



Product Specification

1	Introduction	1
1.1	Purpose	1
1.2	Scope	1
1.3	References	1
2	Overall Description	2
3	Product Backlog	2
4	Nonfunctional Requirements	13
4.1	Performance Requirements	13
4.2	Safety Requirements	13
4.3	Security Requirements	13
4.4	Software Quality Attributes	14
4.5	Other Requirements	14
5	Appendix A: Terminology/Glossary/Definitions list	14
6	Appendix B: Analysis Models	16
7	Signatures	23

1 Introduction

1.1 Purpose

This document is made for describe a simple login form with register form with ReactJS Front-end and Laravel Back-end made for SaaS Apps for all contributors on this Application and/or Software.

1.2 Scope

To make login and registering basics for a SaaS Apps with React (JavaScript) and Laravel (PHP).

1.3 References

Laravel: <https://laravel.com/>

ReactJS: <https://reactjs.org/>

TypeScript: <https://www.typescriptlang.org/>

react-easy-crop: <https://www.npmjs.com/package/react-easy-crop>

React Redux: <https://react-redux.js.org/>

React Native: <https://reactnative.dev/>



FORMIK: <https://formik.org/>

2 Overall Description

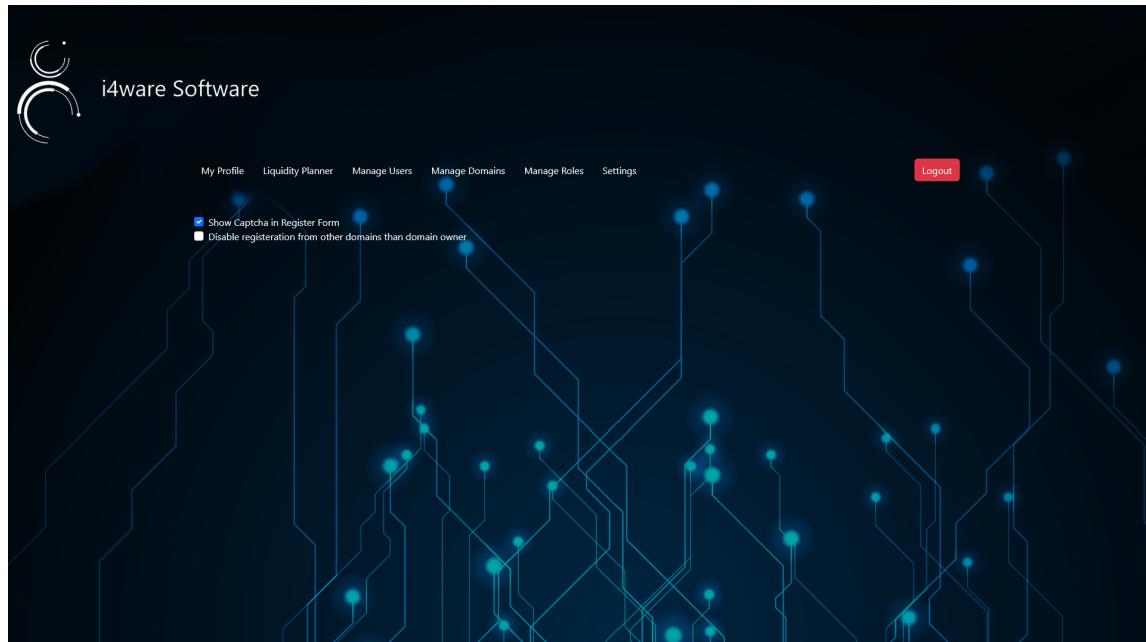
This is a simple login form with register form with ReactJS Front-end and Laravel Back-end made for SaaS Applications.

3 Product Backlog

SAAS-480 - Settings Page

Make a settings page. [■ Must Have, ✓ Done, [Target Version]]

SAAS-481 - Settings Page



SAAS-482 - Captcha for Signup



Name
John Doe
We'll never share your name with anyone else.

Gender
Male
We'll never share your gender with anyone else.

Email
john.doe@domain.com
We'll never share your email with anyone else.

Domain
www.domain.com
We'll never share your domain with anyone else.

Password
Password is strongly encrypted and is secure in our database.

Confirm Password
Password is strongly encrypted and is secure in our database.

w6cOgP

Insert captcha

Agreed on [Privacy Policy](#) and [Data Processing Agreement](#)

[Register](#)

Already have an account? [Login here](#)

Must Have, Accepted, [Planned In]

SAAS-451 - Email Verification

Make simple email verification page. Should Have, Done, [Target Version]

Your e-mail is verified. You can now login.
[Go to Login? Login here](#)

Must Have, Accepted, [Planned In]

SAAS-392 - Login Form

A simple login form with React. Must Have, Done, [Target Version]

SAAS-395 - Login Form



The screenshot shows a dark-themed registration form. At the top left is the i4ware Software logo. In the top right corner, there is a "Login" button. The main form area contains fields for "Email address" and "Password", both with placeholder text and "Forgot password?" links. A "Submit" button is at the bottom. Below the form, a note says "Don't have an account? [Register](#) or forgot a password? [Reset](#)". The background features a blue circuit board pattern.

Must Have, Accepted, [Planned In]]

SAAS-391 - Register Form

A simple register form with React. [Must Have, Done, [Target Version]]

The screenshot shows a detailed registration form. It includes fields for "Name" (with placeholder "John Doe"), "Gender" (with dropdown options "Male" and "Female"), "Email" (with placeholder "john.doe@domain.com"), "Domain" (with placeholder "www.domain.com"), "Password" (with placeholder "password123"), "Confirm Password" (with placeholder "password123"), and a "Captcha" field containing "w6cO8P". There is also a checkbox for "Agreed on Privacy Policy and Data Processing Agreement". A "Register" button is at the bottom. Below the form, a note says "Already have an account? [Login here](#)". The background features a blue circuit board pattern.

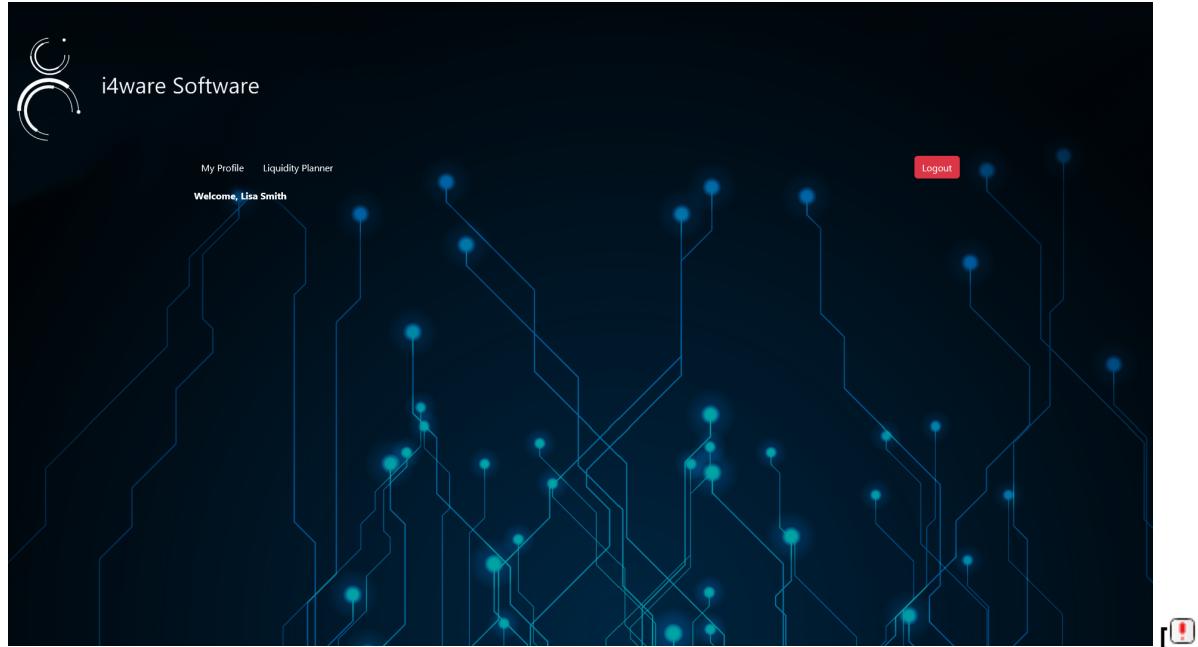
Must Have, Accepted, [Planned In]]

SAAS-402 - Welcome Page

Content page after login.

[Must Have, Done, [Target Version]]

SAAS-403 - Welcome Page



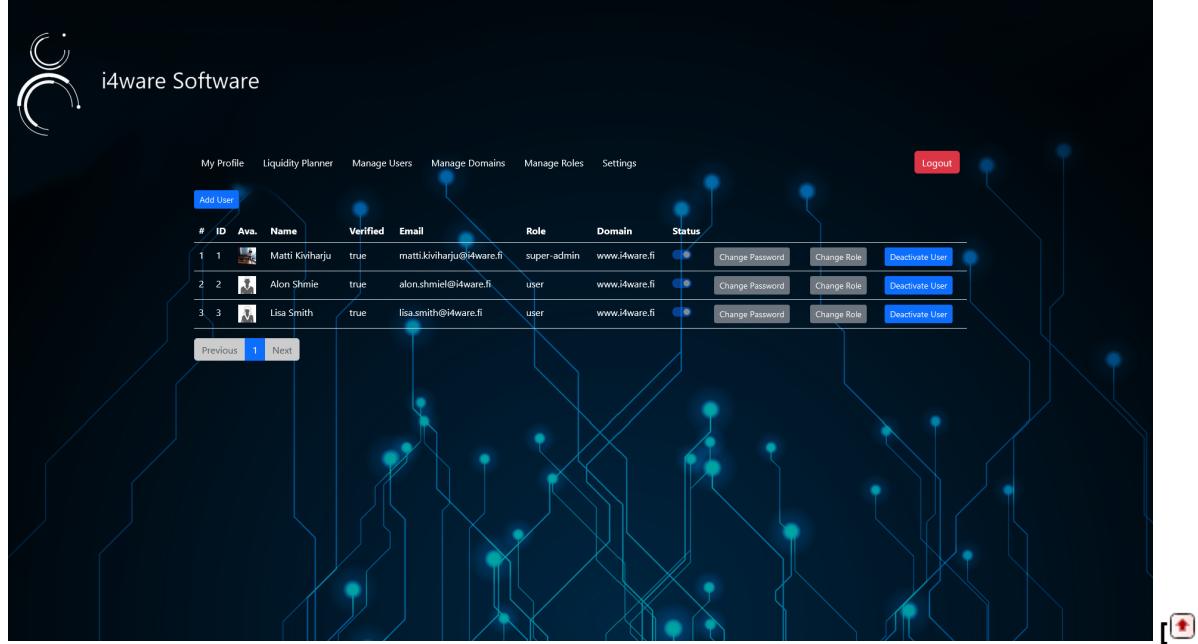
Must Have, Accepted, [Planned In]]

SAAS-467 - Admin Page for Manage Users

Make User Management.

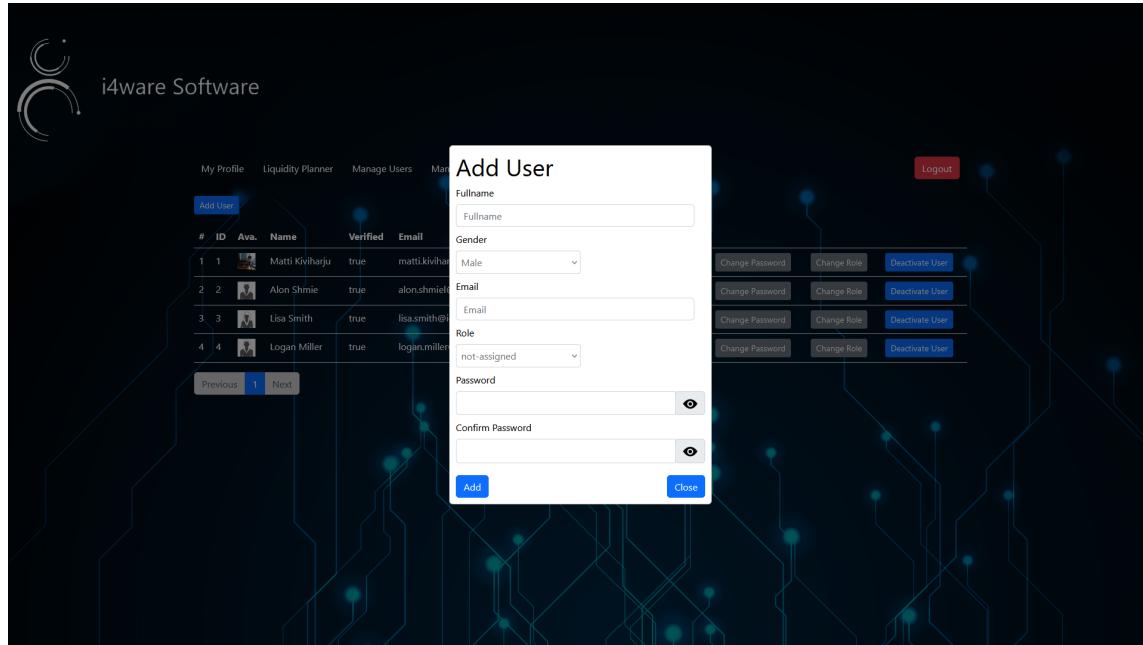
[Should Have, Done, [Target Version]]

SAAS-468 - Admin Page for Manage Users



Should Have, Accepted, [Planned In]]

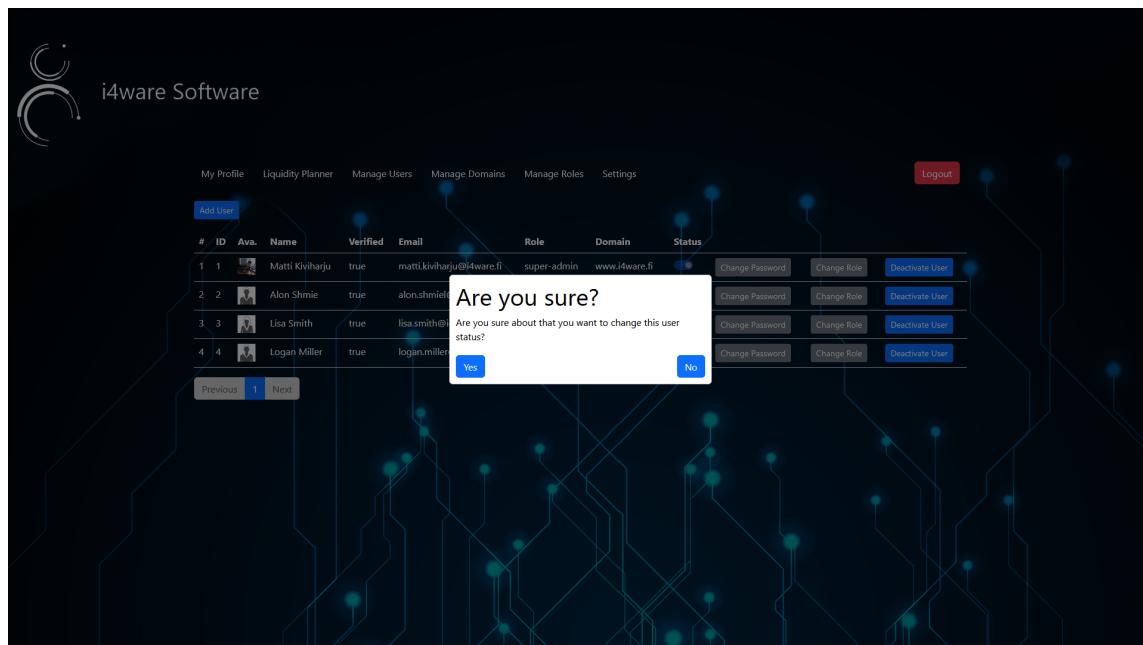
SAAS-645 - Add User Button and Modal Window with user details form



[S]

should Have, Accepted, [Planned In]]

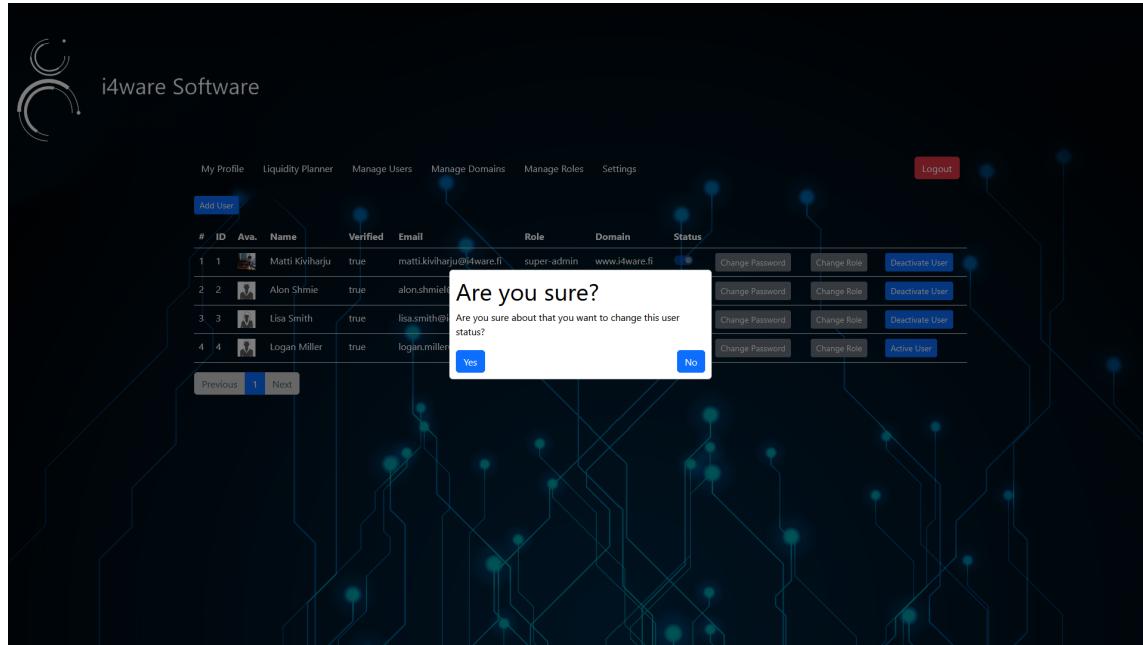
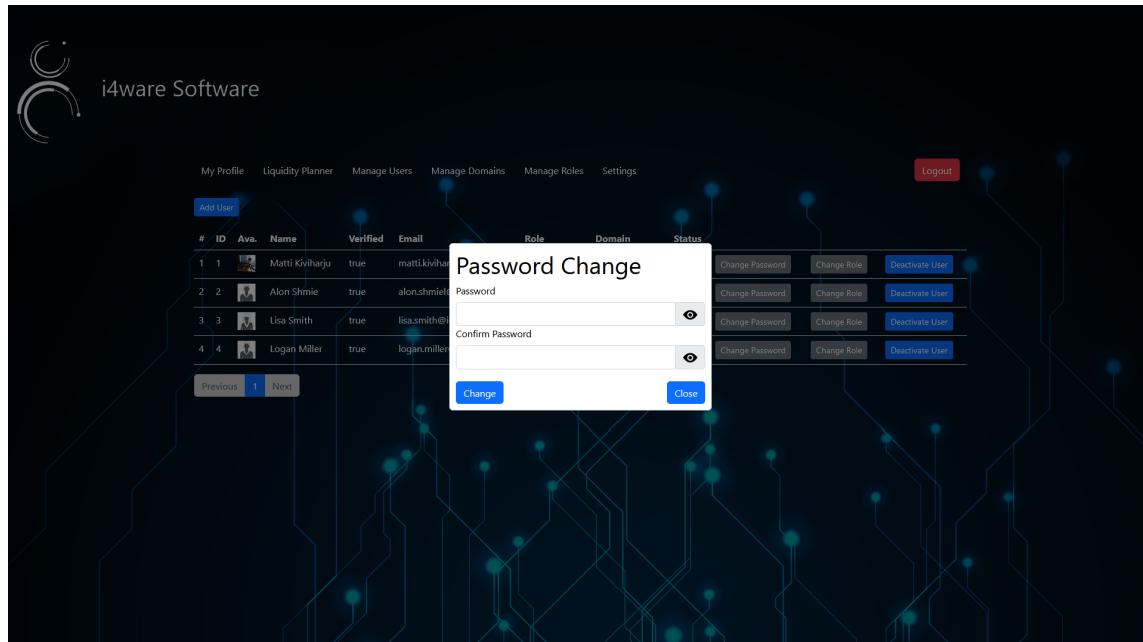
SAAS-646 - Deactivate User Button and Modal Window for it to confirmation



[S]

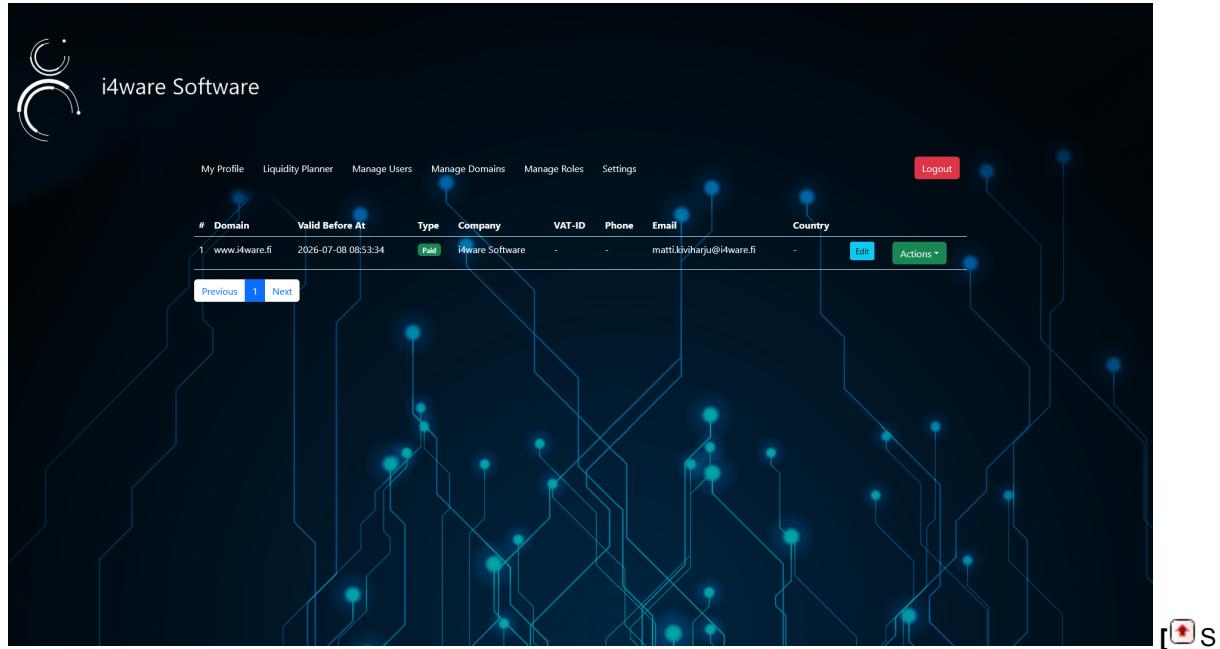
should Have, Accepted, [Planned In]]

SAAS-647 - Activate User button and Modal Window for it with confirmation

[* S]Should Have, ✓ Accepted, [Planned In]]**SAAS-648 - Change Password button with Modal Window with form**[* S]Should Have, ✓ Accepted, [Planned In]]**SAAS-465 - Admin Page for Manage Domains**

Manage Domain Subscriptions, etc.

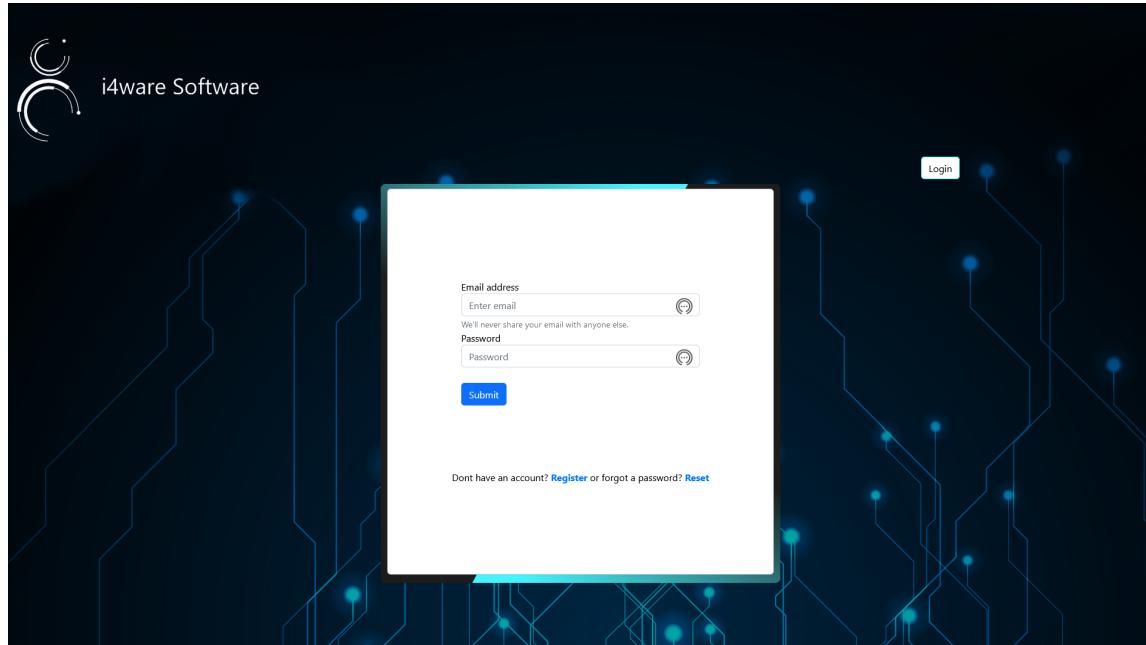
[* Should Have, ✓ Done, [Target Version]]**SAAS-466 - Admin Page for Manage Domains**



[*] Should Have, [Accepted], [Planned In]

SAAS-649 - Password Reset functionality for login form

Password Reset Link:

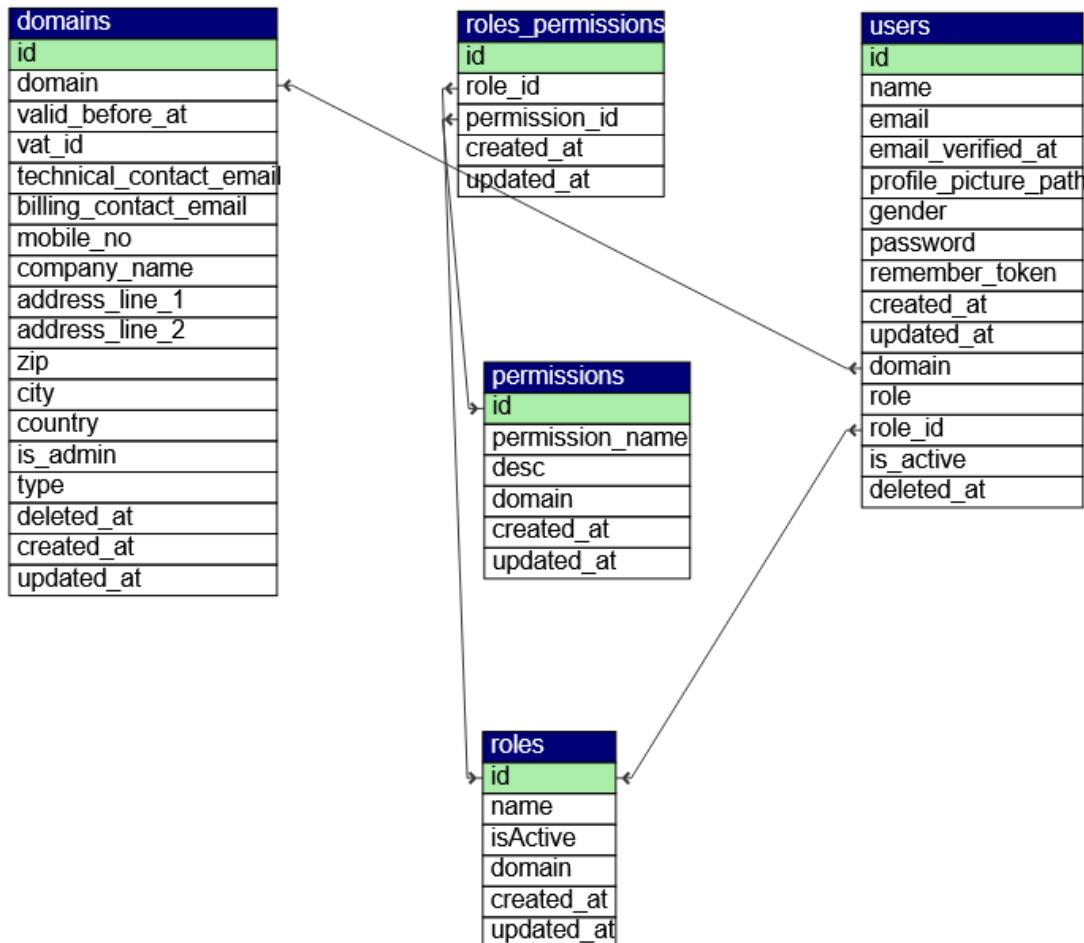


Here is an example: <https://www.itsolutionstuff.com/post/laravel-custom-forgot-reset-password-exampleexample.html>

[*] Should Have, [Accepted], [Planned In]

SAAS-658 - More better ACL

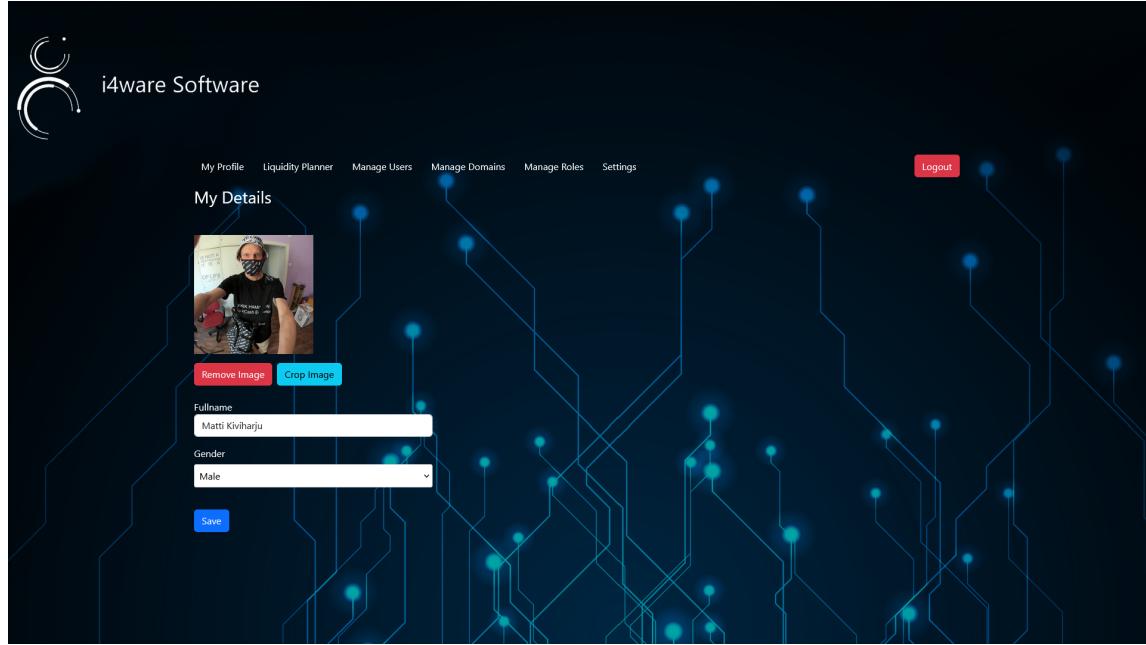
Access Control List like in Zion Framework for Zend Framework 1 implemented to Laravel 9.



[S

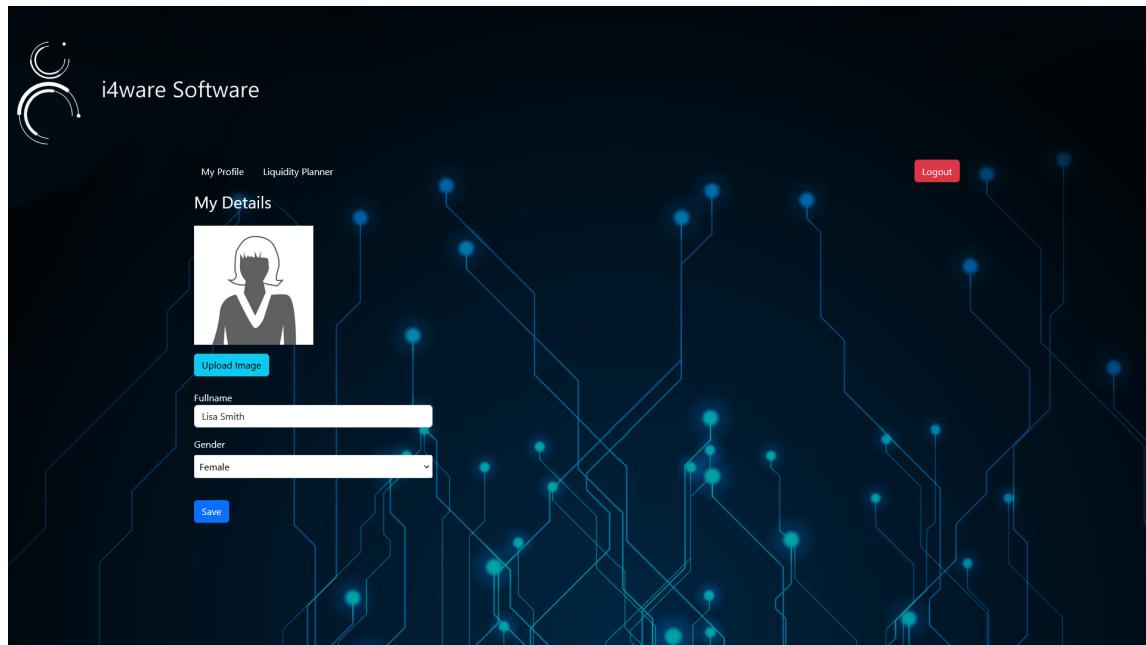
Should Have, Accepted, [Planned In]

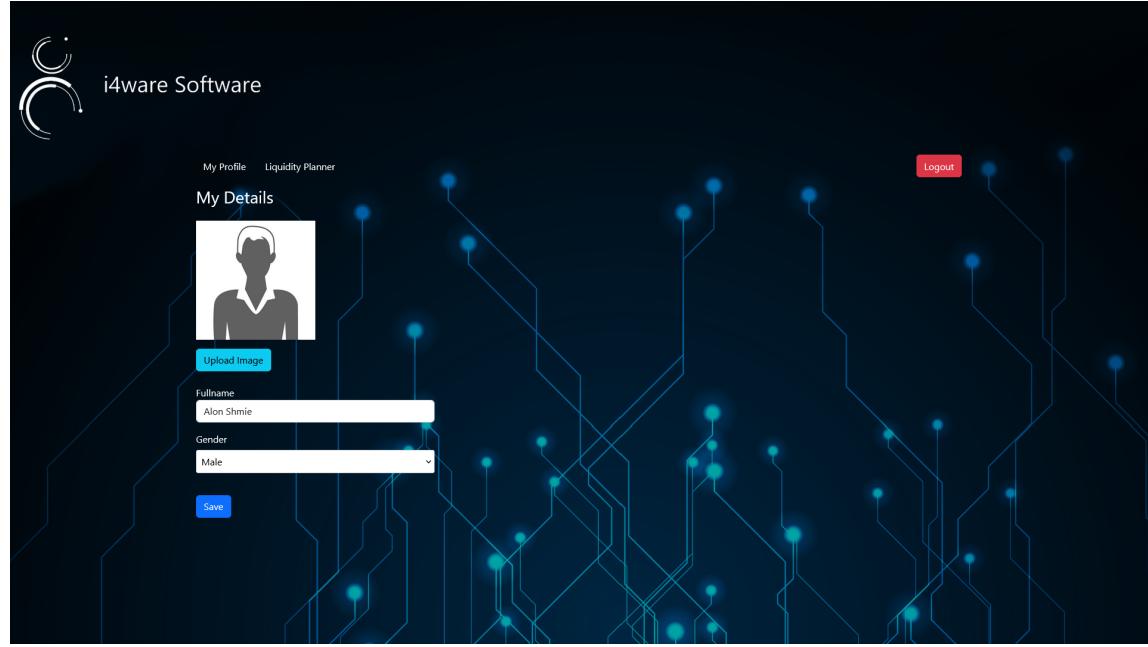
SAAS-723 - My Profile Page



Above is profile page with uploaded profile picture.

Also there is default profile pictures based to gander below:





[S]

should Have, Accepted, [Planned In]]

SAAS-722 - Liquidity Planner Page

This page is just a demo for potential customers what we can do in example.

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Totals
Liquid funds												
Cashbox/bank	10.000,00	9.756,23	10.706,50	8.074,77	7.102,26	8.700,59	2.901,09	-841,71	-6.087,38	-15.769,09	-15.373,83	
Inflow												
Sales	33.998,8	34.172,12	34.454,16	35.281,63	30.306,70	29.866,77	31.271,97	32.098,06	27.113,92	36.369,46	31.659,89	395.940,06
Loan disbursement	5.058,43	4.733,23	4.664,49	6.000,18	4.825,58	5.688,22	5.498,60	5.299,96	4.680,26	4.371,80	4.941,01	61.023,36
Private deposits/equity	685,89	846,47	483,02	578,2	748,55	20124	671,20	771	926,45	722,76	780,92	6.101,18
Other incoming payments	7039,84	6.904,49	6.992,91	7.098,16	7.041,19	6.953,33	6.983,76	7.042,64	7.000,33	7.002,94	7.048,86	84.303,92
Other income												0,00
Cash in (total)	46.835,37	46.296,31	46.984,55	46.837,79	47.932,02	41.931,46	44.348,55	44.351,15	38.736,36	48.466,86	48.430,63	547.176,55
Outflow												
Use of goods/materials	2.036,04	1.941,54	2.078,67	1.865,24	1.968,74	2.046,62	1.939,48	2.032,43	1.934,65	2.067,08	2.090,01	24.123,38
Heating, electricity, water, gas	975,36	920,26	996,83	902,21	1.029,97	1.008,57	1.072,31	1.046,69	1.004,54	1.090,33	1.032,82	12.615,12
Personnel costs	974,49	943,06	1.023,95	1.047,39	999,33	988,09	922,77	905,65	1.001,95	977,57	1.078,44	11.670,70
Credit line overdraft												30.488,72

[S]

should Have, Accepted, [Planned In]]

SAAS-724 - Mage Roles Page



i4ware Software

My Profile Liquidity Planner Manage Users Manage Domains Manage Roles Settings Logout

#	Name
1	super-admin
2	admin
3	user

Add Previous Next Edit Remove

i4ware Software

My Profile Liquidity Planner Manage Users Manage Domains Manage Roles Settings Logout

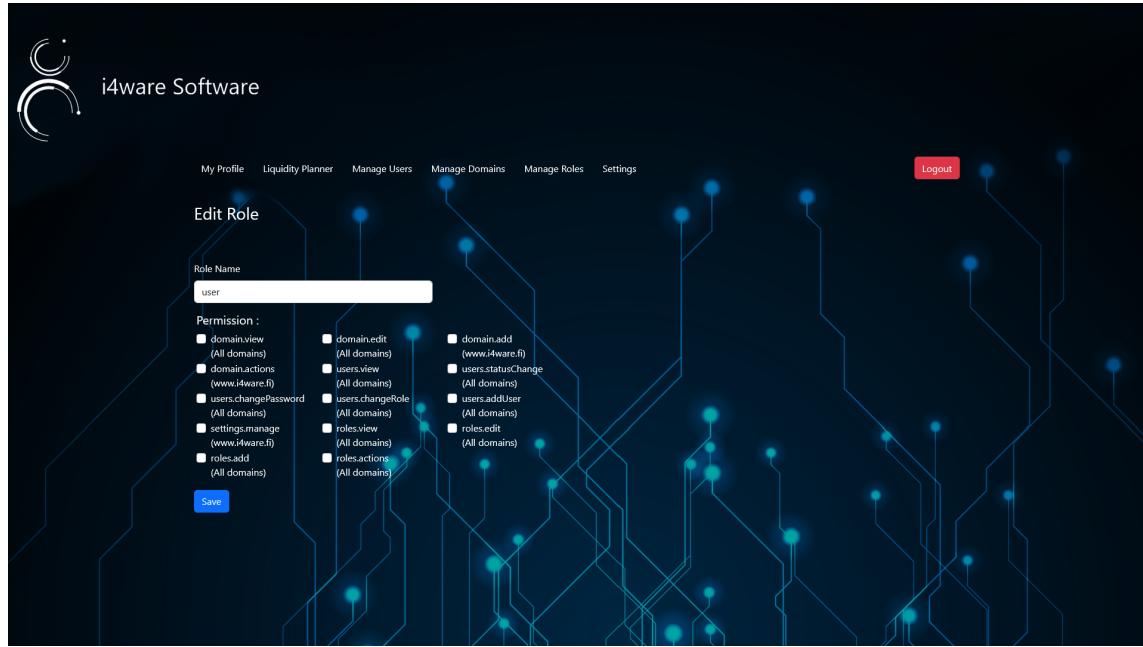
Add Role

Role Name:

Permission :

- domain:view (All domains)
- domain:actions (www.i4ware.fi)
- users:changePassword (All domains)
- settings:manage (www.i4ware.fi)
- roles:add (All domains)
- domain:edit (All domains)
- users:view (All domains)
- users:changeRole (All domains)
- roles:view (All domains)
- roles:actions (All domains)
- domain.add (www.i4ware.fi)
- users:statusChange (All domains)
- users:addUser (All domains)
- roles:edit (All domains)

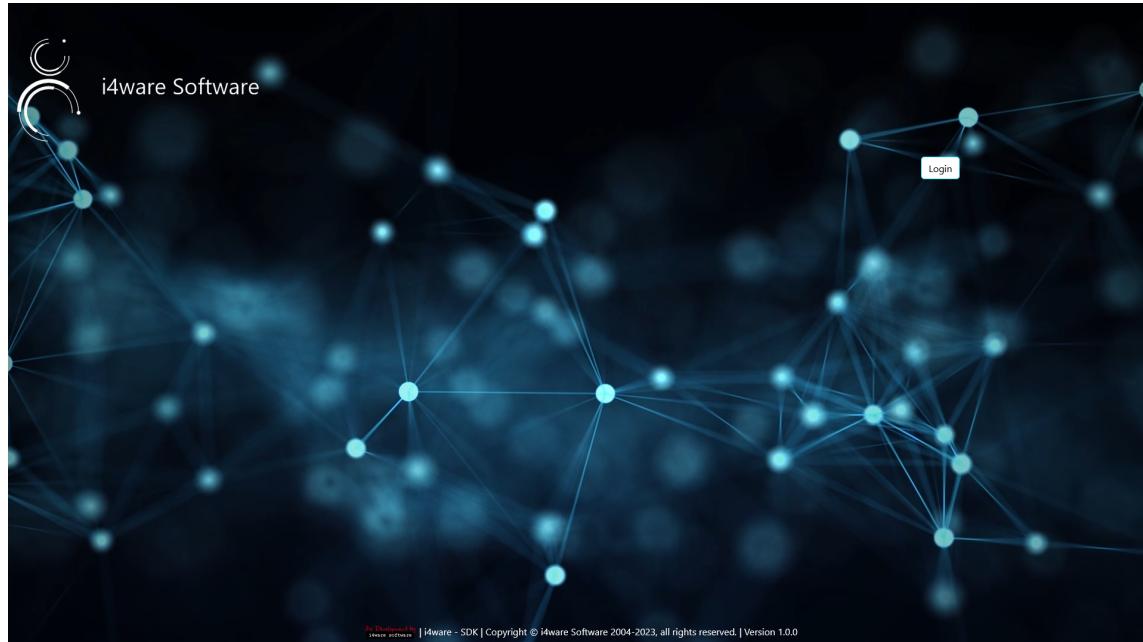
Save



[S]

Should Have, Accepted, [Planned In]

SAAS-725 - Copyright Information Page with Animation Video



[S]

Should Have, Accepted, [Planned In]



4 Nonfunctional Requirements

4.1 Performance Requirements

If there are performance requirements for the product under various circumstances, state them here and explain their rationale, to help the developers understand the intent and make suitable design choices. Specify the timing relationships for real time systems. Make such requirements as specific as possible. You may need to state performance requirements for individual functional requirements or features.

SAAS-441 - Virtual Private Server with enough CPU, RAM and HDD

PHP software will work with minimal CPU, RAM and HDD. Much of these is not needed. [Should Have, Approved, [Target Version]]

4.2 Safety Requirements

Specify those requirements that are concerned with possible loss, damage, or harm that could result from the use of the product. Define any safeguards or actions that must be taken, as well as actions that must be prevented. Refer to any external policies or regulations that state safety issues that affect the product's design or use. Define any safety certifications that must be satisfied.

SAAS-453 - Development in local Machines

Development in local Machines. [Must Have, Approved, [Target Version]]

4.3 Security Requirements

Specify any requirements regarding security or privacy issues surrounding use of the product or protection of the data used or created by the product. Define any user identity authentication requirements. Refer to any external policies or regulations containing security issues that affect the product. Define any security or privacy certifications that must be satisfied.

SAAS-455 - Laravel's Build-in Passport and JWT for make cross-domain Authentication between ReactJS GUI and Back-end

Tutorials is here: <https://laravel.com/docs/9.x/passport> and <https://medium.com/technology-hits/laravel-react-fullstack-application-with-passport-redux-and-material-ui-backend-1c4b38b08b01> [Should Have, Approved, [Target Version]]

SAAS-454 - Right Configuration in Apache for SSL Certificate with grade A to A+

Right Configuration in Apache for SSL Certificate with grade A to A+ [Must Have, Approved, [Target Version]]



4.4 Software Quality Attributes

Specify any additional quality characteristics for the product that will be important to either the customers or the developers. Some to consider are: adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usability. Write these to be specific, quantitative, and verifiable when possible. At the least, clarify the relative preferences for various attributes, such as ease of use over ease of learning.

SAAS-446 - Responsive General User Interface

Must work in mobile and desktop environment in the all screen sizes. [Should Have, Approved, [Target Version]]

4.5 Other Requirements

Define any other requirements not covered elsewhere in the specification. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.

SAAS-442 - MIT licensed Source Code

It is recommended that we copy only a MIT licensed Source Code from the Internet if needed and otherwise all contributors writing it them selves. [Should Have, Approved, [Target Version]]

SAAS-445 - Support for MySQL, PostgreSQL, etc.

Support for MySQL, PostgreSQL, etc. This is database independent based to Laravel PHP Framework, [Should Have, Approved, [Target Version]]

SAAS-444 - Multilingual

Support for English and Finnish is minimum requirement. [Should Have, Approved, [Target Version]]

SAAS-443 - Maximum reuse cases

Re-usability is this software main focus. [Should Have, Approved, [Target Version]]

SAAS-447 - Development Environment in XAMPP

Developing this software in XAMPP in local machine. [Should Have, Approved, [Target Version]]

SAAS-448 - GIT for store a Source Code Version Control System

GIT like Atlassian Bitbucket to store a Source Code and it's changes, etc. [Should Have, Approved, [Target Version]]

5 Appendix A: Terminology/Glossary/Definitions list

Define all the terms necessary to properly interpret this document, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each document.

- **Personal Access Tokens:** Sometimes, your users may want to issue access tokens to themselves without going through the typical authorization code redirect flow. Allowing users to issue tokens to themselves via your application's UI can be useful for allowing users to experiment with your API or may



serve as a simpler approach to issuing access tokens in general. See details here: <https://laravel.com/docs/8.x/sanctum> and here: <https://laravel.com/docs/8.x/passport#personal-access-tokens>

- **Domains:** Domains in this software is a like space for single company and all it's employees. *Valid_before_at field in this database table is a date-time field for a purpose i.e which date and time this user's domain is effective like for 30 days after user is made a domain for him/her self and his/her company.*
- **Formik:** This is framewok to build forms in React, without the tears.
- **React:** React (also known as **React.js** or **ReactJS**) is a *free and open-source front-end JavaScript library for building user interfaces based on components. It is maintained by Meta (formerly Facebook) and a community of individual developers and companies.*
- **TypeScript:** TypeScript is a strongly typed programming language that builds on JavaScript, giving you better tooling at any scale.
- **Entity–relationship model:** An **entity–relationship model** (or **ER model**) describes interrelated things of interest in a specific domain of knowledge. A basic ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between **entities** (instances of those entity types).
- **Flow Diagram:** Flow diagram is a collective term for a **diagram** representing a flow or set of dynamic relationships in a system. The term **flow diagram** is also used as a synonym for **flowchart**, and sometimes as a counterpart of the **flowchart**.
- **Use Case Diagram:** A **use case diagram** is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses. The actors are often shown as stick figures.
- **React Native:** React Native is an **open-source UI software framework** created by **Meta Platforms, Inc.** It is used to develop applications for **Android, Android TV, iOS, macOS, tvOS, Web, Windows and UWP** by enabling developers to use the **React** framework along with native platform capabilities. It is used to develop the **Android and iOS applications** at **Facebook, Microsoft, and Shopify**. It is also being used to develop virtual reality applications at **Oculus**.
- **React Redux:** Redux is an **open-source JavaScript library** for managing and centralizing application **state**. It is most commonly used with libraries such as **React** or **Angular** for building **user interfaces**. Similar to (and inspired by) Facebook's **Flux architecture**, it was created by **Dan Abramov and Andrew Clark**. Since mid-2016, the primary maintainers are **Mark Erikson and Tim Dorr**.



6 Appendix B: Analysis Models

Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.

SAAS-397 - failed_jobs ERD



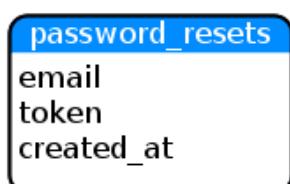
[! Must Have, ! Accepted, [Planned In]]

SAAS-398 - migrations ERD



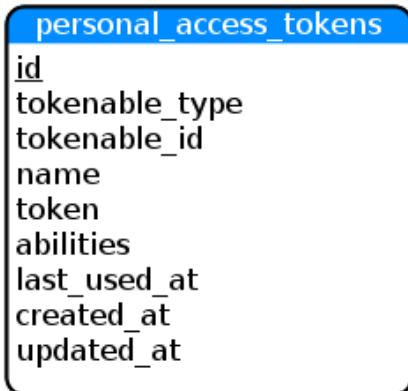
[! Must Have, ! Accepted, [Planned In]]

SAAS-399 - password_resets ERD



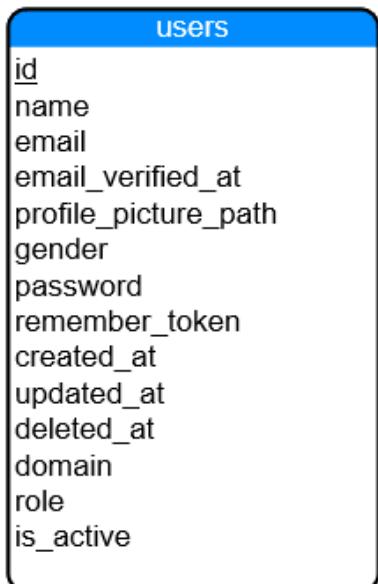
[! Must Have, ! Accepted, [Planned In]]

SAAS-400 - personal_access_tokens ERD



[Must Have, Accepted, [Planned In]]

SAAS-401 - users ERD

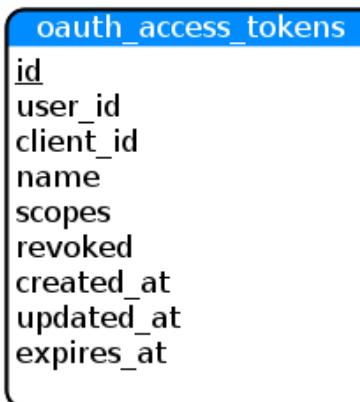


[Must Have, Accepted, [Planned In]]

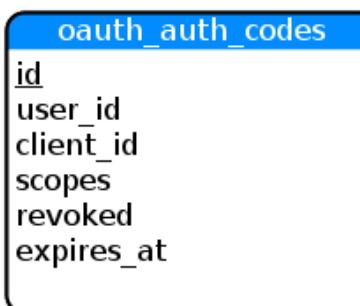
SAAS-404 - domains ERD



[Must Have, Accepted, [Planned In]]

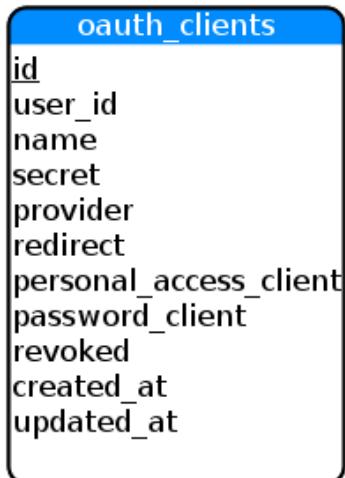
SAAS-456 - oauth_access_tokens ERD

[Should Have, Accepted, [Planned In]]

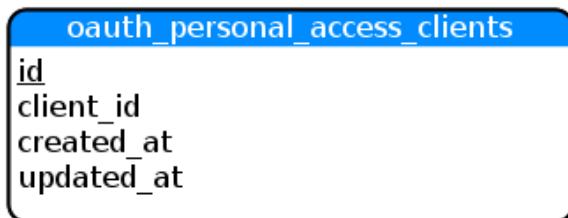
SAAS-457 - oauth_auth_codes ERD



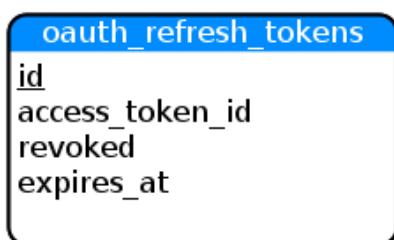
[Should Have, Accepted, [Planned In]]

SAAS-458 - oauth_clients ERD

[Should Have, Accepted, [Planned In]]

SAAS-459 - oauth_personal_access_clients ERD

[Should Have, Accepted, [Planned In]]

SAAS-460 - oauth_refresh_tokens ERD

[Should Have, Accepted, [Planned In]]

SAAS-479 - settings ERD



settings	
PK	id
	setting_key
	setting_value
	created_at
	updated_at

[!] Must Have, 🟢 Accepted, [Planned In]]

SAAS-691 - permissions ERD

roles	
PK	id
	permission_name
	desc
	domain
	created_at
	updated_at

[!] Should Have, 🟢 Accepted, [Planned In]]

SAAS-690 - roles ERD

permissions	
PK	id
	name
	isActive
	created_at
	updated_at

[!] Should Have, 🟢 Accepted, [Planned In]]

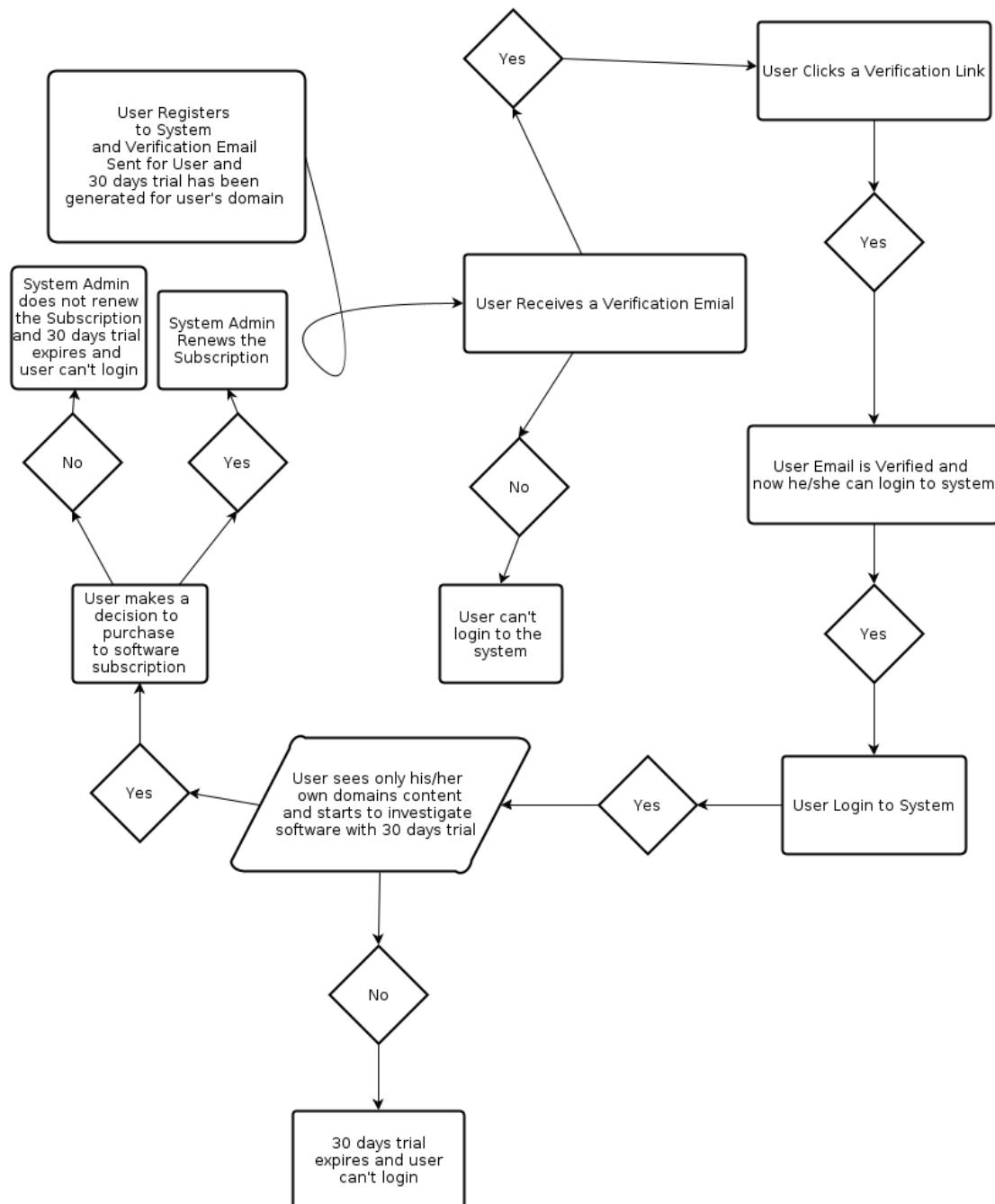
SAAS-692 - roles_permissions ERD



roles_permissions	
PK	id
	role_id
	permission_id
	created_at
	updated_at

[Should Have, Accepted, [Planned In]]

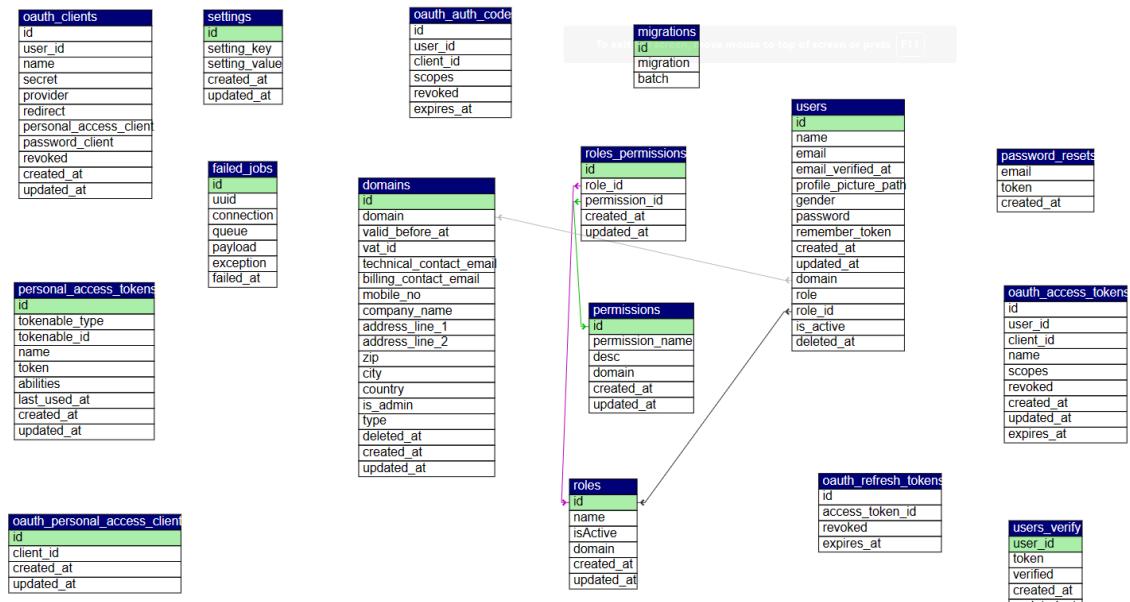
SAAS-461 - Use Case as Flow Diagram



[Sh]

ould Have, [Accepted, [Planned In]]

SAAS-721 - Database Entity Relationship Diagram



ould Have, Accepted, [Planned In]]

Sh

7 Signatures

Document Properties

Status: Approved

Version: 1.0.3

Author: Matti Kiviharju

Created: 2011-01-27 14:35

Approved Versions

Current Document version 1.0.3 is approved.

Approved Versions:

- [1.0.0](#) (2022-03-12 06:41)
- [1.0.1](#) (2022-07-28 20:19)
- [1.0.2](#) (2022-11-12 13:31)

Document Signatures

Reviewed (2022-11-12 13:28)

Matti Kiviharju

Signed

2022-11-12 13:23

**Approved** (2022-11-12 13:31) **Logan Miller** (Approver)

Signed 2022-11-12 13:30

 **Will Smith** (Approver)

Signed 2022-11-12 13:31

 **Matti Kiviharju**

Signed 2022-11-12 13:28