Report for Group Seashell

Analyzing the Trace Data

We started our work by creating an account with Celonis. To make collaboration easier, we invited team members into one workspace. From the Celonis data pool, we uploaded trace data, checked the data schema, and imported the data. In the data model configuration, we specified the case ID, activity name, and timeline columns. Then, in the Studio tab, we created a view and used the "Process Explorer" component to have a comprehensive view of the trace data.

Before exploring the map of the trace data, we experimented with the "Trace Data Activity Connections" slider. This allowed us to increase or decrease the level of detail in the connections. Increasing it added more edges (connections), while decreasing it simplified the process map by showing only the most common transitions. We wanted to include all edge-case scenarios in our visualization, so we used the maximum number of connections.

First, the process starts with 1,000 traces. We interpreted this to mean that InnoTrial had 1,000 contracts. For all those contracts, InnoTrial checks the database to see if they already have enough test subjects. In 403 cases, they had enough test subjects and directly informed CuraMedica without recruiting any. The other 597 cases did not have enough test subjects, which is why InnoTrial launched a recruitment campaign.

A closer examination of the recruitment campaign reveals the following sequence. After the campaign is launched, InnoTrial handles the applications that are received. Once all applications are collected, they conduct a pre-screening survey for each applicant. This is where we have a sub-process. In the pre-screening survey, an applicant can be either accepted or rejected. This sub-process ends with informing Curamedica if they have enough test subjects or launching a new recruitment campaign if they still don't. Finally, we have 35 exceptional contracts in which InnoTrial ended the process after recruitment. We assumed that these 35 cases took too long during the recruitment phase for the clinical trial.

After successful recruitment, InnoTrial starts conducting medical evaluations for each test subject. We interpret this as a loop in BPMN. This does not happen simultaneously for all participants. In fact, it happens in a sequential order. The trace data shows that the next participant can only be tested once the previous one has finished.

We took one participant's journey as an example to make the explanation as clear as possible. After the medical evaluation, a participant is checked for eligibility. If InnoTrial has enough data to conclude whether the participant is eligible or not, then they can continue to register them in the system. Otherwise, they will conduct another medical examination until they have enough data to conclude. After that, InnoTrial creates compensation agreements and consent forms, and informs the participant about the result of the evaluation. From this point on, a participant can be included or excluded from the trial. If the participant is included, InnoTrial archives the documents and creates a payment for the participant, who now becomes a test subject. After the last participant is checked, InnoTrial can proceed to the next steps where our trace data ends.

We also identified another exception. Trace data shows that 313 cases out of 1,000 prematurely ended right after checking the last person for eligibility. Usually, a participant would be informed about exclusion or inclusion. But in those 313 cases, the system stopped tracking. While we don't have a clear explanation as of why this is happening, we have some assumptions:

1) System failure, but this is very unlikely to have occurred because more than 30% of the traces experienced this issue. We assume that this should have already raised concerns in the company, as this percentage could lead to significant losses.
2) InnoTrial interrupted the process intentionally, for example a manual intervention. They could already have had enough test subjects, and there was no need for more.

We discussed how to model that exception. One possible way to represent it is to use an "Interrupting Boundary Event" on the sub-process or the task itself. This would allow us to break the loop if any abnormalities occur. As Signavio does not support this feature natively, we opted for another approach which is to add a condition after checking eligibility. This would allow us to break the sequence loop but not the whole process.

BPMN Modeling Changes

After we assessed the Celonis process map, we made revisions to the recruitment and medical evaluation parts of the BPMN model. Mostly, we focused on modularity and exception handling. Real trace data gave us a slightly different perspective. We outlined the differences between the old and the new BPMN models.

Based on the trace data, we analyzed that our BPMN model [Figure 1] aligned correctly with the trace data until the one-month wait timer. The trace data included a "Handle Applications" task, which we added to the revised model [Figure 2]. While our old model [Figure 1] appeared functional, the processes followed a linear flow. The model lacked modular structure, where each applicant was not handled individually. The repeated actions were modeled sequentially without a clear indication that they occurred per applicant. We didn’t have proper exception handling to address the scenario where the project is abandoned due to recruitment taking too long (based on our assumption).

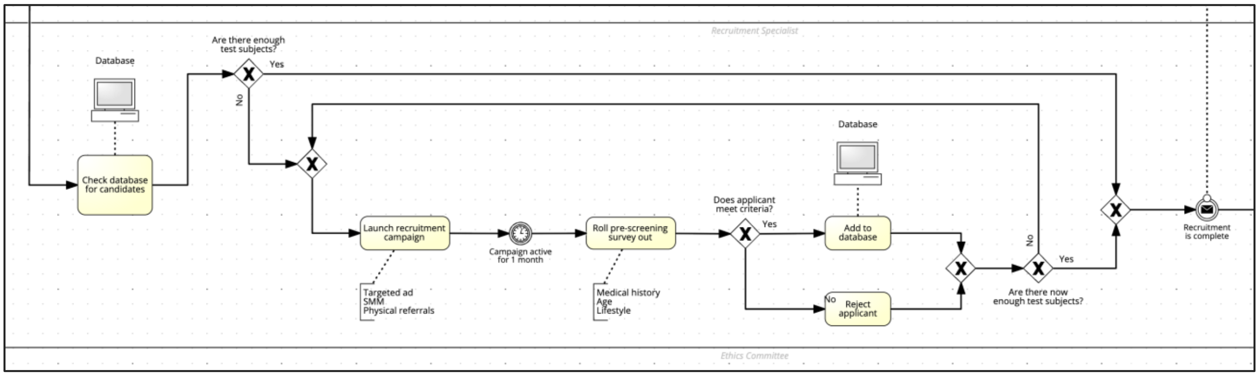
Recruitment Specialist
Database
Check database for candidates
Are there enough test subjects?
Yes
No
Launch recruitment campaign
Targeted ad
SMM
Physical referrals
Campaign active for 1 month
Roll pre-screening survey out
Medical history
Age
Lifestyle
Does applicant meet criteria?
Yes
No
Database
Add to database
Reject applicant
No
Yes
Are there now enough test subjects?
Recruitment is complete
Ethics Committee

Figure 1. Old BPMN Model for Recruitment Process

In the revised model [Figure 2], we introduced a multi-instance sequential sub-process called “Participant Pre-Screening.” It indicates that the tasks inside the sub-process are executed once per applicant, as the timeline in the trace data showed that a new applicant is checked only after the previous one finishes the survey. The loop is maintained until the last available applicant is checked. To handle the previously mentioned exception, we created a gateway to check whether the project remains within scope. The project could be abandoned if recruitment extends beyond a reasonable time, and InnoTrial will send a message to CuraMedica about the project’s termination.

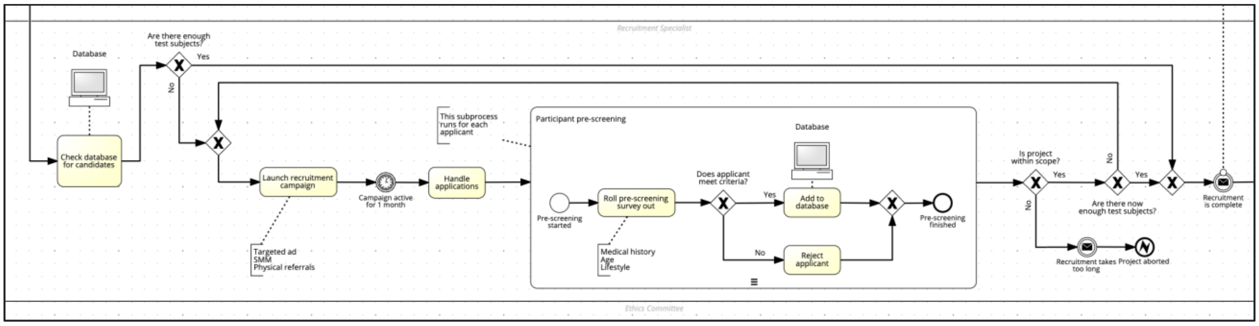
Recruitment Specialist
Database
Check database for candidates
Are there enough test subjects?
Yes
No
Launch recruitment campaign
Targeted ad
SMM
Physical referrals
Campaign active for 1 month
Handle applications
This subprocess runs for each applicant
Participant pre-screening
Pre-screening started
Roll pre-screening survey out
Medical history
Age
Lifestyle
Does applicant meet criteria?
Yes
No
Database
Add to database
Reject applicant
Pre-screening finished
Is project within scope?
Yes
No
No
Yes
Are there now enough test subjects?
Recruitment is complete
Recruitment takes too long
Project aborted
Ethics Committee

Figure 2. New BPMN Model for Recruitment Process

After the recruitment process was completed, we moved on to the next phase, which was medical evaluation and participant onboarding. In the old model [Figure 3], we encountered the same modularity issue as in the recruitment process [Figure 1], where it was not clearly indicated that the process is executed for each participant. Candidate eligibility checks and the decision to include or exclude participants in the trial followed directly after medical evaluations. It was also somewhat confusing whether documents should be sent for excluded candidates, as the processes of informing excluded and included candidates converged again before the documents were sent. This also applied to the initial payment, as it was unclear whether excluded candidates received initial payments.

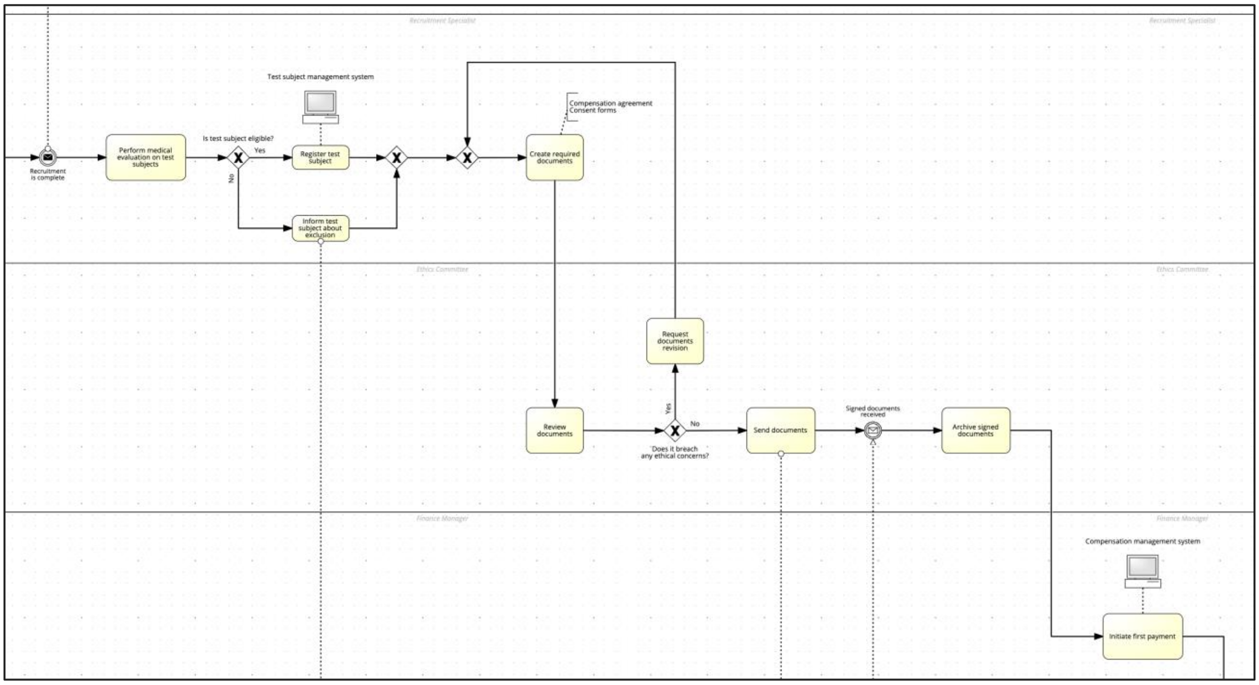
Recruitment Specialist
Recruitment Specialist
Test subject management system
Recruitment is complete
Perform medical evaluation on test subjects
Is test subject eligible?
Yes
No
Register test subject
Inform test subject about exclusion
Compensation agreement
Consent forms
Create required documents
Ethics Committee
Ethics Committee
Request documents revision
Review documents
Yes
No
Does it breach any ethical concerns?
Send documents
Signed documents received
Archive signed documents
Finance Manager
Finance Manager
Compensation management system
Initiate first payment

Figure 3. Old BPMN Model for Medical Evaluation and Participant Onboarding

In the revised model [Figure 4], we first introduced a multi-instance sequential sub-process to encapsulate the entire medical evaluation and onboarding process. It begins with a clearly defined event, and we introduced a condition that checks whether there is enough data to determine a candidate's eligibility. If not, the model loops back to allow for additional medical evaluations. We added this because it aligns with the trace data, which showed repeating medical evaluations for the same candidate. Moreover, we register test subjects and create documents before informing candidates, as the trace data indicates. Documents are only archived if the test subject is included. Those who are excluded terminate the current instance of the loop with a "Candidate Excluded" event. Unlike the old model [Figure 3], we reassigned the task of sending documents to the Recruitment Specialists rather than the Ethics Committee. This is not explicitly stated in the trace data, but it is our logical assumption. In our revised sub-process [Figure 4], it is clearly shown that only included candidates receive initial payments.

Regarding the exception identified at the "Check for Eligibility" task, we conducted a detailed analysis of the trace data and observed that in the 313 cases that ended at this point, more than 80% of the evaluated participants had already been included in the clinical trial. This strong majority led us to reasonably assume that InnoTrial had reached its recruitment target and any additional test subjects are no longer required. To reflect this behavior in the model, we introduced a conditional flow immediately following the “Check for Eligibility” task in the new model [Figure 4]. This condition checks whether the recruitment quota has been met or not. If so, it allows the “Medical Evaluation and Participant Onboarding” subprocess to terminate early. As a result, the process transitions directly to the next phase, which is administering the drugs, without processing further candidates unnecessarily. This modeling approach aligns with the observed data.

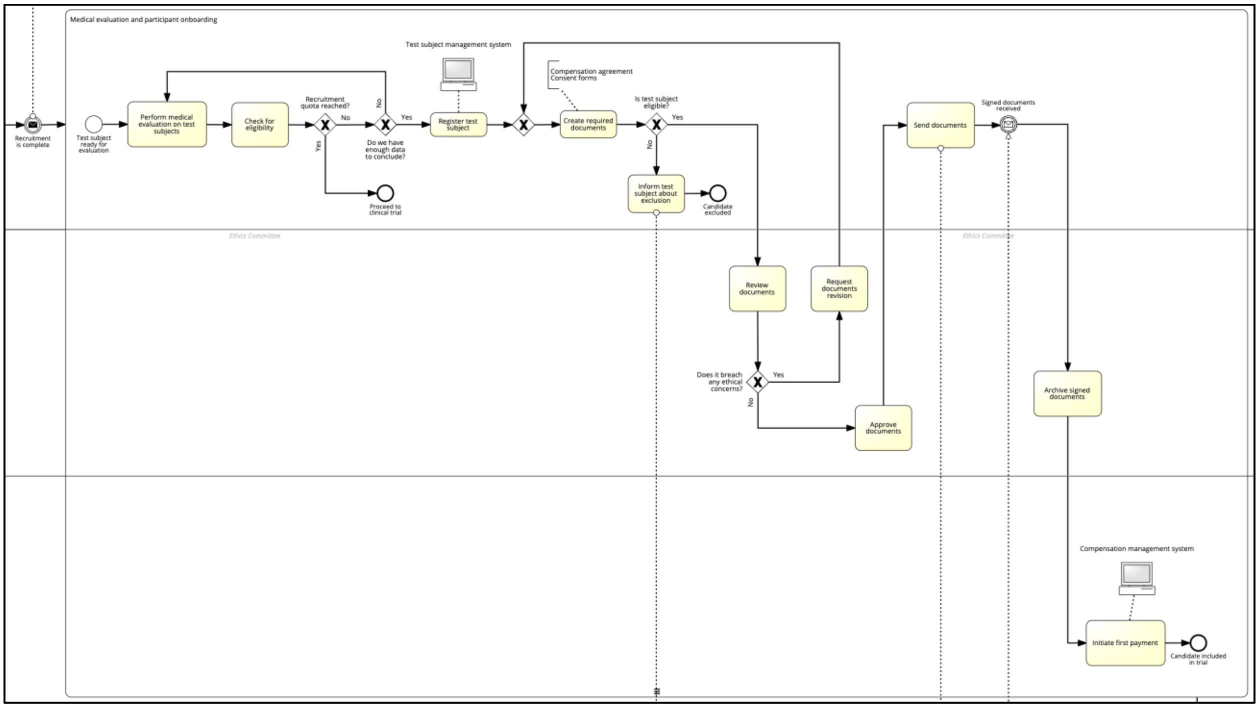
Medical evaluation and participant onboarding
Recruitment is complete
Test subject ready for evaluation
Perform medical evaluation on test subjects
Check for eligibility
Recruitment quota reached?
No
Yes
No
Yes
Do we have enough data to conclude?
Proceed to clinical trial
Test subject management system
Register test subject
Compensation agreement
Consent forms
Create required documents
Is test subject eligible?
Yes
No
Inform test subject about exclusion
Candidate excluded
Send documents
Signed documents received
Ethics Committee
Ethics Committee
Review documents
Request documents revision
Does it breach any ethical concerns?
Yes
No
Approve documents
Archive signed documents
Compensation management system
Initiate first payment
Candidate included in trial

Figure 4. New BPMN Model for Medical Evaluation and Participant Onboarding

We encountered several issues while redesigning the model. One of the challenges was having a sub-process that spans multiple swimlanes. We tried to use the “Event Subprocess” in Signavio but found that it cannot span across multiple swimlanes. The next option was to use grouping under the artifacts section, but this option was not suitable because it did not support looping. In the end, we used an expanded sub-process but initially struggled to understand how Signavio handles it.

In summary, the new version of the model provides improved logic, better exception handling, and clearer modularity, which aligns more closely with the trace data.