# 3 Requirements

## 3.1 Introduction

The user will be able to login when they use the application (they don’t have to), first they will need to register an account which will be stored in the database. The user will have to login using an email and password, they will also need to enter their shoe size and general clothing size.

There will be an admin for the application so for when they login they will have control over certain aspects of the application. They will have CRUD functionality so that it is easy to manipulate the contents of a certain page by altering the database.

The user will be able to browse products, filter and sort products and also bid on the products they desire. There is no cart, as far as purchase goes there will only be a successful bid pop up as a visual confirmation

## 3.2 Requirements gathering

### 3.2.1 Similar applications

o Screenshots

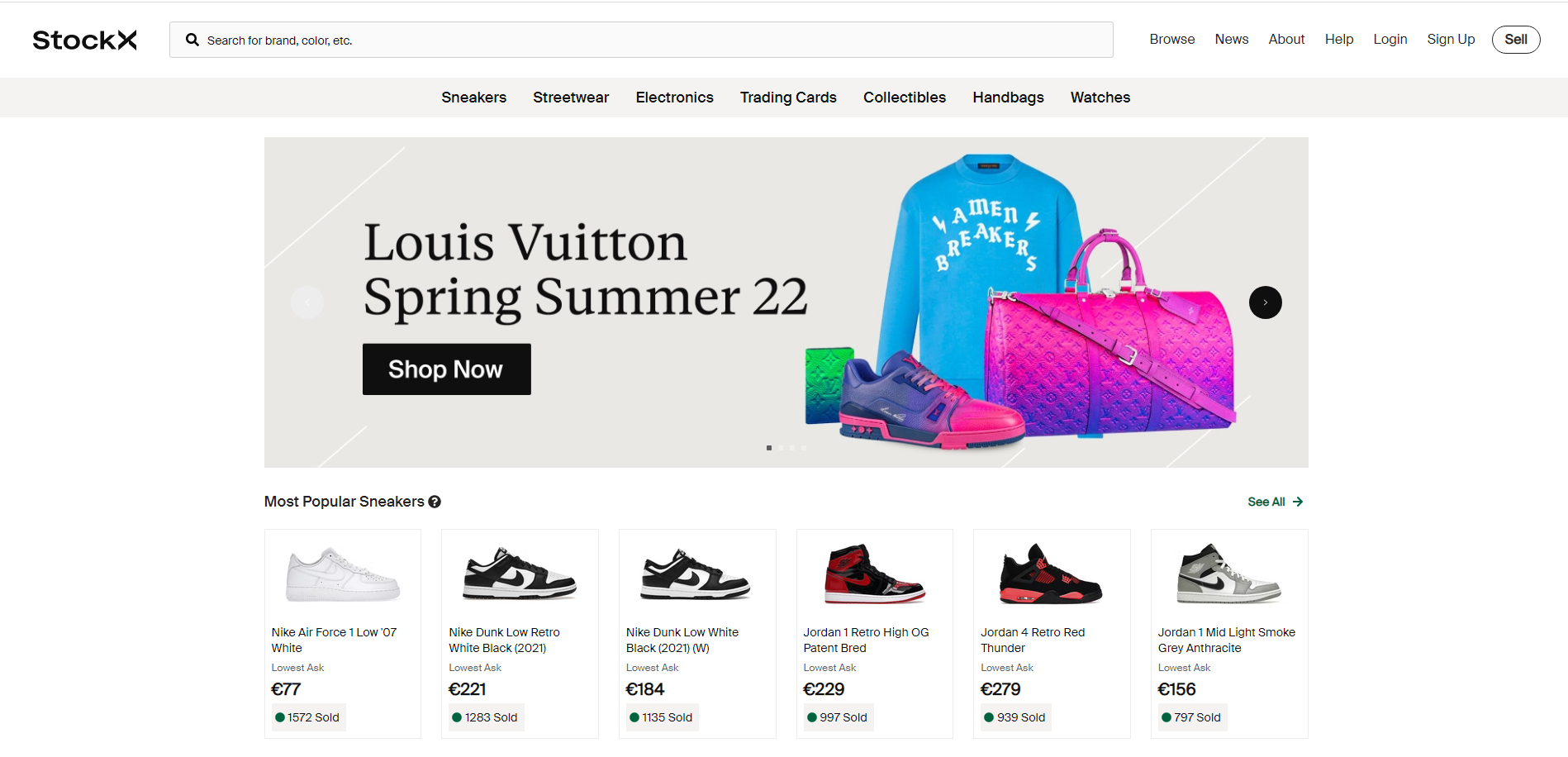


Figure The homepage of stockx

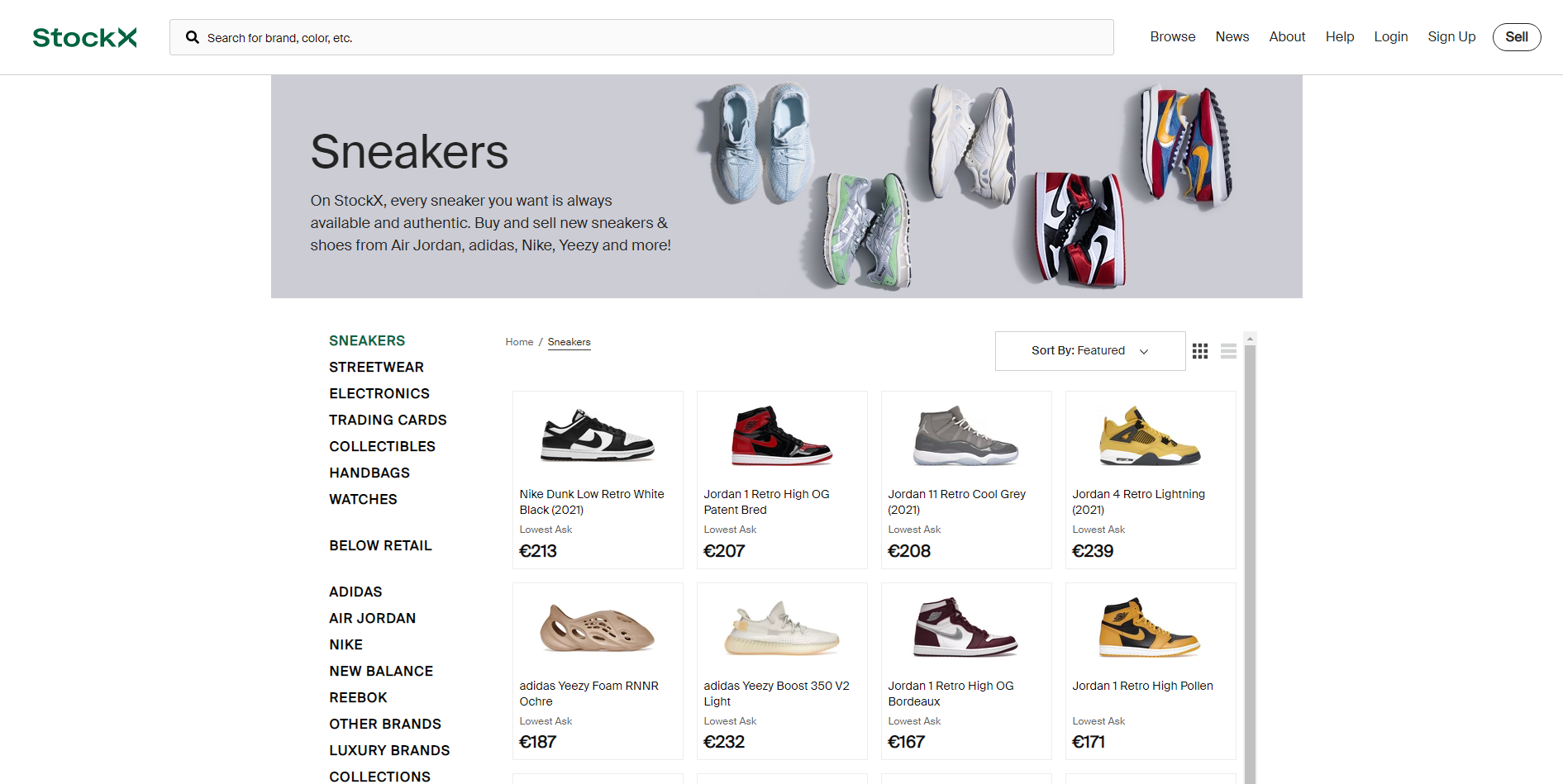


Figure The browse page for sneakers with some sort and filter options

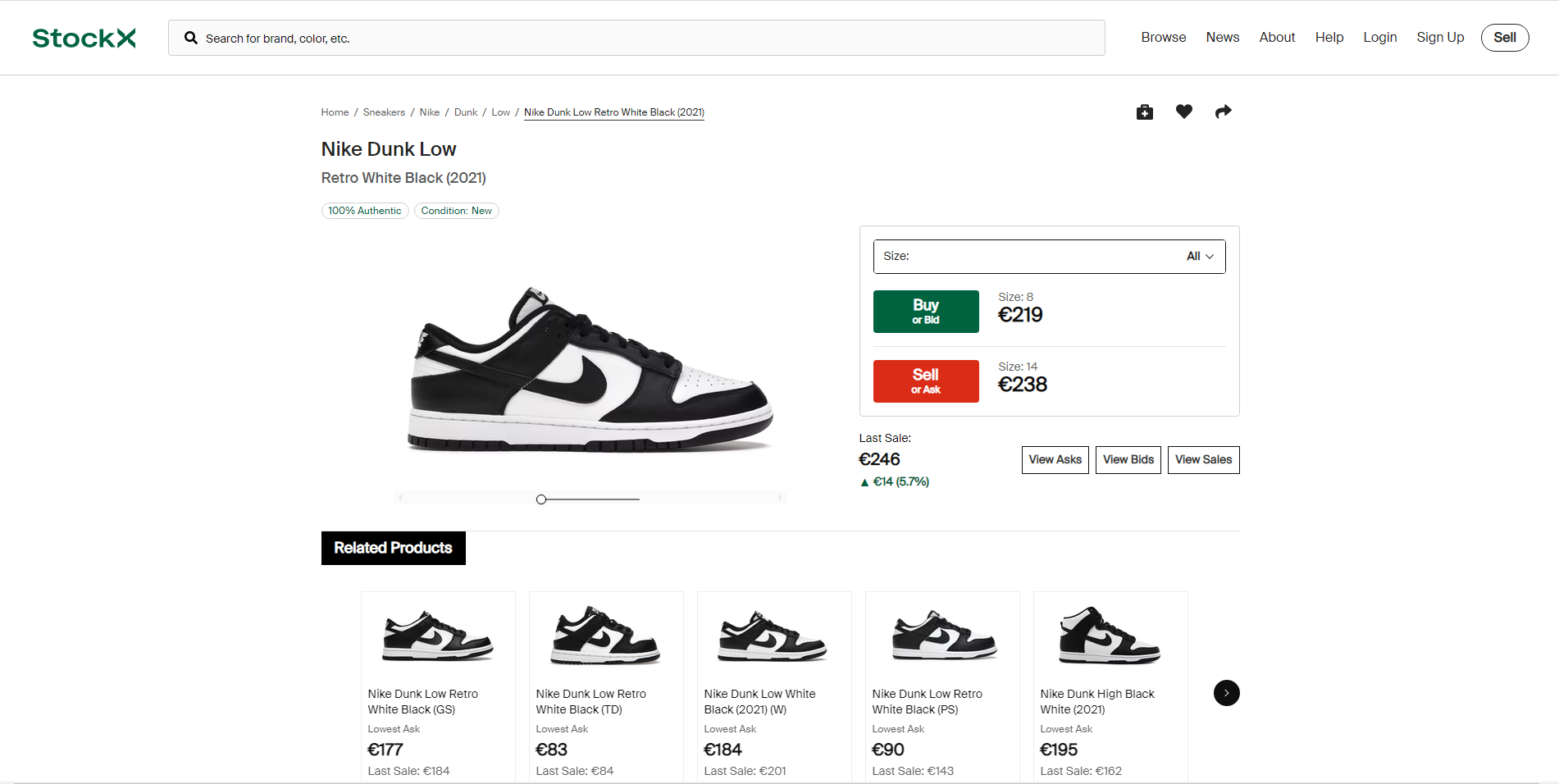


Figure The product view page with ability to bid

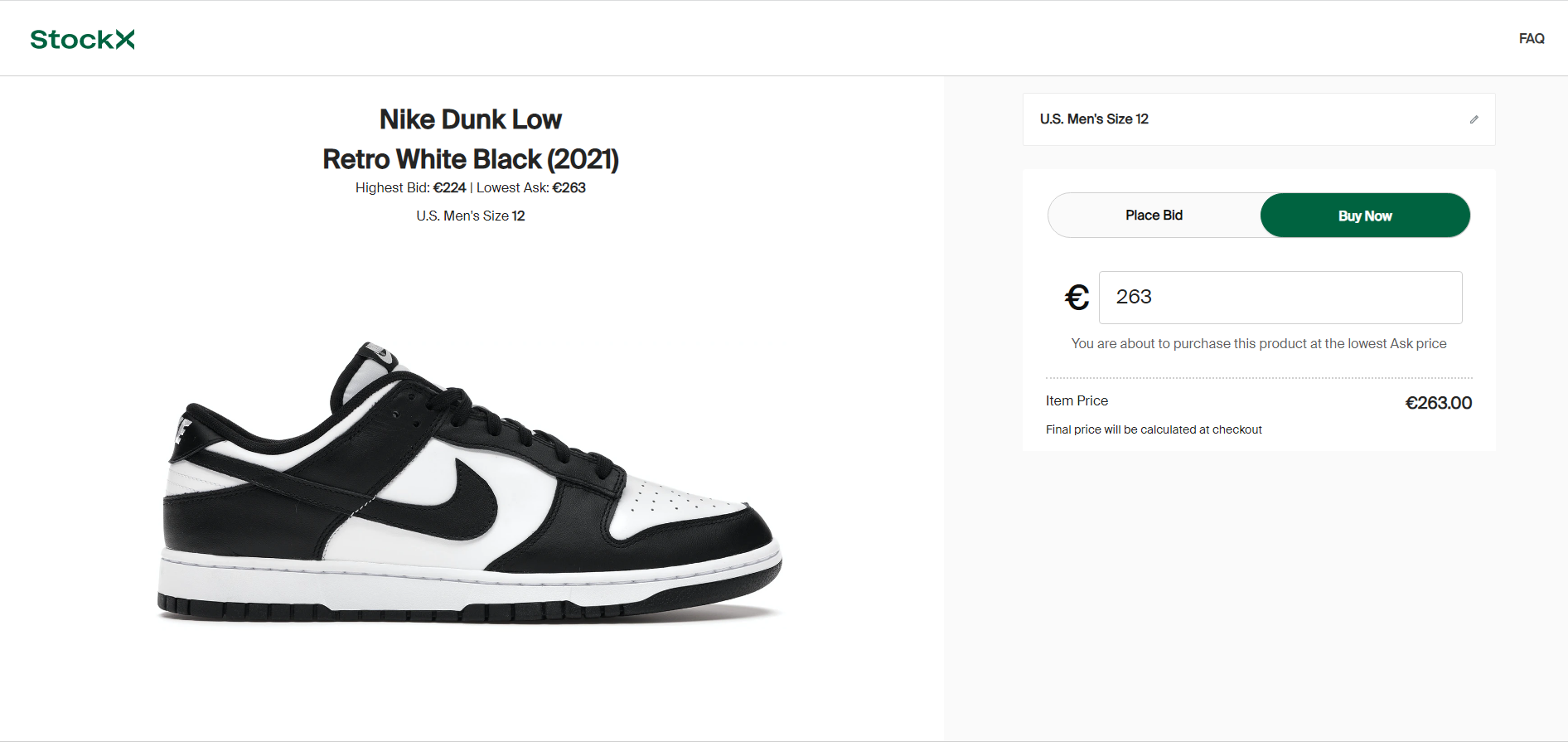


Figure The bidding and selling page for products

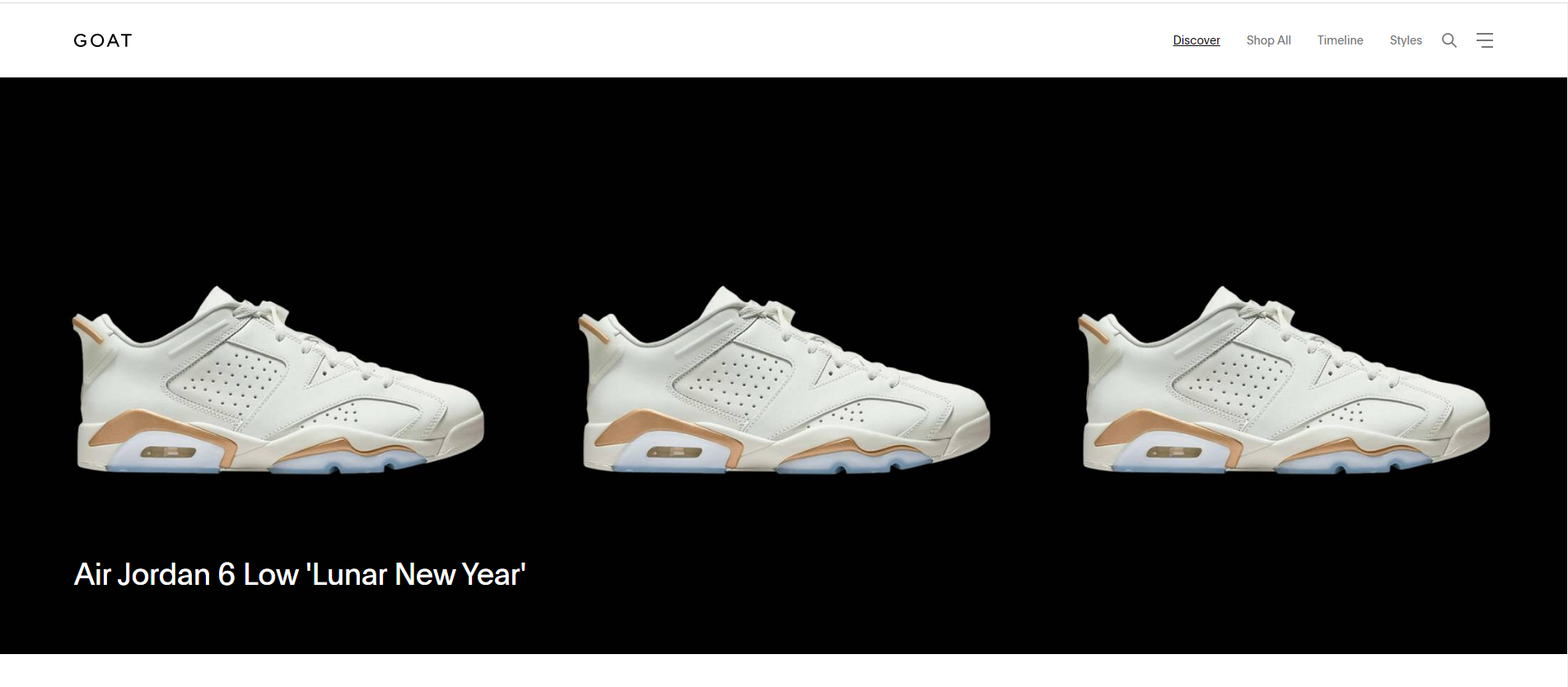


Figure GOAT homepage

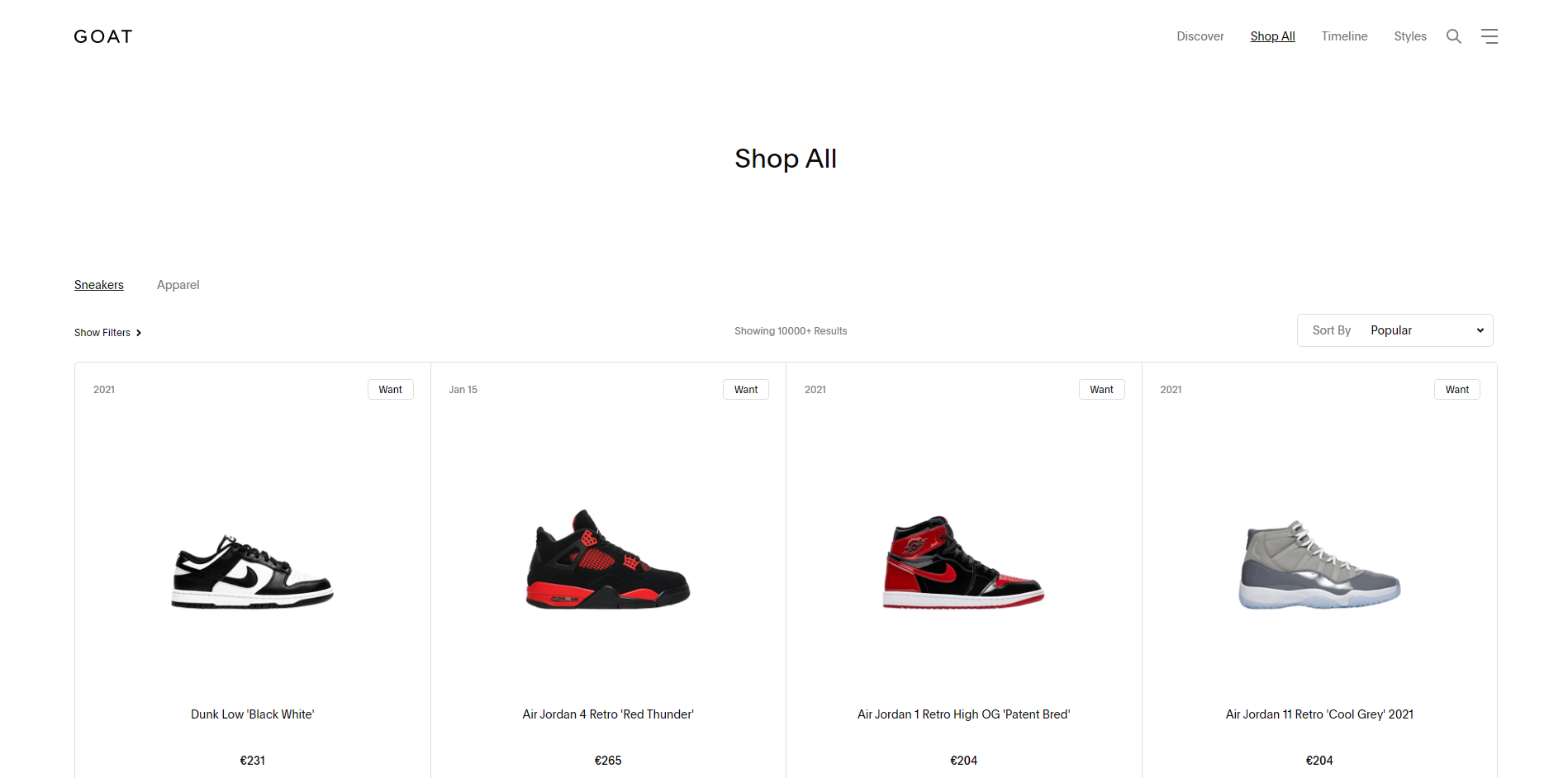


Figure The browse shoes page for GOAT

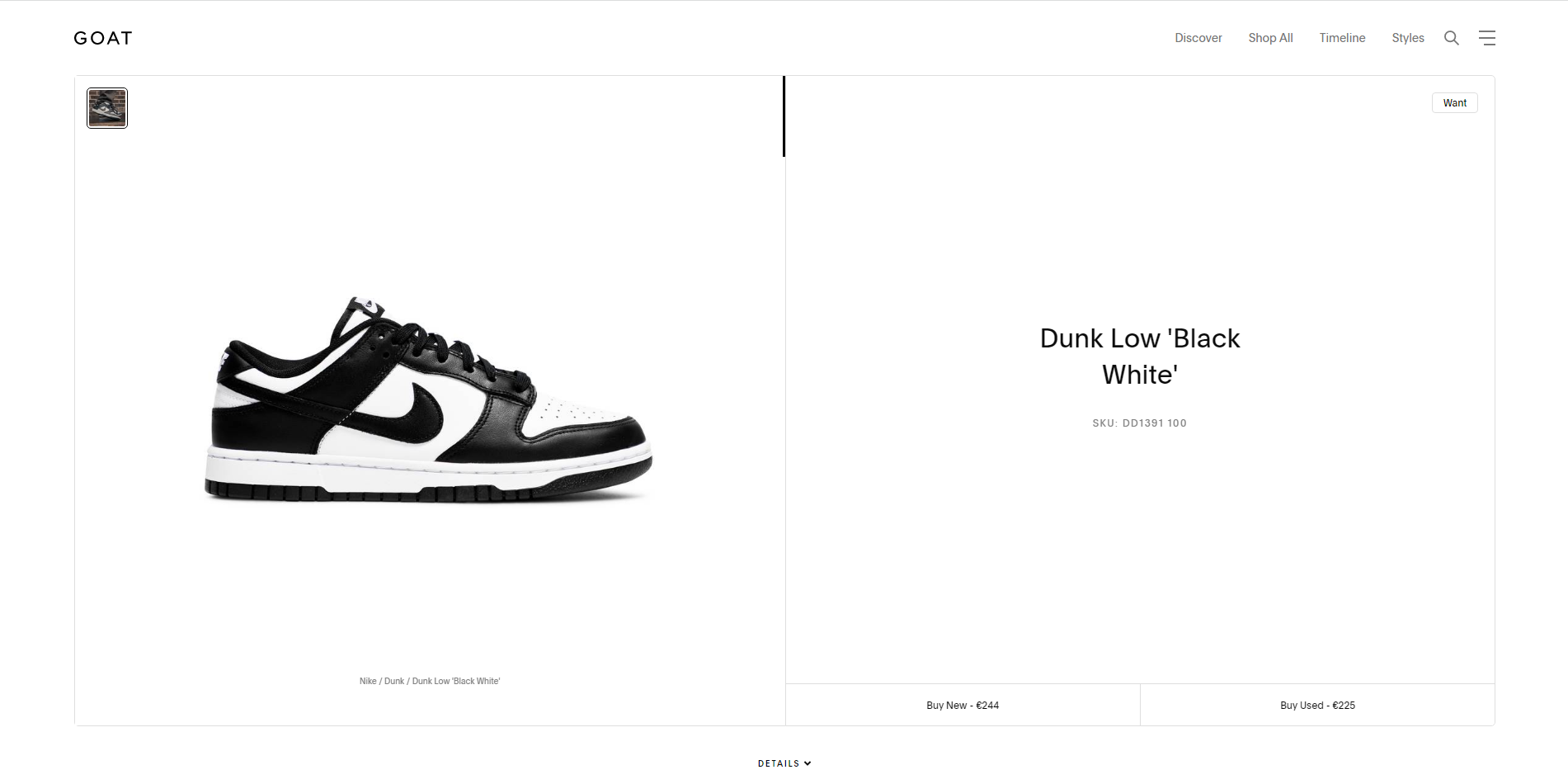


Figure The product view page with ability to bid

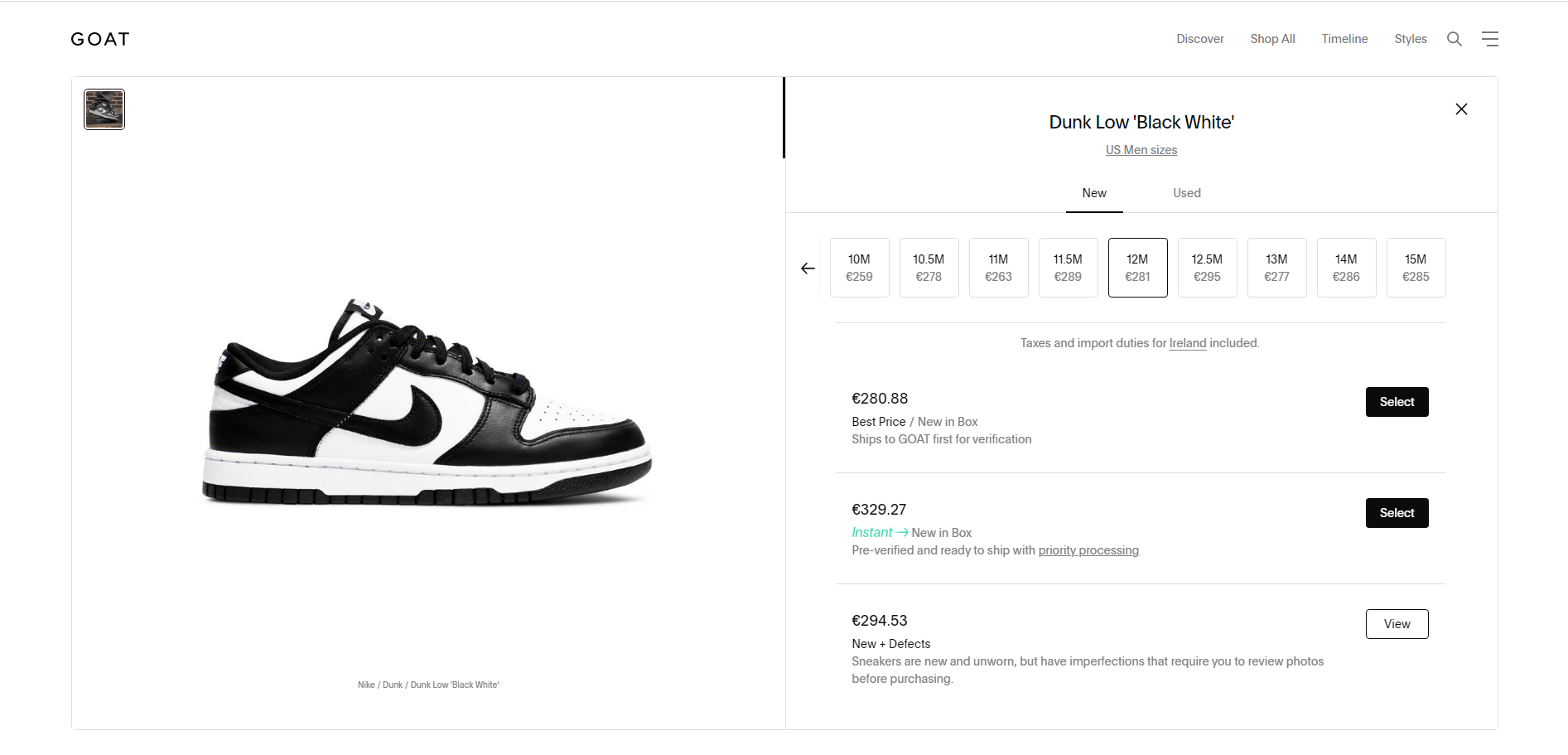


Figure The product view page with ability to bid

o Descriptions

The screen shots above are the basic flow for the process of buying a pair of shoes on both stockx and GOAT, it starts off on the browse page and it quickly allows you to buy or bid for a pair of shoes. I found that stockx was slightly easier to use and more clear.

o Advantages

Both sites have a quick way to find products as they both have search options.

Pictures of the products with clear descriptions.

Coloured buttons to differentiate between different services.

o Disadvantages

Prices are for specific sizes which is misleading.

New users can find the searching process hard on GOAT as there is no clear browse link.

On GOAT the buying process uses one click too many and can be frustrating

### 3.2.2 Interviews

Conduct interviews with 2 or 3 users to find out what the important features for them for the app are. There may be various issues that arise in multiple interviews. These can be grouped together into a number of themes.

## 3.3 Requirements modelling

### 3.3.1 Functional requirements

Users will be able to browse products

Users can register/login

Users can bid or sell

Admin has crud functionality over application

### 3.3.2 Non-functional requirements

o Usability

The application should work in all browser sizes and still keep its design identity

The application should have the ability to go back to a page easily or back to the homepage

o Performance

There will be visual signs that inform users on how their request is coming along e.g. a spinner when they bid on a product

o Security

When a user registers or logs in they must follow a password convention that requires certain characters

If the bid is successful they should be redirected to the correct page

### 3.3.3 Use Case Diagrams

Consists of actors and use cases. You should document each individual use case.

Delete the following diagram and insert your diagram. Use draw.io

## 3.4 Feasibility

For the project I will be using Laravel, MySQL, GitHub and bootstrap. I will be using Laravel to build the application, this is where I will be making web pages and controllers to control the flow of the application. Laravel has bootstrap built into it so I will be using it to build the webpages using HTML and CSS. Laravel will be communicating to the MySQL database where I will be retrieving data to populate my application. This will all be hosted on GitHub where I can access and change files remotely