COMETCHAT INTERNSHIP TASK REPORT

React UI Kit Integration – Complete Chat Application

Candidate: *Darshan Kumar K*
Date: *November 2025*

Table of Contents

1. Introduction

This document covers my work for the **CometChat Internship Task**, where I integrated the **CometChat React UI Kit** into a fully functioning chat application. While completing the task, I explored multiple areas of the CometChat ecosystem — the dashboard, UI Kit Builder, documentation, and sample app — and documented all the issues I encountered along with suggestions on how the user experience can be improved.

The goal of this report is to present:

- My implementation process

- Screenshots of the working application

- A detailed list of issues observed

- Practical recommendations based on real usage

2. Objective

The key objectives of this internship task were:

- Build a working React chat app with CometChat's UI Kit

- Configure and initialize the SDK with App ID, Region, and Auth Key

- Implement user login using UID

- Display Chats, Users, Groups, and Calls

- Explore CometChat's UI Kit Builder and documentation

- Identify issues and suggest improvements

I completed all required steps and documented everything in detail.

3. Tools & Technologies Used

- **React** — UI and component structure

- **CometChat React UI Kit** — Pre-built chat, calling, and user components

- **JavaScript / TypeScript**

- **Node.js & npm**

- **VS Code**

- **Google Chrome**

- **Git + GitHub** — Version control

4. Implementation Summary

To complete the task, I used the CometChat React UI Kit (sample app) and connected it with my CometChat dashboard credentials.
I created a test user (UID: **user1**) and verified the functionality end-to-end.
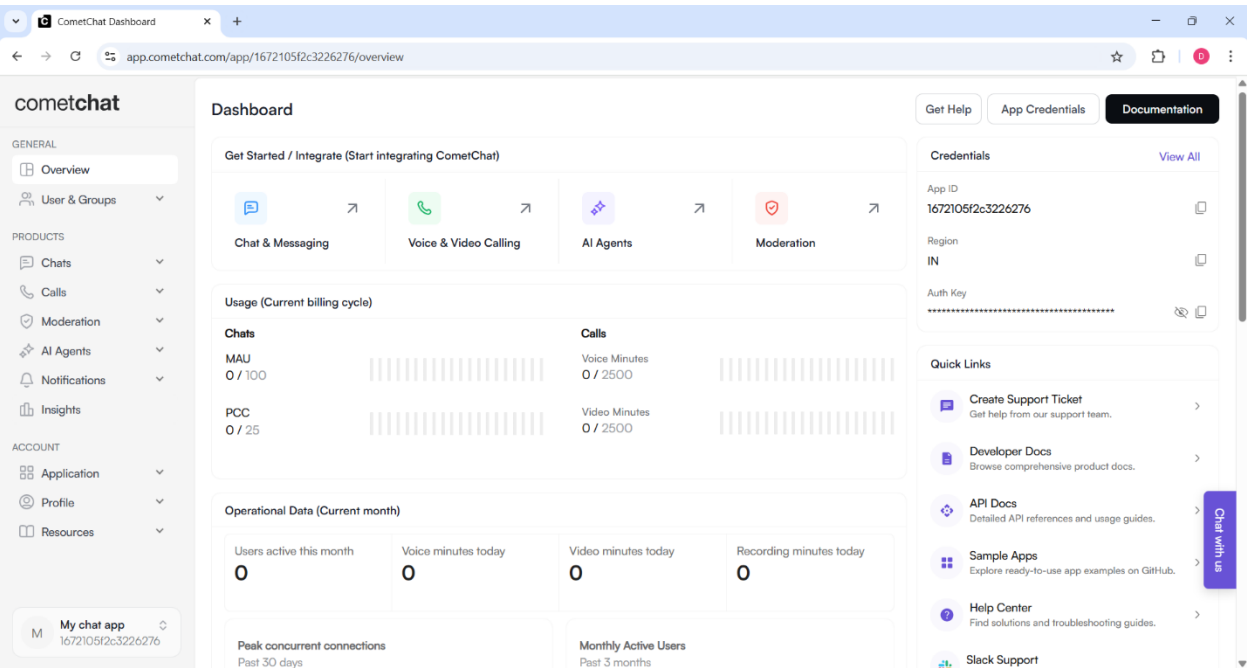
Successful Features Implemented

- **User Login** using UID

- **One-on-One Chats** with real-time updates

- **Typing Indicators**

- **User List** with navigation

- **Groups** (view, join, interact)

- **Calls** (voice/video UI integrated)

- **Smooth navigation** between Chats, Users, Groups, and Calls

The app behaved exactly as expected after connecting correct credentials and logging in with a real UID.
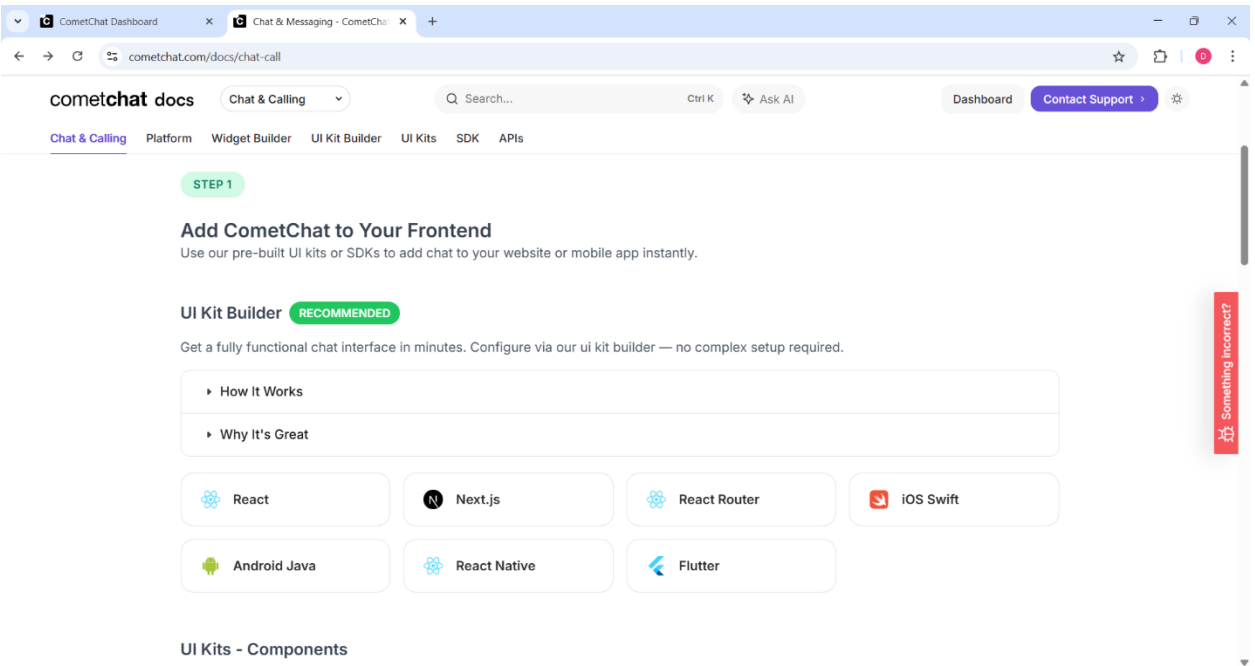
5. Screenshots

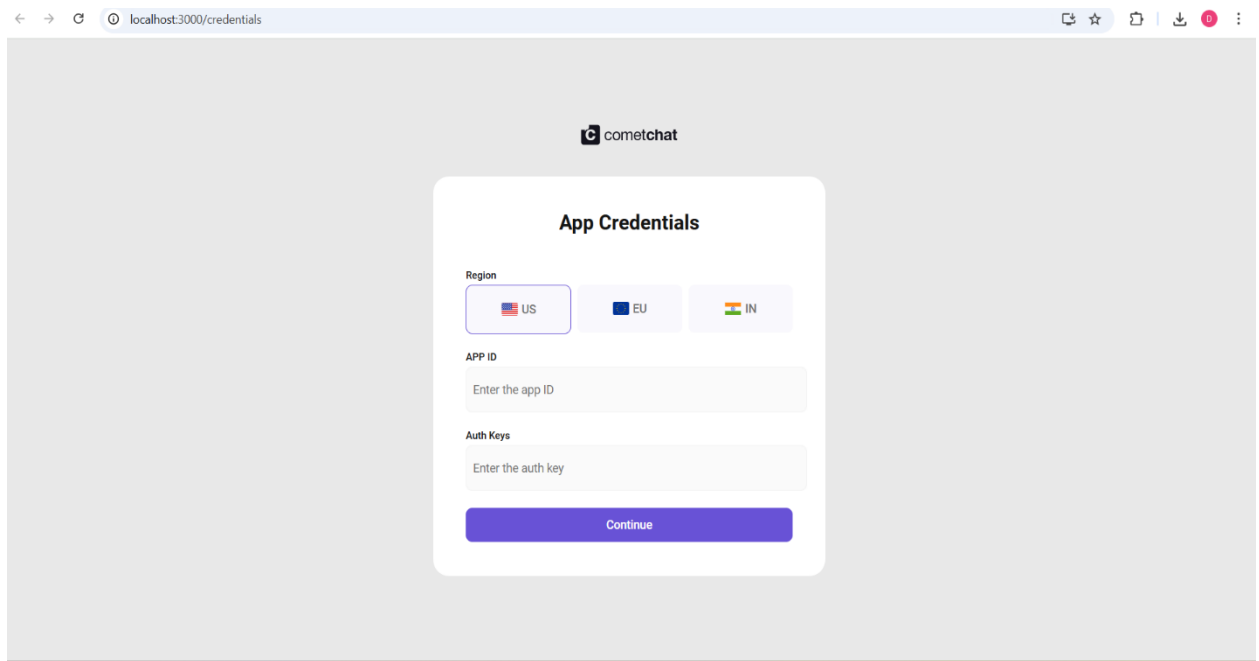*Insert your screenshots in this section. Recommended order:*
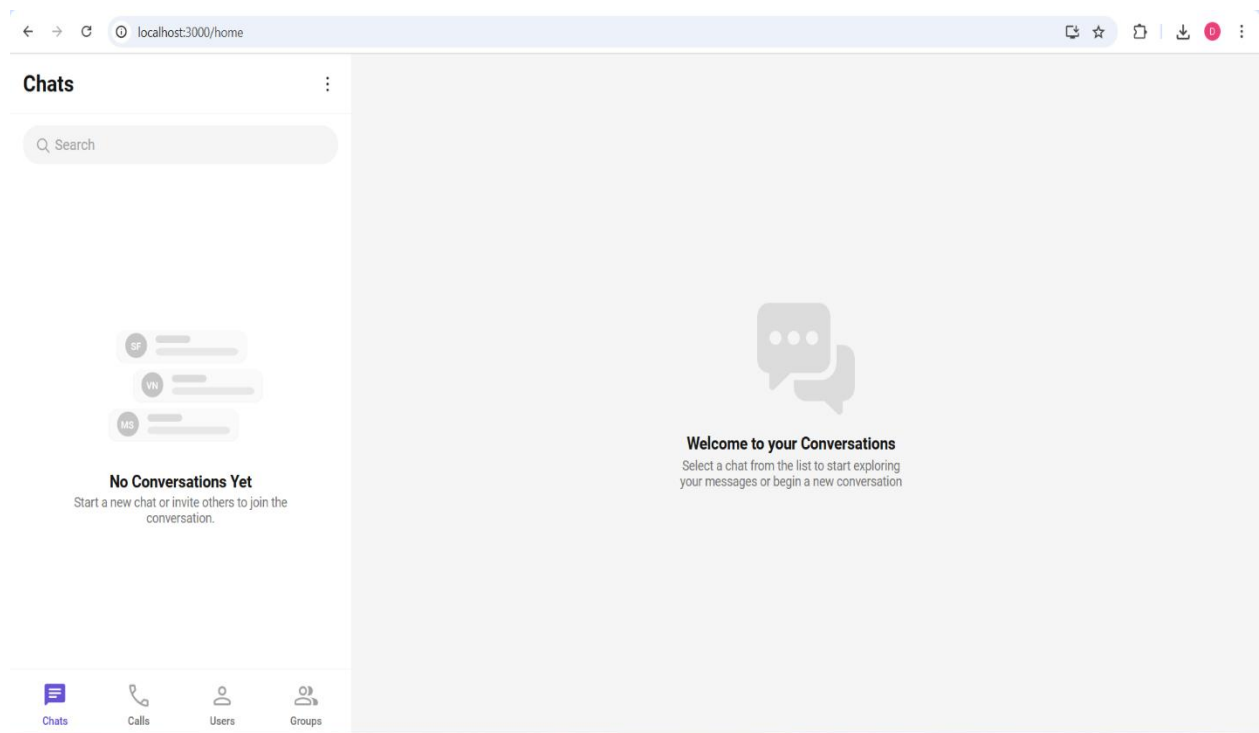
1. Dashboard – App Credentials

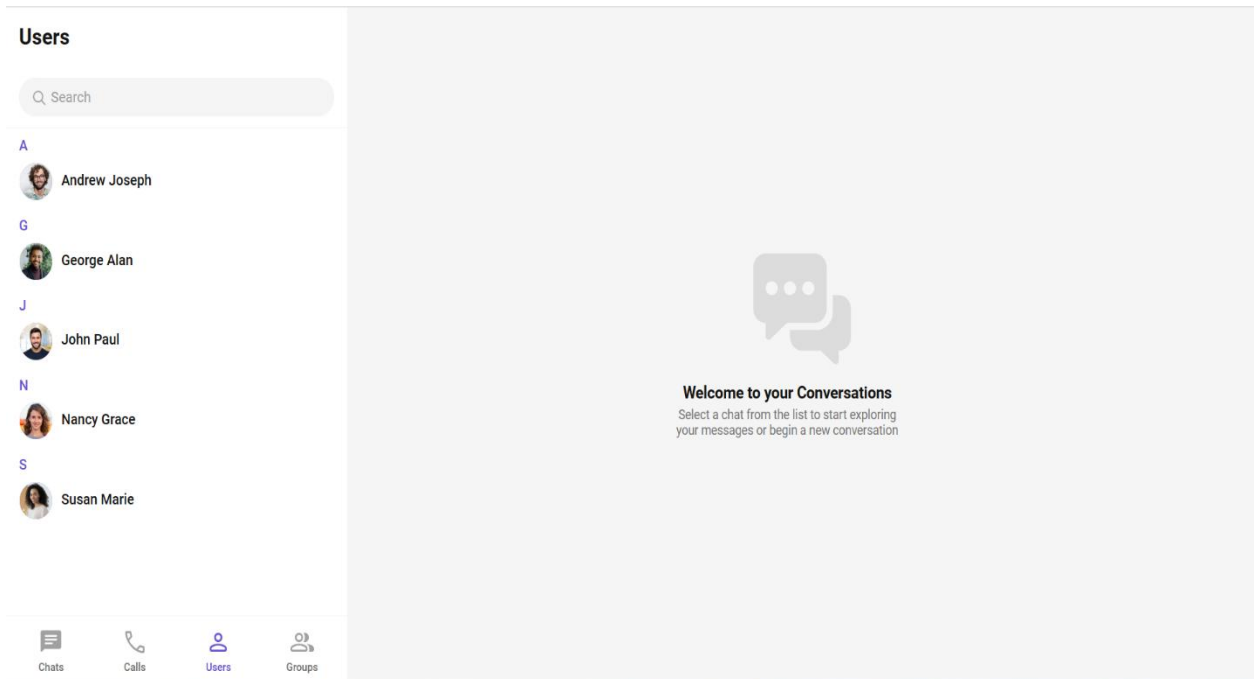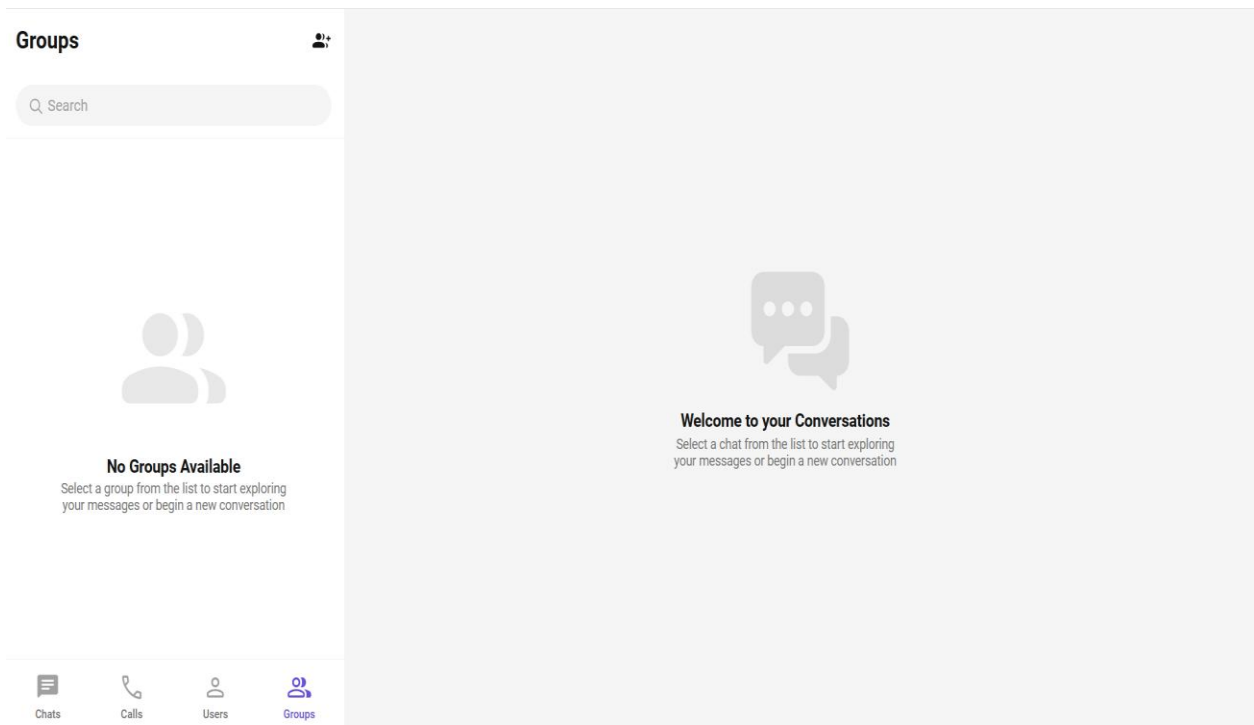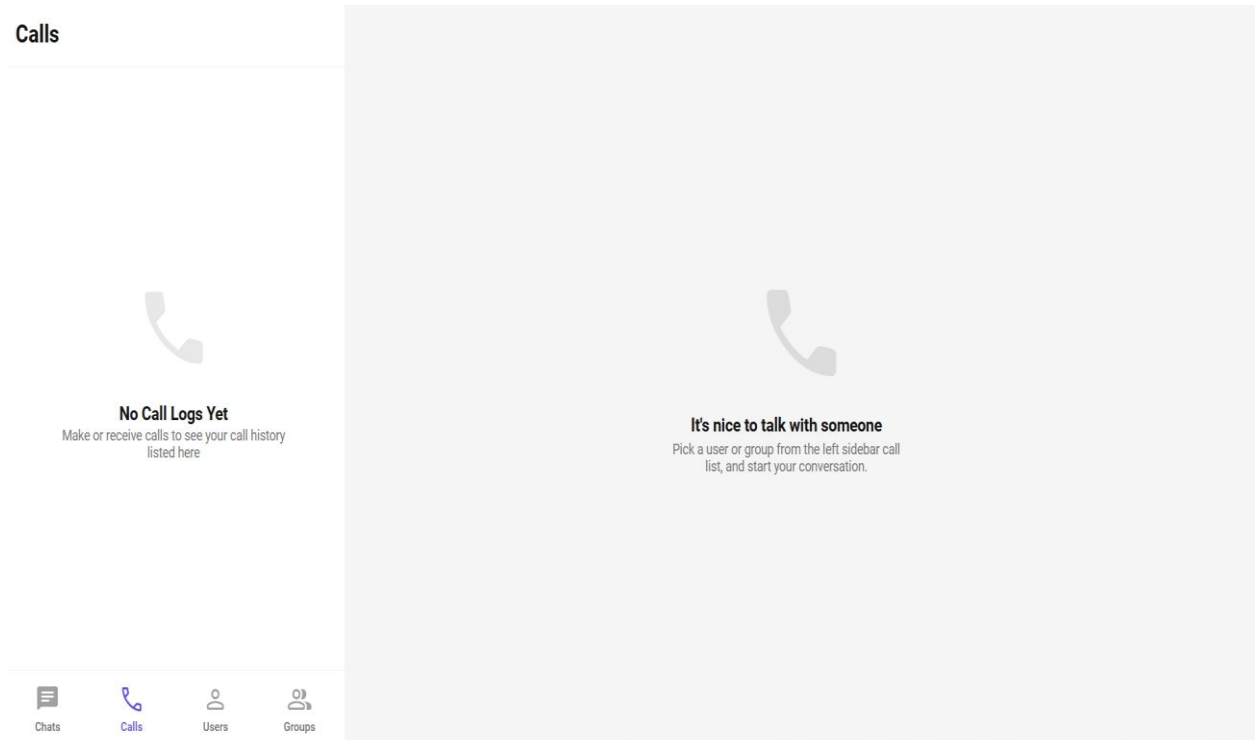2. UI Kit Builder screen

3. Login Screen

4. Chats Screen

5. Users Scree

6. Groups Screen

7. Calls Screen

Screenshots give a clear demonstration that the implementation is working.

6. Issues Identified & Recommendations

This part of the task was extremely valuable because I faced real issues while trying to get everything running.

I've categorized the issues as required: Dashboard, UI Kit Builder, Documentation, and Implementation.

6.1 Dashboard Issues

Issue 1 — UI Kit Builder option missing

In some apps, the **"Integrate → UI Kit Builder"** option doesn't appear.
This makes developers wonder if the feature even exists.

Suggestion:
Show the UI Kit Builder option for all apps.
If something is unavailable, show a short explanation.

Issue 2 — No quick-start section

Important actions (download UI kit, sample app link, credentials) are not placed together, which forces developers to jump between pages.

Suggestion:
Add a "Get Started" panel with quick links right inside the dashboard.

6.2 UI Kit Builder Issues

Issue 1 — Export Code does not download

Clicking **Export Code → React** doesn't actually download anything.
Instead, it shows a login popup even when already logged in.

Suggestion:
Enable direct download when user is authenticated or provide a fallback manual download link.

Issue 2 — 404 errors for React export

React export links generated by the UI Kit Builder often return **404 Not Found**.

Suggestion:
Fix or validate URLs before showing them to users.

6.3 Documentation Issues

Issue 1 — Wrong / outdated npm package names

Docs mention packages like:

npm install @cometchat/uikit-react

but these don't exist.

This leads to installation errors and confusion.

Suggestion:
Update all UI Kit v6 installation commands and remove deprecated package references.

Issue 2 — Sample app steps not clear

The documentation assumes users know where or how to run the sample app.

Suggestion:
Add a simple step-by-step guide:

1. Download Zip

2. Go to sample-app

1. Install dependencies

2. Add credentials

3. Run app

This would save developers a lot of time.

6.4 Implementation Issues

Issue 1 — Sample users cause crashes

The sample users shown inside the sample app (e.g., cometchat-uid-1) do not exist in new developer accounts.
Clicking them causes a login failure and a crash.

Suggestion:
Either auto-create sample users OR hide them for new apps.

Issue 2 — Error messages are not user-friendly

When login fails, the app shows messages like:

[object Object]

which is impossible to understand.

Suggestion:
Show readable errors:
"User does not exist. Please create a user in dashboard."

7. Recommendations Summary

Improving the developer experience would be easy and impactful:

- Add a universal **download UI kit** button

- Fix UI Kit Builder export for React

- Update documentation for installation and running sample app

- Improve sample app error handling

- Provide test users or a "quick import users" feature

- Add a simple onboarding wizard

8. Conclusion

I successfully completed the CometChat Internship Task by integrating the React UI Kit into a working chat application.
The experience helped me understand real-time communication, SDK integration, debugging, and proper project structure.

The issues I encountered gave me valuable insight into how developers approach SDK onboarding, and my suggestions aim to make CometChat even more developer-friendly.

I truly enjoyed building this project and would love the opportunity to continue learning and contributing as an intern at CometChat.

End of Report

Thank you for reviewing my work!