a) Roy Xu

b) Adding Comments: Users select a content item that is visible to them.
The user enters a comment about that content item. On viewing a content item,
the corresponding comments are displayed in reverse chronological order.

c) Users may want to comment on different content items

d) A new table was added called Comment with item_id, comment_id, and commenter_email as the primary key. The table includes the comment body and the comment timestamp

```
e) SELECT * FROM Comment WHERE item_id=%s AND item_id IN
    (SELECT DISTINCT item_id FROM ContentItem WHERE item_id IN
         (SELECT item_id FROM Share INNER JOIN Belong ON
                 Belong.fg_name=Share.fg_name AND
                 Belong.owner_email=Share.owner_email AND Belong.email=%s
         ) OR is_pub OR email_post=%s
   ) ORDER BY comment_time DESC
```

Get all comments by item_id but check if item_id is one of the item_ids that are visible to the user. Also order by comment_time in descending order

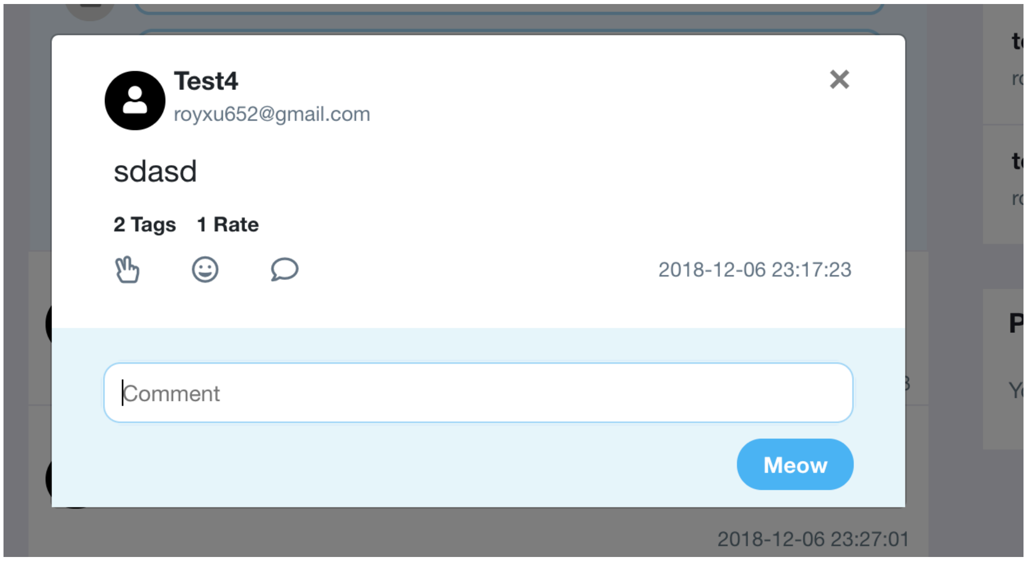
```
INSERT INTO Comment (item_id, comment, commenter_email, comment_time)
         VALUES(%s, %s, %s, NOW())
```

Insert into the comment table with the current time

f) Lines 386 - 408 in db.py
add_comment and get_comments
Lines 198 - 213 in app.py
get_comment and comment

Comments in code

g)



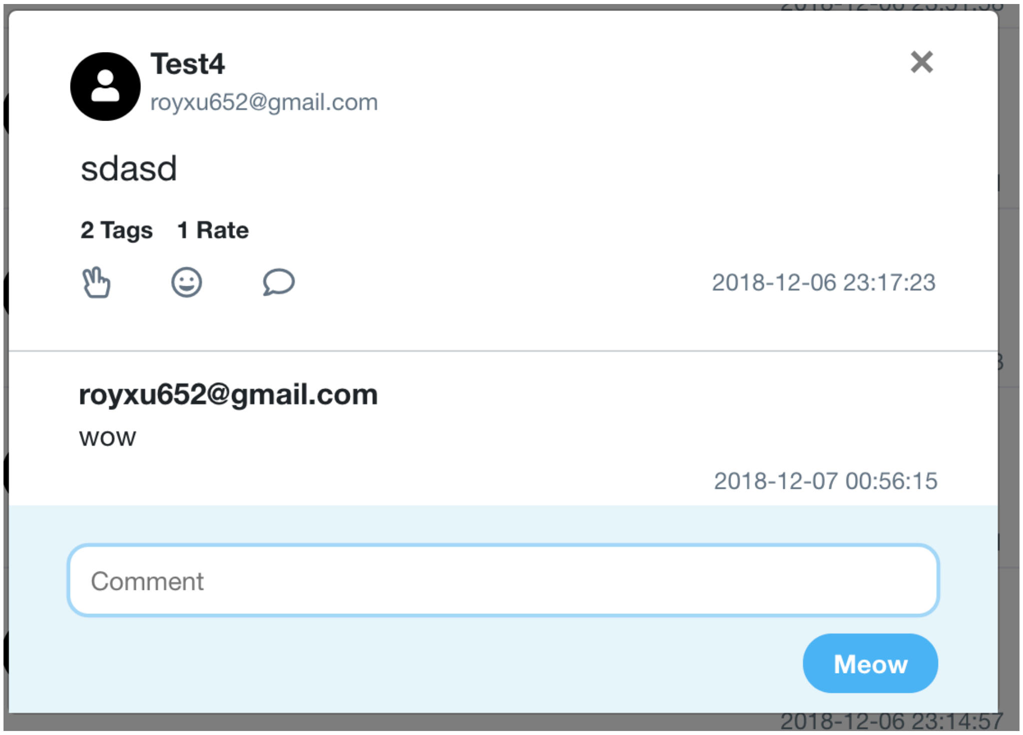
Test4
royxu652@gmail.com

sdasd

2 Tags 1 Rate

2018-12-06 23:17:23

Comment box



Test4
royxu652@gmail.com

sdasd

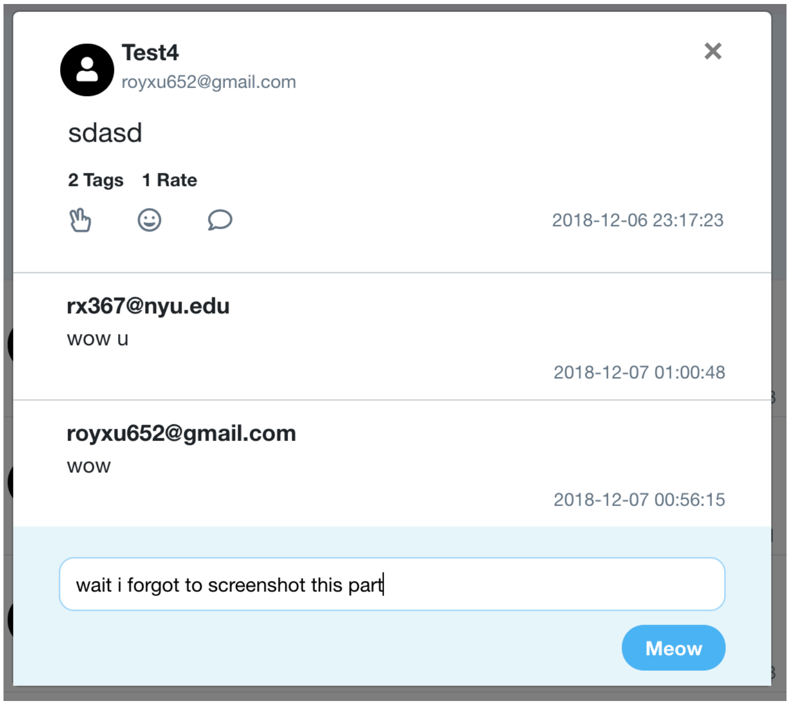
2 Tags 1 Rate

2018-12-06 23:17:23

royxu652@gmail.com
wow

2018-12-07 00:56:15

Posting a comment



Test4
royxu652@gmail.com

sdasd

2 Tags 1 Rate

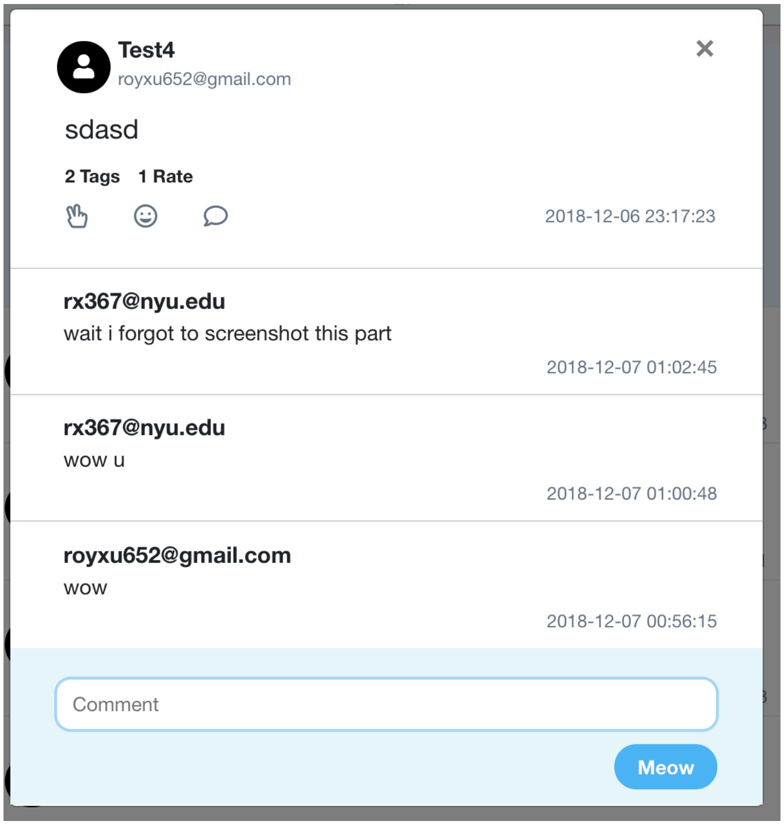
2018-12-06 23:17:23

rx367@nyu.edu
wow u
2018-12-07 01:00:48

royxu652@gmail.com
wow
2018-12-07 00:56:15

wait i forgot to screenshot this part

Another user posting a comment



Test4
royxu652@gmail.com

sdasd

2 Tags 1 Rate

2018-12-06 23:17:23

rx367@nyu.edu
wait i forgot to screenshot this part
2018-12-07 01:02:45

rx367@nyu.edu
wow u
2018-12-07 01:00:48

royxu652@gmail.com
wow
2018-12-07 00:56:15

All comments