a large interactable UI element that redirects the user to a different screen when tapped directly on, on top of being a live video graphic from the raspberrypi camera

Tapping either the tab or the <doorbell options> area should process the <expand> request

user taps on <tap to minimize> bar

user presses <doorbell options> button or <tap to expand> bar

user returned to <doorbell home>

user opens <doorbell settings>

user returned to <doorbell home>

user taps on the feed

user taps the <back> button

v Tap to minimize v

Manage user profiles | Manage face profiles | Doorbell feed history

Doorbell logs

Only directly clicking the tab completes the <minimize> request. When a <minimize> OR <expand> request is processed, the tab reorganizes the screen's elements directly as an animation to push the corresponding elements. The backend also keeps track of the tab's state (minimized/expanded bool)

All the doorbell option buttons are always on the screen with visibility toggles corresponding with the tab's toggle state. The expanded area is located out of the screen's periphery when in the <minimized> state, and moved up and down, pushing the other on-screen elements.

Usernname is checkeed dynamically for uniqueness as it is entered.

When the user attempts to edit the email field, an alert will pop up to inform the user that they will need to enter their system credentials and will be logged out of their account after changing their email and pressing confirm. If a new email is detected after pressing confirm, a popup should appear informing the user to verify the email via the link sent to the new email. After this popup is closed, the user is logged out and needs to login with the new details after verification. Emails always need to be verified before the user can login to their account.

user home | <username> Settings

User Settings

Username (must be unique):

<current username> ✓ not taken

Display Name

<current display name>

Email

<current email>

Change Password

Login and Two-factor Authentication

Automatically use FaceID on login — ON

Two-factor Authentication via email for login on new devices — ON

Authenticated/Familiar Devices

Samsung Galaxy S21 — Info | Logout | Remove

Confirm | Cancel | Revert Changes

user reverts changes to previous state

Change password opens a new screen a follows the same flow as <forgot password> from the login screen

doorbell home | <doorbell #1> settings

About

Doorbell Name

<current name>

Doorbell Password

<current password>

Notifications and Alerts

Enable notifications — ON

Enable in-app alerts — ON

Send notifications when new user adds this doorbell — ON

Send notifications when a familiar user is allowed into the home — OFF

Send notifications for unfamiliar guests at the door — ON

Send notifications for blocklisted users at the door — OFF

Doorbell Behaviour

Smart doorbell facial recognition features — ON

Confirm | Cancel | Revert Changes

user reverts changes

user selects <confirm> OR <cancel> in the <doorbell settings>

Upon disabling this feature, the doorbell no longer opens the door automatically via the facial recognition, or notify the user when unfamiliar guests are at the door. Instead, alerts are only sent when someone presses the physical button. The door can still be remotely controlled. Send a warning before disabling this feature.

Device Info

Name : owner phone

Model : Samsung Galaxy S21

MAC Address : 59-A1-08-23-4B-A2

IPv4 Address : 6.123.149.108

Last login : dd/mm/yy hh:mm:ss (n days ago)

Close | Logout | Remove Device

Taps the <info> button to open the device info popup for more details

User selects one of the operations

This popup appears on top of the user settings screen

Logout ends the session

If last login was less than a day ago, replace text with n hours/mins/secs as required

Remove ends the session and removes the device from recognised devices. Removed devices need to be reauthenticated if attempting to login. Before successfully removing a device, ask for confirmation first through a warning popup.

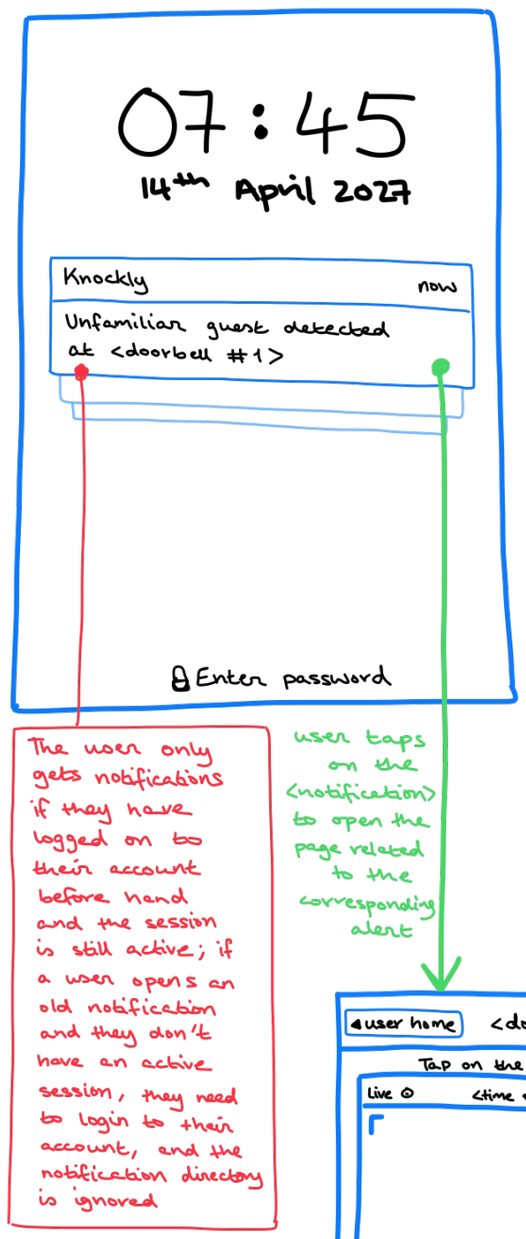
Live ⊙ | <time stamp> | door status open

back

The feed is displayed in full-screen and is not interactable

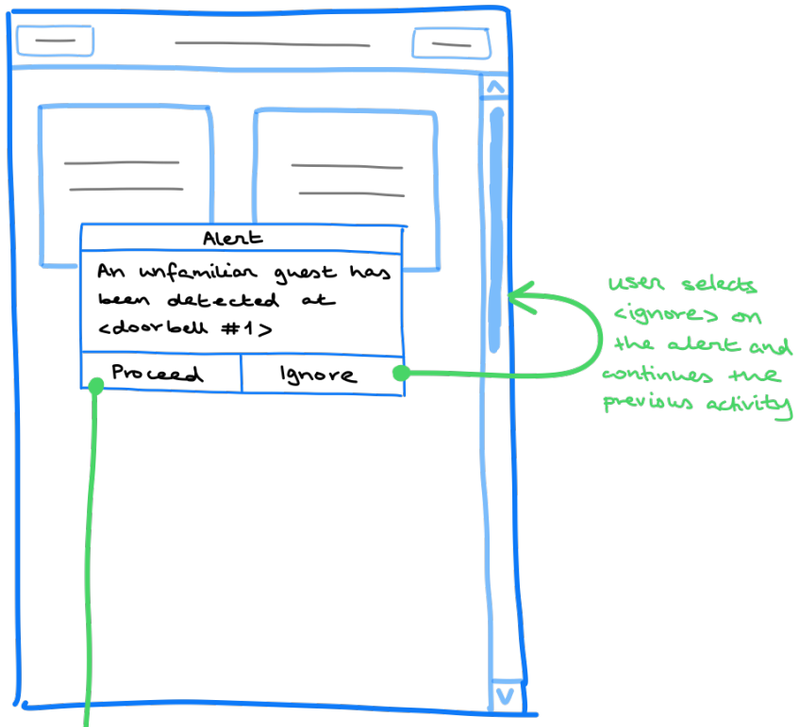
logout
Welcome <username>
settings
<doorbell #1>
add doorbell
user selects <user home>
user taps the <add doorbell> button
user home
Find nearby doorbells
<doorbell #1>
<doorbell #2>
<doorbell #3>
<doorbell #4>
user attempts to add <doorbell #1>
<doorbell #1>
Enter the password for <doorbell #1>
Confirm
Cancel
user presses <confirm> with incorrect password
user presses <cancel> for adding the doorbell
Same as <doorbell settings> from the doorbell home design screens
back
Configure <doorbell #1> settings
Rename doorbell (optional)
<default box name>
Enter new password (optional)
<default box password>
Notifications and Alerts
Enable notifications
Enable in-app alerts
Alert #1
Alert #2
Alert #3
ON
ON
ON
OFF
OFF
doorbell is not configured
Restore to default
Continue ▶
user selects <restore to default>
user presses the <back> button
user selects <continue>
doorbell is already configured
Live ⊙
<time stamp>
door status open
back
(guest view)
back
Setup face Profile
WIP
continue ▶
user presses the <back> button
back
Test camera feed
user will need to manually adjust the camera until they are happy with the feed
user selects <continue>
finish ▶
user selects <finish>
user home
<doorbell #1>
settings
Tap on the feed to expand it
Live ⊙
<time stamp>
door status closed
Open
Close
^ Tap to expand ^
Doorbell Options
(owner view)

Notification and Alert flow examples



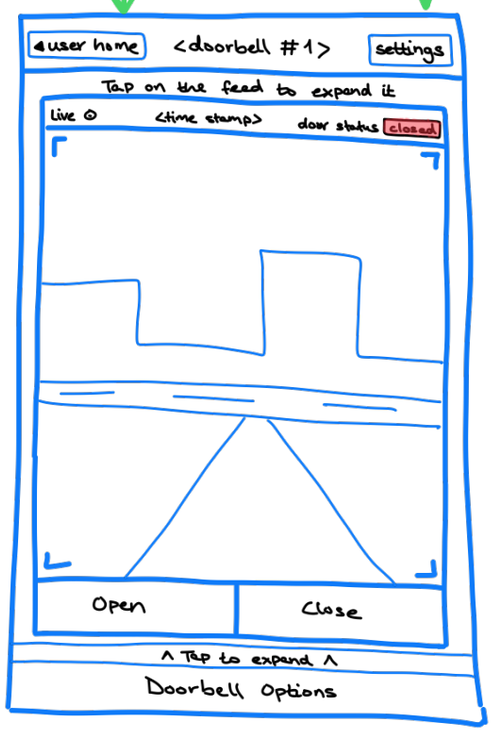
The user only gets notifications if they have logged on to their account before hand and the session is still active; if a user opens an old notification and they don't have an active session, they need to login to their account, and the notification directory is ignored

user taps on the <notification> to open the page related to the corresponding alert



user selects <ignore> on the alert and continues the previous activity

user chooses <proceed> directly to the corresponding page



Types of alerts:

- Facial recognition related alerts (like unfamiliar guest detected at the doorbell)
- User related alerts (new device logged in to your account)
- Doorbell related alerts
  - new user added to doorbell
  - new face profile added to doorbell

The adjust button allows users to drag and drop the users to adjust the positions; it would set each user's button as interactable when in edit mode and reset as uninteractable outside of it.

Dotted lines represent the screen's (hidden) grid design structure

<user home> <doorbell #1> settings

Tap on the feed to expand it

Live ⊙ <time stamp> door status closed

Open | Close

∨ Tap to minimize ∨

Manage user profiles | Manage face profiles | Doorbell feed history

Doorbell logs

user taps the <doorbell home> button

<doorbell home> Manage Users adjust

<username #1> owner settings

<username #2> admin settings

<username #3> guest settings

<username #4> guest settings

user taps the <adjust> button

user taps the <cancel> button

Adjust cancel

Drag and drop to adjust position

<username #1>

<username #2>

<username #3>

<username #4>

<username #5>

user taps the <back> button

<back> <username> permissions

Available Doorbell Options

Permission #1 ON

Permission #2 OFF

Permission #3 ON

These are switches that operate as booleans

Scrollbars should only appear if the number of UI elements exceed the screen

Alerts

Permission #1 OFF

Permission #2 ON

The <role> box is a dropdown that shows all available roles including the currently selected role (highlighted) to choose; the box can be writable but requires input validation. Newly added users default to <guest>. Swapping owners triggers a confirmation popup box.

user selects <confirm>

user selects the <settings> button

user tries to change the current role (dropdown design)

guest

guest

admin

owner

user selects the <manage user profiles> button

Create Roles | Blacklist

user selects <confirm> OR <cancel> on a <delete> operation

user selects <confirm> on a <add to blocklist> operation

Add to Blocklist | Delete User from doorbell | Confirm Changes

user selects <cancel> on an <add to blocklist> operation

user pressed the <back> button

user pressed the <doorbell home> button

user pressed the <back> button

user pressed the <blocklist> button

user pressed the <delete> button

<back> Create Roles <doorbell home>

Role name:

Friend

<Role name> is an input text field

Permissions

Available Doorbell Options

Permission #1 ON

Permission #2 OFF

Permission #3 ON

Alerts

Permission #1 OFF

Permission #2 ON

Managing Options

Permission #1 OFF

Permission #2 OFF

The <permissions> section is designed with a nested grid structure for uniformity in the layout; the grids are hidden on user end

Create Role | Restore to Default

user selects <restore to default>

user selects <create role>

user successfully selects <create role>

<back> Blocklist <doorbell home>

<username #5> move out of blocklist

<username #6> move out of blocklist

<username #7> move out of blocklist

user presses the <delete> button

user presses the <move out of blocklist> button

user selects <confirm> OR <cancel> on a <move out of blocklist> operation

The <username> shown in BOTH the user list and blocklist must be the UNIQUE configured <username> rather than the <display name> (if configured) to avoid confusion between users and to easily differentiate between users.

The <blocklist> blocks users from adding the doorbell to their account centers; blocked users get an error alert when attempting to add the doorbell.

user selects <add to blocklist>

Alert

Are you sure you want to [complete user action that triggered this warning]?

Confirm | Cancel

When a confirmation popup appears, the background becomes darker and cannot be interacted with until the popup is closed and the operation is complete

user home
<doorbell #1>
settings
Tap on the feed to expand it
live
<time stamp>
door status closed
Open
Close
Tap to minimize
Manage user profiles
Manage face profiles
Doorbell feed history
Doorbell logs
user selects <doorbell home>
doorbell home
Manage face Profiles
Add new profile
<owner profile>
Rename
<guest profile>
Rename
Move to blocklist
Delete
user selects <manage face profiles>
user selects <confirm> OR <cancel> for completing <move to blocklist>
Blocklist
user selects <move to blocklist>
user selects <confirm> OR <cancel> for profile deletion
user selects <delete> for <guest>
user selects <manage face profiles> to go back
user selects <add new profile>
user selects <confirm> OR <cancel>
user taps <rename> for <owner>
<new profile name>
confirm
cancel
The box changes into this fragment which includes a writable text-box when rename is selected for a profile.
user taps <rename> for <blocked user>
user selects <finish> on face profile setup
manage face profiles
New face Profile
Name the new face profile
<admin profile>
continue
user selects <back> to make changes or go back to previous screens
user presses <continue> on the face profile setup
back
Scan face Profile
WIP
finish
back
Blocklist
<blocked profile #1>
Rename
Move out of blocklist
Delete
<blocked profile #2>
Rename
Move out of blocklist
Delete
user selects <move out of blocklist> for <blocked profile #1>
user selects <confirm> OR <cancel> for <move out of blocklist> operation
user selects <delete> for <profile #2>
user selects <confirm> OR <cancel> for the <delete> operation
user selects <confirm> OR <cancel> for the rename operation
Alert
Are you sure you want to [complete the action that triggered this warning]?
Confirm
Cancel
The blocklist for face profiles affects the facial recognition; when a blocked profile is detected by the door, an alert is triggered and the buzzer is also triggered (for 5 seconds)

The sync function should check the footage on the pi, compare it to the videos stored app side, and only upload videos that are NOT on the app onto the app side. Each video should have a unique ID within the metadata that can be used to identify the videos. The sync would then start from the OLDEST videos on the app side stack and compare the IDs with the pi's video ID for matches. For all the missing video IDs on the app compared to the pi, ONLY download those videos. A more in-depth explanation of the sync algorithm is in another section.

user selects <cancel> and interrupts the sync process; no changes are made

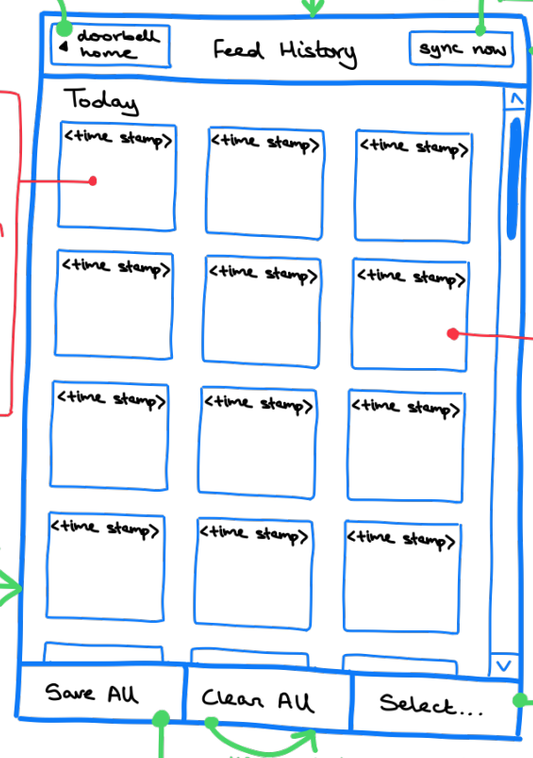
Syncing...
Sync in progress
40%
cancel | close

user selects <close> after the sync is successfully completed

user selects <sync now>

user presses the <doorbell home> button

user presses the <doorbell feed history> button

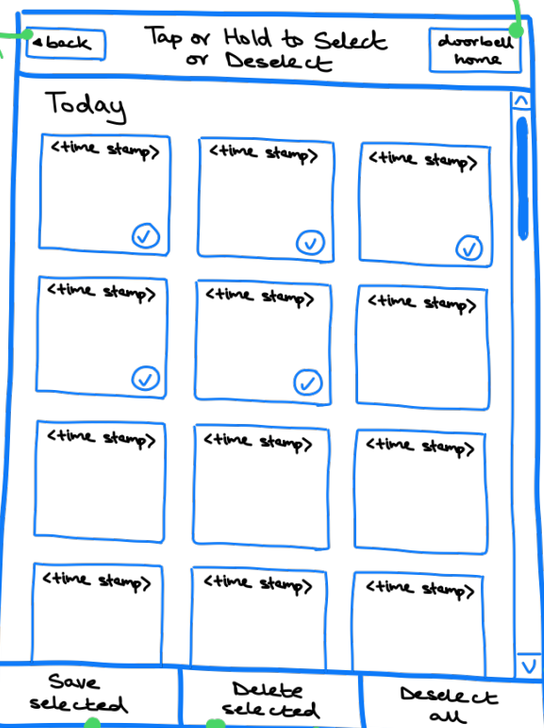


user selects <clear all>

Every video has metadata - it should at least include a unique ID for identification and the time stamp for the starting time of recording; this is stored for easier processing

The feed history stores video every 5 minutes. Each video is therefore 5 minutes long and stores up to 72 hours of footage. Footage older than 72 hours is automatically deleted. This is done to optimize space so that the videos don't take long to upload or download. This can be optimized by implementing an arraylist that deletes the oldest video every time a new video is added (to the front of the list/ latest element). Old videos can also be deleted by comparing the time stamp of the newest video and deleting the videos with time stamps older than 72 hours.

user taps the <back> button

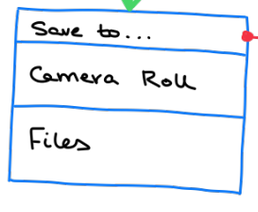


user chooses <select>

user chooses <delete>

user selects <deselect all>

user selects <save>



The <save to> dropdown appears when users try to save footage. The options include the camera roll (requires access) or files (also requires access), but can add more location options or option to share (for example via email) as well if needed. The dropdown appears on the same screen above the bottom task bar.

Sync function flow:

1. Pause video upload to the pi's video database and have every video sent to a separate buffer location to be uploaded while the sync occurs between the pi and the app
2. Find oldest video's ID on the pi
3. Start at the bottom of the app's video list and compare the video IDs with the pi's video ID from step 2 until a match is found (oldest - newest)

4a. No match is found and all videos from the pi need to be downloaded; optionally delete all the app videos if any since they would be older than 72 hours [end of this scenario]

4b. A match is found on the app side and it is designated the current oldest video on the app end; optionally delete the older videos from this point

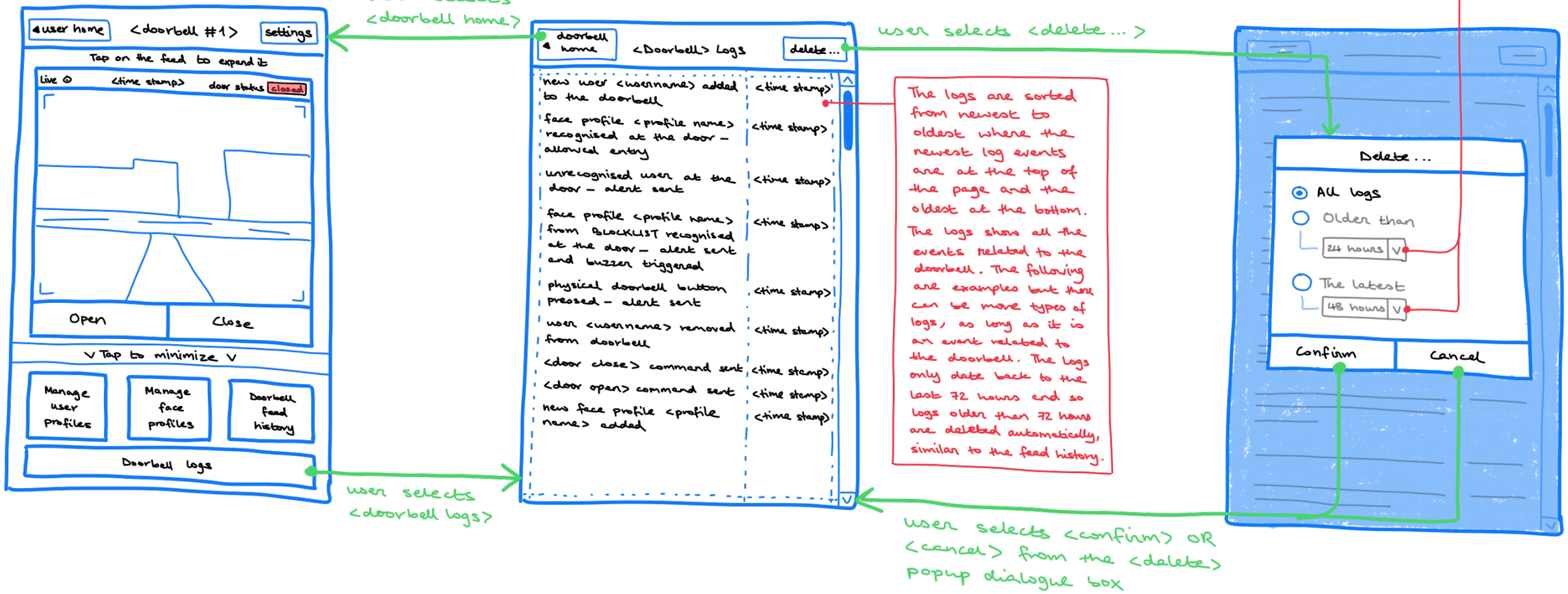
5. Compare the video ID of the next video on the app end to the ID of the next video on the pi to check for matches

6a. It matches and the same is repeated for the next videos (so repeating steps 5 and 6a in a loop till all the videos app side are covered); all the videos are synced and only the videos beyond the newest video on app side need to be downloaded [end of this scenario]

6b. The next video's ID app side from step 5 and the next video's ID pi side do NOT match; store the app's video ID from step 5 and then go through the rest of the pi's video IDs until a match is found, storing the IDs of all the videos that do not match from the pi's side and storing the index of the pi's video ID match

7. When the match is found, get the ID of the next video from the app side and then repeat step 6b - start the pi's video comparison from the index stored at the end of 6b

8. Now there should be a list that stores all the IDs of the videos on the pi that are missing from the app; download ONLY these video IDs on the app [end of this scenario]


user home
<doorbell #1>
settings
Tap on the feed to expand it
Live
<time stamp>
door status closed
Open
Close
Tap to minimize
Manage user profiles
Manage face profiles
Doorbell feed history
Doorbell logs
user selects <doorbell home>
doorbell home
<Doorbell> Logs
delete...
new user <username> added to the doorbell
face profile <profile name> recognised at the door - allowed entry
unrecognised user at the door - alert sent
face profile <profile name> from BLOCKLIST recognised at the door - alert sent and buzzer triggered
physical doorbell button pressed - alert sent
user <username> removed from doorbell
<door close> command sent
<door open> command sent
new face profile <profile name> added
<time stamp>
user selects <doorbell logs>
user selects <delete...>
The logs are sorted from newest to oldest where the newest log events are at the top of the page and the oldest at the bottom.
The logs show all the events related to the doorbell. The following are examples but there can be more types of logs, as long as it is an event related to the doorbell. The logs only date back to the last 72 hours and so logs older than 72 hours are deleted automatically, similar to the feed history.
The options for the time duration dropdown are the following - 30 minutes, 1 hour, 2 hours, 4 hours, 8 hours, 12 hours, 24 hours, 48 hours and 72 hours
Delete...
All logs
Older than
24 hours
The latest
48 hours
Confirm
Cancel
user selects <confirm> OR <cancel> from the <delete> popup dialogue box