



MONASH
University



UpScale



BigCommerce Plugin for ChessWorld SKU Management

Project Requirement Document

Version 4.0

Written by:

Rui En Koe

Alex Chan

Ashley Warden

Liangdi Wang

Luqmaan Yurzaa

Wen Cheng Huong



Document History and Version Control

Date (dd/mm/yy)	Version	Author	Description
24/03/2025	1.0	Wen Cheng Huong	Initial Project Requirement Document Draft
06/04/2025	1.0	Rui En Koe Alexander Chan Wen Cheng Huong	Update PRD: 1. Added NFRs under section 3.1 Non-Functional Requirements
08/04/2025	1.0	Liangdi Wang	Update PRD: 1. Added technical justification under section 3.2 Technical Considerations
17/04/2025	2.0	Wen Cheng Huong	Update PRD: 1. Updated the requirements based on the client's feedback on the UI mockup and incorporated the necessary changes.
18/04/2025	2.0	Rui En Koe	Update User Stories with the latest requirements based on the client's feedback
30/04/2025	2.0	Alexander Chan Rui En Koe	Update PRD: 1. Update variants requirement based on the client's requirement. 2. Update User Stories on the variant part.
18/08/2025	3.0	Wen Cheng Huong	Update PRD: 1. Included high level requirements for additional feature suggested by the client.
31/08/2025	3.0	Rui En Koe	Update PRD: 1. User Stories with the latest requirements based on the client's feedback. 2. Included high level requirements for additional features proposed by the client.
5/10/2025	3.0	Rui En Koe	Update PRD: 1. User Stories with the latest requirements based on the client's feedback.
6/10/2025	4.0	Rui En Koe	Finalised User Stories and PRD.
27/10/2025	4.0	Rui En Koe	Update and Revise PRD: 1. User Stores
28/10/2025	4.0	Wen Cheng Huong	Updated section on scope negotiations.



Contents

1. Project Overview	4
1.1 Project Description	4
1.2 Problem Statement	4
1.3 Project High Level Objectives	4
1.4 Project Vision	5
2. Plugin Requirements and Features	6
2.1 High-level project requirements as per client meeting	6
2.2 Additional project requirements as per client feedback on UI mockup	7
Additional Requirements for SKU Management	7
1. SKU Creation Parameters	7
2. SKU Edit Page (Product Edit Page)	7
3. SKU Selection Modal	7
4. Variant-Level SKU Assignment	8
5. Stock Display & CSV Export Considerations	8
6. SKU Performance Tracking (optional)	8
2.3 Additional Features Initial High-Level Requirements as per Client Request	8
2.4 Additional Project Requirements as per Client Feedback on UAT (optional)	9
3. Other Requirements and Considerations	11
3.1 Non-Functional Requirements	11
3.2 Technical Considerations	11
4. Scope Negotiations	14
4.1 Initial Scope vs Negotiated Scope	14
4.2 Other Scope Negotiations	15
5. User Stories	16
6. Use Cases	24



1. Project Overview

1.1 Project Description

Chessworld, a premium online retailer specializing in chess-related products, requires a robust inventory management solution for its **BigCommerce** store. The project aims to develop a **custom plugin** that enables accurate **SKU-level inventory tracking** while supporting product bundling. The plugin will ensure that bundled products—composed of multiple SKUs—are correctly tracked to prevent overselling. By seamlessly integrating with BigCommerce's existing product and order management system, the solution will provide an intuitive interface for managing bundled SKUs efficiently. This will enhance Chessworld's ability to manage inventory in real-time, ensuring a seamless shopping experience for customers while maintaining operational efficiency.

1.2 Problem Statement

Chessworld sells chess sets that often consist of multiple SKUs from different manufacturers, such as chess boards and chess pieces, which can be sold separately or as part of a bundle. However, **BigCommerce lacks native support for SKU-level inventory tracking within bundled products**, leading to **overselling risks** when a shared SKU runs out of stock. The current system does not synchronize inventory across bundles and individual SKUs, creating potential stock discrepancies. This gap in inventory management affects order fulfillment accuracy and operational efficiency. Chessworld needs a solution that allows **independent SKU tracking within bundles**, ensuring inventory accuracy and preventing overselling while maintaining seamless integration with BigCommerce.

1.3 Project High Level Objectives

- Develop a BigCommerce plugin to manage bundled products with accurate SKU tracking.
- Enable seamless SKU synchronization across individual products and bundles.
- Prevent overselling when a shared SKU is out of stock.
- Provide a simple and intuitive user interface for inventory management.

1.4 Project Vision

Our plugin is designed to support Chess World by enabling independent SKU tracking within bundled products. With real-time SKU-level tracking and automatic stock synchronization between individual SKUs and bundles, it removes the need for manual updates and significantly reduces the risk of overselling. Unlike Smart Manager, our plugin offers automated, SKU-level inventory tracking tailored specifically for bundled products.

Seamlessly integrated with BigCommerce, the plugin improves order fulfillment accuracy and streamlines operations, directly addressing Chess World's current challenges in managing SKU-level inventory.

In addition, the plugin features a clean, spreadsheet-style interface that makes inventory management intuitive and efficient. Its simple, minimalist design delivers an all-in-one solution that prioritizes ease of use and operational excellence for Chess World.



2. Plugin Requirements and Features

2.1 High-level project requirements as per client meeting

MeetingMinutes_24/03/2025

Essential Requirements:

- **Independent SKU Tracking:** Ensure each SKU in a bundle is tracked independently.
- **Bundle Management:** Create and manage product bundles composed of multiple SKUs.
- **Stock Synchronization:** Automatically update inventory when an SKU is sold individually or as part of a bundle.
- **Overselling Prevention:** Restrict bundle sales when any component SKU is out of stock.
- **BigCommerce Integration:** Connect with BigCommerce's existing product and order management system.
- **Admin Interface:** Provide an easy-to-use interface for managing bundled SKUs.

Ideal Features:

- **Left Navigation Panel:** Display all products and SKUs for easy access.
- **Bundle Creation UI:** Search-and-select-based interface for adding SKUs to a bundle.
- **Inventory Dashboard:** Provide real-time stock levels for products (bundled SKUs) and individual SKUs.



2.2 Additional project requirements as per client feedback on UI mockup

MeetingMinutes_30/04/2025

Additional Requirements for SKU Management

1. SKU Creation Parameters

- **Minimum Required Fields:**
 - SKU number
 - SKU name
 - Stock level
- **Optional Fields (time-permitting):**
 - Weight
 - Price
 - Custom Fields (e.g. manufacturer, manufacturer price, landed cost, selling price, margin, total stock revenue)
- **Custom Field Support (optional):**
 - Ability to add optional custom fields

Separation into **custom SKU categories** (e.g. "Chessboard") — separate from product categories

2. SKU Edit Page (Product Edit Page)

- Should be **minimalistic**, not identical to product page
- **No need** for:
 - Publishing functionality
 - Description
 - Photos
 - SEO optimization
- **Optionally** calculate **aggregated values** (e.g. total weight or price when multiple SKUs are used in a product)

3. SKU Selection Modal

- Acceptable to open from a modal (as in the wireframe)
- However, a **simple search field embedded directly** on the product edit form is **sufficient**
 - No need for sorting or price manipulation
 - Selected SKUs can be displayed inline (e.g. separated by commas, with remove buttons)



MONASH
University



UpScale



4. Variant-Level SKU Assignment

- SKUs are assigned **per Variant** (e.g. color, size, material)
- This assignment will be managed in a **different section** of the product form
- **Different sku variants** can be **selected to create variants** of bundles all under the same product.

5. Stock Display & CSV Export Considerations

- Stock and SKU data format:
 - For export: **avoid using commas** to separate SKU names or stock numbers
 - On UI display: use readable formatting
 - Example: **AUS-DR** & **L388I0DR** for SKUs
 - Use **;** for stock numbers
- Sorting:
 - Ensure **sortable logic** for columns like "Current Stock" when multiple values exist

6. SKU Performance Tracking (optional)

- Ability to track **SKU "success rate"** across different products it's used in (definition to be clarified)

2.3 Additional Features Initial High-Level Requirements as per Client Request

Client requested potential new features at the start of milestone 3. Initial details and high-level requirements are available under meeting minutes listed:

[MeetingMinutes_11/08/2025](#)

Essential Requirement:

Allow merchants to apply discounts to an entire product category instead of configuring product discounts individually.

- Example: Category "Chess Board" is 20% off will result in all chess board products having 20% off.



Client also proposed an advanced and valuable enhancement in the email following the second UAT testing — the ability to schedule when a discount is applied. The details are outlined in the email:

... “Side note - An advanced (and very useful feature) would be if it could actually be scheduled when the discount is applied. In that case - when a sale starts "tomorrow", the user doesn't have to stay awake till midnight to click the button... ”

Essential Requirement:

Allow merchants to configure the discount schedule in advance, eliminating the need for manual activation at the start of a sale period.

2.4 Additional Project Requirements as per Client Feedback on UAT (optional)

1. Access Product Bundle Manager from the product page

- Add the Product Bundle Manager option in the Product page actions (...)
 - Clicking the button would open up the manager on the page.

2. Bundle Indication

- A column on the product list page to indicate whether a product is a bundle (may not be technically possible in the current system)
- Alternatives: (either one)
 - Create a **parameter** to **filter products by bundle status**.
 - Use a **Product Category** as a **manual workaround** (but prone to errors).
 - **Automatic detection** of bundle status would be preferable over manual tagging.

3. SKU creation and appending

- **Bundle with variants:**
 - **High level**, eg. BUN-REX (no requirements)
 - Variant: "**SKU1 & SKU2**" & & **SKUn**"
 - Eg. EXP-S-95 & DIS-S-50
- **Bundle with no variants**
 - "**SKU1 & SKU2**" & & **SKUn**"
 - eg. oakch-small & oakbrd-sm & timer

4. Price Calculation

- **Automatic Price Calculation:**
 - **Bundle price** should be **calculated** as the **sum of its components**.
- **Override Option:**
 - Users must still have the option to **override** the **calculated price manually**.
- **Bundles with Variants:**
 - **Default product price** should **match** the **price of the first variant**.
- Similar to weight calculation



3. Other Requirements and Considerations

3.1 Non-Functional Requirements

- **Performance:** The plugin should not slow down store operations or impact page load times.
- **Scalability:** Support thousands of products and SKUs without performance degradation.
- **Security:** Ensure secure API communication and data handling.
- **Usability:** Provide a clean, simple, and responsive UI for efficient inventory management.
- **Compatibility:** Ensure the plugin is compatible with the existing BigCommerce product and any other plugins during integration.
- **Maintainability:** The development of the plugin should be documented well and easily modifiable for future modifications and/or additional features.

3.2 Technical Considerations

To ensure a smooth development process and seamless integration with the BigCommerce platform, the following technical considerations have been taken into account for the development of the SKU Management plugin for ChessWorld. We are presenting multiple options for the tech stack, along with a rationale for our preferred approach:

Tech Stack Options:

- **Option 1 (Recommended):** Node.js, React, Next.js, and BigDesign with MySQL
 - **Frontend:** React.js, Next.js, and BigDesign.
 - **Backend:** Node.js.
 - **Database:** MySQL.
- **Option 2:** Python, Flask, and a Relational Database (PostgreSQL)
 - **Frontend:** HTML.
 - **Backend:** Python (with the Flask framework).
 - **Database:** PostgreSQL or MySQL.
- **Option 3:** Laravel (PHP Framework) and React with MySQL
 - **Frontend:** React.js
 - **Backend:** Laravel (PHP framework).
 - **Database:** MySQL



Rationale and Comparison:

- **Option 1 (Recommended):**

- This stack is well-documented by BigCommerce, with readily available sample apps (<https://github.com/bigcommerce/sample-app-nodejs>) and guides. The sample-app-nodejs repository has recent commits, which indicates active maintenance, and the sample app includes NextJS, BigDesign, Typescript, and React.
- The BigCommerce quick start guide also uses this stack and we tried it and got it working locally.
- Node.js's non-blocking I/O model is well-suited for handling many concurrent requests, making it a good choice for scalable API development.
- BigDesign provides a pre-built UI component library to align with the BigCommerce design system.

- **Option 2:**

- BigCommerce provides a sample app in Python using Flask - (<https://github.com/bigcommerce/hello-world-app-python-flask>). This sample app uses SQLAlchemy to connect to the database.
- Python is known for its readability and ease of use.
- More backend-focused which could make sense for a plugin
- **The number of commits and its most recent commit are significantly less than option 1.**

- **Option 3:**

- BigCommerce has a sample app using Laravel with React - (<https://github.com/bigcommerce/laravel-react-sample-app>).
- Many of the other options can be used for the front end
- **Less community support and fewer BigCommerce specific examples.**

BigCommerce API Usage: Utilize REST and GraphQL APIs for product, order, and inventory synchronization.

- **REST**

- **Catalog API (Manages products, categories, brands, bulk pricing rules, and more.)** ← main focus
- Management API
- Payments API
- Webhooks

- **GraphQL**

- Storefront API
- Admin API
- Account API
- Storefront playground

Considerations:

- The team will need to familiarize themselves with the BigCommerce API documentation, the specifics of interacting with it using any of the specified frameworks, and adhere to BigCommerce's app development guidelines to ensure a secure and compliant plugin.
- Scalability should be considered when designing the database schema and API interactions, especially if the client anticipates more complex product bundles in the future.



4. Scope Negotiations

4.1 Initial Scope vs Negotiated Scope

The table below detailed the changes between the initial requirements and scope collected during the first meetings and the negotiated requirements and scope over the whole project.

Initial Requirements and Scope (M1)	Negotiated Requirements and Scope (M2 - 4)
Bundle creation and management with independent SKU-level inventory tracking including variant-level SKUs.	Bundle creation and management with independent SKU-level inventory tracking including variant-level SKUs, with bundle indication.
Optional automated calculation of bundle weight and price.	Automated calculation of bundle weight and price with override features.
Stock tracking and synchronisation with overselling prevention when a bundle is sold.	Stock tracking and synchronisation with overselling prevention for when a bundle is sold, refunded and partially refunded.
Seamless BigCommerce platform integration with features available directly in the native UI.	Seamless BigCommerce platform integration with accessible features without changes to the native UI.
Simple and intuitive UI.	Simple and intuitive UI.
	Feature to apply discounts for all products in a category at once, including scheduled discounts.



4.2 Other Scope Negotiations

Due to limitations with the BigCommerce platform and the plans chosen for the deployment platform during the development process, certain limitations caused scope negotiations. These are listed below:

- 1. Problem:** Cannot create bundles directly on the native UI due to platform limitations.
Negotiated Scope: Implemented side panel on products page to access the features with minimal complications.
- 2. Problem:** Performance during bundles creation.
Negotiated Scope: Not a huge issue since bundle creation will be rare and the performance issues do not affect the storefront.
- 3. Problem:** Race conditions/API may cause low performance in high volume periods
Negotiated Scope: Ensured store is self healing during low volume periods.
- 4. Problem:** The current version of Vercel (Free) does not allow for multiple cron jobs, causing difficulties in discount scheduling.
Negotiated Scope: Did without the automatic discount scheduling using multiple cron jobs.



4. User Stories

A user story is a short narrative that describes a feature the client wants the system to have. It typically follows this simple format:

As a <role>, I want to <desirable action/reaction/state> so that <benefit>

Besides following the format above, crafting a good user story should also adhere to the INVEST criteria—**Independent, Negotiable, Valuable, Estimable, Small, and Testable**, to avoid implementation issues in future sprints.

We will convert the list of requirements into user stories using the INVEST criteria, as shown in Table 1 below.

No	Story Points	User Story	User
Feature			
US01	13	As a shop retailer, I want to track each SKU in a bundle independently so that I can maintain accurate inventory counts for individual products.	Shop Retailer
US02	11	As a shop retailer, I want to create product bundles made up of multiple SKUs so that I can offer customised product sets to customers.	Shop Retailer
US03	11	As a shop retailer, I want to have a system to update SKU inventory levels automatically when a product is sold, either individually or as part of a bundle so that stock data remains consistent and up to date.	Shop Retailer
US04	7	As a shop retailer, I want to have bundle sales to be restricted if any component in the bundle is out of stock so that I can avoid overselling available products.	Shop Retailer
US05	17	As a shop retailer, I want to have a plugin that can integrate with BigCommerce's product and order management system so that I can manage all inventory and orders within a unified platform.	Shop Retailer
US06	5	As a shop retailer, I want to be able to view the SKUs section displayed in the left navigation panel so that I can quickly browse and access inventory items.	Shop Retailer
US07	11	As a shop retailer, I want a search-and-select interface to add SKUs when creating bundles so that I can efficiently build product sets.	Shop Retailer
US08	7	As a shop retailer, I want to view a real-time inventory dashboard showing stock levels for individual and bundled SKUs so that I can make informed restocking decisions.	Shop Retailer
NFR			
US09	2	As a shop retailer, I want the design of the plugin to be simple for managing bundles and linking SKUs so that I can easily configure and update product bundle details.	Shop Retailer
US10	17	As a developer, I want the plugin to handle thousands of products and SKUs so that the store can grow without performance issues.	Developer



US11	13	As a developer, I want the plugin to run efficiently without affecting page load times so that my customers have a seamless shopping experience.	Developer
US12	5	As a developer, I want to construct the plugin code to be modular and well-documented, so that it is easy to maintain and update in the future.	Developer
US13	7	As a developer, I want to use secure API communication and data handling when building the plugin to handle thousands of products and SKUs so that the store can grow without performance issues.	Developer

Table 1. User Stories BigCommerce Plugin

Table 2 below has been updated with the latest requirements based on client feedback on the UI mockup.

No	Story Points	User Story	User
Feature			
Requirement Sets 1			
US01	13	As a shop retailer, I want to track each SKU in a bundle independently, so that I can maintain accurate inventory counts for individual products.	Shop Retailer
US02	11	As a shop retailer, I want to create product bundles made up of multiple SKUs, so that I can offer customised product sets to customers.	Shop Retailer
US03	11	As a shop retailer, I want to have a system to update SKU inventory levels automatically when a product is sold, either individually or as part of a bundle, so that stock data remains consistent and up to date.	Shop Retailer
US04	7	As a shop retailer, I want to have a system that restricts bundle sales if any component in the bundle is out of stock, so that I can avoid overselling available products.	Shop Retailer
US05	17	As a shop retailer, I want to have a plugin that can integrate with BigCommerce's product and order management system, so that I can manage all inventory and orders within a unified platform.	Shop Retailer
US06	5	As a shop retailer, I want to be able to view the SKUs section displayed in the left navigation panel, so that I can quickly browse and access inventory items.	Shop Retailer
US07	11	As a shop retailer, I want to search and select SKUs directly from the product edit form using a simple search field, so I can quickly assign SKUs without leaving the page.	Shop Retailer
US08	7	As a shop retailer, I want to view a real-time inventory dashboard showing stock levels for individual and bundled SKUs so that I can make informed restocking decisions.	Shop Retailer
Requirement Sets 2			
US14	3	As a shop retailer, I want to create or edit SKU with the SKU number, SKU name, and stock level so that I can focus solely on SKU-related data.	Shop Retailer
US15	3	As a shop retailer, I want to have the selected SKUs to be displayed as comma-separated values, so that I can easily view all the bundle SKUs in a single line.	Shop Retailer
US16	5	As a shop retailer, I want to display the selected SKUs with a close button for	Shop Retailer



		removal, so that I can easily manage assigned SKUs on the product edit form.	
US17	13	As a shop retailer, I want to assign different SKUs to product variants, so that I can accurately track the stock of each SKU.	Shop Retailer
US18	11	As a shop retailer, I want to display SKU in a readable format on the product page, so that I can understand SKU groupings at a glance.	Shop Retailer
US19	11	As a shop retailer, I want to view stock numbers in semicolon-separated format on the product page, so that I can clearly distinguish and interpret multiple stock values.	Shop Retailer
US20	17	As a shop retailer, I want to export stock numbers and SKU names without using commas between values, so that I can export CSVs without breaking the format.	Shop Retailer
US21	13	As a shop retailer, I want to sort SKU listings by current stock levels when multiple values exist, so that I can identify low-stock SKUs more easily.	Shop Retailer
US26	13	As a shop retailer, I want to select different SKU variants to each bundle variant, so that I can easily track stock per bundle variation.	Shop Retailer
Optional Requirements			
US22	7	As a shop retailer, I want to add weight, price, and custom fields when creating a SKU, so that I can track more detailed inventory attributes if needed.	Shop Retailer
US23	11	As a shop retailer, I want to define custom fields and assign SKUs to custom SKU categories, so that I can extend SKU details beyond product categories.	Shop Retailer
US24	13	As a shop retailer, I want to view aggregated SKU values, so that I can estimate product metrics.	Shop Retailer
US25	17	As a shop retailer, I want to track SKU's success rate across different products, so that I can identify high-performing inventory components.	Shop Retailer
NFR			
US09	2	As a shop retailer, I want the design of the plugin to be simple for managing bundles and linking inventory codes so that I can easily configure and update product bundle details.	Shop Retailer
US10	17	As a developer, I want the plugin to handle thousands of products and SKUs so that the store can grow without performance issues.	Developer
US11	13	As a developer, I want the plugin to run efficiently without affecting page load times so that my customers have a seamless shopping experience.	Developer
US12	5	As a developer, I want to construct the plugin code to be modular and well-documented, so that it is easy to maintain and update in the future.	Developer

Table 2. Revised User Stories BigCommerce Plugin

Additionally, the following link is the user stories spreadsheet that is presented in more detail and organised.

User Stories Link:

[📄 BigCommercePlugin - User Stories - M2](#)

Table 3 below has been updated based on the latest client requirements and feedback received during Milestone 3.

No	Story Points	User Story	User
Feature			
Requirement Sets 1			
US01	13	As a shop retailer, I want to track each SKU in a bundle independently, so that I can maintain accurate inventory counts for individual products.	Shop Retailer
US02	11	As a shop retailer, I want to create product bundles made up of multiple SKUs, so that I can offer customised product sets to customers.	Shop Retailer
US03	11	As a shop retailer, I want to have a system to update SKU inventory levels automatically when a product is sold, either individually or as part of a bundle, so that stock data remains consistent and up to date.	Shop Retailer
US04	7	As a shop retailer, I want to have a system that restricts bundle sales if any component in the bundle is out of stock, so that I can avoid overselling available products.	Shop Retailer
US05	17	As a shop retailer, I want to have a plugin that can integrate with BigCommerce's product and order management system, so that I can manage all inventory and orders within a unified platform.	Shop Retailer
US06	5	As a shop retailer, I want to be able to view the SKUs section displayed in the left navigation panel, so that I can quickly browse and access inventory items.	Shop Retailer
US07	11	As a shop retailer, I want to search and select SKUs directly from the product edit form using a simple search field, so I can quickly assign SKUs without leaving the page.	Shop Retailer
US08	7	As a shop retailer, I want to view a real-time inventory dashboard showing stock levels for individual and bundled SKUs so that I can make informed restocking decisions.	Shop Retailer
Requirement Sets 2			
US14	3	As a shop retailer, I want to create or edit SKU with the SKU number, SKU name, and stock level so that I can focus solely on SKU-related data.	Shop Retailer
US15	3	As a shop retailer, I want to have the selected SKUs to be displayed as comma-separated values, so that I can easily view all the bundle SKUs in a single line.	Shop Retailer
US16	5	As a shop retailer, I want to display the selected SKUs with a close button for removal, so that I can easily manage assigned SKUs on the product edit form.	Shop Retailer
US17	13	As a shop retailer, I want to assign different SKUs to product variants, so that I can accurately track the stock of each SKU.	Shop Retailer
US18	11	As a shop retailer, I want to display SKU in a readable format on the product page, so that I can understand SKU groupings at a glance.	Shop Retailer
US19	11	As a shop retailer, I want to view stock numbers in semicolon-separated format on the product page, so that I can clearly distinguish and interpret multiple stock values.	Shop Retailer
US20	17	As a shop retailer, I want to export stock numbers and SKU names without using commas between values, so that I can export CSVs without breaking the format.	Shop Retailer
US21	13	As a shop retailer, I want to sort SKU listings by current stock levels when multiple values exist, so that I can identify low-stock SKUs more easily.	Shop Retailer
US26	13	As a shop retailer, I want to select different SKU variants for each bundle variant, so that I can easily track stock per bundle variation.	Shop Retailer



Requirement Sets 3			
US27	11	As a shop retailer, I want to apply a discount to an entire product category so that I can save time and avoid configuring discounts individually for each product.	Shop Retailer
US28	7	As a shop retailer, I want to remove or edit a category-wide discount so that I can adjust promotions based on business needs.	Shop Retailer
US29	3	As a shop retailer, I want to view the discounted price for each product category along with its status so that I can verify which categories have discounts applied and ensure pricing accuracy.	Shop Retailer
Requirement Sets 4			
US33	13	As a shop retailer, I want to schedule when a discount is applied so that sales can automatically start and end at specific times without manual intervention.	Shop Retailer
Optional Requirements			
US22	7	As a shop retailer, I want to add weight, price, and custom fields when creating a SKU, so that I can track more detailed inventory attributes if needed.	Shop Retailer
US23	11	As a shop retailer, I want to define custom fields and assign SKUs to custom SKU categories, so that I can extend SKU details beyond product categories.	Shop Retailer
US24	13	As a shop retailer, I want to calculate aggregated SKU values automatically based on its components, so that I can save time and reduce errors, while retaining the option to override values manually.	Shop Retailer
US25	17	As a shop retailer, I want to track SKU's success rate across different products, so that I can identify high-performing inventory components.	Shop Retailer
US30	11	As a shop retailer, I want to have a Product Bundle Manager option available in the product page actions menu, so that I can quickly access bundle management features directly from the product page.	Shop Retailer
US31	11	As a shop retailer, I want a clear indication on the product list page of whether a product is a bundle, so that I can easily identify bundles without relying on manual workarounds or separate filtering methods.	Shop Retailer
US32	7	As a shop retailer, I want to create SKUs for bundles using a clear and structured naming format, so that I can easily identify and manage bundled products, with or without variants.	Shop Retailer
NFR			
US09	2	As a shop retailer, I want the design of the plugin to be simple for managing bundles and linking inventory codes so that I can easily configure and update product bundle details.	Shop Retailer
US10	17	As a developer, I want the plugin to handle thousands of products and SKUs so that the store can grow without performance issues.	Developer
US11	13	As a developer, I want the plugin to run efficiently without affecting page load times so that my customers have a seamless shopping experience.	Developer
US12	5	As a developer, I want to construct the plugin code to be modular and well-documented, so that it is easy to maintain and update in the future.	Developer
US13	7	As a developer, I want to use secure API communication and data handling when building the plugin to handle thousands of products and SKUs so that the store can grow without performance issues.	Developer

Table 3. Revised User Stories BigCommerce Plugin



Additionally, the following link is the user stories spreadsheet that is presented in more detail and organised.

User Stories Link:

[+ BigCommercePlugin - User Stories M3](#)

Below is the latest version of the user stories, aligned with the client's expectations, along with the link to the spreadsheet.

No.	Story Points	User Story	User
Feature			
Requirement Sets 1			
US01	13	As a shop retailer, I want to track each SKU in a bundle independently, so that I can maintain accurate inventory counts for individual products.	Shop Retailer
US02	11	As a shop retailer, I want to create product bundles made up of multiple SKUs, so that I can offer customised product sets to customers.	Shop Retailer
US36	11	As a shop retailer, I want to indicate the created product is bundle and each SKU stock level directly in the side panel, so that I can define bundle information efficiently.	Shop Retailer
US03	11	As a shop retailer, I want to have a system to update SKU inventory levels automatically when a product is sold, either individually or as part of a bundle, so that stock data remains consistent and up to date.	Shop Retailer
US04	7	As a shop retailer, I want to have a system that restricts bundle sales if any component in the bundle is out of stock, so that I can avoid overselling available products	Shop Retailer
US05	17	As a shop retailer, I want to have a plugin that can integrate with BigCommerce's product and order management system, so that I can manage all inventory and orders within a unified platform.	Shop Retailer
US06	5	As a shop retailer, I want to be able to view the SKUs section, so that I can quickly browse and access inventory items.	Shop Retailer
US07	11	As a shop retailer, I want to search and select SKUs directly from the product edit form using a simple search field, so I can quickly assign SKUs without leaving the page.	Shop Retailer
US08	7	As a shop retailer, I want to view a real-time inventory dashboard showing stock levels for individual and bundled SKUs so that I can make informed restocking decisions.	Shop Retailer
Requirement Sets 2			
US14	3	As a shop retailer, I want to create or edit SKU with the SKU number, SKU name, and stock level so that I can focus solely on SKU-related data.	Shop Retailer
US15	3	As a shop retailer, I want to have the selected SKUs to be displayed as comma-separated values, so that I can easily view all the bundle SKUs in a single line.	Shop Retailer
US16	5	As a shop retailer, I want to display the selected SKUs with a close button for removal, so that I can easily manage assigned SKUs on the product edit form.	Shop Retailer
US17	13	As a shop retailer, I want to assign different SKUs to product variants, so that I can accurately track the stock of each SKU.	Shop Retailer
US18	11	As a shop retailer, I want to display SKU in a readable format on the product page, so that I can understand SKU groupings at a glance.	Shop Retailer

US19	11	As a shop retailer, I want to view stock numbers in semicolon-separated format on the product page, so that I can clearly distinguish and interpret multiple stock values.	Shop Retailer
US21	13	As a shop retailer, I want to sort SKU listings by current stock levels when multiple values exist, so that I can identify low-stock SKUs more easily.	Shop Retailer
US26	13	As a shop retailer, I want to select different SKU variants for each bundle variant, so that I can easily track stock per bundle variation.	Shop Retailer
Requirement Sets 3			
US27	11	As a shop retailer, I want to apply a discount to an entire product category so that I can save time and avoid configuring discounts individually for each product.	Shop Retailer
US28	7	As a shop retailer, I want to edit or remove a category-wide discount so that I can adjust promotions based on business needs.	Shop Retailer
US29	3	As a shop retailer, I want to view the discounted type for each product category along with its status so that I can verify which categories have discounts applied and ensure pricing accuracy.	Shop Retailer
Requirement Sets 4			
US33	13	As a shop retailer, I want to schedule when a discount is applied so that sales can automatically start and end at specific times without manual intervention.	Shop Retailer
US34	11	As a shop retailer, I want have a system to automatically restore SKU inventory levels when a product or bundle is fully refunded, so that inventory accurately reflects available stock.	Shop Retailer
US35	13	As a shop retailer, I want to have a system to partially restore SKU inventory levels when a product or bundle is partially refunded, so that only refunded items are replenished.	Shop Retailer
Optional Requirements			
US20	17	As a shop retailer, I want to export stock numbers and SKU names without using commas between values, so that I can export CSVs without breaking the format.	Shop Retailer
US22	7	As a shop retailer, I want to add weight, price, and custom fields when creating a SKU, so that I can track more detailed inventory attributes if needed.	Shop Retailer
US23	11	As a shop retailer, I want to organise SKUs into their own category structure so that I can manage and analyse stock more effectively, independent of existing product categories.	Shop Retailer
US24	13	As a shop retailer, I want to calculate aggregated SKU values automatically based on its components, so that I can save time and reduce errors, while retaining the option to override values manually.	Shop Retailer
US25	17	As a shop retailer, I want to track SKU's success rate across different products, so that I can identify high-performing inventory components.	Shop Retailer
US30	11	As a shop retailer, I want to have a Product Bundle Manager option available in the product page actions menu, so that I can quickly access bundle management features directly from the product page.	Shop Retailer
US31	11	As a shop retailer, I want a clear indication on the product list page of whether a product is a bundle, so that I can easily identify bundles without relying on manual workarounds or separate filtering methods.	Shop Retailer
US32	7	As a shop retailer, I want to create SKUs for bundles using a clear and structured	Shop Retailer



		naming format, so that I can easily identify and manage bundled products, with or without variants.	
NFR			
US09	2	As a shop retailer, I want the design of the plugin to be simple for managing bundles and linking inventory codes so that I can easily configure and update product bundle details.	Shop Retailer
US10	17	As a developer, I want the plugin to handle thousands of products and SKUs so that the store can grow without performance issues.	Developer

Table 4. BigCommerce Plugin: Latest Version of User Stories

Latest User Stories Link:

[BigCommercePlugin - User Stories](#)

5. Use Cases

In this section, the use case diagram below represents the core functionalities of the BigCommerce Plugin, derived from all the elicited requirements provided by the client throughout the project.

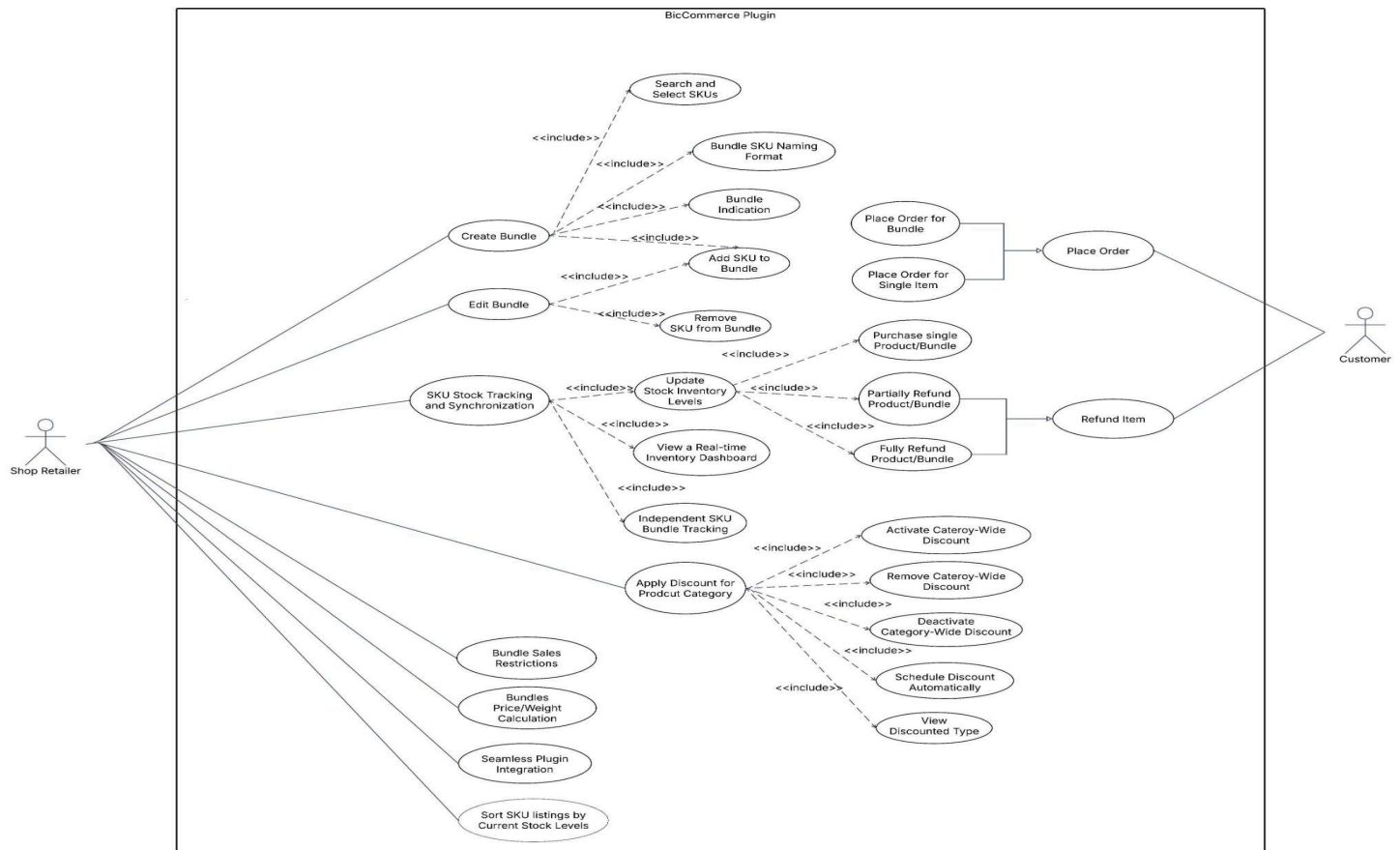


Figure 1: Use Case Diagram of BigCommerce Plugin