**A**

**Project Report**

**on**

**E-Commerce Integration with ERP System**

**Developed By**

***Viraj Varmora [2305112130025]***

**developed at**

**eGrivory Technology Pvt. Ltd. – Rajkot**

**as**

**Partial Fulfillment of IVth Semester of**

**Master of Computer Applications**

**for A.Y. 2024 - 2025**

**Under The Guidance of**

**Prof. Sumit Kumar Soni**

**Submitted To**

**Department of MCA**

**Faculty of IT & Computer Science**

**PARUL University**





**CERTIFICATE**

This is to certify that **Mr. Viraj Varmora, Enrollment No. 2305112130025** student of Master of Computer Applications has satisfactorily completed the Major Project on **“E-Commerce Integration with ERP System”** at **eGrivory Technology Pvt. Ltd.** as fulfillment of MCA Semester IV.

Seat No. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date of Submission: \_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Internal Guide Project Coordinator Director - MCA

**Department of MCA**

**Faculty of IT & Computer Science**

**PARUL University, Vadodara**

**COMPANY PROFILE**

eGrivory was formulated with the objective of providing a complete solution to all your Odoo needs in a cost-effective manner. Skilled in its implementation, customization, module development services and more, eGrivory provides the appropriate answer to all your business needs.

Our company was established keeping the needs of the growing market in mind. Hence every day our skilled team works towards creating an environment our company can burgeon in. having years of experience in the field, our team understands your business needs and shapes the perfect solution to cater to your individual needs. The founders of eGrivory have been avid practitioners of the technology and now follow their vision by providing services which create the means to excel.

Our experts specifically customize offerings to cater to your requirements. Based on the industry you are operating in or individual needs, customization lets you target a larger yet better segmented audience. With business requirements growing and changing with the market, customization becomes an integral step for strengthening your hold on the market.

|  |  |
| --- | --- |
| **Company Name** | eGrivory Technology Pvt. Ltd- Rajkot |
| **Working Technology** | Odoo/Python |
| **Company Address** | F01, Maruti Complex, Near Atmiya Hostel, Munjka, Rajkot |
| **Website** | http://www.egrivory.com |

**PREFACE**

The word “E-Commerce Integration With ERP System **eCommerce integration** enables business managers to improve profits by analyzing sales and inventory levels. Managers can maintain inventory and sales via automatic stock adjustments and reporting. In addition, dedicated customer portals will enable business managers to keep customer data organized.

The word “Project” itself has a lot deep and realistic sense in its own right. Truly speaking for a fresher project work is the first milestone not for achievement but for sustaining it’s realism for all his/her future endeavor.

The objective of this project work is to familiarize us with the weaving water of realistic and make us little but more practical. It helps us to know what role we are going to play in the real world.

This project during study is really precious one. It provides opportunity to meet and work with different Tools with very good guidance of our professor. It helps to know their views and gain something from their attitudes and performances.

It was a privilege for us to work in reputed organization like **Department Of MCA PARUL UNIVERSITY** under such a good guidance. This has given us an opportunity to work in truly professional environment where teamwork scores over individual efforts.

**ACKNOWLEDGMENT**

I would like to express my deepest gratitude to my external guide **Mr. Vivek Paghadal (eGrivory Technology Pvt. Ltd.)** and my Internal Faculty Guide **Prof. Sumit Kumar Soni (M.C.A.), Department Of MCA PARUL UNIVERSITY** for his valuable guidance and continual encouragement throughout the major project. I am heartily thankful to him for his time to time suggestions and the clarity of the concepts of the topic that helped me a lot during this project.

I am grateful to the head of our Department **Prof. Vivek Dave (M.C.A.), Department Of MCA PARUL UNIVERSITY** for his continual kind words of encouragement and motivation throughout the Major Project.

**INDEX**

|  |  |  |
| --- | --- | --- |
| **No.** | **Description** | **Page No.** |
| **1.** | **Introduction**   * 1. Odoo to Shopify   2. Shopify to Odoo   3. Odoo   4. Shopify Connector   5. Python | **1 - 3** |
| **2.** | **Requirement Determination**  2.1 Requirement Determination  2.2 Project Module  2.3 Hardware and Software Requirement  2.4 Feasibility Study | **4 - 8** |
| **3.** | **System Design**  3.1 Data Flow Diagram ( 0 Level)  3.2 Data Flow Diagram ( 1 Level)  3.3 E-R Diagram  3.4 Data Dictionary  3.5 Screenshots | **9 - 38** |
| **4.** | **Development Standards**  4.1 Coding Standards | **39** |
| **5.** | **Agile Documentation** | **40 – 41** |
| **6.** | **Testing**  6.1 System Testing  6.2 Testing Methodology | **42 - 45** |
| **7.** | **Future Enhancement** | **46** |
| **8.** | **Bibliography** | **47** |

**Introduction**

* 1. **Odoo to Shopify - Export Operations**
* Export and update products and product details
* Export and update product stock, price and images
* Export and Update Custom & Smart Collections
* Publish or Unpublish products to Shopify store from Odoo
* Update Order Status of Completed and Cancelled orders
* Manage Refunds

**1.2 Shopify to Odoo - Import Operations**

* Import or Synchronize products
* Import Product Stock, price and image
* Import Unfulfilled Orders
* Import Collections
* Import Customer Details
* Import, Process and Reconcile the Shopify Payouts transactions.

**1.3 Odoo**

Odoo is a suite of business management software tools including CRM, e-commerce, billing, accounting, manufacturing, warehouse, project management, and inventory management to name a few.

Odoo extensible architecture allows a large number of freelancers and organizations to develop Odoo Apps or Modules and place them in the 1 marketplace for sale or to be downloaded for free.

**What is Odoo used for ??**

ODOO, formerly known as OpenERP (Enterprise Resource Planning), is a platform that companies can use to easily manage the basics of the company such as materials and warehouse management, human resources, finance, accounting, sales and many other enterprise features.

This is being done by means of an intuitive user interface that can be easily extended with community modules or with customized modules that suit the client's purposes. Helping you make smart decisions every day, ODOO can be used by companies of all sizes, offering a clear and integrated view of your business.

**Benefits Of Odoo**

* Multiple Modules Integration. As discussed above, Odoo consists of more than 35 modules that are the essential components of a business.
* Customer Support. ...
* Customizable Architecture. ...
* Comprehensive in Nature. ...
* Pricing. ...
* Limited Support. ...
* Maintenance Issues. ...
* Pricing Plans.

**1.4 Shopify Connector**

* Shopify is a powerful ecommerce website solution that allows you to sell online by providing everything you need to create an online store.
* Shopify integrations consist of two important components, Triggers and Actions. Triggers are used when Shopify is your “Source” application and you want to sync-up data from Shopify to your target system.
* Actions are used when Shopify is your “Target” application and you want to sync-up data from another business system into this application.

**How does Shopify Connector work ?**

* Adeptia Connect allows you to connect and integrate Shopify data with any business application. Our platform allows you to:
* Choose from a library of shared connections or create your own connection through a wizard and get started in minutes.
* Sync Shopify data such as Orders, Customers, Products, custom objects to and from your Databases, ERP, CRM, and Marketing platforms.
* Graphically map Shopify data to any other data format in minutes.
* Automate data exchange between Shopify and NetSuite.
* Sync Shopify data with third-party APIs through Webhooks.
* Sync Shopify data with Xero, Salesforce, QuickBooks, Sage One or FreshBooks.
* Sync Shopify data with on-premise applications or databases via Secure Agent.

**1.5 Python**

* Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.
* It is simple and easy to learn and provides lots of high-level data structures.
* Python makes the development and debugging *fast* because there is no compilation step included in Python development, and edit-test-debug cycle is very fast.

**Benefits Of Python**

* Easy to Read, Learn and Write. Python is a high-level programming language that has English-like syntax. ...
* Improved Productivity. Python is a very productive language. ...
* Interpreted Language. ...
* Dynamically Typed. ...
* Free and Open-Source. ...
* Vast Libraries Support. ...
* Portability**.**

1. **Requirement Determinations**

**2.1 Requirement Determination**

Requirement for the eCommerce ERP System are defined as below:

**Administrative**

* These features help your e-store to work seamlessly and easily.
* In this section Shopify offers the least admin feature compared to Odoo.
* Shopify does not have Bulk order processing and bulk product import/export modules.
* Odoo supports the API and User permission and role.

**Promotion and Marketing**

* In this aspect, both the platforms come with a decent list of modules for the promotion and marketing of the website.
* Shopify, as a SaaS platform, does not feature many modules like product comparison, product building, etc.
* On the other hand, Odoo is just amazing in this part. But here you need to customize most of the features to implement it to make it work**.**

**Inventory Management**

* When it comes to inventory management, again Shopify fails in many aspects.
* Inventory management is an essential part of any business, especially for those who rely heavily on storage and inventory.
* Here also another platform brings better features.

**Reports and analysis**

* It is of the essence for an e-commerce store to generate correct reports for future use in business growth.
* Odoo develops accurate sales reports coupled with inventory, sales, and product performance summary reports.
* It includes inventory reports, invoicing, A/B testing reports, sales summary reports, and many reports as per your business requirements.
* Both platforms also provide the feature of traffic analytics.

**2.2 Project Module**

Following project module we will design base of the client requirements understand.

**Sales**

* Typical sales process includes processes like Sales queries & inquiry analysis & handling, quotation drafting, accepting sales orders, drafting sales invoices with proper taxation, dispatch/Shipment of material or service, tracking pending sales orders.
* All these sales transactions are managed by the sales module of ERP.
* CRM module can take the help of the Sales module for future opportunity creation & lead generation.

**Purchase**

* As the name indicates, purchase modules take care of all the processes that are part of the procurement of items or raw materials that are required for the organization.
* The purchase module consists of functionalities like supplier/vendor listing, supplier & item linking, sending quotation requests to vendors, receiving & recording quotations, analysis of quotations, preparing purchase orders, tracking the purchase items, preparing GRNs(Good Receipt Notes) & updating stocks & various reports.
* Purchase module is integrated with Inventory module &  Engineering/production module for updating of stocks.

**Inventory**

* An inventory module can be used to track the stock of items. Items can be identified by unique serial numbers. Using that unique numbers inventory system can keep track of items and trace their current location in the organization.
* e.g. you have purchased 100 hard disks, so using the inventory system you can track how many hard disks are installed, where they are installed, how many hard disks are remaining, etc.
* The inventory module includes functionalities like inventory control, master units, stock utilization reporting, etc.

**Finance & Accounting**

* The whole inflow & outflow of money/capital is managed by the finance module.
* This module keeps track of all account-related transactions like expenditures, Balance sheets, account ledgers, budgeting, bank statements, payment receipts, tax management, etc.
* Financial reporting is an easy task for this module of ERP.
* Any Financial data that is required for running the business is available with one click in Finance module.

**Hardware and Software Requirement**

**Software Requirement**

|  |  |
| --- | --- |
| **Software Use** | **Remarks** |
| **Odoo** | Write to a programming logic, application design |
| **Shopify** | Shopify is an easy to use online store |
| **Web Browser** | Web application work with web browser |

**Hardware Requirement**

|  |  |  |
| --- | --- | --- |
| **Category** | **Client side** | **Server Side** |
| **Processor** | Intel Core | Intel i3 |
| **RAM** | 512 GB | 4GB |
| **Hard Drive** | 1GB | 500 GB |
| **Internet Connection** | Yes | Yes |
| **Operating System** | Web Browser Supported | Ubuntu |

**2.4 Feasibility Study**

Feasibility study is a process to check possibilities of system development. It is a method to check various different requirements and availability of financial & technical resources.

Before starting the process various parameters must be checked like:

* Estimated finance is there or not?
* The man power to operate the system is there or not?
* The man power is trained or not?

All the above conditions must be satisfied to start the project. This is why in depth analysis of feasibility is carried out.

There are three different ways feasibility can be tested:

1) Economical Feasibility

2) Technical Feasibility

3) Operational Feasibility.

* **Economic Feasibility:**

For declaring that the system is economically feasible, system will be cost effective and budgetary constraints, it should be cheap and quick. There isn’t any extra requirement of peripheral or software for development of system as it can be completed with the available resource.

* This system is a product developed for a government agency to reduce the manual data processing.
* Overall objective of this reduce human error, crime identification, reduce the officer time, working process and get some amount.
* **Technical Feasibility:**

Technical feasibility refers to the ability of the process to take advantages of the current state of the technology in pursuing further improvement. The front end tool proposed is easily compatible with the current hardware configuration in the organization. The back end tool proposed has the capacity to hold the data required for using the new system.

* + Our system requires apache web server to run the web application. There is no need to purchase any license.
  + This website is running on any Windows or Linux based operating system which can be installed easily.
  + No need to install any other hardware or software so its technically feasible**.**
* **Operational Feasibility:**

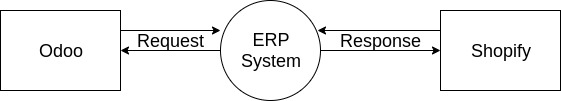
By feasibility study we mean study of the current operational systems and brief consideration of alternative methods computerizing these tasks. The purpose of the feasibility study is to investigate the present system, evaluate the possible application of computer-based methods, select a tentative system, evaluate the cost and effectiveness of the proposed system, evaluate the impact of the proposed system on existing personal and ascertain the need for new personnel.

* + This site is developed entirely using handlebar,ExpressJS and NodeJS which is quite easy to learn and develop any web application.
  + For development purpose trainees and employees are required but its cost is reliable and affordable.
  + This site contains design -dependent parameters such as reliability, maintainability, supportability, usability, affordability and other important payment assurance which is secure.

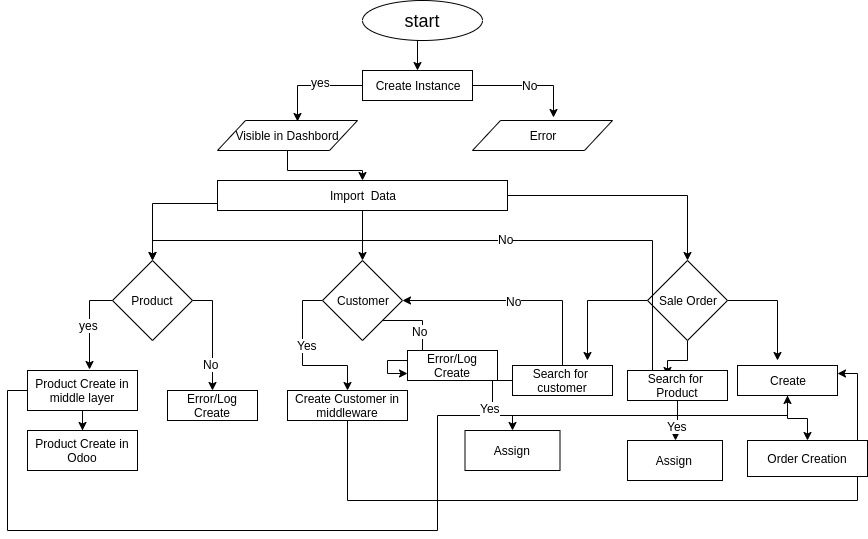
1. **System Design**

**3.1 Data-Flow Diagram**

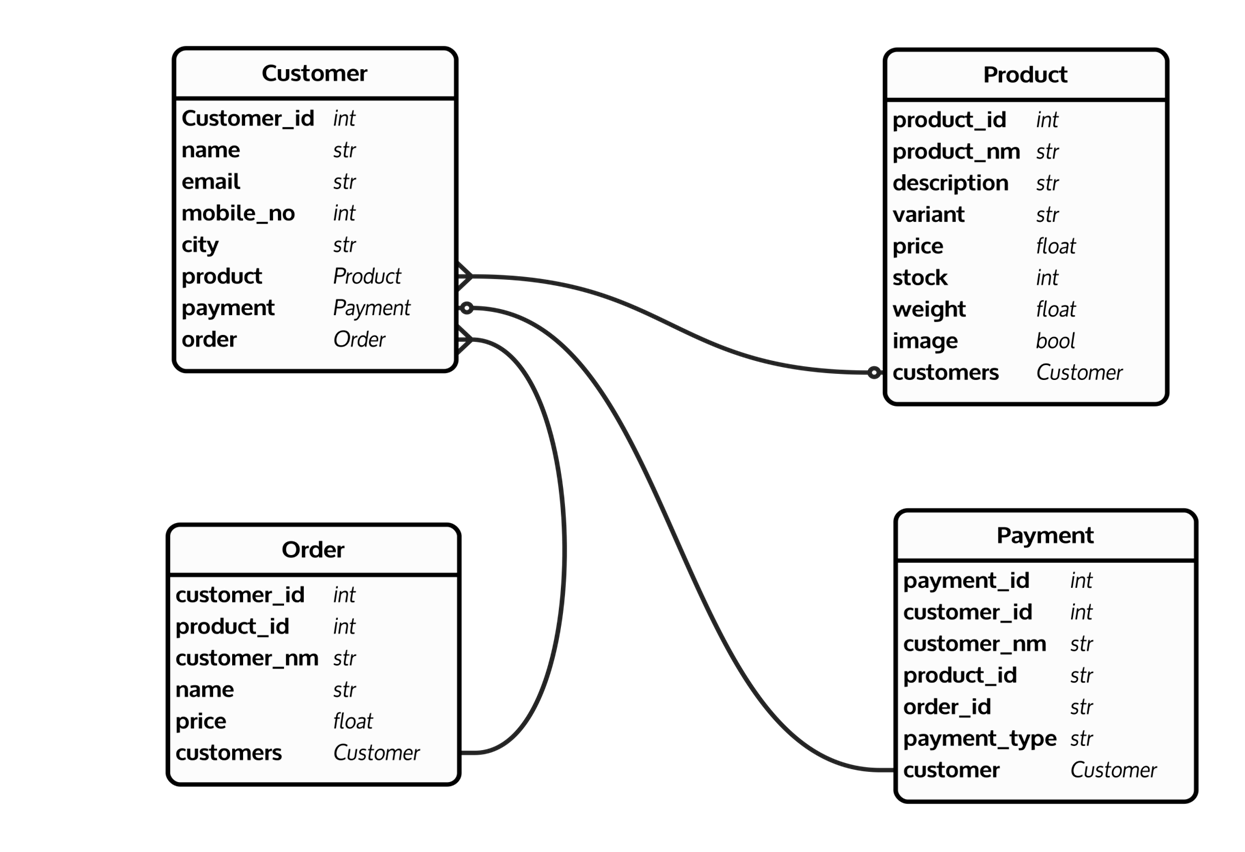
* **Context level Diagram**



**3.2 Data-Flow Diagram**



**3.3 E-R Diagram**



* 1. **Data Dictionary**

1. **Customers**

|  |  |  |  |
| --- | --- | --- | --- |
| Filed Name | Data Type | Constraint | Explanation |
| Name | string | Required | Name of Customer |
| odoo\_customer | String | Required | Name of Odoo customer |
| Instance | String | Required | Odoo Instance |
| shopify\_customer\_id | Ineteger | Required | Shopify Customer Id |
| update\_required | String | Required | Customer Required update |

|  |  |  |  |
| --- | --- | --- | --- |
| Filed Name | Data Type | Constraint | Explanation |
| name | string | Required | Sale Order of name |
| odoo\_sale\_order | Many2one | Required | Connect for odoo sale order |
| instance | Many2one | Required | Sale Order Instance |
| Payment\_gateway | Many2one | Required | Order of payment |
| shopify\_sale\_id | Integer | Auto Fetch | Connect with shopify sale order |
| update\_required | Boolean | Required | Order of required update |

1. **Sale Order**
2. **Product**

|  |  |  |  |
| --- | --- | --- | --- |
| Filed Name | Data Type | Constraint | Explanation |
| odoo\_product | Many2one | Auto Generated | Product generate |
| Instance | Many2one | Required | Product Instance |
| shopify\_product.id | Integer | Auto Fetch | Connect with odoo |
| total\_variant | Integer | Auto Fetch | Connect with shopify |
| sale\_count | Integer | Required | Order of item to count |
| categories | Many2one | Required | Product Categories |
| product\_variant | One2many | Auto Generated | Auto generate Based on Product |
| Filed Name | Data Type | Constraint | Explanation |
| name | string | Required | Name of Product |
| sale\_price | Float | Auto Fetch | Price of Sale Product |
| internal\_reference | string | Auto Generated | Unique |
| weight | Float | Required | Product Weight |
| barcode | String | Auto Generated | Product Barcode |
| quantity\_available | Float | Required | Quantity is available yes or no |
| attibutes\_lines | One2many | Required | Attributes of size, color,etc required |
| description | String | Required | Product related decriptio |
| image | Binary | Required | Product Image |

**4.Instant**

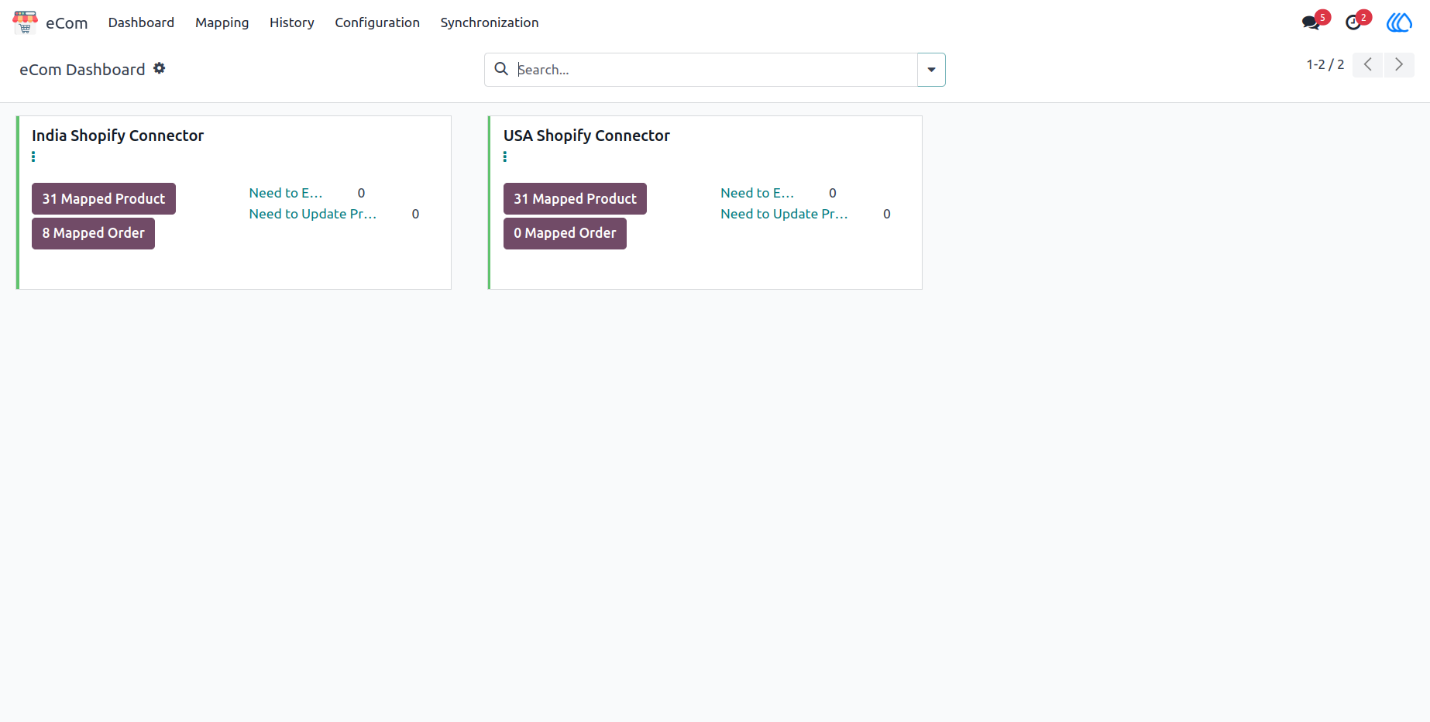
|  |  |  |  |
| --- | --- | --- | --- |
| Filed Name | Data Type | Constraint | Explanation |
| Name | string | Required | Name Product instance |
| Provider | String | Required | Product holder |
| url | String | Required | Url of shopify to Connect |
| User | Many2one | Required | Name of Instance user |
| create.date | Date | Required | Instance date |
| connection\_message | String | Required | Connection message |
| api\_key | String | Auto Fetch | Product of API key to auto generated |
| password | String | Auto Fetch | Product of password to auto generated |
| version\_shopify | String | Auto Fetch | Version of Shopify |
| shop\_name | String | Required | Name of shop |
| sale\_order\_name | String | Required | Name of Sale Order |
| stock.manage | Selection | Required | Manage of stock |
| add\_tax | Selection | Integer | Add Tax of Sale price |
| update\_stock\_at\_export | Selection | Required | Required of any export update |
| price\_rules | Many2one | Float | Price |

**5.History**

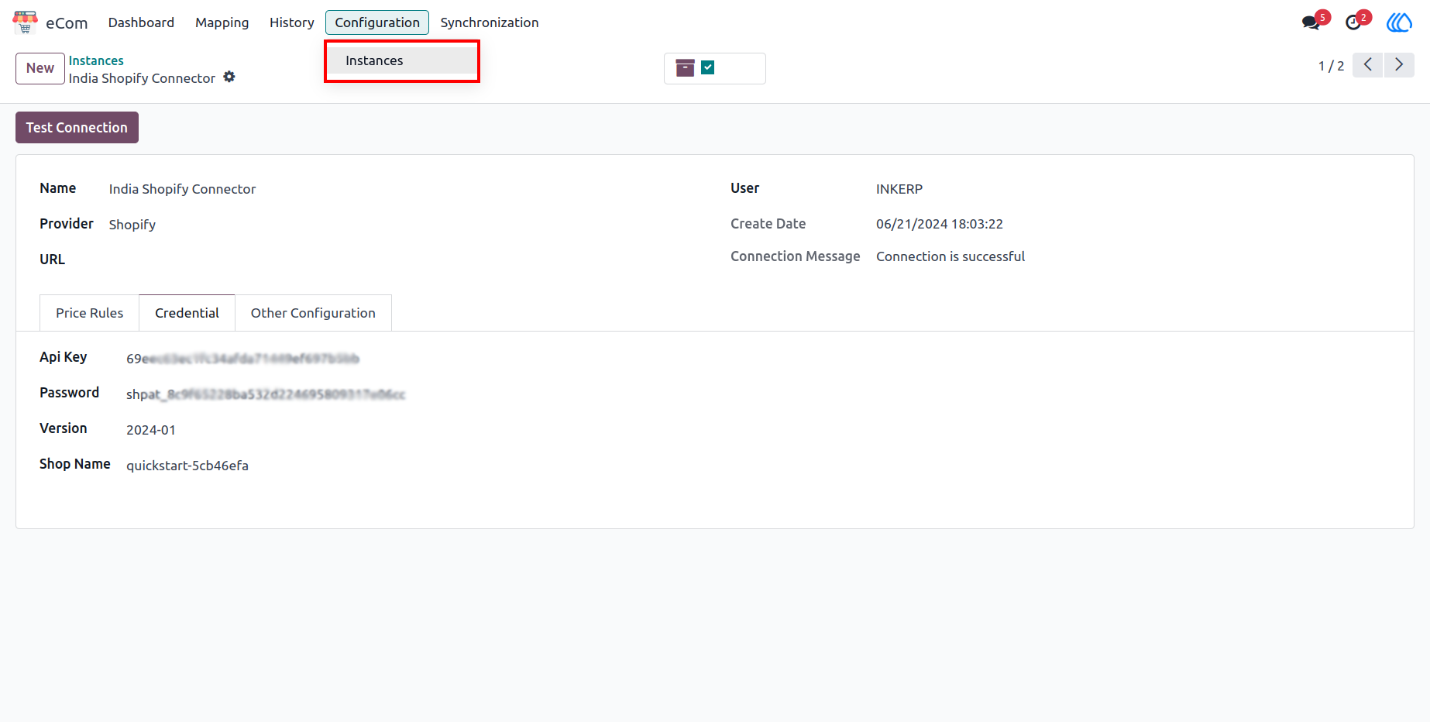
|  |  |  |  |
| --- | --- | --- | --- |
| Filed Name | Data Type | Constraint | Explanation |
| name | string | Required | Name |
| process | Selection | Auto Fetch | Connect with shopify to process on |
| process\_on | Selection | Auto Fetch | Connect with shopify to process on |
| create\_date | date | Required | Required date |
| parent | Many2one | Required | History of parent |
| Status | Selection | Required | Status of |
| instance | Many2one | Required | Burglary/Panic/Robbery/Fire |
| abstract | Text | Required | true/false |
| child\_history | Many2one | Required | Comment Date & Time date |

* 1. **Screenshots**

1. **Dashboard**

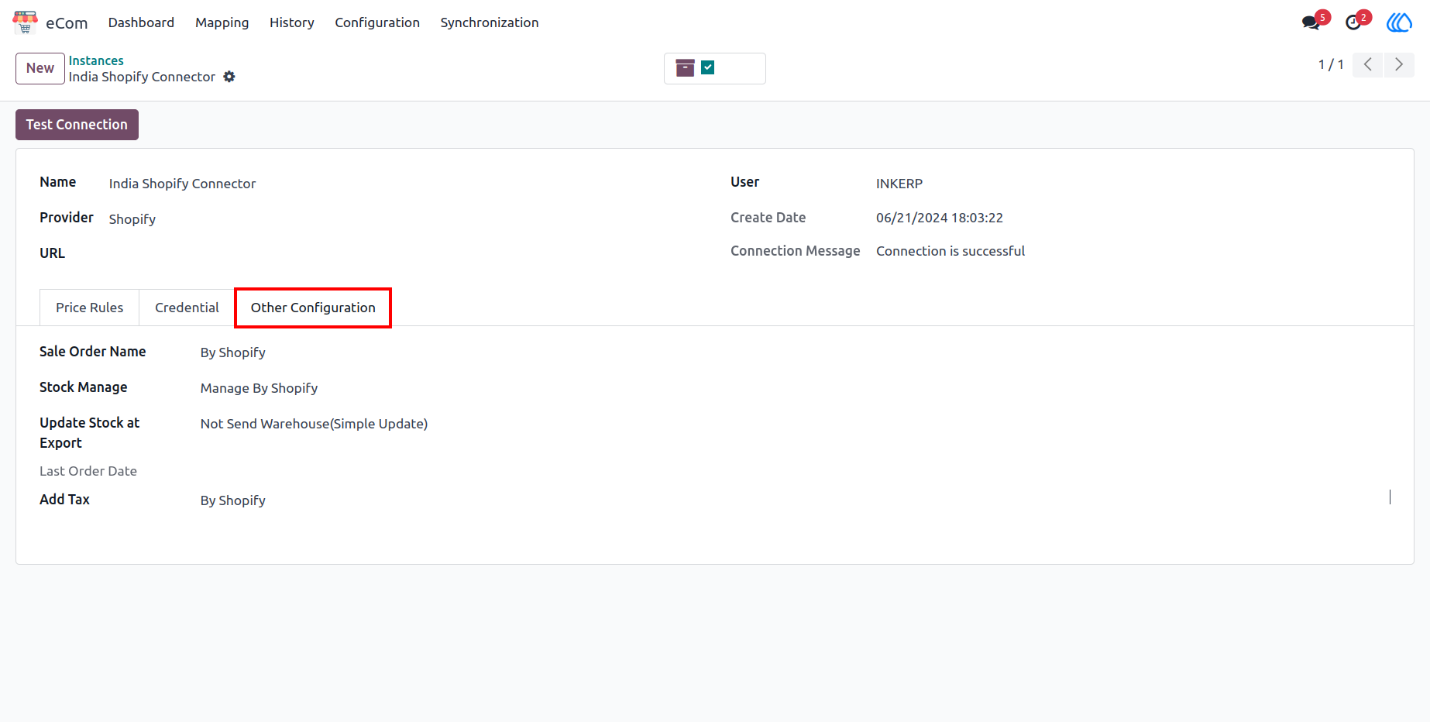


**2.Instance Creation**



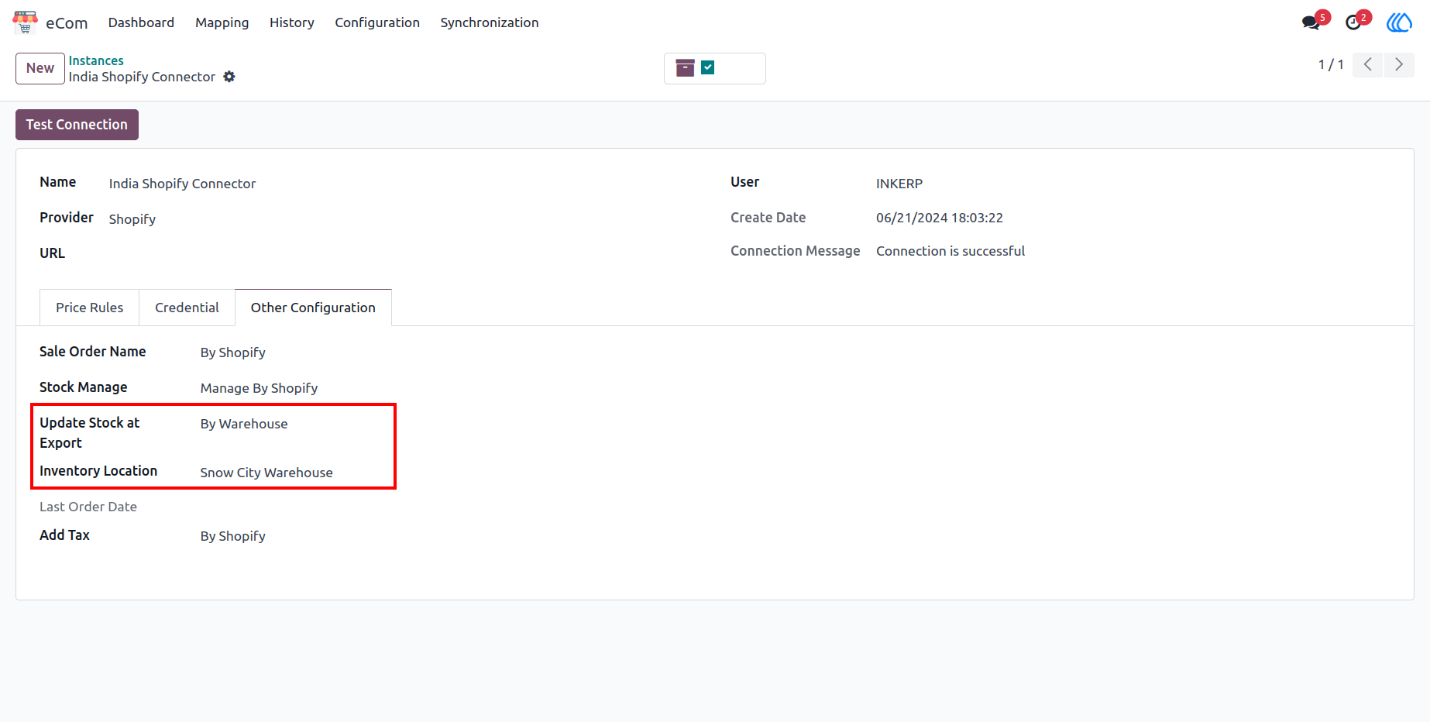
**2.1) Other Configuration Tab in Instance**

* Navigation: eCom -> Configuration -> Instances
* Sale Order Name: This will allows user to choose whether name of the sale order will be managed by Shopify or Odoo.
* Stock manage: This will allows user to choose whether stock will be managed from Odoo or user will update stock manually on Shopify.
* Add Tax: Here user can choose whether tax will be imported from shopify or will be managed by Odoo as per default Odoo functionality.
* Stock Update: Here user can choose all or multi warehouse if user choose multi warehouse it will enable option for user to choose desired warehouse from which stock need to be exported**.**



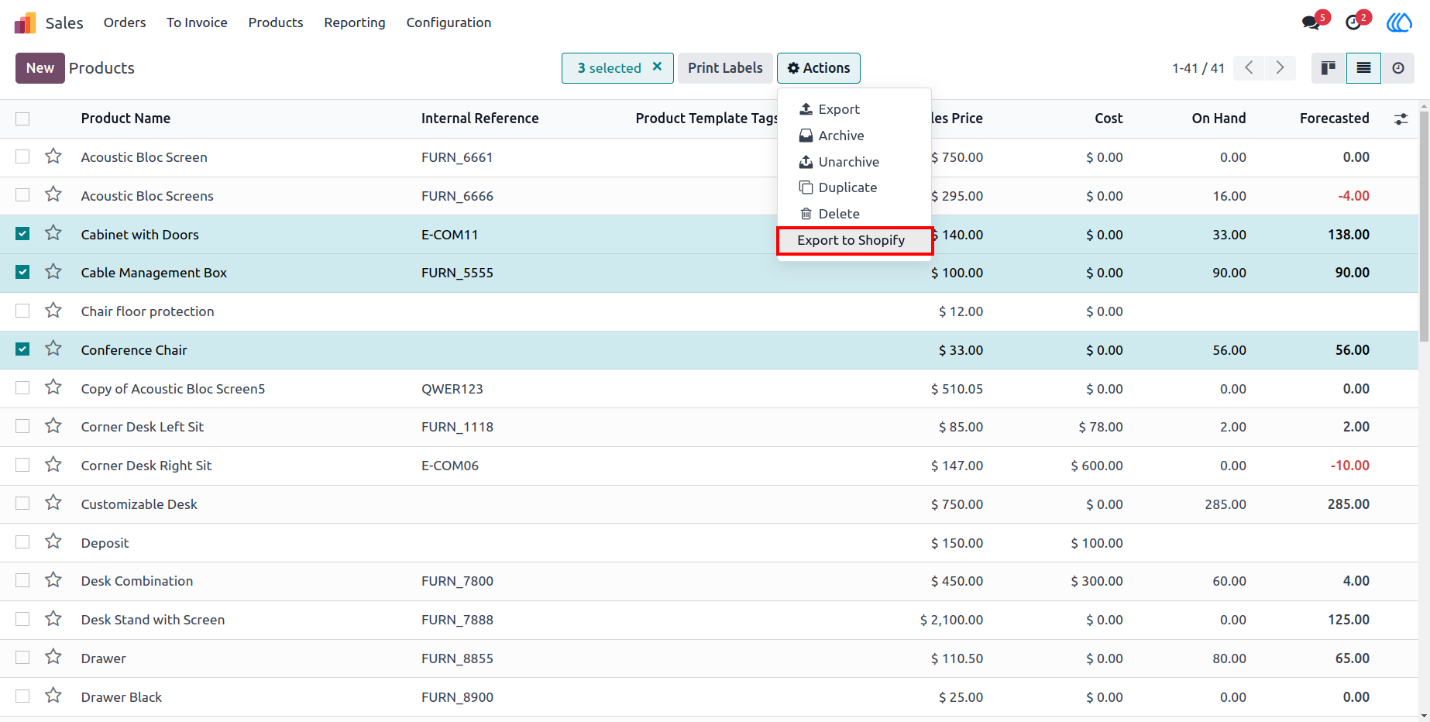
**2.2) Warehouse Configuration for Stock Update**

By choosing warehouse connector will send stock of only that Warehouse to the Shopify.



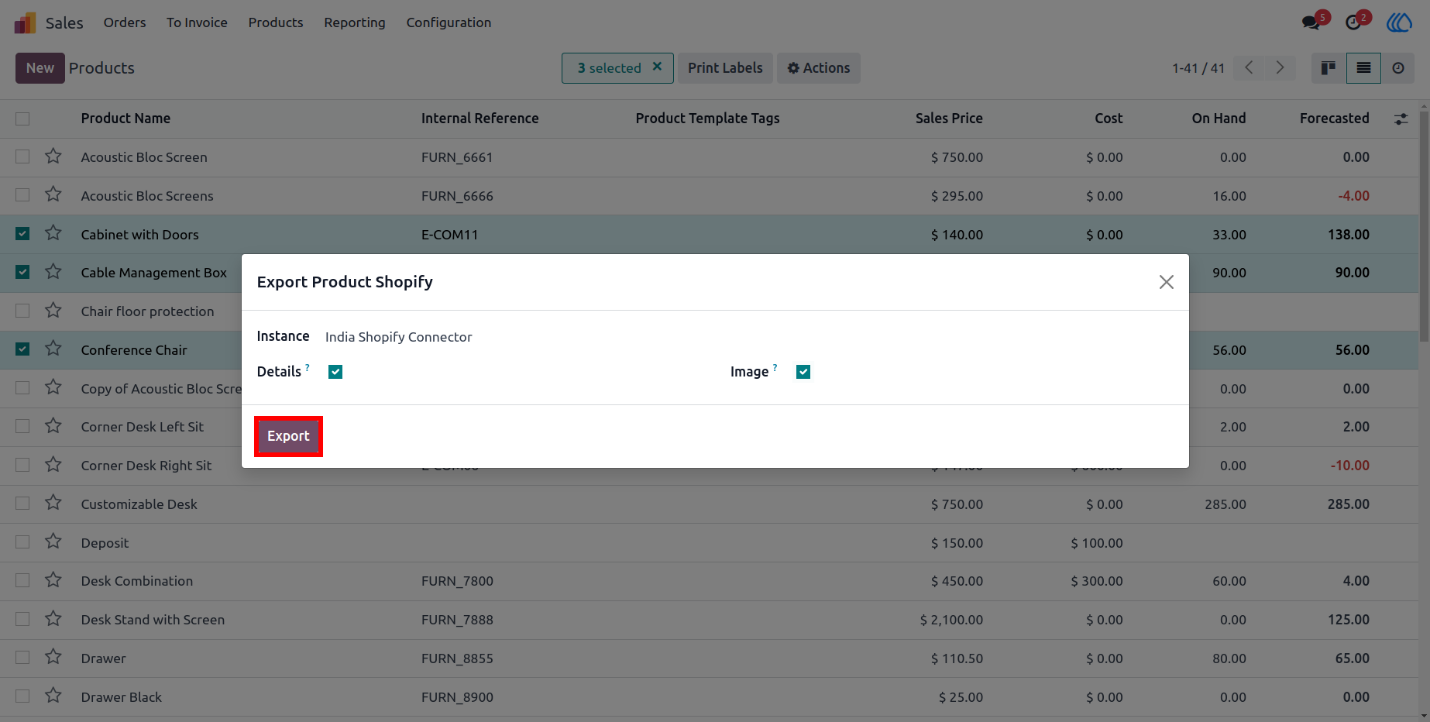
**3.Product**

**3.1.Exporting product to Middle Layer**

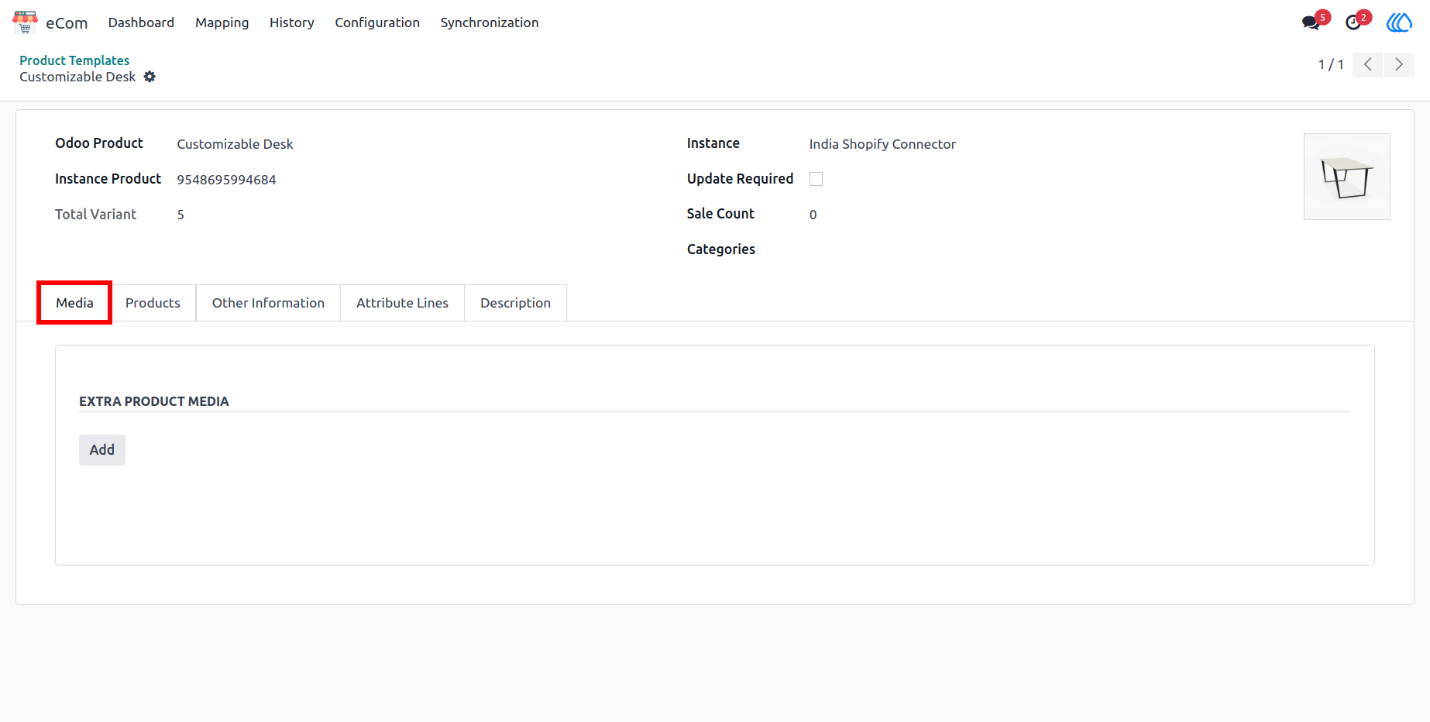


**3.2 Export product to Middle Layer Pop-up**

* Instance: Here user gets the option to choose the instance where he want to export this Products.
* Details: By checking this it will updates details of product into middle layer
* Image: This will allows user to choose whether to export image or not.

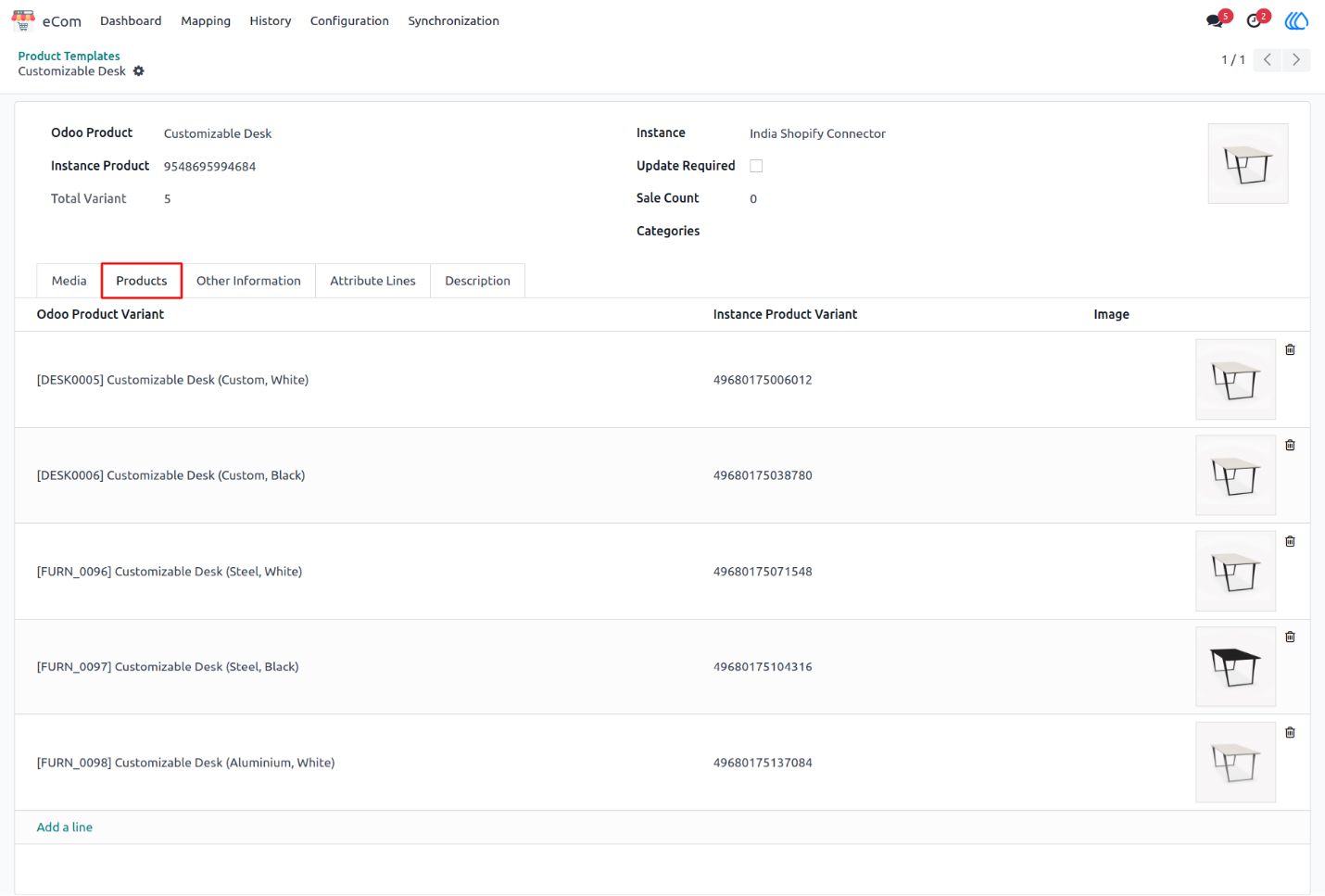


**4.Detailed view of Product Template in Middle Layer  
Navigation: eCom -> Mapping -> Product Templates**

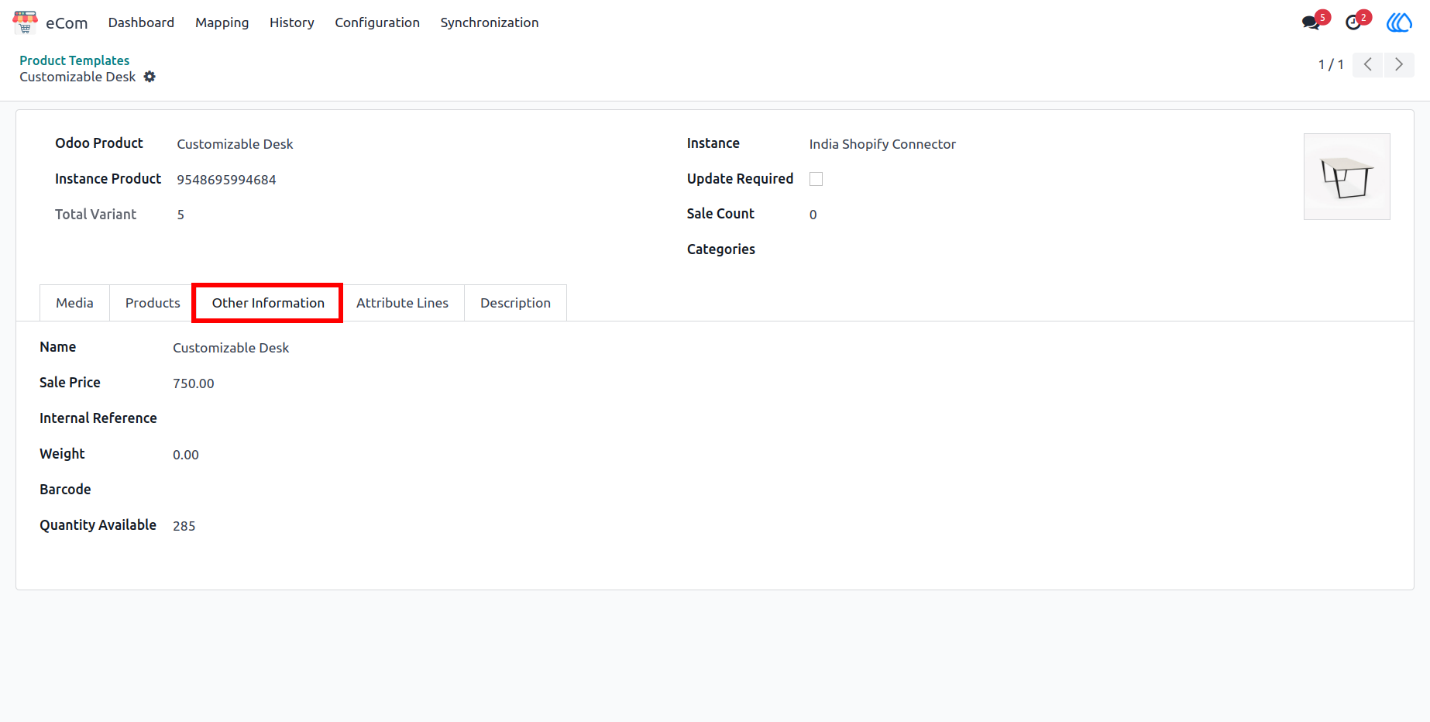


**4.1 Product Tab**

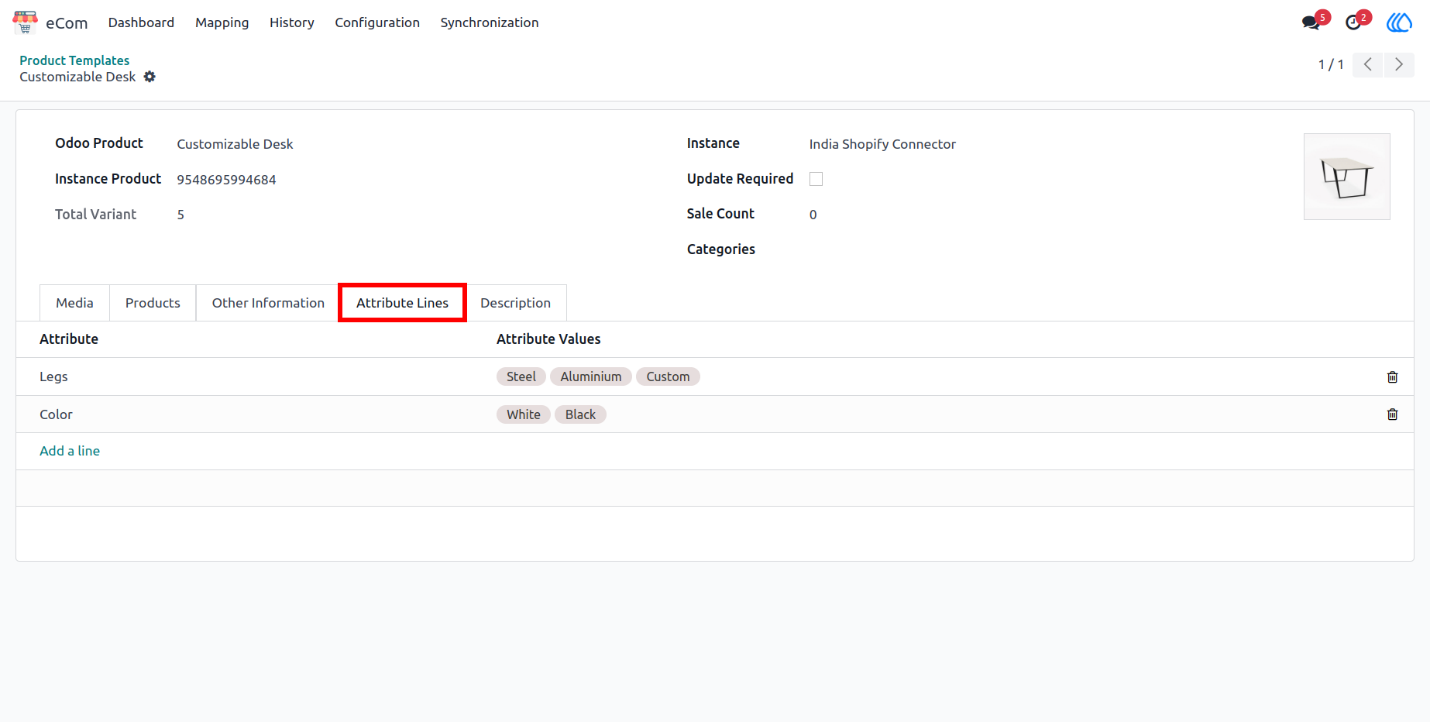
Here user can find the variant of that product and if that is already mapped with Shopify then it will also show Shopify ID.



**4.2 Other Information Tab**

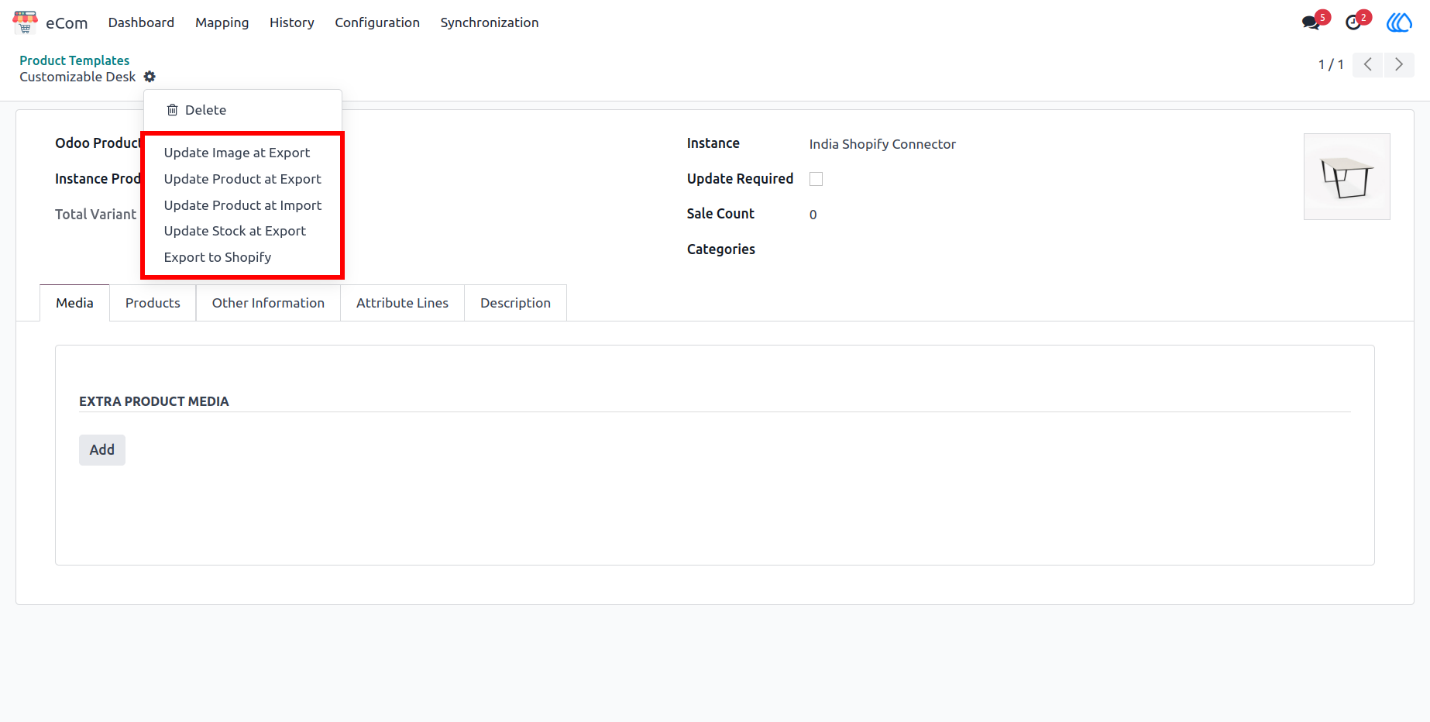


**4.3 Attributes Line Tab**



**5.Exporting Product to Shopify From Middle Layer**

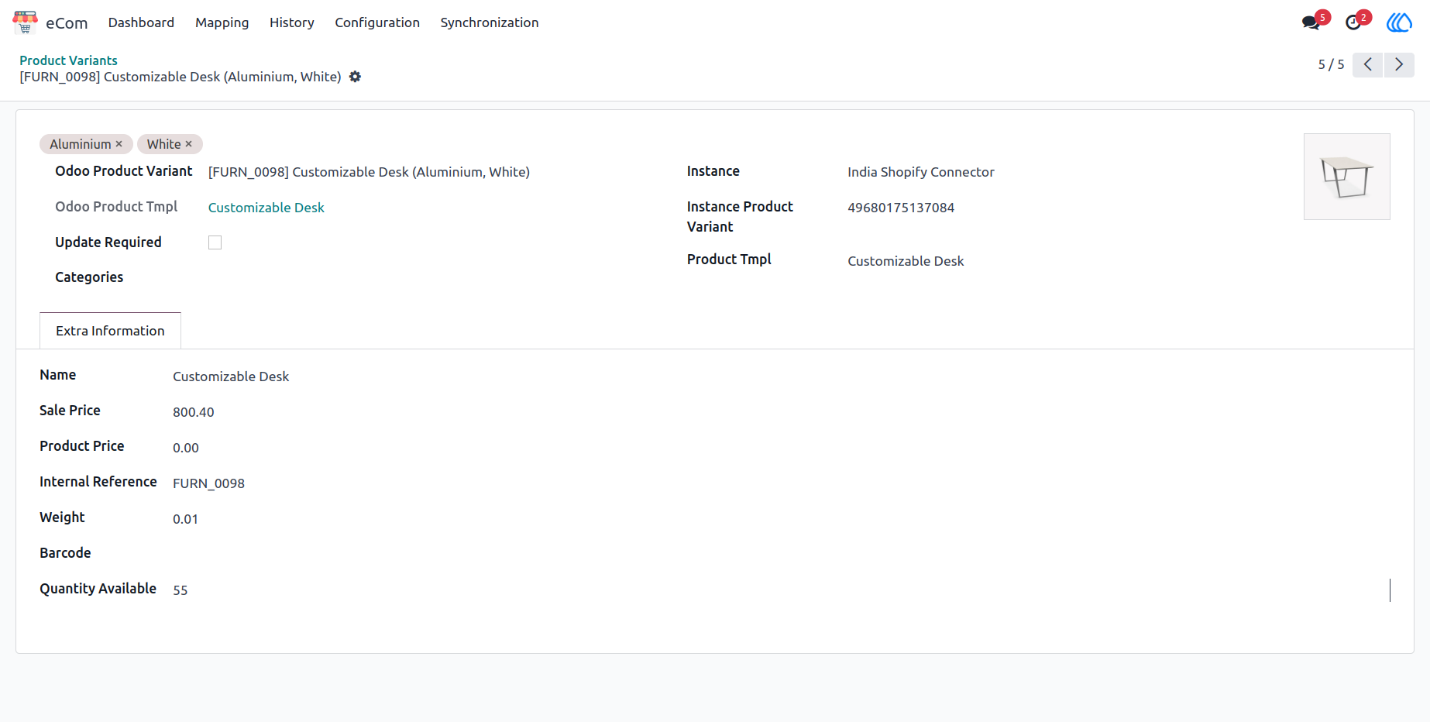
By clicking action button in Product template of middle layer user can export product to Shopify. If will only export product if Update required is checked.



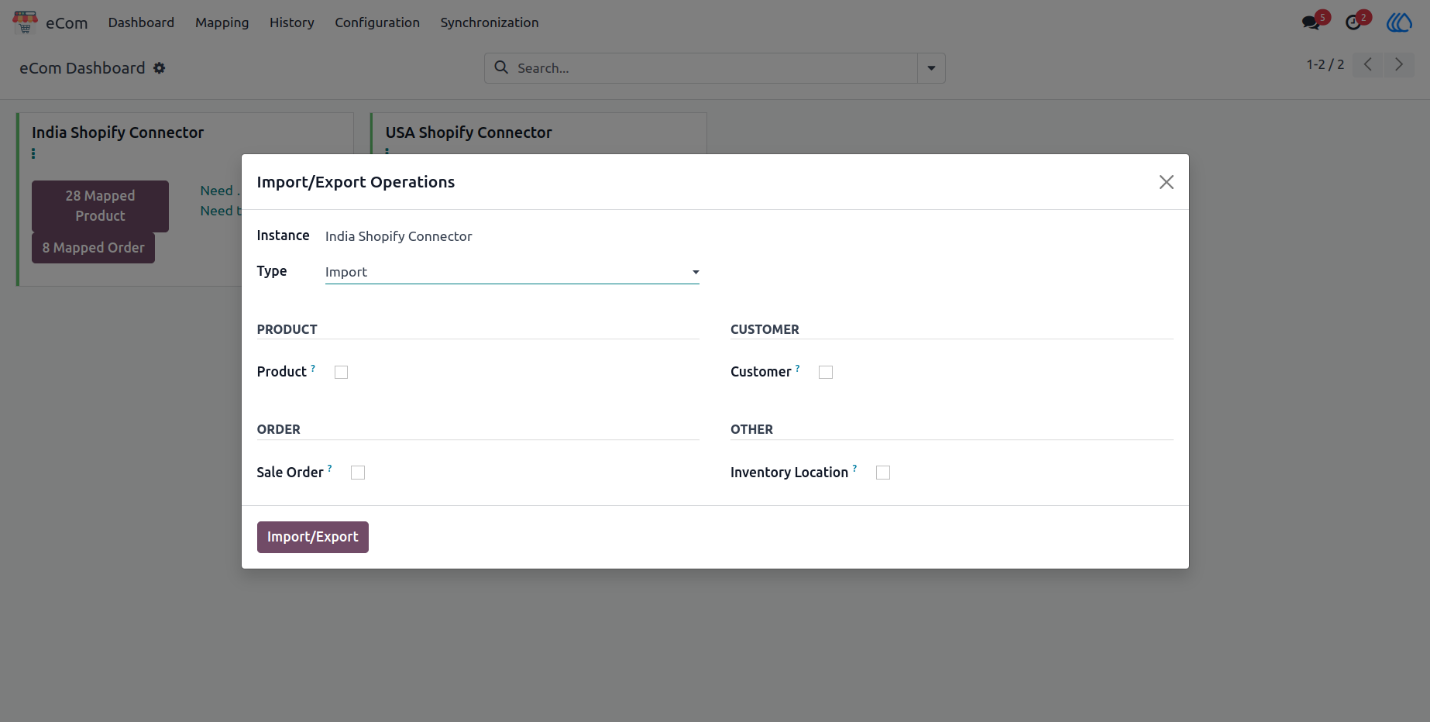
**6.Form View of Product Variant Middle Layer Navigation**

eCom -> Mapping -> Product Variants

Here user can find Product Variant related details in middle layer with Shopify and Odoo details

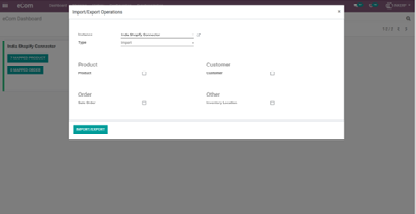


**7. Bulk Import, Export and Update Operations Popup**Navigation: eCom -> Synchronization -> Import/Export Operations  
Instance: Here user can choose the instance for Operation.  
Type: Here user can select which type of operation He/She wants to perform.



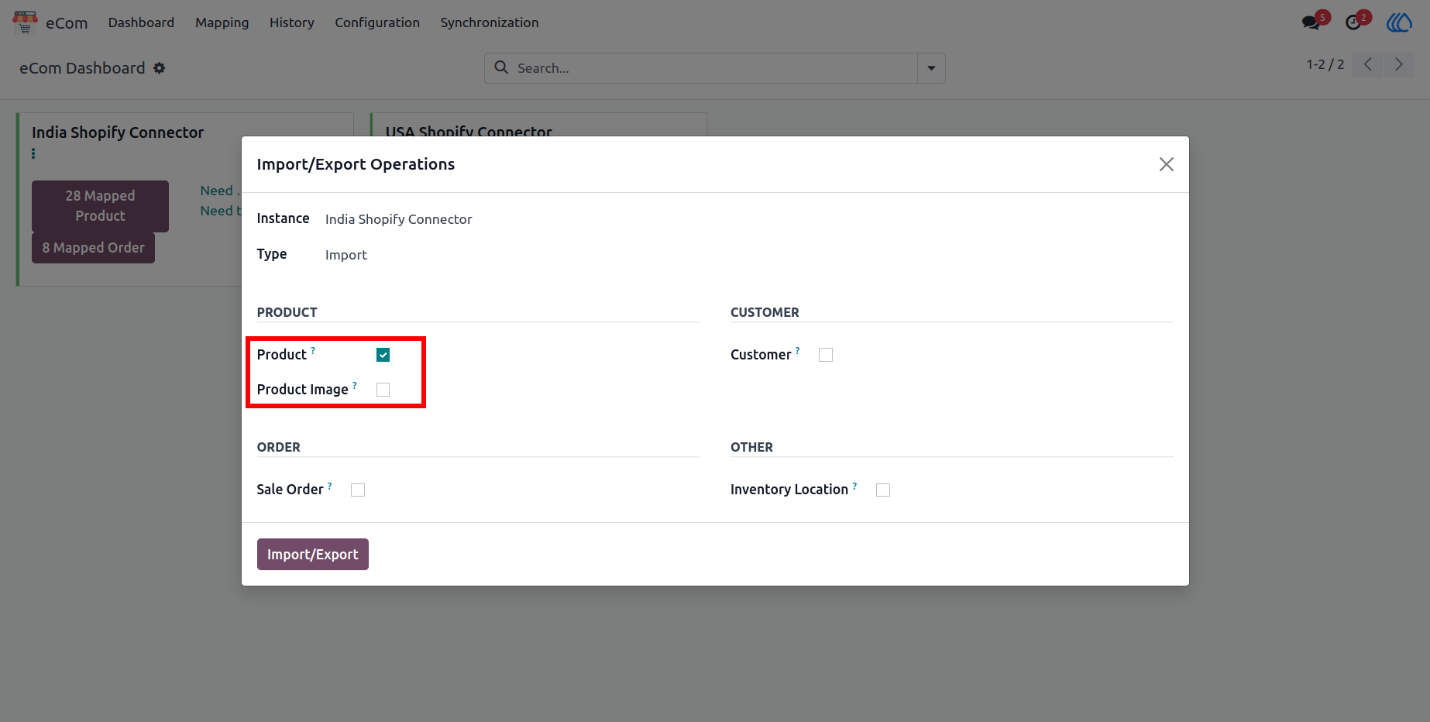
**7.1 Import Operation**

In Import operation user can choose Product, Sale Order, Customer or Inventory location to import such data**.**

****

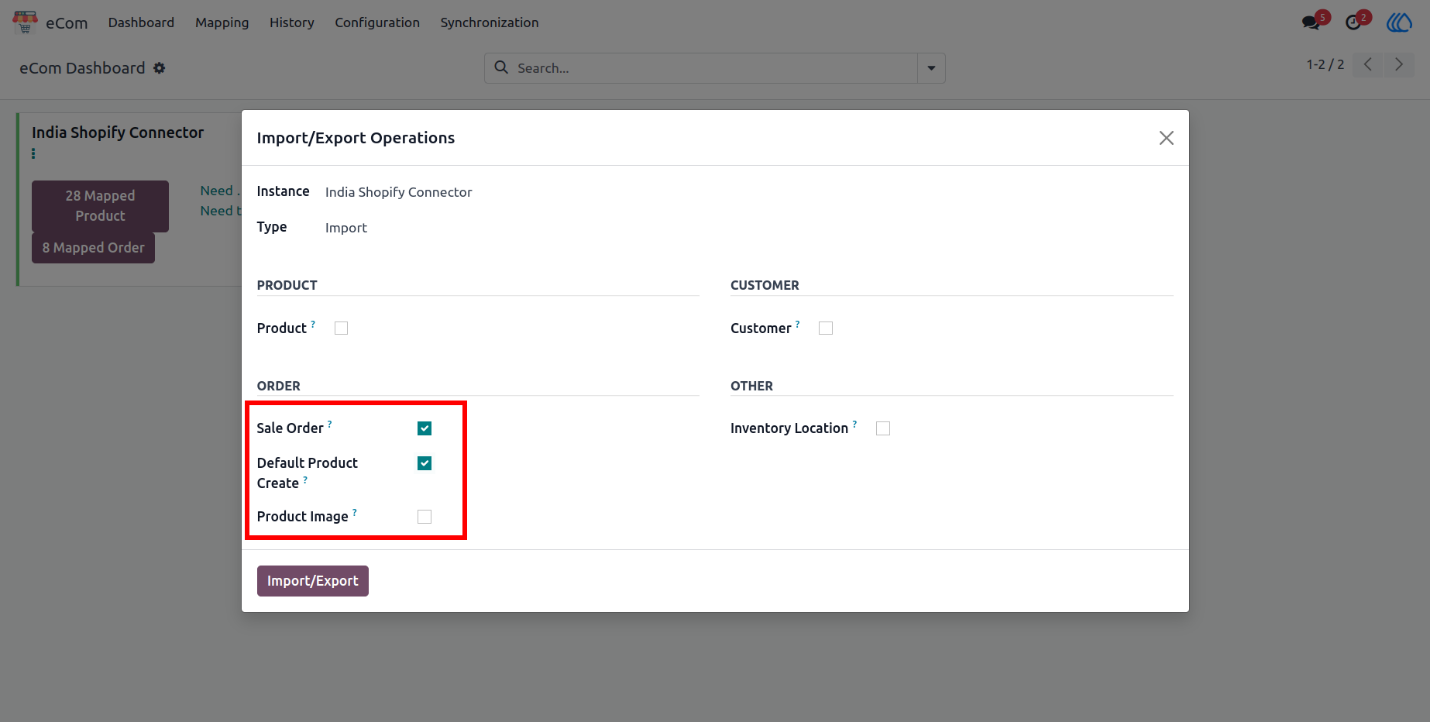
**7.1.1 Product Import Operation**

Once user select Product for the operation it will reveal one more optional "Product Image". By which user can decide whether to import image or not**.**



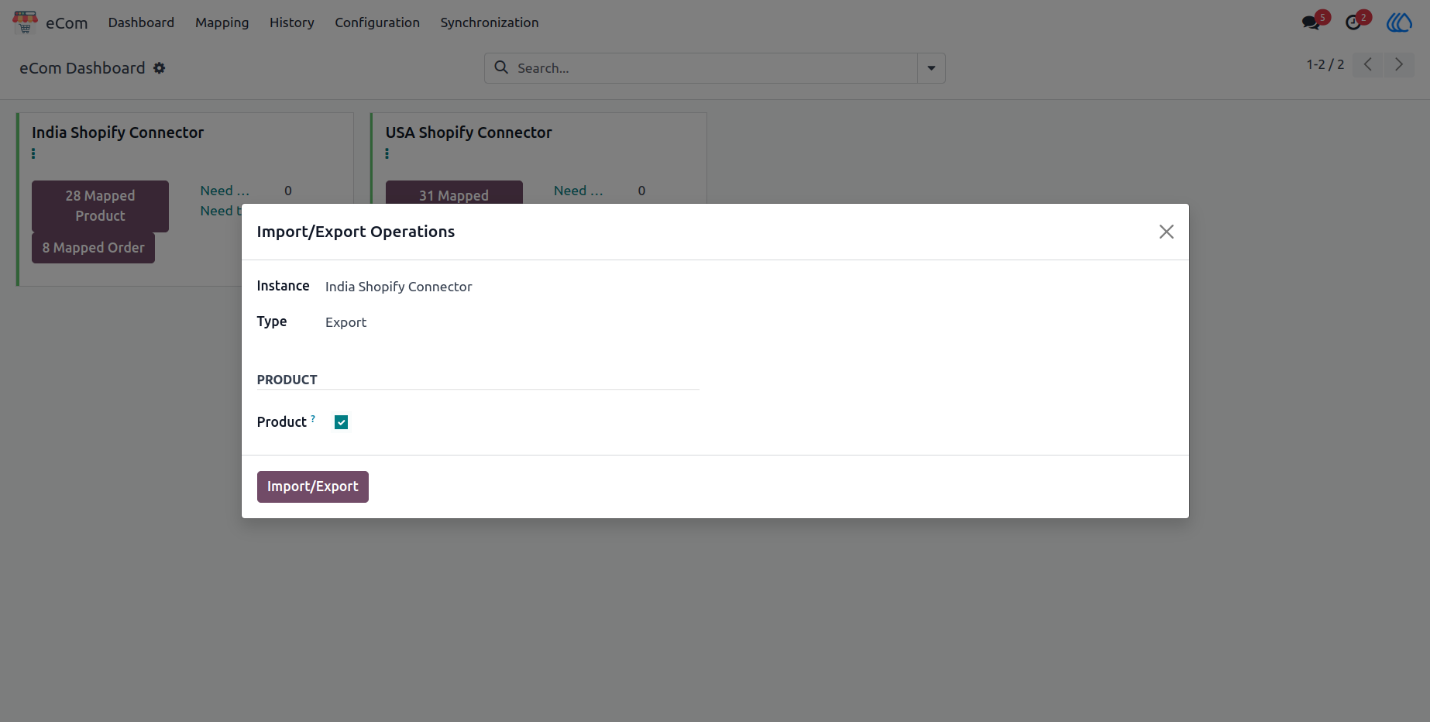
**7.1.2 Sale Order Import Operation**

Once user select Sale Order for the operation it will reveal one more option "Create Missing Product". By which user can decide whether to import and create missing product or not**.**



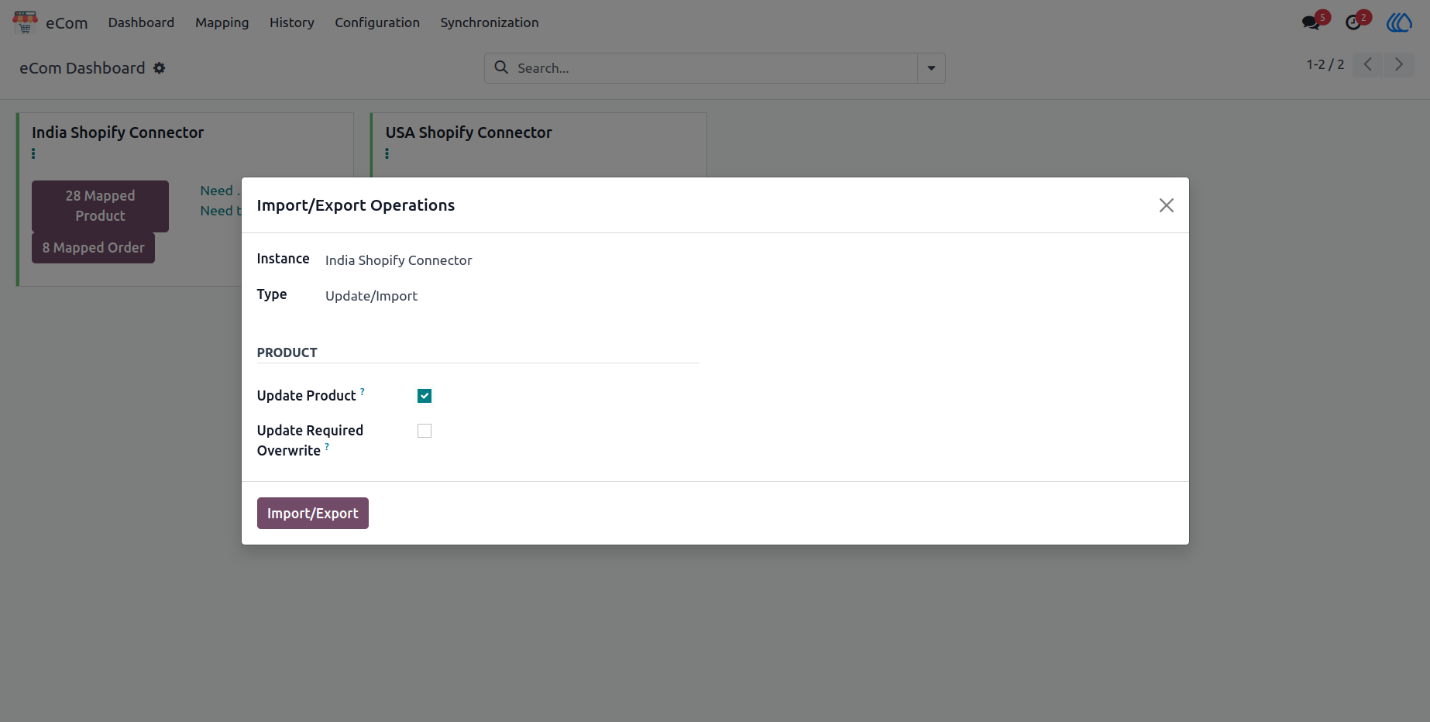
**7.2 Import Operation**

By selecting Export in Type it will show export operation selection where user can choose product and can export product to the Shopify.



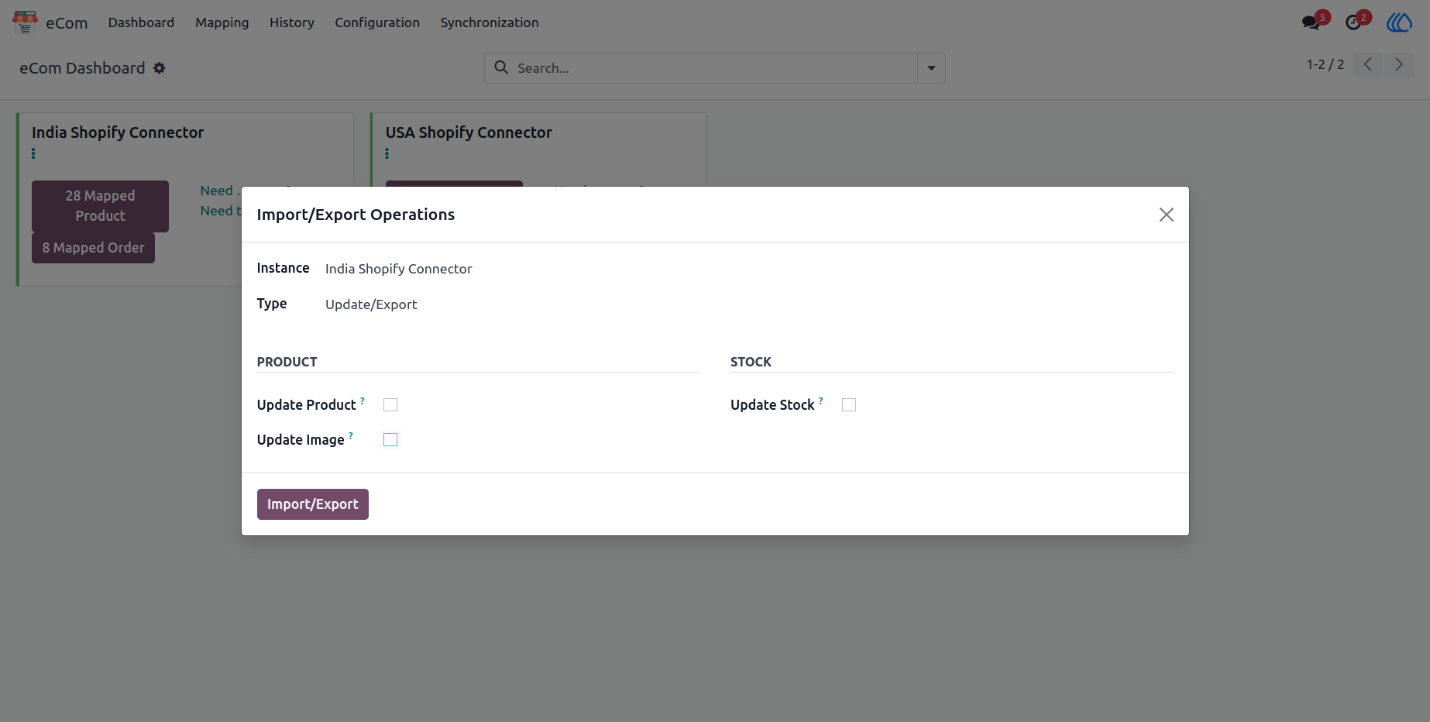
**7.3 Update/ Import Operation**

By selecting Update/Import in Type. It will show import operation selection where user can choose product it will also reveal one more selection Update Required Override. If user choose override operation it will import all product data from Shopify update to middle layer regardless there is any changes in Odoo or not.



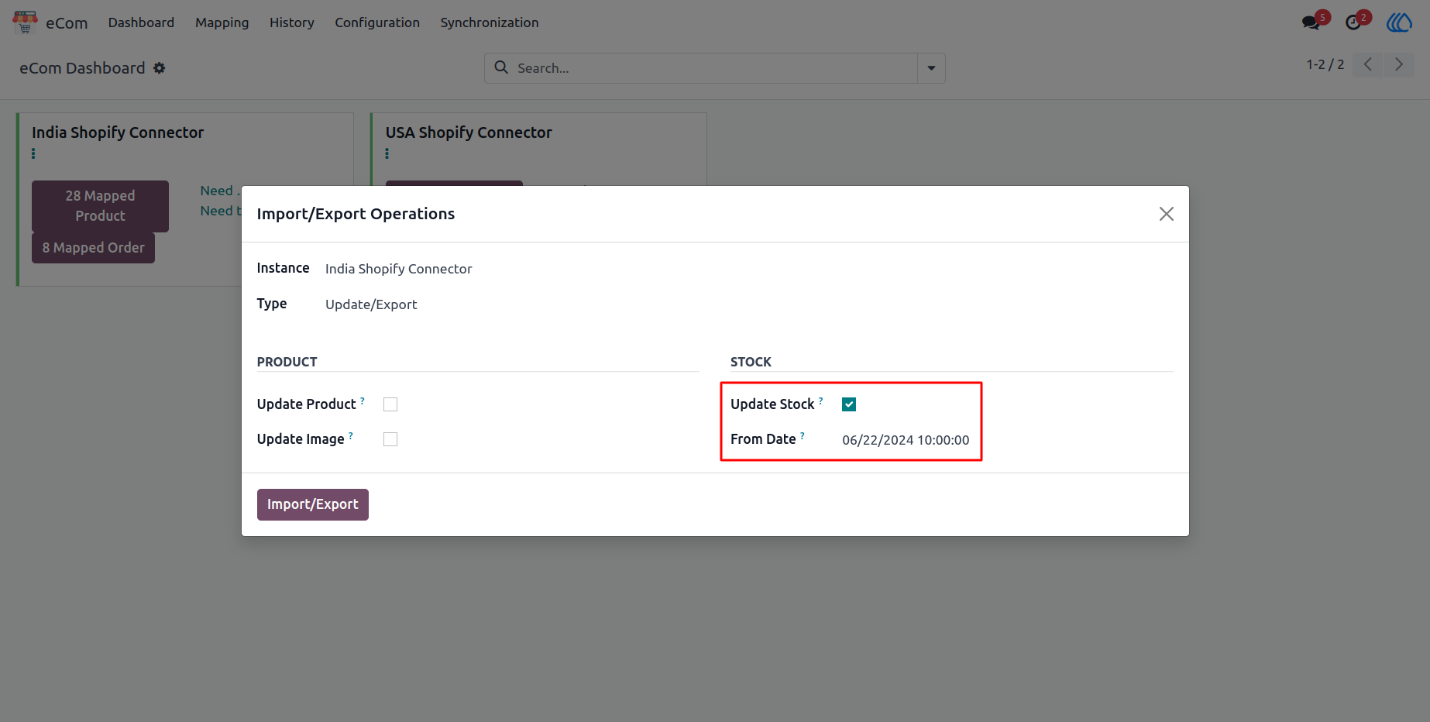
**7.4 Update/ Export Operation**

By selecting Update/Export in Type. It will show Update/Export operation selection where user can choose Update Product, Update Image, Update Stock.



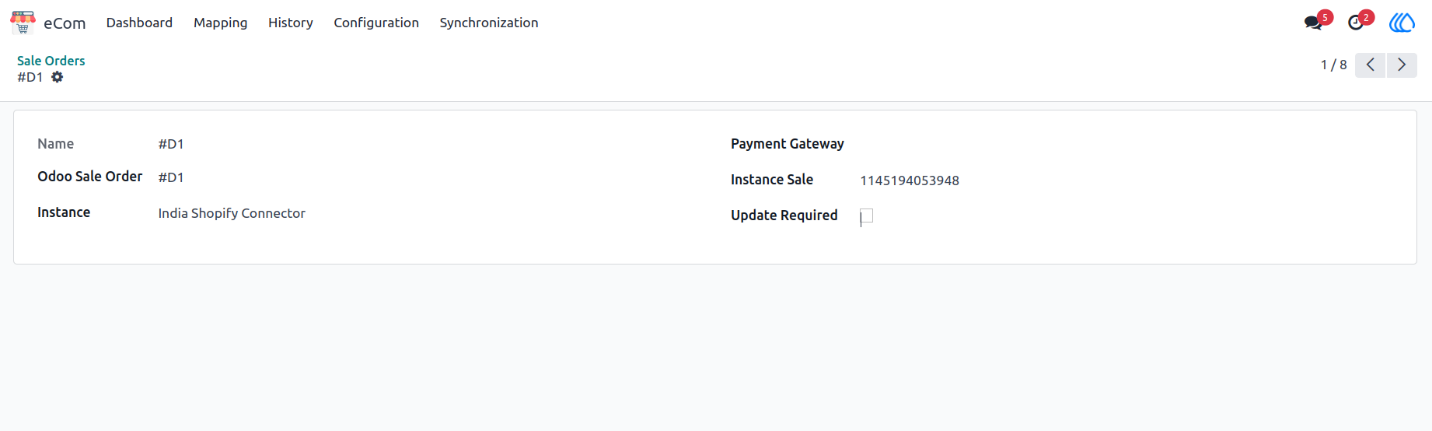
**7.4.1 Update/ Export Stock Operation**

When User choose Update stock it will reveal date picker so if only those products stock will be exported which stock is update after that date and not all product's stock.



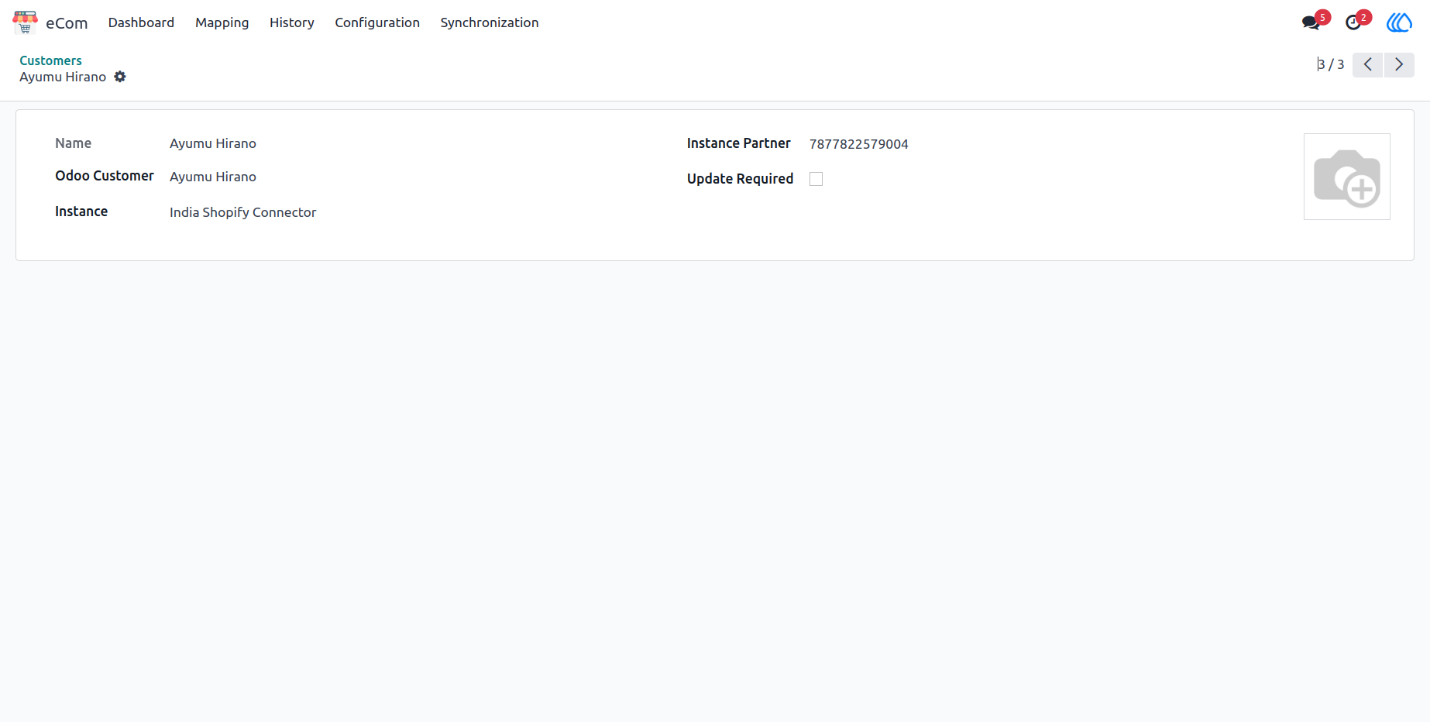
**8. Form View of Sale Order Mapping in Middle Layer**

**Navigation : eCom -> Mapping -> Sale Orders**



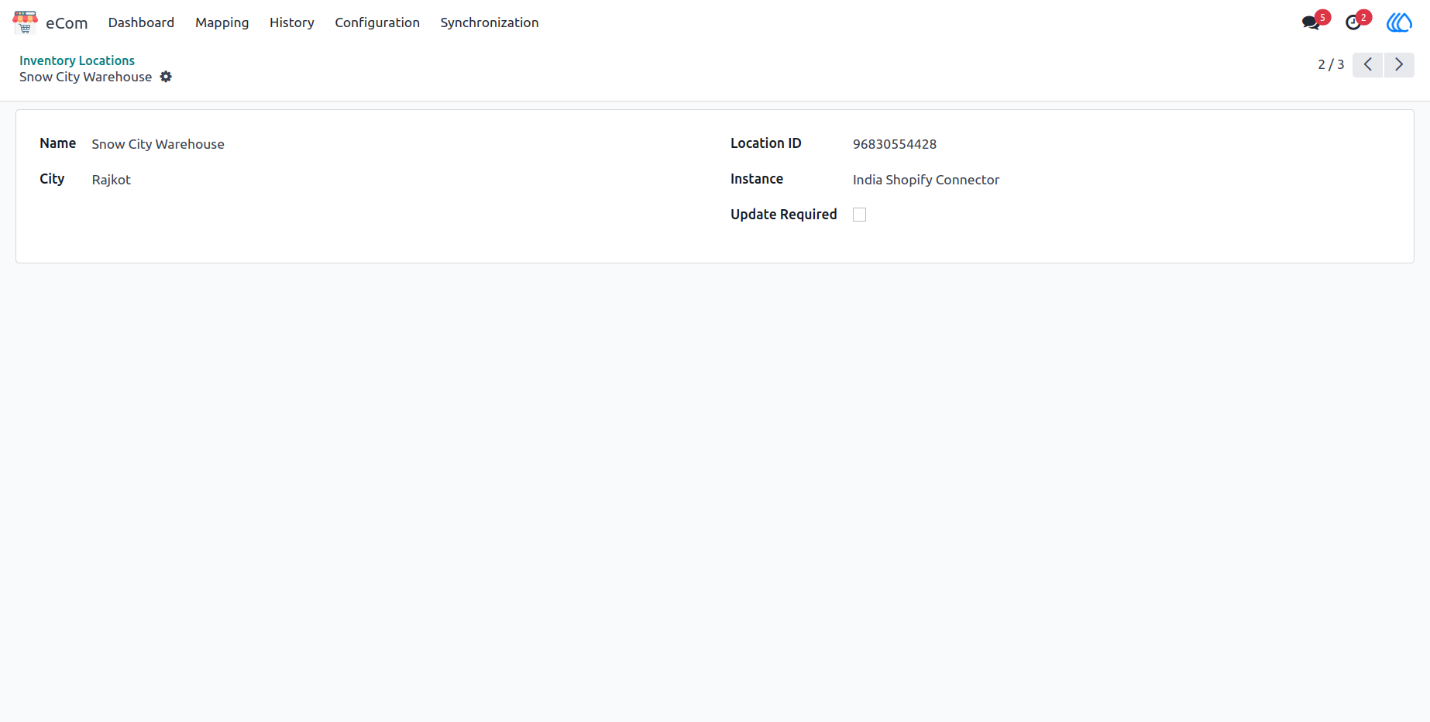
**9.Form View of Customer Mapping in Middle Layer**

**Navigation : eCom -> Mapping -> Customer**



**10.Form View of Warehouse Location Mapping in Middle Layer**

**Navigation:  eCom -> Mapping -> Inventory Locations**



**4. Development Standards**

**4.1 Coding Standard**

Software developers are like lone wolves who prefer to work as individuals rather than in a group. I too come in that category. This can sometimes create problems when the need to work in groups arises in a project.

There are both *pros* and *cons* involved when working in groups. On the plus side, there are *more insights*available to a problem which can help get to a better solution, but at the same time there can be collaboration issues when one developer has to go through another developer’s code to debug or review it.

Each individual has his/her own way of writing code and more often than not, there is a high degree of variation among a group of developers. To avoid such problems when working in a team, most of the programming language communities follow standard coding guidelines. By following these guidelines every developer can write code in a specific way that is known to their teammates. This practice can help teams save time reading the code written by other members.

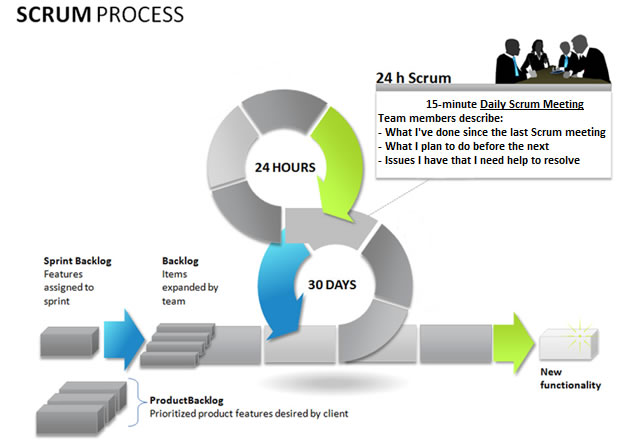
This article is a guide for writing consistent and aesthetically pleasing NodeJScode. It is inspired by what is popular within the community, and also features some personal opinions.

**5 Agile Documentation**

* Agile software development refers to a group of software development methodologies based on iterative development, where requirements and solutions evolve through collaboration between self-organizing cross-functional teams.
* Agile methods or Agile processes generally promote a disciplined project management process that encourages frequent inspection and adaptation, a leadership philosophy that encourages teamwork, self-organization and accountability, a set of engineering best practices intended to allow for rapid delivery of high-quality software, and a business approach that aligns development with customer needs and company goals.
* Agile development refers to any development process that is aligned with the concepts of the Agile Manifesto. The Manifesto was developed by a group fourteen leading figures in the software industry, and reflects their experience of what approaches do and do not work for software development.



* Scrum is a subset of Agile. It is a lightweight process framework for agile development, and the most widely-used one.
* **A “process framework” is a particular set of practices that must be followed in order for a process to be consistent with the framework.**
* **“Lightweight” means that the overhead of the process is kept as small as possible, to maximize the amount of productive time available for getting useful work done.**
* A Scrum process  is distinguished from other agile processes by specific concepts and practices, divided into the three categories of Roles, Artifacts, and Time Boxes. These and other terms used in Scrum are defined below.
* Scrum is most often used to manage complex software and product development, using iterative and incremental practices. Scrum significantly increases productivity and reduces time to benefits relative to classic “Waterfall” processes. Scrum processes enable organizations to adjust smoothly to rapidly-changing requirements, and produce a product that meets evolving business goals. An agile Scrum process benefits the organization by helping it to.
* Increase the quality of the deliverables
* Cope better with change (and expect the changes)
* Provide better estimates while spending less time creating them
* Be more in control of the project schedule and state



**6 Testing**

**6.1 System Testing**

System testing is the process of evaluation system to detect differences between given input and expected output. Also to assess the feature of a system. Testing assesses the quality of the product. System testing is a process that should be done during the development process. In other words System testing is a verification and validation process.

**6.1.1 Verification**

Verification is the process to make sure the product satisfies the conditions imposed at the start of the development phase. In other words, to make sure the product behaves the way we want it to.

**6.1.2 Validation**

Validation is the process to make sure the product satisfies the specified requirements at the end of the development phase. In other words, to make sure the product is built as per customer requirements.

**6.2 Testing Methodology**

There are three major types of testing are as follows:

1. **White Box Testing**
2. **Black Box Testing**
3. **Grey Box Testing**

**1. White Box Testing**

White box sometimes called “Glass box testing” is a test case design uses the control structure of the procedural design to drive test case. Using white box testing methods, the following tests were made on the system all independent paths within a module have been exercised once. In our system, ensuring that case was selected and executed checked all case structures. The bugs that were prevailing in some part of the code where fixed.

**2. Black Box Testing**

Black box testing focuses on the functional requirements of the software. This is black box testing enables the software engineering to derive a set of input conditions that will fully exercise all functional requirements for a program. Black box testing is not an alternative to white box testing rather it is complementary approach that is likely to uncover a different class of errors that white box methods like..

* Interface errors
* Performance in data structure
* Performance errors
* Initializing and termination errors

**3. Grey Box Testing**

Grey Box Testing involves having knowledge of internal data structures and algorithms for purposes of designing the test cases, but testing at the user, or black-box level. The tester is not required to have a full access to the software's source code. However, modifying a data repository does qualify as grey box, as the user would not normally be able to change the data outside of the system under test. Grey box testing may also include reverse engineering to determine, for instance, boundary values or error messages.

* + 1. **Other Types of Testing**
* Unit Testing
* Integration Testing
* Functional Testing
* System Testing
* Stress Testing
* Performance Testing
* Usability Testing
* Acceptance Testing
* Regression Testing
* Beta Testing

**Unit Testing**

Unit testing is the testing of an individual unit or group of related units. It falls under the class of white box testing. It is often done by the programmer to test that the unit he/she has implemented is producing expected output against given input.

In my project I have tested all the individual units for various types of errors and cross checked whether the unit performs as per expectations or not.

**Integration Testing**

Integration testing is testing in which a group of components are combined to produce output. Also, the interaction between software and hardware is tested in integration testing if software and hardware components have any relation. It may fall under both white box testing and black box testing

In my application after testing the entire unit separately, I integrated them and applied integration testing. For example, testing the whole profile module.

**Functional Testing**

Functional testing is the testing to ensure that the specified functionality required in the system requirements works. It falls under the class of black box testing.

I have tested all the required basic functionality of application i.e. move data from one module to another module, managing search etc.

**System Testing**

System testing is the testing to ensure that by putting the software in different environments (e.g., Operating Systems) it still works. System testing is done with full system implementation and environment. It falls under the class of black box testing.

**Stress Testing**

Stress testing is the testing to evaluate how system behaves under unfavourable conditions. Testing is conducted at beyond limits of the specifications. It falls under the class of black box testing.

**Performance Testing**

Performance testing is the testing to assess the speed and effectiveness of the system and to make sure it is generating results within a specified time as in performance requirements. It falls under the class of black box testing.

**Usability Testing**

Usability testing is performed to the perspective of the client, to evaluate how the GUI is user-friendly? How easily can the client learn? After learning how to use, how proficiently can the client perform? How pleasing is it to use its design? This falls under the class of black box testing.

Tested the User interface of the application and try to make it more users friendly so any user can easily understand it.

**Acceptance Testing**

Acceptance testing is often done by the customer to ensure that the delivered product meets the requirements and works as the customer expected. It falls under the class of black box testing.

I have submitted the application to the client and it is tested by the client.

**Regression Testing**

Regression testing is the testing after modification of a system, component, or a group of related units to ensure that the modification is working correctly and is not damaging or imposing other modules to produce unexpected results. It falls under the class of black box testing.

After doing some modification the application is tested and it will work well and does not damage any module or result.

**Beta Testing**

Beta testing is the testing which is done by end users, a team outside development, or publicly releasing full pre-version of the product which is known as beta version. The aim of beta testing is to cover unexpected errors. It falls under the class of black box testing.

**7 Future Enhancements**

This E-Commerce ERP system main scope is:

* **Mapping a Product**
* **Mapping a Customer**
* **Mapping sale order**
* **Main part is history , history is work for customer data, product to maintain and sale order to all over created but this data is very large so history is work for this all data is no created but history check the data and display the not created data.**
* **Manage Stock**

**8 Bibliography**

Websites:

* [**https://www.odoo.com/**](https://nodejs.org/en/docs/)
* **https://www.shopify.in/**
* **https://www.odoo.com/documentation/13.0/**
* [**https://shopify.dev/docs/admin-api/rest/reference**](https://shopify.dev/docs/admin-api/rest/reference)