Assignment: Develop an instant messaging application

Submit by January 10, 11:59 PM PT

Important Information

It is especially important to submit this assignment before the deadline, January 10, 11:59 PM PT, because it must be graded by others. If you submit late, there may not be enough classmates around to review your work. This makes it difficult - and in some cases, impossible - to produce a grade. Submit on time to avoid these risks.

Instructions

In this peer assessment you will be working on an instant messaging app called **minstant!**

You are provided with a working starter application and you have to use the skills you have learned in the course to improve the application in various ways. The starter application can be downloaded below:

[minstant.zip](https://d3c33hcgiwev3.cloudfront.net/_ae4c17581bebbc8d03addcb99cc63e05_minstant.zip?Expires=1452038400&Signature=QpckSrT1b00RULAvQ3m~Lkwzvqvkt72WyVqwFMZmQaS9Wm-kWrtj27BwX5PVEu~JfFu1EHBPjqjfx1lWV7yzT23N5GwV3FDcAbqP7L4sbTTx3-kQBRMZS6kfoZzhAEWpJpfBtX2QY9oE16UBoFhpbKRf8eSLygnYmOlN9O4tTTQ_&Key-Pair-Id=APKAJLTNE6QMUY6HBC5A)

You should download and run the starter application, then test it out by logging in as different users in two different web browsers at the same time, and seeing if the users can talk to each other. It creates a set of test user accounts automatically, so you can log in as user1@test.com ... user8@test.com with the password test123.

Task 1: Improve the look and feel

Adapt the templates and helper functions so that the messaging window displays users’ avatars next to their messages. Feel free to add other enhancements!

Task 2: Implement data writing security

Remove the insecure package from the application and implement a Meteor method to allow the insertion of chat items in the Chats collection. Test that you cannot insert items directly any more.

Task 3: Implement data reading security

Remove the autopublish package from the application and implement publish and subscribe for Chats. Users should only be able to retrieve chats that have their user id in either the user1Id field or the user2Id field. Test by logging in as different users and checking what you can see

Challenge: Implement emoticons

Can you implement emoticon functionality which allows the user to insert graphical emoticons into their message? Emoticons are small icons such as smiley faces which are typical of this kind of application.

Review criterialess

Your work will be reviewed by people using it and testing out the security in the browser console. Reviewers will be instructed to check that each of the required things have been implemented. 70% of the mark is available for the three main tasks. 30% extra can be gained by attempting the challenge. The mark is broken down as follows:

Submit a working link: 10%

Implement well laid out avatars in chat window: 20%

Implement methods to prevent direct inserts: 20%

Implement publish and subscribe to prevent users seeing conversations they were not involved in: 20%

Implement challenge: 30%

The pass mark is 60%.