­­­­­

Solution Design

Document

MTM

|  |  |
| --- | --- |
|  |  |



Indholdfortegnelse

[1. Procesdetaljer 3](#_Toc145505368)

[1.1 Udfyldelse af SDD 3](#_Toc145505369)

[2. Kørsel 4](#_Toc145505370)

[2.1 Driftsdetaljer for processen 4](#_Toc145505374)

[2.2 Packages and Namespaces 5](#_Toc145505375)

[3. Procesbeskrivelse 6](#_Toc145505376)

[3.1 Driftsdetaljer for processen 6](#_Toc145505379)

[3.2 Workflows 6](#_Toc145505380)

[3.3 Flowdiagram 6](#_Toc145505381)

[3.4 In/out argumenter 7](#_Toc145505382)

[4. Fejlhåndtering 8](#_Toc145505383)

[4.1 Potentielle fejl 8](#_Toc145505384)

[4.2 Undtagelser 8](#_Toc145505385)

[5. Andre bemærkninger 8](#_Toc145505386)

[5.1 Nice to know 8](#_Toc145505387)

[5.2 Fremtidige forbedringer 9](#_Toc145505388)

# Procesdetaljer

|  |  |
| --- | --- |
| Item | Beskrivelse |
| Procesnavn | NovaToFilarkiv |
| Robot type | Unattended |
| Orchestrator Queuename | NovaToFilarkiv |
| Skalerbar | Ja |
| UiPath version | 2023.4.2 |
| Dokument version | 1.0 |
| Git Link | [mtm-aarhus/NovaToFilarki (github.com)](https://github.com/mtm-aarhus/NovaToFilarki) |

|  |  |
| --- | --- |
| Rolle | Navn |
| Procesekspert | Kristian Andersen |
| Procesejer | Hanne Eline Povlsen |
| Udvikler | Gustav Chatterton |

*Procesekspert = Vedkomne som laver processen manuelt - Procesejer = Proceseksperts overordnet.*

## Udfyldelse af SDD

Solution Design Document skal udfyldes for hver automatiseringsproces. Dokumentationen placeres i procesmappen under ”Documentation”. Det er den enkelte udviklers ansvar at commit and push til [mtm-aarhus (github.com)](https://github.com/mtm-aarhus)

# Kørsel



## Driftsdetaljer for processen

Beskriv detaljerne for den automatiserede proces ved at udfylde tabellen nedenfor.

|  |  |  |
| --- | --- | --- |
| Navn | Beskrivelse  Udfyld felterne - tomme felter er ikke tilladt. Hvis sektionen ikke er relevant for din automatisering, så marker den som n/a. | |
| Environment detaljer | ***Processen anvender Nova API, Filarkiv API og netværkskald til at overfører dokumenter fra Nova til Filarkiv*** | |
| Forudsætninger for kørsel | *Liste med sagsnumre skal være oploaded til orchestrator som queueItems. Blacklistede ord, som titlen og beskrivelsen ikke må indeholde skal være opdateret.* | |
| Adgangskrav | *KMDNova bearer token, Filarkiv bearer token, KMDNova Login credentials, Database login* | |
| Input Data | *Liste med sagsnumre* | |
| Angiv SQL Server | "Data Source=nova2filarkiv.database.windows.net;Initial Catalog=nova2filarkiv\_test;Persist Security Info=True;User ID="+Login+";Password="+PasswordStr+";Encrypt=False" | |
| Forventet output | Dokumentet overført fra Nova til Filarkiv + registrering af hvilke dokumenter samt ekstra info i database | |
| Hvordan starter processen | *Processen starter via Orchestrator queueItems og triggers* | |
| Orchestrator queuename | ***NovaToFilarkiv*** | |
| Assets navne og beskrivelse | FilarkivAPIToken | *Token til filarkiv* |
| FilarkivTokenTimestamp | *Token timestamp* |
| KMDAPIToken | *KMDAPIToken* |
| TokenTimestamp | *Token timestamp* |
| KMDNovaLogin | *Login til KMD NOVA* |
| Nova2Filarkiv\_Credentials | *Credentials til Nova2FilArkiv databasen* |
| KMDRequestVerification | *Fra netværkskald* |
| KMDVerificationToken | *Fra netværkskald* |
|  | KMDLogonWebSession | *Fra netværkskald* |
| Trigger navn |  | |
| Trigger tidspunkt | Engangsrobot som skal kører i weekenderne | |

## Packages and Namespaces

Inkluder listen over tilføjet dependencies og namespaces. Marker hvis pakken ikke er udgivet af UiPath og derfor på sigt kan blive udateret.

**Packages:**

|  |  |  |
| --- | --- | --- |
| Navn og version | Beskrivelse | Uipath |
| *RestSharp 106.15.0* |  | ***Nej*** |
| UiPath.Database.Activities 1.7.1 |  | ***Ja*** |
| UiPath.Excel.Activities 2.20.2 |  | ***Ja*** |
| UiPath.System.Activities 23.4.3 |  | ***Ja*** |
| UiPath.Testing.Activities 23.4.1 |  | ***Ja*** |
| UiPath.UIAutomation.Activities 23.4.6 |  | ***Ja*** |
| UiPath.WebAPI.Activities 1.18.0 |  | ***Ja*** |

**Namespaces:**

Udfyld kun, hvis Namespace ikke er inkluderet og derfor manuelt importeret

|  |  |
| --- | --- |
| Navn | Beskrivelse |
| *Restshap* |  |
| *RestSharp.Authenticators* |  |
| *RestSharp.Authenticators.OAuth* |  |
| *RestSharp.Extensions* |  |
| *RestSharp.Serialization* |  |

# Procesbeskrivelse



## 3.1 Driftsdetaljer for processen

Formålet med processen er at overfører dokumenter fra KMD Nova til Filarkiv for bestemte sager. Grundet et overlap mellem af nyt arkiveringssystem, skal alle sager fra omkring november 2021 til d.d. Enkelte specifikke dokumenter skal ikke overføres, hvilket processen tager højde for. Processen starter ved at indhente QueueItems fra Orchestrator, som er de sagsnumre der skal overføres. Herefter processeres et sagsnummer ad gangen, hvor dets tilhørende dokumenter bliver overført til Filarkiv. Undervejs tjekkes der for om sagen i forvejen findes i Filarkiv vha. API, hvis den gør dette, overføres de dokumenter, som findes ikke igen. Herefter hentes informationen fra KMDNova via API, som skal sendes til Filarkiv. Ved hjælp af netværkskald, batch sendes dokumenterne til fra KMDNova til Filarkiv. Efterfølgende registreres i KMDNova at det enkelte dokument er overført. Til sidst logges det i en database, hvilke dokumenter, som er overført og tidspunktet for overførslen.

En ”opfølger” robot, tjekker efterfølgende om dokumentet kan findes i Filarkiv, og markeres true/false i databasen.

## 3.2 Workflows

Udfyld en kort beskrivelse af formålet med hvert framework workflow

|  |  |
| --- | --- |
| Workflow navn | Beskrivelse |
| InitAllApplications - Novalogin | *Henter requestinformation, verificationToken og LogonWebSession.* |
| GetTransactionData | *Henter sagsnumre fra orchestrator* |
| Process | *Finder de specifikke dokumenter og deres information, som skal overføres til fra KMDNova til Filarkiv* |
| CheckIfCaseExistsInFilArkiv | *Via API-kald tjekker om sagen findes i Filarkiv. Hvis sagen findes er outputtet en datatabel med dokumentnumre fra de dokumenter som er overført.* |
| GetFilArkivAccessToken | *Henter bearer Token til API. Hvis Token er hentet for under 1 time siden, anvendes det gamle, som ligger i Orchestrator som Token* |
| GetCaseAndDocumentInfo | *Henter sagsinformation herunder: CaseId, DocumentID, CaseNumber, DocumentNumber og title.* |
| GetAcessToken | *Henter bearer token til KMDNova* |
| GetCaseUuid | *Henter sags uuid, som skal anvendes i næste workflow* |
| GetDocumentInfo\_New | *Henter sagsinformation, som bliver outputtet i datatabel. Undervejs tjekkes der for om dokumenterne indeholder en række ”blacklistede” ord, som dokumentet ikke må indeholde. Derudover tjekkes der for om fil-typen er excel.* |
| RemoveDocumentDuplicates | *Tjekker om nogen dokumenter i forvejen findes i Filarkiv. Hvis de gør, så fjernes de fra listen med dokumenter, som skal overføres.* |
| SendDocumentToFilarkiv | *Overfører dokumenterne via netværkskald invoket i C# kode. Dokumenterne sendes som et batch.* |
| RegisterTransferInNova | *Registrerer via API-kald til KMDNova at det enkelte dokument er overført* |
| LoginforToDataBase | *Logger den overførte information samt tidspunkt for overførsel til database. Oprettet, hvis der skulle forekomme nogen fejl eller datalæk, så er der styr på hvilken information der er overført.* |
| CheckIfDataInStorageBucket | *Workflow som vil blive relevant, hvis der ikke er nogen rækker at bliver overført. Relevant, hvis robotten tidligere har fejlet i at logge informationen til databasen. Hvis robotten fejler, bliver det information, som ikke er overført, tilføjet til orchestrator Storagebucket, som I dette workflow hentes.* |
| End process | *N/A* |

## 3.3 Flowdiagram

Udfyld et flowdiagram over processen med følgende tegneværktøj: [**https://draw.io/**](https://draw.io/)

**Et billede, der indeholder tekst, skærmbillede, diagram, design

Automatisk genereret beskrivelse**

## 3.4 In/out argumenter

|  |  |  |
| --- | --- | --- |
| Workflow | In-argument | Out-argument |
| InitAllApplications - Novalogin |  | Assets:  KMDRequestVerification  KMDVerificationToken  LogonWebSession |
| GetTransactionData | In\_TransactionItem |  |
| Process |  |  |
| CheckIfCaseExistsInFilArkiv | In\_CaseNumber | Out\_dt\_DocumentNumber  Out\_DoesCaseExist |
| GetFilArkivAccessToken |  | Out\_Token |
| GetCaseAndDocumentInfo | in\_CaseNumber | out\_dt\_CaseAndDocumentInfo  out\_CaseID  out\_KMDAPIToken |
| GetAcessToken |  | out\_Token |
| GetCaseUuid | *in\_CaseNumber*  *in\_Token* | out\_CaseUuid |
| GetDocumentInfo\_New | *in\_CaseNumber*  *in\_Token*  *in\_CaseUuid* | out\_dt\_CaseAndDocumentInfo |
| RemoveDocumentDuplicates | *in\_dt\_DocumentNumber*  *in\_dt\_CaseAndDocumentInfo* | out\_dt\_FilteredResult |
| SendDocumentToFilarkiv | *in\_CaseId*  *in\_CaseNumber* | in\_dt\_CaseAndDocumentInfo |
| RegisterTransferInNova | *in\_dt\_CaseAndDocumentInfo*  *in\_KMDAPIToken* |  |
| LoginforToDataBase | *in\_dt\_CaseAndDocumentInfo*  *in\_CaseNumber* |  |
| CheckIfDataInStorageBucket | *in\_CaseNumber* | out\_DataInStorageBucket  out\_dt\_StorageBucket |
| End process |  |  |

# Fejlhåndtering

## Potentielle fejl

1. Hvis processen fejler efter at have overført dokumenterne til filarkiv, mangler det at blive registreret i KMDNova og logget til databasen. Der er derfor placeret en try catch omkring de to workflows. Fejler processen i et af disse, bliver listen med dokumenterne grebet i catchen, hvor de uploades til Orchestrator StorageBucket. Ved genkørslen, vil det oprindelige dokument listen, ikke indeholde nogen information, hvilke resulterer i at der tjekkes om listen i Storagebucket indeholder data. Hvis storagebucket indeholder data, vil denne blive hentes og efterfølgende logget og registreret i hhv. databasen og KMDNova. Er der ingen information, betyder det enten at sagen er kørt eller der ikke er nogen dokumenter, som mangler at blive overført.

## Undtagelser

1. Inderholder dokumenterne et af følgende ord, skal det ikke overføres til Filarkiv:
   1. Høringssvar
   2. Indsigelser
   3. Naboorientering
   4. Orientering + er udgående
   5. Information + er udgående
2. Hvis dokumenter er af filtypen .xlsx, skal det ikke overføres.

# Andre bemærkninger

## Nice to know

Angiv her eventuelle punkter, hvilke du anser som relevant for processen. Dette kunne være specifikke dynamiske selectors, ”invoke” code forklaringer og beskrivelse af specifikke kodeopbygnings valg.

|  |  |  |
| --- | --- | --- |
| Workflow | Omhandler | Forklaring |
| *GetDocumentInfo\_New* | *Levenstein algoritme* | *I workflowet skal de enkelte dokumenter (titler og beskrivelse) tjekkes igennem for en række blacklistede ord. For at gøre dette, bliver sætningerne brudt ned i til det enkelte ord, som tjekkes om dets længde er indenfor +-1 bogstav. Hvis det er dette, bliver det sat på en liste til videre brug. Ordene på denne liste bliver tjekket igennem vha. Levenstein algoritmen. Denne bryder ordene det i elementer og tjekker om hvert element, er lig med det element, som kontrol ordet har på den plads. Længden af ordet har en indflydelse på hvor høj/lav procentgrænsen sættes. Ved alle ord, skal der være plads til 2 fejl, så ved længere ord, skal procentgrænsen være højere.* |
| *Process – try catch – exception* | *Upload til Orchestrator storage bucket* | *Hvis robotten fejler I at logge information til databasen, bliver fejlen grebet i Catch-delen. Her uploadedes datatabellen som csv-format under navnet Casenumber + ”\_MissingInDatabase”. Således kan csv-filen efterfølgende downloaded og anvendes til hvert enkelt queueitem, hvis det bliver nødvendigt.* |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## Fremtidige forbedringer

* Hvis KMDNova på sigt opretter en overfør til Filarkiv funktion i API’et, kan denne erstatte netværkskaldet.