

# IMDB DRINKS – RATINGS & REVIEWS REQUIREMENTS

## 1. Introduction

The “IMDB Drinks” platform aims to let users discover, rate, and review different types of drinks. This section of requirements focuses on how users **rate** drinks (using star ratings), **write reviews**, **search and filter** drinks by rating or name, and how the system handles **CAPTCHA** validation and **guest (not logged in) user behavior**.

## 2. Glossary

- **Drink:** An item in the database representing a beverage.
- **Star Rating:** A numeric rating from 1 (lowest) to 5 (highest).
- **Review:** A user-submitted text associated with a star rating for a specific drink.
- **CAPTCHA:** A challenge to verify that a review is submitted by a human and not a bot.
- **Guest User:** A user who is not currently logged in.

## 3. Stakeholders & Owners

- **Ratings Team (Your Team):** Primarily responsible for implementing the rating logic, review submission, and filtering by rating.
- **Search & UI Team (Cross-Team):** Responsible for the search bar UI/UX and ensuring the front-end handles search and filter interactions correctly.
- **Authentication/Authorization Team (Cross-Team):** Ensures only logged-in users can submit reviews and enforces sign-in flows.
- **Security/Infrastructure Team (Cross-Team):** Provides and maintains CAPTCHA functionality.

---

## 4. Functional Requirements

### 4.1 Rating & Review Submission

#### 1. FR1: Star Ratings

- **Description:** The system shall allow users to rate each drink using an integer star rating from **1 to 5**.
- **Owner:** Ratings Team
- **Verification:**
  - Attempt to submit a rating below 1 or above 5 → System rejects and displays error.
  - Submit valid star rating → System accepts and saves the rating.
- **Feasibility:** The database can store an integer value (1–5) for each user–drink rating.

#### 2. FR2: Text Reviews

- **Description:** The system shall allow users to submit a **textual review** alongside their star rating.
- **Owner:** Ratings Team
- **Constraints:**

- Text field must support at least 500 characters.
    - Images in reviews are **not allowed** (see FR3).
  - **Verification:**
    - Attempt to attach an image → System displays an error and disallows submission.
    - Submit textual review → System saves and associates the review with the user and the drink.
3. **FR3: No Images in Reviews**
- **Description:** The system shall **not** allow users to attach or embed images in their reviews.
  - **Owner:** Ratings Team
  - **Verification:**
    - Attempt to upload an image or embed an image link → System rejects or strips the image content.
  - **Rationale:** Ensures uniform text-based reviews and avoids moderation overhead.
4. **FR4: CAPTCHA for Review Submission** (Cross-Team: Security/Infrastructure)
- **Description:** The system shall require the user to **successfully solve a CAPTCHA** before submitting a new review.
  - **Owner:** Ratings Team (implementation trigger) / Security Team (CAPTCHA service)
  - **Constraints:**
    - CAPTCHA must be displayed each time a new review form is submitted.
    - If the CAPTCHA is not solved correctly, the review is **not** saved.
  - **Verification:**
    - Submit a review with incorrect CAPTCHA → Review fails to post.
    - Submit a review with correct CAPTCHA → Review is posted successfully.
5. **FR5: Logged-In Users Can Submit Reviews** (Cross-Team: Authentication/Authorization)
- **Description:** Only **logged-in** users shall be able to submit reviews and ratings.
  - **Owner:** Ratings Team (front-end check), Auth Team (back-end check)
  - **Verification:**
    - A guest user tries to submit a review → System redirects to Sign-In page or displays an error.
    - A logged-in user can submit a review → System processes normally.

---

## 4.2 Viewing & Filtering Reviews

6. **FR6: View Reviews Without Logging In**
- **Description:** The system shall allow **any user (guest or logged-in)** to see the existing reviews for a drink.
  - **Owner:** Ratings Team

- **Constraints:**
    - A “View Reviews” button or link is displayed for each drink.
    - If a guest user clicks it, they can read reviews but **cannot** post.
  - **Verification:**
    - Guest user sees reviews page → No “Submit Review” form is shown.
    - Logged-in user sees reviews page → “Submit Review” form is shown.
7. **FR7: Average Star Rating**
- **Description:** The system shall calculate and display an **average star rating** for each drink based on all submitted user ratings.
  - **Owner:** Ratings Team
  - **Verification:**
    - If a drink has ratings of 5, 3, 2 → Average is 3.3 (rounded to one decimal or displayed as needed).
    - Display the average rating in the drink’s detail page or listing.
8. **FR8: Filter Drinks by Rating**
- **Description:** The system shall allow users to **filter** the list of drinks by a **minimum star rating** (e.g., “Show only drinks rated 4 or above”).
  - **Owner:** Ratings Team
  - **Verification:**
    - Set filter to 4 → Only drinks with an average rating  $\geq 4.0$  are displayed.
    - Clear filter → All drinks are displayed.

---

#### 4.3 Searching & Related Drinks

9. **FR9: Search Bar by Name** (Cross-Team: Search & UI)
- **Description:** The system shall provide a **search bar** where users can type a drink’s name (or part of it) to find matching drinks.
  - **Owner:** Ratings Team (back-end filter), Search & UI Team (front-end)
  - **Constraints:**
    - Case-insensitive search.
    - Partial match accepted (e.g., searching “Col” can return “Cola” and “Cold Brew Coffee”).
  - **Verification:**
    - Searching for “Tea” → All drinks containing “Tea” in the name appear.
    - If no match → “No results found” message is displayed.
10. **FR10: Related Drinks**
- **Description:** The system shall display “Related Drinks” for each drink, based on shared tags, categories, or flavor profiles (*exact logic decided by your domain needs*).
  - **Owner:** Ratings Team (algorithm), Search & UI Team (display)
  - **Verification:**
    - Drink detail page for “Green Tea” → “Related Drinks” might show other teas or herbal drinks.
    - No relevant category → “No related drinks found.”
-

#### 4.4 Guest vs. Logged-In Behavior

##### 11. FR11: Button to See Reviews When Not Logged In

- **Description:** A “**See Reviews**” button or link shall be visible even if the user is not logged in, but attempting to **add** a review triggers the Sign-In flow.
- **Owner:** Ratings Team, Auth Team
- **Verification:**
  - Guest user sees “See Reviews” → Can read existing reviews.
  - Guest user clicks “Add Review” → System prompts for login.

##### 12. FR12: Prompt Sign-In If User Tries to Filter/Sort Reviews

- **Description:** The system may allow all users to filter reviews by rating or date. However, if advanced features (e.g., “Mark review as helpful”) exist, they require login.
- **Owner:** Ratings Team, Auth Team
- **Verification:**
  - Guest user attempts an advanced review action → Sign-In page.
  - Logged-in user can perform advanced actions successfully.

---

#### 5. Non-Functional Requirements (NFR)

##### 1. NFR1: Performance

- The system shall display search/filter results within **2 seconds** for up to **10,000** drinks in the database.

##### 2. NFR2: Reliability

- The system shall ensure that reviews are never lost once submitted. If an error occurs, the user is notified, and the review is not partially saved.

##### 3. NFR3: Usability

- The star rating widget must be straightforward to use (e.g., clickable stars).
- The CAPTCHA must be user-friendly (e.g., reCAPTCHA or similar).

##### 4. NFR4: Security

- All review submission endpoints must be protected against spam by CAPTCHA.
- Only authenticated users can alter or delete their own reviews.

##### 5. NFR5: Scalability

- The ratings system must handle up to **1,000,000** total reviews without degrading average search/filter times beyond **2 seconds**.

---

#### 6. Cross-Team Requirements Summary

- **CAPTCHA Integration:**
  - **Owner:** Ratings Team → triggers the CAPTCHA on the review form;
  - **Security Team** → provides the CAPTCHA service and verification endpoint.
- **Authentication:**
  - **Owner:** Auth Team → ensures login status;
  - **Ratings Team** → checks the user’s session before letting them submit a review.
- **Search Bar & Filtering:**

- **Owner:** Search & UI Team → front-end search bar UI;
- **Ratings Team** → implements the server-side filtering logic by rating, name, etc.

---

## 7. Requirement Traceability & Verification

Req. ID	Title	Verification Method	Owner
FR1	Star Ratings	Unit tests; manual test	Ratings Team
FR2	Text Reviews	Unit tests; manual test	Ratings Team
FR3	No Images in Reviews	Manual test; code review	Ratings Team
FR4	CAPTCHA for Review Submission	Integration test w/ Security	Ratings + Security
FR5	Logged-In Users Can Submit	Auth test; manual test	Ratings + Auth
FR6	View Reviews Without Logging In	UI test; manual test	Ratings Team
FR7	Average Star Rating	Unit test (calculation)	Ratings Team
FR8	Filter Drinks by Rating	Unit test; integration test	Ratings Team
FR9	Search Bar by Name	UI & integration test	Ratings + Search/UI
FR10	Related Drinks	UI & logic test	Ratings + Search/UI
FR11	Button to See Reviews (Guest)	UI test	Ratings + Auth
FR12	Prompt Sign-In for Add Review	UI test; auth test	Ratings + Auth

---

## 8. Feasibility & Constraints

- **Database:** Must support storing each drink's ratings and text reviews.
  - **Time Constraint:** These features must be implemented by the end of the next sprint (two weeks).
  - **Legal/Content Moderation:** Since images are disallowed, textual moderation is simpler. If you later add images, you will need a separate moderation policy.
- 

## 9. Approval & Sign-Off

All stakeholders have reviewed these requirements. By signing off, each team agrees to implement their respective components and meet the acceptance criteria outlined.

Role/Team	Name/Signature	Date
<b>Ratings Team Lead</b>		2025-03-14
<b>Search &amp; UI Team</b>		2025-03-14
<b>Auth Team</b>		2025-03-14
<b>Security Team</b>		2025-03-14
<b>Product Owner</b>		2025-03-14

---

## Final Note

This document covers **only** the Ratings & Reviews requirements (plus search/filter aspects that intersect).