Automating Onboarding

By Gabriel Chandler, Karim Mahfouz and Michael Knudsen

Team Member Information

Michael knudsen

Karim Mahfouz

Gabriel Chandler

Client information

Documentor - Electronic visitation verification

Documentor is a company that provides services to care givers, while integrating with state requirements.

Their services help providers track their time, set goals and makes complying with state regulations easier.

Business Requirements

- *BR1* - **Once a client is ready to be onboarded, the process needs to be automated at every point possible.** - I think this is one of the business requirements because it is almost word for word what we were told what we needed to do. It is a high level declaration of goals that leads to one or more projects to deliver a solution.

Use cases

Actors

- Agent
  - The person on our side of Client On boarding, in charge of sending quickbooks invoice, creating support account, and creating inital access account
- Client
  - Client we wish to add to our system

Use cases

- **id** - **name** - **reason** - **actors** - **flow** - **business req id**
- **UC1** - check invoices - As a documentor agent it is important to be able to view all invoice so you can check for errors or customers who haven't paid - **client** - go to documentor admin panel and quickly view states of invoices - BR1
- **UC2** - begin onboarding process - The process will always start with this step - **Agent** - enter client email into an admin page to being onboarding process - BR1
- **UC3** - receive updates - its important for the Client to know where they are in the process - **Client** - client should receive email updates regarding status of onboarding process - BR1
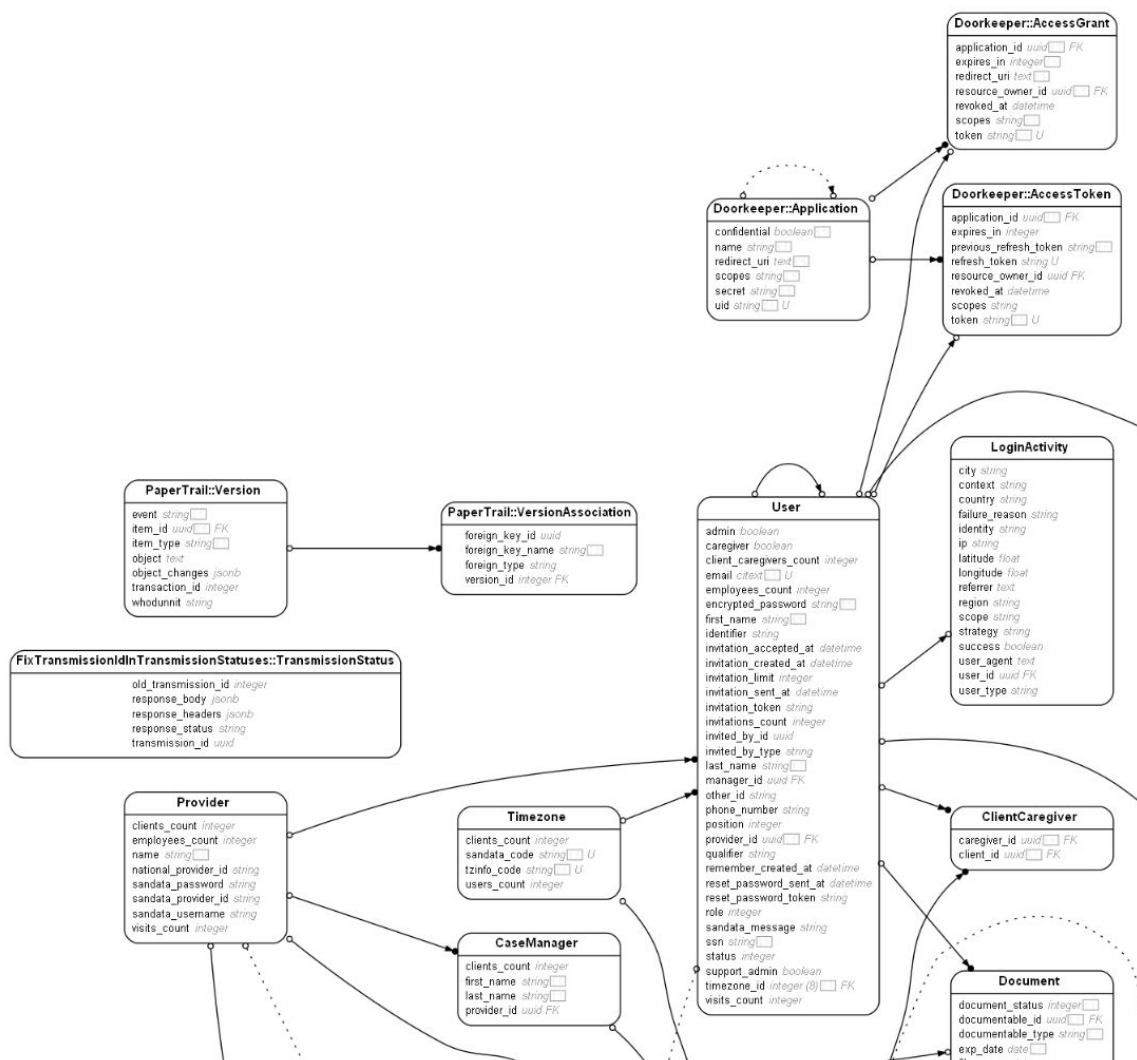
Requirements

Functional requirements

- **FR1** - Automatically create Support account - BR1 - HIGH
- **FR2** - Automatically create initial account - BR1 - HIGH
- **FR3** - Automatically create quickbooks invoice - BR1 - HIGH
- **FR4** - Automatically retrieve billing information from client - BR1 - MEDIUM
- **FR5** - display where in onboarding process a client is - BR1 - MEDIUM

Non-Functional requirements

- **NF1** - View All onboarding invoices - BR1 - MEDIUM
- **NF2** - Sort onboarding invoices - BR1 - MEDIUM
- **NF3** - Log administrative actions - BR1 - LOW
- **NF4** - When an invoice is paid change the invoices display to reflect payment status - BR1 - LOW
- **NF5** - list all clients currently in the inboarding process - BR1 - MEDIUM
- **NF6** - Clients should beable to change their onboarding portal password - BR1 - MEDIUM

Domain Model

Doorkeeper::AccessGrant
- application_id *uuid* FK
- expires_in *integer*
- redirect_uri *text*
- resource_owner_id *uuid* FK
- revoked_at *datetime*
- scopes *string*
- token *string* U

Doorkeeper::AccessToken
- application_id *uuid* FK
- expires_in *integer*
- previous_refresh_token *string*
- refresh_token *string* U
- resource_owner_id *uuid* FK
- revoked_at *datetime*
- scopes *string*
- token *string* U

Doorkeeper::Application
- confidential *boolean*
- name *string*
- redirect_uri *text*
- scopes *string*
- secret *string*
- uid *string* U

LoginActivity
- city *string*
- context *string*
- country *string*
- failure_reason *string*
- identity *string*
- ip *string*
- latitude *float*
- longitude *float*
- referrer *text*
- region *string*
- scope *string*
- strategy *string*
- success *boolean*
- user_agent *text*
- user_id *uuid* FK
- user_type *string*

PaperTrail::Version
- event *string*
- item_id *uuid* FK
- item_type *string*
- object *text*
- object_changes *jsonb*
- transaction_id *integer*
- whodunnit *string*

PaperTrail::VersionAssociation
- foreign_key_id *uuid*
- foreign_key_name *string*
- foreign_type *string*
- version_id *integer* FK

User
- admin *boolean*
- caregiver *boolean*
- client_caregivers_count *integer*
- email *citext* U
- employees_count *integer*
- encrypted_password *string*
- first_name *string*
- identifier *string*
- invitation_accepted_at *datetime*
- invitation_created_at *datetime*
- invitation_limit *integer*
- invitation_sent_at *datetime*
- invitation_token *string*
- invitations_count *integer*
- invited_by_id *uuid*
- invited_by_type *string*
- last_name *string*
- manager_id *uuid* FK
- other_id *string*
- phone_number *string*
- position *integer*
- provider_id *uuid* FK
- qualifier *string*
- remember_created_at *datetime*
- reset_password_sent_at *datetime*
- reset_password_token *string*
- role *integer*
- sandata_message *string*
- ssn *string*
- status *integer*
- support_admin *boolean*
- timezone_id *integer (8)* FK
- visits_count *integer*

FixTransmissionIdInTransmissionStatuses::TransmissionStatus
- old_transmission_id *integer*
- response_body *jsonb*
- response_headers *jsonb*
- response_status *string*
- transmission_id *uuid*

Provider
- clients_count *integer*
- employees_count *integer*
- name *string*
- national_provider_id *string*
- sandata_password *string*
- sandata_provider_id *string*
- sandata_username *string*
- visits_count *integer*

Timezone
- clients_count *integer*
- sandata_code *string* U
- tzinfo_code *string* U
- users_count *integer*

ClientCaregiver
- caregiver_id *uuid* FK
- client_id *uuid* FK

CaseManager
- clients_count *integer*
- first_name *string*
- last_name *string*
- provider_id *uuid* FK

Document
- document_status *integer*
- documentable_id *uuid* FK
- documentable_type *string*
- exp_date *date*

Tech Stack

Ruby on Rails

- We will use Ruby on Rails because we are going to be building onto the clients' already existing code base, which is primarily Ruby on Rails

HTML and CSS

- The client does not use any frameworks or libraries like react; they use plain HTML and CSS, so, we will be using that as well for any front end elements.

AWS

- The details of this are not clear yet, however we know the client uses AWS and may need to use it for secure data storage

Prototype

https://www.figma.com/file/JtbzjFG4QsLgTjNHs5cOAr/capstone-project?node-id=0%3A1

First Iteration Features

Mentor feedback

Suggestion: Switch from simple forum to complex portal

Changes: Switched from simple forum to complex portal

Suggestion: Polish our business requirements

Changes: added more detail to BR

Client feedback

The client agreed with most of our design decisions. He emphasized the need for simplicity and gave us feedback regarding how to serve information to the user. The current plan is to have a mini webapp that gives the client updates on the status of their onboarding

Interesting slide