REQUIREMENT DOCUMENT
Purpose
To provide users users with security and guarantee when buying and selling products online.

Scope
The application is so that:
- Users can validate the products they are buying, before and after receiving the product.
- Users should be able to confirm the product they are purchasing.

Product perspective
The application stores the following information:
- User information. This is basic information about the user eg. their names, email and location.
- Product information. Relevant information about the product of purchase eg. name, its description, the price, and the users in play – the buyer and the seller
- Calender information. Dates relevant to the product purchase eg. delivery date, confirmation date

Product flow
Buyer and Seller signup to the appliation.
Buyer generates a product form and fills relevant information regarding the product.
Buyer adds a Seller identifier (id, email), and the product form is sent to the Seller.
Seller confirms that it is the right product.
- An agreement is made.

*** Buyer receives the right product ***
Buyer logs in and confirms receiving the right product.
Sender receives the payment.

*** Buyer receives the wrong product ***
Buyer logs in and declines receiving the right product.
Buyer and Seller have to meet an agreement. Meanwhile the application holds the money.

*** Buyer does not confirm receiving the product ***
Sender receives the payment.

Buyer claims not to have received the product
Seller claims to have sent the product

Terms of the application
Both the users are to agree on the application terms before any transaction happens.
- The buyer has to deposit money to their wallet before they can proceed with the transaction.
- The grace period a buyer gets after the delivery date to confirm receiving the right product.
  After the period is up, the money deposited by the buyer is immediately sent to the seller.
- The buyer gets a 3 day period to confirm receiving
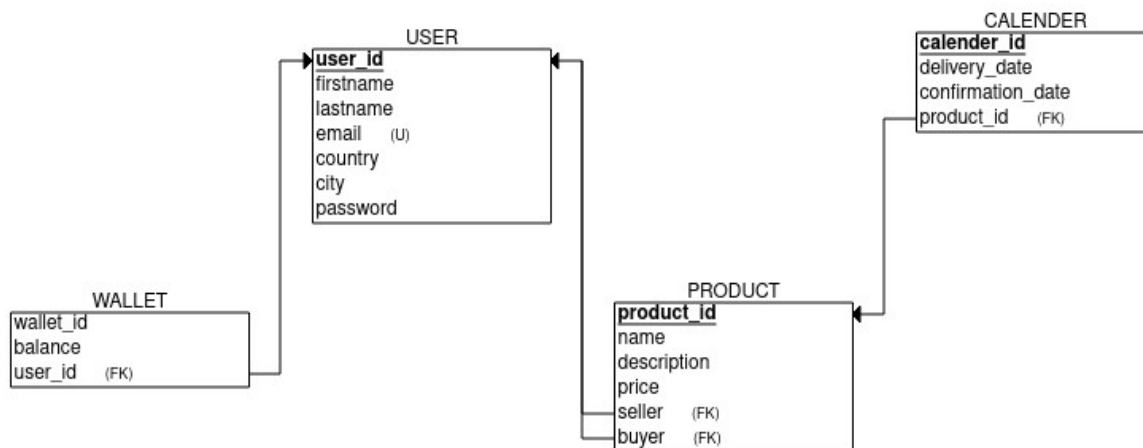
Terms of transaction
During the initial setup of a product purchase, the key aspects related to the transaction are discussed and agreed upon. The product details, the price, and the delivery date.

Technologies
Frontend – ReactJS
Backend – NodeJS
Database – PostgreSQL

Login/Signup
User stories
1. As a new user, I want to register by giving my name, email, country, city and password so that the system can remember me and my data.

2. As a registered user, I want to log in with my username and password so that the system can authenticate me and I can trust it.

Endpoints
/signup

```
{
        firstname: …,
        lastname: ….,
        email: …,
        country: …,
        city: …,
        password: …
}
```

- Frontend
  - Does not submit if blanks exist
  - Show user errors from backend (status = 4xx)

- Backend
  - Return error if email exists, status = 409
    ```
    {
            error: 'email_already_exists',
            message: 'The email already exists /...'
    }
    ```
  - Return success message on successful signup, status = 200
    ```
    {
            message: "Successful signup",
            token: auth_token
    }
    ```

/login

```
{
        email: …,
        password: …,
}
```

- Frontend
  - Does not submit if blanks exist
  - Show user errors from backend (status = 4xx)

- Backend
  - Return error if email and password don't match, status = 401
    ```
    {
            error: 'incorrect_email_password',
            message: 'Incorrect email or password'
    }
    ```
  - Return success message on successful login, status = 200
    ```
    {
            message: "Successful login",
            token: auth_token
    }
    ```

Page components
Signup page
- Pretty form with fields:
    - Firstname
    - Lastname
    - Email
    - Country
    - City
    - Password
- "SIGNUP" button

Login page
- Pretty form with fields:
    - Email
    - Password
- "LOGIN" button