Social Network

The Kata

Implement a console-based social networking application that satisfies the scenarios listed below.
The application must use the console for input and output.

Users submit commands to the application. Commands always start with the user's name.

Do not worry about handling any exceptions or invalid commands. Assume the user will always type the correct command.

Do not make it work over a network or across processes, do it all in memory assuming users are on the same terminal.

Non-existing users should be created as they post their first message. The application should not start with a predefined list of users.

User Behavior Scenarios

Implement the following:

Posting

Alice can publish messages to a personal timeline.

> Alice /post What a wonderfully sunny day!

Reading

Bob can view Alice's timeline

> Bob /timeline Alice

Following

Charlie can subscribe to Alice's timeline and view an aggregate list of all subscriptions

> Charlie /follow Alice

Wall

Charlie can view an aggregate list of all the people he has subscribed to follow.

> Charlie /wall

Note: If Charlie follows Alice and Bob he will see a time sequenced list of post from them on his wall. That is the post are displayed most recent at the top.

Mentions

Bob can link to Charlie in a message using '@'.

> Bob /post @Charlie what are your plans tonight?

Note: this is not a new command, just an expansion of how the /post command works. Any mentions should appear on a user's wall even if they do not follow the user.

Direct Messages

Mallory can send a private message to Alice.
> Mallory /send_message Alice

Alice can view all private messages
> Alice /view_messages