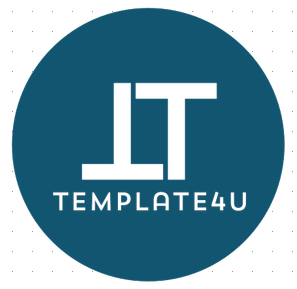
**Template4u LTD**

****

**Company description**

Our company will sell website templates.

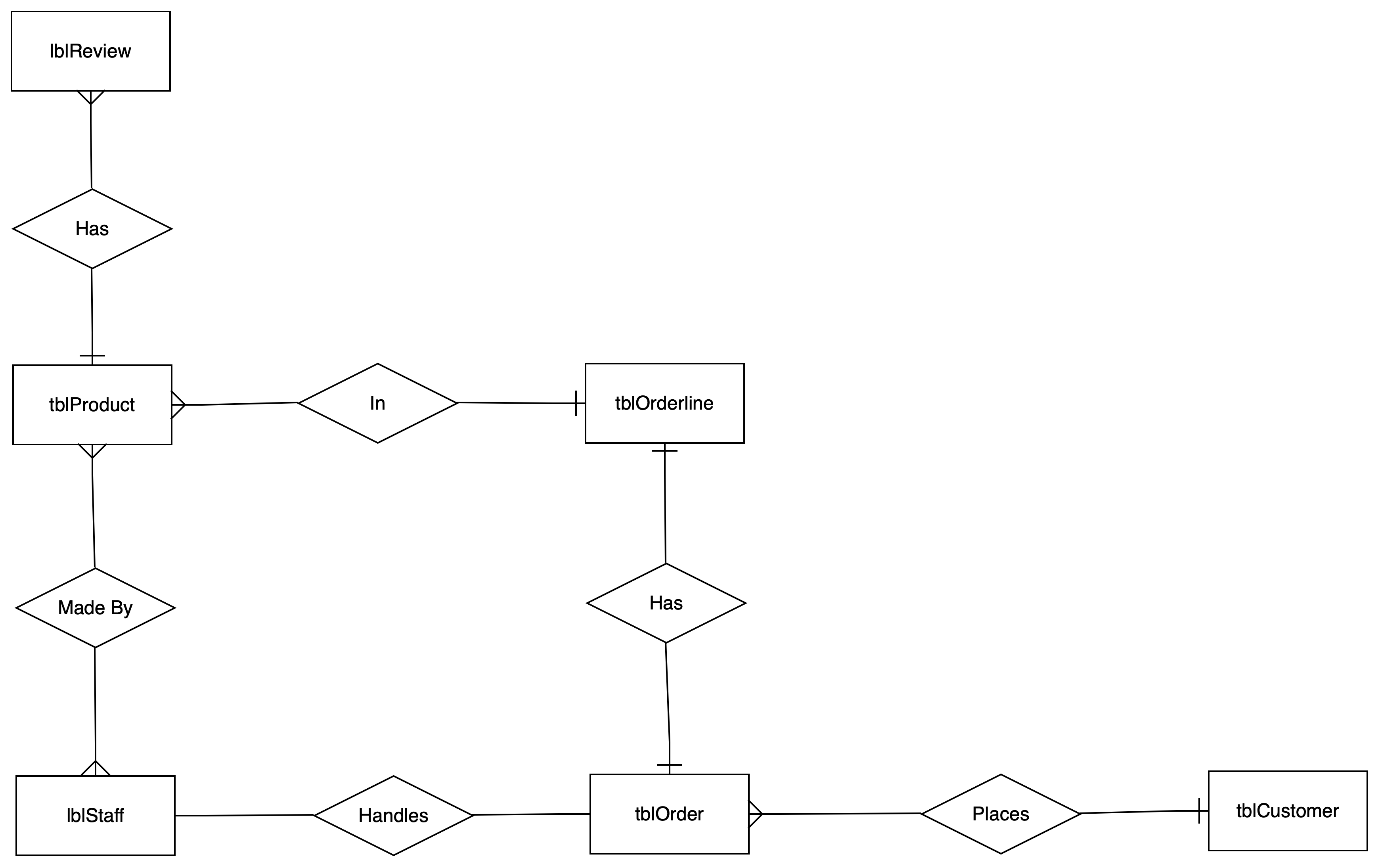
|  |  |
| --- | --- |
| Order | Jakub Zarzycki |
| Staff | Alex Longhurst |
| Customers | Patryk Wegrzynski |
| Products list | Dawid Malecki |
| Template reviews | Dominik Michalski |

**System Overview**

This company will sell website templates to the general public. They can order an already created design which they can use to form their own webpage. They can browse and filter this available collection of webpage designs. There will be an option for customers to add their web page templates to a shopping cart and create an order. After having made an order, the customer can leave a review for the template. They will then be required to create a customer account where they specify payment and personal details. They will also be able to review and validate the orders in their shopping cart.

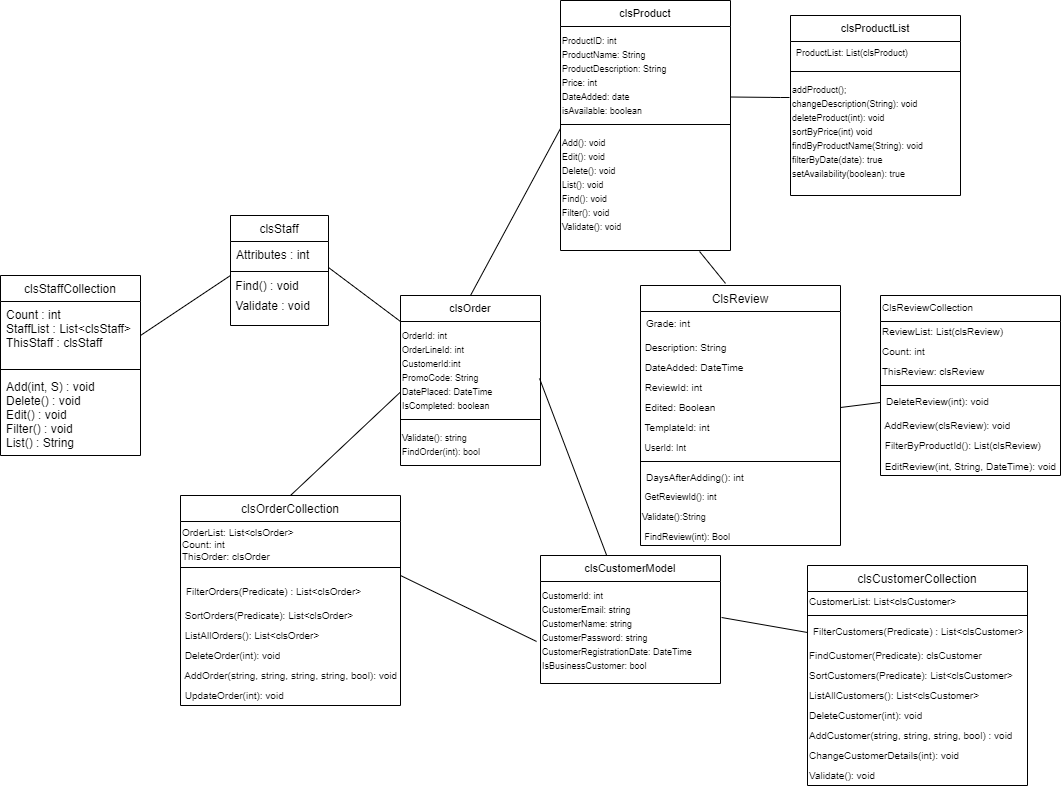
Once an order is created by the customer it will be added to a list of orders for processing at the back end. The back end will allow for a range of admin functions such as managing customer data, staff data, stock and running various reports on data stored in the system e.g. orders placed.

**ERD**

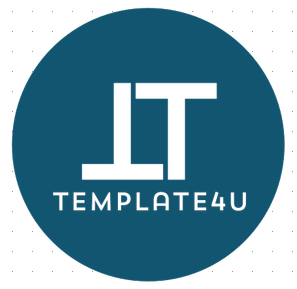


**Creates**

**Class Diagram**



**Template4u LTD**

****

**Allocated Component:** Template reviews

**Developer:** Dominik Michalski

**Overview:**

After template choice, purchase and implementation customers have a chance to rate chosen product. Review will consist of star-based rate (1-5 stars) and comment which is optional. Customers can add a review and edit it. However, when they once added a review, they cannot delete it. To prevent bad reviews caused by circumstances that happened after long time period, editing should be restricted - it will be possible only two weeks after it has been added. It will be marked as edited as well. App is to display list of templates with their user satisfaction rate (1.0-5.0) which is a mean of all reviews. When template is clicked it shows another list consists of usernames together with their reviews. On the back-end staff can delete any review that is not consistent with company policies. Staff cannot edit or add any reviews.

**Individual Table Designs**

tblReview

|  |  |
| --- | --- |
| ReviewID | Int Primary Key |
| Grade | Int not NULL |
| ReviewComment | Text |
| DateAdded | Date |
| UserID | varchar(20) Foreign Key not NULL |
| Edited | Boolean |
| ProductId | int Foreign key not NULL |

Use Cases:

Component – Review

Author – Dominik

|  |  |
| --- | --- |
| Use Case Name | Insert Review |
| Use Case Description | User can add a review, it must contain grade and product id |
| Use Case Author | Dominik |
| Actor(s) | Users |
| Locations | Mainly back end |
| Primary pathway | User adds review using web form, it appears in the database to be validated by administrator |
| Alternate pathways | User adds description as well |
| Exception pathways | User doesn’t specify which product has been reviewed, or product hasn’t been bought by this customer so it cannot be reviewed, database connection failure is possible as well |

|  |  |
| --- | --- |
| Use Case Name | Edit Review |
| Use Case Description | User can edit added review if it’s not too late (2 weeks after adding) |
| Use Case Author | Dominik |
| Actor(s) | Users |
| Locations | Mainly back end |
| Primary pathway | User edits a review using web form and specifying new grade |
| Alternate pathways | User edits also a description |
| Exception pathways | It has been more than 2 weeks since review was added,  database connection failure is possible as well |

|  |  |
| --- | --- |
| Use Case Name | View reviews |
| Use Case Description | User can view reviews of chosen template |
| Use Case Author | Dominik |
| Actor(s) | Users |
| Locations | Front end (grid with reviews) with back end (Database) |
| Primary pathway | User specifies of which template review is to be viewed, then grid with all reviews appears |
| Alternate pathways | -------------------------------------- |
| Exception pathways | There are no reviews for given template, database failure |

|  |  |
| --- | --- |
| Use Case Name | Sort reviews |
| Use Case Description | User can sort reviews |
| Use Case Author | Dominik |
| Actor(s) | Users |
| Locations | Back End (selecting from database) and front end (view in app) |
| Primary pathway | User chooses in which way they want to sort selected reviews (date, grade) |
| Alternate pathways | -------------------------------------- |
| Exception pathways | Database failure |

|  |  |
| --- | --- |
| Use Case Name | Validate review |
| Use Case Description | Administrators must check the review before adding if it is consistent with law and company policy |
| Use Case Author | Dominik |
| Actor(s) | Administrator |
| Locations | Backend |
| Primary pathway | Review is accepted as it is consistent with the policy |
| Alternate pathways | Review is rejected as it breaches company policy or, in general, law |
| Exception pathways | Database failure |

|  |  |
| --- | --- |
| Use Case Name | Find Review |
| Use Case Description | User can find review using its Id |
| Use Case Author | Dominik |
| Actor(s) | Users |
| Locations | Backend |
| Primary pathway | Requested review appears on the screen |
| Alternate pathways | -------------------------------------- |
| Exception pathways | Database failure, ID does not exist |

**Individual Use case diagram**

<<include>>

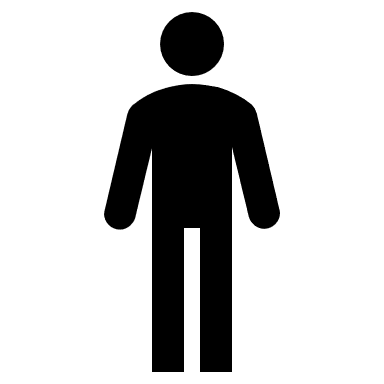
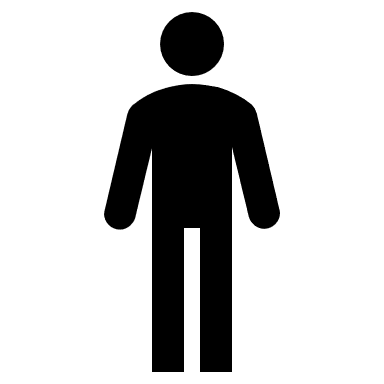
Validate Review

Add Review

<<include>>

<<include>>

Find Review



<<include>>

Edit Review

Delete Review Review

**User**

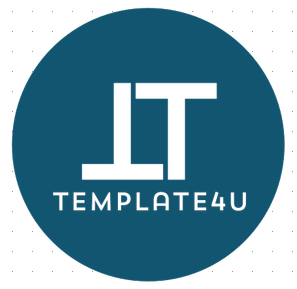
**Administrator**

Filter Reviews

<<extend>>

List Reviews

**Template4u LTD**

****

**Allocated Component:** Customer

**Developer:** Patryk Wegrzynski

**Overview:**

A customer visits the website to buy the templates for the site. The system differentiates if the customer is a business or private customer. To become a customer, the user first has to sign up for the web service. During the signup process, the customer has to provide the website with some data within the form, such as user credentials, address, and check if the user is signing up for business or private purposes. Once the user signs up for the website, the templates are available to add to the basket and be checked out. User can change their data or close the account if needed.

On the back end, the customer system is responsible for associating the basket with a specific user to finalise the order correctly.

Customer must be added to the database to continue to the website. Still, first, the customer credentials have to be validated through the system. Once the customer is in the database, the account can be edited or deleted if necessary. During the registration process user sets the account type as a business or just standard account. Based on that info users can be filtered and special promotions can be added to the business or standard clients.

**Database name:** Template4u

**Table name:** Customer

**Database design:**

|  |  |
| --- | --- |
| **Columns** | **Properties** |
| CustomerId | Int Primary Key |
| CustomerEmail | VARCHAR(50) |
| CustomerPassword | VARCHAR(50) |
| CustomerName | VARCHAR(50) |
| CustomerRegistrationDate | DateTime |
| IsCustomerBussinessClient | Boolean |

**Use cases:**

Component – Customer

Author – Patryk

|  |  |
| --- | --- |
| Use Case Name (Short two- or three-word name) | Add Customer |
| Use Case Description (Short description) | Adding customer to the system database |
| Use Case Author(s) (Who wrote this) | Patryk |
| Actor(s) (Who does this) | Customer / Administrator |
| Locations (Where does this happen) | Class lib functionality |
| Primary pathway (What is the normal “happy path” for this use case?) | Add user to database |
| Alternate pathways (What other paths are there that are not the “happy path”?) | The user cannot be added to the system – an error message occurs. |
| Exception pathways (What could possibly go wrong?) | Database connection fails.  User already exists in the database. |

|  |  |
| --- | --- |
| Use Case Name (Short two- or three-word name) | Update Customer |
| Use Case Description (Short description) | Updating user related data in the database |
| Use Case Author(s) (Who wrote this) | Patryk |
| Actor(s) (Who does this) | Customer / Administrator |
| Locations (Where does this happen) | Class lib functionality |
| Primary pathway (What is the normal “happy path” for this use case?) | User data successfully updated in the database. |
| Alternate pathways (What other paths are there that are not the “happy path”?) | User does not exist in the database so it cannot be updated.  User tries to update account with incorrect data – error occurs. |
| Exception pathways (What could possibly go wrong?) | Database connection fails.  User does not exist in the database. |

|  |  |
| --- | --- |
| Use Case Name (Short two- or three-word name) | Delete Customer |
| Use Case Description (Short description) | Deletes user from the system database. |
| Use Case Author(s) (Who wrote this) | Patryk |
| Actor(s) (Who does this) | Customer / Administrator |
| Locations (Where does this happen) | Class lib functionality |
| Primary pathway (What is the normal “happy path” for this use case?) | User successfully deleted from the database. |
| Alternate pathways (What other paths are there that are not the “happy path”?) | User cannot be deleted as he does not exist in the database. |
| Exception pathways (What could possibly go wrong?) | Database connection fails.  User does not exist in the database. |

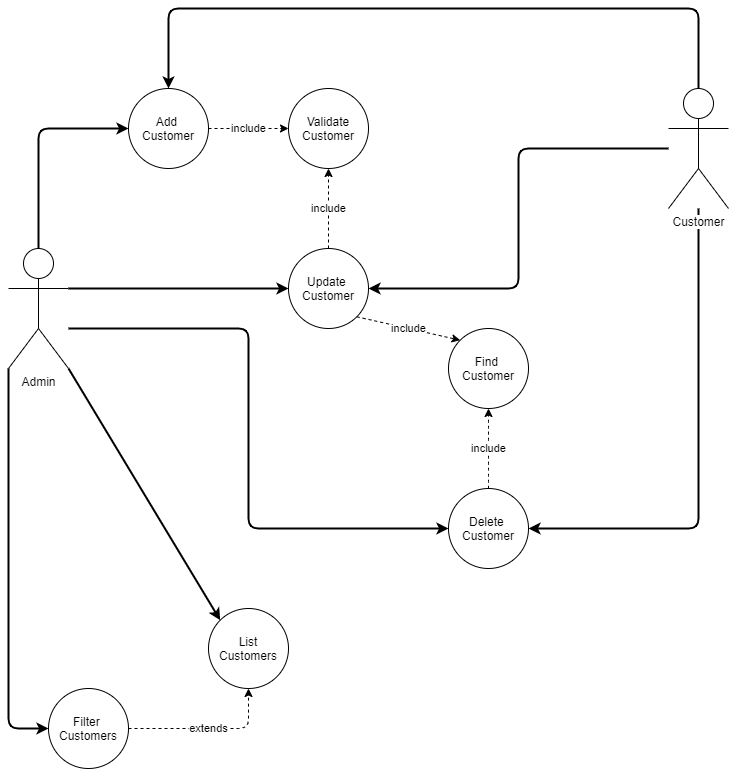
|  |  |
| --- | --- |
| Use Case Name (Short two- or three-word name) | Validate |
| Use Case Description (Short description) | Check if data provided is correct and can be added to the database. |
| Use Case Author(s) (Who wrote this) | Patryk |
| Actor(s) (Who does this) | Administrator |
| Locations (Where does this happen) | Class lib |
| Primary pathway (What is the normal “happy path” for this use case?) | Data is validated and correct. Data added to the database system. |
| Alternate pathways (What other paths are there that are not the “happy path”?) | Data is invalid and cannot be added to the database. |
| Exception pathways (What could possibly go wrong?) | Database connection fails.  Data is not provided in correct format; invalid data – error occurs |

|  |  |
| --- | --- |
| Use Case Name (Short two or three word name) | Find |
| Use Case Description (Short description) | Search for the user in the database. |
| Use Case Author(s) (Who wrote this) | Patryk |
| Actor(s) (Who does this) | Customer / Administrator |
| Locations (Where does this happen) | Front End / Class lib |
| Primary pathway (What is the normal “happy path” for this use case?) | User has been found. Show user data to the customer |
| Alternate pathways (What other paths are there that are not the “happy path”?) | User does not exist in the database. |
| Exception pathways (What could possibly go wrong?) | Database connection fails.  User does not exist in the database – error occurs |

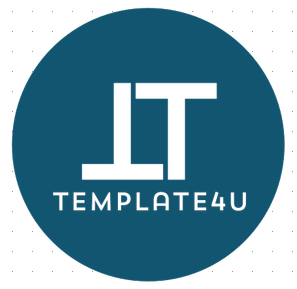
|  |  |
| --- | --- |
| Use Case Name (Short two- or three-word name) | Filter |
| Use Case Description (Short description) | Filter all user |
| Use Case Author(s) (Who wrote this) | Patryk |
| Actor(s) (Who does this) | Administrator |
| Locations (Where does this happen) | Front End / Class lib |
| Primary pathway (What is the normal “happy path” for this use case?) | List all filtered customer within the database. |
| Alternate pathways (What other paths are there that are not the “happy path”?) | There is no user within specified filter.  System show no users |
| Exception pathways (What could possibly go wrong?) | Database connection fails.  Users within specified filter does not exist – error occurs |

|  |  |
| --- | --- |
| Use Case Name (Short two- or three-word name) | List |
| Use Case Description (Short description) | Lists all users available in the system database. |
| Use Case Author(s) (Who wrote this) | Patryk |
| Actor(s) (Who does this) | Administrator |
| Locations (Where does this happen) | Front End / Class lib |
| Primary pathway (What is the normal “happy path” for this use case?) | System lists all customers. |
| Alternate pathways (What other paths are there that are not the “happy path”?) | There is no data available in the system. |
| Exception pathways (What could possibly go wrong?) | Database connection fails.  No users exist in the system – error occurs |

**Individual Use Case Diagram:**



**Template4u LTD**

****

**Template4u**

**Allocated Component:** Order

**Developer:** Jakub Zarzycki

**Overview:**

This component is created for the Administrator, so it is easier for him/her to handle orders. Administrator has the option to list all the orders that are currently in the database. He/She, can find a particular order using OrderId and can also update it or remove it. Moreover, Administrator is able to add new orders. He/She can filter and sort orders by a given predicate. This component is also available for user. User can add and edit his orders.

**Oder Table design:**

OrderId Int Primary Key

CustomerId Int Foreign key

OrderLineId Int Foreign key

PromoCode Text

DatePlaced Date

IsCompleted Boolean

**Use cases:**

|  |  |
| --- | --- |
| Use Case Name (Short two or three word name) | Add order |
| Use Case Description (Short description) | Administrator/user adds an order to the list. |
| Use Case Author(s) (Who wrote this) | Jakub |
| Actor(s) (Who does this) | Administrator/user |
| Locations (Where does this happen) | Front and back end functionality |
| Primary pathway (What is the normal “happy path” for this use case?) | Add order  Administrator/user adds an order to the database.  An order is added to the order collection. |
| Alternate pathways (What other paths are there that are not the “happy path”?) | Some required fields are missing- an error is displayed saying so.  Invalid data is entered and an error appears. |
| Exception pathways (What could possibly go wrong?) | Database connection fails  Error displayed advising of connection problem |

|  |  |
| --- | --- |
| Use Case Name (Short two or three word name) | List orders. |
| Use Case Description (Short description) | Administrator views a list of orders in the database. |
| Use Case Author(s) (Who wrote this) | Jakub |
| Actor(s) (Who does this) | Administrator |
| Locations (Where does this happen) | Front and back end functionality |
| Primary pathway (What is the normal “happy path” for this use case?) | Open list of orders.  Administrator clicks on the list orders button.  A list of orders in the database is displayed to the Administrator. |
| Alternate pathways (What other paths are there that are not the “happy path”?) | Order list is empty – a message is displayed saying so. |
| Exception pathways (What could possibly go wrong?) | Database connection fails  Error displayed advising of connection problem |

|  |  |
| --- | --- |
| Use Case Name (Short two or three word name) | Remove order |
| Use Case Description (Short description) | Admin removes order from the order collection. |
| Use Case Author(s) (Who wrote this) | Jakub |
| Actor(s) (Who does this) | Administrator |
| Locations (Where does this happen) | Front and back end functionality |
| Primary pathway (What is the normal “happy path” for this use case?) | Remove order  Administrator removes order from the order list.  Order is removed from the list of orders in order collection. |
| Alternate pathways (What other paths are there that are not the “happy path”?) | There’s no such order in the database – an error is displayed. |
| Exception pathways (What could possibly go wrong?) | Database connection fails  Error displayed advising of connection problem |

|  |  |
| --- | --- |
| Use Case Name (Short two or three word name) | Edit Order |
| Use Case Description (Short description) | Administrator/user edits an order |
| Use Case Author(s) (Who wrote this) | Jakub |
| Actor(s) (Who does this) | Administrator/user |
| Locations (Where does this happen) | Front and back end functionality |
| Primary pathway (What is the normal “happy path” for this use case?) | Update Order  Administrator/user finds an order by its id and edits it. |
| Alternate pathways (What other paths are there that are not the “happy path”?) | Invalid data is entered and an error appears.  Invalid id, there is no such order, an error appears.  Some required fields are missing- an error is displayed saying so. |
| Exception pathways (What could possibly go wrong?) | Database connection fails  Error displayed advising of connection problem |

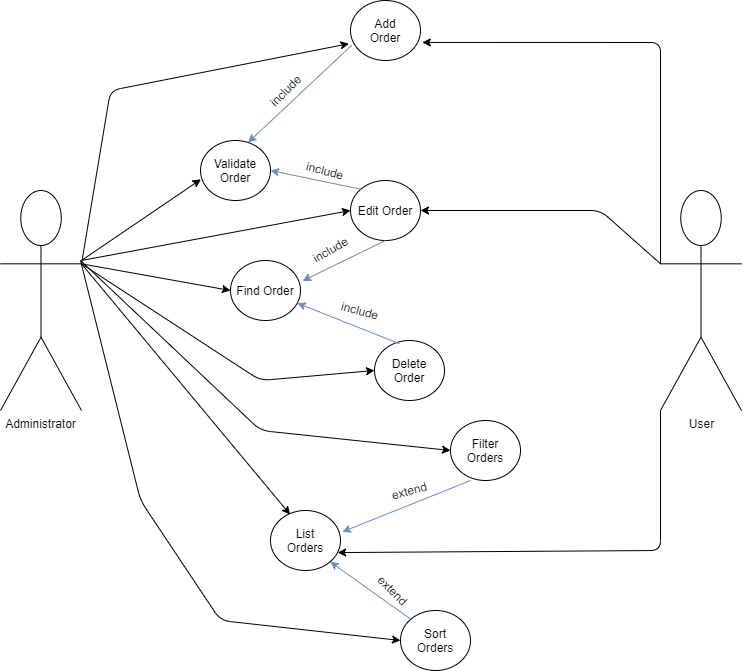
|  |  |
| --- | --- |
| Use Case Name (Short two or three word name) | Filter orders |
| Use Case Description (Short description) | Administrator filters the order collection using a predicate. |
| Use Case Author(s) (Who wrote this) | Jakub |
| Actor(s) (Who does this) | Administrator |
| Locations (Where does this happen) | Front and back end functionality |
| Primary pathway (What is the normal “happy path” for this use case?) | Filter orders  Administrator views a list of orders filtered by a given predicate. |
| Alternate pathways (What other paths are there that are not the “happy path”?) | Invalid data is entered as a predicate and an error appears.  Order list is empty, an error appears. |
| Exception pathways (What could possibly go wrong?) | Database connection fails  Error displayed advising of connection problem |

|  |  |
| --- | --- |
| Use Case Name (Short two or three word name) | Sort orders |
| Use Case Description (Short description) | Administrator sorts the order collection using a predicate. |
| Use Case Author(s) (Who wrote this) | Jakub |
| Actor(s) (Who does this) | Administrator |
| Locations (Where does this happen) | Front and back end functionality |
| Primary pathway (What is the normal “happy path” for this use case?) | Sort orders  Administrator views a list of orders sorted by a given predicate. |
| Alternate pathways (What other paths are there that are not the “happy path”?) | Invalid data is entered as a predicate and an error appears.  Order list is empty, an error appears. |
| Exception pathways (What could possibly go wrong?) | Database connection fails  Error displayed advising of connection problem |

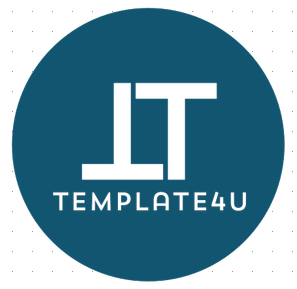
|  |  |
| --- | --- |
| Use Case Name (Short two or three word name) | Find order |
| Use Case Description (Short description) | Administrator finds the order by id. |
| Use Case Author(s) (Who wrote this) | Jakub |
| Actor(s) (Who does this) | Administrator |
| Locations (Where does this happen) | Front and back end functionality |
| Primary pathway (What is the normal “happy path” for this use case?) | Find order  Administrator uses order id to find a particular order then its displayed on the screen. |
| Alternate pathways (What other paths are there that are not the “happy path”?) | There’s no such order- an error is displayed. |
| Exception pathways (What could possibly go wrong?) | Database connection fails  Error displayed advising of connection problem |

|  |  |
| --- | --- |
| Use Case Name (Short two or three word name) | Validate |
| Use Case Description (Short description) | Checks if entered data is correct and can be added to the database. |
| Use Case Author(s) (Who wrote this) | Jakub |
| Actor(s) (Who does this) | Administrator |
| Locations (Where does this happen) | Back end functionality |
| Primary pathway (What is the normal “happy path” for this use case?) | Validate  Data is validated and then, added to the database |
| Alternate pathways (What other paths are there that are not the “happy path”?) | Data is invalid and an error appears. |
| Exception pathways (What could possibly go wrong?) | Database connection fails  Error displayed advising of connection problem |

**Use case diagram:**

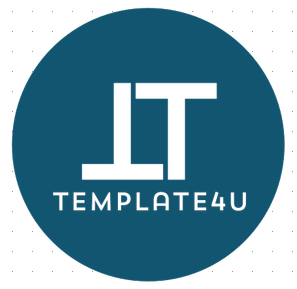


**Template4u LTD**

****

Alex

**Template4u LTD**

****

Dawid