Vlocity Process Library

*Explore & Checkout for Off-Platform Usage (DC WC + DC SDK)*

*Oct 01, 2020*

**Revision Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Document History** | | | |
| **Version** | **Date** | **Editor** | **Comments** |
| 0.1 | 01/10/2020 | AD | Created First Draft |
|  |  |  |  |

**Acronyms:**

|  |  |
| --- | --- |
| **Terms** | **Description** |
| **DC** | **Digital Commerce** |
| **WC** | **Web Components** |
| **SDK** | **Software Development Kit** |

**Table of Contents**

[Description 4](#_heading=h.gjdgxs)

[Overview 4](#_heading=h.30j0zll)

[Process Flow Diagram 4](#_heading=h.1fob9te)

[Use Case Scenario 4](#_heading=h.3znysh7)

[Assumptions 5](#_heading=h.2et92p0)

[Process Package Includes 5](#_heading=h.1ci93xb)

[Configuration Requirements 5](#_heading=h.3rdcrjn)

# Description

This process provides off-platform support to allow users to browse, configure, and add offers to a basket followed by a checkout process.

# Overview

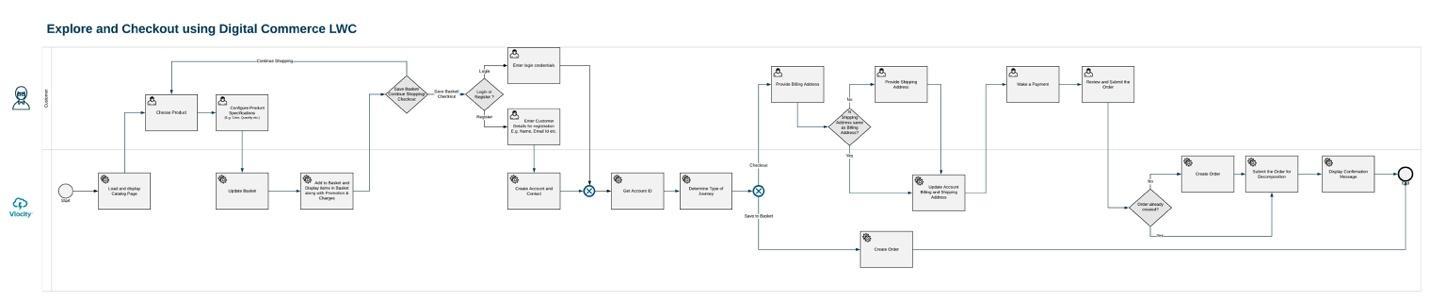
The Explore and Checkout process is built using Digital Commerce Lightning Web Components (LWC) and allows customers to browse and configure products as well as convert their basket to a Salesforce Order by providing the ability to log in to CRM as an identified user (Account), optionally call a payment gateway system, and submit their order. Customer Service Agents can see the items in a customer's cart based on the order ID.

There are three variants of the Digital Commerce process using Lightning Web Components on Vlocity Process Library. Users may choose to consume any of these listings based on their requirements. The first two groups are to support On-Platform DC LWC. The second group additionally contains a Datapack file packaging sample Omniscript that embeds DC LWC to implement the ‘Buy’ flow. The third group is to support the off-platform usage of DC WC/ DC SDK.

Refer to the below table to determine the different kinds of artifacts used in these groups:

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Process Name** | **Artifacts** |
| 1 | Explore & Checkout (DC LWC Only) | * Zip file containing HTML and meta.xml files of corresponding DC LWCs in Managed Package * Datapack file containing VIP for Checkout Process * DX file containing data to be imported into org to run Sample DC LWC App. * A zip file containing Secure server reference implementation embedding the DC SDK to perform login and redirect subsequent calls to DC API via the Secure server. |
| 2 | Explore & Checkout (LWC Omniscript + DC LWC) | * Zip file containing HTML and meta.xml files of corresponding DC LWCs in Managed Package * Datapack file containing Sample Omniscript that embeds DC LWC to implement buy flow * DX file containing data to be imported into org to run Sample Omniscript that embeds DC LWC to implement buy flow |
| 3 | Explore & Checkout for Off-Platform Usage (DC WC + DC SDK) | * A zip file containing Reference App that showcases how DC WC can be customized * Datapack file containing VIP for Checkout Process * DX file containing data to be imported into org to run Sample DC WC App * A zip file containing Secure server reference implementation embedding the DC SDK to perform login and redirect subsequent calls to DC API via the Secure server. |

# Process Flow Diagram



<https://www.lucidchart.com/invitations/accept/17903a56-b239-470a-b8c2-2d9a89f230bf>

# Use Case Scenario

A user browses products from the Catalog page and adds them to the basket. The user can then log in or register to save the basket or check out by optionally completing payment and submitting the order. The Save Basket feature provides an ability to suspend the ordering process transaction by the user to be retrieved at a later point in Salesforce (for example, by a call center agent). Once the customer's basket has been reviewed and finalized, they can begin the checkout process. The Checkout feature is an encapsulation process consisting of the following: Login or Registration, Retrieving Basket, Connecting to Payment Gateway, Submitting Order to Order Management.

# Assumptions

1. Catalog, whose unique ID needs to be passed to Web Components which is present in the Org.
2. The cache is generated after importing product data. See [Preparing to Use API Caching](https://docs.vlocity.com/en/Preparing-to-Use-Vlocity-Communications-API-Caching.html)
3. Configure the product and add it to the basket.
4. Saving a basket creates an order with status as “Saved” in order to be retrieved at a later point.
5. After the customer's basket has been finalized, the Payment process is initiated.
6. The process ends by creating and submitting the order from Salesforce.
7. Braintree has been used as a payment gateway for this solution. The implementation team may replace it with any other payment gateway of their choice.

**Process Package Includes**

DataPack developed and tested on - ***<CME 109>***

**Process Components**

* **ZIP file containing Reference App that showcases how DC WC can be customized**
* **Datapack file containing VIP for Checkout Process (4)**
  + dc\_createAccount
  + dc\_updateBillingDetails
  + dc\_createOrder
  + dc\_submitOrder
* **DX file containing data to be imported into org to run Sample DC WC App**
* **ZIP file containing Server SDK reference implementation.**

# Configuration Requirements

*Create offers with prices and add them to catalogs.*

*(Optional) Configure Attribute-Based Pricing (ABP)*

*Run CMT administration jobs (cached API jobs).*

*Install the latest Node JS into your machine before doing npm install.*

**Note**: Please raise a ticket with support and provide the following information to gain access:

1. Vlocity Sales Rep Name
2. Client Name / Project Name
3. Timeline (Artifact Needed By)

All of the above information is required. Request the ticket to be sent to Digital Commerce Product Management.