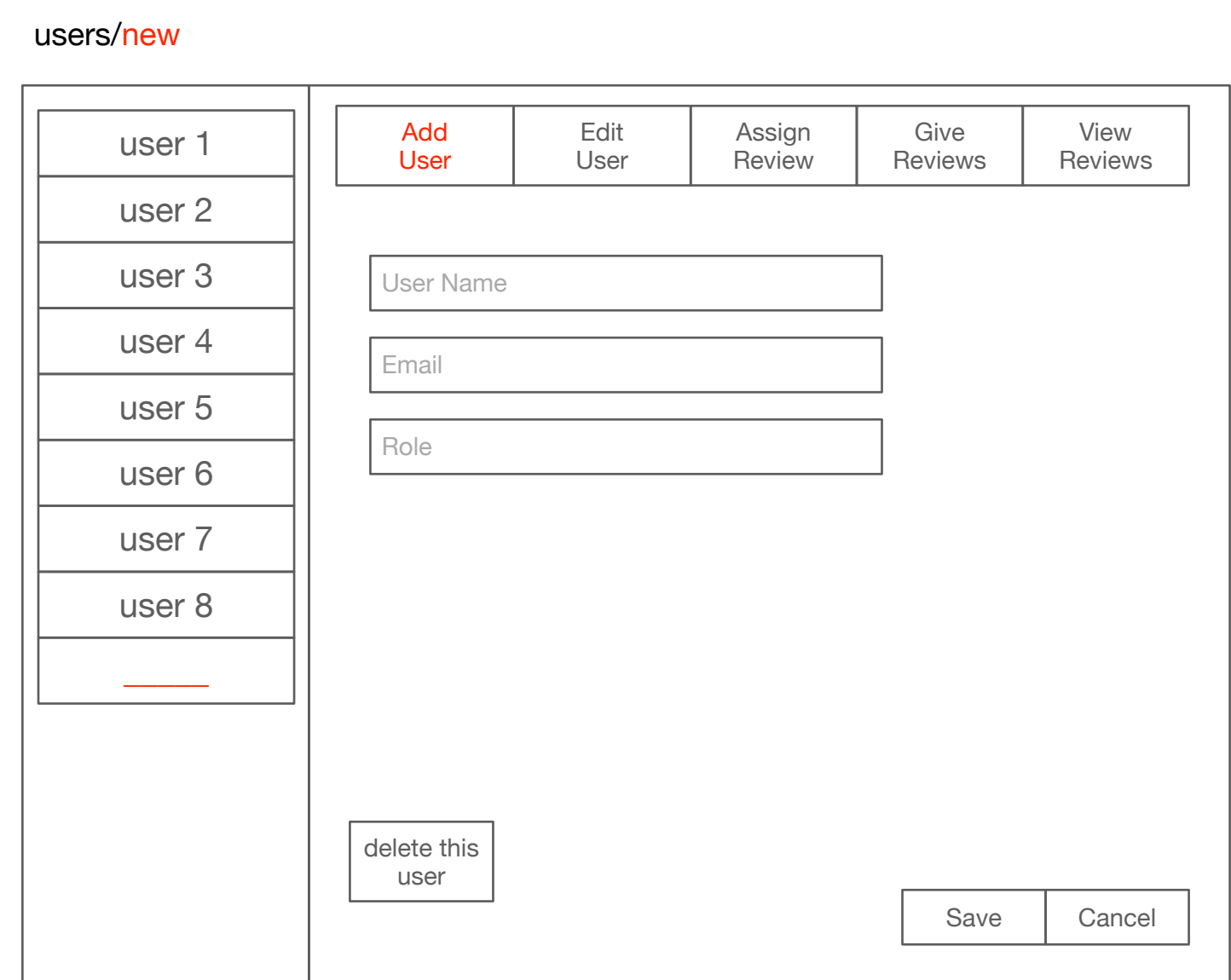


<UserView/>

onIntialized: GET api/id_user1

onDeleteBegin: DELETE api/users/id_user1

onDeleteSuccess: GET api/usersCatalog



<UserAdd/>

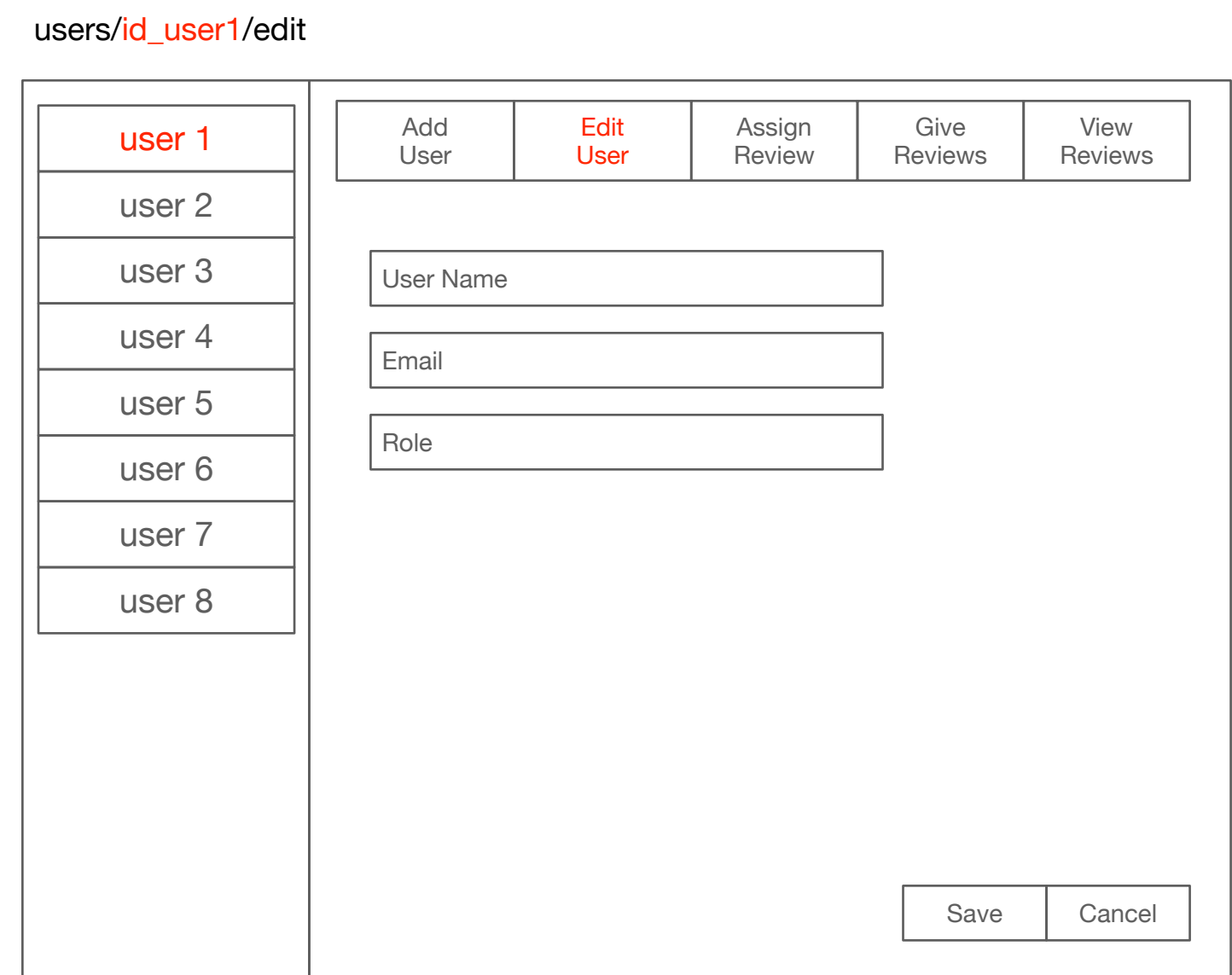
onSaveBegin: POST api/users

*api-server will create {assingment} on server and link its id_assignment to this user

onSaveSaveSuccess:

- navigate to view new user

- GET api/usersCatalog



<UserEdit/>

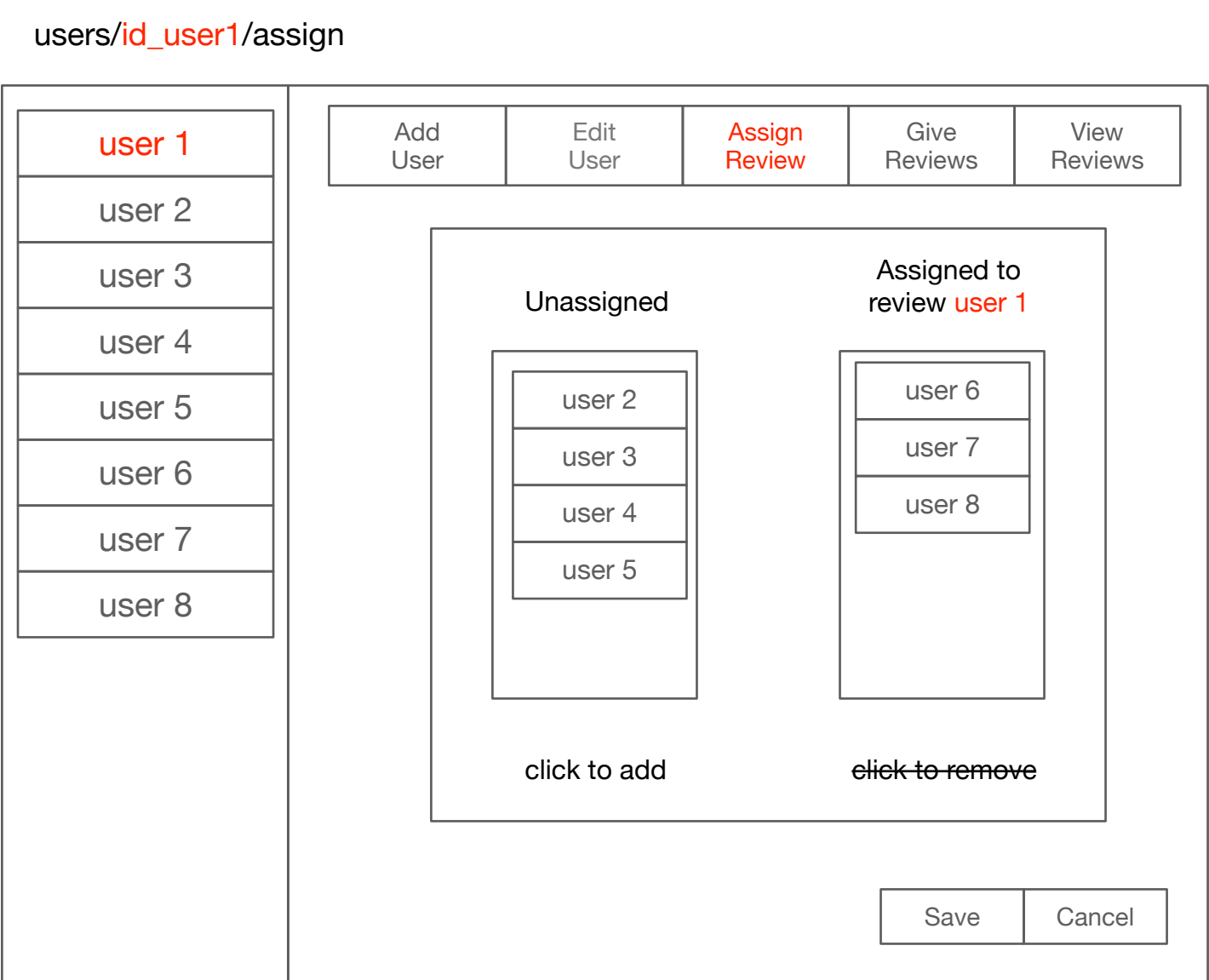
onIntialized: GET api/users/id_user1

onSaveBegin: PUT api/users/id_user1

onSaveSuccess:

- update {user_1} on users resources

- navigate to view user_1



<ReviewAssign/>

onIntialized:

- GET api/assignment/id1_assignment

onUserSelection:

add/remove id_user to draft_assignment.reviews

onSaveReviewBegin:

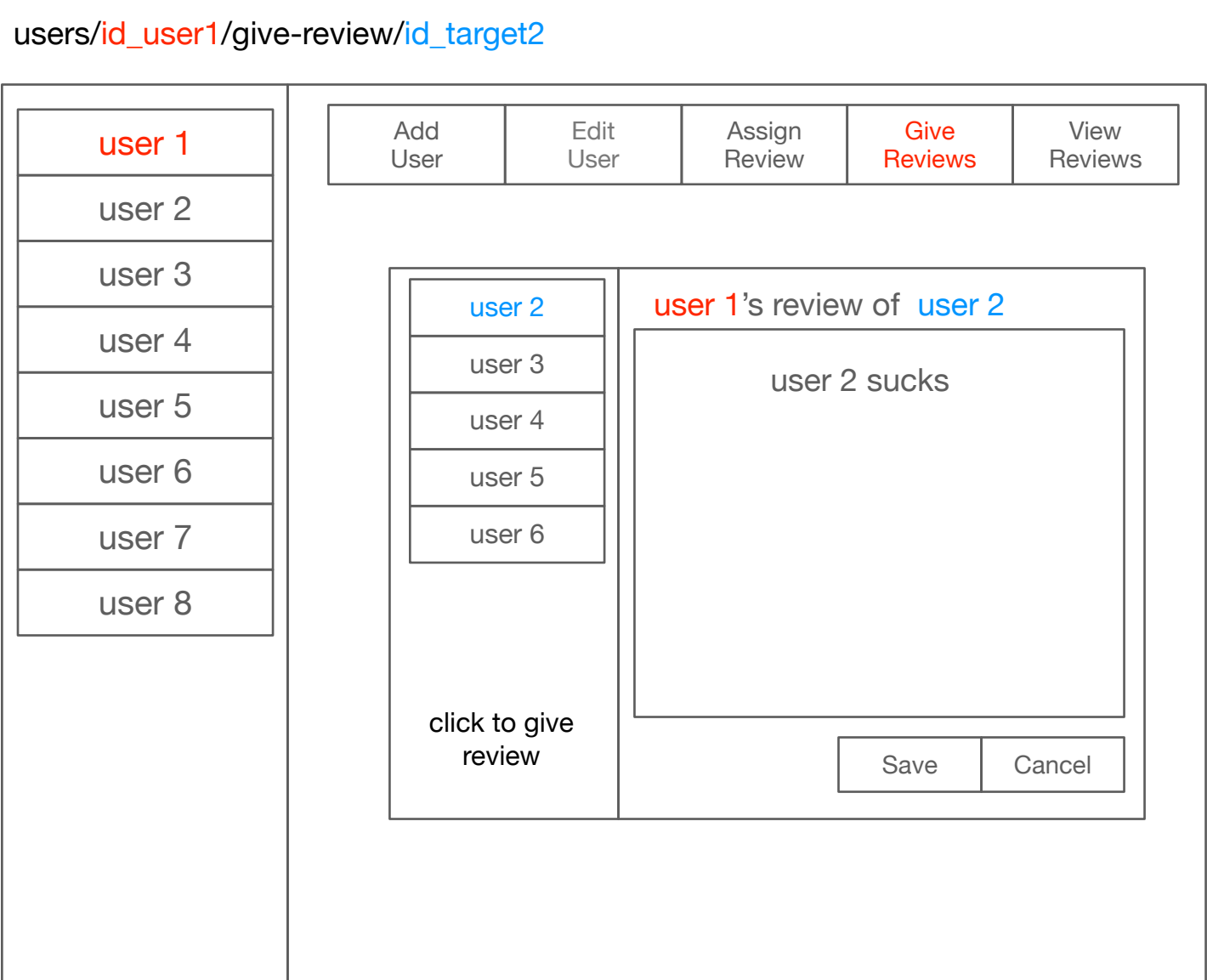
- Patch api/assignment/id1_assignment

onSaveSuccess:

- no special task need to exec

[!] client never create the {assignment}, it is created on server when user is created

[!] remove assignment is not implement



<ReviewEdit>

onIntialized:

- GET api/assignment/id1_assignment

- collection all missing user in local resource and do GET api/users/batch-get([id_user2, id_user3])

onUserSelection:

Get api/review/id_review1

onSaveReviewBegin:

has_id_review1 : PUT: api/reviews/id({draft_review1})

no_id_review: POST api/reviews({draft_reviewNew})

onSaveSuccess:

- no task need to exec

review_draft = {

_id: id_review7,

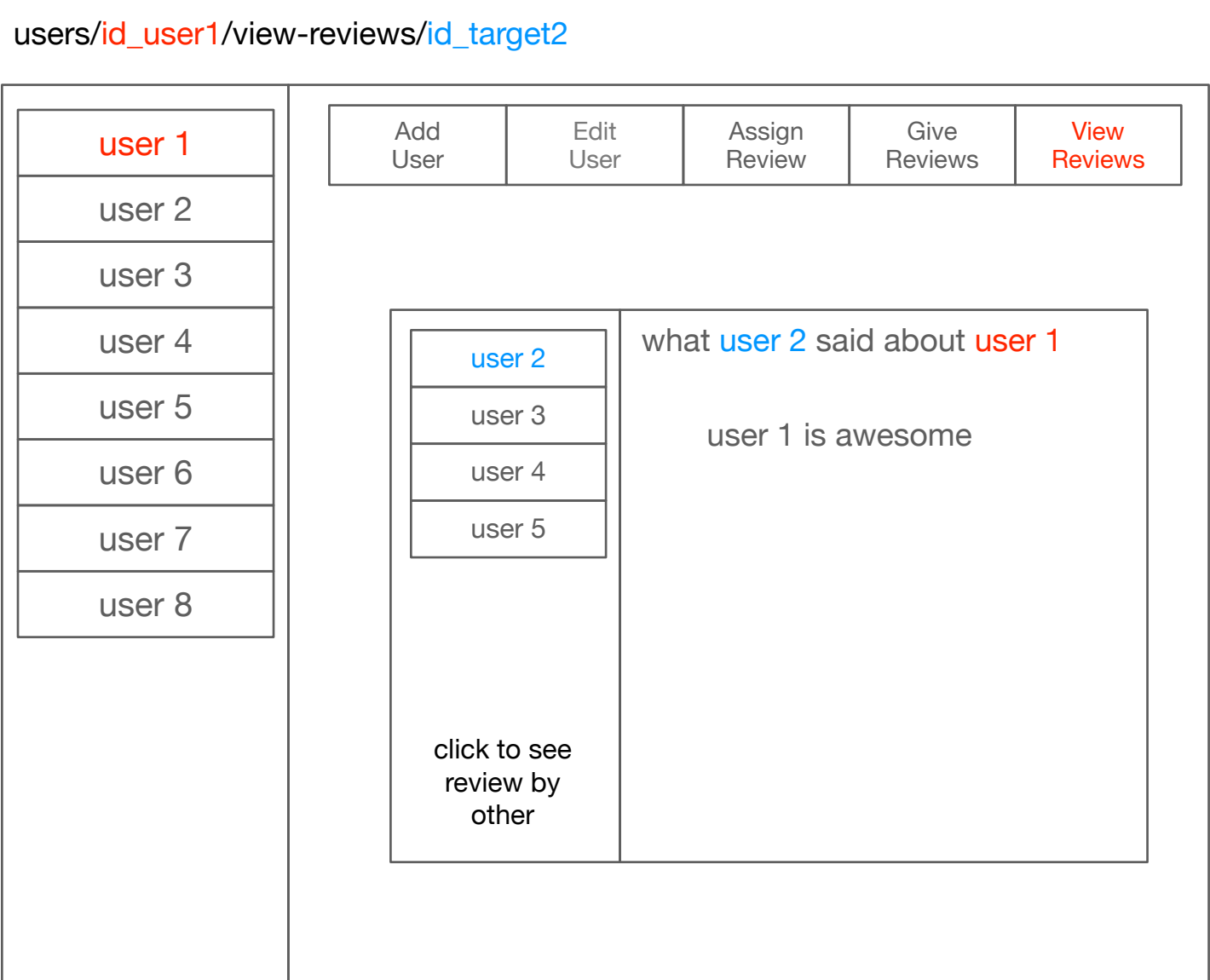
reviewee: id_target2,

reviewer: id_user1,

on POST server push id_review7 into id_target2.reviewers

user_2 = {

_id: id_target2,

userName: "markZuckerberg",

<ReviewView/>

Subcomponent

onIntialized:

- GET api/user/1

- GET api/reviews/batch-get([user_1.reviewers])

onUserSelection:

show review_2.content