# Problem

# Requirements

|  |  |
| --- | --- |
| R1 | A user can be registered |
| R2 |  |
| R3 |  |
| R4 |  |
| R5 |  |
| R6 | A parcel has a valid name, type, weight, and seller. |
| R7 | An order has a recipient and seller. |
| R8 |  |
| R9 |  |
| R10 |  |
| R11 |  |
| R12 |  |
| R13 |  |
| R14 |  |
| R15 |  |
| R16 |  |
| R17 |  |
| R18 |  |
| R19 |  |
| R20 |  |
| R21 |  |
| R22 |  |
| R23 |  |
| R24 |  |
| R25 |  |

# Test cases

## Normal values

## Invalid values

## Boundary values

## Special cases

# Test plan

Note: UT = Unit Test; IT = Integration Test

|  |  |  |  |
| --- | --- | --- | --- |
| Test | Instruction | Expected result(s) | Actual result(s) |
| User Manager | | | |
| UT1 | Find user by user ID (where ID exists in db and where its absent) |  |  |
| UT2 | Find user by username (where username exists in db and where its absent) |  |  |
|  | Find user role by username |  |  |
|  | View all users |  |  |
|  | Create user |  |  |
| User Gateway | | | |
|  | Insert user |  |  |
|  | Find user by user ID |  |  |
|  | Find user by username |  |  |
|  | Find user role by username |  |  |
|  | Find all users |  |  |
| Order Manager | | | |
|  | Create order |  |  |
|  | Edit order |  |  |
|  | Delete order |  |  |
|  | Find order by order ID |  |  |
|  | View recipient deliveries |  |  |
|  | View driver metrics |  |  |
|  | View order parcels |  |  |
|  | View all deliveries |  |  |
| Order Gateway | | | |
|  | Insert order |  |  |
|  | Update order |  |  |
|  | Delete order |  |  |
|  | Find order by order ID |  |  |
|  | Find recipient deliveries |  |  |
|  | Find driver metrics |  |  |
|  | Find all deliveries |  |  |
|  | Find order parcels |  |  |
| Transaction Manager | | | |
|  | Add transaction |  |  |
|  | Remove transaction |  |  |
|  | Find transaction by transaction ID |  |  |
|  | View order transactions |  |  |
| Transaction Gateway | | | |
|  | Insert transaction |  |  |
|  | Delete transaction |  |  |
|  | Find transaction by transaction ID |  |  |
|  | Find order transactions |  |  |
| Parcel Manager | | | |
|  | Create parcel |  |  |
|  | Edit parcel |  |  |
|  | Delete parcel |  |  |
|  | Find parcel by parcel ID |  |  |
|  | View all parcels |  |  |
| Parcel Gateway | | | |
|  | Insert parcel |  |  |
|  | Update parcel |  |  |
|  | Delete parcel |  |  |
|  | Find parcel by parcel ID |  |  |
|  | Find all parcels |  |  |
|  | Find order parcels |  |  |
| RegisterBean (register bean doesn’t have dependency injection) | | | |
|  | Get next user ID returns a positive integer |  |  |
|  | Validate inputs returns true when password1 matches password2 |  |  |
|  | Validate inputs returns false when password1 does not match password2 |  |  |
|  | Hash password returns a hashed string |  |  |
|  | Register will create a new user and apply the role “Recipient” by default. |  |  |
| LoginBean (login bean doesn’t have dependency injection) | | | |
|  | Check credentials returns true when valid login details are provided for a user that exists in the database. |  |  |
|  | Check credentials returns false when invalid login details are provided or when the user does not exist in the database. |  |  |
|  | User page returns a UI string relative to the role of the current user using the application. |  |  |
|  | Find user role returns the correct role for the current user using the application. |  |  |
|  | Clear credentials will clear the visible form and values in the managed bean. |  |  |
|  | Logout will clear user details within the managed bean, then return the string “Login\_UI”. |  |  |
| DbManager | | | |
|  | Get connection returns a valid connection that can be used in the rest of the application. |  |  |
| Metric DTO | | | |
|  | Can get name from a Metric DTO |  |  |
|  | Can get delivery count from a Metric DTO |  |  |
|  | Can get days to complete from a Metric DTO |  |  |
| Order DTO | | | |
|  | Can get ID from an Order DTO |  |  |
|  | Can get recipient as a User DTO from an Order DTO |  |  |
|  | Can get driver as a User DTO from an Order DTO |  |  |
|  | Can get seller as a User DTO from an Order DTO |  |  |
|  | Can get the date an order was added from an Order DTO |  |  |
|  | Can get whether an order is completed from an Order DTO |  |  |
|  | Can get whether an order is completed and if so, the date the order was completed from an Order DTO |  |  |
| Parcel DTO | | | |
|  | Can get ID from a Parcel DTO |  |  |
|  | Can get name of parcel from a Parcel DTO |  |  |
|  | Can get type of parcel from a Parcel DTO |  |  |
|  | Can get weight of a parcel in grams from a Parcel DTO |  |  |
|  | Can get seller as a User DTO from a Parcel DTO |  |  |
|  | Can get the date a parcel was added to the system from a Parcel DTO |  |  |
|  | Can get the date a parcel was last modified from a Parcel DTO |  |  |
|  | Can get the number of times a parcel has been sold from a Parcel DTO |  |  |
|  | Can set and get the quantity of a parcel in the order from a Parcel DTO |  |  |
| Transaction DTO | | | |
|  | Can get ID from a transaction DTO |  |  |
|  | Can get the OrderID of a transaction from a transaction DTO |  |  |
|  | Can get the name of a transaction from a transaction DTO |  |  |
|  | Can get the user who added the transaction as a User DTO from a transaction DTO |  |  |
|  | Can get the date that a transaction was added from a Transaction DTO |  |  |
| User DTO | | | |
|  | Can get user ID from a User DTO |  |  |
|  | Can get first name from a User DTO |  |  |
|  | Can get last name from a User DTO |  |  |
|  | Can get username from a User DTO |  |  |
|  | Can get the hashed password from a User DTO |  |  |
|  | Can get the date a user was added to the database from a User DTO |  |  |
|  | Can get the date a user was last modified from a User DTO |  |  |
|  | Can get the address line one from a User DTO |  |  |
|  | Can get the town from a User DTO |  |  |
|  | Can get the county from a User DTO |  |  |
|  | Can get the postcode from a User DTO |  |  |
|  | Can get the email address from a User DTO |  |  |
|  | Can get the phone number from a User DTO |  |  |
|  | Can get whether a user is active in the database from a User DTO |  |  |
|  | Can get the users role from a User DTO |  |  |
| Register\_UI | | | |
|  | RegisterCommand |  |  |
|  | RegisterCommandFactory |  |  |
|  | Create user command |  |  |
| Login\_UI | | | |
|  | LoginCommand |  |  |
|  | LoginCommandFactory |  |  |
|  | View user by username command |  |  |
|  | View user role command |  |  |
| Recipient\_UI | | | |
|  | RecipientCommand |  |  |
|  | RecipientCommandFactory |  |  |
|  | View recipient deliveries command |  |  |
|  | View delivery progress command |  |  |
| Driver\_UI | | | |
|  | DriverCommand |  |  |
|  | DriverCommandFactory |  |  |
|  | Add transaction command |  |  |
|  | Remove transaction command |  |  |
|  | View all deliveries command |  |  |
| Seller\_UI | | | |
|  | SellerCommand |  |  |
|  | SellerCommandFactory |  |  |
|  | Create order command |  |  |
|  | Create parcel command |  |  |
|  | Delete order command |  |  |
|  | Delete parcel command |  |  |
|  | Edit order command |  |  |
|  | Edit parcel command |  |  |
|  | View all parcels command |  |  |
|  | View all users command |  |  |
|  | View driver metrics command |  |  |
|  | View order parcels command |  |  |
|  | View order transactions command |  |  |
|  | View parcel command |  |  |
|  | View user command |  |  |
|  |  |  |  |
|  | Create a parcel with valid values. | Creates a new parcel with the details provided. |  |
|  | Create parcel with an invalid weight of “heavy” (string when expecting an integer) and valid other values for isolation. | Parcel is not created and … (handle? Error?) |  |
|  |  |  |  |
| Integration Tests | | | |
| IT11 | - Create a recipient user with username “Jimmy”.  - Create a seller user with username “Sally”.  - **Create order** where “Jimmy” is the recipient and “Sally” is the seller.  **-** Run **View recipient deliveries** when using the application as “Jimmy”. | Returns a list Jimmy’s orders containing at least the one where Sally is the seller. | (result and evidence) |
| IT12 | - Create a recipient user with username “Jimmy”.  - Create a seller user with username “Sally”.  - **Create order** where “Jimmy” is the recipient and “Sally” is the seller.  - Create a driver user with username “Bob”.  - **Add transaction** “Picked up” to the order while using the application as “Bob” (Driver).  - **Add transaction** “Dropped off” to the order while using the application as “Bob” (Driver).  - **Remove transaction** “Dropped off” from the order while using the application as “Bob” (Driver).  - Run **View delivery progress** against the order created. | Returns details of the order where “Jimmy” is the recipient, “Sally” is the seller, “Bob” is the driver and shows 1x transaction of “Picked up”, and no other transactions. |  |
| IT13 | - Create a recipient user with username “Jimmy”.  - Create a seller user with username “Sally”.  - **Create order** where “Jimmy” is the recipient and “Sally” is the seller.  - Create a driver user with username “Bob”.  - Run **View all deliveries** while using the application as “Bob” (Driver). | Returns list of all orders. |  |
| IT14 | - Create a seller user with username “Sally”.  - **Create parcel** with name=”Old item”, type=”Book”, weight=500, sellerID=(ID of Sally).  - **Edit parcel** updating to name=”New item”, type=”Box”, weight=10, sellerID=3 (Existing user “seller”’s ID).  - View details of parcel that was just created then edited. | Returns details of the parcel: name=“New item”, type=”Box”, weight=10, sellerID=3. |  |
| IT15 | - Create a seller user with username “Sally”.  - **Create parcel** with name=”Old item”, type=”Book”, weight=500, sellerID=(ID of Sally).  - **Delete parcel** that was just created. | Parcel does not exist in database and can not be seen within the application. |  |
| IT16 | - Create a recipient user with username “Jimmy”.  - Create a seller user with username “Sally”.  - **Create order** with recipientId=(ID of “Jimmy”), sellerId=(ID of Sally).  - **Edit order** updating to recipientId=1 (Existing recipient users ID), sellerId=3 (Existing seller users ID).  - View details of order that was just created then edited. | Returns details of the new order: recipientId=1, sellerId=3. |  |
| IT17 | - Create a recipient user with username “Jimmy”.  - Create a seller user with username “Sally”.  - **Create order** with recipientId=(ID of “Jimmy”), sellerId=(ID of “Sally”).  - **Delete order** that was just created. | Order no longer exists in database and can not be seen within the application. |  |
| IT18 | - Create a recipient user with username “Jimmy”.  - Create a seller user with username “Sally”.  - **Create order** where “Jimmy” is the recipient and “Sally” is the seller.  - **Create parcel** with name=”Old item”, type=”Book”, weight=500, sellerID=(ID of Sally).  - Add new parcel to new order with quantity 2.  - **Delete parcel** that was just added to the order.  - View details of order that had the parcel deleted. | Order no longer lists the now deleted parcel. When a parcel is deleted, all orders with that parcel are then updated to not include the parcel. |  |
| IT19 | - Create a driver user with username “Bob”.  - Create a driver user with username “Dave”.  - Create a recipient user with username “Jimmy”.  - Create a seller user with username “Sally”.  - **Create order** “Order1” where “Jimmy” is the recipient and “Sally” is the seller.  - **Create order** “Order2” where “Jimmy” is the recipient and “Sally” is the seller.  - **Create order** “Order3” where “Jimmy” is the recipient and “Sally” is the seller.  - **Add transaction** “Picked up” to “Order1” while using the application as “Bob”.  - **Add transaction** “Dropped off” to “Order1” while using the application as “Bob”.  - **Add transaction** “Picked up” to “Order2” while using the application as “Bob”.  - **Add transaction** “Dropped off” to “Order2” while using the application as “Bob”.  - **Add transaction** “Picked up” to “Order3” while using the application as “Dave”.  - **Add transaction** “Dropped off” to “Order3” while using the application as “Dave”.  - Run **View driver metrics** while using the application as “Sally” (Seller). | Returns list of metrics showing “Bob” with 2 total deliveries and average days to complete 0 & “Dave” with 1 total delivery and average days to complete 0. |  |