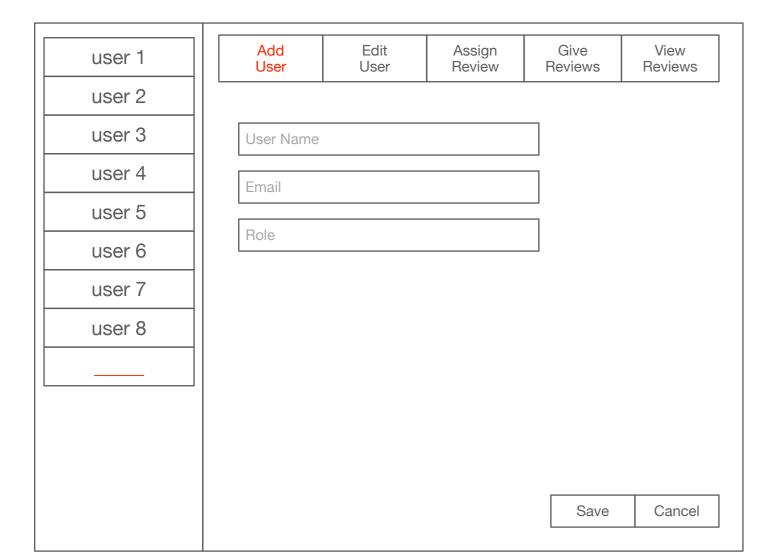


onInitialized: GET api/usersCatelog

Subcomponent

onInitialized: GET api/id_user1

users/new



onInitialized: GET api/usersCatelog

Subcomponent

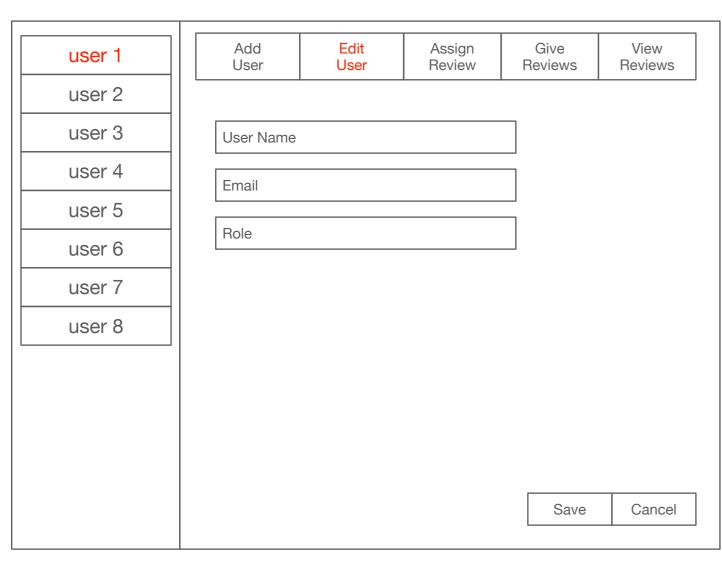
onSaveBegin: POST api/users

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

onSaveSaveSuccess:

- insert {user_newld} into users resources
- navigate to view new user
- GET api/usersCatelog

users/1/edit



onInitialized: GET api/usersCagtelog

Subcomponent

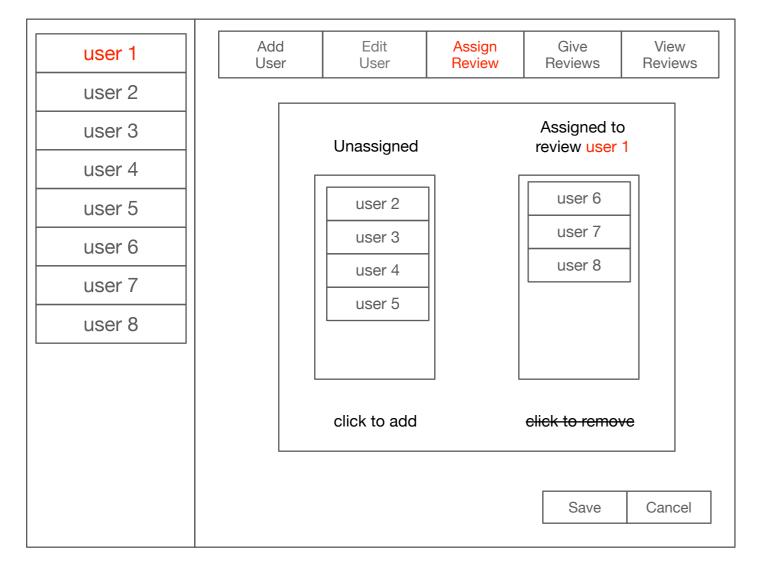
onInitialized: GET api/users/id_user1

onSaveBegin: PUT api/users/id_user1

onSaveSuccess:

- update {user_1} on users resources - navigate to view user_1

users/1/assign



onInitialized:

GET api/usersCatelog

Subcomponent

onInitialized: - GET api/assignment/id1_assignment

onUserSelection:

add/remove id_user to draft_assignment.reviews

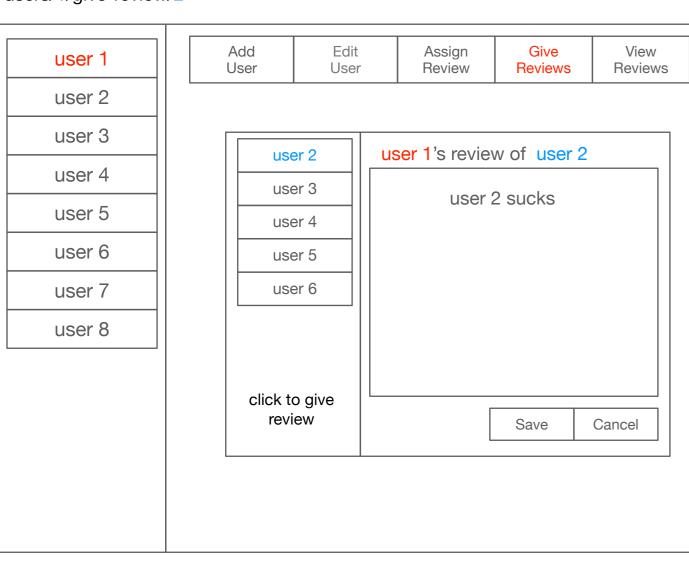
onSaveReviewBegin: - Patch api/assignment/id1_assignment

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

users/1/give-review/2



onInitialized: GET api/userCatelog

Subcomponent

onInitialized:

- GET api/assignment/id1_assignment - collection all missing user in local resource and do GET api/users/batch-get([id_user2, id_user3])

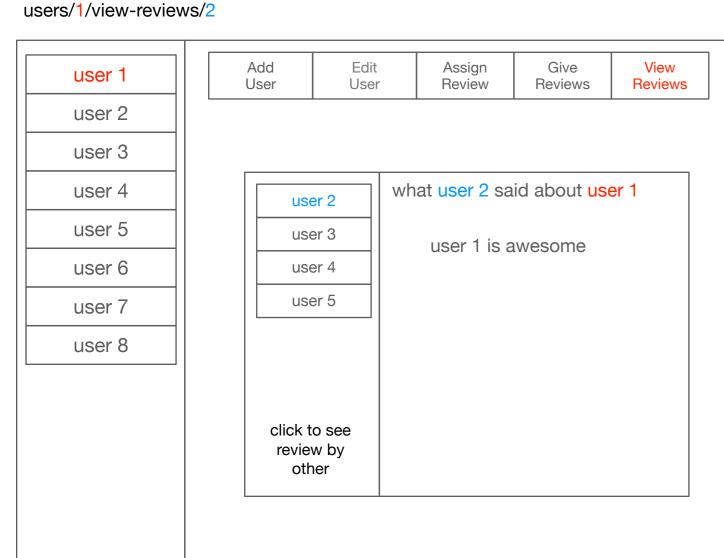
onUserSelection: Get api/review/id_review1

onSaveReviewBegin:

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

onSaveSuccess: - no special task need to exec

> review_draft = { _id: id_review7, reviewee: id_user2, reviewer: id_user1, on POST server push id_review7 into content: 'user2 sucks' user_2.reviewers $user_2 = {$ _id: id_user2, userName: "markZuckerberg", reviewers: [id_review1, id_review2, ... id_review7], reviewersAssignment: id1_assignment



onInitialized: GET api/userCatelog

Subcomponent

onInitialized:

- GET api/user/1 - GET api/reviews/batch-get([user_1.reviewers])

onUserSelection:

show review_2.content