

# Web programming report - Starry Sketch

Raul Pop  
Cezara Danciu

May 2025

## 1 Introduction

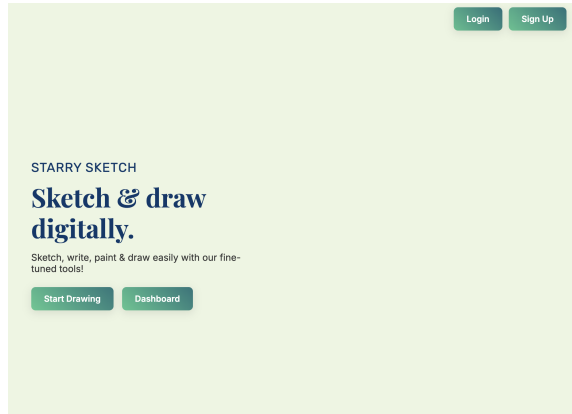
This document describes the requirements and structure of a browser-based drawing application. Users can sign up, log in, and manage multiple drawing canvases. Drawings are saved locally in the browser, and each user has a personal dashboard to open or create canvases.

## 2 Project Overview

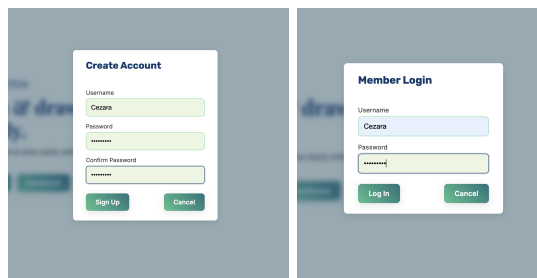
- **Purpose:** Allow users to create, edit, and save freeform drawings on a virtually infinite canvas.
- **Users:** Multiple registered users, each with a private dashboard of saved canvases.
- **Core Features:**
  1. User authentication (sign-up, log-in, log-out)
  2. Infinite scrolling canvas with drawing tools
  3. Per-user dashboard for managing saved canvases
  4. Inter-user interaction (send canvases to other users)

## 3 Functional Requirements

### 3.1 Start Page



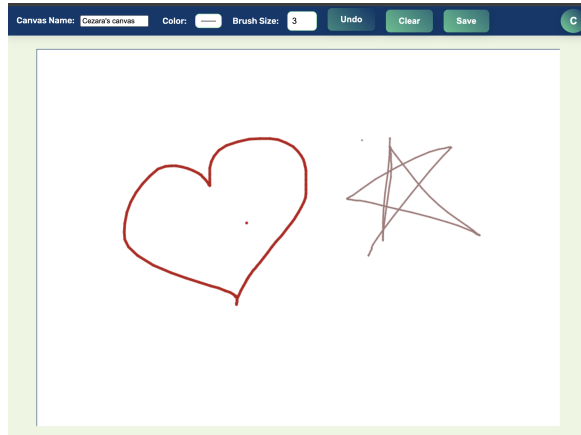
- **Sign Up:** The button creates a new user account (username + password) stored in the local memory of the browser and automatically signs you in, displaying a welcome message.
- **Log In:** Button opens credential prompt; checks the user and password, then:
  - Displays 'Welcome back!'
  - Sets a `currentUser` session variable.



- **Start Button:** Opens a new empty canvas named *Canvas 1*.
- **Dashboard Button:** Redirects to the user's dashboard page.

Note that users and their canvases are only stored locally as we have been instructed that working on the server-side is beyond the curriculum covered so far.

## 3.2 Notebook Page (Canvas Editor)



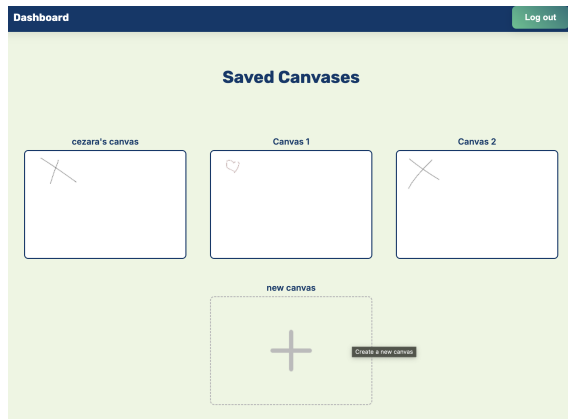
### Toolbar Components

- **Color Picker:** HTML input for brush color.
- **Size Selector:** HTML input for brush size.
- **User-Initial Button:** Circular, displays current user's initial (or "?" if none); hover over the tooltip "Go to dashboard"; click redirects to dashboard.
- **Canvas Name Textbox:** Editable label; defaults to "Canvas 1", subsequent canvases named "Canvas X" with the first unused index; responsive to resizing.
- **Clear Button:** Clears all strokes.
- **Undo Button:** Reverts last stroke.
- **Save Button:**
  - If new: appends to saved list.
  - If existing: overwrites existing entry.

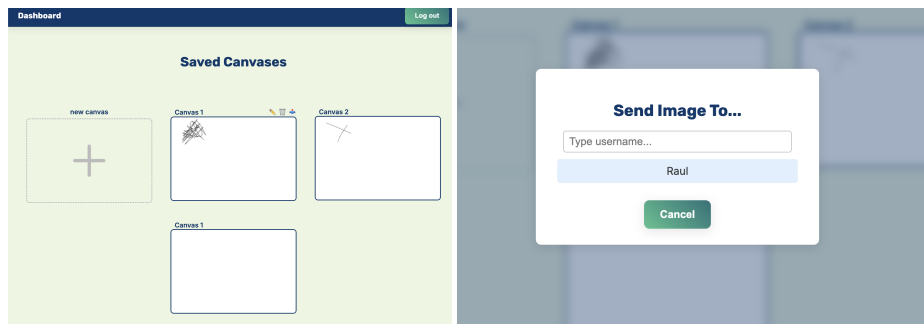
### Drawing Canvas

- Initial virtual size: **3000 × 2000 pixels**.
- Viewport smaller; user scrolls to pan.
- Expands horizontally/vertically when scrolling beyond bounds.

### 3.3 Dashboard



- **Log Out Button:** Clears session and redirects to Start Page.
- **Saved Canvases Grid:**
  - Dynamically displays thumbnails and names of all canvases for the current user.
  - Layout adjusts to viewport.
  - “Create New Canvas” dummy file opens a new canvas.
  - Clicking a saved file opens it in the Notebook Page.
- **Canvas hover options:** these show when hovering close to the name of the canvas
  - Edit name button
  - Delete button
  - Send button



## 4 Data Management

- **Storage:** Browser localStorage.
- **User Records:**

```
{
  "username": {
    "password": "<password>",
    "canvases": {
      "Canvas 1": "<serialized_data>",
      "Canvas 2": "<...>"
    }
  }
}
```

- **Session State:** currentUser variable holds logged-in username.

## 5 Naming Conventions & Versioning

- **New canvases:** “Canvas 1”, “Canvas 2”, etc.. A new canvas always receives the lowest positive integer that isn’t present in the other canvases with default names.
- **Saving an existing canvas** replaces its data.

## 6 User Flow Summary

1. **New User:** Sign up → Start Page → Start Button → Notebook Page with “Canvas 1”.
2. **Returning User:** Log in → “Welcome back!” → choose Start or Dashboard.
3. **Working on Canvas:** Draw → Save → Dashboard → view updated list.
4. **Managing Canvases:** Dashboard → select canvas or create new → edit → log out.

## 7 Technical Considerations

- **Technologies:** HTML5 <canvas>, JavaScript and CSS.
- **Responsive Design:** Toolbar and grid adapt to screen size.
- **Interactive:** Users can send eachother canvases.
- **Performance:** Efficient serialization/deserialization; debounce scroll/resize events.

## 8 Conclusion

This report lists the essential, functional requirements for the canvas drawing application without extra embellishments. Implementation should follow these specifications directly.