Exhibit A

Mobile/Web Application for WiseLabs

Table of Contents

Introduction 3

Purpose 3

Confidentiality of Document 3

Key Terms & Milestones Related to Vendor SOW 3

Milestones/Sprints 4

Scope Details/Functional Requirements 5

Project Resources 11

Appendix A – Legacy Views for Provider+Office Managers 12

# Introduction

## Purpose

This document is intended to define the functional specifications for a new Mobile/Web Application initiative that is underway at WiseLabs (a unit of WiseHoldings Technology, LLC). This document is confidential and may not be shared with any other party without prior approval from WiseLabs. This Statement of Work (SOW) provides sufficient information to allow Vendor a complete understanding of the solution WiseLabs wishes to be created.

## Confidentiality of Document

This document is furnished to Vendor for the sole purpose of enabling them to build the products andservices described herein to WiseLabs. Any disclosure, publication, release or transfer of this document, any provision hereof, or any information contained herein, to any person (other than Vendor’s personnel to whom disclosure is necessary for the preparation of a response, all of whom must be bound to observe this prohibition) without the prior written consent of WiseLabs is prohibited. WiseLabs reserves the right to release or disclose any of the information provided by Vendor to its associated divisions, contractors, or any of its affiliates.

# Key Terms & Milestones Related to Vendor SOW

* The new application will be written to reside in the standard WiseLabs software stack which will use these technologies:
  + Apache 2.x (or similar as needed)
  + Ruby 1.9+ and Rails 3.2+
  + Jquery Mobile 1.1
  + MySQL 5.x (or similar data store)
  + Github Source Control
* Project coordination, documentation and bug tracking will use WiseLabs’s PivotalTracker account and Github’s Wiki.
* WiseLabs Github instance will be leveraged as the source code repository.
* WiseLabs Heroku servers will be leveraged during the process and credentials will be provided.
* The application must leverage the JQM HTML5 framework to not only scale across all mobile devices, but cross-browser (IE 9+, FF 10+, and Chrome 17+) as a web-application.
* All webpages related to appointment booking and post log-in should be SSL encrypted (e.g. whole site will be running as HTTPS).
* While TDD is not a requirement, well documented unit tests, or similar, should be planned for and documented within the code with test coverage ratios of at least 80%. A testing suite must be used.
* Vendor will conduct thorough internal QA testing which is built into the project estimate. We also account for user acceptance testing (UAT) as a series of check points every two weeks.
* The application will need to leverage PhoneGap, or similar, to the submission of the packaged applications to both Apple and Google is a requirement.
* WiseLabs will specify preferred bug reporting and tracking method. Bugs (errors) found with the initial website based on spec’d functionality will be fixed.. A “bug” is defined as something that causes the application to improperly behave, crash, or not function as specified in requirements.
* Resumes of the dedicated resources that will be working on this project must be included. If resources change for any reason during the process, payment may be withheld by WiseLabs. The following resources will be working on this project: Veena Mohan, Sreejesh .K.P, and one senior level technical resource.
* This will be a fixed-bid project and payments will only be made upon milestone delivery.
* The Vendor will be responsible for all releases and deployments until the project is complete.
* WiseLabs to provide hosting contract for development and production environments
* Vendor to setup production deployment strategy and plan for the WiseLabs
* Vendor will provide deployment scripts and execution thereof
* The Vendor will write stories and epics relevant to each sprint and velocity will be tracked on PivotalTracker.

# Milestones/Sprints

Each sprint will be two weeks long and will involve 2 resources developing 40 hours each week). At the end of each two week sprint, a demo of the capabilities will be performed on a production server environment such as staging.sampleurl.com.

|  |  |
| --- | --- |
| Sprint/Milestone 1  (Week 1 & 2) | * Infrastructure Planning and Set Up. * Creation of all Stories and Epics in Pivotal Tracker * Admin Section * Admin Member/Provider Account Management * Admin Configuring Globals * Web UI- Graphics/Screen Designs (default JQM assets to be used) * REST Compliant API * QA/Testing |
| Sprint/Milestone 2  (Week 3 & 4) | * Member/Non-Member/Provider Sign up/In/Out * Member/Non-Member Social Network Integration * Member Account Management * Provider Account Management * Member/Non-Member Profile Management * Listing Hotlines * Listing News Feed * QA/Testing |
| Sprint/Milestone 3  (Week 5 & 6) | * Member Management of Appoinments * Provider Management of Appoinments * Reviews in Facebook/Twitter * Provider Communication with Members * Integrate Mozeo and Sendgrid APIs for Communications * Search Feature * Manage Schedule Configurations * QA/Testing/UAT |
| Sprint/Milestone 4  (Week 7) | * Phonegap Packaging & Delivery to Apple + Play * QA/Testing/UAT |

.

# Scope Details/Functional Requirements

As a whole, the scope is to build a mobile/web application that can serve as a platform for industry specific applications of such a platform. Below are the work efforts included, in their entirety, for the four Sprints listed above.

* Develop a new Mobile/Web application with the following Member-facing features:
  + **Ability to view a Icon-based home page upon startup;**
  + **Ability to search across data stores with a delineated claimed/unclaimed record structure;**
  + **Ability to view a phone list of important numbers;**
  + **Ability to create, view, edit a Member profile;**
  + **Ability to view an RSS feed from a url in a structure fashion;**
  + **Ability to book appointments and manage appointments with Service Providers who have claimed records within the data store;**
  + **Ability to sign-in/out with email or social integration (Facebook and Twitter);**
  + **Ability to write Reviews that will be immediately visible on a Service Providers profile with social integration (Facebook and Twitter).**
* Develop a new Mobile/Web application with the following Service Provider-facing features:
  + **Service Provider Sign Up/In/Out**
  + **Service Provider Account Management**
* Administrative Account Management for WiseLabs for Members and Service Providers.
* Integrate with the existing Client-server APIs for viewing of appointment availability.
* Integrated with external services that the platform will leverage for transaction emails and text messages (Sendgrid API and Mozeo API).
* Provide full technical specification documentation of the entire application.

The specific functionality that the new **Benji platform should provide is listed below** and all of the functionality below is included with the four Sprints above.

* **Non-Member/Member Search** 
  + Ability for a Non-Member/Member to search and view Provider(s)/Facilities
  + Ability for a Non-Member/Member to search and view a News/RSS Feed
  + Ability for a Non-Member/Member to view Hotlines
  + Ability for a Non-Member/Member to create and view their Profile.
* **Non-Member/Member Sign Up/In/Out** 
  + Ability for a Non-Member to create a Member account (from appointment booking process or elsewhere on www.sampleurl.com);
  + Ability for a Member to Sign In/Out out of their Account.
* **Non-Member/Member Appointment Scheduling**
  + Ability for a Non-Member/Member to book an appointment with a Provider/Facility on www.sampleurl.com;
* **Member Account Management**
  + Ability for a Member to manage appointments booked, as well as basic account variables, which will be integrated into the Member’s Profile. A Member should be able to:
    - View Prior and Future Appointments;
    - Adjust a Future Appointment;
    - Cancel a Future Appointment;
    - Confirm a Future Appointment;
    - Remove a Prior/Future Appointment from their Profile;
    - Manage Member-provided information submitted during the sign up and Appointment Scheduling process.
* **Provider + Office Manager Sign Up/In/Out (Provider + Office Manager = Service Provider type)**
  + Ability for an Provider/Office Manager to create an Account;
  + Ability for an Provider/Office Manager to Sign In/Out out of their Account.
* **Provider + Office Manager Account Management**
  + The ability for an Provider/Office Manger to create the business rules dictating how they accept online appointments. (Note: The setup (adding Providers, adding accepted health plans, configuring appointment reasons, creating schedules, adding locations) portion of the application is generally used most heavily just after the signup process, then used lightly thereafter. Also, the ability for the Office Manager to manage 1 or 1,100 Providers is a consideration.)
  + The Provider/Office Manager must have the ability to:
    - Ability to create/manage additional Office Manager users types;
    - Ability to create/claim/edit Providers (Profiles) from the Benji data store;
    - Ability to create/ edit Support Staff for Providers;
    - Ability to communicate pre/post appointment booking information to Non-Members/Members upon time select and post-appointment booking;
    - Ability to associate Providers with Locations and the ability to add/edit/delete Locations;
    - Ability to associate insurance plans with Providers profiles;
    - Ability to edit Provider schedules (by Location, by Provider) and the ability to add recurring and one-time blocks to Provider schedules (by Location, by Provider);
    - Ability to attach Appointment Reasons (Reasons) to Providers and enable/disable Reasons, adjust times for Reasons, adjust rooms by reasons;
    - Ability to configure appointment confirmation notifications by Provider;
  + Providers + Office Managers require a Dashboard (or similar) to view confirmed and non-confirmed appointments by day;
  + Providers + Office Managers require an Activity Feed (or similar) to manage certain front-office action that result from online scheduling, including the ability to;
    - View an appointment request;
    - Accept an appointment request;
    - Decline an appointment request;
  + A messaging framework must be leveraged to facilitate Provider + Office Manager communications with Members.
* **Benji Administrative Account Management (Members, Providers, Office Managers)**
  + Benji-only internal views for account management purposes for Members, Providers, and Office Managers. WiseLabs Admins should have the ability to:
    - Ghost into all account types as a Benji Admin;
    - View a list or all Provider and Office Manager accounts with the ability to configure all Provider/Office Manager account management variables;
    - View a list or all Member accounts with the ability to configure all Member account management variables, as well as Deactivate users;
    - View and edit the Provider + Office Manager Scheduling Configurations.
  + The display of the scheduling view that Non-Members/Members see during the appointment time selection process has different Scheduling Configurations. The Scheduling Configurations vary at the Account level and should be toggle-able by Benji Admin Users at the Account level, as follows:
    - No Online Scheduling – This setting is for Providers that are not premium subscribers to the Benji’s Online Scheduling solution. There will be a default placeholder message or no message at all for these Providers;
    - Full Integration – This setting is for Providers subscribing to the online scheduling service that are leveraging the client server tool for read-only display of appointment availability;
    - Read/Write Integration (Same as Full Integration, but the Activity Feed will be impacted) - This is a placeholder and will function as the Full Integration setting currently;
    - Non-Integrated – This is a Provider/Office Manager account using the system stand-alone and the schedule is being driven by the Provider + Office Manager>Scheduling Account Management function.
  + The display of the scheduling view that Non-Members/Members see during the appointment time selection process has different View Configurations. The View Configurations vary at the Provider level and should be toggle-able by Benji Admin Users at the Provider level, as follows:
    - Appointment Booking Cut-off Days – This variable limits the max number of days from the current date that an appointment can be booked online;
    - Current Day Scheduling Only – This variable limits the max number of days from the current date that an appointment can be booked online to the current day;
    - Public Profile – This variable prevents an Providers profile from being public
    - Jump to Nearest (Non-toggle-able) – This allows the user to jump out to the nearest available appointment time if none are available in the given view;
    - Third-party minimum time delimiter – A time delimiter must be set in either 10 or 15 minute intervals. This time delimiter indicates the manner by-which times are displayed. For example, a Office Manager might set the times to display as 9:00,9:10, 9:20, 9:30…or 9:00, 9:15, 9:30, 9:45…
    - Rolling and Slotted Views – These are the two views that dictate how appointment time slots are displayed to the Non-Member/Member Users. For example, if the Client-server API indicated that there are 90 minutes of availability from 9:00AM to 10:30AM and an Appointment Reason selected requires 60 mins of a Providers time, then the two setting would display times different. The Rolling setting would display: 9:00A, 9:15A, 9:30A; and a Slotted setting would display: 9:00A only (Note: a 15 min delimiter is used in the example). In the Slotted section there need to be another 30 mins of availability (or availability from 9:00A – 11:00A to display both a 9A and 10A).
* **Benji REST Compliant API** 
  + - In order to effectively integrate with existing systems, where needed, WiseLabs /Benji will need a RESTFUL API. All third-party system data will be stored in a small mapping database such as SQLLite on the client’s server and then backed up to a service such as Amazon S3 nightly. This should allow WiseLabs/Benji to just respond with it’s primary keys for each member and appointment request.
    - Member and appointment syncing would happen over standard REST methods and URLS.

**Member API**

|  |  |  |
| --- | --- | --- |
| **HTTP Method** | **API Endpoint** | **Action** |
| POST | /api/v1/member | Create new member |
| PUT | /api/v1/member/:id | Update member |
| DELETE | /api/v1/member/:id | Delete member |
| GET | /api/v1/member/:id | Retrieve member |

The response from each request to WiseLabs/Benji API would then return success or failure.  Errors would be included if the request failed, if it succeeded the primary\_key for the member would then be stored in the local SQLLite database mapping to the primary key in the PMS.

**Appointments API**

|  |  |  |
| --- | --- | --- |
| **HTTP Method** | **API Endpoint** | **Action** |
| POST | /api/v1/appointments | Create new appointment |
| PUT | /api/v1/ appointments/:id | Update appointment |
| DELETE | /api/v1/ appointments/:id | Delete appointment |
| GET | /api/v1/ appointments/:id | Retrieve appointment |

The appointments API would function in a similar manner, after an appointment has been created in WiseLabs/Benji a record is stored in the SQLLite database mapping the PMS appointment to the one stored in WiseLabs/Benji.

**Providers API**

|  |  |  |
| --- | --- | --- |
| **HTTP Method** | **API Endpoint** | **Action** |
| GET | /api/v1/providers | Retrieve all provider data at a location |

**Rooms API**

|  |  |  |
| --- | --- | --- |
| **HTTP Method** | **API Endpoint** | **Action** |
| GET | /api/v1/rooms | Retrieve all room data at a location |

# Project Resources + Infrastructure

|  |  |  |
| --- | --- | --- |
| **Project Resources** | **Description** | **Hours per week** |
| One Senior level technical resource | This person will be primarily responsible for technical and architectural oversight. This person will also contribute at an implementation level as necessary and act as liaison between the WiseLabs team and the Vendors technical and implementation team. | 10 |
| 2 mid to senior level resources with either Ruby on Rails and/or JQM skills | These developers will be responsible for providing implementation expertise and contribute to the architecture discussion. These persons will be hands on, working full time to deliver product. | 80 |

|  |  |
| --- | --- |
| **Infrastructure** |  |
| Hosting Setup and Deployment | * Work with WiseLands to deploy development codebase instances to development and staging servers as required |

# Payment Schedule

|  |  |  |
| --- | --- | --- |
| **Fixed Payments** | **Target Date** | **Amount** |
| Sprint 1 | Start date + two calendar weeks | $2,006.25 |
| Sprint 2 | Start date + four calendar weeks | $2,006.25 |
| Sprint 3 | Start date + six calendar weeks | $2,006.25 |
| Sprint 4 | Start date + eight calendar weeks | $2,006.25 |
| Bonus Payment | This bonus is to be paid directly to the two dedicated engineers on the project and not to be allocated to elsewhere. | $500.00 |
| Total | Target Start Date 5/21/12 | $8,525.00 |

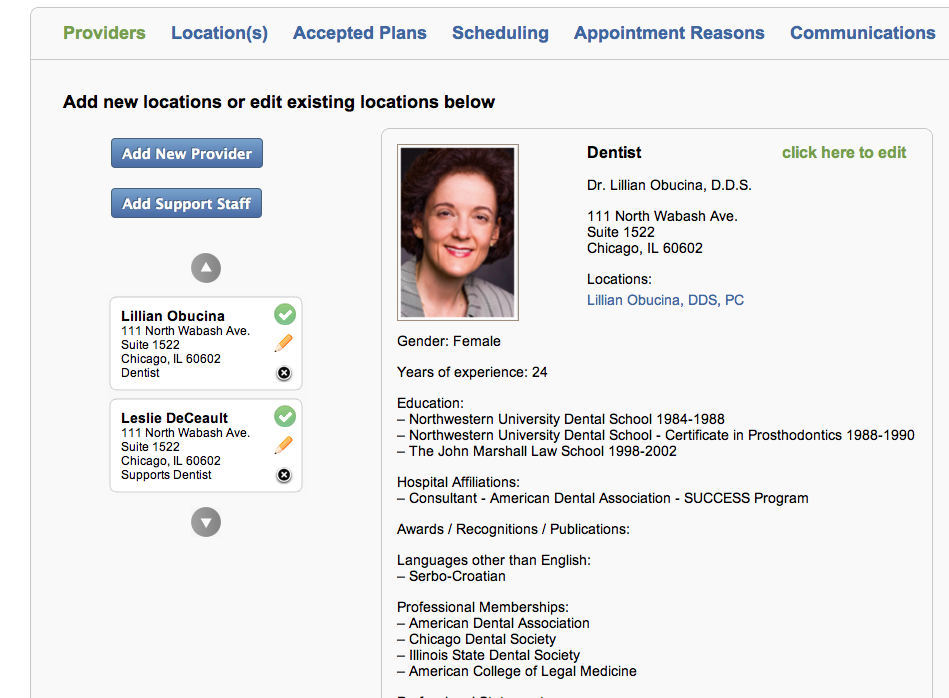
# Appendix A – Legacy Views for Provider+Office Managers

**Account Management (Premium Subscribers)**

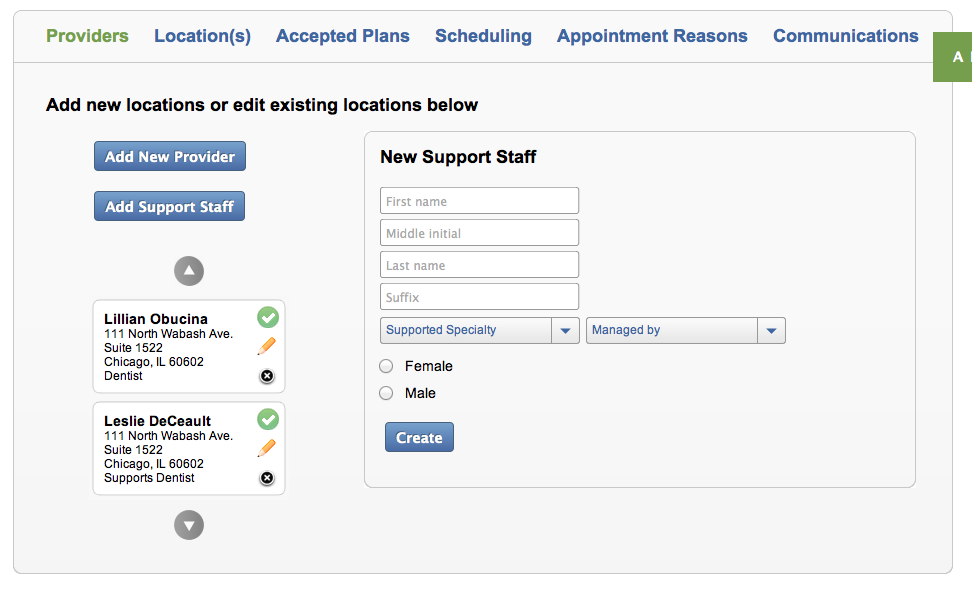
* **Provider + Office Manager Account Management View Examples**

The ability for a Office Manger User (or Provider User) to create the business rules dictating how they accept online appointments: This portion of the application is generally used most heavily just after the signup process, then used lightly thereafter.

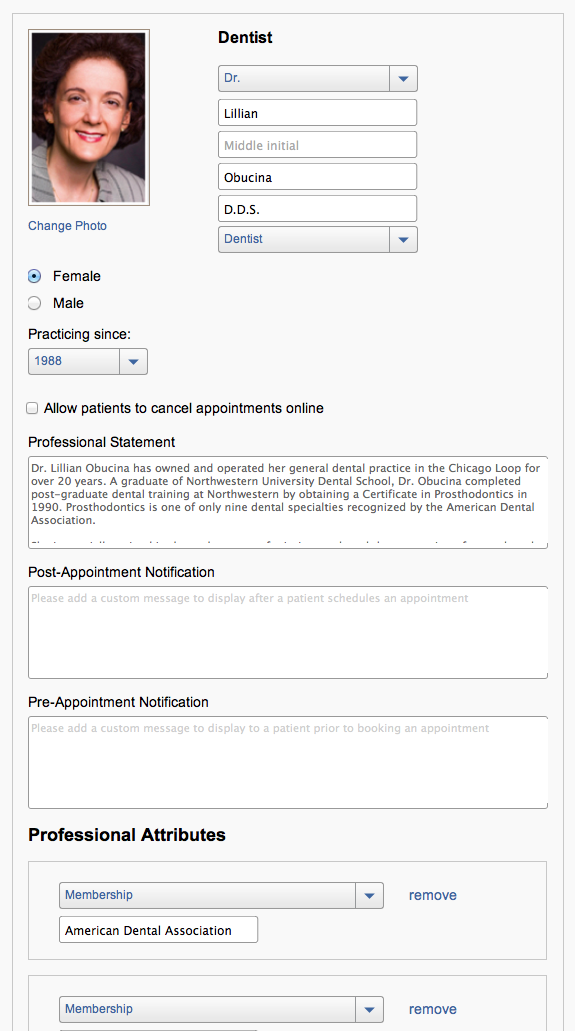
* **Legacy View of Providers Tab.**



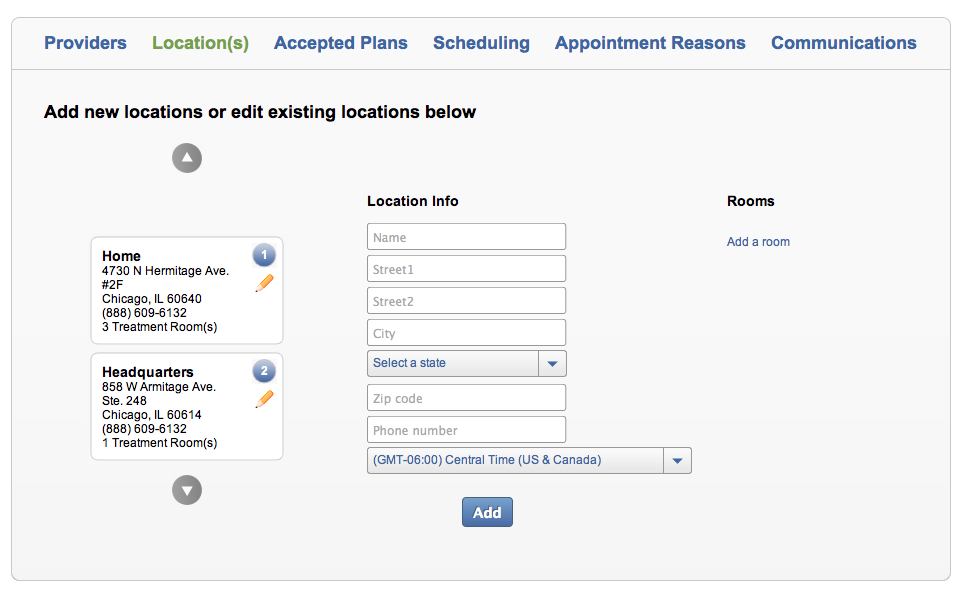
* Providers Tab Detail - Claiming, Creating, Viewing, Removing, and Editing Provider Profiles
  + Add Provider: The Office Manger user must be able to search for Providers across the datastore and Claim a provider to add to their practice.
  + Add Support Staff: Support staff are used for resources that see Members/Patients, but will not have a public profile on the Benji site. These resources are used to create more “inventory” or availability in the Providers schedule. The Support Staff resource always Supports a specific specialty and is Managed by an Provider. See Legacy View below. In the edit view the user can also add a photo.



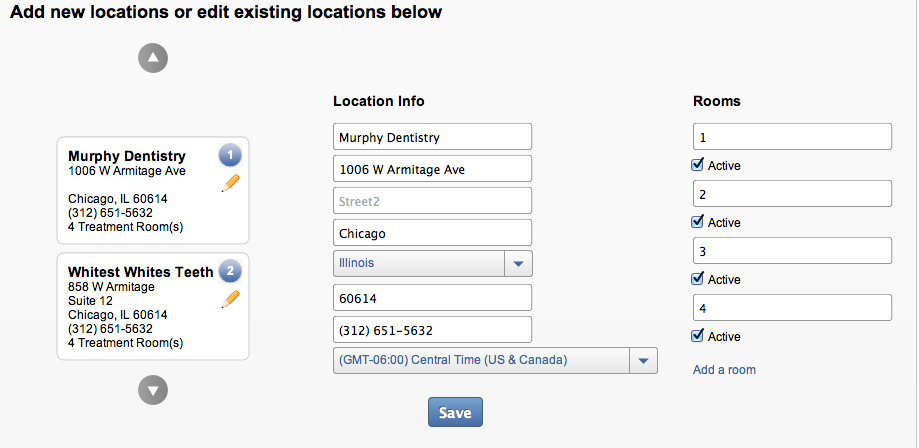
* + Edit: The Office Manager user must have the ability to edit the Provider profiles:
    - Basic Demo Data, Photo (with Uploader and Cropper) Years of Experience, Specialty, and Professional Attributes (Hospital Affiliations, Memberships, Awards, Certifications, etc. NOTE: This must be mapped to Provider Profiles. See Legacy View below.
    - Toggle for online appointment cancelations, by Provider/Provider
    - Pre-Appointment and Post-appointment Booking Notifications. These are messages that appear just after the time (e.g. 1:00P) is selected during the Member/Patient appointment booking process.



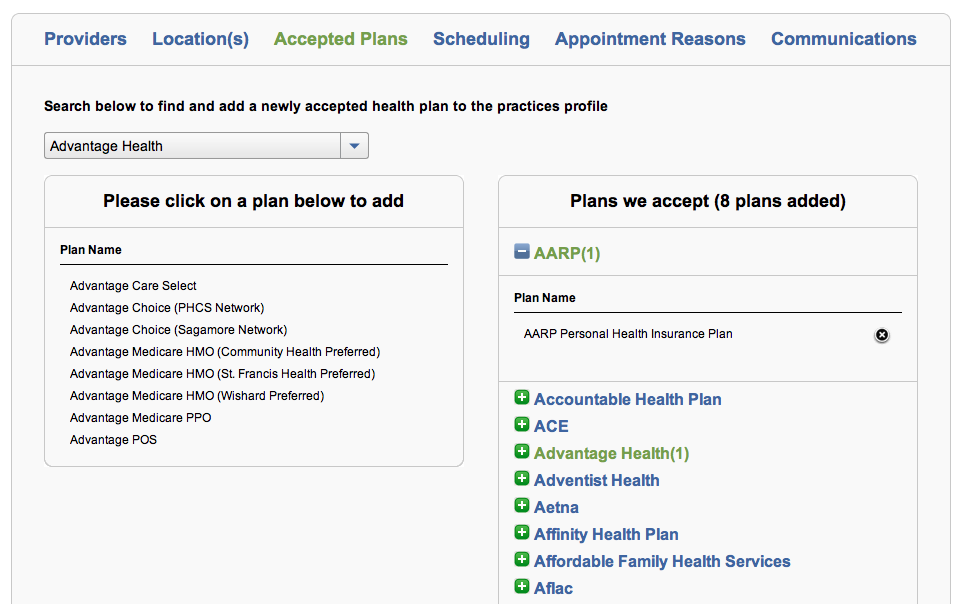
* Locations Tab: The Office Manager has the ability to edit location(s) for the Providers attached to that account. Legacy View below. The timezone would ideally be keyed off the zipcode, or similar. The rooms can be renamed by the user and can also be checked as “Active” If not active, the room is removed from inventory.



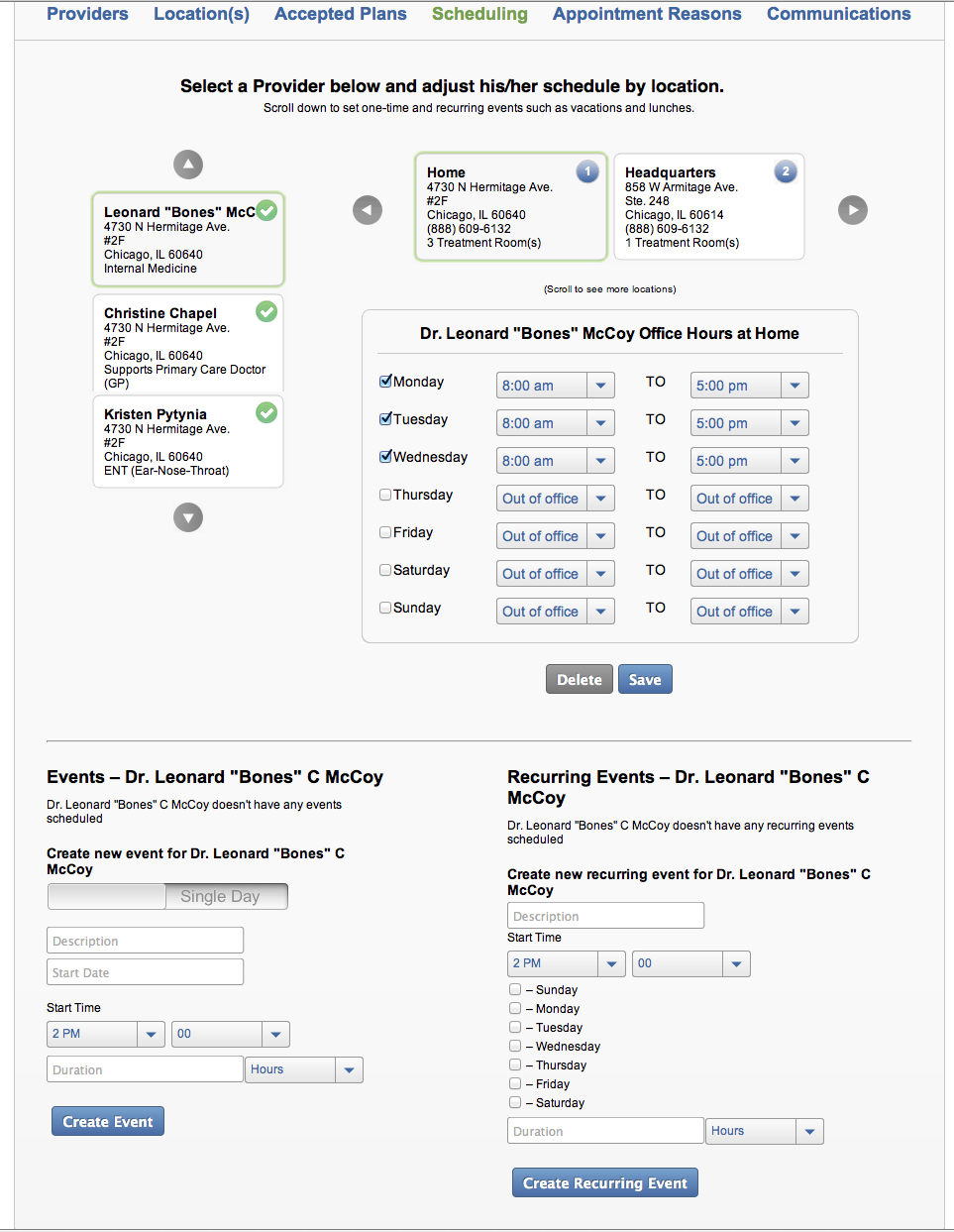
Legacy Edit View



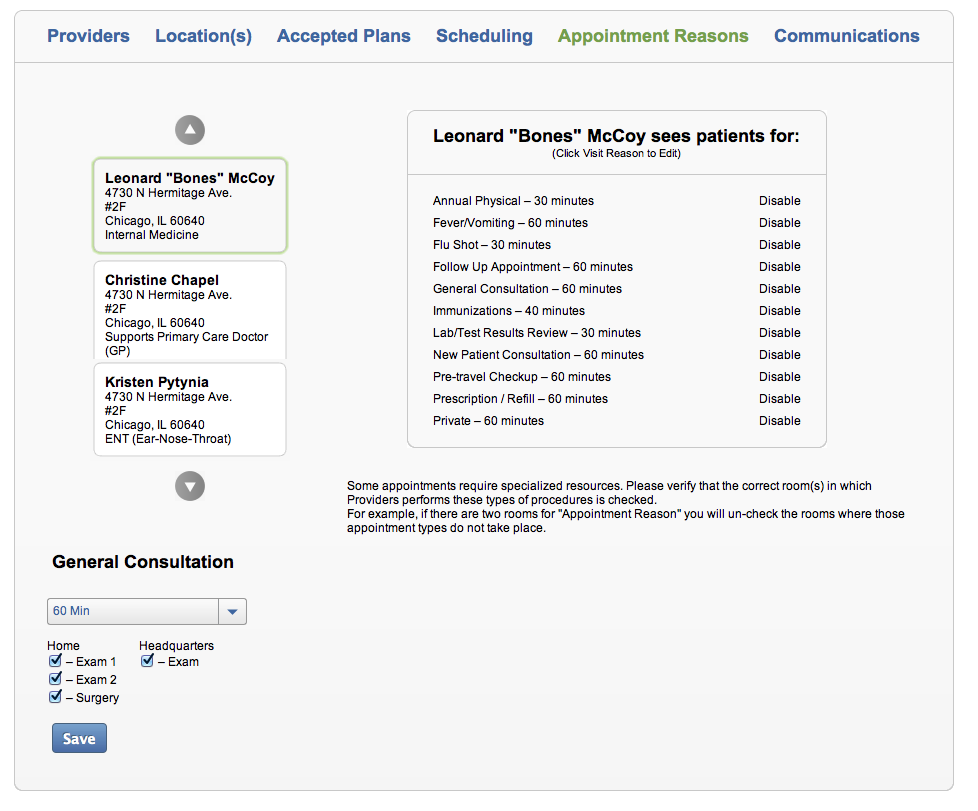
* Accepted Health Plans Tab – This if view/claim/unclaim tab where the Office Manager user can browser Carriers and then the subsequent plans that are accepted by the user. The Legacy view us below. A Master list will be created and maintained.



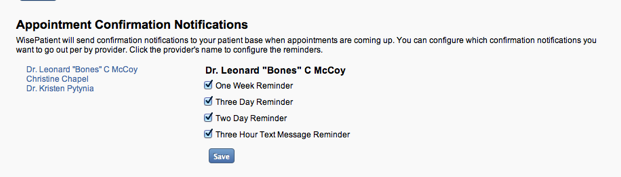
* Scheduling Tab - The Office Manager user has the ability to setup schedules by Provider/Provider by Location.
  + The Legacy view contains the placards for location and Provider/Providers and the resulting editable schedule below. A default 9A – 5P schedule is set for the first location for each provider. The times should be in 10 or 15 min increments (See Time Delimiter in Scope Details) and there should be toggle allowing the user to select what matched their practice style (default to 15). There also must be an indication of whether the Provider if “Out of Office”. An “Apply All” functionality would be required to allow quick/bulk day editing.
  + One Time Events and Recurring Events. The User also has the ability to create one-time and recurring events that will apply “blocks” on availability associated with that Provider./Provider. All of the configurability of such blocks can clearly be seen in the Legacy View below.



* Appointment Reasons
  + The appointment reasons tab allows practice to filter appointment types as it relates to resources (Providers and Staff, and rooms). There is a Master List of Appointment Reasons by Specialty. Each of these reasons can be adjusted by provider, appointments duration, and associated rooms. The appointment reasons itself can be Disabled.  See Legacy View below or CS FSD 3.2.5



* Communications
  + Appointment confirmation notifications can be customized on the communications tab. A list of all of the Providers in the practice can be seen and the types of reminders that are sent out can be configured by Provider. See Legacy View below.



* Bulk Update
  + The ability for an Administrative User (Benji Admin) to bulk upload/edit all of the configuration steps.

