

# Purchase Request Lifecycle

---

## Step 1: Frontend Receives Request

The frontend service exposes a POST endpoint at `/purchase/:id`. This endpoint receives requests from the client (such as a web browser). The request body contains the number of copies the user wants to purchase.

## Step 2: Round-Robin Service Selection

The frontend service uses a round-robin algorithm to select which catalog service and order service to use. It rotates between `catalog-service-1` and `catalog-service-2`, and between `order-service-1` and `order-service-2`.

## Step 3: Forward Request to Order Service

The frontend forwards the purchase request to the selected order service using a POST request to `/PurchaseBook/:id`, including the quantity and selected catalog service name in the body.

## Step 4: Order Service Handles Purchase

The selected order service processes the purchase by:

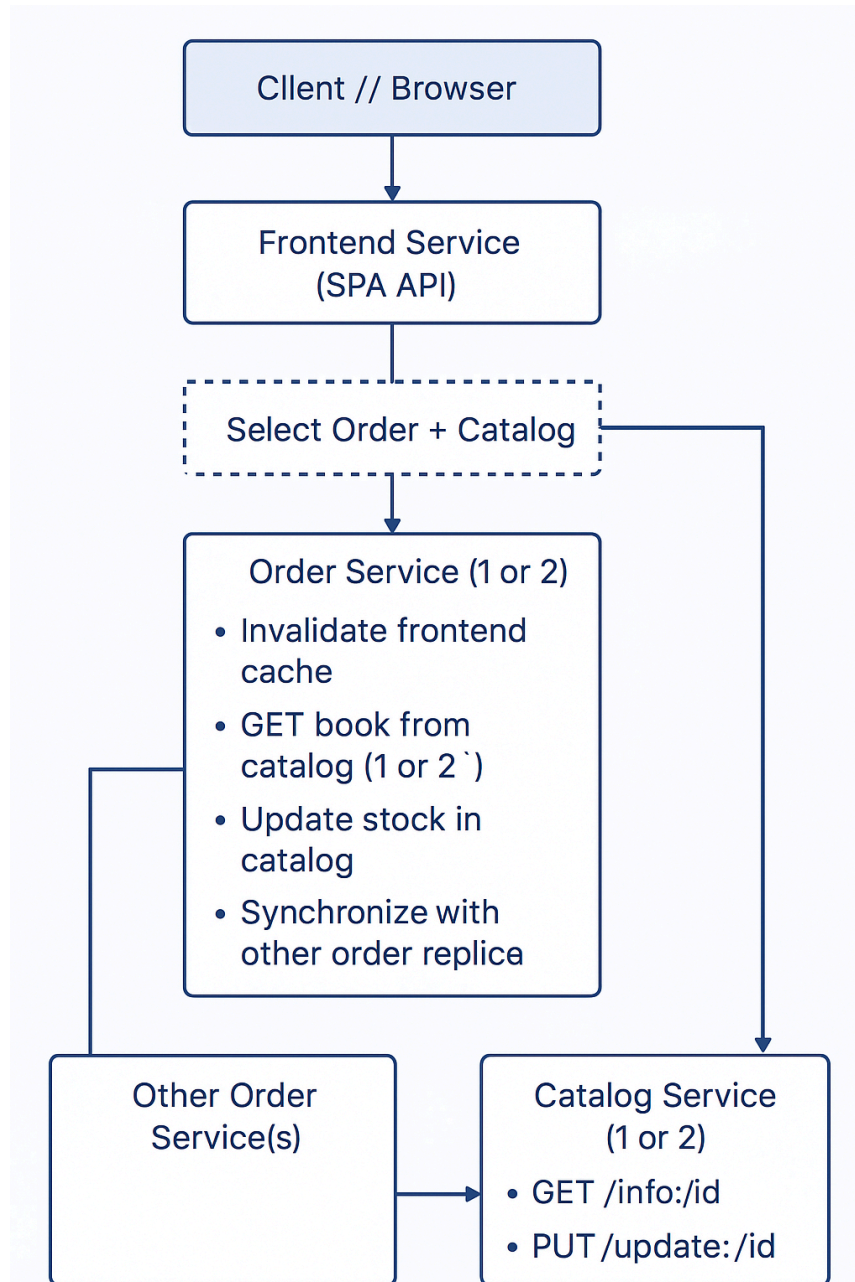
- Invalidating the frontend cache for the book ID
- Fetching the book info from the selected catalog service
- Checking available stock
- Updating the stock in the catalog if enough is available
- Synchronizing the update with other order service replicas
- Inserting the order into its own SQLite database
- Returning a success or error response

## Step 5: Response Sent to Client

The response from the order service is returned to the frontend service, which logs the activity and sends the final result back to the client.

# flowchart

---



# Performance

---

- Before caching:

- computed the average of **10** requests per query type

```
front-end-service | Average Response Time after 10 requests: 11.70 ms
front-end-service | Average Response Time after 10 requests: 71.80 ms
```

- After caching

- First, the cache will be empty so every access will be a miss, but after that the hit rate will start to increase.
- in the image the first 10 requests had 4 misses, which is why the average response time was 6.80, but in the next 10 requests there was only 1 miss so the average response time was significantly decreased to 1.20.

```
front-end-service | Server is running on port 3002..
front-end-service | Cache miss for topic : distributed systems
front-end-service | Cache hit for topic : distributed systems
front-end-service | Cache miss for topic : undergraduate school
front-end-service | Cache hit for topic : undergraduate school
front-end-service | Cache miss for book with id : 1
front-end-service | Cache hit for book with id : 1
front-end-service | Cache hit for topic : undergraduate school
front-end-service | Cache hit for book with id : 1
front-end-service | Cache miss for book with id : 3
front-end-service | Cache hit for book with id : 3
front-end-service | Average Response Time after 10 requests: 6.80 ms
front-end-service | Cache hit for topic : undergraduate school
front-end-service | Cache hit for topic : distributed systems
front-end-service | Cache hit for book with id : 3
front-end-service | Cache hit for book with id : 1
front-end-service | Cache miss for book with id : 2
front-end-service | Cache hit for topic : distributed systems
front-end-service | Cache hit for book with id : 1
front-end-service | Cache hit for book with id : 3
front-end-service | Cache hit for topic : undergraduate school
front-end-service | Cache hit for book with id : 1
front-end-service | Average Response Time after 10 requests: 1.20 ms
```

**By : Areen Halabi & Wasan Awwad**