

Chrome Extension Development Exercise

Update: Oct-2024

Overview

1. In this coding exercise, you will create a Chrome browser extension that displays organization-wide messages from an admin to users.
2. This exercise is designed to evaluate your frontend development skills, understanding of browser extension architecture, and ability to implement real-world features.
3. Please don't spend more than 2h on this assignment.

Requirements

Functional Requirements

1. Create a Chrome extension that displays messages from an organization's admin to users
2. The extension should:
 - Show a badge icon when there are unread messages
 - Display messages in a popup when the extension icon is clicked
 - Allow users to mark messages as read
 - Store message history locally

Technical Requirements

1. Backend Integration

- Mock an API endpoint that would provide messages in this format:

```
```json
{
 "messages": [
 {
 "id": "msg123",
 "content": "Team meeting at 3 PM today 😊",
 "priority": "high",
 "timestamp": "2024-09-30T15:00:00Z",
 "read": false
 }
]
}
```
```

2. Frontend Implementation

- Use React/VUE for the popup UI
- Implement proper state management
- Style the interface using CSS (Tailwind/Bootstrap is a plus)

3. Extension Architecture

- Implement a background script to check for new messages periodically
- Use Chrome storage API to persist message state
- Handle different states: loading, error, no messages, unread messages

Evaluation Criteria

- Code organization and architecture
- State management and data flow
- Error handling and edge cases
- UI/UX considerations
- Performance optimization
- Documentation
- Testing
- CI/CD

Bonus Points

- Implement message categories or priority levels
- Add notification sounds for high-priority messages
- Create a options page for user preferences
- **Write (more types of) tests**
- **CI/CD**

Time Allocation

- Please don't spend more than ~2h to complete this exercise
- Focus on core functionality first, then move to bonus features if time permits

Submission Guidelines

1. Create a GitHub repository with your solution
2. Include:
 - All source code
 - A README with:
 - Setup instructions + Installation and testing
 - Any assumptions made

- Architectural decisions
 - Future improvements
3. Provide a short video or screenshots demonstrating the working extension

Getting Started

1. Create a new Chrome extension manifest (v3)
2. Set up a React/Vue development environment
3. Implement the basic extension structure:

```
extension-folder/  
├── manifest.json  
├── background.js  
├── popup/  
│   ├── popup.html  
│   ├── popup.js  
│   └── popup.css  
├── options/  
│   ├── options.html  
│   └── options.js  
└── assets/  
    └── icons/  
...
```

Interview Follow-up

Be prepared to discuss:

- Your architectural decisions
- How you would scale this for a large organization
- Security considerations
- Performance optimizations
- How you'd handle offline functionality

Good luck!

We're excited to see your solution.