

# Tasks Instructions

ID	1
Title	Implement Models and Database Migrations
Description	<p>We need to create the Document, DocumentVersion and DocumentTask models. It should also include their respective migrations.</p> <p>This will require the <i>php artisan make:model {Model} -m</i> command.</p>
Acceptance Criterias	<ol style="list-style-type: none"><li>Must include all the properties and foreign key relationships described on the model's diagram.</li><li>Must include unit tests to check if documents, document_versions and document_tasks table exists.</li><li>Must include unit tests to check if the create method on classes returns an object with the columns.</li></ol>
Resources	<ul style="list-style-type: none"><li><a href="https://laravel.com/docs/9.x/eloquent">https://laravel.com/docs/9.x/eloquent</a></li><li><a href="https://laravel.com/docs/9.x/database">https://laravel.com/docs/9.x/database</a></li><li><a href="https://laravel.com/docs/9.x/migrations">https://laravel.com/docs/9.x/migrations</a></li><li>Proposed.png in /Diagrams/Backend/Models</li></ul>
Knowledge	Laravel

ID	2
Title	Create Controllers, Policies, Requests and Observers
Description	We need to create the basic scaffolding which future programming tasks will be performed.
Acceptance Criterias	<ol style="list-style-type: none"><li>Must include all the described classes on diagrams.</li></ol>
Resources	<ul style="list-style-type: none"><li><a href="https://laravel.com/docs/9.x/controllers#resource-controllers">https://laravel.com/docs/9.x/controllers#resource-controllers</a></li><li><a href="https://laravel.com/docs/9.x/authorization#creating-policies">https://laravel.com/docs/9.x/authorization#creating-policies</a></li><li><a href="https://laravel.com/docs/9.x/validation#form-request-validation">https://laravel.com/docs/9.x/validation#form-request-validation</a></li><li><a href="https://laravel.com/docs/9.x/eloquent#observers">https://laravel.com/docs/9.x/eloquent#observers</a></li><li>Proposed.png in /Diagrams/Backend/Controllers folder.</li><li>Proposed.png in /Diagrams/Backend/Policies folder.</li><li>Proposed.png in /Diagrams/Backend/Observers folder.</li><li>Proposed Requests.png in /Diagrams/Backend folder.</li></ul>
Knowledge	Laravel

ID	3
Title	Implement DocumentsController and his related Requests
Description	<p>We need to implement the controller methods:</p> <ul style="list-style-type: none"><li>Index method should return a paginated collection of Documents including their relationships and filtered by the user company.</li><li>Store method should create a new document instance and return the recently created instance.</li><li>Versions method must return a paginated collection of the related Document Versions.</li><li>AddVersion method must create a related Document Version and return the recently created instance.</li><li>Tasks method must return a paginated collection of the related Document Tasks.</li><li>AddTask method must create a related Document Task and return the recently created instance.</li><li>Destroy method must delete the instance and his related Versions and Tasks.</li><li>Update method must update the Document instance using request inputs like name and description.</li></ul> <p>All the previous will require his respective Validation rules implemented on the Requests.</p> <ul style="list-style-type: none"><li>StoreRequest must require name, description as strings and file as binary input.</li><li>UpdateRequest must require only name and description as strings.</li></ul>
Acceptance Criterias	<ol style="list-style-type: none"><li>Controllers must complain with the issue description.</li></ol>
Resources	<ul style="list-style-type: none"><li>DocumentsController on Proposed.png in /Diagrams/Backend/Controllers folder.</li><li>Proposed Requests.png in /Diagrams/Backend folder.</li></ul>
Knowledge	Laravel, Eloquent, Responses, Requests, Validation Rules and Relationships.

ID	4
Title	Implement DocumentVersionsController and his related Requests
Description	We need to implement the download controller method and implement the validation rules:  The method must return a BinaryResponse of the file located on the path property of the Document Version instance.
Acceptance Criterias	1. Must include a unit testing to check if the response is a BinaryResponse.
Resources	<ul style="list-style-type: none"><li>- <a href="https://laravel.com/docs/9.x/responses#file-responses">https://laravel.com/docs/9.x/responses#file-responses</a></li><li>- DocumentsController on Proposed.png in /Diagrams/Backend/Controllers folder.</li><li>- Proposed Requests.png in /Diagrams/Backend folder.</li></ul>
Knowledge	Laravel, Eloquent, Responses, Requests and Validation Rules.

ID	5
Title	Implement DocumentTasksController and his related Requests
Description	We need to implement the controller methods: <ul style="list-style-type: none"><li>- Update method must update the Document Task instance using the description request input.</li><li>- Destroy method must delete the Document Task instance.</li></ul> All the previous will require his respective Validation rules implemented on the Requests. <ul style="list-style-type: none"><li>- UpdateRequest must require description as string.</li><li>- DestroyRequest will not require any input.</li></ul>
Acceptance Criterias	1. Must include a unit testing to check if the response is a BinaryResponse.
Resources	<ul style="list-style-type: none"><li>- <a href="https://laravel.com/docs/9.x/responses#file-responses">https://laravel.com/docs/9.x/responses#file-responses</a></li><li>- DocumentTasksController on Proposed.png in /Diagrams/Backend/Controllers folder.</li><li>- Proposed Requests.png in /Diagrams/Backend folder.</li></ul>
Knowledge	Laravel, Eloquent, Responses, Requests and Validation Rules.

ID	6
Title	Implement DocumentObserver created method to create the related document version one
Description	We need to implement the created method on the DocumentObserver class:  When the Document is created then we need to create a related Document Version and the code must be 1.  You can access the related uploaded file using request helper.
Acceptance Criterias	1. The DocumentObserver must create a DocumentVersion and store the document file on storage.
Resources	<ul style="list-style-type: none"><li>- <a href="https://laravel.com/docs/9.x/eloquent#observers">https://laravel.com/docs/9.x/eloquent#observers</a></li><li>- Proposed.png in /Diagrams/Backend/Observers folder.</li></ul>
Knowledge	Laravel, Eloquent, Observers and Storage.

ID	7
Title	Implement Policies methods and use on Controllers
Description	We need to implement the policies described on the diagrams and we need to ensure that users have the right permissions via checking if the records belong to the company.  Also we need to use Gate authorize on controllers.
Acceptance Criterias	1. The Policies must check if records company_id is equal to the connected one.
Resources	<ul style="list-style-type: none"><li>- <a href="https://laravel.com/docs/9.x/authorization#writing-policies">https://laravel.com/docs/9.x/authorization#writing-policies</a></li><li>- <a href="https://laravel.com/docs/9.x/authorization#authorizing-or-throwing-exceptions">https://laravel.com/docs/9.x/authorization#authorizing-or-throwing-exceptions</a></li><li>- Proposed.png in /Diagrams/Backend/Controllers folder.</li></ul>
Knowledge	Laravel, Policies and Controllers.

ID	8
Title	Create StorageObject, StorageResolver and CustomStorage classes with empty methods
Description	We need to create the basic scaffolding to make future programming tasks.
Acceptance Criterias	<div>1. The classes must have the described methods and properties on the Proposed Storages diagram.</div>
Resources	<div><div>- <a href="https://www.php.net/manual/en/language.oop5.php">https://www.php.net/manual/en/language.oop5.php</a></div><div>- <a href="https://www.php.net/manual/en/language.oop5.inheritance.php">https://www.php.net/manual/en/language.oop5.inheritance.php</a></div><div>- Proposed Storages.png in /Diagrams/Backend folder.</div></div>
Knowledge	PHP and Inheritance.

ID	9
Title	Implement StorageResolver methods using user company storage and token properties.
Description	<div>We need to implement the StorageResolver class methods and properties.</div> <div><div>- Contracts property is a key-value array and the key is a snake case string and the value is the invocable implementation class.</div><div>- Resolve method must find the company storage value on the contracts property keys and if the company storage property is null then will use the default storage, if not then will make the implementation class.</div></div> <div>Also we need to extend the storage as is described in the AppServiceProviders on the Laravel custom file system documentation using the Contracts properties.</div> <div>Token must be used as authentication token to the custom implementation.</div>
Acceptance Criterias	<div>1. AppServiceProvider must implement a custom file system storage.</div> <div>2. Method must return the custom implementation resolved.</div>
Resources	<div><div>- <a href="https://laravel.com/docs/9.x/filesystem#custom-file-systems">https://laravel.com/docs/9.x/filesystem#custom-file-systems</a></div><div>- Proposed Storages.png in /Diagrams/Backend folder.</div></div>
Knowledge	Laravel, Filesystems, Contracts.

ID	10
Title	Implement StorageResolver on addVersion and store methods of DocumentsController and download method of DocumentsVersionController
Description	<div>We need to implement the StorageResolver on DocumentsController and DocumentVersionsController.</div> <div>To do that we need to ask if the user company has the storage property as not null.</div> <div>If that is true then we need to resolve the storage to make the uploads and download operations.</div>
Acceptance Criterias	<div>1. The Controllers must implement the StorageResolver.</div> <div>2. Unit tests must be included to probe the feature.</div>
Resources	<div><div>- Proposed Storages.png in /Diagrams/Backend folder.</div><div>- Proposed.png in /Diagrams/Backend/Controllers folder.</div></div>
Knowledge	Laravel.

ID	11
Title	Implement Routes for Controllers
Description	We need to implement the controllers' routes using resource controllers routing and custom methods.
Acceptance Criterias	<div>1. The routes must be tested using Http Testing.</div>
Resources	<div><div>- <a href="https://laravel.com/docs/9.x/http-tests#main-content">https://laravel.com/docs/9.x/http-tests#main-content</a></div><div>- <a href="https://laravel.com/docs/9.x/routing#main-content">https://laravel.com/docs/9.x/routing#main-content</a></div></div>
Knowledge	Laravel and Routing.

ID	12
Title	Implement Paginator, Column and Link Types
Description	We need to create the title types in the client/types folder.
Acceptance Criterias	1. The types must comply with the diagram.
Resources	<ul style="list-style-type: none"><li>- <a href="https://www.typescriptlang.org/docs/handbook/basic-types.html">https://www.typescriptlang.org/docs/handbook/basic-types.html</a></li><li>- Types.png in /Diagrams/Frontend folder.</li></ul>
Knowledge	Javascript, Typescript and Types.

ID	13
Title	Implement Document, DocumentTask and DocumentVersion Types
Description	We need to create the title types in the client/types folder.
Acceptance Criterias	1. The types must comply with the diagram.
Resources	<ul style="list-style-type: none"><li>- <a href="https://www.typescriptlang.org/docs/handbook/basic-types.html">https://www.typescriptlang.org/docs/handbook/basic-types.html</a></li><li>- Types.png in /Diagrams/Frontend folder.</li></ul>
Knowledge	Javascript, Typescript and Types.

ID	14
Title	Implement VTable component using VSlots and VSlot Props to render table headers and rows
Description	<p>We need to create a table component:</p> <ul style="list-style-type: none"><li>- The table component must receive a schema prop as Array of Column.</li><li>- The table component must receive a records prop as Array of Any.</li><li>- The table must render the columns using vslots passing the row record.</li><li>- The table must render all the records rows using the defined slots.</li></ul>
Acceptance Criterias	1. The components must be included in an implementation example file.
Resources	<ul style="list-style-type: none"><li>- <a href="https://vuejs.org/guide/components/slots.html#scoped-slots">https://vuejs.org/guide/components/slots.html#scoped-slots</a></li><li>- <a href="https://vuejs.org/guide/components/props.html#props">https://vuejs.org/guide/components/props.html#props</a></li><li>- <a href="https://vuejs.org/guide/essentials/list.html">https://vuejs.org/guide/essentials/list.html</a></li><li>- Diagrams in /Diagrams/Wireframes folder.</li></ul>
Knowledge	Vue, Vue Slots and Vue Props.

ID	15
Title	Implement VNavbar, VHeader, VButton and VInput components
Description	<p>We need to create a set of components.</p> <ul style="list-style-type: none"><li>- VHeader is a component to have a couple of slots:<ul style="list-style-type: none"><li>- LeftSide slot to include a title and more details</li><li>- RightSide slot to include buttons for actions.</li></ul></li><li>- VNavbar is a component to have a must receive a Array of Links and must render a group of buttons to make VueRouter redirections.</li><li>- VButton is a button component to add custom handlers on click.</li><li>- VInput is an input component to create forms and must support types of input like text, textarea and file.</li></ul>
Acceptance Criterias	1. The components must be included in the implementation example file.
Resources	<ul style="list-style-type: none"><li>- <a href="https://vuejs.org/guide/essentials/event-handling.html">https://vuejs.org/guide/essentials/event-handling.html</a></li><li>- <a href="https://vuejs.org/guide/essentials/forms.html">https://vuejs.org/guide/essentials/forms.html</a></li><li>- Diagrams in /Diagrams/Wireframes folder.</li></ul>
Knowledge	Vue.

ID	16
Title	Implement Paginator component
Description	<p>We need to create a component to render table footers including Results per page, Total of Records, Previous and Next buttons.</p> <p>We need to respect the Paginator Schema of Laravel.</p> <p>We need to disable the Previous and Next button if it doesn't exist.</p>
Acceptance Criterias	<ol style="list-style-type: none"><li>The components must be included in the implementation example file.</li></ol>
Resources	<ul style="list-style-type: none"><li><a href="https://vuejs.org/guide/essentials/reactivity-fundamentals.html">https://vuejs.org/guide/essentials/reactivity-fundamentals.html</a></li><li><a href="https://laravel.com/docs/9.x/pagination">https://laravel.com/docs/9.x/pagination</a></li><li>Diagrams in /Diagrams/Frontend/Wireframes folder.</li><li>Shared Components.png in /Diagrams/Frontend folder.</li></ul>
Knowledge	Laravel, Vue, Paginator.

ID	17
Title	Implement Modal and User Avatar components
Description	<p>We need to create two components:</p> <ul style="list-style-type: none"><li>Modal is a modal component using a card and a 35% black background.</li><li>User Avatar is a user icon component which enable a tooltip with the user name when the mouse is hover.</li></ul>
Acceptance Criterias	<ol style="list-style-type: none"><li>The components must be included in the implementation example file.</li></ol>
Resources	<ul style="list-style-type: none"><li><a href="https://www.w3schools.com/howto/howto_css_modals.asp">https://www.w3schools.com/howto/howto_css_modals.asp</a></li><li><a href="https://www.w3schools.com/howto/howto_css_tooltip.asp">https://www.w3schools.com/howto/howto_css_tooltip.asp</a></li><li>Diagrams in /Diagrams/Frontend/Wireframes folder.</li><li>Shared Components.png in /Diagrams/Frontend folder.</li></ul>
Knowledge	Vue, HTML, CSS.

ID	18
Title	Create List.vue, Versions.vue and Tasks.vue including properties and empty methods
Description	<p>We need to create the basic scaffolding for pages.</p> <p>Methods and properties are described on the Pages diagram.</p>
Acceptance Criterias	<ol style="list-style-type: none"><li>The components must include the methods and attributes described on the diagram.</li></ol>
Resources	<ul style="list-style-type: none"><li><a href="https://vuejs.org/api/composition-api-setup.html#composition-api-setup">https://vuejs.org/api/composition-api-setup.html#composition-api-setup</a></li><li>Pages.png in /Diagrams/Frontend folder.</li></ul>
Knowledge	Vue, Composition API.

ID	19
Title	Implement List.vue, Versions.vue and Tasks.vue fetch methods using VTable, Column, Paginator and invoke in onMounted
Description	<p>We need to implement fetch methods on the described pages and invoke it in the onMounted method.</p> <p>We also need to implement the VTable and Paginator.</p> <p>We need to use axios to call the resources api endpoints and fill the records and paginator properties.</p>
Acceptance Criterias	<ol style="list-style-type: none"><li>The pages must render the related records.</li></ol>
Resources	<ul style="list-style-type: none"><li><a href="https://vuejs.org/api/composition-api-lifecycle.html#onmounted">https://vuejs.org/api/composition-api-lifecycle.html#onmounted</a></li><li><a href="https://axios.nuxtjs.org/">https://axios.nuxtjs.org/</a></li><li>Diagrams in /Diagrams/Frontend/Wireframes folder.</li><li>Pages.png in /Diagrams/Frontend folder.</li></ul>
Knowledge	Vue, Composition API, Axios.

ID	20
Title	Implement Versions.vue download method to retrieve a binary response
Description	We need to add a button on the Version VTable slots as actions to download the file.  We need to use axios to call the download api and create a downloadable link on vue.
Acceptance Criterias	<ol style="list-style-type: none"><li>The versions table must have a column with name actions.</li><li>The actions column must have a download button link.</li></ol>
Resources	<ul style="list-style-type: none"><li><a href="https://axios.nuxtjs.org/">https://axios.nuxtjs.org/</a></li><li><a href="https://vuejs.org/guide/essentials/event-handling.html">https://vuejs.org/guide/essentials/event-handling.html</a></li></ul>
Knowledge	Vue, Composition API, Axios, Binary Responses.

ID	21
Title	Implement List.vue create form template in a modal, add and save methods
Description	The form must: <ul style="list-style-type: none"><li>Use the draft property.</li><li>Added as a modal.</li><li>Implement the required Api calls.</li><li>Must call the fetch method after the create.</li></ul>
Acceptance Criterias	<ol style="list-style-type: none"><li>The form must have validation messages.</li><li>The form must use the v-input component.</li><li>The form must have cancel and store buttons.</li></ol>
Resources	<ul style="list-style-type: none"><li><a href="https://axios.nuxtjs.org/">https://axios.nuxtjs.org/</a></li><li><a href="https://vuejs.org/guide/essentials/event-handling.html">https://vuejs.org/guide/essentials/event-handling.html</a></li><li>Add Document.png in the /Diagrams/Frontend/Wireframes folder.</li></ul>
Knowledge	Vue, Composition API, Axios.

ID	22
Title	Implement Versions.vue create form template in a modal, add and save methods
Description	The form must: <ul style="list-style-type: none"><li>Use the draft property.</li><li>Added as a modal.</li><li>Implement the required Api calls.</li><li>Must call the fetch method after the create.</li></ul>
Acceptance Criterias	<ol style="list-style-type: none"><li>The form must have validation messages.</li><li>The form must use the v-input component.</li><li>The form must have cancel and store buttons.</li></ol>
Resources	<ul style="list-style-type: none"><li><a href="https://axios.nuxtjs.org/">https://axios.nuxtjs.org/</a></li><li><a href="https://vuejs.org/guide/essentials/event-handling.html">https://vuejs.org/guide/essentials/event-handling.html</a></li><li>Add Version.png in the /Diagrams/Frontend/Wireframes folder.</li></ul>
Knowledge	Vue, Composition API, Axios.

ID	23
Title	Implement Tasks.vue create form template in a modal, add and save methods
Description	The form must: <ul style="list-style-type: none"><li>Use the draft property.</li><li>Added as a modal.</li><li>Implement the required Api calls.</li><li>Must call the fetch method after the create.</li></ul>
Acceptance Criterias	<ol style="list-style-type: none"><li>The form must have validation messages.</li><li>The form must use the v-input component.</li><li>The form must have cancel and store buttons.</li></ol>
Resources	<ul style="list-style-type: none"><li><a href="https://axios.nuxtjs.org/">https://axios.nuxtjs.org/</a></li><li><a href="https://vuejs.org/guide/essentials/event-handling.html">https://vuejs.org/guide/essentials/event-handling.html</a></li><li>Add Task.png in the /Diagrams/Frontend/Wireframes folder.</li></ul>
Knowledge	Vue, Composition API, Axios.



ID	24
Title	Implement List.vue update form template in a modal, add and save methods
Description	<div>The form must:<ul style="list-style-type: none"><li>- Use the draft property.</li><li>- Added as a modal.</li><li>- Implement the required Api calls.</li><li>- Must call the fetch method after the update.</li></ul></div>
Acceptance Criterias	<div><div>1. The form must have validation messages.</div><div>2. The form must use the v-input component.</div><div>3. The form must have cancel and update buttons.</div></div>
Resources	<div><ul style="list-style-type: none"><li>- <a href="https://axios.nuxtjs.org/">https://axios.nuxtjs.org/</a></li><li>- <a href="https://vuejs.org/guide/essentials/event-handling.html">https://vuejs.org/guide/essentials/event-handling.html</a></li><li>- Edit Document.png in the /Diagrams/Frontend/Wireframes folder.</li></ul></div>
Knowledge	Vue, Composition API, Axios.

ID	25
Title	Implement Tasks.vue update form template in a modal, add and save methods
Description	<div>The form must:<ul style="list-style-type: none"><li>- Use the draft property.</li><li>- Added as a modal.</li><li>- Implement the required Api calls.</li><li>- Must call the fetch method after the update.</li></ul></div>
Acceptance Criterias	<div><div>1. The form must have validation messages.</div><div>2. The form must use the v-input component.</div><div>3. The form must have cancel and update buttons.</div></div>
Resources	<div><ul style="list-style-type: none"><li>- <a href="https://axios.nuxtjs.org/">https://axios.nuxtjs.org/</a></li><li>- <a href="https://vuejs.org/guide/essentials/event-handling.html">https://vuejs.org/guide/essentials/event-handling.html</a></li><li>- Edit Document.png in the /Diagrams/Frontend/Wireframes folder as guide.</li></ul></div>
Knowledge	Vue, Composition API, Axios.

ID	26
Title	Implement Tasks.vue finish method and button calling the finish task route
Description	<div>We need to add a button on the Task VTable slots as actions to mark the task as completed.</div> <div>We need to use axios to call the completed task api and execute the fetch method after that.</div>
Acceptance Criterias	<div><div>1. The tasks table must have a column with name actions.</div><div>2. The actions column must have a finish button link.</div></div>
Resources	<div><ul style="list-style-type: none"><li>- <a href="https://axios.nuxtjs.org/">https://axios.nuxtjs.org/</a></li><li>- <a href="https://vuejs.org/guide/essentials/event-handling.html">https://vuejs.org/guide/essentials/event-handling.html</a></li></ul></div>
Knowledge	Vue, Composition API, Axios, Binary Responses.

ID	27
Title	Implement List.vue remove method and button calling the destroy document route
Description	<div>The form must:<ul style="list-style-type: none"><li>- Use the draft property.</li><li>- Added as a modal.</li><li>- Implement the required Api calls.</li><li>- Must call the fetch method after the update.</li></ul></div>
Acceptance Criterias	<div><div>4. The form must have validation messages.</div><div>5. The form must use the v-input component.</div><div>6. The form must have cancel and update buttons.</div></div>
Resources	<div><ul style="list-style-type: none"><li>- <a href="https://axios.nuxtjs.org/">https://axios.nuxtjs.org/</a></li><li>- <a href="https://vuejs.org/guide/essentials/event-handling.html">https://vuejs.org/guide/essentials/event-handling.html</a></li><li>- Remove Document.png in the /Diagrams/Frontend/Wireframes folder.</li></ul></div>
Knowledge	Vue, Composition API, Axios.

ID	28
Title	Implement Tasks.vue remove method and button calling the destroy document route
Description	<div>The form must:<ul style="list-style-type: none"><li>- Use the draft property.</li><li>- Added as a modal.</li><li>- Implement the required Api calls.</li><li>- Must call the fetch method after the update.</li></ul></div>
Acceptance Criterias	<div><div>7. The form must have validation messages.</div><div>8. The form must use the v-input component.</div><div>9. The form must have cancel and update buttons.</div></div>
Resources	<div><ul style="list-style-type: none"><li>- <a href="https://axios.nuxtjs.org/">https://axios.nuxtjs.org/</a></li><li>- <a href="https://vuejs.org/guide/essentials/event-handling.html">https://vuejs.org/guide/essentials/event-handling.html</a></li><li>- Remove Document.png in the /Diagrams/Frontend/Wireframes folder <i>as guide</i>.</li></ul></div>
Knowledge	Vue, Composition API, Axios.