Software Requirements Specification

for

# Stock X Low-Ball Bidding Bot

Version 1.0

Prepared by Doğan Ali ŞAN

27.05.2025

Revision History

| Date | Version | Description | Author |
| --- | --- | --- | --- |
| 27/05 | 1.0 | Original | Doğan Ali ŞAN |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

1. Introduction

# 1.1 Purpose (main scenario)

A humanly bot that scans [stockx.com](http://stocksx.com) to find filter matching products to low-ball bid on. After buying the bot will sell the product on another website. When selling it should talk humanly, arrange prices according to a filter that will be provided to it to make a deal. This bot will be duplicated 100 times and each individual bot will not conflict with each other make their biddings and gain profit.

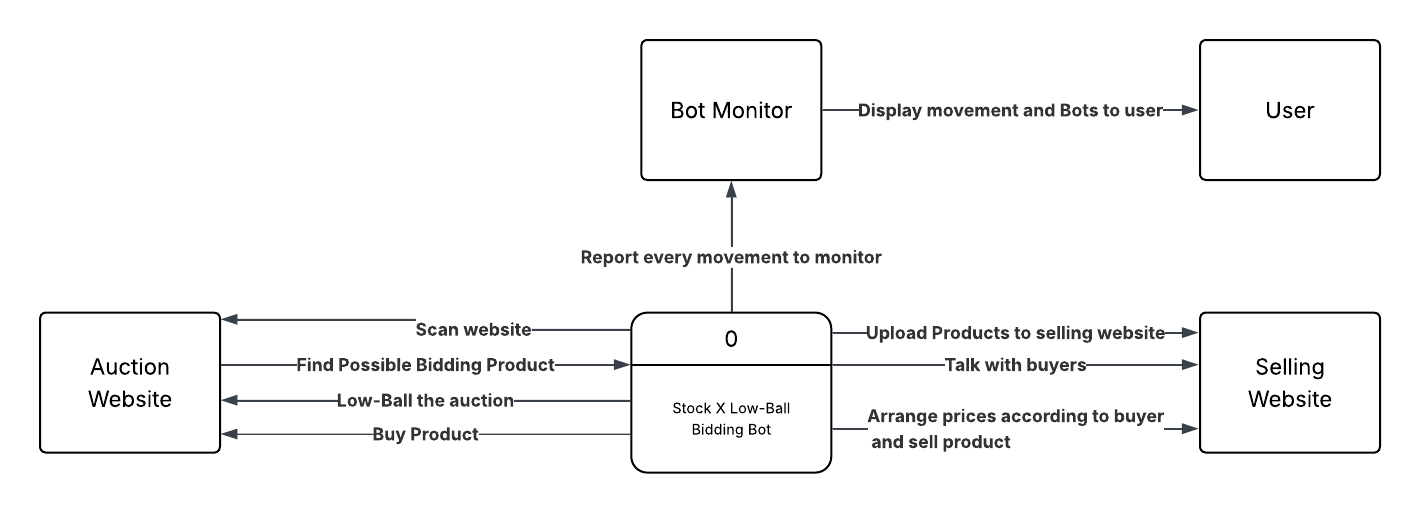
# 1.2 Project Scope

Stock X Low-Ball Bidding Bot will gain profit by low-balling on desperate sellers.

2. Overall Description

# 2.1 Description

Bidding bot is an automated way for finding cheap and quality products. The context diagram below illustrates the system interfaces and external entities that will be involved in this release.



Caption

# 2.2 User Class

| User Class | Description and Characteristics |
| --- | --- |
| User | User can actively monitor and maintain bots, change their action. |

# 2.3 Operation Environment

**2.3.1** Monitoring system must operate correctly with the most updated versions of the following web browsers:

Microsoft Edge versions; Firefox; Google Chrome; and Safari.

**2.3.2** Monitoring system must be optimized for desktops as well as smartphones and tablets

**2.3.3** General system must operate correctly within a VPS.

**2.4 Design and Implementation Constraints**

**2.4.1**  Bidding bots will make bids and sells while their action is updated on monitoring website and can be stopped or changed.

**2.5 Assumptions**

**2.5.1** Monitoring website will be available 24-7 outside of scheduled maintenance

**2.5.2** Bidding bots must operate correctly unless told otherwise

3. Business Rules

# 3.1 Rules List

| Rule ID | Rule Definition | Type | Impacted Requirement |
| --- | --- | --- | --- |
|  |  |  |  |

4. Functional Requirements (FR)

# 4.1 User

**4.1.1 U - User must be able to identify all self-services through the website homepage**

**4.1.1.1 FR** System must display the following self-services when the **Monitor** option is selected on the site home page:

* + Active Bot count
  + Active Bot actions
  + Active Bot action canceling/stopping buttons
  + Bought product count

**4.1.1.2 FR** System must display the following self-services when the Operation option is selected on the site home page:

* + Bidding Bots Operation Start Button
  + Bidding Bots Operation Stop Button

**4.1.1.3 FR** System must display the following self-services when the Products option Is selected on the site home page:

* + Products that are already bought
  + Which Bot bought it
  + Buying price
  + Selling price
  + Profit

**5. Use Cases**

**5.1 Use Case 1**

|  |  |
| --- | --- |
| Name (description) | Stock X Low-Ball Bidding Bot |
| Requirement ID | 4.1.1.1 - 4 FR - User can see active bidding bots and interact |
| Pre-conditions | User opened monitor page |
| Trigger | N/A |
| Post-conditions | Confirmation email sent to User |

|  |  |
| --- | --- |
| **Main Success Path**  **(Primary Flow)** | User Stops working bidding bot to check its actions |

| Actor Actions | System Response |
| --- | --- |
| 1. User selects monitor page | 2. System displays activate bots with stopping options |
| 3. User selects required options and settings | 4. System displays prompt to confirm and stops selected bidding bot |

6. Data requirements

# 6.1 Logical Data Model

**6.1.1 Entity Relationship Diagram for Stock X Low-Ball Bidding Bot**

# 6.2 Data Dictionary

**6.2.1 Data Dictionary for Collaborative Task Management System**

| Attribute | Entity | Description | Required | Type | Source Database | Length | Default Val | Comments |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Product Name | Product | Name of Bought Product | Yes | Text | Products | N/A | N/A | N/A |
| Product description | Product | Description of bought product | Yes | Text | Products | N/A | N/A | N/A |
| Product Date | Product | Buying date of the product | Yes | Date | Products | 6 | N/A | Date format MM/DD/YY |
| Product İmage list | Product | List of the images that product has | Yes | List | Products | N/A | N/A | List = İmage Links |

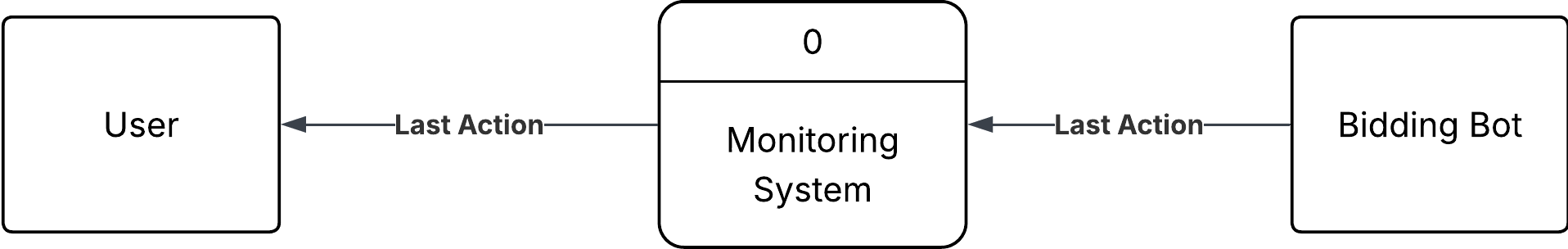
# 6.3 Data Maintenance

**6.3.1** The website must retain Product history for up to 365 days

7. External Requirements

# 7.1 User Interface

**7.1.1 UI** System must provide smooth browsing.



# 7.2 Software Interfaces

**7.2.1 SI - Bidding bots**

**7.2.1.1 SI -** Bidding bots must transmit every action immediately to Action database so monitoring system can fetch its movements.

**7.2.2 SI - Monitoring System**

**7.2.2.1 SI -** Monitoring system must fetch live data from databases and display immediately when there is change.

**7.2.2.2 SI -** Monitoring system must work immediately to take action when user interacts with a bot.

# 7.3 Communication Interface

**7.3.1 CI** - System must send an email message to the user to alert an interaction made

8. Nonfunctional Requirements

# 8.1 Availability Requirements

**8.1.1 NFR** - System Must be available 99% of the time outside of scheduled maintenance

# 8.2 Compatibility Requirements

**8.2.1 NFR** - System must be compatible with different operating systems, such as Windows, macOS, Linux, and mobile operating systems like IOS and Android

**8.2.2 NFR** - System must be compatible with device types, including desktop computers, laptops, tablets, and smartphones

# 8.3 Reliability Requirements

**8.3.1 NFR** - System must install any available software updates nightly to remain up to date

**8.3.2 NFR** - System must have failover mechanisms to ensure continuous functioning of the website in case of software failures

# 8.4 Scalability Requirements

**8.4.1 NFR** - System must be able to handle a large volume of concurrent users without affecting performance of availability

**8.4.2 NFR** - System must have as caching mechanism that stores frequently accessed data and pages, reducing, the load on the database and speeding up page load times

# 8.5 Localization Requirements

**8.5.1 NFR** - System must support local data and time formats that are commonly used in the team members country or region

**8.5.2 NFR** - System must support the translation of all text content into the specified languages, including buttons, menus, and error messages

9. Reporting Requirements

# 9.1 Report - 1

**9.1.1 Report Title**

9.1.1.1 Report title is User Action

9.1.1.2 Report ID is USRACT-RPT

# 9.1.2 Report - 1

**9.1.2.1 Report Title**

9.1.2.2 Report title is bot action

9.1.2.3 Report ID is BOTACT-RPT

# 9.1.3 Headers and Footer

9.1.3.1 Report header must contain the report title, User name / bot name, and the date range specified

9.1.3.2 Report body must show the action taken

9.1.3.3 Report footer must show the page number.

# 9.1.4 Special Rules or Conditions

9.1.4.1 Report must receive a date range input to generate results

9.1.4.2 Report must allow for filtering by the various departments in addition to the date range

# 9.1.5 Field Order

9.1.5.1 Fields should be in the following order

* + User ID
  + Users Name
  + Action data

# 9.1.6 Totaling

9.1.6.1 Report must total the number of tasks based on all tasks or filtered tasks

# 10. Glossary of Terms

**10.1 Team member -** A member of a group

**10.2 Project manager** - Leader of the group

# 11. Approvals and Sign-off

| Name | Position | Reviewer | Approval | Signature | Date |
| --- | --- | --- | --- | --- | --- |
| Doğan Ali Şan | Developer | X | X | X | 13/12 |