Lab Guide

Using Task Mining for Discovery, Monitoring, and Analysis of User Interactions on Desktops

Part 1 of 2

V 1.14.1

Lab Version 1.0

Paul Pacholski

pacholsk@ca.ibm.com

Daniele Ingrami

Daniele.Ingrami@ibm.com

Table of Contents

1 Introduction

1.1 Task Mining in IBM Process Mining

Task Mining is the discovery, monitoring, and analysis of user interaction data on desktops through the recording of frontend activities.

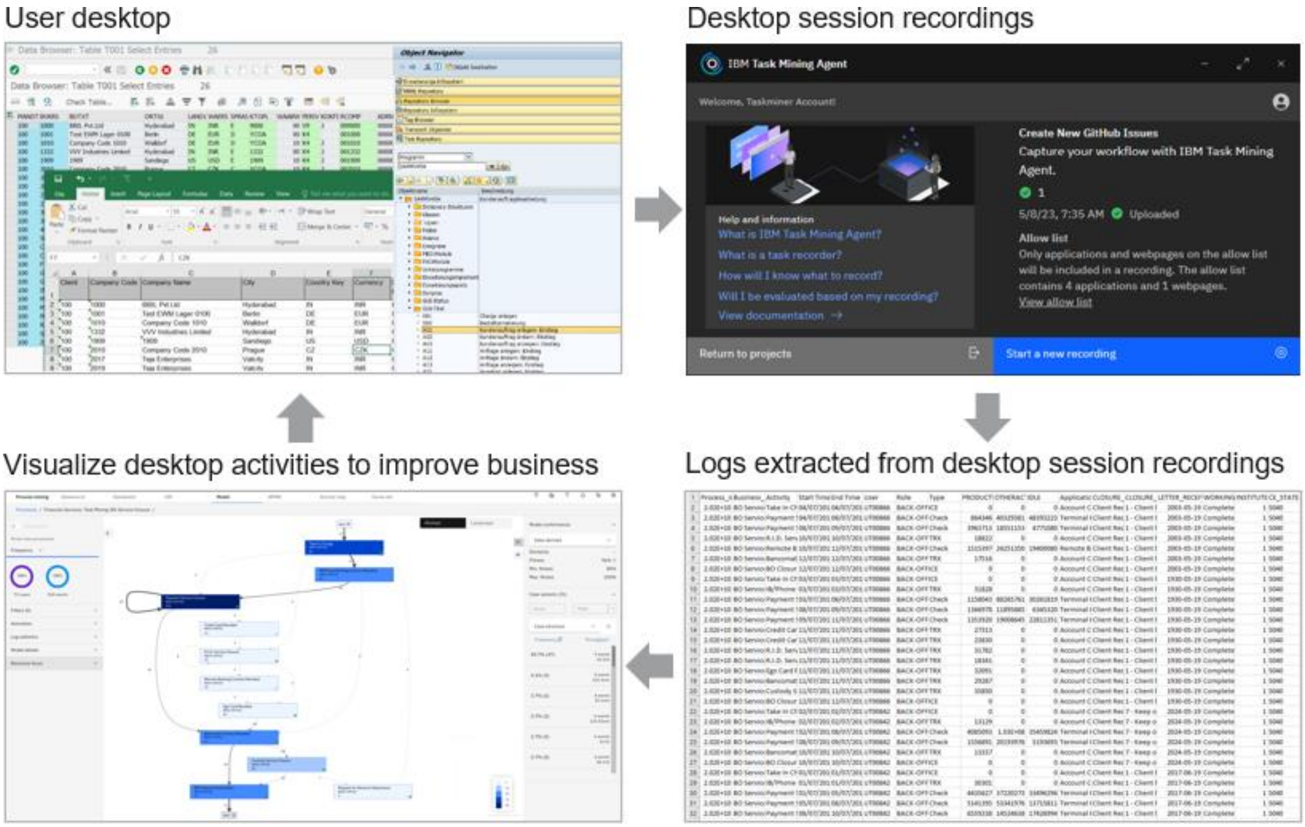

User desktop
Desktop session recordings
Visualize desktop activities to improve business
Logs extracted from desktop session recordings
IBM Task Mining Agent
Welcome, Taskminer Account!
Help and information
What is IBM Task Mining Agent?
What is a task recorder?
How will I know what to record?
Will I be evaluated based on my recording?
View documentation →
Create New GitHub Issues
Capture your workflow with IBM Task Mining Agent.
1
5/8/23, 7:35 AM
Uploaded
Allow list
Only applications and webpages on the allow list will be included in a recording. The allow list contains 4 applications and 1 webpages.
View allow list
Return to projects
Start a new recording

Figure 1. Task Mining

Here are the key insights obtainable from task mining:

Productivity - You can discover how much time users allocate to the Process and how idle the activities are because of context switches. For example, users stopped working on the activity to work on something else.

Precisely calculate the costs of your Process based on the productive time of your resources on the Process.

Understand which applications users are working on the most.

Working Patterns - You can discover the main patterns of performing a business activity and the most efficient ways to complete the activity by identifying deviations and inefficiencies.

Set the most efficient patterns as best practices for the employees.

Understand the root causes of inefficiencies and take action to solve them.

Automation Opportunities - You can discover the working patterns to be automated, with the best tradeoff between benefits and complexity.

Simulate the automation of the most suitable working patterns and verify performance and cost benefits.

Getting a complete picture of the Process through the combination of business data and user interaction data creates the complete picture of the Process, which can be analyzed from the business and task levels.

1.2 Task Mining Programming Model An Introduction

The IBM Task Mining architecture includes four components—TM Web, TM Agent, TM Backend, and Persistency Layer—that record, collect, store, and process data. The architecture ensures data security offered by the Persistency Layer that enables 256-bit Key Encryption. In addition, it transforms and loads the data from IBM Task Mining into IBM Process Mining.

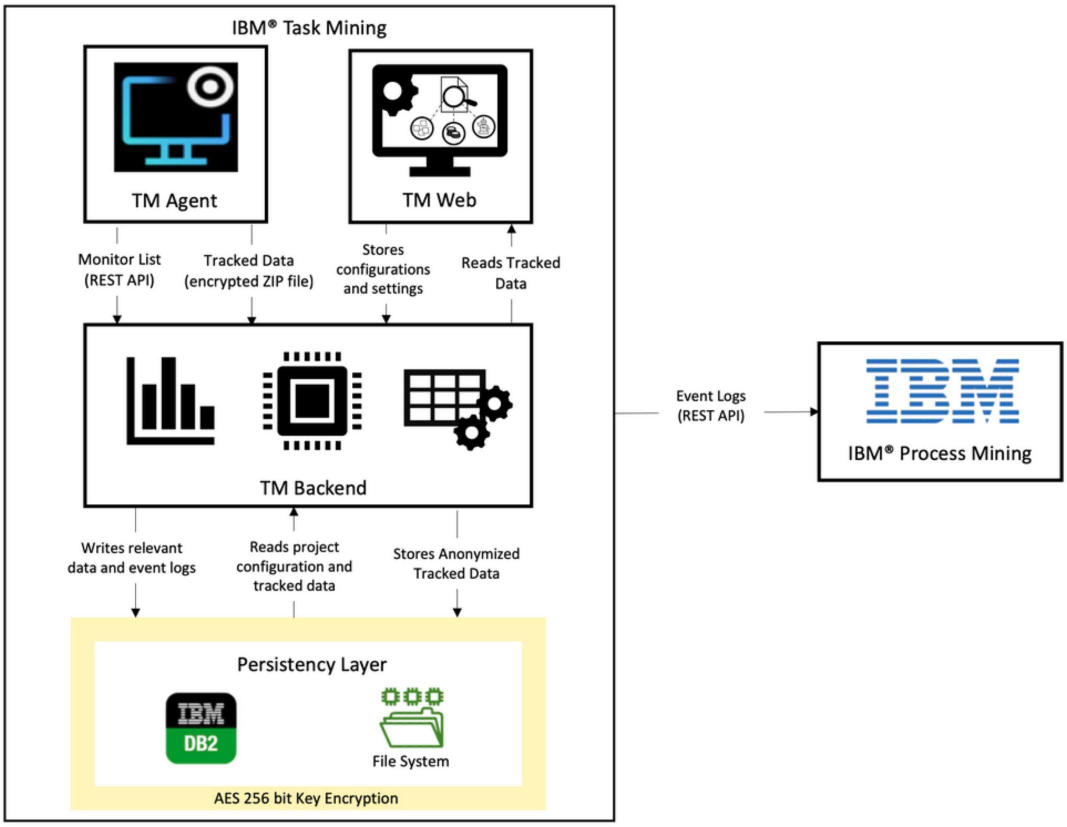

IBM® Task Mining
TM Agent
TM Web
Monitor List (REST API)
Tracked Data (encrypted ZIP file)
Stores configurations and settings
Reads Tracked Data
TM Backend
Event Logs (REST API)
IBM
IBM® Process Mining
Writes relevant data and event logs
Reads project configuration and tracked data
Stores Anonymized Tracked Data
Persistency Layer
IBM DB2
File System
AES 256 bit Key Encryption

Figure 2. Components and Data Flow in IBM Task Mining

Component	Abbreviation	Description
Task Mining Web	TM Web	The web-based user interface (WUI) to configure settings in IBM Task Mining, add or create a process for Task Mining, manage users, interact with the acquired data, and view insights on the user interaction on the system
Task Mining Agent	TM Agent	The user interface to track, record, and save the on-screen activities on any selected application in the system
Task Mining Backend	TM Backend	The backend component collects data from the TM Agent and processes the data to transform it into useful information for IBM Process Mining.
Persistency Layer		Stores and protects the data with 256-bit Key Encryption.

The Persistence Layer in TM Backend receives data from the Task Mining Agent (TM Agent) and stores them in the filesystem. It works as a central hub for all the clients and provides the required services using Rest API.

The TM Backend processes the data logged by the TM Agent in Persistency Layer according to the configuration meta-data. It helps to correlate user actions to business transactions and extract the relevant data required for process mining and analysis. TM Backend then processes the generated data to create the event logs uploaded to IBM Process Mining using REST API over HTTPS.

1.3 Lab Introduction

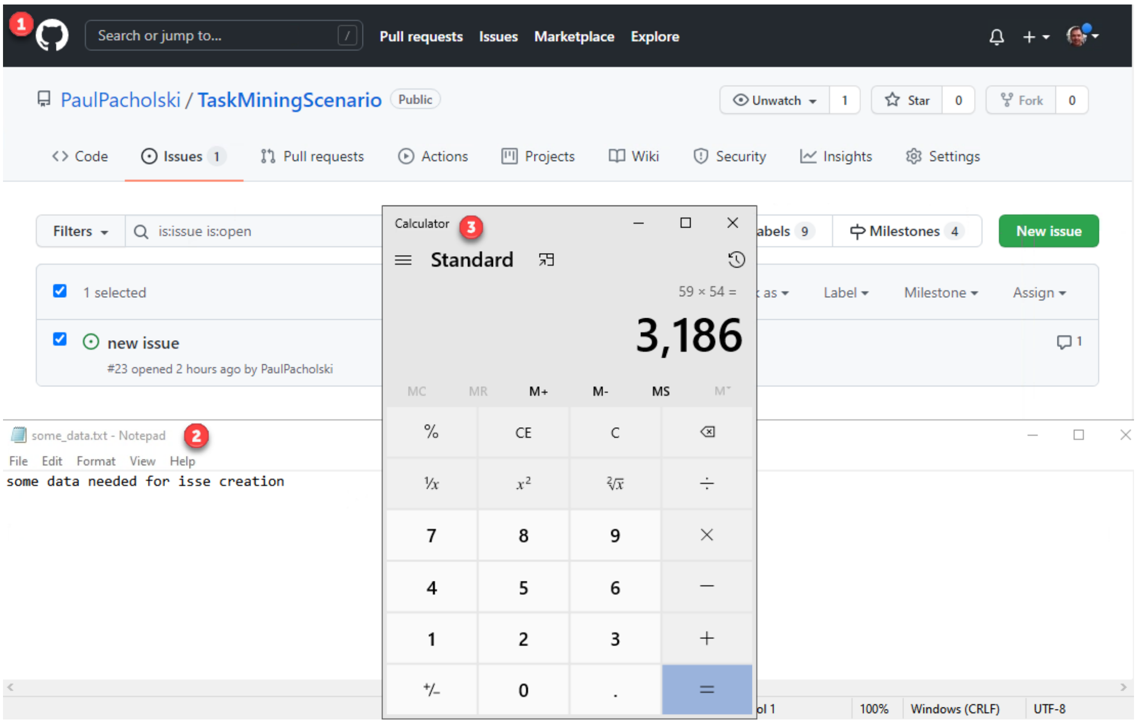
1.3.1 Business Scenario

Opening the GitHub Issue is a critical desktop activity. It is a swivel-chair activity in an automated process. The business owners of the automated Process want to improve lead time and costs associated with the Open GitHub Issue activity.

In this lab, we will use Task Mining to conduct a time study and cost analysis of the desktop polycations and activities involved in the Opening GitHub Issues activity.

There are three desktop applications involved in creating GitHub Issues:

1. GitHub web applications
2. Notepad
3. Desktop Calculator



59 × 54 =

3,186

```
some data needed for isse creation
```

Figure 3. Desk Applications

1.3.2 Lab Objectives

This lab will focus on how the Task Mining feature is used, not how the TM model was developed. In the second lab of this lab series, you will learn how the Task Mining project was built and configured!

In this lab, we will use a pre-built Task Mining project already configured to receive and process events from our GitHub Issues desktop environment.

Here are the critical lab steps:

1. First, you will record opening a GitHub issue in the Windows desktop and examine the recorded desktop events in the Process Mining model.
2. Next, you will examine the Process Model that includes previously recorded Cases and the new Case you just recorded.
3. Finally, you will examine a custom Dashboard showing a summary of the desktop activities involved in creating GitHub Issues.

2 Lab Instructions

2.1 Lab Setup

2.1.1 GitHub

To complete this lab, you must get a GitHub user id for https://www.github.com.

2.1.2 Setup IBM Process and Task Mining Environment

_1. Download the IBM Process Mining and Task Mining Environment document.

_2. Follow the instructions in **2.4.2 Accessing Task Mining Client VM Using RDP.**

2.2 Record Opening of a GitHub Issue

We have created a Task Mining model to monitor desktop events generated when opening GitHub Issues. The Task Mining Agent installed on the Windows 10 desktop will record events generated by these three desktop applications: Web browser (github.com), Notepad, and Calculator. All these applications are used to open GitHub Issues.

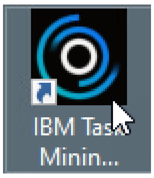
The IBM Task Mining Agent will send the events to the Task Mining model. You will examine this Model later in this lab, and if you like, you can try lab 2 to learn how to build it.

The Task Mining model will aggregate and process the events received from the IBM Task Mining Agent and send them to the Process Mining model as CSV files.

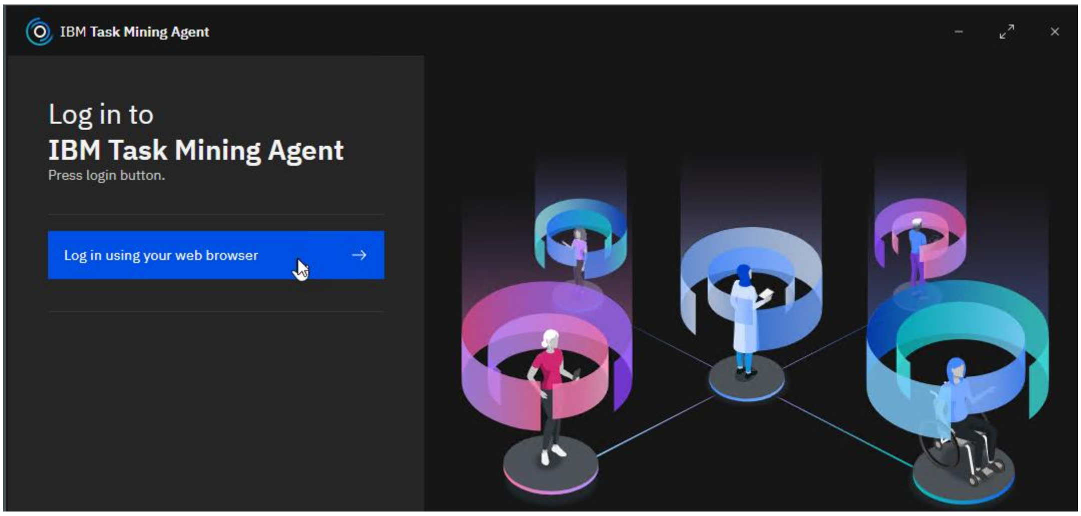
2.2.1 Start IBM Task Mining Agent

_1. Switch to **IBM Task Mining Client VM** RDP desktop.

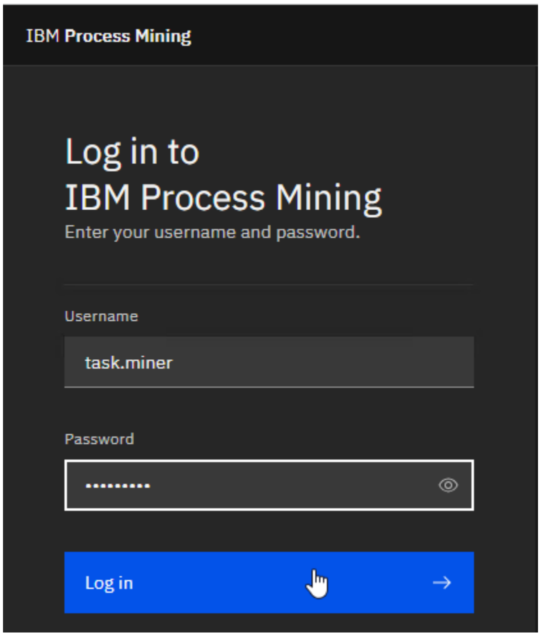
_2. On the desktop, double-click **IBM Task Mining Agent.**



_3. Click **Log in using your web browser.**



_4. On the *Log in window*, enter the credentials of **task.miner / IBMDem0s!** and click **Log in**



IBM Process Mining

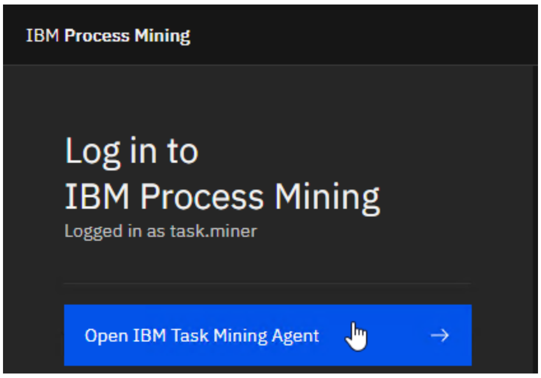
Log in to
IBM Process Mining

Enter your username and password.

Username: task.miner

Password: ••••••••••

_5. Click **Open IBM Task Mining Agent.**



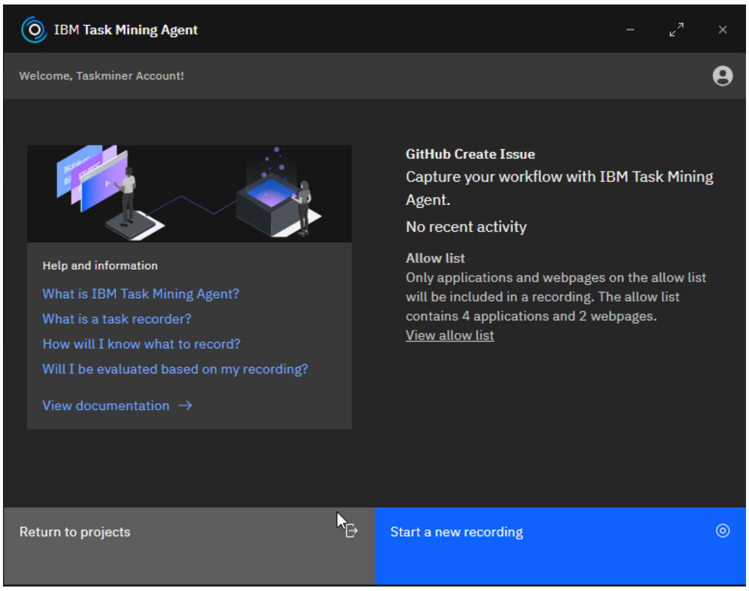
IBM Process Mining

Log in to
IBM Process Mining

Logged in as task.miner

_6. **Close** web browser

You should now see the Task Mining Agent window.



IBM Task Mining Agent

Welcome, Taskminer Account!

Help and information

What is IBM Task Mining Agent?

What is a task recorder?

How will I know what to record?

Will I be evaluated based on my recording?

View documentation →

GitHub Create Issue

Capture your workflow with IBM Task Mining Agent.

No recent activity

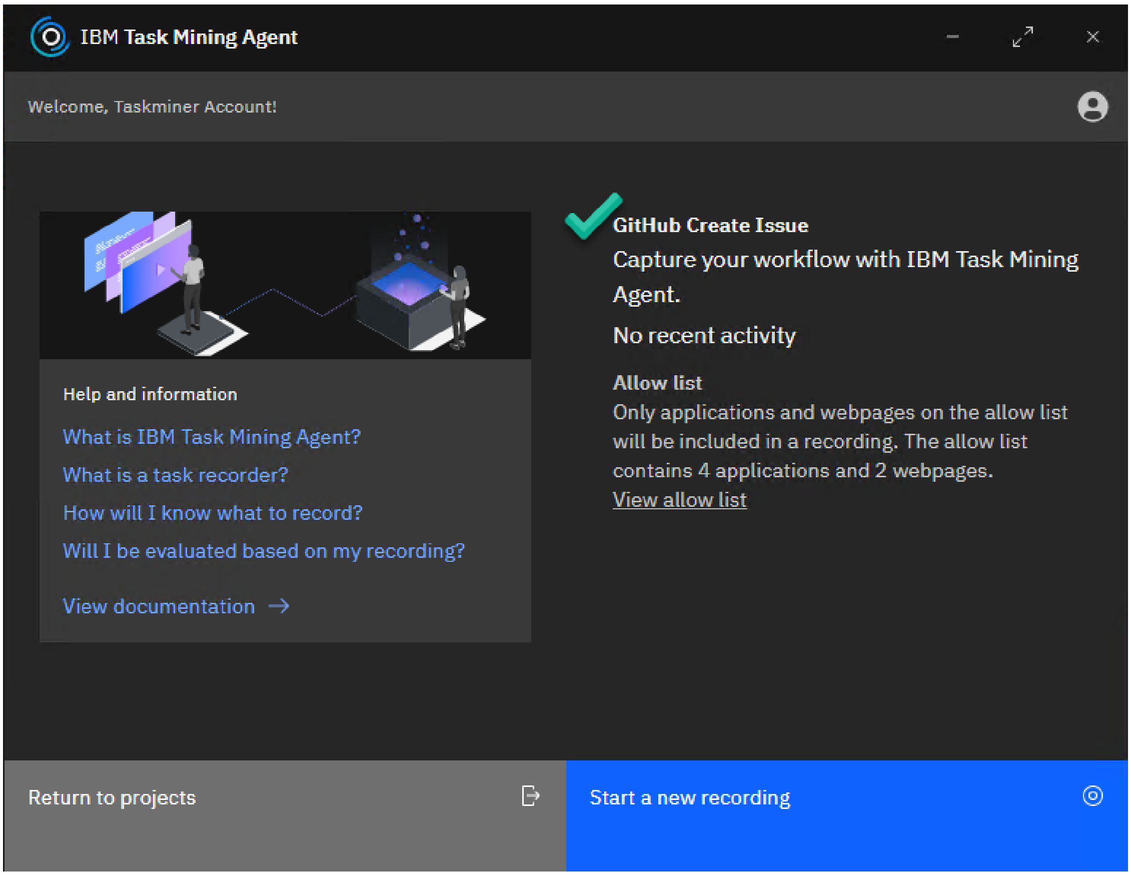
Allow list

Only applications and webpages on the allow list will be included in a recording. The allow list contains 4 applications and 2 webpages.

View allow list

2.2.2 Examine IBM Task Mining Agent

_1. In the **IBM Task Mining Client VM**, switch to the **IBM Task Mining Agent** window.



IBM Task Mining Agent

Welcome, Taskminer Account!

Help and information

What is IBM Task Mining Agent?

What is a task recorder?

How will I know what to record?

Will I be evaluated based on my recording?

View documentation →

✓ GitHub Create Issue

Capture your workflow with IBM Task Mining Agent.

No recent activity

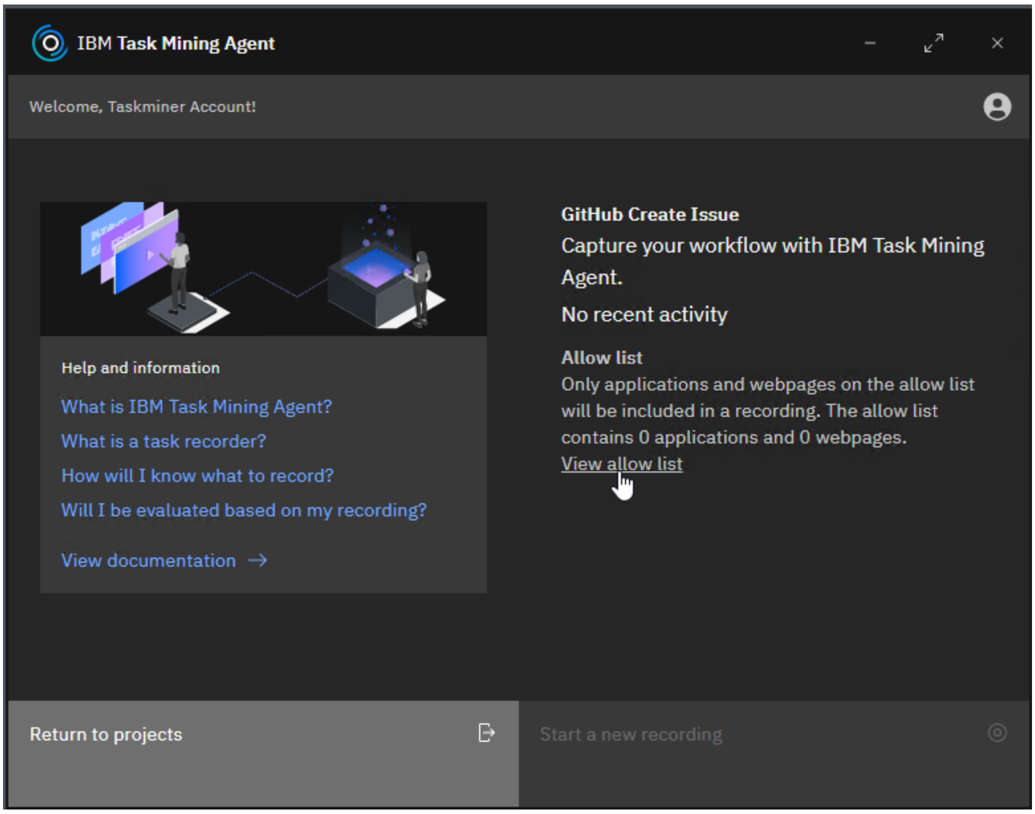
Allow list

Only applications and webpages on the allow list will be included in a recording. The allow list contains 4 applications and 2 webpages.

View allow list

Note that the *GitHub Issue Insert* Task Mining project has already been selected. The events recorded by the IBM Task Mining Agent will be sent to this Task Mining project.

_2. Click **View allow list**



IBM Task Mining Agent

Welcome, Taskminer Account!

Help and information

What is IBM Task Mining Agent?

What is a task recorder?

How will I know what to record?

Will I be evaluated based on my recording?

View documentation →

GitHub Create Issue

Capture your workflow with IBM Task Mining Agent.

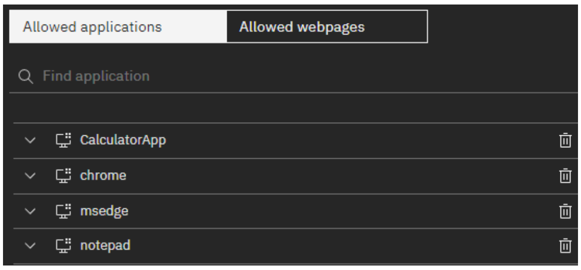
No recent activity

Allow list

Only applications and webpages on the allow list will be included in a recording. The allow list contains 0 applications and 0 webpages.

View allow list

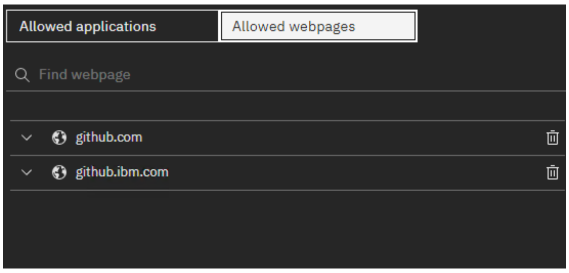
Note that the URLs and applications required to record GitHub Issues opening desktop tasks have already been selected for you.



Allowed applications | Allowed webpages

Find application

- CalculatorApp
- chrome
- msedge
- notepad

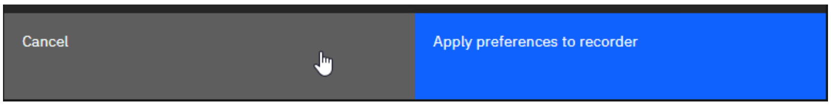


Allowed applications | Allowed webpages

Find webpage

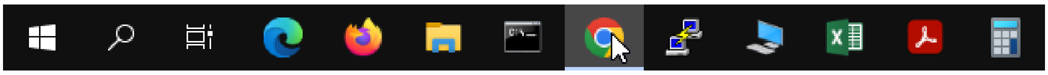
- github.com
- github.ibm.com

_3. Click **Cancel** to close the *Monitoring List* window.

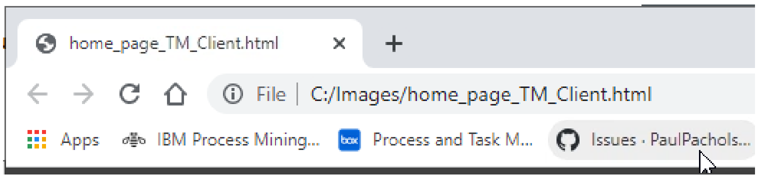


2.2.3 Start GitHub Web application

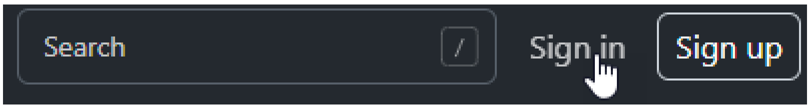
_1. Start **Google Chrome** web browser



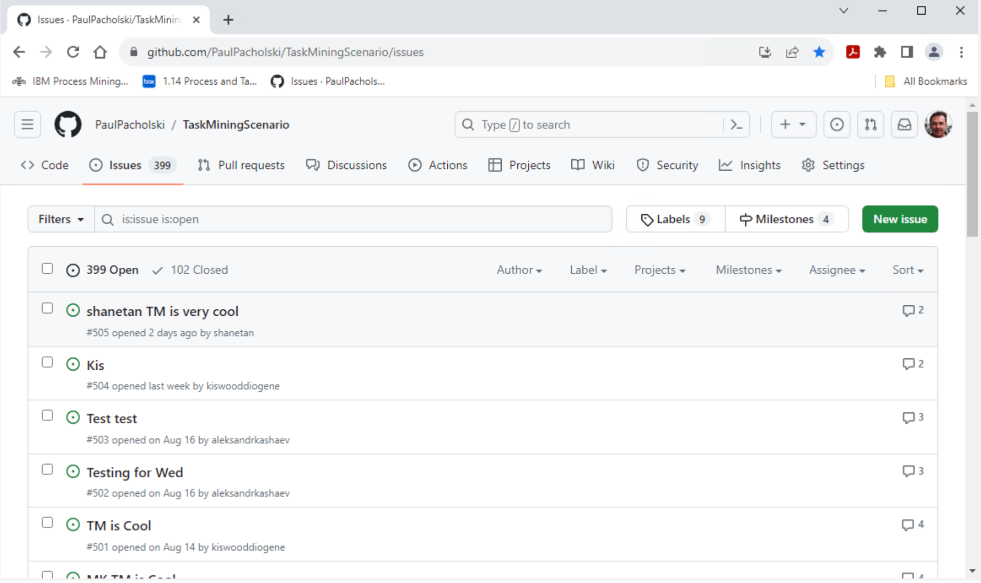
_2. Click **Issues. PaulPachols…** bookmark



_3. Click **Sign In** and sign in with your GitHub user id and password.



Note that you may see a lot of open issues in this GitHub project!



□ 399 Open ✓ 102 Closed

□ shanetan TM is very cool — 2
#505 opened 2 days ago by shanetan

□ Kis — 2
#504 opened last week by kiswooddiogene

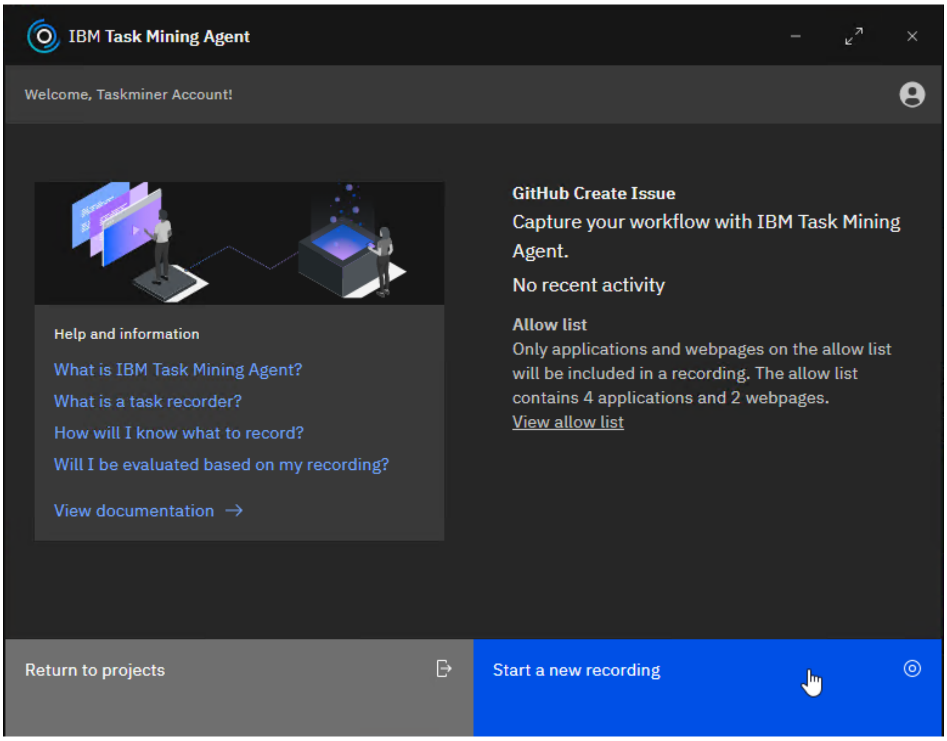
□ Test test — 3
#503 opened on Aug 16 by aleksandrkashaev

□ Testing for Wed — 3
#502 opened on Aug 16 by aleksandrkashaev

□ TM is Cool — 4
#501 opened on Aug 14 by kiswooddiogene

2.2.4 Record Desktop Events

_1. Switch to the *IBM Task Mining Agent* window and click **Start a new recording.**

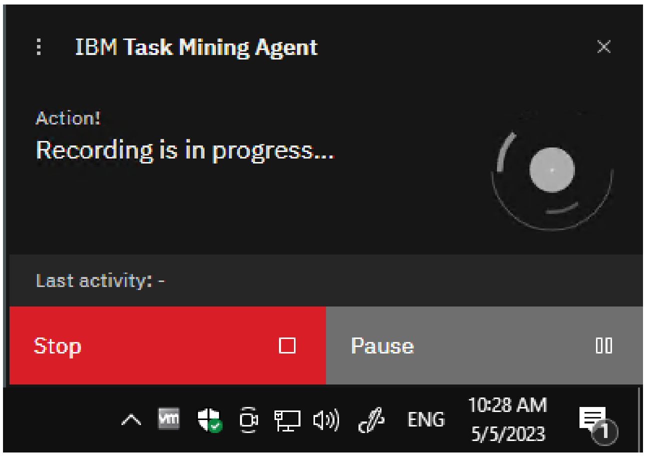


GitHub Create Issue
Capture your workflow with IBM Task Mining Agent.
No recent activity

Allow list
Only applications and webpages on the allow list will be included in a recording. The allow list contains 4 applications and 2 webpages.

The IBM Task Mining Agent is now ready to send all recorded desktop events to the **GitHub Create Issue** TM project we have created for you.

_2. Wait until you see the **Recording in progress...**



IBM Task Mining Agent

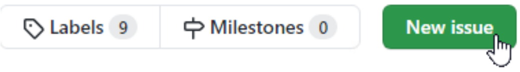
Action!

Recording is in progress...

Last activity: -

_3. Switch to **Chrome** Web browser.

_4. Click the **New issue** button.

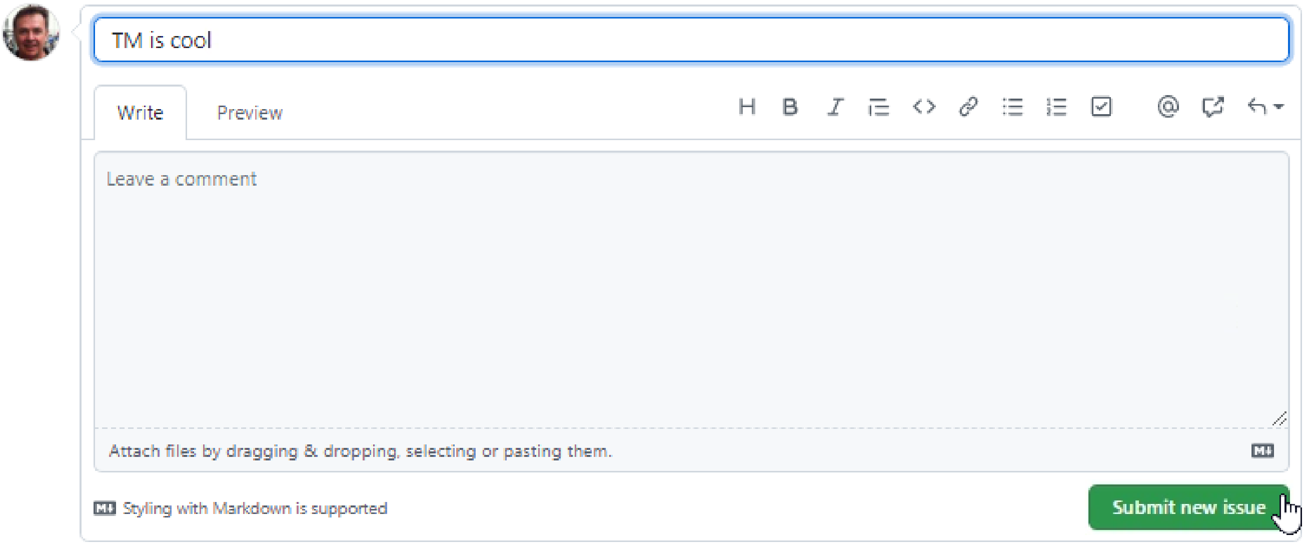


_5. For *Title* enter, **TM is cool**

> Note: since you are using a shared environment when creating a new issue, please use your user name as a prefix in the issue title. For example, **<user-id> TM is cool**

_6. **Wait** 5 seconds

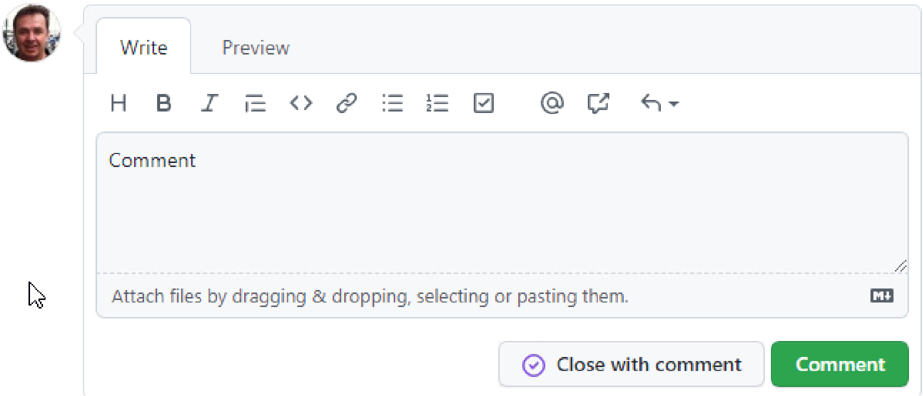
_7. Click **Submit new issue** button



TM is cool

Write
Preview
Leave a comment
Attach files by dragging & dropping, selecting or pasting them.
Styling with Markdown is supported
Submit new issue

_8. Type **Comment** in the comment area


Write
Preview
Attach files by dragging & dropping, selecting or pasting them.
Close with comment
Comment

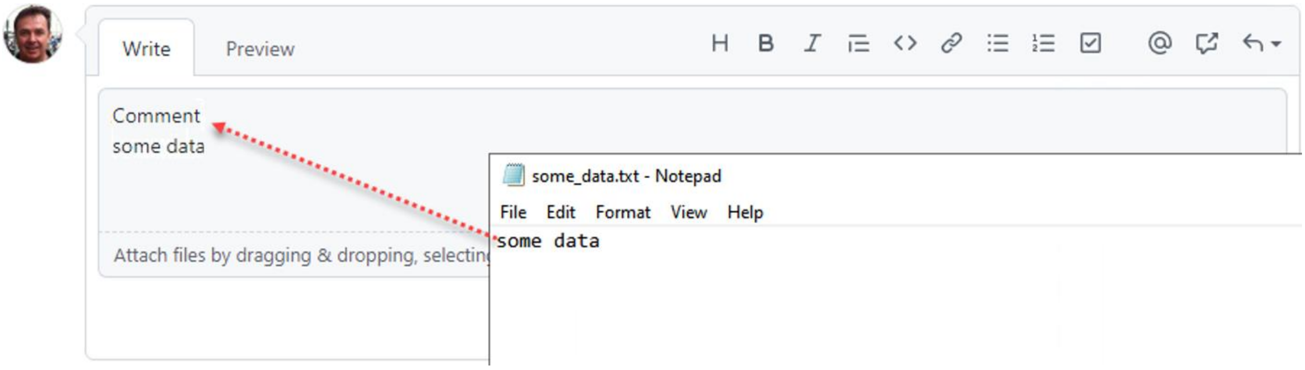
Comment

_9. Double-click **some_data...** document icon on the desktop

_4. **Copy** "some data" from Notepad to the clipboard.

_5. **Wait** 20 seconds.

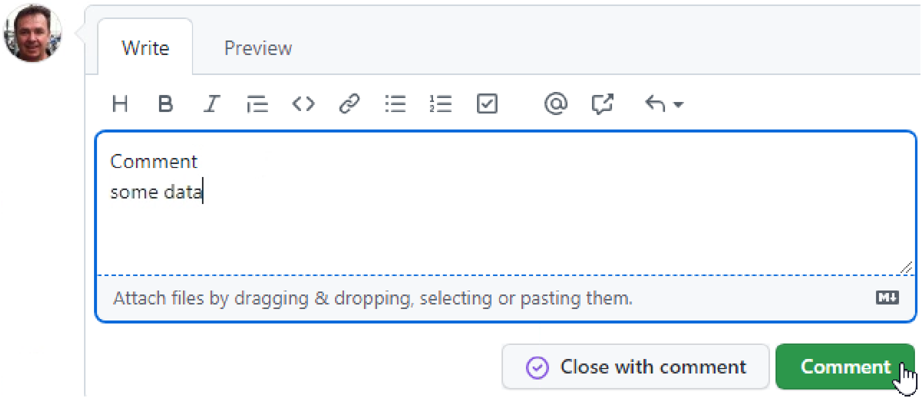
_6. Switch to the **GitHub Issues web page** and then **paste from the clipboard** to the *Comment section* on GitHub.



Comment
some data

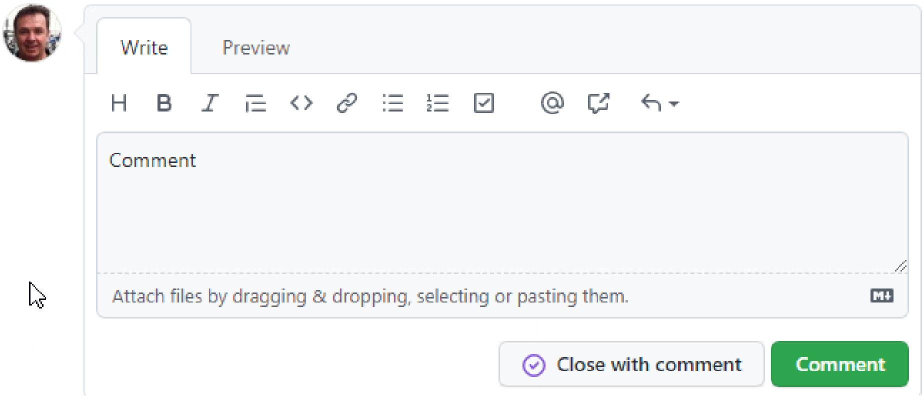
some data

_7. Click the **Comment** button.


Write
Preview
Attach files by dragging & dropping, selecting or pasting them.
Close with comment
Comment

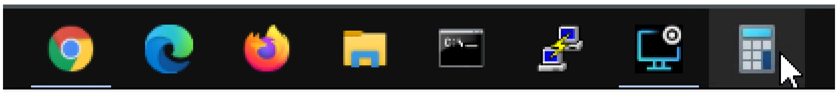
Comment
some data

_8. Type **Comment** in the comment area

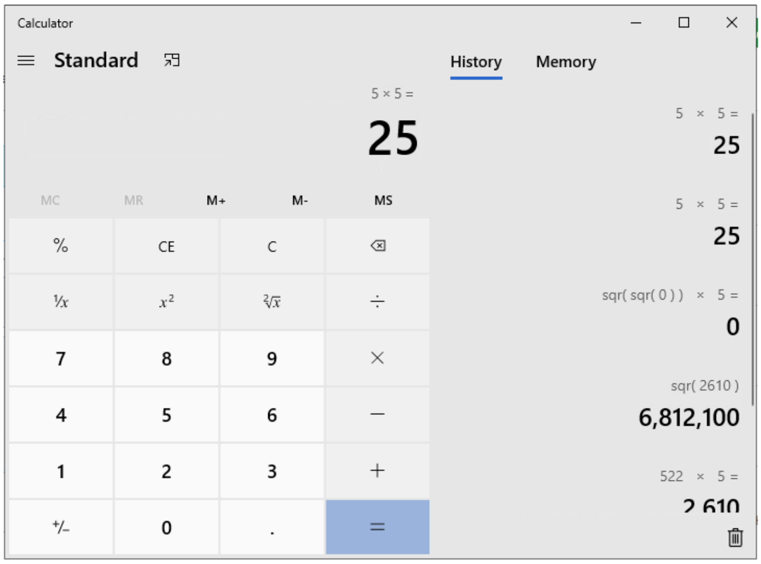

Write
Preview
Attach files by dragging & dropping, selecting or pasting them.
Close with comment
Comment

Comment

_9. From the *Windows Task Bar,* click **Calculator.**



_10. Use the Calculator app to perform **several (4-5) calculations** (for example, 5 x 5 = 25)



5 × 5 =
25

5 × 5 =
25

5 × 5 =
25

sqr(sqr(0)) × 5 =
0

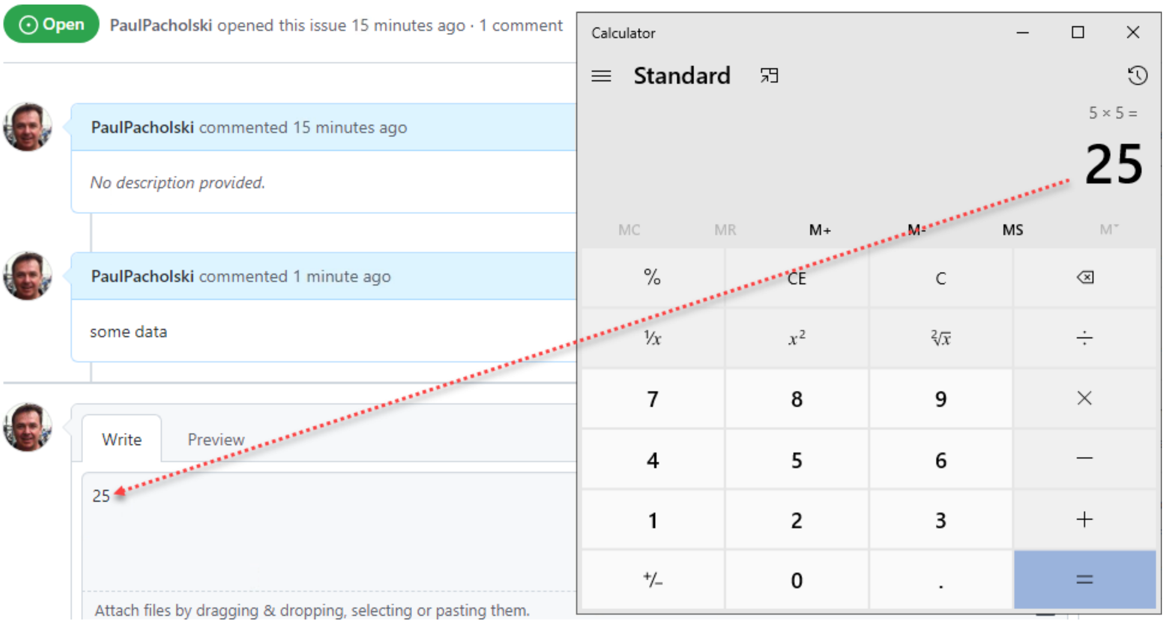
sqr(2610)
6,812,100

522 × 5 =
2,610

_11. **Copy** the result from the Calculator to the clipboard

_12. **Wait** 20 seconds

_13. Switch to the **GitHub Issues web page** and then **paste from the clipboard** to the *Comment section* on GitHub.


PaulPacholski opened this issue 15 minutes ago · 1 comment

PaulPacholski commented 15 minutes ago

No description provided.

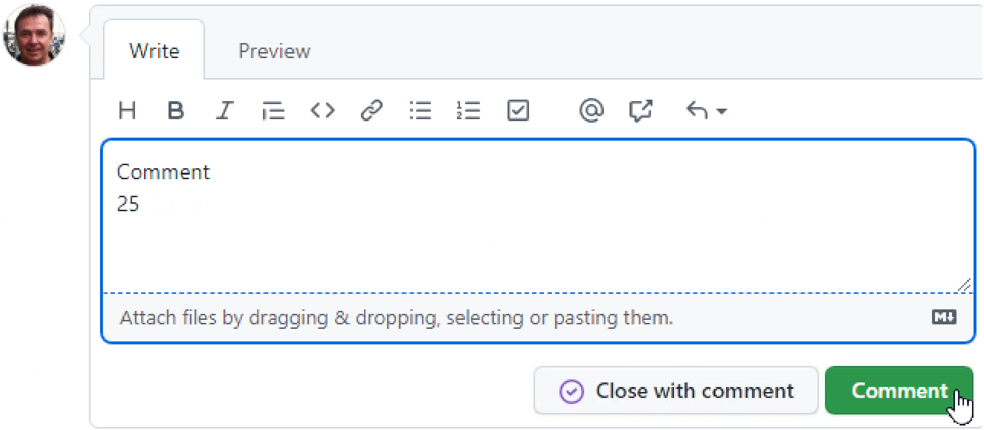
PaulPacholski commented 1 minute ago

some data

25

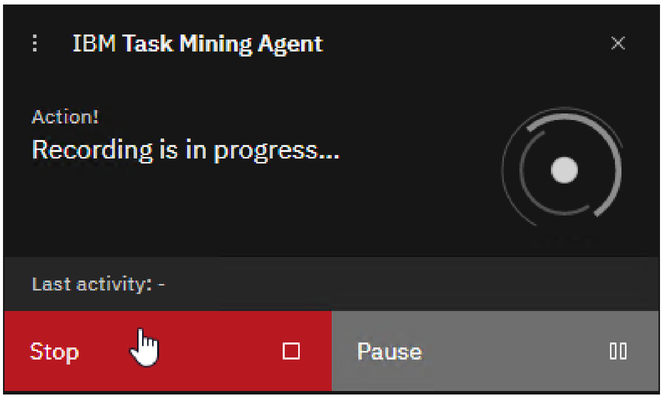
Attach files by dragging & dropping, selecting or pasting them.

_14. Click **Comment** button


Comment
25

Attach files by dragging & dropping, selecting or pasting them.

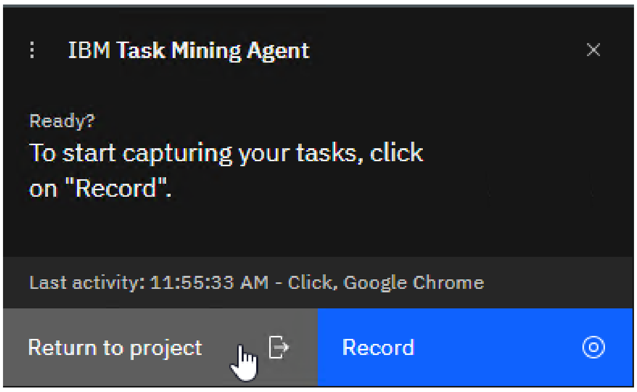
_15. In *IBM Task Mining Agent* window, click **Stop.**


Action!

Recording is in progress...

Last activity: -

_16. In *IBM Task Mining Agent* window, click **Return to project**.



IBM Task Mining Agent

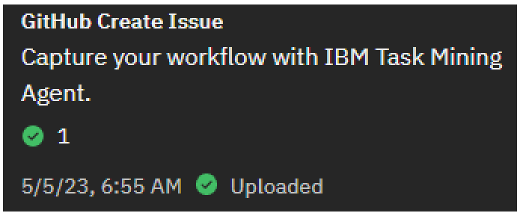
Ready?

To start capturing your tasks, click on "Record".

Last activity: 11:55:33 AM - Click, Google Chrome

_17. Click **Return to Project**

You should now see a successful upload message.



GitHub Create Issue

Capture your workflow with IBM Task Mining Agent.

✓ 1

5/5/23, 6:55 AM ✓ Uploaded

2.3 Examine the Task Mining Model

In the last steps, you recorded desktop activities in opening a new GitHub issue. The events the IBM Task Mining Agent recorded were sent to the TM model, which converted them to CSV files and then forwarded them to a PM model.

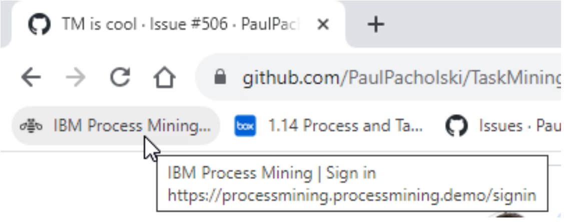
This part of the lab will examine the PM project that displays the recorded desktop events in a process mining format.

Since we have already recorded several Cases of opening GitHub Issues, when you open the Process Mining project, you will also see your Case and what has already been recorded.

2.3.1 Open Process Mining Project

_1. **Maximize** Chrome web browser and **maximize** it.

_2. Click **IBM Process Mining...** bookmark

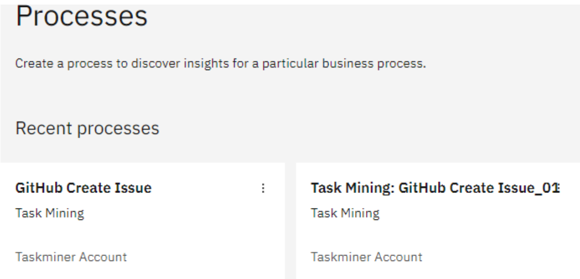


TM is cool · Issue #506 · PaulPac
github.com/PaulPacholski/TaskMining
IBM Process Mining...
1.14 Process and Ta...
Issues · Pau

IBM Process Mining | Sign in
https://processmining.processmining.demo/signin

_3. Enter the credentials of **task.miner** and **IBMDem0s!** and click **Log in.**

You should now see two new processes.



Processes

Create a process to discover insights for a particular business process.

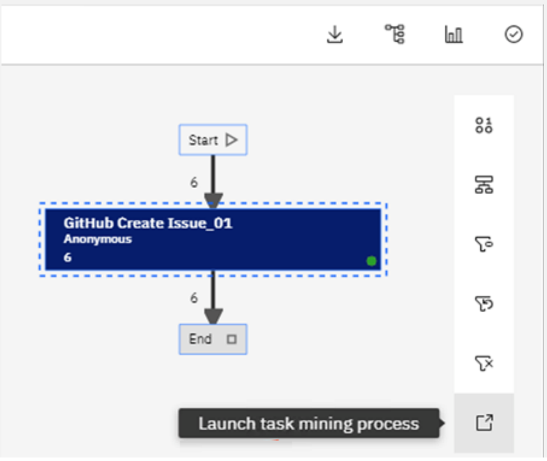
Recent processes

GitHub Create Issue ⋮
Task Mining
Taskminer Account

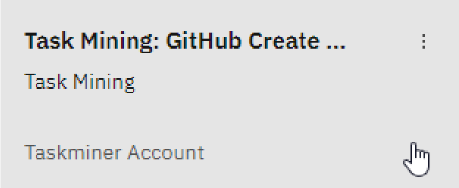
Task Mining: GitHub Create Issue_01
Task Mining
Taskminer Account

What are these two processes, and how are they related?

The **GitHub Create Issue** is the parent Process Mining process. The **Task Mining: GitHub Create Issue_01** is the "child" Task Mining process. Typically a parent Task Mining process includes many activities. In our Case, the parent Task Mining process has only one activity, but not all have an associated Task Mining process. If you open **GitHub Create Issue,** you can navigate to the associated Task Mining process by selecting **Launch task mining process** as shown below:

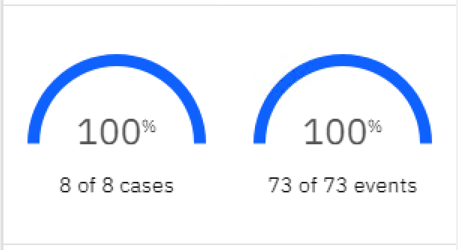


_4. Click **Task Mining: GitHub Create Issue_01** process.



Task Mining: GitHub Create ... ⋮
Task Mining
Taskminer Account

You should now see the cases count as 8. We recorded 7 cases for you, so the count was 7 before your recording.



100%
8 of 8 cases

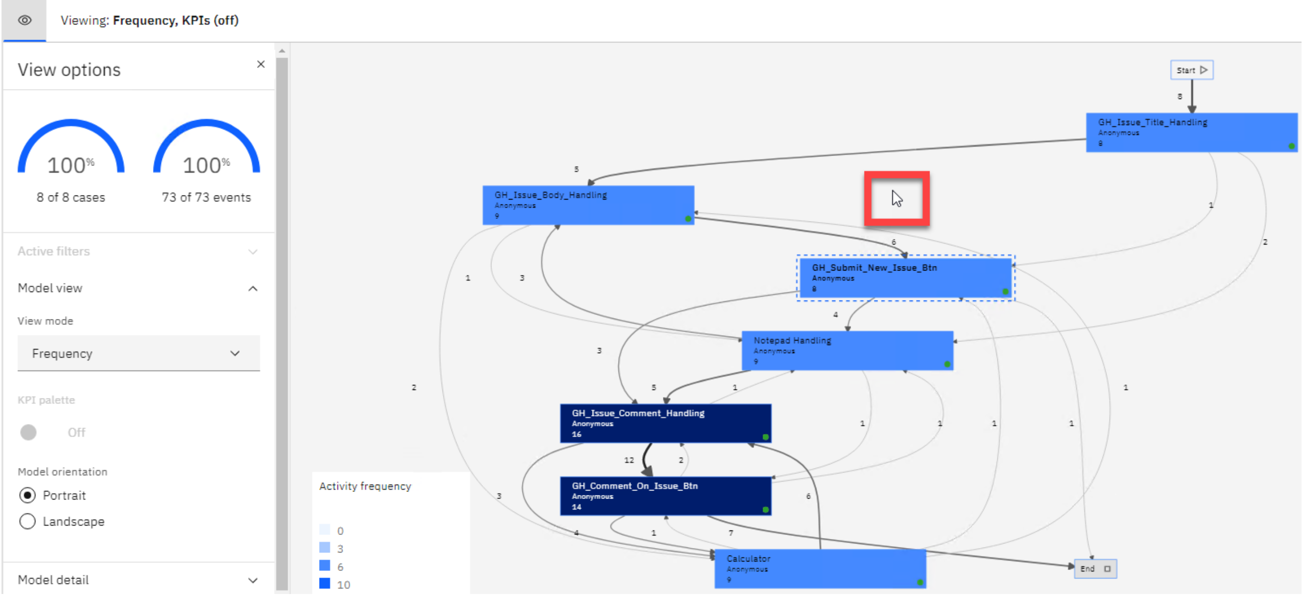
100%
73 of 73 events

2.3.2 Analyze Rework

Activities repeated more than once in the same process instance are defined as Rework. Activities with Rework typically can become candidates for automation by RPA.

In our task mining scenario, any repeated activity (for example, adding comments more than once and repeatedly using desktop apps as a source of text) when opening a GitHub Issue is considered Rework and undesirable.

_1. Click and **hold the left mouse button** and then **drag** the Model to position it in the center of the display.



Viewing: **Frequency, KPIs (off)**

View options

100% — 8 of 8 cases

100% — 73 of 73 events

Active filters

Model view

View mode: Frequency

KPI palette: Off

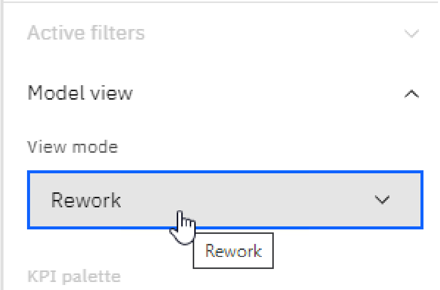
Model orientation

● Portrait

○ Landscape

Model detail

_2. For the *Model view*, select **Rework**



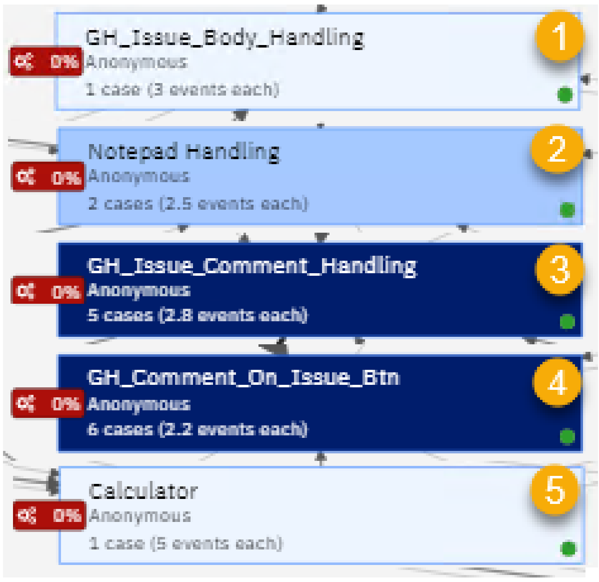
Active filters

Model view

View mode: Rework

_3. A closer look at that data indicates that there are two significant sources of rework:

- Notepad and Calculator was used more than once to complete a single Case
- Comments were added multiple times per Case.



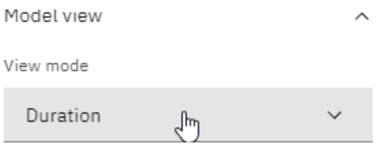
1. In 1 Case, the GitHub Issue body was updated 3 times.
2. In 2 Cases, the Comment field was updated an average 2.5 times.
3. In 5 Cases, the Add Comment button was clicked an average of 2.8 times.

   This means the comment field is updated several times when his button is clicked.
4. In 6 Cases, the Notepad was used an average 2.2 times.
5. In 1 Case, the Calculator was used 5 times.

2.3.3 Analyze KPI

Let's examine how the GitHub Create Issue process conforms to the Process KPI. Specifically, we will explore the Activities and transition with maximum durations.

_1. For the *Model view perspective,* select **Duration**

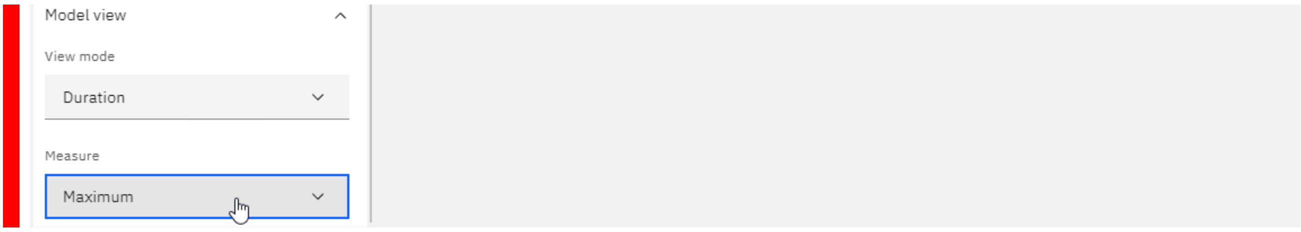


Model view

View mode

Duration

_2. For *Measure,* select **Maximum**



Model view

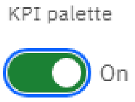
View mode

Duration

Measure

Maximum

_3. For *KPI palette,* select **On**

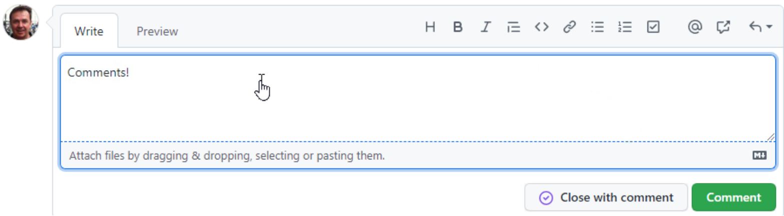


KPI palette

On

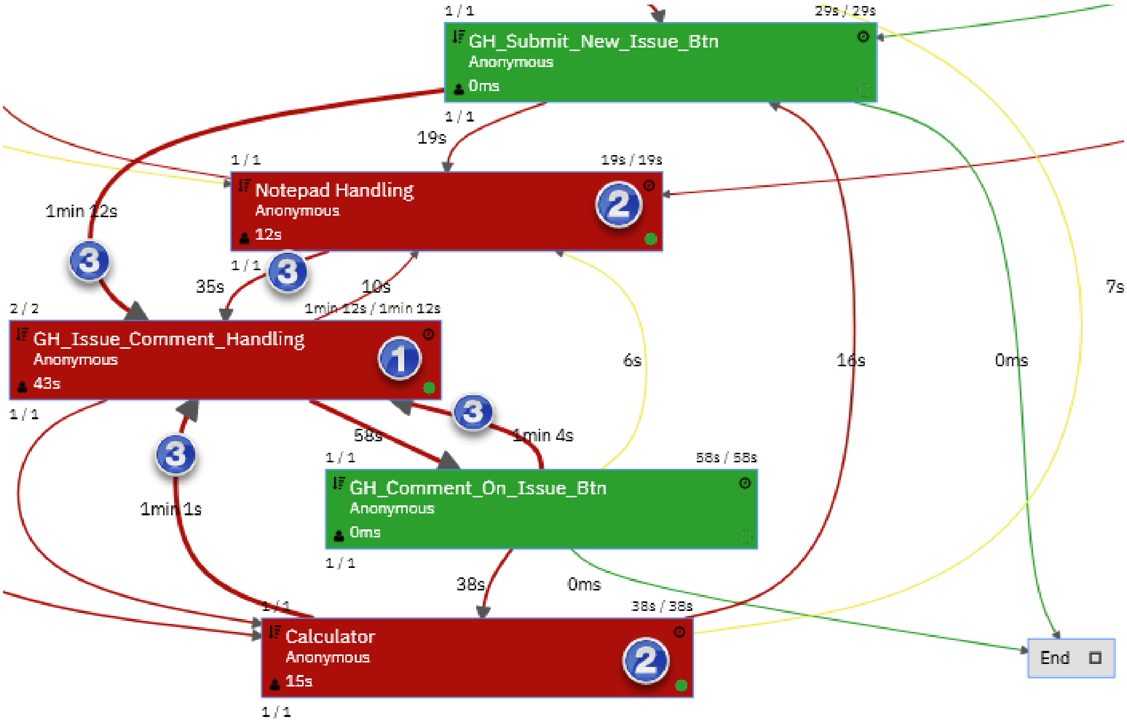
The red-colored transition and red-colored activities indicate that the duration KPI were exceeded.

It appears that much time was spent working with Comments:

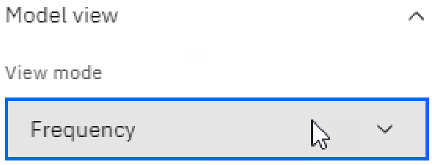

Write
Preview
Comments!
Attach files by dragging & dropping, selecting or pasting them.
Close with comment
Comment

Specifically, in the following aspect of entering Comments after an issue was submitted:

1. Editing Comments
2. Time spent using the Calculator or Notepad when working on Comments
3. Just "thinking" before actually starting to edit a Comment



_4. Set the *View mode* to **Frequency**.



Model view

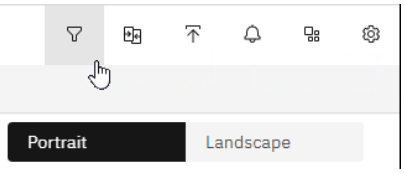
View mode

Frequency

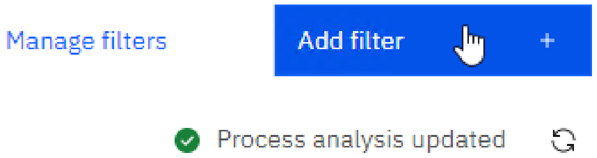
2.3.4 Find the "TM is cool" Case

We will now use a Filter to find the Case representing your recording.

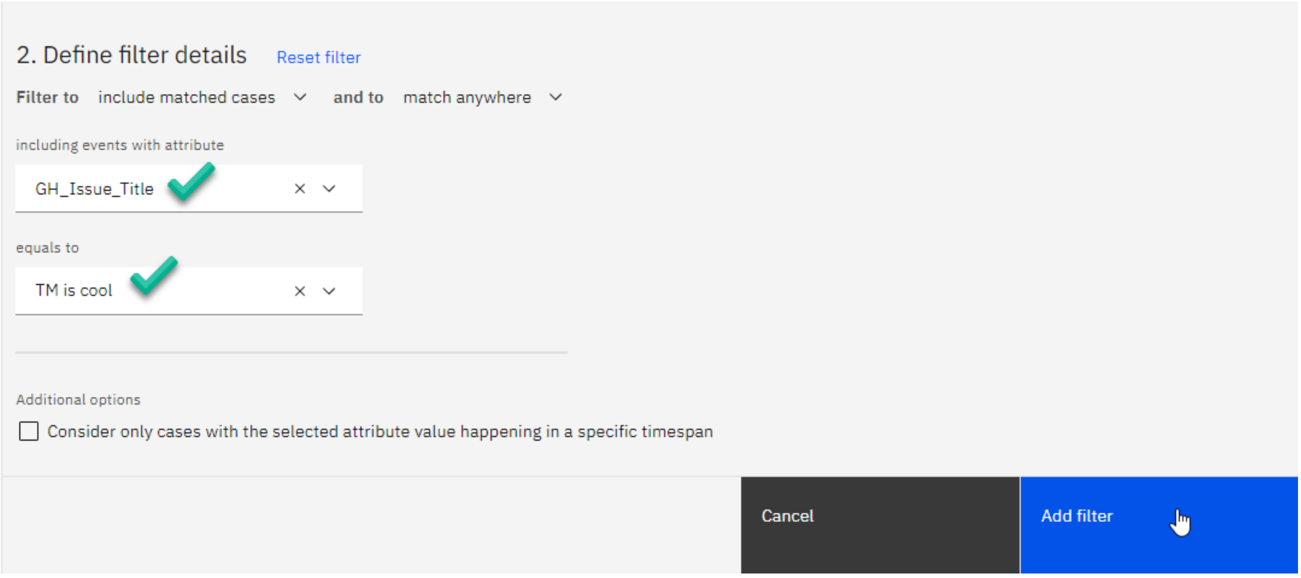
_1. Click **Filter** icon



_2. Click **Add filter +**



_3. For the *including events with attribute,* select **GH_Issue_Title;** for equal to choose **TM is cool** and then click **Add filter**



2. Define filter details

Filter to include matched cases and to match anywhere

including events with attribute: GH_Issue_Title

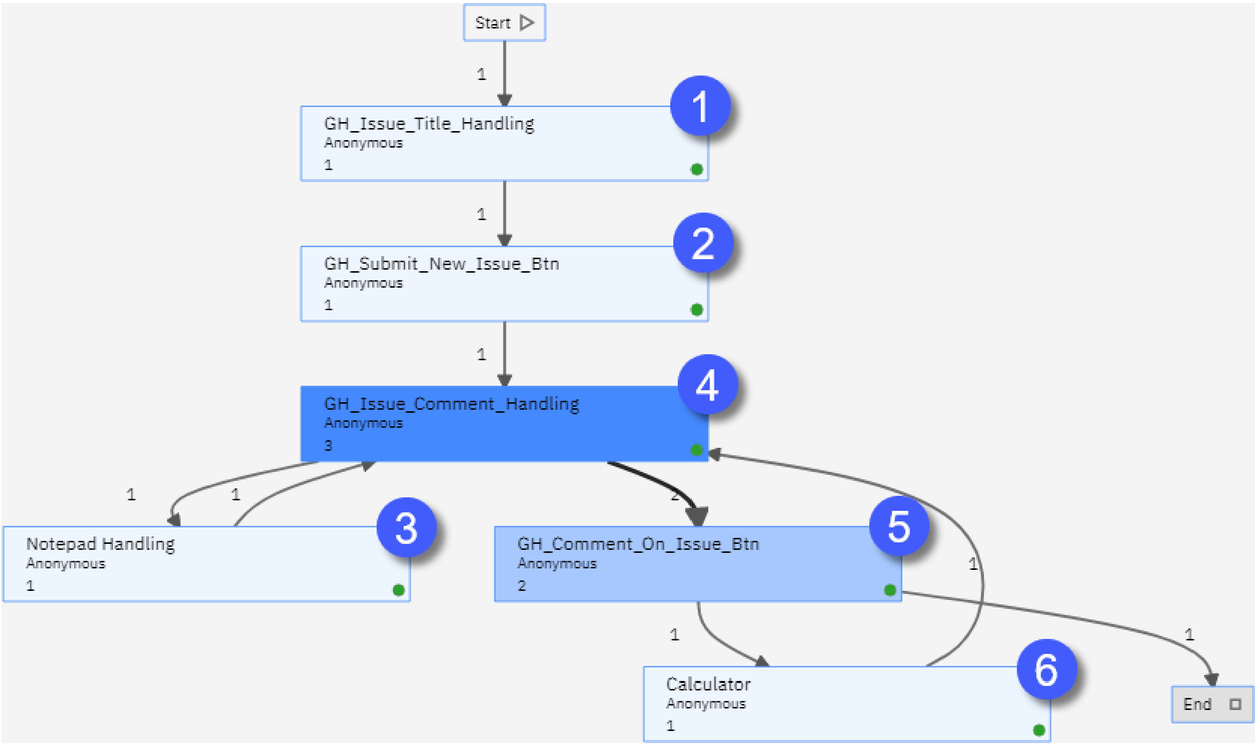
equals to: TM is cool

Additional options

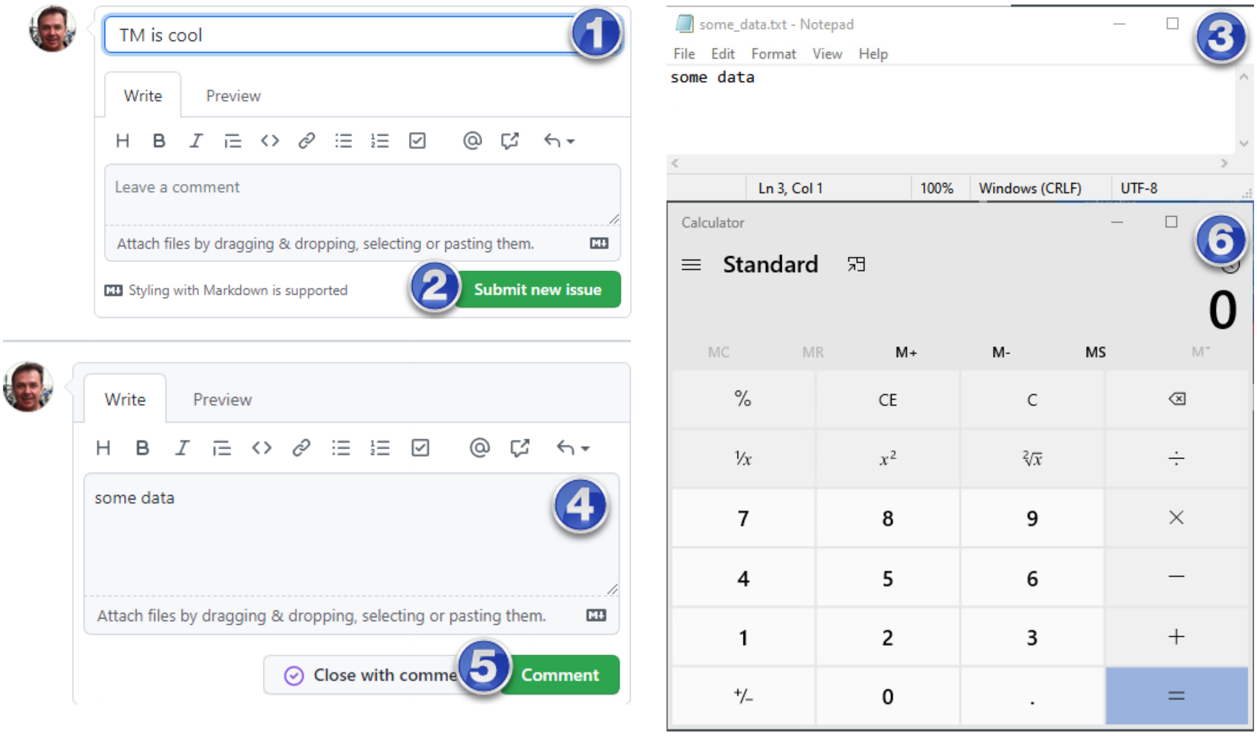
□ Consider only cases with the selected attribute value happening in a specific timespan

Let's examine what was recorded when you created the "TM is cool" issue and how the desktop actions you recorded were rendered as a Process in IBM Process Mining.

Activities Rendered in Process Mining



Desktop Actions


TM is cool

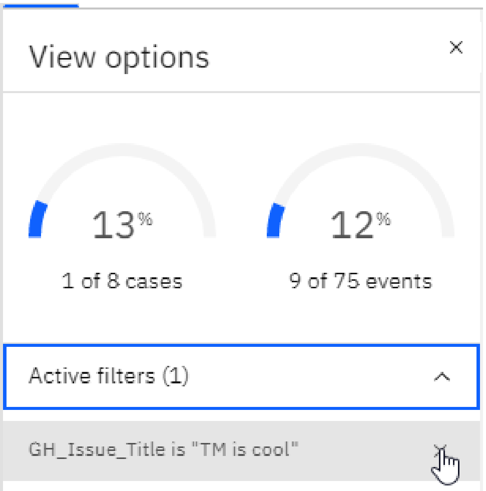
some_data.txt - Notepad

some data

Ln 3, Col 1 | 100% | Windows (CRLF) | UTF-8

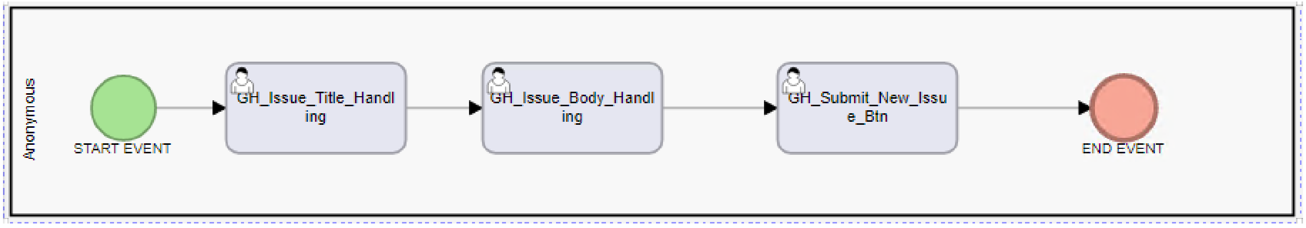
some data

_1. Click **X** on *GH_IssueTitle is "TM is cool"*



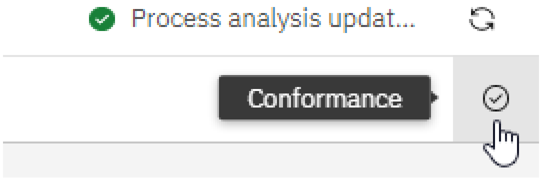
2.3.5 Conformance Analysis

Conformance Analysis is used to determine the deviations from a reference model. The reference model represents the ideal use case for opening new GitHub Issues and is defined using BPMN notation, as shown below:



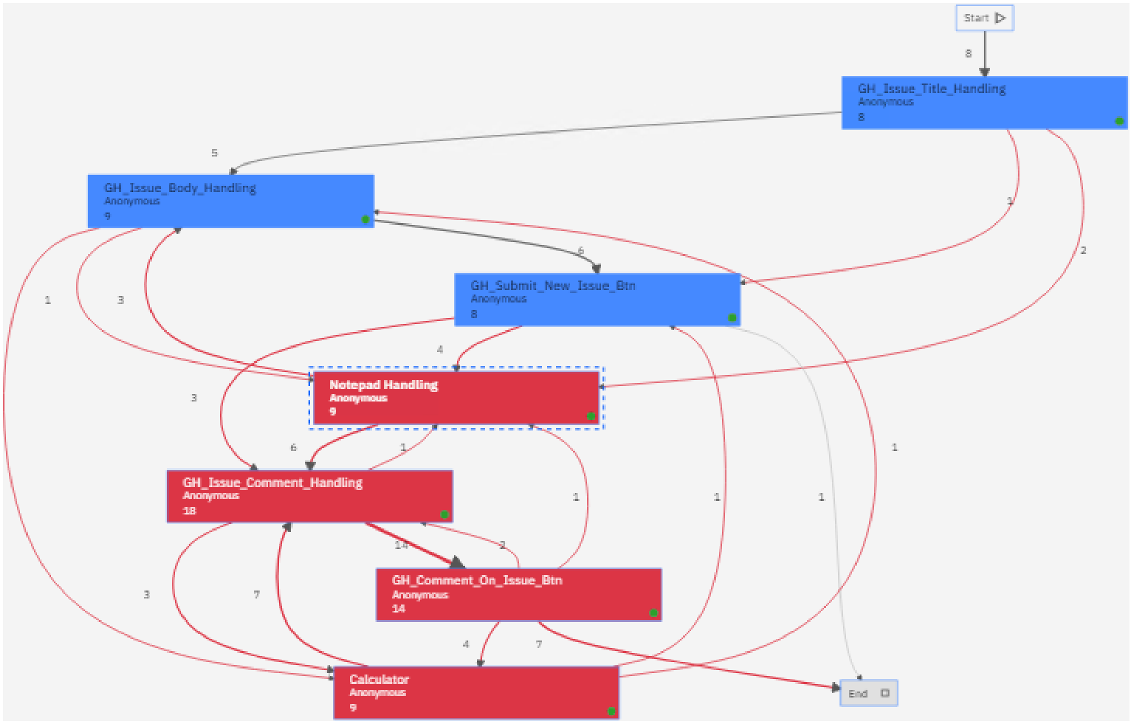
The Reference Model (ideal situation) tells us that the user opening a new GiyGun issue should define the issue title, fill in the issue detail, and click the New Issue button! That is all! Just three steps.

_1. Click the **Conformance** icon.



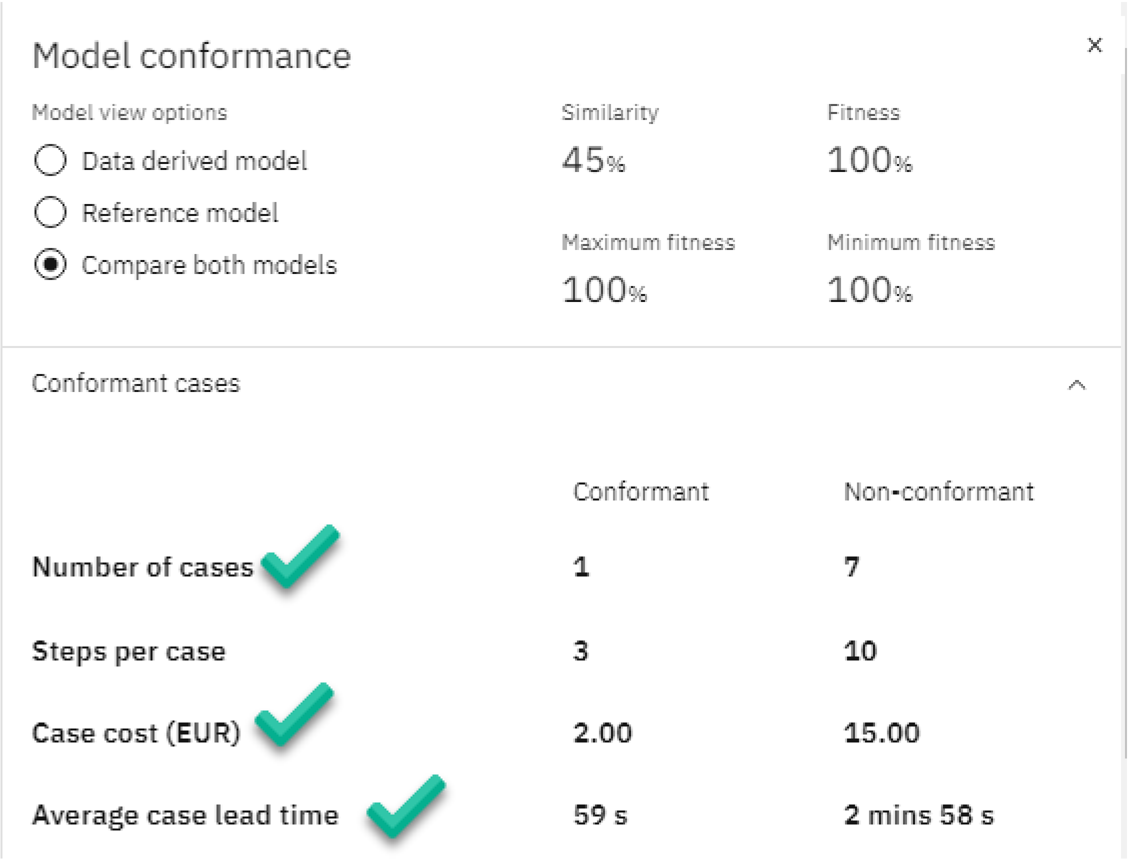
_2. Examine the Model Conformance diagram.

The View displays (in red) unexpected activities and unexpected transitions.



_3. Examine the Model Conformance summary.

Note that: (1) only 1 Case was conformant; (2) the average cost of non-conformant Case was EUR 15 versus EUR 2 for conformant; (3) the average duration of non-conformant Case was 2 mins 58 s versus 59 s for conformant.



Model conformance ×

Model view options

○ Data derived model

○ Reference model

● Compare both models

Similarity	Fitness
45%	100%
Maximum fitness	Minimum fitness
100%	100%

Conformant cases

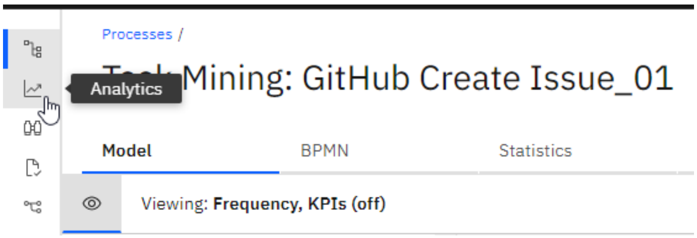
	Conformant	Non-conformant
Number of cases ✓	1	7
Steps per case	3	10
Case cost (EUR) ✓	2.00	15.00
Average case lead time ✓	59 s	2 mins 58 s

2.4 Examine Custom Process Mining Dashboard

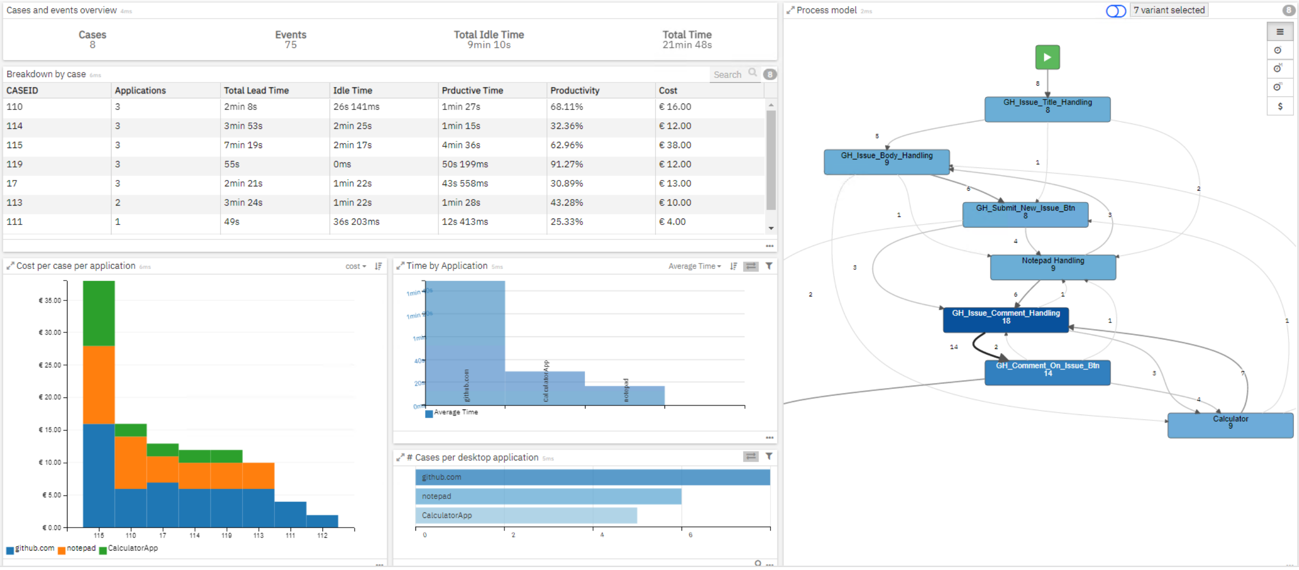
We will examine a custom Dashboard showing a summary of the desktop activities involved in creating GitHub Issues.

2.4.1 Open Dashboard

_1. Click the **Analytics**


Processes /
Analytics
Mining: GitHub Create Issue_01
Model
BPMN
Statistics
Viewing: Frequency, KPIs (off)

You should now see the *GitHub Issues – Custom Dashboard*!



CASEID	Applications	Total Lead Time	Idle Time	Prductive Time	Productivity	Cost
110	3	2min 8s	26s 141ms	1min 27s	68.11%	€ 16.00
114	3	3min 53s	2min 25s	1min 15s	32.36%	€ 12.00
115	3	7min 19s	2min 17s	4min 36s	62.96%	€ 38.00
119	3	55s	0ms	50s 199ms	91.27%	€ 12.00
17	3	2min 21s	1min 22s	43s 558ms	30.89%	€ 13.00
113	2	3min 24s	1min 22s	1min 28s	43.28%	€ 10.00
111	1	49s	36s 203ms	12s 413ms	25.33%	€ 4.00

2.4.2 Use "GitHub Issues – Custom Dashboard" to Gain Insights

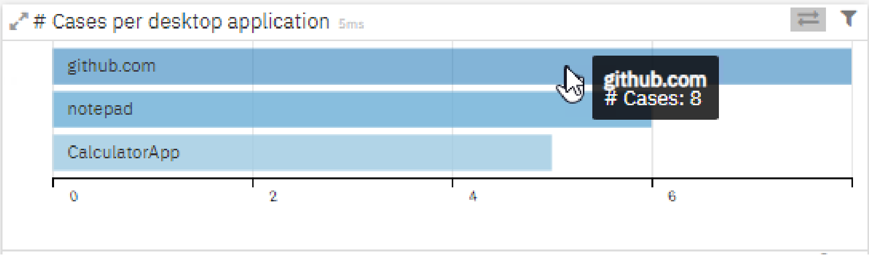
This custom dashboard includes several charts which can be used to analyze the desktop activities involved in opening new GitHub Issues.

The key metrics we want to:

- Explore the Use of the Calculator App
- Explore the Cost of Issue Creation
- Time Study of Applications Involved in Issue Creation

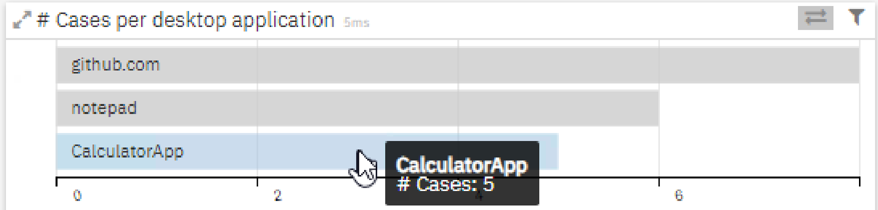
2.4.2.1 Explore the Use of the Calculator App

_1. Take a look at the Cases per desktop application Chart



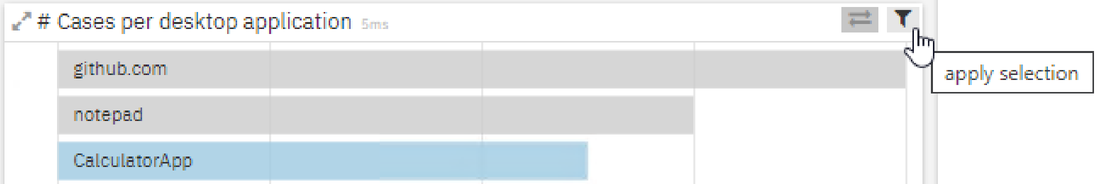
We can see that all 8 Cases used gitgub.com (which is obvious), but not all used Calculator or Notepad.

_2. Click the **CalculatorApp ba**



Cases per desktop application 5ms

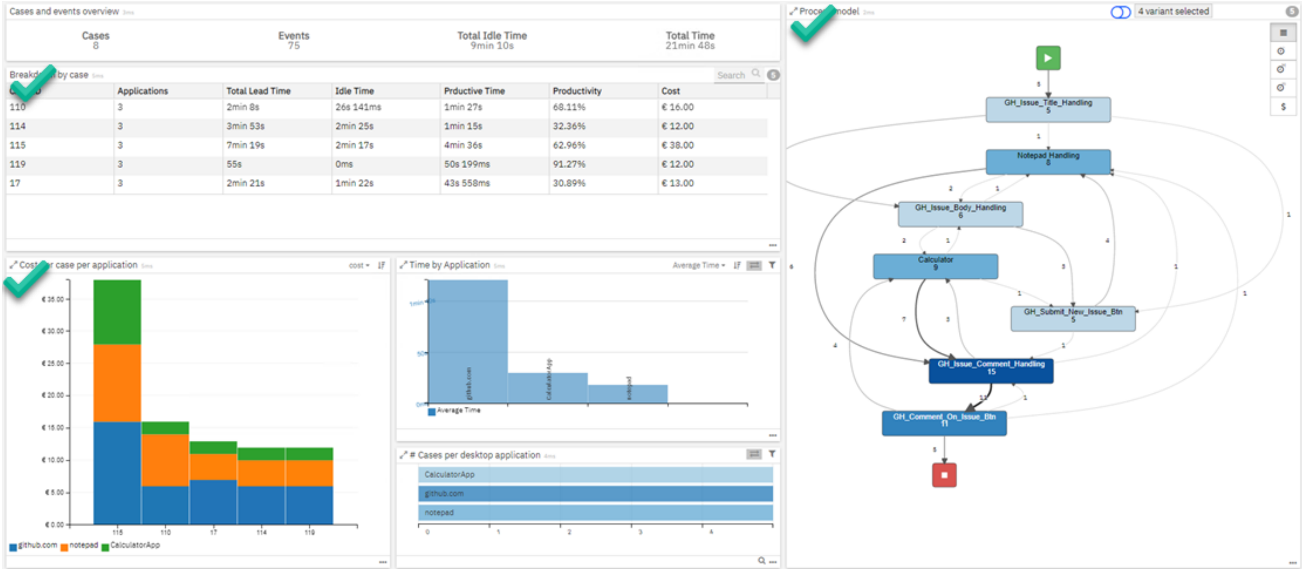
_3. Click *apply selection* **Filter icon**



Cases per desktop application 5ms

Note that now you see only Cases in which the Calculator App was used!

_4. Examine **Breakdown by Case** and **Cost per Case per application Charts** (they now inlude only 5 cases) and **Process model** Chart (it now displays only 4 variants)



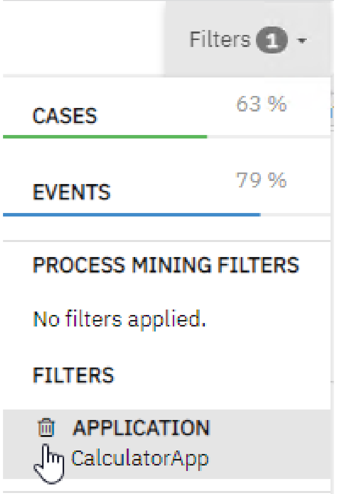
Cases and events overview

Cases	Events	Total Idle Time	Total Time
8	75	9min 10s	21min 48s

Breakdown by case

Case ID	Applications	Total Lead Time	Idle Time	Prductive Time	Productivity	Cost
110	3	2min 8s	26s 141ms	1min 27s	68.11%	€ 16.00
114	3	3min 53s	2min 25s	1min 15s	32.36%	€ 12.00
115	3	7min 19s	2min 17s	4min 36s	62.96%	€ 38.00
119	3	55s	0ms	50s 199ms	91.27%	€ 12.00
17	3	2min 21s	1min 22s	43s 558ms	30.89%	€ 13.00

_5. In the *Filters*, and then click the **Garbage Can** to remove the **CalculatorApp** filter.



CASES 63 %

EVENTS 79 %

PROCESS MINING FILTERS

No filters applied.

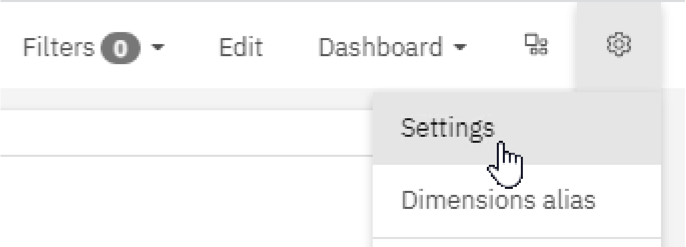
FILTERS

APPLICATION
CalculatorApp

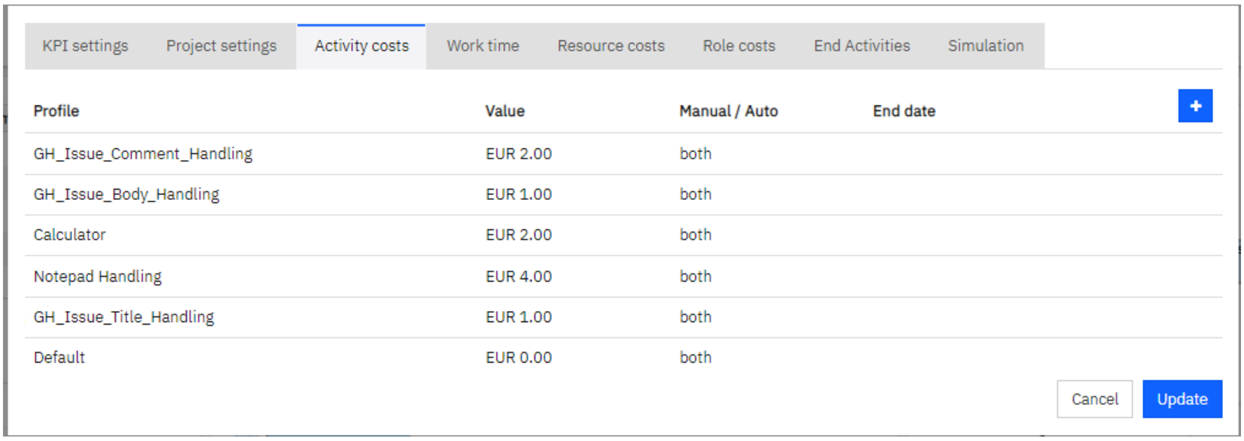
2.4.2.2 Explore the Cost of Issue Creation

The first step in performing the cost analysis is to configure *Activity costs*.

_1. Select **Settings**



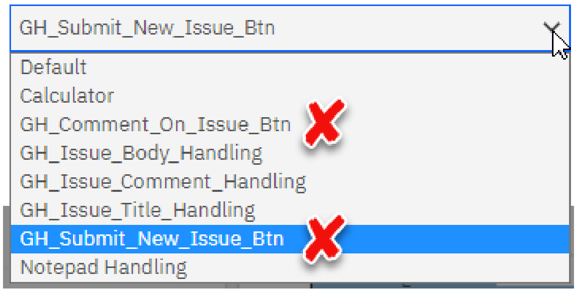
_2. Click Activity costs



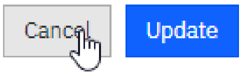
Profile	Value	Manual / Auto	End date
GH_Issue_Comment_Handling	EUR 2.00	both	
GH_Issue_Body_Handling	EUR 1.00	both	
Calculator	EUR 2.00	both	
Notepad Handling	EUR 4.00	both	
GH_Issue_Title_Handling	EUR 1.00	both	
Default	EUR 0.00	both	

Note that we have already configured various activity costs for you.

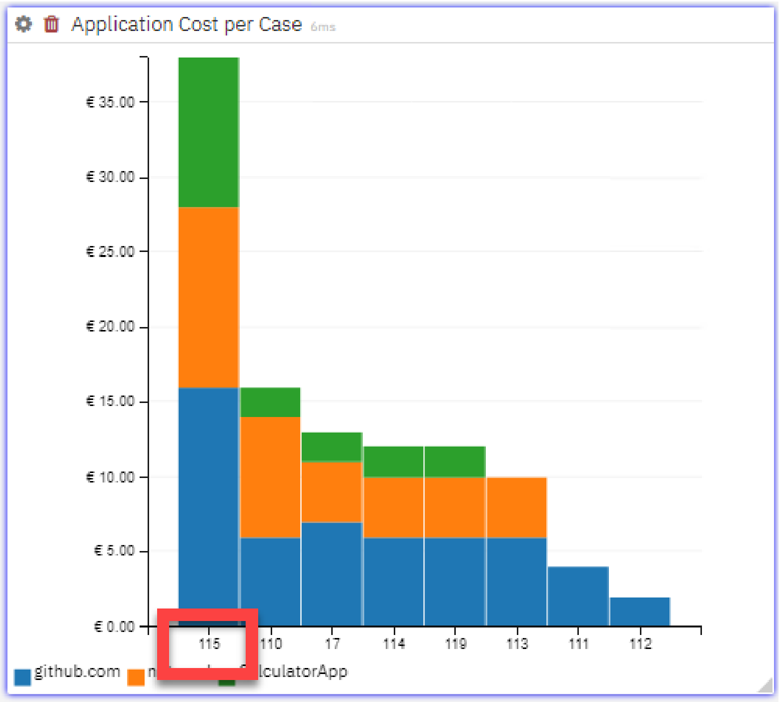
Also, note that we did not include activities representing button clicks. They do not have a real business meaning and were used only to capture desktop activities' start and stop events.



_3. Click the **Cancel** to close the *Settings Window*.



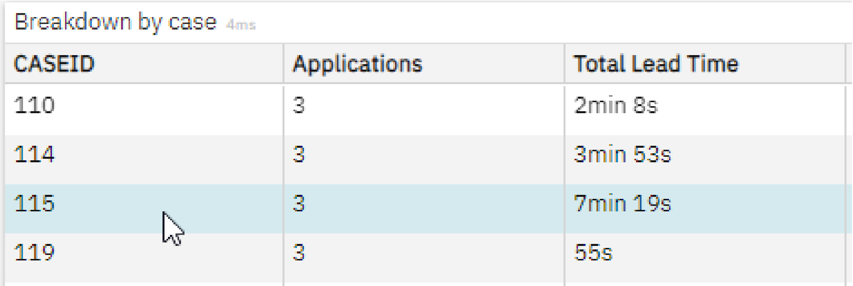
_4. Examine the *Application Cost per Case* Chart.



This Chart can be used to determine the costliest Case and most costly application. The Y-axis represents the costs, and the X-axis represents a Case. You can see how a Case cost is split between the three applications we used to open a new GitHub Issue.

Note that the Case with CASEID 115 is the costliest!

_5. In the *Breakdown by case* chart, click the **row with CASEID 115**

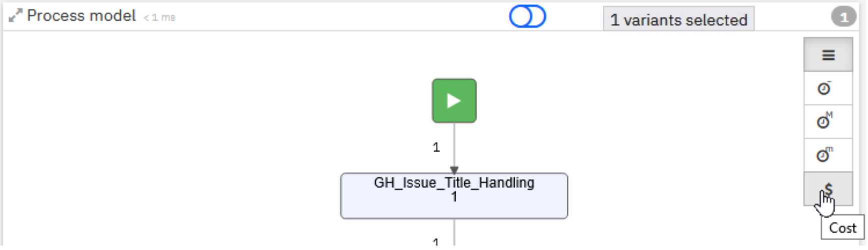


Breakdown by case 4ms

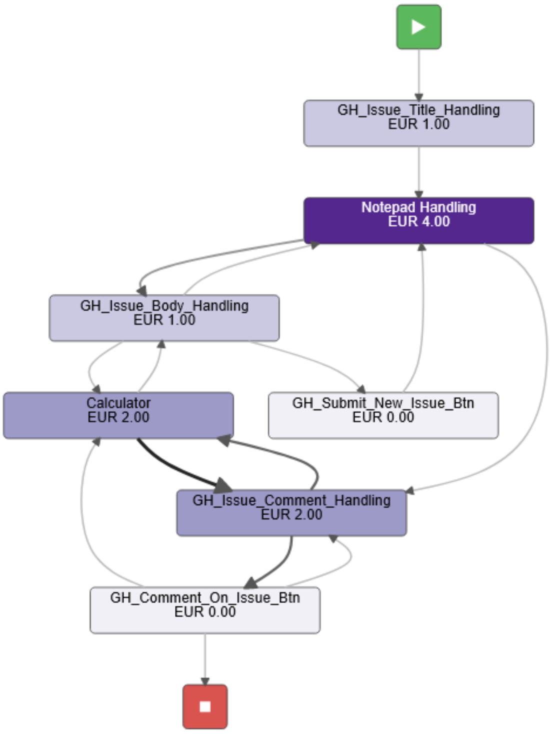
CASEID	Applications	Total Lead Time
110	3	2min 8s
114	3	3min 53s
115	3	7min 19s
119	3	55s

Note, this action automatically applies a filter that can change most of the charts in the dashboard to include only the Case you selected.

_6. Switch to the **Process model** Chart and click the **$ (Cost)** icon



You should now see the Cost View



Notice :

- The Noted Handing, Calculator, and GH_Isse_Commnet_Handing are the costliest activities (indicated by dark activity color)
- The activities are repeated multiple times
- The GitHub Issue Creation included 4 steps: Providing the GitHub Issue title, Typing the content of the GitHub Issue Body, and Addition of three comments

If you scrutinize the Process diagram, you can see what the user did to create this GitHub Issue:

Provided GitHub Issue Title

- GH_Issue_Title_Handling > Notepad Handling

Typed the content of the GitHub Issue Body

- GH_Issue_Body_Handling > Notepad Handling > GH_Issue_Body_Handling > Calculator
- GH_Issue_Body_Handling
- GH_Submit_New_Issue_Btn

Added Comment 1

- Notepad Handling > GH_Issue_Comment_Handling

- GH_Comment_On_Issue_Btn

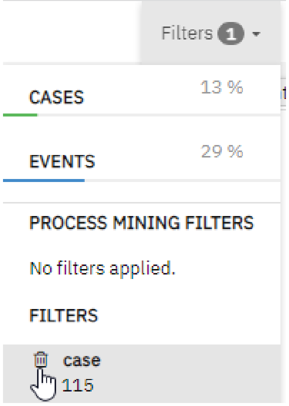
Added Comment 2

- GH_Issue_Comment_Handling > Calculator > GH_Issue_Comment_Handling > Calculator > GH_Issue_Comment_Handling
- GH_Comment_On_Issue_Btn

Added Comment 3

- Calculator > GH_Issue_Comment_Handling > Calculator > GH_Issue_Comment_Handling
- GH_Comment_On_Issue_Btn

_7. In the *Filters,* and then click the **Garbage Can** to remove the **115** filter.



Filters 1

CASES 13 %

EVENTS 29 %

PROCESS MINING FILTERS

No filters applied.

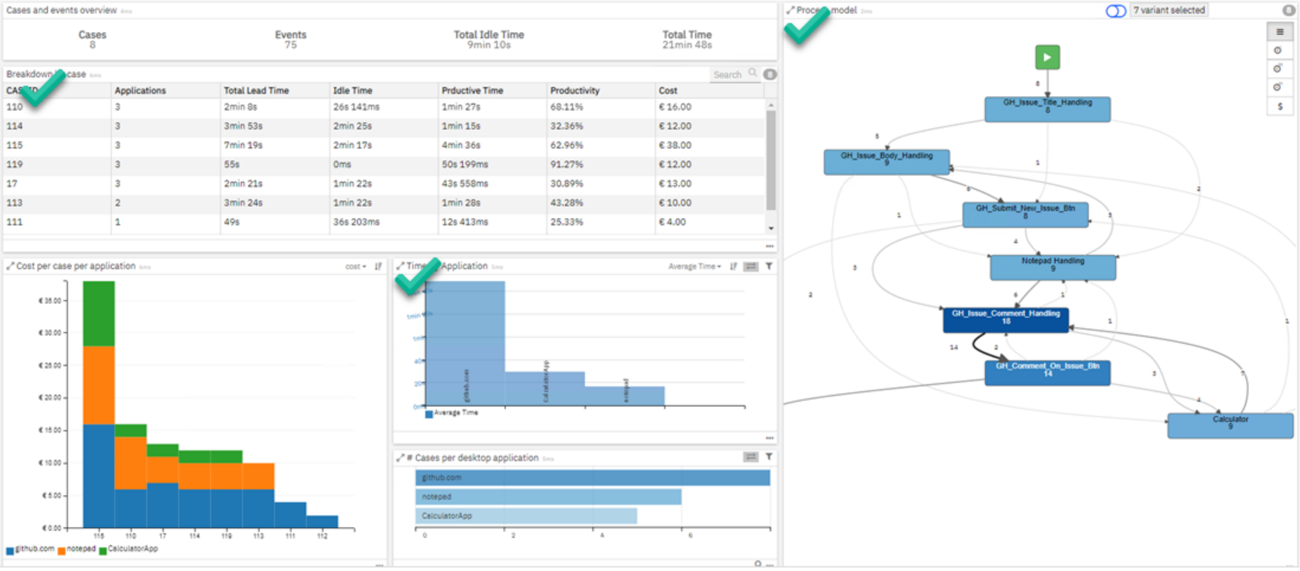
FILTERS

case
115

_8. Click the **Garbage Can** to delete the **APPLICATION** filter in the *Filters Chart*.

2.4.2.3 Time Study of Applications Involved in GitHub Issue Creation

We will use the following three charts: *Breakdown by case*, *Time by application*, and *Process model* to conduct a time study to identify bottlenecks and applications contributing to high Lead Time and Idle Time.



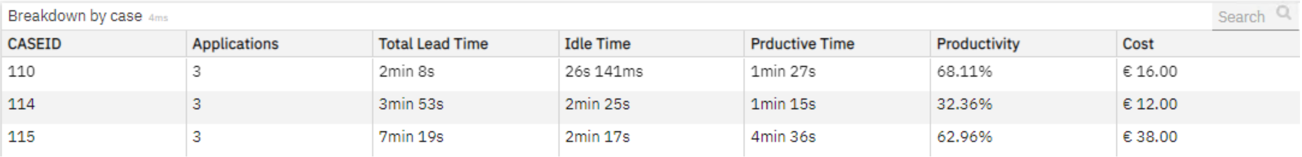
Cases and events overview

Cases	Events	Total Idle Time	Total Time
8	75	9min 10s	21min 48s

Breakdown by case

CASEID	Applications	Total Lead Time	Idle Time	Prductive Time	Productivity	Cost
110	3	2min 8s	26s 141ms	1min 27s	68.11%	€ 16.00
114	3	3min 53s	2min 25s	1min 15s	32.36%	€ 12.00
115	3	7min 19s	2min 17s	4min 36s	62.96%	€ 38.00
119	3	55s	0ms	50s 199ms	91.27%	€ 12.00
17	3	2min 21s	1min 22s	43s 558ms	30.89%	€ 13.00
113	2	3min 24s	1min 22s	1min 28s	43.28%	€ 10.00
111	1	49s	36s 203ms	12s 413ms	25.33%	€ 4.00

_1. Let's examine the contents of the **Breakdown by case** Chart that can be helpful in a time study.



Breakdown by case

CASEID	Applications	Total Lead Time	Idle Time	Prductive Time	Productivity	Cost
110	3	2min 8s	26s 141ms	1min 27s	68.11%	€ 16.00
114	3	3min 53s	2min 25s	1min 15s	32.36%	€ 12.00
115	3	7min 19s	2min 17s	4min 36s	62.96%	€ 38.00

CASEID – identifies the Case. Your recording created CASEID 8.

Applications – show how many applications were in a Case (Calculator, Notepad, github.com).

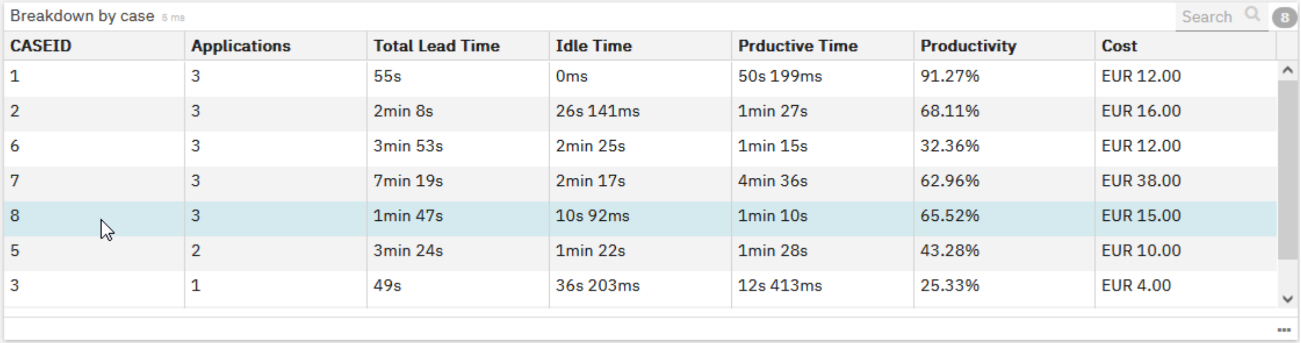
Total Lead Time – shows the time from the start of a Case until its conclusion. It tells us how long it took to open a GitHub Issue.

Idle Time – is when the user has not been active on the pages/windows associated with the Case during the execution of the Case itself. This metric tells us if a user was distracted or needed a lot of "think time" when opening a GitHub Issue.

Productive Time – is the opposite of Idle Time. This is when the user is typing or using the mouse.

Productivity – Productive Time / Total Lead Time. A high percentage indicates a high idle or think time.

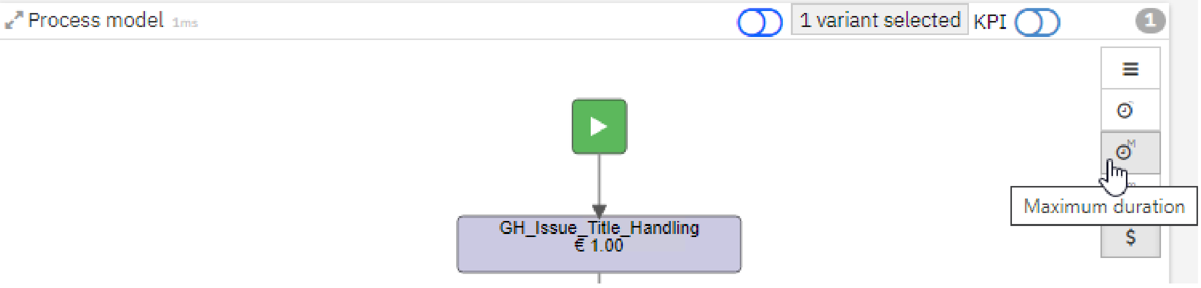
_2. Click on the row with *CASEID* **17** (the Case you just recorded) to create a filter showing only the data for this Case.



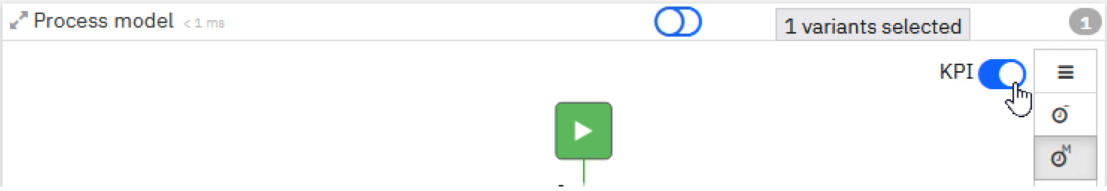
CASEID	Applications	Total Lead Time	Idle Time	Prductive Time	Productivity	Cost
1	3	55s	0ms	50s 199ms	91.27%	EUR 12.00
2	3	2min 8s	26s 141ms	1min 27s	68.11%	EUR 16.00
6	3	3min 53s	2min 25s	1min 15s	32.36%	EUR 12.00
7	3	7min 19s	2min 17s	4min 36s	62.96%	EUR 38.00
8	3	1min 47s	10s 92ms	1min 10s	65.52%	EUR 15.00
5	2	3min 24s	1min 22s	1min 28s	43.28%	EUR 10.00
3	1	49s	36s 203ms	12s 413ms	25.33%	EUR 4.00

We will now examine the **Process model** Chart to understand the root cause of the Lead Time and Idle Time we observed in Case 17.

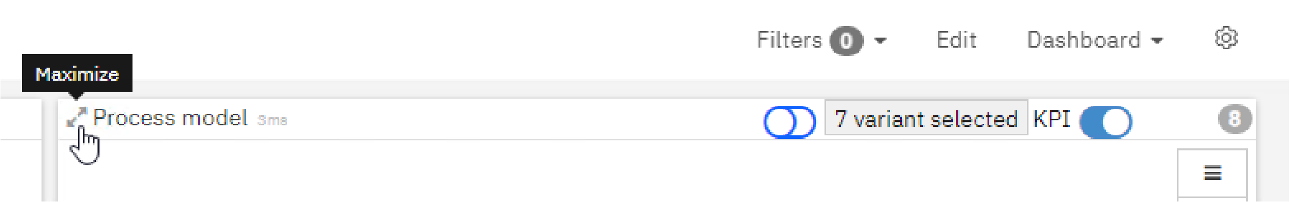
_3. In the *Process model* chart, click the **Maximum duration** icon.



_4. Switch on the **KPI** toggle

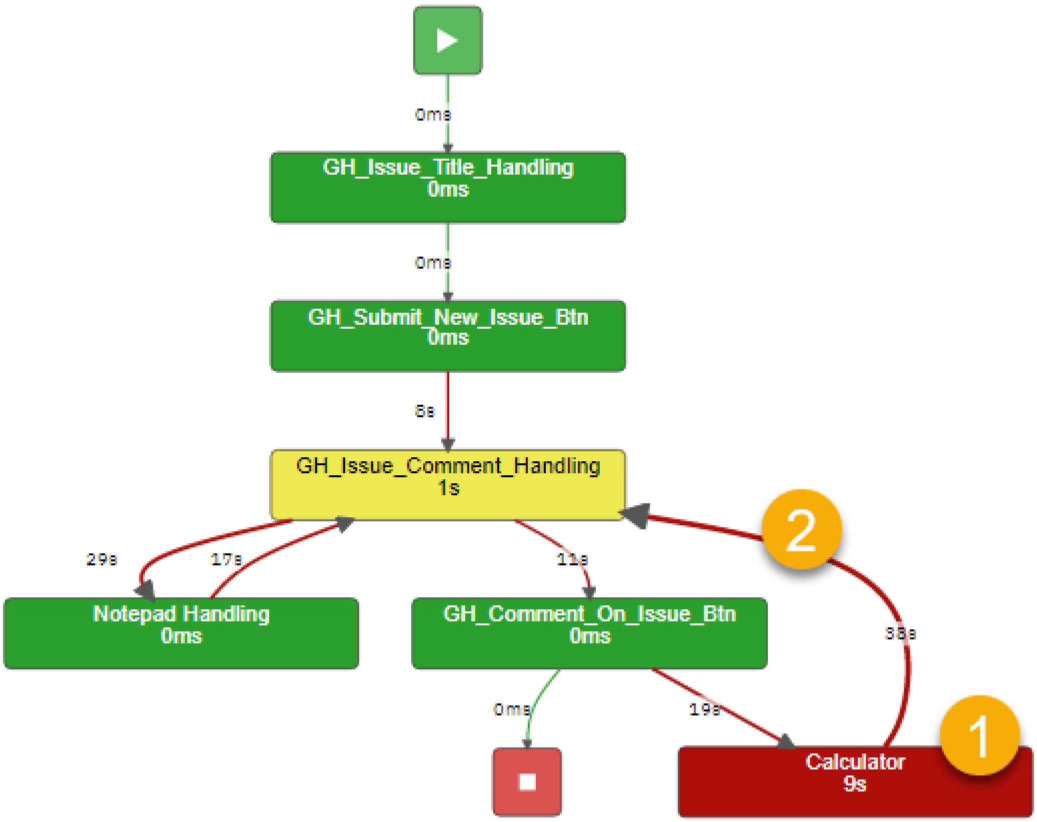


_5. Click **Mximixe** and move the flow dugram to the center of the Process model widget.



_6. Let's examine the **Process model** chart showing the **Maximum Duration** values.

> Note that your time values or the sequence of Activities may differ depending on how well you followed the lab instructions. 😊



1. The duration values inside activities tell us how much time was spent performing an activity. For example, the user spent 9 seconds working with the Calculator App.
2. The transition duration values tell us how much time it took between activities. For example, it took the user 38 seconds to switch from Calculator App back to the Comment area in the GitHub Issues web page.

3 Lab Summary

In this lab, you explored the newly re-designed in v1.14 Task Mining feature of IBM Process Mining by performing the following:

1. First, you recorded opening a GitHub issue in the Windows desktop and examined the recorded desktop events in the Process Mining model.
2. Next, you examined the Process Model that includes the Cases recorded previously and the new Case you recorded in the first part of the lab.
3. Next, you examined a custom Dashboard showing a summary of the desktop activities involved in creating GitHub Issues.

NOTICES

This information was developed for products and services offered in the USA.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering the subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
United States of America

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow the disclaimers of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements, or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

TRADEMARKS

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

IT Infrastructure Library is a Registered Trade Mark of AXELOS Limited.

ITIL is a Registered Trade Mark of AXELOS Limited.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Linear Tape-Open, LTO, the LTO Logo, Ultrium, and the Ultrium logo are trademarks of HP, IBM Corp., and Quantum in the U.S. and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.