**MuAPP**

It is a very basic and simple dictation system which will record dictation on a mobile device and using web services audio files will be uploaded to and FTP server. Upon upload files will be aligned on the server and allocated to the respective transcriber. Every transcriber after transcription will forward that file to the next level for Quality check. There would be three levels of Quality check (transcription level included). Once text file is check and ready to be delivered to the client QA level 3 will deliver the file to the client. Where client will have the access to actually review the file and make changes accordingly. At the same point doctors can mark their transcribed files to be complete and signoff which than can be printed out by the practice staff.

**Modules needed for this MuApp**

***Mobile Application***

* Simple mobile app need to authenticate login from the server.
* Successful login will take the user to the first screen where they can choose to dictate or sync.
* Once selected dictate it will take user to the demographic screen where patient demographic can be entered. Along with the selection of Document type (appointment type).
* Once demographic is entered user will hit next and go to recording page.
* Dictator will dictate as per document type selected.
* Once done with dictation will sync the device with server and audio file will be uploaded to the server.

***Server Application***

Login screen will be the entry point and will check the Credentials, Roles of the user, Permissions assigned and Access granted by admin to user.

There would be 3 main access levels of the system.

1. System Admin (MuApp admin)
2. Transcription Company (Transcribers / Quality Assurance)
3. Practice (End User/ Clients)

Each level would have different sub levels.

**Transcription Company**

1. Tranco Admin
2. Transcribers
3. Quality level 1
4. Quality Level 2

**Practice (End User/ Clients)**

1. Practice Admin (Full Access)
2. Provider (Full Access)
3. Front desk /Staff (Limited Access)
4. Nurse /Staff (Limited Access)
5. Biller /Staff (Limited Access)

These roles are dynamically setup in the database. There need to be just check marks to provide the access desired for the user role.

**Policy Definitions.**

Policy for System Admin:

Policy for Tranco Admin:

Policy for Transcriber:

Policy for QA1:

Policy for QA2:

Policy for Practice Admin:

Policy for Practice Staff:

Policy for Providers:

**Design of the Website**

Pages to be designed:

1. Login
2. Forgot Password
3. Super Admin
   1. Manage Tranco Users(Add, Edit, Delete)
   2. Manage Practice Users (Add, Edit, Delete)
   3. Manage Audio List (Add, Edit, Delete)
   4. Manage Company (Add, Edit, Delete)
   5. Create Audio Reader
   6. Manage Patient
4. Tranco
   1. Tranco Admin
      1. View Activities
      2. Forward Document to QA levels
      3. Revert Document
      4. Forward Document to Practice Admin
   2. QA Levels
      1. Listen Audio and Write to Document
      2. Read and write Forward Document to QA Level 2
      3. Read and Write Forward Document to QA Level Tranco Admin
5. Practice
   1. Practice Admin
      1. Receptionist Authorization
      2. Doctor Authorization
      3. Nurse Authorization
      4. Accountant Authorization
6. Patient
   1. Manage Patient (Create history, Add, Edit, Delete)
7. Document template and Print template control
   1. Document template Maintenance (Add, Edit, Delete)
   2. Print template Maintenance (Add, Edit, Delete)
      1. Header creation
      2. Footer Creation
      3. Provider Signature integration
      4. Defining the page margins
      5. Font size and styling
8. Current Procedural terminology definition
9. Transcription page with Word Editor
10. QA Level Editor page for editing the text.

**Development**

***Platform:***

As most of the activities would be performed on the server side leaving the mobile app light and simpler to use for the providers PHP development is being used. This way we will be able to perform activities and run reports at will.

***Development strategy:***

1. Design and creation of Database.
2. Roles and access definition in database.
3. Document type creation and binding.
4. Audio file naming convention.
5. Pooling strategy to the Transcribers.
6. UI based on rights of the users.
7. UI for workflow and client maintenance.
8. Enabling Web services for data transmission.
9. Template creation and association options.
10. User’s creation and association options.
11. Client creation and role assignments.
12. Enabling Web services

***Development order:***

**Database Creation:**

**Tables to be created:**

* Company: Practices & Tranco client code mapping
* Users: All users of Tranco and practices
* User Rights: listen, Edit, View, view all, sign off, dictate, practice assignment
* Document types: System defined default types. Practice defined types
* Dictating rights: only providers should be the one having these rights
* Audio files with details: Naming Convention, Patient Demographics, provider details, document type, templates assignment.
* Transcribed files:
* Client workflow: Dictation-> File Storage-> Transcription-> QA1-> QA2-> Marked for review-> MD review-> Print & Mail-> Repository custom folders can be added on client side on request.
* Transcriber’s pool: practice assignment to one or more transcribers. Or relating it to the dictating pool of doctors.
* Dictating pool: Assignment of providers to one or to the transcriber’s pool.
* Document Templates: Document types allocated to the practices, providers and type of visits
* Print Templates: Header and Footer of the letter head. Signature of the provider associated to the template & practice.

**Web Services:**

The Web Configuration Service should perform the following steps when the application starts:

* It starts at the application startup time.
* It checks the Documents directory for a copy of the dictated files. If they are found, the web Configuration Service finishes upload process in the background.
* If the Configuration Service does not locate audio files in the Documents directory, the following takes place:
* If the adf-config.xml file indicates that the Configuration Service is not in use, then the Configuration Service copies the managed files from the application bundle to the Documents directory managed folder, and then the Configuration Service finishes its process.
* If the adf-config.xml file indicates that the Configuration Service is in use, then the Configuration Service tries to download the managed files in the order they are listed.
* It checks for stored credentials from the secure store. If the credentials are found and they provide access to the Configuration Server, then the connections.xml file is downloaded and placed in the Documents directory managed folder, and at this point the Configuration Service finishes its process.
* If the credentials are not found or the stored credentials fail to provide access to the Configuration Server, the user is prompted for the connection information (user name, password, and client code).
* The Configuration Service reattempts the prompt or uploads five times before failing and dropping out of the application.
* After a successful connection has been made to the Configuration Server, the connections.xml file is downloaded and the connection information is stored for later use.