Scopic

sales@scopicsoftware.com
+1 (508) 886-3240,
+1 (855) 717-5586 (toll free)
12 Carly Circle,
Rutland, MA, 01543, US
scopicsoftware.com

Full-stack Test Task

Scopic

sales@scopicsoftware.com
+1 (508) 886-3240,
+1 (855) 717-5586 (toll free)
12 Carly Circle,
Rutland, MA, 01543, US
scopicsoftware.com

scopicsoftware.com

12 Carly Circle
Rutland MA 01543 US

+1 (508) 886-3240
+1 (855) 717-5586 (toll free)

jobs@scopicsoftware.com

Task Description

Your task is to create a web auction application for an antique items seller. The application will allow users to bid on antique items displayed in the site and admin users to set up items for auction. Product management and auctioning are within the scope of the application; shopping cart and payment integration are not.

Pages

1. Home Page

This should be the first page the user is navigated to after logging in to the system. On this page, the user can see the list of existing item (how the list is displayed is your choice, but keep in mind the basic principles of good usability).

It should have pagination with 10 items per page. It should also have a search box to filter the items by the Name and Description fields. The Price column should be sortable.

2. Item Details Page

Each Auction item will have a *Bid Now* button next to it. When the Bid Now button is clicked, the client is sent to the details page of the item. The Item Details page displays a countdown timer showing how much time is left before the bidding is closed (the close date and time for the auctions are set by the admin). The client should be able to make a bid on the item. The bid made should start at a value that is higher than the last bid made on the item (i.e. If the previous user made a bid for $15, the start bid for the next user should be $16)

Once the bid is made, the user clicks the *Submit Bid* button. One user can make multiple bids on the same item, as long as their bid is not the highest in the system.

sales@scopicsoftware.com
+1 (508) 886-3240,
+1 (855) 717-5586 (toll free)
12 Carly Circle,
Rutland, MA, 01543, US
scopicsoftware.com

The auto-bidding feature

The ability to activate the auto-bidding will be possible in the **Item detail page**. A user can activate auto-bidding by clicking on a checkbox in the item detail page aside the Bid now button.
The next time someone else makes a bid on the marked item, the bot should automatically outbid them by 1.

Configuring the Auto-bidding

The Auto-bidding parameters can be configured by the user in a separate page (you can choose at your discretion how to display / show the configurations page).

These parameters are listed below:

1) **Maximum bid amount** (showing the maximum amount the bot can use for auto bidding purposes)

Note: This maximum amount will be split between all items where we have activated auto-bidding.

Important Note: Be mindful of the concurrency issues with auto-bidding!

When there is not enough funds anymore, the auto-bidding will stop.

Requirements

1. General Requirements

Front End and Back End implementations are required. REST API should be the communication standard between the Front End and Back End.

There should be dummy user authentication implemented into the system with hardcoded user credentials in the code. No need to have users stored in the Database.

There should be 1 role in the system for Regular users. Their credentials are as follows:

- **Regular**: user1/user2

The users should be able to:

- View the list of items
- View the details of a single item

sales@scopicsoftware.com
+1 (508) 886-3240,
+1 (855) 717-5586 (toll free)
12 Carly Circle,
Rutland, MA, 01543, US
scopicsoftware.com

Important note: There won't be an administrator panel to manage the items – so you can use Demo data pre-stored in the database. Make sure to account for all the necessary attributes

2. Technical Requirements

- Use PHP / Laravel on the Back-end and React.js on the Front End.
- Use Database engine of your choice.
- Implement form submission validation on the Front End side.
- Implement validation on the Back End side for the creation and modification of items.
- Use git to commit your work.
- Function body should not exceed 30 LOC.
- Class should not exceed 200 LOC.
- Stick to DRY principles.
- Base your implementation on design patterns.
- Layout should be responsive and follow UI/UX best practices.
- Any Image files should be uploaded and shared in a zip file.

Summary of main features:

1. Home Page – Item's listing (preferably in gallery view)
2. Item Detail Page with Item bidding history
3. Bid Now functionality
4. Auto-bidding functionality