

CASE STUDIES

White Rabbit Express Order Fulfillment Case Study



White Rabbit Japan

Using Aspose.PDF .NET to streamline order fulfillment operations

Max Hodges, CEO, January 18th 2016

About White Rabbit Japan

We run White Rabbit Express, a Tokyo-based proxy buying service. We've helped tens of thousands of people in over 100 countries buy-and-ship Japanese products which are difficult to get outside of Japan.

Our order fulfilment operations were very inefficient and manual labor intensive. This was especially painful during Black Friday and Cyber Monday events which increased orders by several times our normal volume resulting in delays and an overworked staff.

Taking cues from Toyota's process management principles, we were able to eliminate huge amounts of waste by allocating roles differently, creating custom software to manage the workflows, integrating barcode label printers and scanners, and reducing wasteful motions by replacing keyboards and mice with touch-screen monitors throughout the department.

Problem

Aspose.PDF .NET came to our aid in two key areas: programmatic printing of PDF documents and automatically adding signatures to those documents.

Part of our shipping process involves submitting data to a Japan Post API. The API returns several shipping documents--dispatch note, customs declaration, and commercial invoice--in PDF format. Printing these documents from our web application is time-consuming for users, who need to open the PDF in a new window, open the print dialog, click print, then close the tab. Since this activity is highly mechanical, it was clearly something we wanted to reduce.

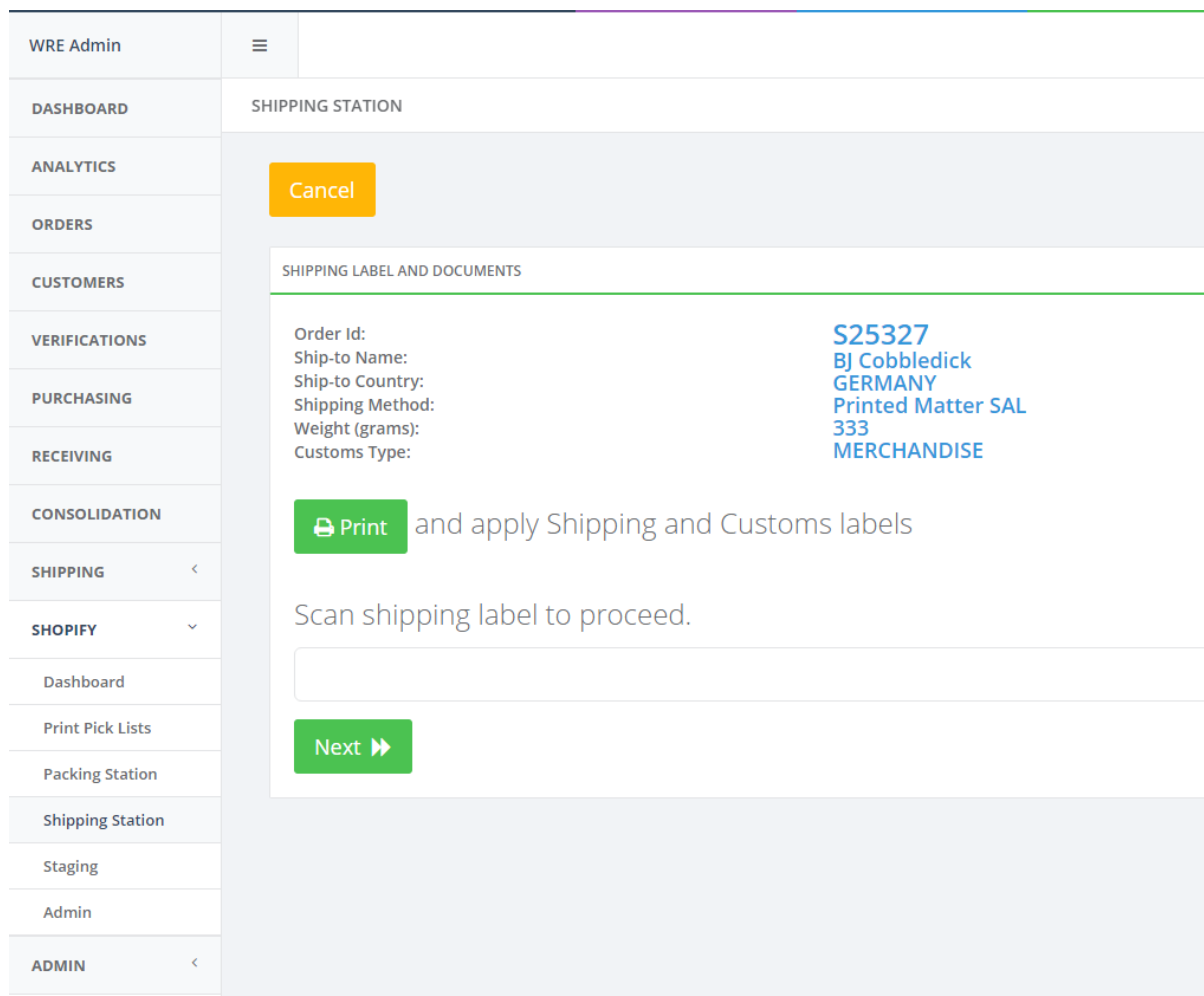
Once these documents are printed, our shippers need to sign as many as seven pages per shipment using a ballpoint pen. We decided to see what could be done to get signatures digitally inserted into the shipping documents.

Solution

We built a .NET application using C# to handle the printing of PDFs using Aspose.PDF .NET. Our application runs as a web server on the local machine, listening on a localhost port. PDF

documents are posted from our Meteor web application to the localhost port where the .NET application receives them and handles the printing. There are no print dialogs, no extra buttons to click, and no browser tabs for the user to close.

Next we took a photo of a signature and saved it as a PNG image file with a transparent background. Using Aspose.PDF .NET [search tools](#) we programmatically iterate through each page of the PDF documents looking for keywords which indicate a signature is required field. The signature [image file is then inserted into the PDF](#) before we resave and print it.



WRE Admin

DASHBOARD

ANALYTICS

ORDERS

CUSTOMERS

VERIFICATIONS

PURCHASING

RECEIVING

CONSOLIDATION

SHIPPING

SHOPIFY

Dashboard

Print Pick Lists

Packing Station

Shipping Station

Staging

Admin

ADMIN

SHIPPING STATION

Cancel

SHIPPING LABEL AND DOCUMENTS



Order Id: **S25327**
 Ship-to Name: **BJ Cobbledick**
 Ship-to Country: **GERMANY**
 Shipping Method: **Printed Matter SAL**
 Weight (grams): **333**
 Customs Type: **MERCHANDISE**

Print and apply Shipping and Customs labels

Scan shipping label to proceed.

Next

Figure 1: Our in-house order fulfilment web application. Shipping document files in PDF format are passed to a .NET app running Aspose.PDF components.

 国際スピード郵便																											
送り状 (Dispatch Note)		物品用 (For goods)																									
JAPAN		JAPAN POST																									
郵便番号 154-0001 JAPAN		お問い合わせ番号 (item number) EL 034 206																									
From (Sender) Name & Address WHITE RABBIT JAPAN 3-21-28 IKEJIRI 9TH FLOOR SETAGAYA-KU TOKYO		受付年月日 Date mailed 年 (Year) 月 (Month) 日 (Date) 2016 01 13																									
受付時刻 Time mailed 時 (hour) 分 (Minute)		郵便料金 28 合計金																									
総重量 Total gross weight 1035 g		To (Addressee) Name & Address [Redacted] SOUTH AUSTRALIA Country AUSTRALIA																									
内容品の詳細な記載 Detailed description of contents		日本円換算合計 Total Value																									
<table border="1"> <thead> <tr> <th>内容品名</th> <th>数量</th> <th>単価</th> <th>小計</th> </tr> </thead> <tbody> <tr> <td>BOOK</td> <td>1</td> <td>USD12.00</td> <td>USD12.00</td> </tr> <tr> <td>STATIONARY</td> <td>2</td> <td>USD6.00</td> <td>USD12.00</td> </tr> <tr> <td>STATIONARY</td> <td>1</td> <td>USD5.00</td> <td>USD5.00</td> </tr> <tr> <td>BOOK</td> <td>1</td> <td>USD11.00</td> <td>USD11.00</td> </tr> <tr> <td>BOOK</td> <td>1</td> <td>USD17.65</td> <td>USD17.65</td> </tr> </tbody> </table>		内容品名	数量	単価	小計	BOOK	1	USD12.00	USD12.00	STATIONARY	2	USD6.00	USD12.00	STATIONARY	1	USD5.00	USD5.00	BOOK	1	USD11.00	USD11.00	BOOK	1	USD17.65	USD17.65	TEL FAX 次の場合に (x) <input type="checkbox"/> 贈物 a gift <input checked="" type="checkbox"/> 商品 merchan <input type="checkbox"/> 身の回り personal	
内容品名	数量	単価	小計																								
BOOK	1	USD12.00	USD12.00																								
STATIONARY	2	USD6.00	USD12.00																								
STATIONARY	1	USD5.00	USD5.00																								
BOOK	1	USD11.00	USD11.00																								
BOOK	1	USD17.65	USD17.65																								
FOR DETAILS, SEE ATTACHED DOCUMENT		日本円換算合計 Total Value																									
<input checked="" type="checkbox"/> 上記内容品は危険物に該当しないことを確認済です I checked that contents above are not dangerous goods.		<input type="checkbox"/> 20万円超 申告対象郵便物																									
ご署名 Signature of the sender Max Hodges		この郵便物は Number of this pieces 番目 Total number of pieces																									
(To Post and Customs) This is EMS Dispatch Note of delivery attached on		ご注意! この用紙は送り状 専用バウチに入れ																									

Fold in half and enclose in the pouch

Figure 2: Signatures like these are inserted on shipping documents automatically before being sent to the printer.

Experience

Finding a solution: We evaluated several solutions before selecting Aspose.PDF .NET. Acrobat Reader lacked an API and was hard to remote control. GhostScript was very low-level, so required a lot of work to do simple things. PDFSharp was basically a wrapper for Adobe Acrobat.

Implementation: Using a trial version of Aspose.PDF .NET I was able to start printing PDFs programmatically in minutes, and has a working solution deployed within half-a-day. The signature image merge feature was implemented in about 3 hours of work. The code examples in the online programmer's guide at aspose.com were very valuable in helping us quickly build our solutions.

Outcome: We believe making many small improves to a process can result in large improvements. Speeding up a process by just one minute when shipping 10-20K packages results in several hundred hours of time saved. Aspose.PDF .NET allowed us to very rapidly deliver an automated solution, freeing our team from the manual grind of repetitive tasks.