Disclaimer : Dependencies are left as empty as they might be a subject to modification later on in the project :

UC Name	*ADD NEW BOOK*
Summary	*Enter a new book information into the database, making it available for sale or further management*
Dependency	
Actors	*Primary Actor – Bookstore Manager* *Secondary Actor – Bookstore Database*
Preconditions	*- The system is running accordingly* *- Database should be functional* *-The user has logged in into the system* *-The user needs to have the necessary permissions or roles*
Description of the Main Sequence	● *Step 1: Manager clicks on "Add New Book" option* ● *Step 2: Manager will be shown a window containing data fields relevant to the book information which he is obliged to fill up to successfully add a book* ● *Step 3: System validates the information if there are any errors or inconsistencies* ● *Step 4 : Manager confirms the action after reviewing it* ● *Step 5 : The System adds the book into the database and gives feedback to the user if the action has been successful through a popup message*
Description of the Alternative Sequence	*Can be any of the following : Invalid Input, System Error, Duplicate Book Entry* ● *Step 1 : Step 1: Manager clicks on "Add New Book" option* ● *Step 2: Manager will be shown a window containing data fields relevant to the book information which he is obliged to fill up to successfully add a book* ● *Step 3: System validates the information if there are any errors or inconsistencies* ● *Step 4 : System warns the Manager with a pop up message about the alternative outcome and hints at the problem/problems*

Non functional requirements	*Response Time: The system should add the new book to the database and confirm the action to the user within a reasonable time*
	Access Control: Only authenticated and authorized users (like administrators or managers) should be able to add new books to the system
	User Interface: The interface for adding a new book should be intuitive and easy to use. It should provide clear fields for entering book information like title, author, ISBN, price, etc.
	Modifiability: It should be easy to modify the system to add new fields for book information or to change the way book information is handled
	Security: The system should ensure that only authorized users can add books. It should also protect sensitive information such as book supplier details.
Postconditions	*Book has been added to the system and it is available for costumers to purchase. Database has been updated with this new book and it's information*

UC Name	*REMOVE BOOK*
Summary	*Removing an existing Book in the E-showcase and database*
Dependency	
Actors	*Primary Actor – Bookstore Manager* *Secondary Actor – Bookstore Database*
Preconditions	*- The system is running accordingly* *- Database should be functional* *-The user has logged in into the system* *-The user needs to have the necessary permissions or roles*
Description of the Main Sequence	● *Step 1: Manager clicks on "Remove Book" option* ● *Step 2: Manager will be shown a popup window asking to Insert the book name, author and publisher* ● *Step 3 : Manager confirms the action after reviewing the input* ● *Step 4 : The System removes the book from the database and gives feedback to the user if the action has been successful through a popup message*
Description of the Alternative Sequence	*Can be any of the following : Invalid Input, System Error,* ● *Step 1 : Step 1: Manager clicks on "Remove Book" option* ● *Step 2: Manager will be shown a popup window asking to Insert the book name, author and publisher* ● *Step 3 : Manager confirms the action after reviewing the input* ● *Step 4: System validates the information if there are any errors or inconsistencies* ● *Step 5 : System warns the Manager with a pop up message about the alternative outcome and hints at the problem/problems*

Non functional requirements	*Response Time: The system should add the new book to the database and confirm the action to the user within a reasonable time* *Access Control: Only authenticated and authorized users (like administrators or managers) should be able to remove books from the system* *User Interface: The interface for removing books should be intuitive and easy to use. It should provide clear fields for entering book information like title, author and publisher.* *Modifiability: It should be easy to modify the system to add new fields for the information required to remove a book* *Security: The system should ensure that only authorized users can remove books*
Postconditions	*Book has been removed the system and it is not available for costumers to purchase anymore or to be found through search . Database has been updated and this book is no longer in it.*

UC Name	EDIT BOOK CHARACTERISTICS
Summary	*User is able to edit the characteristics or details of a book in the bookstore's inventory/database.*
Dependency	
Actors	*Primary Actor – Bookstore Manager* *Secondary Actor – Bookstore Database*
Preconditions	*- The system is running accordingly* *- Database should be functional* *-The user has logged in into the system* *-The user needs to have the necessary permissions or roles*
Description of the Main Sequence	● *Step 1: Manager clicks on "Edit" option* ● *Step 2: Manager will be shown a window containing data fields of the book information which he can edit to his need* ● *Step 3: System validates the information if there are any errors or inconsistencies* ● *Step 4 : Manager confirms the action after reviewing it* ● *Step 5 : The System saves the book into the database with the new edited information and gives feedback to the user if the action has been successful through a popup message*
Description of the Alternative Sequence	*Can be any of the following : Invalid Input, System Error* ● *Step 1: Manager clicks on "Edit" option* ● *Step 2: Manager will be shown a window containing data fields of the book information which he can edit to his need* ● *Step 3: System validates the information if there are any errors or inconsistencies* ● *Step 4 : Manager confirms the action after reviewing it* ● *Step 5 : System warns the Manager with a pop up message about the alternative outcome and hints at*

	the problem/problems
Non functional requirements	*Response Time: The system should edit information of the database and confirm the action to the user within a reasonable time* *Access Control: Only authenticated and authorized users (like administrators or managers) should be able to remove books from the system* *User Interface: The interface for editing books information should be intuitive and easy to use. It should provide clear fields for editing book information* *Security: The system should ensure that only authorized users can edit book characteristics*
Postconditions	*Book information has been edited the system and it is available for costumers to purchase with updated information. Database has been updated.*

UC Name	*CHECK BOOK STATISTICS*
Summary	*User is shown the statistics of a book in the bookstore's inventory/database.*
Dependency	
Actors	*Primary Actor – Bookstore Manager/Admin* *Secondary Actor – Bookstore Database*
Preconditions	*- The system is running accordingly* *- Database should be functional* *-The user has logged in into the system* *-The user needs to have the necessary permissions or roles*
Description of the Main Sequence	● *Step 1: Manager navigates to Statistics page* ● *Step 2: Manager will be shown a list of books which he can right click, press "view statistic"* ● *Step 3: Manager is redirected to a page containing all the statistics of the book, including revenue generated, sales etc*
Description of the Alternative Sequence	*System Error or Database Error* ● *Step 1: Manager navigates to Statistics page* ● *Step 2: Manager will be shown a list of books which he can right click, press "view statistic"* ● *Step 3: Manager is redirected to a page containing all the statistics of the book, including revenue generated, sales etc otherwise if there's a system error or Database error a window pops up informing the user with the current problem*
Non functional requirements	*Response Time: Response time should be reasonable depending on the book and the information* *Access Control: Only authenticated and authorized users (like administrators or managers) should be able to remove*

	books from the system *User Interface: Interface should help the manager understand the statistics with the help of graphs, tables, charts etc.*
Postconditions	*Statistics page works as intended*

UC Name	*CHECK WAREHOUSE INVENTORY*
Summary	*User can navigate through the Inventory of the bookstore*
Dependency	
Actors	*Primary Actor – Bookstore Manager/Admin* *Secondary Actor – Bookstore Database*
Preconditions	*- The system is running accordingly* *- Database should be functional* *-The user has logged in into the system* *-The user needs to have the necessary permissions or roles*
Description of the Main Sequence	● *Step 1: Manager navigates to Inventory page* ● *Step 2: Manager will be shown a list of products with them being in an ascending order based on stock* ● *Step 3: Every time a product is at a "critical" stock level the user shall be notified*
Description of the Alternative Sequence	*System Error or Database Error* ● *Step 1: Manager navigates to Inventory page* ● *Step 2: Error message pops up with the corrensponding error code (database or system error)* ● *Step 3: Error is filed and sent to the maintenance team*
Non functional requirements	*Response Time: Response time should be reasonable depending on the amount of products* *Access Control: Only authenticated and authorized users (like administrators or managers) should be able to remove books from the system* *User Interface: Interface should help the manager understand the statistics with the help of graphs, tables,*

	charts etc.
Postconditions	*Inventory page works as intended*