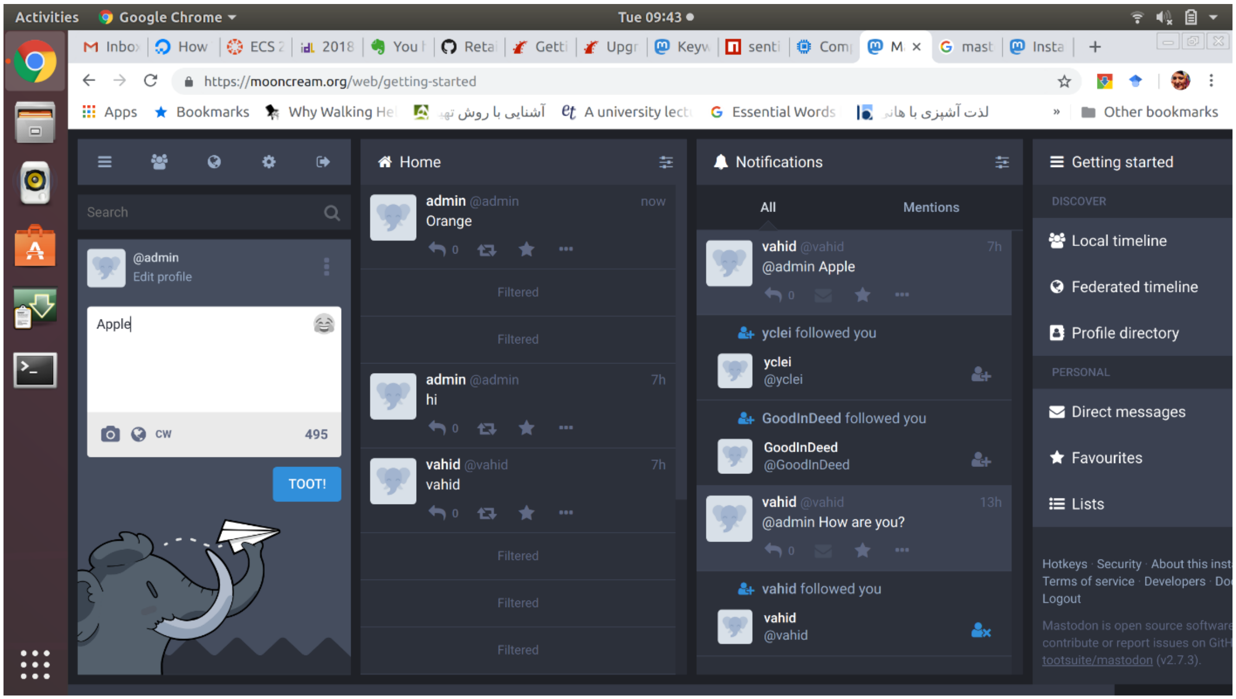
Week 3 Report

Fatemeh Radaei, Weijia Xing, Yuchu Lei

We have been working on Mastodon implementation, setting up a server and making some changes to the Mastodon system. We have also been working on improving the UI of current client system, inspired by Pinafore. We are also figuring out if it makes sense to add a section in the UI settings for our word filter system or in the Mastodon.

To test our idea, we found the section of the Mastodon code that deals with the users posts and messages. We added a simple filter that filters any message that is “Apple” and it is working :D

Next, we are going to add a sentiment analysis to the posts and messages and filter them based on their sentiment score. For example, if the sentiment score of the text is lower than a threshold, we will filter it.



@admin
Edit profile

Apple

495

admin @admin now
Orange

Filtered

Filtered

admin @admin 7h
hi

vahid @vahid 7h
vahid

Filtered

Filtered

Filtered

vahid @vahid 7h
@admin Apple

yclei followed you
yclei
@yclei

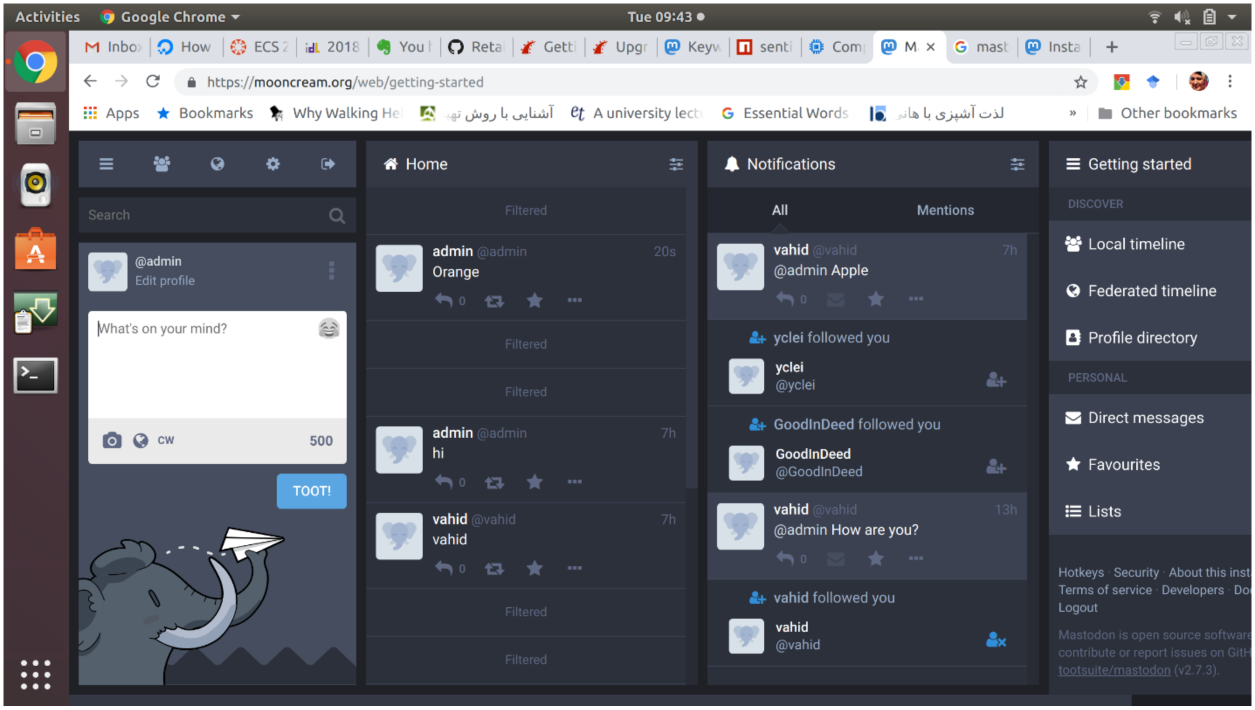
GoodInDeed followed you
GoodInDeed
@GoodInDeed

vahid @vahid 13h
@admin How are you?

vahid followed you
vahid
@vahid

Hotkeys · Security · About this inst
Terms of service · Developers · Do
Logout

Mastodon is open source software
contribute or report issues on GitH
tootsuite/mastodon (v2.7.3).



@admin
Edit profile

What's on your mind?

500

Filtered

admin @admin 20s
Orange

Filtered

Filtered

admin @admin 7h
hi

vahid @vahid 7h
vahid

Filtered

Filtered

vahid @vahid 7h
@admin Apple

yclei followed you
yclei
@yclei

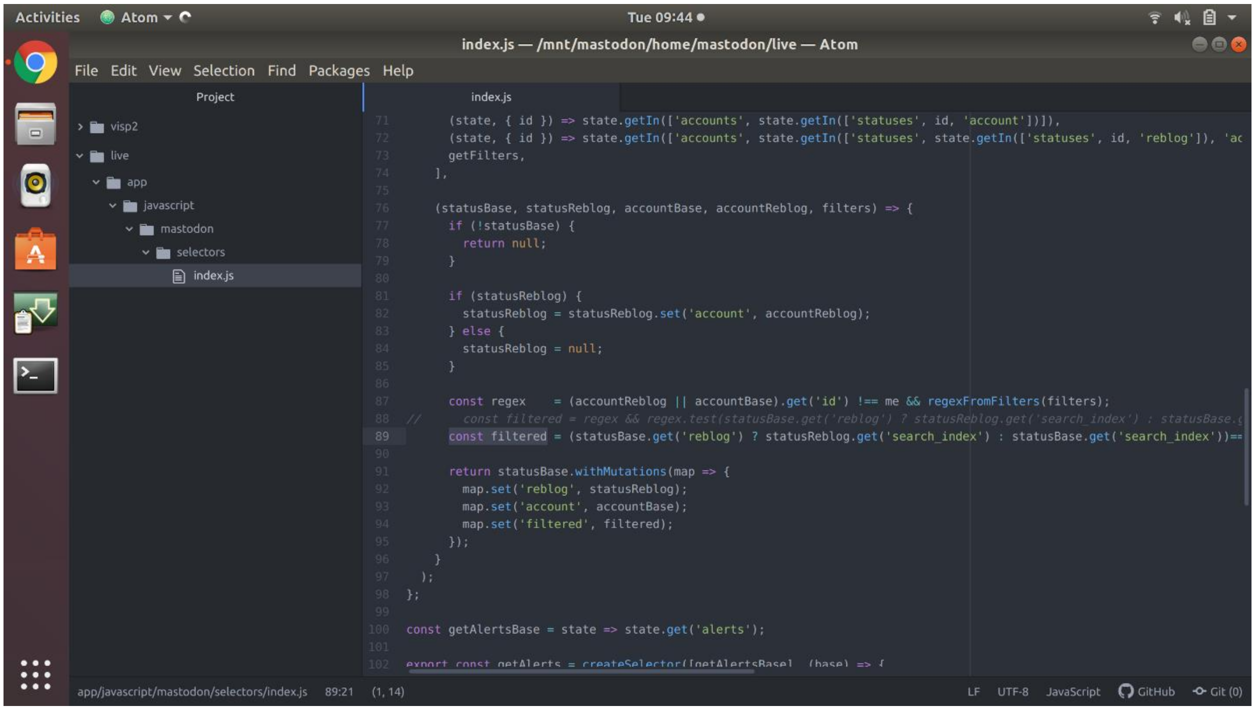
GoodInDeed followed you
GoodInDeed
@GoodInDeed

vahid @vahid 13h
@admin How are you?

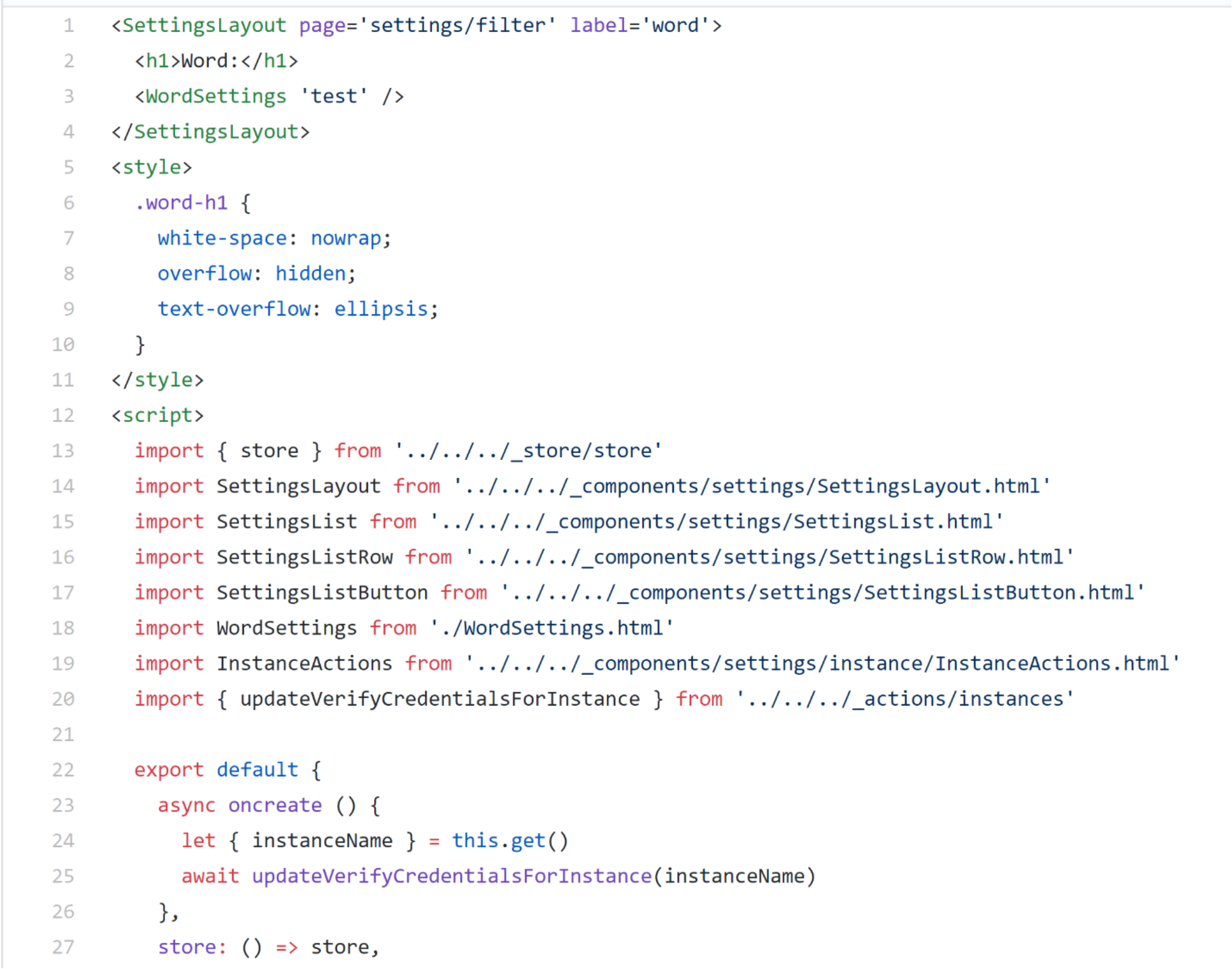
vahid followed you
vahid
@vahid

Hotkeys · Security · About this inst
Terms of service · Developers · Do
Logout

Mastodon is open source software
contribute or report issues on GitH
tootsuite/mastodon (v2.7.3).



```
      (state, { id }) => state.getIn(['accounts', state.getIn(['statuses', id, 'account'])]),
      (state, { id }) => state.getIn(['accounts', state.getIn(['statuses', state.getIn(['statuses', id, 'reblog']), 'ac
      getFilters,
    ],

    (statusBase, statusReblog, accountBase, accountReblog, filters) => {
      if (!statusBase) {
        return null;
      }

      if (statusReblog) {
        statusReblog = statusReblog.set('account', accountReblog);
      } else {
        statusReblog = null;
      }

      const regex    = (accountReblog || accountBase).get('id') !== me && regexFromFilters(filters);
//      const filtered = regex && regex.test(statusBase.get('reblog') ? statusReblog.get('search_index') : statusBase.g
      const filtered = (statusBase.get('reblog') ? statusReblog.get('search_index') : statusBase.get('search_index'))==

      return statusBase.withMutations(map => {
        map.set('reblog', statusReblog);
        map.set('account', accountBase);
        map.set('filtered', filtered);
      });
    }
  );
};

const getAlertsBase = state => state.get('alerts');

export const getAlerts = createSelector([getAlertsBase], (base) => {
```



```
<SettingsLayout page='settings/filter' label='word'>
  <h1>Word:</h1>
  <WordSettings 'test' />
</SettingsLayout>
<style>
  .word-h1 {
    white-space: nowrap;
    overflow: hidden;
    text-overflow: ellipsis;
  }
</style>
<script>
  import { store } from '../../../_store/store'
  import SettingsLayout from '../../../_components/settings/SettingsLayout.html'
  import SettingsList from '../../../_components/settings/SettingsList.html'
  import SettingsListRow from '../../../_components/settings/SettingsListRow.html'
  import SettingsListButton from '../../../_components/settings/SettingsListButton.html'
  import WordSettings from './WordSettings.html'
  import InstanceActions from '../../../_components/settings/instance/InstanceActions.html'
  import { updateVerifyCredentialsForInstance } from '../../../_actions/instances'

  export default {
    async oncreate () {
      let { instanceName } = this.get()
      await updateVerifyCredentialsForInstance(instanceName)
    },
    store: () => store,
```