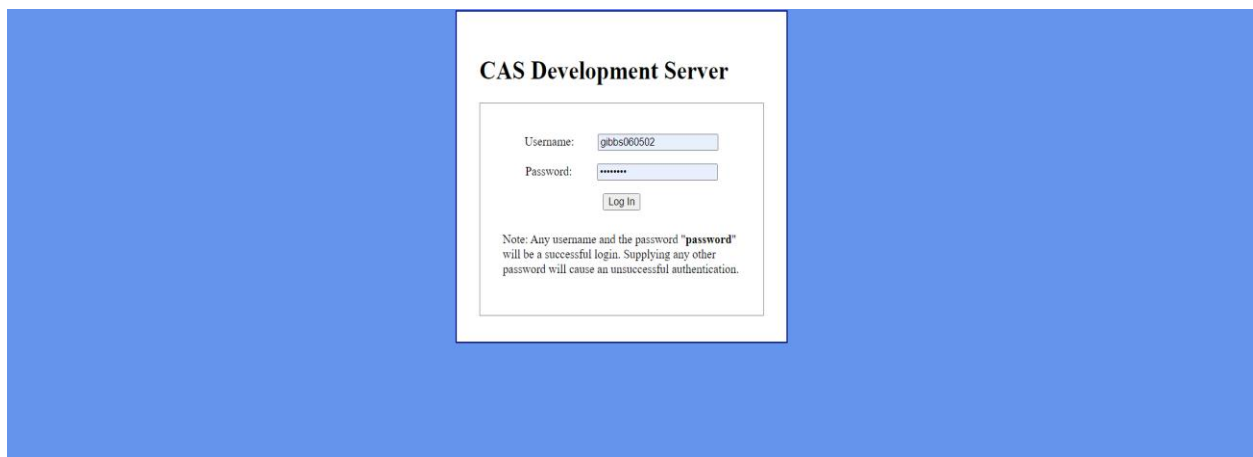


## API TEST

The expected API request return:

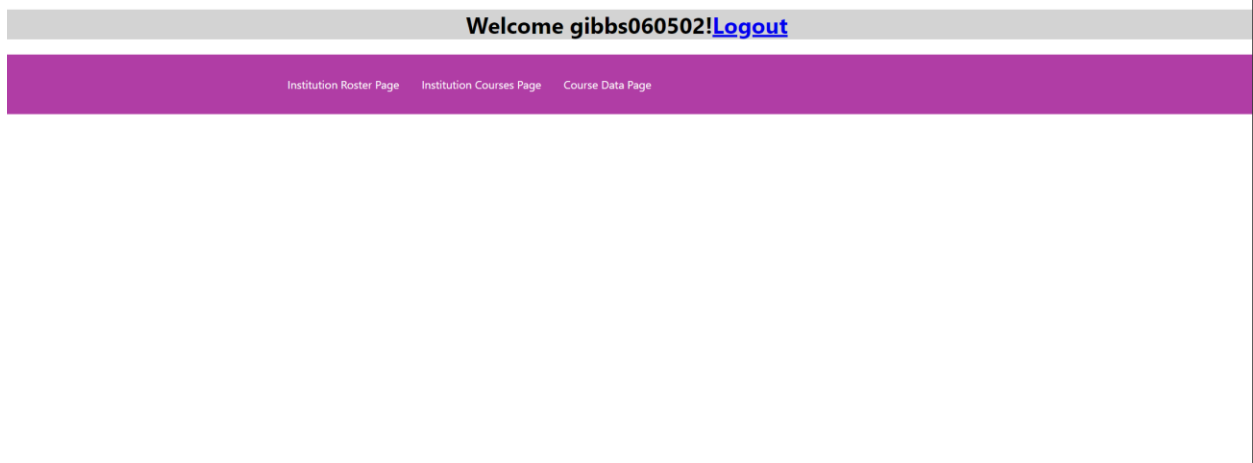
```
{
  "_id": "MNZtvxJYmY5GQ4w6z",
  "firstName": "Nicholas",
  "lastName": "Sternecker",
  "email": "nsternecker@ksu.edu"
},
{
  "_id": "QrALMiztKSx3YzDTv",
  "firstName": "Student",
  "lastName": "Gibson",
  "email": "gibson.williams2020@gmail.com"
},
{
  "_id": "XH7rtnv44jtkRcBEr",
  "firstName": "Test",
  "lastName": "Student",
  "email": "student+LQRXWcdc4PpjY72bf@perusall.com"
},
{
  "_id": "ba8FE5bnAZtB6x254",
  "firstName": "Nate",
  "lastName": "Lillich",
  "email": "nlillich@ksu.edu"
},
{
  "_id": "jdGGwy2z58RCimQCh",
  "firstName": "Nathan",
  "lastName": "Herscovici",
  "email": "nherscovici@ksu.edu"
},
{
  "_id": "yj6ngEo5ybTevTDag",
  "firstName": "Gibson",
  "lastName": "Williams",
  "email": "gibbs060502@ksu.edu"
}
```

When you launch the application it will bring you to the sign in screen. This can be passed by giving anything in the username box and “password” as the password.



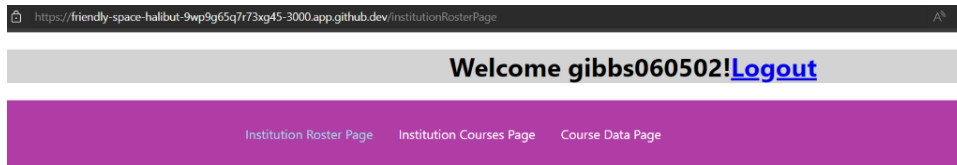
The image shows a login interface for the 'CAS Development Server'. It features a white rectangular box centered on a blue background. Inside the box, there are two input fields: 'Username' with the value 'gibbs060502' and 'Password' with masked characters '\*\*\*\*\*'. Below these fields is a 'Log In' button. A note at the bottom of the box states: 'Note: Any username and the password "password" will be a successful login. Supplying any other password will cause an unsuccessful authentication.'

Once you log in you will be brought to the welcome page.



Shows the communication between server and CAS system:

```
coding-textbook-server | Printing req.session.username: undefined
coding-textbook-server | Printing req.session.username: undefined
coding-textbook-server | Inside /login
coding-textbook-server | Inside /login
coding-textbook-server | Inside /ticket
coding-textbook-server | Successfully logged in: gibbs060502
coding-textbook-server | Printing req.session.username: gibbs060502
coding-textbook-server | Username: gibbs060502
coding-textbook-server | Printing req.session.username: gibbs060502
coding-textbook-server | Username: gibbs060502
```



### Perusall API User Return:

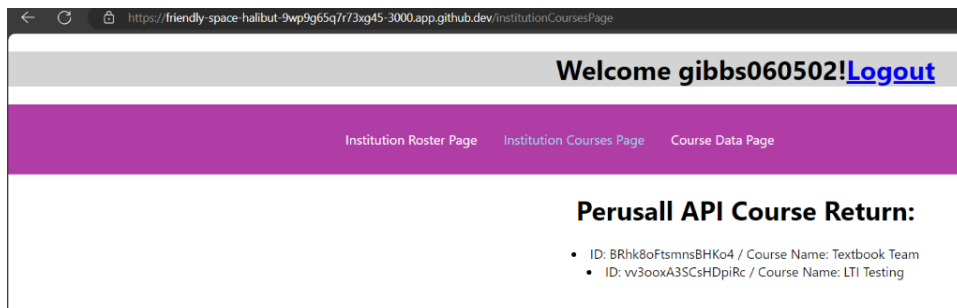
- ID: MNZtwJYmY5GQ4w6z / First Name: Nicholas / Last Name: Sternecker / Email: nsternecker@ksu.edu
- ID: QrALMiztKSx3YzDTv / First Name: Student / Last Name: Gibson / Email: gibson.williams2020@gmail.com
- ID: XH7rtv44jtkRcBEr / First Name: Test / Last Name: Student / Email: student+LQRXWcd4PpY72bf@perusall.com
  - ID: ba8FE5bnAZtB6x254 / First Name: Nate / Last Name: Lillich / Email: nillich@ksu.edu
- ID: jdGgwy2z58RCrmQCh / First Name: Nathan / Last Name: Herscovici / Email: nherscovici@ksu.edu
- ID: yj6ngEo5ybTevTDag / First Name: Gibson / Last Name: Williams / Email: gibbs060502@ksu.edu

## Institution\_roster:

Parameters: None

Institution Roster Page click should send request to server. Server should reach out to perusall API and request information. Information should be passed back to server. Server should pass information back to client side. Client side should then display information. Should return a list of objects in the form of `{[_id: string, FirstName: string, lastName: string, email: string]}`.

Content from Perusall API call for users displayed within the client test page:



### Perusall API Course Return:

- ID: BRhk8oFtsmns8HKo4 / Course Name: Textbook Team
- ID: w3ooxA3SCsHDpiRc / Course Name: LTI Testing

## Institution\_courses

Parameters: None

Button click should send request to server. Server should reach out to perusall API and request information. Information should be passed back to server. Server should pass information back to client side. Client side should

then display information. Should return a list of objects in the form of [{ \_id: string, name: string }]

https://friendly-space-halibut-9wp9g65q7r73xg45-3000.app.github.dev/courseDataPage

Welcome gibbs060502! [Logout](#)

[Institution Roster Page](#) [Institution Courses Page](#) [Course Data Page](#)

Enter the course id:

**Perusall API Course Return:**

**Course Name: Textbook Team**

- ID: BRhk8oFtsmnsBHKo4
- Instructor Id: yj6ngEo5ybTevTDag
- Instructor Id: ba8FE5bnAZtB6x254
- Student Id: MNZtvxJYmY5GQ4w6z
- Student Id: jdGGwy2z58RCrmQCh
- Student Id: XH7rtnv44jtkRcBEr

Course\_data:

Parameters: id of course to find information

on Course Data Page will open the page. The user then has to enter the ID of the course to be searched. Server should reach out to perusall API and request information. Information should be passed back to server. Server should pass information back to client side. Client side should then display information. Should return an object in the form of { \_id: string, instructorIds: [string], name: string, lowestPossibleScore: int, highestPossibleScore: int, inactiveStudentIds: [string], studentIds: [string] }

Below is a screenshot of postman returns that shows the data above is data from Perusall

```
1  [
2    {
3      "_id": "MNZtvxJYmY5GQ4w6z",
4      "firstName": "Nicholas",
5      "lastName": "Sternecker",
6      "email": "nsternecker@ksu.edu"
7    },
8    {
9      "_id": "QrALMiztKSx3YzDTv",
10     "firstName": "Student",
11     "lastName": "Gibson",
12     "email": "gibson.williams2020@gmail.com"
13   },
14   {
15     "_id": "XH7rtnv44jtkRcBEr",
16     "firstName": "Test",
17     "lastName": "Student",
18     "email": "student+LQRXWcdc4PpjY72bf@perusall.com"
19   },
20   {
21     "_id": "ba8FE5bnAZtB6x254",
22     "firstName": "Nate",
23     "lastName": "Lillich",
24     "email": "nlillich@ksu.edu"
25   },
26   {
27     "_id": "jdGGwy2z58RCrmQCh",
28     "firstName": "Nathan",
29     "lastName": "Herscovici",
30     "email": "nherscovici@ksu.edu"
31   },
32   {
33     "_id": "yj6ngEo5ybTevTDag",
34     "firstName": "Gibson",
35     "lastName": "Williams",
36     "email": "gibbs060502@ksu.edu"
37   }
38 ]
```

```

1  {
2    "_id": "BRhk8oFtsmnsBHKo4",
3    "name": "Textbook Team",
4    "instructorIds": [
5      "yj6ngEo5ybTevTDag",
6      "ba8FE5bnAZtB6x254"
7    ],
8    "studentIds": [
9      "MNZtvxJYmY5GQ4w6z",
10     "jdGGwy2z58RCrmQCh",
11     "XH7rtnv44jtkRcBEr"
12   ],
13   "inactiveStudentIds": [],
14   "lti": {
15     "version": "1.1",
16     "institutionId": "9RFriSoEyL8PuvJh6",
17     "instanceId": "7db438871375c02373713c12c73869ff2f470b68.k-state.instructure.com",
18     "contextId": "97b4be803cd06efa24f5f02dbc008e8917103cc6",
19     "lmsName": "canvas"
20   },
21   "lowestPossibleScore": 0,
22   "highestPossibleScore": 3,
23   "ltiInstanceId": "7db438871375c02373713c12c73869ff2f470b68.k-state.instructure.com",
24   "ltiContextId": "97b4be803cd06efa24f5f02dbc008e8917103cc6"
25 }

```

```

1  [
2    {
3      "_id": "BRhk8oFtsmnsBHKo4",
4      "name": "Textbook Team"
5    },
6    {
7      "_id": "vv3ooxA3SCsHDpiRc",
8      "name": "LTI Testing"
9    }
10 ]

```