

# Chat Limit Feature Documentation

## Overview

The chat limit feature restricts the number of messages free users can send to the AI consultant. This document provides a comprehensive guide to how the chat limit works, where it appears in the UI, and how it integrates with the subscription plans.

## Chat Limit Implementation

Feature	Description	Screenshot Location
Initial Chat Count	New users are given 10 free messages upon registration	User registration screen
Remaining Messages Display	Shows the number of remaining messages for free users in the chat interface	Bottom of chat input area
Message Decrement	Each message sent by a free user decreases their chat count by 1	Chat interface
Limit Warning	When users have 3 or fewer messages left, the count is displayed in red	Chat input area
Limit Reached	When users have 0 messages left, they are prompted to upgrade to continue	Chat interface and redirect to pricing page
Unlimited Messages	Users with Basic or Premium plans have unlimited messages	N/A

## User Interface Elements

### Chat Input Area

Element	Description	Visibility Condition
Message Counter	Shows "You have X free messages remaining"	Only visible for free users
Warning Text	Displays in red when 3 or fewer messages remain	Only visible when count $\leq$ 3
Upgrade Prompt	Shows when all messages are used	Only visible when count = 0
Character Counter	Shows current message length/maximum allowed (1000)	Visible for all users

## Error Messages

Message	Trigger	Action
"You've reached your free message limit. Please	User with 0 messages	Redirects to pricing page

upgrade to a premium plan to continue chatting."	tries to send a message	
"Your message exceeds the maximum length of 1000 characters. Please shorten it."	Message exceeds character limit	Prevents message from being sent

## Subscription Plan Integration

Plan	Chat Access	Message Limit	Price
Free (No Plan)	Limited	10 messages	₹0
Basic	Unlimited	No limit	₹[Basic Plan Price]
Premium	Unlimited	No limit	₹[Premium Plan Price]
Pay-Per-Call	Unlimited	No limit	₹[Pay-Per-Call Price]

## Technical Implementation

### User Data Structure

The chat count is stored in the user's profile in Firebase:

```
// User data structure with chat count
{
  uid: string,
  email: string,
  displayName: string,
  // ... other user fields
  chatCount: number, // Number of remaining messages for free users
  plan: string | null // User's subscription plan (null, 'basic', 'premium', 'p.
```

### Chat Count Management

Operation	Code Location	Description
Initialization	authSlice.ts	New users get 10 free messages on registration
Decrement	chatSlice.ts	Chat count decreases by 1 when a free user sends a message
Check	ChatInterface.tsx	Before sending a message, system checks if user has remaining messages
Update	authSlice.ts	Redux state and Firestore database are updated when chat count changes

## Code Snippets

### Checking Chat Count Before Sending Message

```
// Check if user has a premium plan
const hasPremiumPlan = user.plan === 'premium' || user.plan === 'basic';

// If user doesn't have a premium plan, check chat count
if (!hasPremiumPlan && (user.chatCount === undefined || user.chatCount <= 0)) {
```

```

    toast.error("You've reached your free message limit. Please upgrade to a premium plan.");
    setInputLoading(false);
    navigate('/pricing');
    return;
}

```

## Displaying Remaining Messages

```

{/* Message count indicator for non-premium users */}
{user && !user.plan && (
  <div className="mb-2 text-xs text-center">
    <span className={`font-medium ${user.chatCount || 0} > 3 ? 'text-gray-600' : 'text-gray-500'}>
      {user.chatCount || 0} > 3 ? `You have ${user.chatCount} free messages. Please upgrade to continue.` :
      : 'You have used all your free messages. Please upgrade to continue.'}
    </span>
    {user.chatCount || 0} <= 3 && (user.chatCount || 0) > 0 && (
      <Link to="/pricing" className="ml-1 text-indigo-500 hover:text-indigo-600">
        Upgrade
      </Link>
    )
  </div>
)}

```

## Decrementing Chat Count

```

// Only decrement chat count for non-premium users
if (!hasPremiumPlan && userData.chatCount !== undefined) {
  // Calculate new chat count
  const newChatCount = Math.max(0, userData.chatCount - 1);

  // Update Firestore
  await updateDoc(userRef, {
    chatCount: newChatCount
  });

  // Update Redux state
  dispatch(updateChatCount({
    uid: userId,
    chatCount: newChatCount
  }));
}

```

# User Flow

## 1. New User Registration

- User creates an account
- System initializes `chatCount` to 10
- User can send up to 10 messages to the AI

## 2. Using Free Messages

- Each message sent decrements `chatCount` by 1
- UI displays remaining message count

- When count reaches 3 or fewer, warning appears in red

### 3. Limit Reached

- When `chatCount` reaches 0, user cannot send more messages
- System shows error message and redirects to pricing page
- User must upgrade to continue using the chat

### 4. Subscription Upgrade

- User purchases Basic or Premium plan
- Chat limit is removed, allowing unlimited messages
- Message counter no longer appears in the UI

## Implementation Checklist

- [x] Initialize chat count for new users
- [x] Display remaining messages in chat interface
- [x] Decrement count when messages are sent
- [x] Show warnings when count is low
- [x] Block messages when count reaches zero
- [x] Redirect to pricing page when limit is reached
- [x] Remove limits for paid subscription users

## Screenshot Placement Guide

Screenshot Description	UI Location	Key Elements to Highlight
New User Chat Interface	Chat page for free user	Message counter showing 10 messages
Low Count Warning	Chat input area	Red text showing low message count
Limit Reached Error	Toast notification	Error message about reaching limit
Pricing Page Redirect	Pricing page	Available subscription options
Unlimited Chat (Paid User)	Chat interface for paid user	Absence of message counter

## Testing Instructions

1. Create a new test user account
2. Verify initial chat count is set to 10
3. Send messages and confirm count decrements correctly
4. Test warning appearance when count is low
5. Verify user cannot send messages when count reaches 0
6. Test upgrade flow to ensure limits are removed with subscription

## Additional Notes

- The chat limit only applies to text messages, not to other features
- Character limit (1000 chars) applies to all users regardless of subscription

- Admin users are not subject to chat limits
- Chat count does not reset automatically - users must upgrade to continue after using all free messages