**A functional requirement describes *what* a software system should do, while non-functional requirements place constraints on *how* the system will do so.**

|  |  |  |
| --- | --- | --- |
| **Date Modified** | **Change made** | **Author** |
| 10/11/18 | Created the document | Philip Bedward |
| 11/01/2018 | Added a little more granularity to Functional Requirements based on feedback from Kal and Nagesh | Matt Dunn |
| 11/6/2018 | Changed Domain Administrator to Domain Owner to tie this closer to “client account” | Matt Dunn |
| 11/7/2018 | Moved some functional to non-functional requirements and added Vocab to further define what Domain and Section are | Matt Dunn |
| 11/15/2018 | Changed “Domain Owner” to “Client Account Owner” | Matt Dunn |
| 01/22/2019 | Updated formatting, moved many requirements to Functional instead of Non-Functional, Organized and renumbered requirements | Matt Dunn |

In our application, we will be developing three major components of software: a web application, a smart TV application, and cloud infrastructure. These requirements will be split into their application as of right now to make more sense of the requirements, however, this may change in the future when more of our questions have been answered.

## Vocab

**Client Account** - Refers to all entities managed under a client account. This includes all devices, content, and users assigned to it.

**Section** - A subset of devices within a Client Account

**“Their Scope”** - If it is a section admin, their assigned section. If it is a Client Account Owner, all sections within that client account

The web app will support the following roles:

1. **System Administrator** - A user in charge of the entire system; every Client Account, device, user, etc. Essentially IT
2. **Client Account Administrator** - A “root” user in charge of a client account
3. **Section Administrator** - Someone who manages a subset of devices or sections within a Client, but does not have access to everything within it
4. **Unauthenticated user** - An end user who does not have access to the internals of the system, only knows that content can be requested, displayed and scanned

## Front End

### Web App

#### Functional Requirements

1. System Admin
   1. Log in to the InfoKiosk system
   2. Add a Client Account Owner
      1. Link a Client Account Owner to their respective Client Account
   3. View all Client Accounts and their details
      1. This includes Client Account Owners, Sections, Section Admins and Devices
   4. View all existing devices (Registered to the Client Accounts and their respective Sections as well as unassigned)
2. Client Account Administrator **ONLY**
   1. Add other Client Account Owners to the Client Account
      1. Sends an invitation to join the Client Account
   2. Add Sections
   3. Add Section Administrators
      1. Sends an invitation to join the Client Account
   4. Assign a Smart TV to a Section
   5. Able to accept a Smart TV Registration request
   6. Ability to edit client account-wide settings
   7. Ability to view all sections within a Client Account
3. Client Account Administrator **AND** Section Admin
   1. Log In to their account
   2. View content requests for their scope
   3. Can accept or deny open content requests within their scope
   4. Create content for a smart TV device for their scope
      1. What the content form looks like for a smart TV device (subject, message, content)
      2. What is being uploaded / what is the content type (PDFs, images, links, etc)
   5. View all content available to be pushed to smart tv devices for their scope
   6. Push content to a smart tv device within their scope
   7. View all content currently displayed on any particular smart TV device for their scope
   8. View available Devices for each Client Account or section, given their scope
   9. Access their own account settings
      1. Change Password
   10. Request to register a smart TV device in the web app
   11. Ability to join the Client Account via invitation from an existing user
4. Unauthenticated User
   1. Ability to submit a content request to be pushed to a specific device
   2. Content Request must be made for a specific Client Account
   3. Should not be able to “cross-post” to a Client Account they can not physically view; they should only be able to request to Client Accounts they have knowledge of
5. The web app should be able to handle at least one user of each role; System Administrator, Client Account Administrator, Section Administrator
6. The web app should be simple and intuitive
   1. It should take only two tries at most for the user to successfully accomplish their desired task
7. The web app should not reveal sensitive information
8. The web app should be secure and only allow each user role to access their authorized functionality
   1. E.g An end-user shouldn’t be able to add a Client Account
9. Publishing content to a device or section should only send the content to the selected device or devices within the section
10. The web app should be providing responses, even when the application isn’t available
    1. Eg. A response is sent that the application is unavailable instead of an error message

#### Non-functional Requirements

1. The web app should utilize AWS Cognito for account management and authentication
2. The web app should utilize AWS API Gateway and AWS lambda for endpoint functionality
3. The web app should utilize AWS IoT or a message broker to publish messages to smart TV devices
   1. The web app will use a publish/subscribe pattern to send data to Smart TV devices

### Smart TV App

#### Functional Requirements

1. The smart tv app displays content in the form of cards in a tile-like layout
2. The number of cards displayed on the smart tv will be based on the size of the screen based on “real estate”
3. The TV will not “accept” any more cards once it is full, based on this real estate
4. Each card face contains a title, short description as well as a QR code
5. QR code must be authentic/scannable
6. The QR code on a card redirects you to a web page, email address, or other embeddable media upon scanning
7. New cards will be added as they are pushed from the web app
8. Information for a specific card will update if the information is changed on the web app
9. When the card expires, it will be removed from the bulletin board and an open slot will be available to fill with new content
10. Content must have an expiration datetime
11. There must be a link or QR code on the Smart TV screen that links to the Web App in order to submit a content request
12. Registered smart-tv devices must have unique ids
13. A Smart TV will have the ability to be added to a Client Account and Section within that Client Account via the Web App
14. Smart Tv app will have a link to submit a registration request for that TV
15. Users are notified when the connection is interrupted
16. The TV App must have at least 2 CSS/styling sheets available to choose from
17. The updates to the smart tv should be seamless and as real-time as possible
18. Card content should remain on the page, uninterrupted
19. A user should take at most 3 seconds to understand that a QR code needs to be scanned
20. If a SmartTV is not supported, it will not be able to display content

#### Non-functional Requirements

1. The Smart TV App will be implemented on the Android TV platform
2. The smart TV App will use Amazon IoT services to register the TV
3. The Smart TV App will authenticate using AWS Cognito

## 

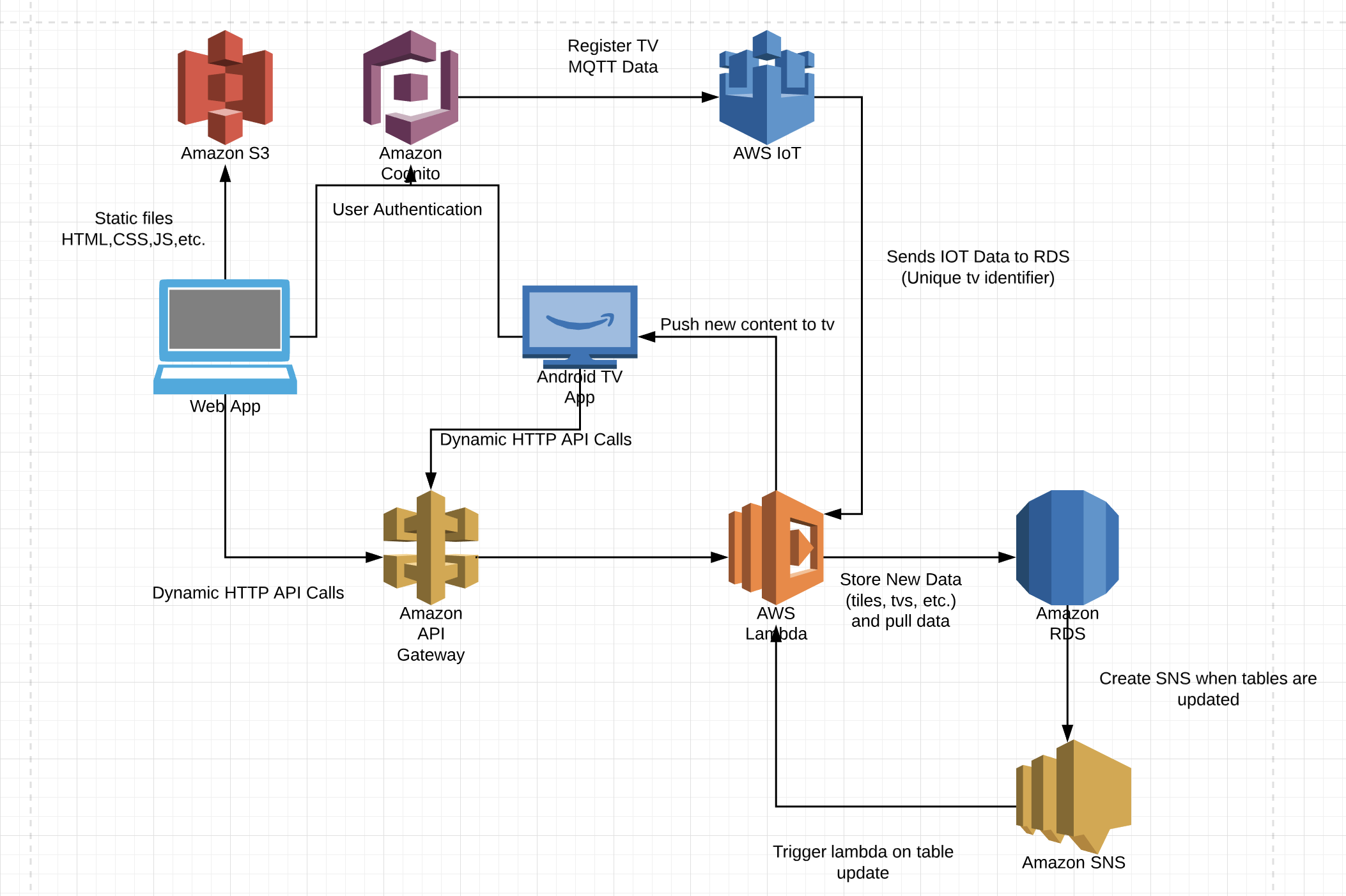
## 

## Back-End

### Infrastructure

#### Non-functional Requirements

1. The static files web app files ( HTML, CSS, JS, Images ) will be stored in S3.
2. Web App and TV data will be stored on RDS.
3. The Web App and Android TV app shall both authorize using AWS Cognito.
4. All Infrastructure shall be created via AWS service provider scripts.
5. RDS shall automatically trigger Lambdas when information is updated.
6. The Android TV App will register TV’s to IoT using the AWS Mobile SDK and MQTT data.
7. The TV will request data from the RDS Table on start.
8. The RDS will push any new data to the target TV.
9. The Web App will push data to RDS using API gateway and lambda.



New requirements?

* Ability to change user password for all accounts from within the app
  + Change display name?
  + Other settings?
* Ability to remove a device from a section
* Ability to remove a device from a Client Account
* Ability to manually remove content from a specific Smart TV
* Ability to edit content
* Ability to delete content
* Ability to name a device from within the app?
* Ability to change a Smart TV’s Section