

AgileWorks-DocuSign

HULFT Square連携用Bricklet仕様書

1.0 版	2024 年 4 月 25 日
-------	-----------------

Brickletプログラム一覧・概要

Brickletプログラム一覧

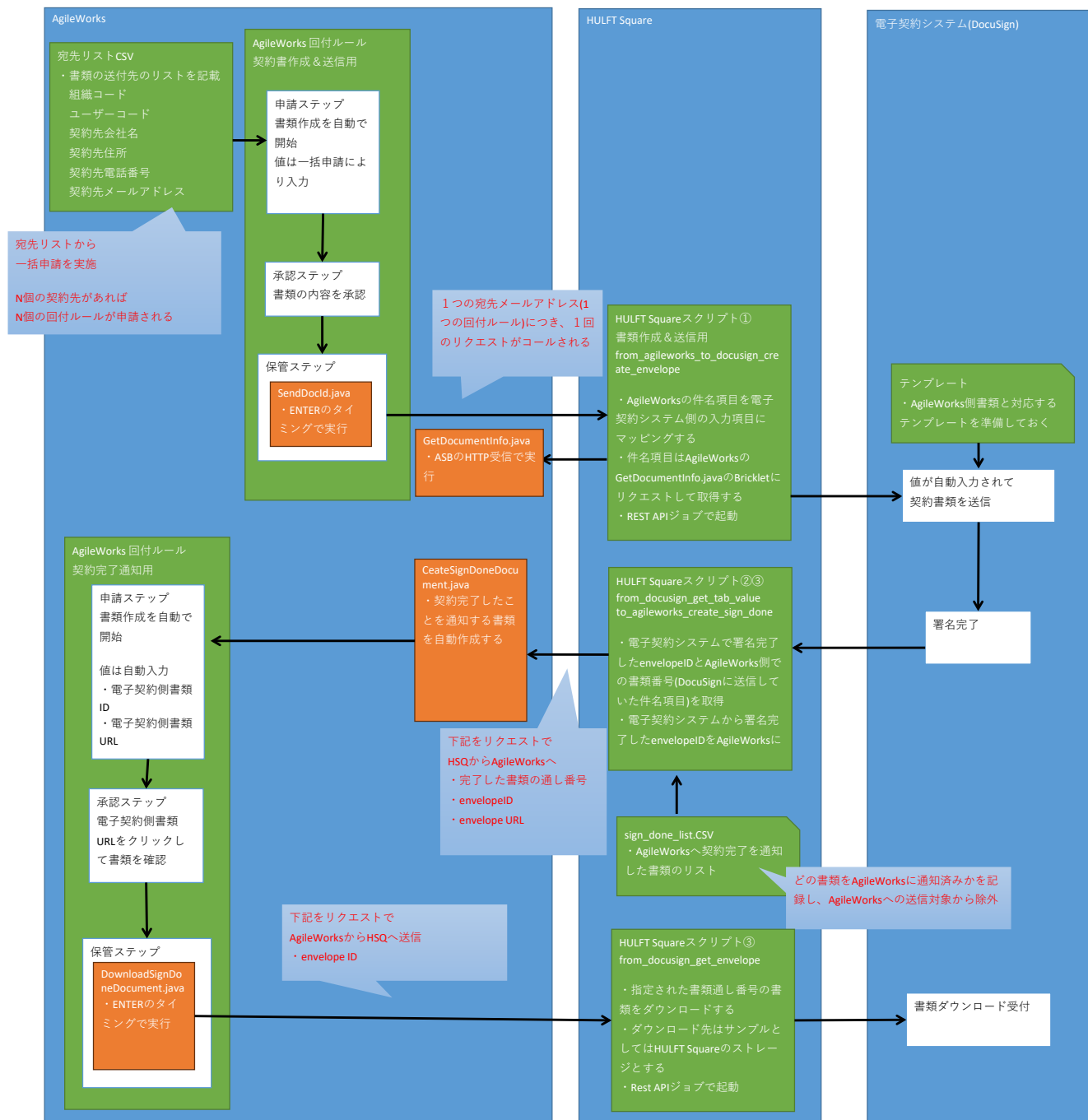
本アプリケーションの動作のために作成が必要なBrickletプログラムとその概要

Brickletプログラム	概要
SendDocId.java	AgileWorksからHULFT SquareのREST APIジョブに対して、AgileWorks側のdocIdを添えてHTTP GETリクエストを送信するBrickletプログラム
GetDocumentInfo.java	HTTPリクエストを受けてAgileWorks内の書類を検索し、その書類に登録されている件名項目をレスポンスとして返すBrickletプログラム
CreateSignDoneDocument.java	HTTPリクエストを受けてAgileWorksに契約完了確認用書類を自動作成して回付を開始する
DownloadSignDoneDocument.java	AgileWorksからHULFT SquareのREST APIジョブに対して、DocuSign側のenvelope idを添えてHTTP GETリクエストを送信するBrickletプログラム

Bricklet動作イメージ

以上のBrickletを組み合わせることで、下記のような動作を実現させることが可能である

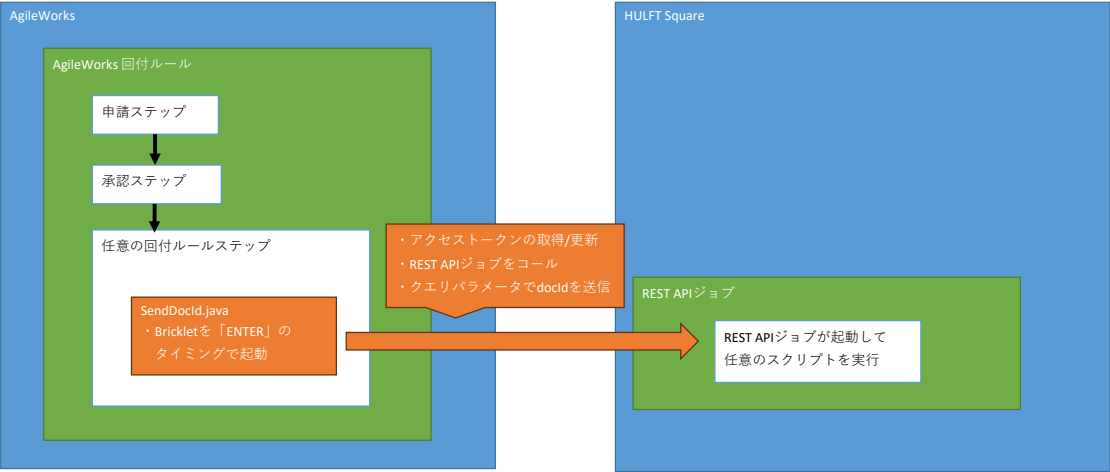
- ・AgileWorksでの書類の入力内容を件名項目として登録し、電子契約システム(DocuSign)へ送り書類を作成
- ・電子契約システム(DocuSign)で処理が完了したことを、AgileWorks側で新規確認用書類を作成することで通知し、完了書類のダウンロードをHULFT Squareを通じて行う



Bricklet「SendDocId」仕様

Bricklet処理概要

- AgileWorksからHULFT SquareのREST APIジョブに対して、HTTP GETリクエストを送信するBrickletプログラム
- ・ HULFT SquareのREST APIジョブのBearer認証のためのアクセストークンの取得/更新
 - ・ HULFT SquareのREST APIジョブをコールするHTTP GETリクエストの送信(クエリパラメータで、回付中書類のdocId<AgileWorks上での書類ID>を送信)
- ※本アプリケーションでは「agileworks_to_docusign_create_envelope」をREST APIジョブにしたものをコールする想定で作成されている



インポートクラス

クラス名
java.io.BufferedReader;
java.io.InputStreamReader;
java.io.PrintStream;
java.net.HttpURLConnection;
java.net.URL;
org.json.JSONObject;
jp.atled.workscape.papi.annotations.PapiInject;
jp.atled.workscape.papi.annotations.Procedure;
jp.atled.workscape.papi.model.plugin.CursorEventRequestModel;
jp.atled.workscape.papi.model.plugin.CursorEventResponseModel;
jp.atled.workscape.papi.service.logging.BrickletLogger;
jp.atled.workscape.papi.service.system.GatewayService;

主要変数

hulftSquareAuthURL1	https://app.square.hulft.com/v1/users/login
hulftSquareAuthURL2	https://app.square.hulft.com/v1/rest-api-token
accessToken1	"<HULFT SquareのREST APIクライアントのアクセストークン>"
accessToken2	"<HULFT SquareのREST APIクライアントのアクセストークン>"
authJsonBody1	<pre>"{" "email":"<HULFT SquareのREST APIクライアントのユーザー名>"," "password":"<HULFT SquareのREST APIクライアントのパスワード>"," }"</pre>
hulftSquareRestApiJobURL	.hulft.com/<HULFT Squareのプロファイル名>/<REST API名>/?docId=">https://利用しているHULFT Squareのドメイン>.hulft.com/<HULFT Squareのプロファイル名>/<REST API名>/?docId=

PapiInject

PapiInject	用途
GatewayService gatewayService	AgileWorksのHTTP操作に使用
BrickletLogger logger	AgileWorksのログ操作に使用

Bricklet処理詳細

public void sendDocId(CursorEventRequestModel req, CursorEventResponseModel res) throws Exception
AgileWorksからHULFT SquareのREST APIジョブに対して、HTTP GETリクエストを送信するBrickletプログラム

- ・ HULFT SquareのREST APIジョブのBearer認証のためのアクセストークンの取得/更新
- ・ HULFT SquareのREST APIジョブをコールするHTTP GETリクエストの送信(クエリパラメータで、 回付中書類のdocId<AgileWorks上での書類ID>を送信)

本Brickletを、電子契約対象の契約書の保管ステップで
ENTERイベントとして使用してください

※本アプリケーションでは「agileworks_to_docusign_create_envelope」をREST APIジョブにしたものをコールする想定で作成されている

下記URIへPUTリクエストを送信。HULFT Squareへの初期認証のためのアクセストークンを取得
https://app.square.hulft.com/v1/users/login

下記URIへPUTリクエストを送信。HULFT SquareへのAPIリクエストのためのアクセストークンを取得
https://app.square.hulft.com/v1/rest-api-token

下記URIへGETリクエストを送信。HULFT SquareのREST APIジョブを実行
https://<利用しているHULFT Squareのドメイン>.hulft.com/<HULFT Squareのプロファイル名>/<REST API名>/?docId=<docIdの値>

使用関数	用途
URL url	HTTPリクエストに使用
URL()	HTTPリクエストに使用
HttpURLConnection conn	HTTPリクエストに使用
(HttpURLConnection) url.openConnection()	HTTPリクエストに使用
conn.setRequestMethod("POST")	HTTPリクエストに使用
conn.setDoInput(true);	HTTPリクエストに使用
conn.setDoOutput(true);	HTTPリクエストに使用
conn.setDoOutput(true);	HTTPリクエストに使用
conn.setRequestProperty()	HTTPリクエストに使用
conn.connect()	HTTPリクエストに使用
PrintStream ps	HTTPリクエストに使用
PrintStream(conn.getOutputStream())	HTTPリクエストに使用
ps.print(authJsonBody1)	HTTPリクエストに使用
ps.close()	HTTPリクエストに使用
BufferedReader br	HTTPレスポンスの受信に使用
BufferedReader(new InputStreamReader(conn.getInputStream()), "UTF-8")	HTTPレスポンスの受信に使用
StringBuilder sb	HTTPレスポンスの受信に使用
new StringBuilder()	HTTPレスポンスの受信に使用
br.readLine()	HTTPレスポンスの受信に使用
sb.append()	HTTPレスポンスの受信に使用
br.close()	HTTPレスポンスの受信に使用
JSONObject jsonObj	HTTPレスポンスの受信に使用
new JSONObject(sb.toString())	HTTPレスポンスの受信に使用
jsonObj.getString()	HTTPレスポンスの受信に使用
req.getWorkflowInfo().getDocId()	AgileWorksのdocIdの取得に使用
logger.info()	AgileWorksでのログ出力に使用
logger.error()	AgileWorksでのログ出力に使用

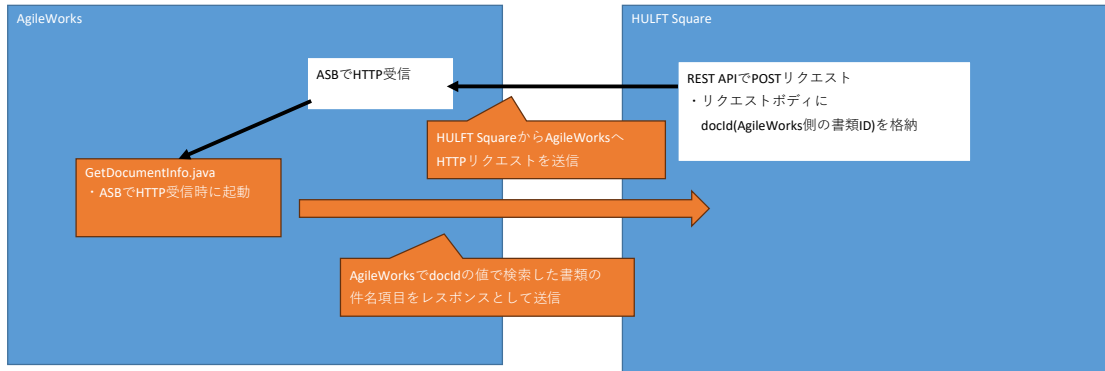
ログ仕様例

ログメッセージ	ログメッセージ	ログ発生タイミング
Error	Fail authorization for HULFT Square. Check user name and password	HULFT SquareのREST APIクライアントのユーザー名とパスワードに誤りがあるとき
Info	Request to HULFT Square https://<利用しているHULFT Squareのドメイン>.hulft.com/<HULFT Squareのプロファイル名>/<REST API名>/?docId=<docIdの値>	接続を試みたHULFT SquareのREST APIのURIを出力
Error	Fail to send request to HULFT Square. Check URL	HULFT SquareのREST APIへの接続に失敗したとき
Info	<HULFT Squareからのレスポンス内容>	HULFT SquareのREST APIへの接続後、受信したレスポンスを出力

Bricklet「GetDocumentInfo」仕様

Bricklet処理概要

- HTTPリクエストを受けてAgileWorks内の書類を検索し、その書類に登録されている件名項目をレスポンスとして返すBrickletプログラム
- ・リクエストボディでdocId(AgileWorks上での書類ID)を受信して書類を検索
 - ・検索で引き当てた書類の件名項目1～20をレスポンスとしてJSON形式で送信
- ※本アプリケーションでは「agileworks_to_docusign_create_envelope」の中でAgileWorks側に格納されているプログラムとして使用する



インポートクラス

クラス名
org.json.JSONException;
org.json.JSONObject;
jp.atled.workscape.papi.annotations.PapiInject;
jp.atled.workscape.papi.annotations.Procedure;
jp.atled.workscape.papi.model.common.EntryListModel;
jp.atled.workscape.papi.model.common.EnumStatusType;
jp.atled.workscape.papi.model.common.RecordListModel;
jp.atled.workscape.papi.model.common.RecordModel;
jp.atled.workscape.papi.model.common.SelectResponseModel;
jp.atled.workscape.papi.model.doc.DocViewColumnModel;
jp.atled.workscape.papi.model.doc.EnumDocViewColumnType;
jp.atled.workscape.papi.model.doc.SelectDocConditionModel;
jp.atled.workscape.papi.model.doc.SelectDocRequestModel;
jp.atled.workscape.papi.model.system.AsbCallRequestModel;
jp.atled.workscape.papi.model.system.AsbCallResponseModel;
jp.atled.workscape.papi.service.doc.DocService;
jp.atled.workscape.papi.service.Logging.BrickletLogger;
jp.atled.workscape.papi.service.org.OrgService;
jp.atled.workscape.papi.service.system.StorageManager;

主要変数

docId	<AgileWorksのdocId> ※HTTPのPOSTリクエストのボディから受信
-------	---

PapiInject

PapiInject	用途
DocService docService	AgileWorksのドキュメント操作に使用
OrgService orgService	AgileWorksの組織操作に使用
StorageManager storageManager	AgileWorksのアーカイブ操作に使用
BrickletLogger logger	AgileWorksのログ操作に使用

Bricklet処理詳細

```
public void getDocumentInfo(AsbCallRequestModel req, AsbCallResponseModel res)
```

HULFT SquareからのHTTPリクエストを受けて

AgileWorks内の書類を検索して

その書類に登録されている件名項目をレスポンスとして返すBrickletプログラム

・リクエストボディでdocId(AgileWorks上での書類ID)を受信して書類を検索

・検索で引き当てた書類の件名項目1～20をレスポンスとしてJSON形式で送信

使用関数	用途
JSONObject jsonObj	リクエストボディのJSONの読み取りに使用
JSONObject(req.getMessage().getText())	リクエストボディのJSONの読み取りに使用
Long.parseLong(jsonObj.getString())	リクエストボディのJSONの読み取りに使用
SelectDocRequestModel selectRequest	AgileWorksのドキュメント検索に使用
docService.createRequestModel()	AgileWorksのドキュメント検索に使用
SelectDocConditionModel condition	AgileWorksのドキュメント検索に使用
selectRequest.getCondition()	AgileWorksのドキュメント検索に使用
condition.setDocId(docId)	AgileWorksのドキュメント検索に使用
DocViewColumnModel docColumnName1	AgileWorksの件名項目の値の取得に仕様(件名項目1件目の例)
selectRequest.getDocView().getColumnList().add()	AgileWorksの件名項目の値の取得に仕様(件名項目1件目の例)
docColumnName1.setType(EnumDocViewColumnType.DOC_LISTNAME1)	AgileWorksの件名項目の値の取得に仕様(件名項目1件目の例)
SelectResponseModel selectResponse	レスポンス用のJSONデータの作成に使用
docService.selectDoc(selectRequest)	レスポンス用のJSONデータの作成に使用
JSONObject resJson	レスポンス用のJSONデータの作成に使用
JSONObject()	レスポンス用のJSONデータの作成に使用
RecordListModel recordList	レスポンス用のJSONデータの作成に使用
selectResponse.getResultSet().getRecordList()	レスポンス用のJSONデータの作成に使用
RecordModel record : recordList	レスポンス用のJSONデータの作成に使用
EntryListModel entryList	レスポンス用のJSONデータの作成に使用
record.getEntryList()	レスポンス用のJSONデータの作成に使用
resJson.put("docListName1",entryList.get(0).getValue())	レスポンス用のJSONデータの作成に使用(件名項目1件目の例)
res.getMessage().setText(resJson.toString())	レスポンス用のJSONデータの作成に使用
logger.info()	AgileWorksでのログ出力に使用
logger.error()	AgileWorksでのログ出力に使用

ログ仕様例

ログメッセージ	ログメッセージ	ログ発生タイミング
Info	"Start Search docId:" <HTTPリクエストで受信したdocId>	AgileWorksを検索するdocIdの値を出力
Info	Search Succeed	docIdの値でAgileWorks内に書類が存在したときに出力
Error	Search Error	docIdの値でAgileWorks内に書類が存在しないときに出力
Error	<格納文字列をそのまま出力>	未知のエラーのときに出力

HTTPリクエスト/レスポンス仕様例

本プログラムはAgileWorksのASBでHTTPエンドポイントとして機能する

リクエストとレスポンスの形式は下記の通り

※本アプリケーションのスキーマはHULFT Squareは下記のリクエスト/レスポンスを前提に作成している

リクエスト仕様

リクエストURL	AgileWorksのASBで指定	
メソッド	POST	
ヘッダ	なし	
クエリパラメータ	なし	
リクエストボディサンプル	{ "docId": string }	
リクエストボディ項目詳細	docId	AgileWorks上の書類ID
文字コード	utf-8	

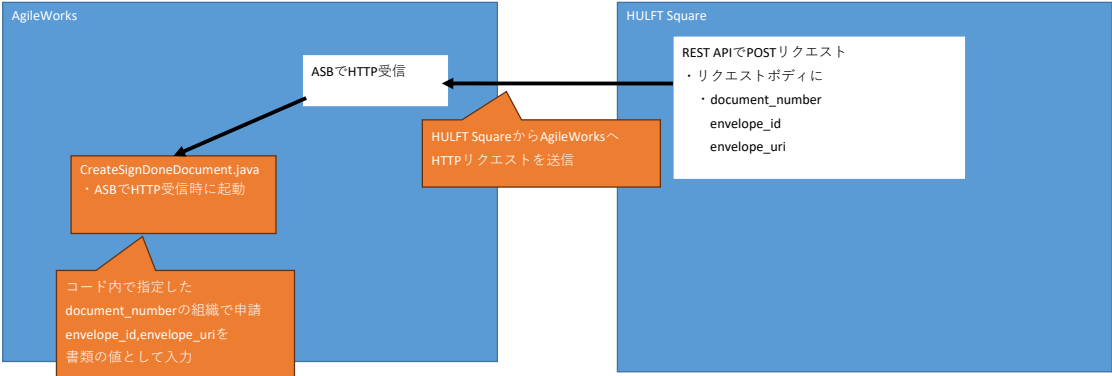
レスポンス仕様

MIMEタイプ	application/json	
レスポンスボディサンプル	{ "docListName9":"","docListName13":"","docListName12":"","docListName11":"","docListName10":"","docListName17" :","docListName16":"","docListName15":"","docListName14":"","docListName1":"","00000036","docListName2":"","2023- 08-25","docListName3":"","XXYY契約 ","docListName19":"","docListName4":"","xx","docListName18":"","docListName5":"","yy","docListName6":"","docListName 7":"","docListName8":"","docListName20":"","" }	AgileWorksから検索した書類の件名項目を出力
文字コード	utf-8	

Bricklet「CreateSignDoneDocument」仕様

Bricklet処理概要

- HTTPリクエストを受けてAgileWorksに契約完了確認書類を自動作成して回付を開始するBrickletプログラム
- ・ リクエストボディで下記を受信
 - 件名項目1
 - ※件名項目1は「書類通し番号」などの他書類と比較して一意の値のものである必要がある
 - 電子契約側書類ID
 - 電子契約側書類URL
- ※本アプリケーションでは「agileworks_create_sign_done」の中でAgileWorks側に格納されているプログラムとして使用する



インポートクラス

クラス名
java.sql.SQLException
java.util.Date
org.json.JSONObject
jp.atled.workscape.papi.annotations.PapiInject
jp.atled.workscape.papi.annotations.Procedure
jp.atled.workscape.papi.model.common.EntryListModel
jp.atled.workscape.papi.model.common.EnumStatusType
jp.atled.workscape.papi.model.common.RecordListModel
jp.atled.workscape.papi.model.common.RecordModel
jp.atled.workscape.papi.model.common.SelectResponseModel
jp.atled.workscape.papi.model.doc.DocModel
jp.atled.workscape.papi.model.doc.DocResponseModel
jp.atled.workscape.papi.model.doc.DocViewColumnModel
jp.atled.workscape.papi.model.doc.EnumDocViewColumnType
jp.atled.workscape.papi.model.doc.PrepareDocRequestModel
jp.atled.workscape.papi.model.doc.PrepareDocResponseModel
jp.atled.workscape.papi.model.doc.SelectDocConditionModel
jp.atled.workscape.papi.model.doc.SelectDocRequestModel
jp.atled.workscape.papi.model.system.AsbCallRequestModel
jp.atled.workscape.papi.model.system.AsbCallResponseModel
jp.atled.workscape.papi.model.workflow.StartRequestModel
jp.atled.workscape.papi.service.doc.DocService
jp.atled.workscape.papi.service.Logging.BrickletLogger
jp.atled.workscape.papi.service.org.OrgService
jp.atled.workscape.papi.service.system.DatabaseHandler
jp.atled.workscape.papi.service.system.StorageManager
jp.atled.workscape.papi.service.system.SystemManager
jp.atled.workscape.papi.service.system.Transaction
jp.atled.workscape.papi.service.workflow.WorkflowService

主要変数

listName1	<AgileWorksの件名項目の1行目> ※HTTPのPOSTリクエストのボディから受信
externalDocumentId	<DocuSign側のenvelope_id> ※HTTPのPOSTリクエストのボディから受信
externalDocumentUri	<DocuSign側のenvelope_uri> ※HTTPのPOSTリクエストのボディから受信
applyUserCode	電子契約完了通知をAgileWorksで申請するユーザーのユーザーコードを入力
applyUnitCode	電子契約完了通知をAgileWorksで申請するユーザーの組織コードを入力(本Brickletプログラム内で自動取得する想定)
criterionDate	電子契約完了通知の作成日 new Date()
formCode	電子契約完了通知のフォームコードを入力
ruleCode	電子契約完了通知の回付ルールコードを入力

PapilInject

PapilInject	用途
WorkflowService workflowService	AgileWorksの回付ルール操作に使用
DocService docService	AgileWorksのドキュメント操作に使用
SystemManager systemManager	AgileWorksのシステム操作に使用
OrgService orgService	AgileWorksの組織操作に使用
StorageManager storageManager	AgileWorksのデータベース操作に使用
BrickletLogger logger	AgileWorksのログ操作に使用

Bricklet処理詳細

public void createSignDoneDocument(AsbCallRequestModel req, AsbCallResponseModel res) throws SQLException

HULFT SquareからのHTTPリクエストを受けて

AgileWorks上で電子契約完了通知のための申請を自動作成して

電子契約完了通知の回付を開始するBrickletプログラム

・リクエストボディで下記を受信

件名項目1

※件名項目1は「書類通し番号」などの他書類と比較して一意の値のものである必要がある

電子契約側書類ID

電子契約側書類URL

使用関数	用途
JSONObject jsonObj	リクエストボディのJSONの読み取りに使用
JSONObject(req.getMessage().getText())	リクエストボディのJSONの読み取りに使用
jsonObj.getString()	リクエストボディのJSONの読み取りに使用
SelectDocRequestModel selectRequest	AgileWorksのドキュメント検索に使用
docService.createRequestModel()	AgileWorksのドキュメント検索に使用
SelectDocConditionModel condition	AgileWorksのドキュメント検索に使用
selectRequest.getCondition()	AgileWorksのドキュメント検索に使用
condition.setListNameKeyword(listName1)	AgileWorksのドキュメント検索に使用
DocViewColumnModel colApplyUnitCode	AgileWorksの組織コードの取得に使用
selectRequest.getDocView().getColumnList().add()	AgileWorksの組織コードの取得に使用
colApplyUnitCode.setType(EnumDocViewColumnType.APPLY_UNIT_CODE)	AgileWorksの組織コードの取得に使用
SelectResponseModel selectResponse	AgileWorksの組織コードの取得に使用
docService.selectDoc(selectRequest)	AgileWorksの組織コードの取得に使用
RecordListModel recordList	AgileWorksの組織コードの取得に使用
selectResponse.getResultSet().getRecordList()	AgileWorksの組織コードの取得に使用
RecordModel record : recordList	AgileWorksの組織コードの取得に使用
EntryListModel entryList	AgileWorksの組織コードの取得に使用
record.getEntryList()	AgileWorksの組織コードの取得に使用
entryList.get(0).getValue()	AgileWorksの組織コードの取得に使用
DatabaseHandler dbh	AgileWorksでの電子契約完了通知の作成に使用
systemManager.getDatabaseHandler()	AgileWorksでの電子契約完了通知の作成に使用
Transaction transaction	AgileWorksでの電子契約完了通知の作成に使用
dbh.transaction()	AgileWorksでの電子契約完了通知の作成に使用
transaction.start()	AgileWorksでのデータベース処理制御のために使用
PrepareDocRequestModel prepareDocRequest	AgileWorksでの電子契約完了通知の作成に使用
docService.createRequestModel(PrepareDocRequestModel.class)	AgileWorksでの電子契約完了通知の作成に使用
prepareDocRequest.setCriterionDate(criterionDate)	AgileWorksでの電子契約完了通知の作成に使用
prepareDocRequest.setApplyUnitCode(applyUnitCode)	AgileWorksでの電子契約完了通知の作成に使用
prepareDocRequest.setApplyUserCode(applyUserCode)	AgileWorksでの電子契約完了通知の作成に使用
prepareDocRequest.setFormCode(formCode)	AgileWorksでの電子契約完了通知の作成に使用
prepareDocRequest.setRuleCode(ruleCode)	AgileWorksでの電子契約完了通知の作成に使用
PrepareDocResponseModel prepareDocResponse	AgileWorksでの電子契約完了通知の作成に使用
docService.prepareDocRequest(prepareDocRequest)	AgileWorksでの電子契約完了通知の作成に使用
DocModel doc = prepareDocResponse.getDoc()	AgileWorksでの電子契約完了通知の作成に使用
DocResponseModel docResponse = docService.addDoc(doc)	AgileWorksでの電子契約完了通知の作成に使用
doc = docResponse.getDoc()	AgileWorksでの電子契約完了通知の作成に使用
StartRequestModel startRequest	AgileWorksでの電子契約完了通知の作成に使用
workflowService.createRequestModel(StartRequestModel.class)	AgileWorksでの電子契約完了通知の作成に使用
startRequest.setApplyUnitCode(applyUnitCode)	AgileWorksでの電子契約完了通知の作成に使用
startRequest.setApplyUserCode(applyUserCode)	AgileWorksでの電子契約完了通知の作成に使用
startRequest.setCriterionDate(criterionDate)	AgileWorksでの電子契約完了通知の作成に使用
startRequest.setDocId(doc.getId())	AgileWorksでの電子契約完了通知の作成に使用
startRequest.setFormCode(formCode)	AgileWorksでの電子契約完了通知の作成に使用
startRequest.setRuleCode(ruleCode)	AgileWorksでの電子契約完了通知の作成に使用
workflowService.start(startRequest)	AgileWorksでの電子契約完了通知の申請開始に使用
doc.getDocData().setFieldData()	DocuSignから取得した封筒IDと封筒URIを反映するのに使用
docService.updateDoc(doc)	DocuSignから取得した封筒IDと封筒URIを反映するのに使用
transaction.commit()	AgileWorksのデータベース処理制御のために使用
res.getMessage().setText()	HTMLのレスポンス用データを作成するのに使用
transaction.rollback()	AgileWorksのデータベース処理制御のために使用
logger.info()	AgileWorksでのログ出力に使用
logger.error()	AgileWorksでのログ出力に使用

ログ仕様例

ログメッセージ	ログメッセージ	ログ発生タイミング
error	search error	AgileWorks側で該当書類が見つからないときに出力
Info	succeed Create Sign Done Document	電子契約完了通知の申請に成功したときに出力
Error	fail Create Sign Done Document	電子契約完了通知の申請に失敗したときに出力

HTTPリクエスト/レスポンス仕様例

本プログラムはAgileWorksのASBでHTTPエンドポイントとして機能する
リクエストとレスポンスの形式は下記の通り
※本アプリケーションのスキーマはHULFT Squareは下記のリクエスト/レスポンスを前提に作成している

リクエスト仕様

リクエストURL	AgileWorksのASBで指定	
メソッド	POST	
ヘッダ	なし	
クエリパラメータ	なし	
リクエストボディサンプル	{ "document_number": string "document_id": string "document_uri": string }	
リクエストボディ項目詳細	document_number	AgileWorks上の件名項目の1行目
	document_id	Docusign側のenvelope id
	document_uri	Docusign側のenvelope uri
文字コード	utf-8	

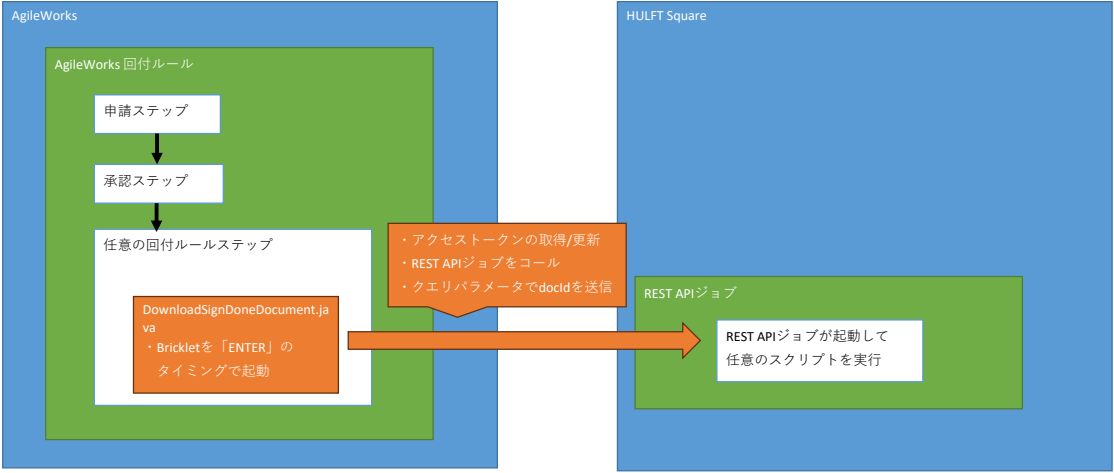
レスポンス仕様

MIMEタイプ	application/json	
レスポンスボディサンプル	succeed	電子契約完了通知の申請に成功したことを通知
文字コード	utf-8	

Bricklet「DownloadSignDoneDocument」仕様

Bricklet処理概要

- AgileWorksからHULFT SquareのREST APIジョブに対して、HTTP GETリクエストを送信するBrickletプログラム
- ・ HULFT SquareのREST APIジョブのBearer認証のためのアクセストークンの取得/更新
 - ・ HULFT SquareのREST APIジョブをコールするHTTP GETリクエストの送信(クエリパラメータで回付中書類書類の件名項目1<DocuSign側のenvelope_idと同等のものを指定>を送信)
- ※本アプリケーションでは「from_docusign_get_envelope」をREST APIジョブにしたものをコールする想定で作成されている



インポートクラス

クラス名	
java.io.BufferedReader	
java.io.InputStreamReader	
java.io.PrintStream	
java.net.HttpURLConnection	
java.net.URL	
org.json.JSONObject	
jp.atled.workscape.papi.annotations.PapiInject	
jp.atled.workscape.papi.annotations.Procedure	
jp.atled.workscape.papi.model.common.EntryListModel	
jp.atled.workscape.papi.model.common.EnumStatusType	
jp.atled.workscape.papi.model.common.RecordListModel	
jp.atled.workscape.papi.model.common.RecordModel	
jp.atled.workscape.papi.model.common.SelectResponseModel	
jp.atled.workscape.papi.model.doc.DocViewColumnModel	
jp.atled.workscape.papi.model.doc.EnumDocViewColumnType	
jp.atled.workscape.papi.model.doc.SelectDocConditionModel	
jp.atled.workscape.papi.model.doc.SelectDocRequestModel	
jp.atled.workscape.papi.model.plugin.CursorEventRequestModel	
jp.atled.workscape.papi.model.plugin.CursorEventResponseModel	
jp.atled.workscape.papi.service.doc.DocService	
jp.atled.workscape.papi.service.logging.BrickletLogger	
jp.atled.workscape.papi.service.system.GatewayService	

主要変数

hulftSquareAuthURL1	https://app.square.hulft.com/v1/users/login
hulftSquareAuthURL2	https://app.square.hulft.com/v1/rest-api-token
accessToken1	"<HULFT SquareのREST APIクライアントのアクセストークン>"
accessToken2	"<HULFT SquareのREST APIクライアントのアクセストークン>"
authJsonBody1	<pre>"{"email":"<HULFT SquareのREST APIクライアントのユーザー名>","password":"<HULFT SquareのREST APIクライアントのパスワード>"}</pre>
hulftSquareRestApiJobURL	<a href="https://<利用しているHULFT Squareのドメイン>.hulft.com/<HULFT Squareのプロファイル名>/<REST API名>/?docid=">https://<利用しているHULFT Squareのドメイン>.hulft.com/<HULFT Squareのプロファイル名>/<REST API名>/?docid=
docListName1Value	AgileWorksの書類の件名項目の1行目を入力する

PapiInject

PapiInject	用途
DocService docService	AgileWorksのドキュメント操作に使用
GatewayService gatewayService	AgileWorksのHTTP操作に使用
BrickletLogger logger	AgileWorksのログ操作に使用

Bricklet処理詳細

public void downloadSignDoneDocument(CursorEventRequestModel req, CursorEventResponseModel res) throws Exception
AgileWorksからHULFT SquareのREST APIジョブに対して、HTTP GETリクエストを送信するBrickletプログラム

- ・ HULFT SquareのREST APIジョブのBearer認証のためのアクセストークンの取得/更新
- ・ HULFT SquareのREST APIジョブをコールするHTTP GETリクエストの送信(クエリパラメータで回付中書類書類の件名項目1<DocuSign側のenvelope_idと同値のものを指定>を送信)

本Brickletを、電子契約完了通知の保管ステップで
ENTERイベントとして使用してください

※本アプリケーションでは「from_docusign_get_envelope」をREST APIジョブにしたものをコールする想定で作成されている

回付書類の件名項目の1行目を取得する

下記URIへPOSTリクエストを送信。HULFT Squareへの初期認証のためのアクセストークンを取得
https://app.square.hulft.com/v1/users/login

下記URIへPUTリクエストを送信。HULFT SquareへのAPIリクエストのためのアクセストークンを取得
https://app.square.hulft.com/v1/rest-api-token

下記URIへGETリクエストを送信。HULFT SquareのREST APIジョブを実行
https://<利用しているHULFT Squareのドメイン>.hulft.com/<HULFT Squareのプロファイル名>/<REST API名>/?envelope_id=<envelope_idの値>

使用関数	用途
SelectDocRequestModel selectRequest	AgileWorksのドキュメント検索に使用
docService.createRequestModel(SelectDocRequestModel.class)	AgileWorksのドキュメント検索に使用
SelectDocConditionModel condition	AgileWorksのドキュメント検索に使用
selectRequest.getCondition()	AgileWorksのドキュメント検索に使用
condition.setDocId(req.getWorkflowInfo().getDocId())	AgileWorksのドキュメント検索に使用
DocViewColumnModel docListName1	AgileWorksのドキュメント検索に使用
selectRequest.getDocView().getColumnList().add()	AgileWorksのドキュメント検索に使用
docListName1.setType(EnumDocViewColumnType.DOC_LISTNAME1)	AgileWorksのドキュメント検索に使用
SelectResponseModel selectResponse	AgileWorksのドキュメント検索結果の取得に使用
docService.selectDoc(selectRequest)	AgileWorksのドキュメント検索結果の取得に使用
selectResponse.getResultStatus().getStatus()	AgileWorksのドキュメント検索結果の取得に使用
RecordListModel recordList	AgileWorksのドキュメント検索結果の取得に使用
selectResponse.getResultSet().getRecordList()	AgileWorksのドキュメント検索結果の取得に使用
RecordModel record : recordList	AgileWorksのドキュメント検索結果の取得に使用
EntryListModel entryList	AgileWorksのドキュメント検索結果の取得に使用
record.getEntryList()	AgileWorksのドキュメント検索結果の取得に使用
entryList.get(0).getValue()	AgileWorksのドキュメント検索結果の取得に使用
URL url	HTTPリクエストに使用
URL()	HTTPリクエストに使用
HttpURLConnection conn	HTTPリクエストに使用
(HttpURLConnection) url.openConnection()	HTTPリクエストに使用
conn.setRequestMethod("POST")	HTTPリクエストに使用
conn.setDoInput(true);	HTTPリクエストに使用
conn.setDoOutput(true);	HTTPリクエストに使用
conn.setDoOutput(true);	HTTPリクエストに使用
conn.setRequestProperty()	HTTPリクエストに使用
conn.connect()	HTTPリクエストに使用
PrintStream ps	HTTPリクエストに使用
PrintStream(conn.getOutputStream())	HTTPリクエストに使用
ps.print(authJsonBody1)	HTTPリクエストに使用
ps.close()	HTTPリクエストに使用
BufferedReader br	HTTPレスポンスの受信に使用
BufferedReader(new InputStreamReader(conn.getInputStream(), "UTF-8"))	HTTPレスポンスの受信に使用
StringBuilder sb	HTTPレスポンスの受信に使用
new StringBuilder()	HTTPレスポンスの受信に使用
br.readLine()	HTTPレスポンスの受信に使用
sb.append()	HTTPレスポンスの受信に使用
br.close()	HTTPレスポンスの受信に使用
JSONObject jsonObj	HTTPレスポンスの受信に使用
new JSONObject(sb.toString())	HTTPレスポンスの受信に使用
jsonObj.getString()	HTTPレスポンスの受信に使用
logger.info()	AgileWorksでのログ出力に使用
logger.error()	AgileWorksでのログ出力に使用

ログ仕様例

ログメッセージ	ログメッセージ	ログ発生タイミング
Error	Fail authorization for HULFT Square. Check user name and password	HULFT SquareのREST APIクライアントのユーザー名とパスワードに誤りがあるとき
Error	Fail authoriation for HULFT Square. Check refresh token	HULFT SquareのREST APIクライアントのリフレッシュトークンに誤りがあるとき
Info	Request to HULFT Square <hulftSquareApiUriの値>	接続を試みたHULFT SquareのREST APIのURIを出力
Error	Fail to send request to HULFT Square. Check URL	HULFT SquareのREST APIへの接続に失敗したとき
Info	<HULFT Squareからのレスポンス内容>	HULFT SquareのREST APIへの接続後、受信したレスポンスを出力