

# P3 - 360º Company Dashboard

INTEROPERABILITY



Master in Informatics and Computing Engineering

Information Systems

**Group H:**

Daniela Quintas Fernandes de Sá - up201405457

Hélder Manuel Mouro Antunes - up201406163

Inês Filipa Noronha Meneses Gomes Proença - up201404228

João Filipe Pereira da Costa - up201403967

Faculdade de Engenharia da Universidade do Porto  
Rua Roberto Frias, sn, 4200-465 Porto, Portugal

November 21, 2017

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Information Architecture</b>	<b>4</b>
2.1	Overview (V_01) . . . . .	4
2.1.1	Features implemented: . . . . .	4
2.1.2	Primavera Webservices used: . . . . .	4
2.1.3	Webservices of our server used: . . . . .	4
2.1.4	To be implemented: . . . . .	4
2.2	Sales Dashboard (V_02) . . . . .	5
2.2.1	Features implemented: . . . . .	5
2.2.2	Webservices of our server used: . . . . .	5
2.2.3	To be implemented: . . . . .	5
2.3	Purchases Dashboard (V_03) . . . . .	6
2.3.1	Features implemented: . . . . .	6
2.3.2	Primavera Webservices used: . . . . .	6
2.3.3	Primavera Webservices used in drilldown functionalities: .	6
2.3.4	To be implemented: . . . . .	6
2.4	Inventory Dashboard (V_04) . . . . .	7
2.4.1	Features implemented: . . . . .	7
2.4.2	Primavera Webservices used: . . . . .	7
2.4.3	To be implemented: . . . . .	7
2.5	Finances Dashboard (V_05) . . . . .	8
2.5.1	To be implemented: . . . . .	8
<b>3</b>	<b>Interoperability with Primavera</b>	<b>9</b>
3.1	Interoperability with own server and data obtained through the SAF-T . . . . .	13

# 1 Introduction

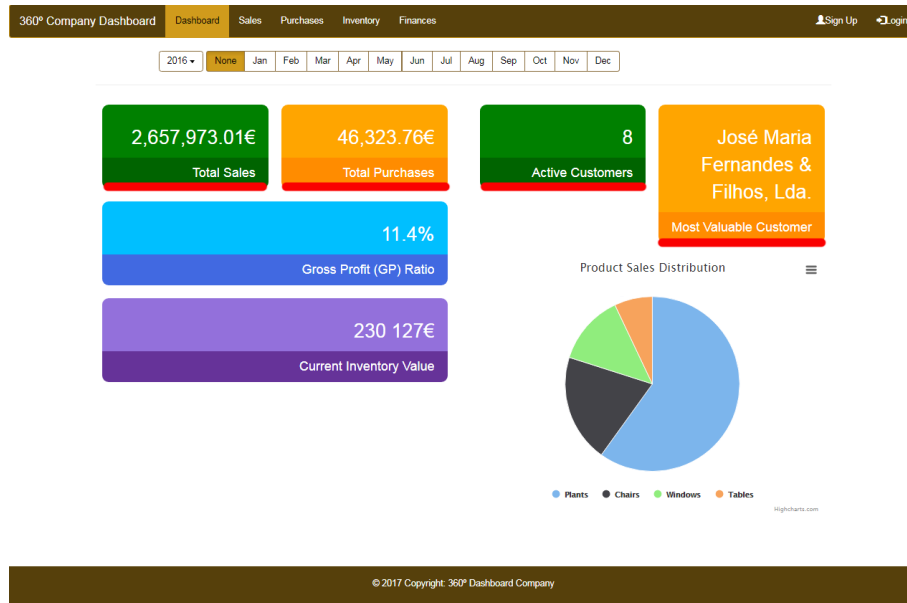
The purpose of this report is to show the work already developed in the core views related to the interoperability with the Primavera ERP via web-services and the SAFT structure.

The core views proposed in the previous report were Overview (V\_01), Sales (V\_02), Purchases (V\_03), Inventory (V\_04) and Finances (V\_05). Amongst them, we decided to focus on the completion of the Sales and Purchases Dashboards to be able to show a more complete integration with both the web-services and SAFT.

In the following chapters, we will explain in detail the work already done in each core view.

## 2 Information Architecture

### 2.1 Overview (V\_01)



#### 2.1.1 Features implemented:

- Total Sales
- Total Purchases
- Active Customers
- Most Valuable Customer

#### 2.1.2 Primavera Webservices used:

- colocar aqui as cenas do primavera

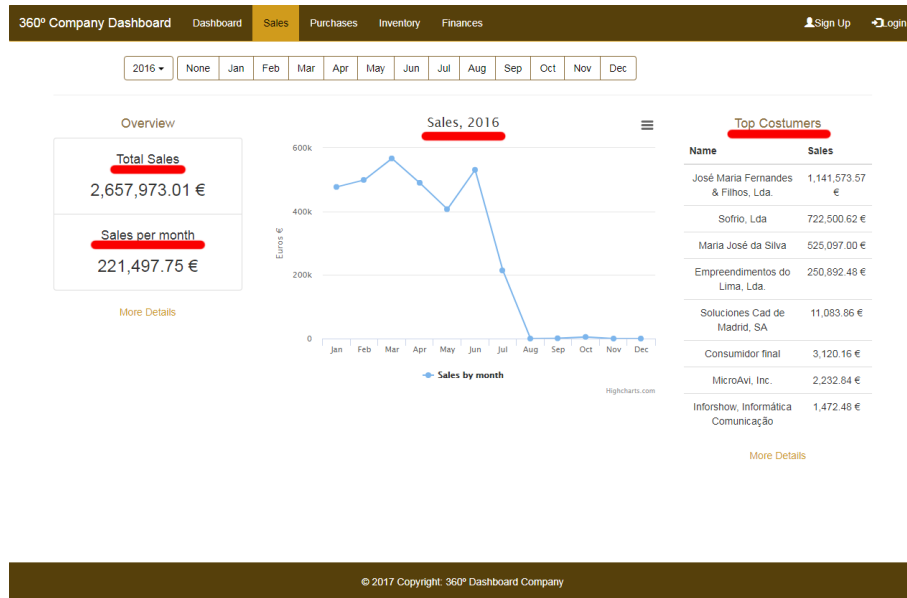
#### 2.1.3 Webservices of our server used:

- /getSalesByYear
- /getSalesByMonth
- /getCustomers

#### 2.1.4 To be implemented:

- Current Inventory Value
- Gross Profit (GP) Ratio
- Product Sales Distribution

## 2.2 Sales Dashboard (V\_02)



### 2.2.1 Features implemented:

- Total Sales
- Sales per month/day
- Sales evolution chart
- Top customers

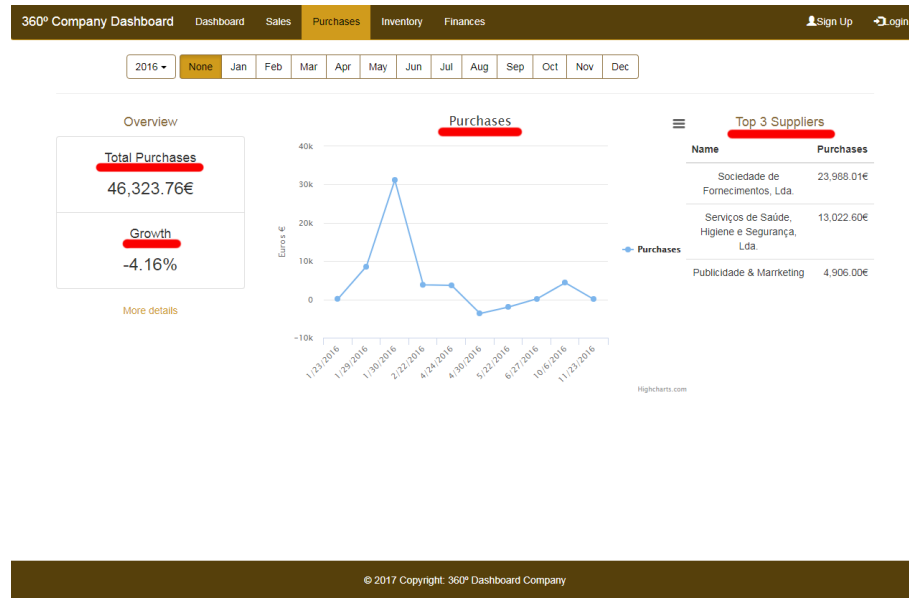
### 2.2.2 Webservices of our server used:

- /getSalesByYear
- /getSalesByMonth
- /getCustomers

### 2.2.3 To be implemented:

- Orders
- Customer debts

## 2.3 Purchases Dashboard (V\_03)



### 2.3.1 Features implemented:

- Show total amount spent in purchases
- Purchases in the selected time period
- Display growth of the amount spent versus the same period the year before
- Show top 3 Suppliers

### 2.3.2 Primavera Webservices used:

- getPurchasesGroupedByDate
- getPurchasesGroupedBySupplier

### 2.3.3 Primavera Webservices used in drilldown functionalities:

- getPurchases
- getPurchasesFromSupplier

### 2.3.4 To be implemented:

- Drilldown purchases by a specific supplier

## 2.4 Inventory Dashboard (V\_04)



### 2.4.1 Features implemented:

- Display inventory value in time frame specified
- Show the distribution of inventory value by category/family

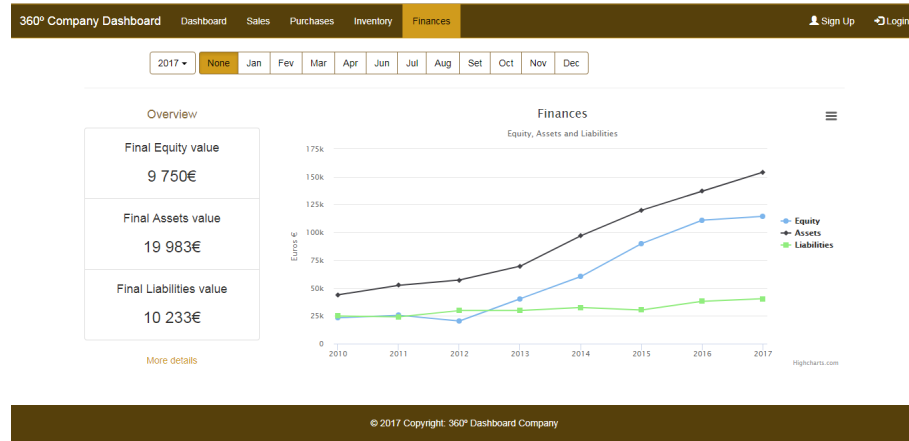
### 2.4.2 Primavera Webservices used:

- getInventory

### 2.4.3 To be implemented:

- Warn user of products with most urgency to be bought (stock below minimum or near)
- Monitor stock of best seller products because this are the most critical not to run out of

## 2.5 Finances Dashboard (V\_05)



No feature has been fully implemented. We have already been able to fetch information from the SAF-T file about accounts and transactions, and store it database.

### 2.5.1 To be implemented:

- Balance Sheet
- Equity, Assets and Liabilities chart



### 3 Interoperability with Primavera

Webservice ID	getPurchases
Webservice Description	Fetches a list with all purchases in specified year and month
Related Core View(s)	Purchases dashboard
Route+Verbs	host/api/Purchases?year=Y&month=M
Method	GET
Input Example	host/api/Purchases?year=2016&month=5
Expected Output	<pre>{   [     {       "Id": "{96F11A5A-E8B3-11DD-A8B6-000C2989A062}",       "DocumentDate": "2016-05-22T00:00:00",       "PaymentDate": "2016-06-21T00:00:00",       "Entity": "F0001",       "EntityName": "Sociedade_de_Fornecimentos,_Lda.",       "DocumentNumber": 14,       "Notes": "",       "TotalValue": 3640,       "DocumentType": "VFA",       "DocumentSeries": "A",       "Items": [         {           "Id": "{7DD7B8E9-E8B3-11DD-A8B6-000C2989A062}",           "Product": "A0001",           "Description": "Computador_Celeron_800MHz",           "Quantity": -8,           "UnitPrice": 455,           "Value": 3640         }       ]     }   ],   ... }</pre>

Webservice ID	getPurchasesGroupedByDate
Webservice Description	Fetches a list with the total spent in purchases per day in specified year and month
Related Core View(s)	Purchases dashboard
Route+Verbs	host/api/Purchases/groupByDate?year=Y&month=M
Method	GET
Input Example	host/api/Purchases/groupByDate?year=2015&month=06
Expected Output	<pre>{[   {     "DocumentDate": "2015-06-01T00:00:00",     "TotalValue": 981,   },   ...]}</pre>

Webservice ID	getPurchasesGroupedBySupplier
Webservice Description	Fetches a list with the total spent in purchases per supplier in specified year and month
Related Core View(s)	Purchases dashboard
Route+Verbs	host/api/Purchases/groupBySupplier?year=Y&month=M
Method	GET
Input Example	host/api/Purchases/groupBySupplier?year=2015&month=06
Expected Output	<pre>{[   {     "Entity": "F0003",     "EntityName": "Companhia_de_Hardware",     "TotalValue": 10203.8   },   ...]}</pre>

Webservice ID	getPurchasesFromSupplier
Webservice Description	Fetches a list with all purchases from a supplier in time interval specified
Related Core View(s)	Purchases dashboard
Route+Verbs	host/api/Purchases?initial=X&final=Y&supplier=Z
Method	GET
Input Example	host/api/Purchases?initial=2015-01-01&final=2016-12-30&supplier=F0001
Expected Output	<pre>{   "purchases" : [     {       "Id": "{B8D74796-8F02-431C-A174-CF737D863497}" ,       "DocumentDate": "2015-06-06T00:00:00" ,       "PaymentDate": "2015-07-06T00:00:00" ,       "Entity": "F0003" ,       "EntityName": "Companhia de Hardware" ,       "DocumentNumber": 5 ,       "Notes": "" ,       "TotalValue": -10203.8 ,       "DocumentType": "VFA" ,       "DocumentSeries": "A" ,       "Items": [         {           "Id": "{6BF51D9E-05A4-47F1-AD87-ECE0E70F2D98}" ,           "Product": "A0003" ,           "Description": "Processador INTEL CORE 2 DUO E6300" ,           "Quantity": -3 ,           "UnitPrice": -1198 ,           "Value": -2875.2         } ,         ... ]       }     } ,     ... ]   } }</pre>

Webservice ID	getInventory
Webservice Description	Fetches a list with all products and theirs respective quantity at a given date(first day of the year/month)
Related Core View(s)	Inventory dashboard
Route+Verbs	host/api/Inventory/date/year/month
Method	GET
Input Example	host/api/Inventory/date/2016/2
Expected Output	<pre>{   "stocks" : [     {       "Article": "A0001",       "CurrentStock": 3,       "Description": "Water_Bottle_25oz",       "Family": "Drinks",       "ReserveStock": "0",       "SubFamily": "H01.013",       "TotalValue": 5,       "UnitPrice": "",       "Warehouse": "A"     },     ... ] }</pre>

### 3.1 Interoperability with own server and data obtained through the SAF-T

Webservice ID	getSalesByMonth
Webservice Description	Fetches a list with all sales invoices in specified month of an year
Related Core View(s)	Sales dashboard, Overview dashboard
Route+Verbs	host/getSalesByMonth?month=X&year=Y
Method	GET
Input Example	host/getSalesByMonth/?month=1&year=2016
Expected Output	<pre>[   {     "InvoiceNo": "FA_2016/1",     "InvoiceStatus": "N",     "InvoiceDate": "2016-01-01T00:00:00.000Z",     "InvoiceType": "FT",     "CustomerID": "PT123456789_C",     "TaxPayable": 54.51,     "NetTotal": 272.54,     "GrossTotal": 327.05,     "Lines": [       {         "lineNumber": "1",         "productCode": "B0007",         "productDescription": "Rato_R17B",         "quantity": "5",         "unitOfMeasure": "UN",         "unitPrice": 0,         "creditAmount": 0,         "debitAmount": null,         "taxType": "IVA",         "taxPercentage": 20,       },       ...     ]   },   ... ]</pre>

Webservice ID	getSalesByYear
Webservice Description	Fetches a list with all sales invoices in specified year
Related Core View(s)	Sales dashboard, Overview dashboard
Route+Verbs	host/getSalesByMonth?year=Y
Method	GET
Input Example	host/getSalesByMonth/?year=2016
Expected Output	<pre>[   {     "InvoiceNo": "FA_2016/1",     "InvoiceStatus": "N",     "InvoiceDate": "2016-01-01T00:00:00.000Z",     "InvoiceType": "FT",     "CustomerID": "PT123456789-C",     "TaxPayable": 54.51,     "NetTotal": 272.54,     "GrossTotal": 327.05,     "Lines": [       {         "lineNumber": "1",         "productCode": "B0007",         "productDescription": "Rato_R17B",         "quantity": "5",         "unitOfMeasure": "UN",         "unitPrice": 0,         "creditAmount": 0,         "debitAmount": null,         "taxType": "IVA",         "taxPercentage": 20,       },       ...     ]   },   ... ]</pre>

Webservice ID	getCustomers
Webservice Description	Fetches a list with all customers in specified year or month
Related Core View(s)	Sales dashboard, Overview dashboard
Route+Verbs	host/getCustomers?year=Y
Method	GET
Input Example	host/getCustomers?year=2016
Expected Output	<pre>[     {       "customer_id": "PT123456789-C",       "account_id": "21111001",       "customer_tax_id": "123456789",       "company_name": "Sofrio , Lda",       "telephone": "200267890",       "fax": "200267899",       "sales": 85411.18     },     ... ]</pre>