 **Authentication Subsystem**: This subsystem would handle user authentication processes, including verifying credentials, managing session tokens, and integrating with email for user verification or password resets.

 **Communication Subsystem**:

* **Messaging Subsystem**: Manages all interactions within the bot environment, including command parsing and response handling.
* **Notification Subsystem**: Handles outgoing notifications, such as email alerts for price changes or availability updates, which could be a distinct or integrated part of the Messaging Subsystem

 **Monitoring Subsystem**:

* **Price Monitoring Subsystem**: Monitors price changes of specified items and logs these changes. It may trigger notifications based on defined criteria.
* **Availability Monitoring Subsystem**: Similar to price monitoring but focuses on the availability status of items or services, providing updates and logs as defined.

 **Data Handling Subsystem**:

* **Data Storage and Retrieval**: Manages data interactions, including saving logs to databases or files and retrieving them for user requests.
* **Data Export Subsystem**: Handles the formatting and exporting of data to various formats like Excel or HTML, serving both internal logging needs and user-requested exports.

 **Browser Operation Subsystem**: Manages all interactions requiring a web browser, such as navigating to websites, logging in, and other browser-based actions.

**Usability**: The system shall be designed to ensure ease of use, where users can effectively interact with the bot to perform tasks such as monitoring prices, checking availability, and receiving notifications with minimal effort and learning curve. It should provide clear, understandable feedback and guidance to the user, enabling them to accomplish their desired tasks efficiently. The user interface should be simple to navigate, with commands that are easy to remember and execute, and errors should be handled gracefully, providing users with informative messages to guide their next steps.

@startuml

skinparam componentStyle uml2

package "Discord Bot System" {

component "Authentication Subsystem" as Auth {

[Authentication] #LightBlue

}

package "Communication Subsystem" as CS{

component "Messaging Subsystem" as Messaging {

[DiscordMessage] #LightGreen

}

component "Notification Subsystem" as Notification {

[Notifications] #Yellow

}

}

package "Monitoring Subsystem" as MS{

component "Price Monitoring Subsystem" as PriceMon {

[Price Monitoring] #Orange

}

component "Availability Monitoring Subsystem" as AvailMon {

[Availability Monitoring] #Pink

}

}

component "Data Handling Subsystem" as DataHandling {

[Data Handling] #Violet

}

component "Browser Operation Subsystem" as BrowserOps {

[Browser Operations] #Purple

}

' Relationships

Auth -right-> CS

CS --> BrowserOps

BrowserOps --> MS

MS-down-> DataHandling : "fetches/stores data"

}

@enduml

