

Client

A startup from Luxembourg used Ensoft to build software for collaboration on documents related to bond.

Challenge

Bond issuing is a complex process. It involves several parties who need to exchange documents, conduct due diligence, ensure legal compliance and go through multiple levels of approval. Sometimes the parties need to prepare dozens of contracts which only differ in a few clauses. All of that takes a lot of time and effort.

One startup based in Luxembourg set out to improve the process. They came up with an idea for an application that would semi-automate agreement creation and allow multiple parties to collaborate on finalizing them. And the idea was so appealing it got investors from the top 10 worldwide investment banks on board.

The startup first hired a freelance developer to build the application. But the developer soon realized the project was too heavy for them to handle and suggested Ensoft as a replacement. We quickly fielded the right team, and the startup chose to go ahead with us.

Process

The team we gathered was quite big. The size changed throughout the project, counting 22 people on average. These included middle-to-senior-level developers specializing in JavaScript, C++ and PHP, a quality assurance wing, a DevOps engineer, a project manager and a business analyst.

We approached the challenge using the Agile, Scrum-oriented practices. And before planning each sprint, we made a point of clarifying the requirements with our client to deal with the overall complexity of the software.

Requirements and priorities kept changing rapidly, and to catch up we had to abandon Scrum and turn to Kanban in the middle of the project. This helped us switch focus from rebuilding the architecture to improving the product following the stakeholders' feedback. At the same time, we continued to gradually refactor the platform. For further new features development we switched back to the Scrum methodology with previously implemented Kanban improvements.

Refactoring the Platform

The previous developer had used C++ to put together a prototype which we were expected to turn into an operating platform. But, after perusing the prototype, we realized the platform couldn't be upgraded/extended unless we refactored the platform. And so we did.

During the refactoring, we had to handle a specific architecture. It represented hundreds of internal dependencies plus integrations with third-party applications. And if that wasn't enough, short timelines and mounting pressure from the stakeholders made the task all the more challenging.

We rose to the challenge by adapting our approach and team composition. And then we had to test all those internal and external dependencies.

Replacing a Subscription-Based Third-Party Service

Originally, our client had used an integration with the third-party tool Contract Express (CI) to generate documents from the information entered in the system. And using that service had cost the startup money.

We built our own internal parser to have the same feature as CI for questionnaires and documents processing, which also optimized and extended the flow of generating documents. Thanks to that, document creation became more user friendly and convenient.

Quality Assurance

The main feature of the platform is user-filled questionnaires. And bugs typically occurred while the system processed user entries, converting questionnaires into documents.

The way processing is carried out depends on two things: the preset template for a particular questionnaire and the software business logic. A minor mistake in either might cause problems, so both needed to be tested. And, on top of that, there are millions of ways in which users may fill out the questionnaires.

So, how to test such a complex system and uncover issues? The answer is via automated testing. We created scripts that generated a multitude of randomly completed questionnaires. But we didn't limit quality assurance to automation — our QA wing accompanied it with manual checks.

Product

The platform we built consists of five applications working together. Let's call them Templates, Contracts, Documents, Collaborations and Management.

The Template application allows users to manage templates and questionnaires, while Contracts and Documents convert questionnaires into documents. The Collaborations application enables multiple users to work together on documents. Finally, Management is where admins set up user roles and permissions.

Let's take a user journey across the platform.

Uploading Templates

It all starts with a template. A template is a DOCX file containing formulas and prepared by the system administrator.

The system admin uploads the template for the system to convert it into a questionnaire. Before uploading, they will be asked to specify which party this document is designated for: an issuer, dealer or internal user.

Creating Questionnaires and Subtemplates

Once the template has been uploaded, the admin switches to another tab. There, they will need to specify a company (dealer or issuer) and a template to be used in creating documents for that party. All options are available in respective dropdown menus.

In addition, the admin can add a subtemplate — an optional, but really useful, feature. Template fields are often filled with heaps of repetitive information, and to avoid entering the same information again, a template sublayer can be reused.

After a questionnaire has been created from the template, the user will see multiple fields of different types, such as dropdowns or radio buttons. They are conveniently organized into several tabs (e.g. financial information, dates and swap details).

Furthermore, once some of the fields are set, new subfields will be revealed. For instance, if a rating agency is specified, the user will see a dropdown menu of ratings to choose from for a specific issuer.

Once all the fields have been completed, the user can save their work either as a questionnaire or as a subtemplate for further uses. The questionnaire fields will be saved in the **Contracts** application. Finally, the questionnaire can be later submitted for collaboration.

Further Actions on Questionnaires

The user can track and edit submitted questionnaires in the respective tab. There, they can check different versions of the submission, follow the changes made to it and push it to the next stages.

Two of these stages are assigning the deal an ID according to European standards and sending a request to the investment network Liquidnet. And these actions are taken through integration with third-party services. The third is finalizing the document.

In addition, the questionnaire can be downloaded as a JSON file for reusing it in the system. And if the user needs to integrate the info with other applications, they can download it in CSV or other popular formats.

Last but far from least, users can collaborate with each other on documents using the comment feature from the Collaborations app. If someone has left a suggestion to a document, another user will see a notification in the respective column. Clicking on it will take the user to the comments section where they will see a proposed change and an accompanying comment.

The user can either accept the suggestion, reject it or propose an alternative.

Collaborations

After the questionnaire has been finalized and the execution version of it has been created, the submission on the list will receive a transaction title. By clicking on this title, the user will be redirected to the **Collaboration** application.

In this application, the user will see various automatically created folders named after entities like Dealers, Investors and Issuers. Each folder contains related documents, and the one that contains the new document that has just been created will be highlighted with a different color.

Now the user can view the document they have created and invite others to collaborate on it by discussing it in the comment section. With each new comment, a new version of the document is created.

Sometimes the user may need to share documents with people who are not registered in the system. And to do so, they can create a public folder in Collaborations, allowing anyone with the link to collaborate on documents.

Results

We built a stable, high-performing product with complex architecture in almost three years. As investment flowed in, our client felt their confidence grow, and ended up

building their own in-house team. Today, they are a successful company with a sizeable workforce distributed across Luxembourg, the United Kingdom, Portugal and France.