

Service Layers for ShiftHub Application

By William Greenlaw

Service Endpoints

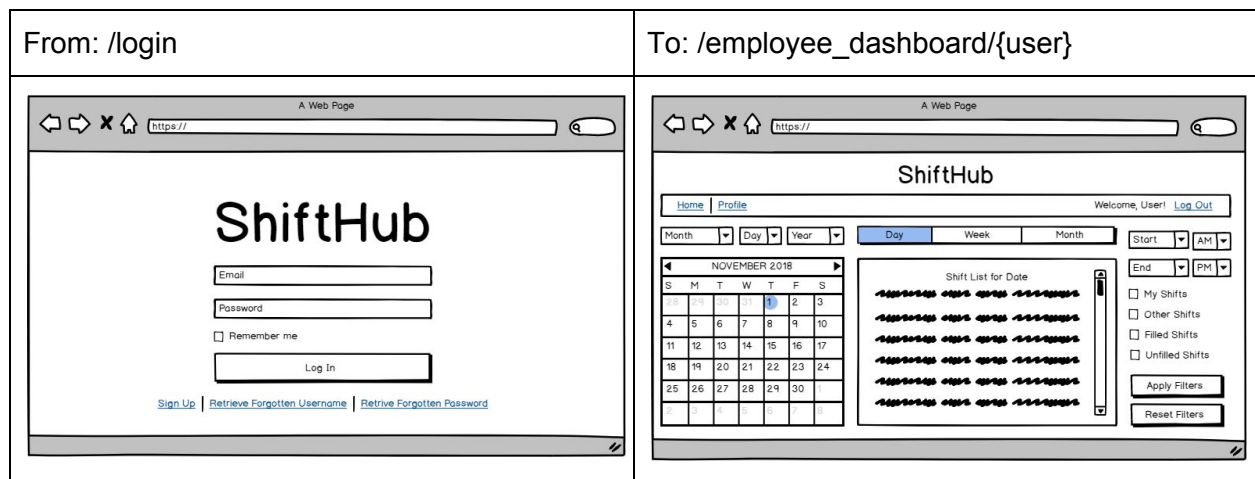
Endpoint:

/employee_dashboard/{user}

Endpoint Description:

The home dashboard that becomes visible when an employee logs in. The dashboard contains information such their schedule for the day and a list of shifts that are available for day. Users will have a different dashboard view based on their user privileges (i.e., employees will have limited access, whereas managers will have full access).

Endpoint Diagram:



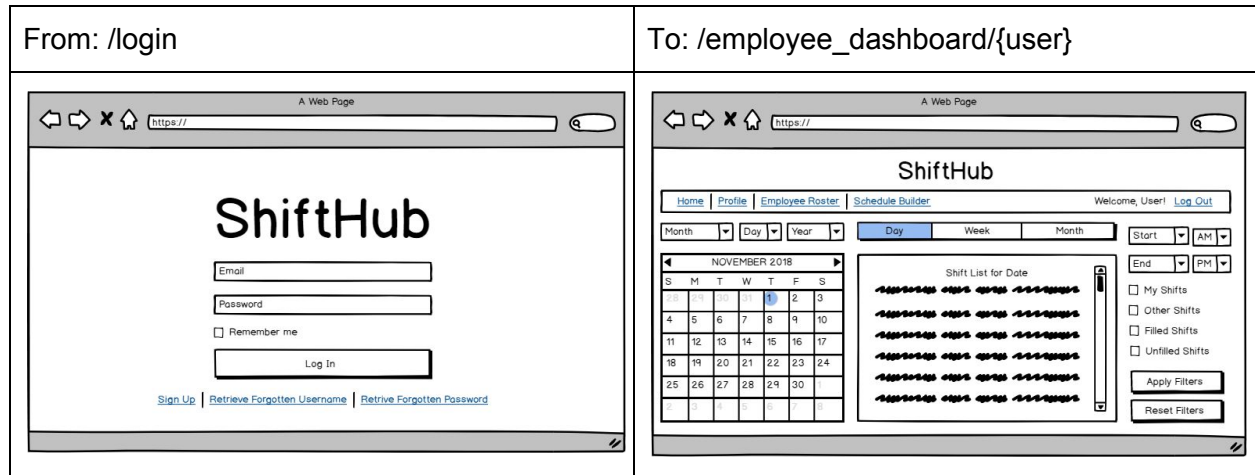
Endpoint:

/manager_dashboard/{user}

Endpoint Description:

The home dashboard that becomes visible when a manager logs in. The manager dashboard looks similar to the employee dashboard, but has additional links and features available.

Endpoint Diagram:



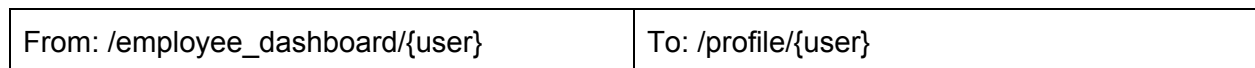
Endpoint:

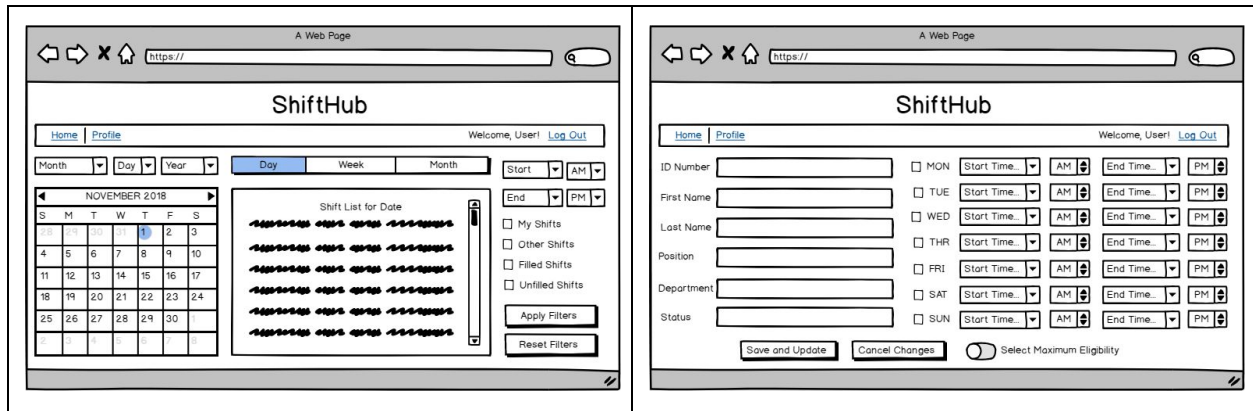
/employee_profile/{user}

Endpoint Description:

The profile page that becomes visible upon clicking on the “Profile” link in the navigation bar. This page appears the same for all users, regardless of account type.

Endpoint Diagram:





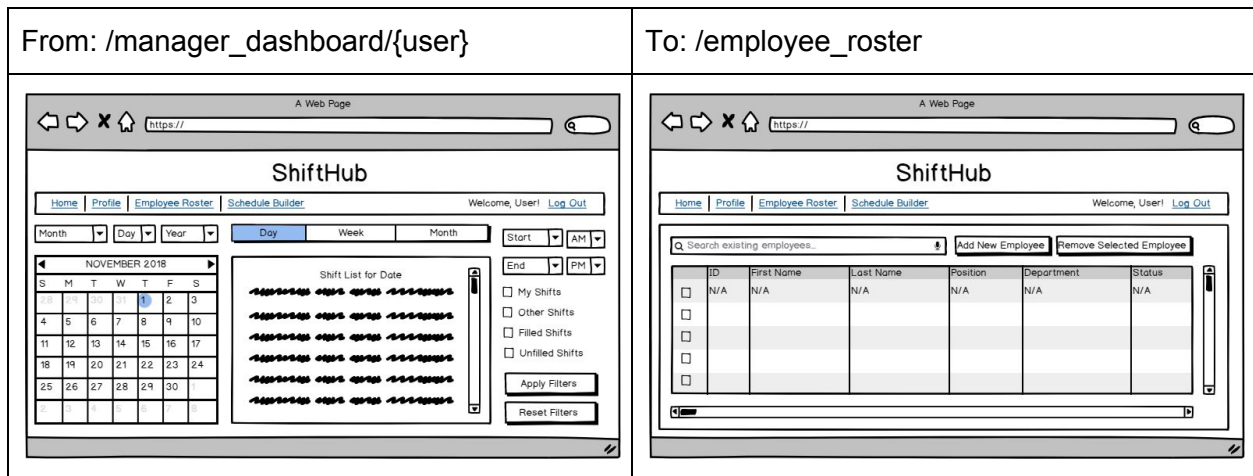
Endpoint:

/employee_roster

Endpoint Description:

The employee roster that is accessed by clicking on the “Employee Roster” link in the navigation bar. This option will only be available to users who are managers.

Endpoint Diagram:



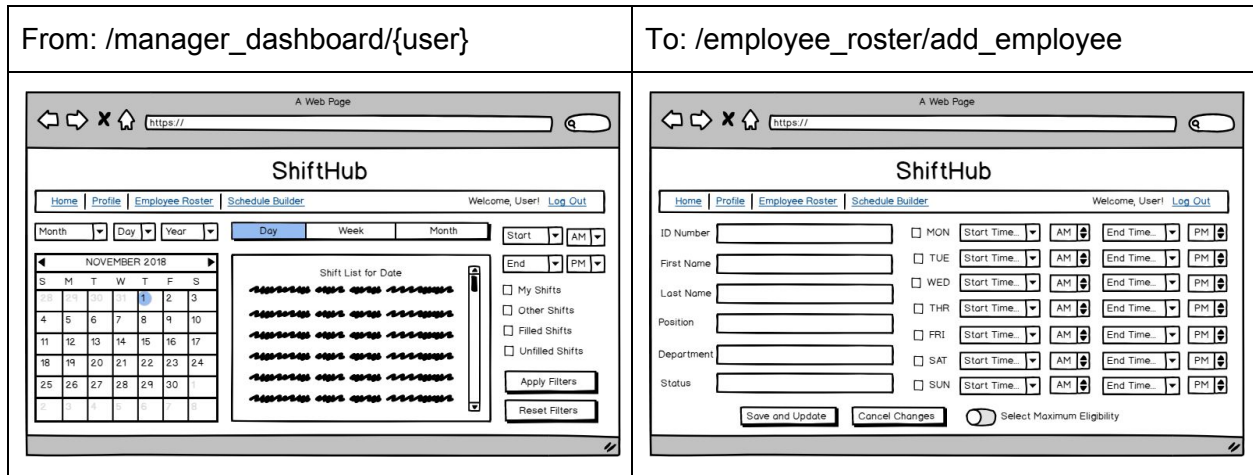
Endpoint:

/employee_roster/add_employee

Endpoint Description:

The new employee form that is accessed by clicking on the “Add New Employee” link in the navigation bar.

Endpoint Diagram:



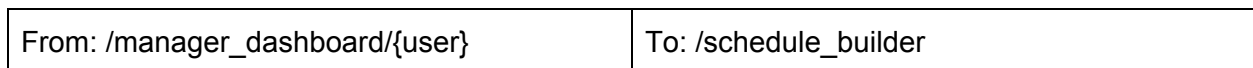
Endpoint:

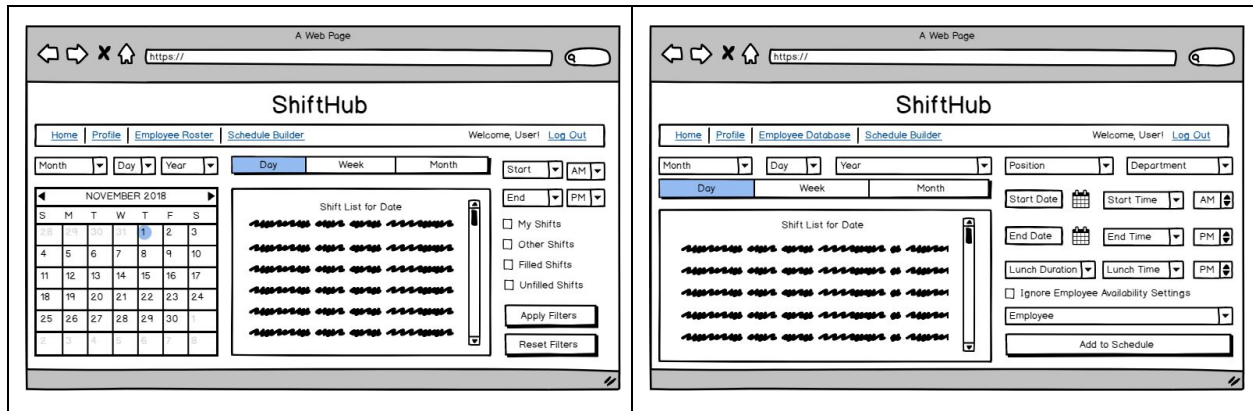
/schedule_builder

Endpoint Description:

The new shift form that is accessed by clicking on the “Schedule Builder” link in the navigation bar.

Endpoint Diagram:





Sample Requests/Responses

Users

POST /user Creates user

Parameters:

This is completed during registration/account creation.

Body:

```
{
  "employeeID": 12345,
  "emailAddress": "johndoe@gmail.com",
  "passwordHash": "Q'rXyGwWn1UGg80T",
  "userType": "Employee",
  "firstName": "John",
  "lastName": "Doe",
  "position": "Help Desk Associate",
  "department": "Customer Service",
  "status": "Full Time",
  "availability": [
    {"dayOfWeek": "Monday", "startTime": "9:00 AM", "endTime": "5:00 PM"},
    {"dayOfWeek": "Tuesday", "startTime": "9:00 AM", "endTime": "5:00 PM"},
    {"dayOfWeek": "Wednesday", "startTime": "9:00 AM", "endTime": "5:00 PM"},
    {"dayOfWeek": "Thursday", "startTime": "9:00 AM", "endTime": "5:00 PM"},
    {"dayOfWeek": "Friday", "startTime": "9:00 AM", "endTime": "5:00 PM"},
    {"dayOfWeek": "Saturday", "startTime": "9:00 AM", "endTime": "5:00 PM"},
    {"dayOfWeek": "Sunday", "startTime": "9:00 AM", "endTime": "5:00 PM"},
  ]
}
```

```
}
```

Responses:

successful operation

GET /user/login Logs user into the system**Parameters:**

Username (the email address for login)

Password (the password for login)

Responses:

Code 200 (successful operation)

Code 400 (invalid username/password supplied)

GET /user/logout Logs user out of current session**Parameters:**

No parameters

Responses:

Default (successful operation)

PUT /user/{username} Updates user**Parameters:**

Information that needs to be updated

Body:

```
{
  "employeeID": 12345,
  "emailAddress": "johndoe@gmail.com",
  "passwordHash": "Q'rXyglWn1UGg80T",
  "userType": "Employee",
  "firstName": "John",
  "lastName": "Doe",
  "position": "Help Desk Associate",
  "department": "Customer Service",
  "status": "Full Time",
```

```
"availability": [  
  {"dayOfWeek": "Monday", "startTime": "9:00 AM", "endTime": "5:00 PM"},  
  {"dayOfWeek": "Tuesday", "startTime": "9:00 AM", "endTime": "5:00 PM"},  
  {"dayOfWeek": "Wednesday", "startTime": "9:00 AM", "endTime": "5:00 PM"},  
  {"dayOfWeek": "Thursday", "startTime": "9:00 AM", "endTime": "5:00 PM"},  
  {"dayOfWeek": "Friday", "startTime": "9:00 AM", "endTime": "5:00 PM"},  
  {"dayOfWeek": "Saturday", "startTime": "9:00 AM", "endTime": "5:00 PM"},  
  {"dayOfWeek": "Sunday", "startTime": "9:00 AM", "endTime": "5:00 PM"},  
]  
}
```

Responses:

Code 400 (invalid user supplied)

Code 404 (user not found)

DELETE /user/{username} Deletes user

Parameters:

EmployeeID (the user that needs to be deleted)

Responses:

Code 400 (invalid user supplied)

Code 404 (user not found)

Schedules

POST /schedule Creates schedule

Parameters:

This can only be done by a manager account.

Body:

```
{  
  "scheduleID": 12345,  
  "startDate": "2019-01-01",  
  "endDate": "2019-12-31",  
  "shifts": [  

```

<pre> {"shiftID": 12345}, {"shiftID": 12346}, {"shiftID": 12347}] } </pre>
Responses:
Successful operation

PUT /schedule Updates schedule

Parameters:
scheduleID (the ID of the schedule that needs to be updated)
Body:
<pre> { "scheduleID": 12345, "startDate": "2019-01-01", "endDate": "2019-12-31", "shifts": [{"shiftID": 12345}, {"shiftID": 12346}, {"shiftID": 12347}] } </pre>
Responses:
Code 400 (invalid schedule supplied)
Code 404 (schedule not found)

DELETE /schedule Deletes schedule

Parameters:
scheduleID (the ID of the schedule that needs to be deleted)
Responses:
Code 400 (invalid user supplied)
Code 404 (user not found)

Shifts

POST /shift Creates shift

Body:
<pre>{ "shiftID": 12345, "employeeID": 12345, "startDate": "2019-01-02", "startTime": "9:00 AM", "endDate": "2019-01-02", "endTime": "5:00 PM", "lunchBreak": true, "lunchBreakDuration": 0.75, "notes": null }</pre>
Responses:
Successful operation

PUT /shift Updates Shift

Parameters:
shiftID (the ID of the shift that needs to be deleted)
Body:
<pre>{ "shiftID": 12345, "employeeID": 12345, "startDate": "2019-01-02", "startTime": "9:00 AM", "endDate": "2019-01-02", "endTime": "5:00 PM", "lunchBreak": true, "lunchBreakDuration": 0.75, "notes": null }</pre>
Responses:
Code 400 (invalid shift supplied)

Code 404 (shift not found)

DELETE /shift Deletes shift

Parameters:
shiftID (the ID of the shift that needs to be deleted)
Responses:
Code 400 (invalid shift supplied)
Code 404 (shift not found)