

SE Factory

Week 8 - Assignments

version 01BD_1703

Duration: 24 hours

Change Log

- Amended the Notes item #2 (added support for channels)

WebSockets

Exercise 1

We have been discussing recently some networking concepts and now is the time for you to start tinkering. For this assignment you will need to build a **realtime chatting application**.

You will achieve this goal by building 3 main components:

1. A **WebSocket Server** built with PHP using this library (devristo/phpws) and by adopting sane OOP standards [You are not allowed to use other library options]
2. A **Laravel Project** that will act as your project's backend and will handle:
 1. User Registration & Authentication
 2. User Information and Chat Data Persistence in the DB
 3. Global Navigation / Access to the chatroom
3. A **Javascript App** that will be the main message box (chatroom) in which users type messages and see the messages of others
 1. This will have the same look and feel as Slack
 2. At a minimum, each message should be accompanied by a **timestamp** and **name** of the owner

Notes:

1. If a user logs-out of the session and logs back in after a while, he/she will be able to see the history of the chats that took place while he/she was gone
2. *Implement Channels and Administrative privileges (i.e. Roles) to create channels*
3. The app should easily support a **minimum** of 10 concurrent users chatting at the same time
4. No user can enter the chatroom without being authenticated

5. The application will be tested by having all current SE Factory students create accounts and attempt to chat at the same time

Wall of Fame:

If your project abides by SE Factory coding standards (that we've been discussing for the past 8 weeks) and If your application is able to handle file sharing you will get your name on the Wall.