1.		In	ntroduktion	2
2.		Vi	illkor för exempelintegration	2
3.		0	orderöverföring Webbshop till Affärssystem	2
	3.1.		Shopify2Fortnox lösning	3
	3.2.		Authorization till Shopify	3
	3.3.		Authorization till Fortnox	4
	2.4.		Kanal i Bosbec	5
	2.5.		Bygga authentification för Fortnox i Bosbec	5
	2.6.		Bygga orderflödet I Bosbec	8
3.		SI	lutord	25

bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701

1. Introduktion

Bosbec är en automatiseringsplattform baserad på low-code som gör det möjligt att enkelt och säkert bygga dina automatiserings- och integrationslösningar. Du skapar dessa lösningar med hjälp av en workflow builder.

Plattformen stöder olika kanaler för att ta emot och skicka data, till exempel http, e-post, textmeddelanden och mer, vilket möjliggör sömlös kommunikation med andra tjänster och system. När du har byggt din lösning kan du omedelbart distribuera den genom att trycka på "live", så att den är redo att användas direkt.

Till skillnad från liknande plattformar där du kan behöva skriva dina egna skript, tillhandahåller Bosbec fördefinierade jobb för olika uppgifter. Detta tillvägagångssätt säkerställer att dina lösningar ständigt testas och verifieras för optimal prestanda. Dessutom erbjuder Bosbec en dynamisk och flexibel miljö, så att du enkelt kan justera och ändra dina arbetsflöden för att möta förändrade behov och krav.

2. Villkor för exempelintegration

I denna integration så finns det vissa villkor om man använder den exakt som den är byggd. Dessa villkor kan du enkelt ändra om du behöver i ditt workflow.

- Orderstatus sätts som Fullfilled.
- Vi skapar kund från Shopify men vi uppdaterar inte kunden om den redan finns i Fortnox.
- Billing adress i Shopify sätts som Kundens adress i Fortnox och Shipping adress sätts som leveransadress i Fortnox
- Kundnumret från Shopify blir kundnummer i Fortnox
- Orderinformation från Shopify läggs in i Fortnox där varje produkt får en egen rad och anges vid benämning utan artikelnummer med antal och pris.
- Frakt läggs in i Fortnox från Shopify.
- Momsen är fast på 25%. I detta exempel så är det en webbshop som säljer till konsument med pris inkl. moms.
- För att bokföra ordern i Fortnox så görs följande;
 - Order document nummer från Shopify används för att ange att produkt är levererad
 - Faktura skapas som "cash invoice"
 - o Fakturan bokförs

3. Orderöverföring Webbshop till Affärssystem

bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701

Varje ny order som kommer in från en webbshop går direkt till Fortnox.

I detta dokument beskrivs steg för steg hur en integration mellan Shopify och Fortnox byggts med Bosbecs plattform. Eftersom detta är ett exempel så bör det poängteras att det konceptuellt lika gärna kan vara en annan webbshop såväl som affärssystem.

Följande steg gås igenom:

- 1. Integrationen mot Fortnox
- 2. Integrationen mot Shopify
- 3. Mappningen och hantering av data mellan Shopify och Fortnox i Bosbecs lösning

3.1. Shopify2Fortnox lösning

Som partner till Bosbec kan ni skapa en integration som ni kan publicera i Fortnox eller skapa en integration direkt för en kund. Vill ni skapa en egen app i Fortnox så finns mer info här <u>https://support.fortnox.se/komigangguide-tjanster/kom-igang-med-utvecklarportalen</u>.

3.2. Authorization till Shopify

För att komma åt Shopify API och generera API-nycklar från din Shopify-administratörspanel, följ dessa steg:

- 1. Logga in på ditt Shopify-konto.
- 2. Navigera till "Appar".
- 3. Gå till "App- och Sales Channel Listing".
- 4. Navigera till "Utveckla appar".
- 5. Klicka på knappen "Skapa en app".
- 6. Ge appen ett lämpligt namn och klicka på "Skapa app".
- 7. Efter att ha skapat appen, gå till fliken "Konfigurera".
- 8. Klicka på knappen "Konfigurera" för Admin API-integration, där du kan ställa in behörigheterna (scopes) för din app, till exempel läs- och skrivåtkomst till olika Shopify-API:er. Välj alla nödvändiga behörigheter för din app och klicka på "Spara".
- 9. I fliken "API-referenser" får du API-nyckeln och API-secret nyckel. API-nyckeln används för autentisering.
- 10. Klicka på knappen "Installera app" för att installera den anpassade appen du har ställt in.
- 11. Efter att du har installerat appen får du "Admin API Access Token", som är din åtkomsttoken för att begära data från Admin API.

Nästa steg är att skapa en webbhook i Shopify. Du kan läsa på följande länk hur du gör <u>https://shopify.dev/docs/apps/webhooks</u>.

bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701

För att skapa endpoint i Bosbec se avsnitt 2.4.

3.3. Authorization till Fortnox

Authorization processen I Fortnox använder sig av OAuth2. Den gäller i två steg;

- 1. Först får man en authorization kod där användaren godkänner applikationen och tillgången till kontot.
- 2. Sedan genereras access token och refresh token.

En övergripande beskrivning finns här https://www.fortnox.se/developer/authorization

Till att börja med skapar man en url som användaren ska klicka på. För att skapa denna behöves följande parametrar;

- Authorisation URL som är https://apps.fortnox.se/oauth-v1/auth
- Client-ID är den publika identifieraren för appen.
- Redirect-uri Kanal som skapas i Bosbec
- Scope (lista på scope finns på denna länk <u>https://www.fortnox.se/developer/guides-and-good-to-know/scopes</u>)
- State parametern används för att lagra begärans specifika data och/eller förhindra CSRF-attacker. Ni kan välja själv ett nummer.
- Access_type Anger om appen kan uppdatera access-tokens när användaren inte är närvarande i webbläsaren.
- Response_type (required) Response_type ska vara code, vilket anger att programmet förväntar sig att få en auktoriseringskod om det lyckas.
- Account_type (valfritt) Anger om ett servicekonto ska skapas. Servicekontot måste vara aktiverat om du använder Developer Portal. Ett servicekonto är inte kopplat till någon specifik användare och har en specifik uppsättning behörigheter som är lämpliga för integrationer inom de begärda omfattningarna. Det kan bara finnas ett servicekonto per client_id och kund. Endast systemadministratörer hos kunden kan auktorisera servicekonton under auktoriseringsprocessen. Det enda giltiga värdet är "service", om ett servicekonto ska skapas.

URL för OAuth 2.0-auktoriseringsbegäran för Fortnox API kommer att se ut så här:

https://apps.fortnox.se/oauth-v1/auth?client_id={Client-ID}&redirect_uri={kanal från_ Bosbec}&scope={companyinformation order customer article bookkeeping invoice_ payment}&state=12345&access_type=offline&response_type=code&account_type=service

För denna integration som hanterar order så har vi använt följande scope (i url):

companyinformation

- order

bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701

- customer
- article
- bookkeeping
- invoice
- payment

Om man vill integrera annat system till samma flöde som t.ex. POS-system, lagerhantering etc kan man behöver fler scopes.

2.4. Kanal i Bosbec

Vi börjar med att skapa en kanal i Bosbec.

1. I plattformen går man till "My Channels" och skapar en http kanal

	<	Q, Search		⊙ A (€) serie	naoven Ags v
bosbe	ome	Channels		0	
ome		9. Search	Incoming Cutpoling	Create a new channel Select channel type	×
ervices	~			Incoming channels	
orkflows	^	Default channel	Default channel	SMS	
My Workflows		6	6	Final Const	
My Channels		4	4		
esources	~	2	2	Ф нттр	
dministrator Tools	~	0	0	🛨 Event	
tatistics	~	VIEW HISTORY	VIEW HISTORY	MOTT	
sport	~	Demo NHttp channel	Fortnox authentification (shopify)	A Group	
		6	6	Outgoing channels	
		4	4	Email	
		2	2		
T		0	0		
			Authenticate Fortnox		

2. Fyller i och sparar kanalen.

W	Http channel	>
Mesr http Upda recei	lages sent using any HTTP-request) to sc/your_name.in.bosbec.io; will be sent to this channe des will be visible in the diagram when messages are ved.	
Fort	nox	
Aut	ription horization	
Met	od	
SLAN	tomain r_name	
in.b	an osbec.io	0
Patl	5	
0	Add a default response	

2.5. Bygga authentification för Fortnox i Bosbec

bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701

Till att börja med så öppnar du ett workflow i Bosbec. Det gör du under "Workflows" och sedan "My Workflows". Du väljer "Create new workflow" och sedan "Empty workflow".

Nu kan du börja bygga din lösning och vi börjar med authentification processen.



För Fortnox authentification finns det en template så du snabbt får upp ett workflow. Du väljer då "Edit" och sedan "Workflow library" och sedan din Fortnox template.



När du sedan importerar workflowet får du det på din arbetsyta.



Nu kan du börja konfigurera ditt workflow. Fördelen med att få det i workflow är att om Fortnox ändrar något i sin process är det enkelt att sedan justera det i workflowet.

1. Börja med att skapa en grupp i Bosbec där du sparar din Token. Detta för att kunna hantera refresh token i processen. Det gör du under "Resources" och "Groups".

bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701



2. Börja med att välja den kanal du skapat. Väljer du att kryssa i "Is Public" så behöver du ingen token. Det kan vara bra när du testar. När du sedan vill att det ska vara skarpt så skapar du en token och det gör du under "Administrator Tools" och "API Tokens".

	Incoming http trigger - Trigger_1	{} ⑦ ×
Http request resource	Select channels to listen to.	*
http_request_resource Resource from trigger_1 Reso	All channels Capture all incoming messages	
	Default channel Http channel	
Incoming http trigger	Demo Http channel	
trigger_1 Description	Fortnox authentification (shopify) Http channel	

- 3. Nu kan du öppna job_4 och fylla i följande uppgifter under "Source".
 - Client ID
 - Client Secret



4. För att det ska vara skarpt så "togglar" du det i övre högra hörnet. Glöm inte att kontinuerligt spara ditt workflow.

Nu är du klar med autentiseringsprocessen och kan Börja bygga ditt orderflöde i Bosbec.

bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701



2.6. Bygga orderflödet I Bosbec

På samma sätt som autentiseringen finns det en template för detta flöde. I denna instruktion kommer vi beskriva i detalj hur det workflow byggts.

För att få en överblick kommer workflowet (autentisering och orderflöde) se ut enligt nedan när det är klart.



- Nedan beskrivs steg för steg hur du skapar ditt workflow. Det handlar om att få in orderdata från Shopify och sedan mappa denna information till Fortnox. För att kunna mappa data enklare kommer vi skapa Json resurser som vi sedan kan använda när vi gör en "Post" till Fortnox.
- 2. Första delen så ska vi få in data från Fortnox till Bosbec som vi sedan kan arbeta med. Vi behöver ha in orderdata, kundinformation och kunddata för denna process.



• Vi börjar med att ta ut ett job som heter "Parse json to resource". Vi skapar en resurs från http-anropet. Dra sedan en pil från job i autentiseringsprocessen "job_8" för att få anropet kopplat.

bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701



 Nu har vi skapat en json resurs och får data till Bosbec. För att se vilken data du får in kan du "debugga" och sedan se på din json resurs. Skapa en order i Shopifybutiken sedan går du till ditt workflow i Bosbec. Välj sedan "View" och sedan "Execution History". Där får du sedan dina körningar. Klicka på en körning för att komma till debugg mode. I debugg mode kan du sedan under "Available Resources" välja vilken av dina resurser du vill se på. I detta fall har vi skapat en json resurs och för att se datan där väljer du "JSON" och sedan "json_resource" och klickar på pennan. Denna funktion hjälper dig att löpande följa ditt workflow (du kan läsa mer om debugging här

https://help.bosbec.com/knowledge-base/troubleshooting-your-workflow/).



 I nästa job vill vi skapa en resurs för produkterna. Vi väljer igen ett "Parse json to resource" jobb och i json objektet line_items som vi vill ha som en separat resurs. Vi skriver in json_resource.line_items under "Json resource" och namnger resursen till "products". Dra sedan en pil från "job_10" till det nya jobbet.

bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701



• Eftersom line_items innehåller array så ska vi för nästa jobb skapa en tom array där vi sedan kan lägga in produktdetaljer som en resurs. I "Json source" lägger vi in en tom array "[]" och namnger den till "productarray".



• I nästa jobb vill se om kunden redan finns i Fortnox. Vi skapar en grupp som vi namnger till "Customers Fortnox". Vi sparar kund id i denna grupp och jämför om det i framtida ordrar finns i denna grupp.

Vi väljer ett "Unit operations" job och sedan väljer vi "Find in group" under "Group name" skriver vi "Customers Fortnox". Vid "Add filter operation" så väljer ni "Meta data filter" och fyller i villkoren. Vid "Left side" skriver ni "unit.metadata.customer_id" och som "Operator" skriver ni "=" och på "Right side" skriver ni "json_resource.customer.id". Vid "Store operation" så väljer du "Store in resource" och skriver i "Resource name" fortnox_customer. Vi väljer att inte spara detta på kontot utan enbart i körningen.

bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701



3. Nu ska vi börja skapa datastrukturen så den mappar mot Fortnox struktur. Vi går igenom denna del av workflowet som hanterar kunddatan.



• Vi väljer ett job som heter "Route from metadata". Detta används för att se om kunden finns i Fortnox eller om det ska skapas en ny kund i Fortnox. Detta ska sedan kopplas ihop med två ytterligare job ett för att skapa kund i Fortnox och ett job för att skapa metadata för customer_id.



bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701

b<mark>o</mark>sbec

Vi börjar med att skapa url för att lägga in kund i Fortnox och för det väljer vi job "Data operation" och sedan "Concat". I Shopify får man kundens namn som first_name och last_name. I Fortnox vill vi slå ihop detta till ett namn. Under "Source with format" skriver vi "{json_resource.customer.first_name} {json_resource.customer.last_name}" och sedan vid destination gör vi om det till metadata och skriver "metadata.name".

Vi fortsätter att addera till en ny operation och väljer igen "Concat". Nu vill vi göra anropet mot Fortnox kundregister till metadata. Vi skriver då in url till anropet samt json datan för kund ID,

"https://api.fortnox.se/3/customers/{json_resource.customer.id}", och gör om det till metadata, och skriver "metadata.customer_url". När vi sedan drar pil till detta job så ska vi konfigurera kopplingen och väljer då "Route destination on no match".



 Nu ska datastrukturen för kunddata skapas mellan Shopify och Fortnox. Vi väljer då job "Json pipeline" och step "transform. På vänstra sidan skriver vi den struktur Fortnox använder och på den högra struktur från Shopify. Det kommer se ut enligt nedan.

```
{
    "Customer": {
        "@url": "{{metadata.customer_url}}",
        "Active": true,
        "Address1": "{{json_resource.billing_address.address1}}",
        "Address2": "{{json_resource.billing_shipping.address2}}",
        "City": "{{json_resource.billing_address.city}}",
        "CustomerNumber": "{{json_resource.customer.id}}",
        "Email": "{{json_resource.customer.email}}",
        "Name": "{{metadata.name}}",
        "Type": "PRIVATE"
    }
}
```

Type i Fortnox anger om det är en privatperson eller företag. I vårt exempel har

bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701



vi valt att inte ta det från Shopify utan direkt i strukturen ange det som privatperson. Ni kan själva välja hur ni vill hantera detta.

Vi väljer att skapa ett nytt "Step" och väljer "Save as a resource" och vid "Resource body" skriver vi "fortnox_body". Detta sparas nu som en resurs för oss. Klicka också i rutan "Use syntax alt1".

Data operations julij, 6	Json pipeline - Job,16 {}	ð ×
Update takes info Update takes info Update takes info Update takes info Same take takes info	Transformation (University of the state of	0
	Alas	0
	Add Execution condition	•
	Select Execution condition	~
-	If Step - Save as resource	-
Just pipeline pib.76	Resource name fortinox_body	0
	Alas	0
	Add Execution condition	a ~
	Select Execution condition	~
	/ Lice syntax alt1	

 Nu ska vi göra anrop mot Fortnox. Vi väljer job "Send http request". Vi skriver url:en, https://api.fortnox.se/3/customers, som anropar Fortnox kundregister och väljer "Post" som "Method". I "Headers" så skriver vi content-type - application/json authorization - metadata.access_token I "Body" skriver vi "fortnox_body". Detta är resursnamnet som vi sparade i job_16.

Send http request -	Job_17	()	6
Url https://			
Method GET			
Headers			
Key	Value		
Body			
00.00:00:30			
Timeout dd.hh.mm.ss			
Response resource nam http_response	ne		
Response type Auto			
Response validation			

 Nu vill vi skapa kund från responsen och lagra kund ID i gruppen "Customers Fortnox". Vi väljer job "Unit operation" och sedan "Create from data". Under "Resource to unit map" så skriver vi vid "key", "metadata.customer_id" och vid value "json_resource.customer.id". Vid "Store operation" väljer ni "Store in group" och skriver "Customers Fortnox" och sedan klickar i rutan "Save to

bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701



account" för att spara det i gruppen.

När du sedan binder ihop job så får du upp en ruta med "Configure Connection" och där väljer du "Jobs to execute after successful response".



• Nu vill vi sätta Kund ID på kunden. Vi väljer då ett job "Data operation" och sedan väljer vi "Set data. I source så tar vi kund ID från Shopify, och i destination skriver vi "metadata.customer_id"

När vi sedan drar pil från "Route to metadata", job_14, så i "Configaration settings" så anges villkoren. Vi väljer "Metadata data route" och namnger routen till "Found in Fortnox" och för villkoren skriver vi vid "Compare operator" tecknet för större än, >, och vid "Compare value" skriver vi 0. Ett kund ID större än noll från Fortnox ska registreras.



4. Nu ska vi börja skapa datastrukturen så den mappar mot Fortnox struktur. Vi går igenom denna del av workflowet som hanterar produktdatan.

bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701



• Vi börjar med att välja ett job "For each resource start". Detta jobb loopar igenom ordern med produkterna så vi får med oss alla produkter från ordern. Vi itererar resursen "products" och namnger den till "product".



 Nu ska vi skapa array för produkt. Vi väljer job "Json pipeline" och sedan step "Transform". På vänstra sidan är struktur från Fortnox och högra är data från Shopify array.

```
{
    "DeliveredQuantity": "{{product.current_quantity}}",
    "Description": "{{product.name}}",
    "OrderedQuantity": "{{product.quantity}}",
    "Price": "{{product.price_set.shop_money.amount}}"
}
```

Vi sparar ner den som en resurs och lägger till ett step "Save as a resource" och namnger resursen till "tmp_product". Väljer ytterligare ett step "Push array item" där source ska vara "tmp_product" och vid "Destination path" skriver vi "productarray". När vi sedan drar pilen till detta job så konfigurerar vi den som "For each jobs".

bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701



• För att avsluta loopen väljer vi job "For each resource stop".



5. Nu vill vi ha leveransdata från array i Shopify data.



 Vi börjar med att välja ett job "For each resource start". Detta jobb loopar igenom pris på produkterna så vi får med oss alla priser från ordern. Vi itererar resursen "json_resource.shipping_lines" och namnger den till " shipping_line". När pil dras till detta job väljs i konfigurationen "Jobs after completed".



bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701

 Nästa job är "Data operation" och "Set data". I "Resource" väljer vi resursen "shipping_line.price" och skapar metadata som vi namnger till "metadata.shipping_price". I "Configuration settings" väljer vi "For each jobs". Vi avlutar loopen med job "For each resort stop".



6. Nu är det dags att bygga strukturen för orderanropet.



 Vi börjar med att välja ett "Data operation" job och en "Concat" operation. Vi anger url för anrop mot Fortnox order med Shopify ID för ordern, <u>https://api.fortnox.se/3/orders/{json_resource.id}</u>, sedan skapar vi metadata av detta, metadata.order_url. När vi sedan drar pil till detta job väljer vi konfiguration "Jobs after completed".

Configure Connection	×
Jobs after completed	~

• Nu vill vi skapa strukturen för ordern. Vi väljer job "Json pipeline" och step "Transformation". På vänstra sidan är struktur från Fortnox och högra är data från Shopify.

bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701

b<mark>o</mark>sbec

```
{
   "@url": "{{metadata.order_url}}",
   "CustomerNumber": "{{metadata.customer_id}}",
   "OrderRows": {{productarray}},
   "VATIncluded": "true",
   "DeliveryState": "delivery",
   "Freight": "{{metadata.shipping_price}}"
   }
}"
```

Vi anger VATIncluded eftersom vi anser att momsen är med i totalpriset. Vi valde tidigare att detta såldes till konsumenter och då är anges pris inkluderat med moms. Vi anger också i klartext att DeliveryState är delivery och det är för att det ska anses som att ordern är betald.

Nu väljer vi ett till step som är ""Save as resource" och väljer som "Resource name" - "jsonbodyfortnox_temp2". Klicka i rutan "Use syntax alt1.



 I nästa job skapar vi ordern som vi ska använda i anrop mot Fortnox. Vi väljer ett "Json pipeline" job och väljer step "Transform". Där tar vi resursen som vi tidigare skapat och skriver från Shopify.

```
{
    "Order":{{jsonbodyfortnox_temp2}}
}
```

Sedan sparar vi den som resurs som vi kallar "jsonbodyfortnox_final". Välj step "Save as a resource". Klicka i rutan "Use syntax alt1.

bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701

	Json pipeline - Job_28	{} ③	×
	# Step - Transform	§ 1	^
→¢	Transformation { "Order":{(jsonbodyfortnox_temp2}} }		0
Json pipeline job_28 Description	Alias		D
	Add Execution condition		^
	Select Execution condition	~	/
	# Step - Save as resource	6 8	^
	Resource name jsonbodyfortnox_final		Ð
	Alias		0
	Add Execution condition		^
	Select Execution condition		,
	✓ Use syntax alt1		
		ADD ST	EP

 Nu ska vi göra orderanropet till Fortnox och väljer job "Send http request". Vi väljer url <u>https://api.fortnox.se/3/orders</u> och ett POST anrop vid headers skriver vi som tidigare anrop till Fortnox och i bodyn skriver vi "jsonbodyfortnox_final".

	Send http request - Job_29	{} \$ \$	ð ×
	Url https://api.fortnox.se/3/orders		0
	Method POST		~
	Headers		
	content-type	application/json	
_	authorization	metadata.access_token	
	Кеу	Value	
	Body jsonbodyfortnox_final		0
	00.00:00:30		
	Timeout dd.hh:mm:ss		
	Response resource name http_response		0
	Response type Auto		~
	Response validation		

7. Nu ska vi få ordern som delivered och uppdatera lagret.



bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701



• Vi väljer job"parse json to resource" och i json source skriver vi "order_response.body.Order" och vid destination "shopify_order".

	Parse json to resource - Job_30	{} ③ ×
Json resource	Json source order_response.body.Order	0
shopify_order esource from job_30	Destination resource name shopify_order	0
· (*)	Deny lookup destination resource name	
irse json to resource job_30		
Description		

 Nu ska vi ta data från Shopify för att uppdatera lagret. I Shopify är det "DocumentNumber" där det finns. Vi väljer job "Data operations" och sedan "Set data" som operation. I "Source" skriver vi

"shopify_order.DocumentNumber" och i "Destination" så skapar vi metadata "metadata.documentnumber". Vi väljer en ny operation "Concat" där vi anger anrop till Fortnox,

https://api.fortnox.se/3/orders/{metadata.documentnumber}/warehouseready och sedan i "Destination" namnger det till "metadata.warehouseready".



 Nu ska vi uppdatera lagret och gör då ett PUT anrop till Fortnox. Vi har tidigare skapat metadata av <u>url:en</u> så vi lägger in "metadata.warehouseready" väljer "PUT" och fyller i "Headers som tidigare. Responsen skriver vi till "http_response_warehouseready".

bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701



8. Nu ska vi Fortnox skapa en faktura så att ordern bokas som betald.



 Vi börjar med att välja job "Data operation" och operation "Concat". I "Source with format" anger vi Fortnox url, <u>https://api.fortnox.se/3/orders/{metadata.documentnumber}/createinvoice</u> och gör metadata av url:en, metadata.createinvoiceurl.



 Nu ska vi skapa faktura och gör då ett PUT anrop till Fortnox. Vi har tidigare skapat metadata av <u>url:en</u> så vi lägger in " metadata.createinvoiceurl" väljer "PUT" och fyller i "Headers som tidigare. Responsen skriver vi till "http_response_invoice".

bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701



9. Nu ska vi uppdatera fakturan till cash invoice så ordern är betald när den kommer till Fortnox.



• Vi väljer job "Data operation" och operation "Concat". I "Source with format" anger vi Fortnox url,

https://api.fortnox.se/3/invoices/{http_response_invoice.body.Order.InvoiceRe ference} och gör metadata av <u>url:en</u>, metadata.invoicupdate.



 Från det tidigare anropet så plockar vi fakturareferens och bygger en url som gör att statusen på fakturan blir cash invoice. Vi gör då ett PUT anrop till Fortnox. Vi använder metadata <u>url:en</u> så vi lägger in "metadata.invoicupdate" väljer "PUT" och fyller i "Headers som tidigare. I "Body" skriver vi;

bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701

```
{"Invoice":{

"InvoiceType": "CASHINVOICE"

}}
```

Och "Response resource name" så namnger vi den till "http_response_iovice_paid".



10. Slutligen ska ordern bokföras i Fortnox.



 Vi väljer job "Data operation" och operation "Concat". I "Source with format" lägger vi in url för att anropa Fortnox, <u>https://api.fortnox.se/3/invoices/{http_response_invoice.body.Order.InvoiceRe</u> <u>ference}/bookkeep</u>. Sedan gör vi metadata av denna url och namnger den till "metadata.bookkeepinvoice" i "Destination".

bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701



 Nu är det bara sista jobbet kvar innan workflowet är klart och det är ett PUT anrop för att bokföra ordern. Vi väljer job "Send http request". Vi gör då ett PUT anrop till Fortnox. Vi använder metadata <u>url:en</u> så vi lägger in "metadata.bookkeepinvoice" väljer "PUT" och fyller i "Headers" som tidigare. I "Body" skriver vi; {

"Invoice": {"Booked": true}}

I "Response Resource Name" namnger vi det till "http_response_bookkeeping".



bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701



3. Slutord

Att bygga workflow är väldigt flexibelt i Bosbec. Detta ger en frihetsgrad att justera förändra när det ändras i systemen man integrerar eller i authentification som man har mot systemen. Till integrationen kan också andra system läggas till för ökad automatisering. T.ex. som i detta exempel så kanske man vill boka köp från ett kassasystem i Fortnox så det direkt finns i bokföringen då adderar man bara systemet och bygger vidare på sitt workflow.

bosbec

Bosbec AB Adress: Norrtullsgatan 6, 5 tr 113 29 Stockholm VAT No: SE556663223701