# Selenium Webdriver Primer

Spensa Technologies, Inc
Sept. 22, 2016

Derrick Kearney
github.com/codedsk
telldsk@gmail.com

# Testing vs Checking

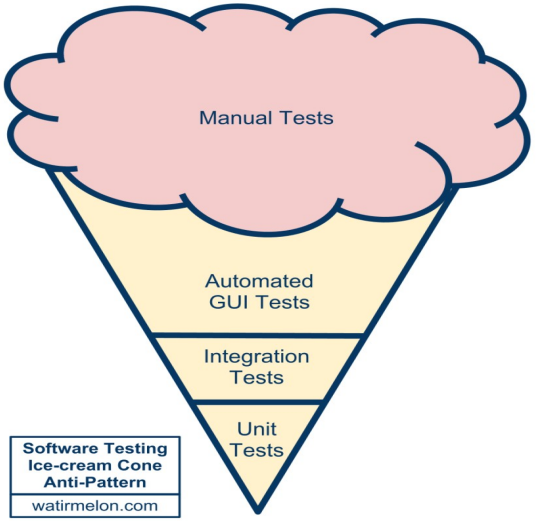
## **Testing is...**

* Exploring & Learning
* Experimenting
* Judgement & Values
* Answers the question
  "*Is there a problem?*"

## **Checking is...**

* Confirming
* Pass / Fail
* "It Works" vs "It Operates"

# Typical Testing Pyramid

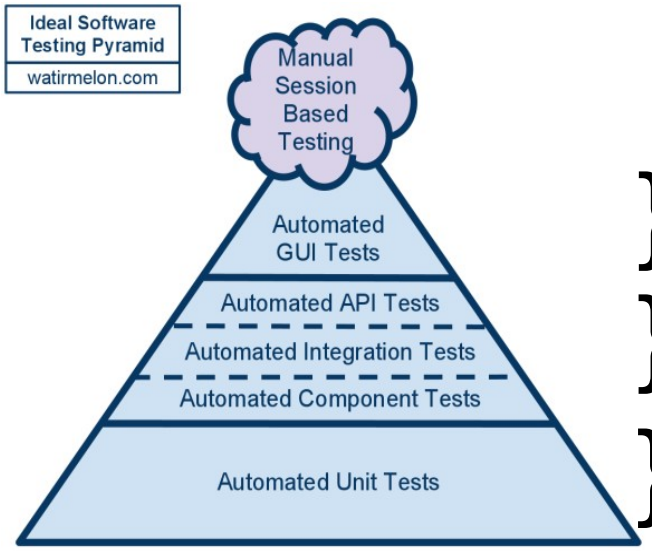



Too many GUI & Manual Tests

Very few Unit & Integration Tests

Ideal Automated Testing Pyramid & Ice-cream Cone Antipattern - http://watirmelon.com/2012/01/31/introducing-the-software-testing-ice-cream-cone/

# Ideal Testing Pyramid



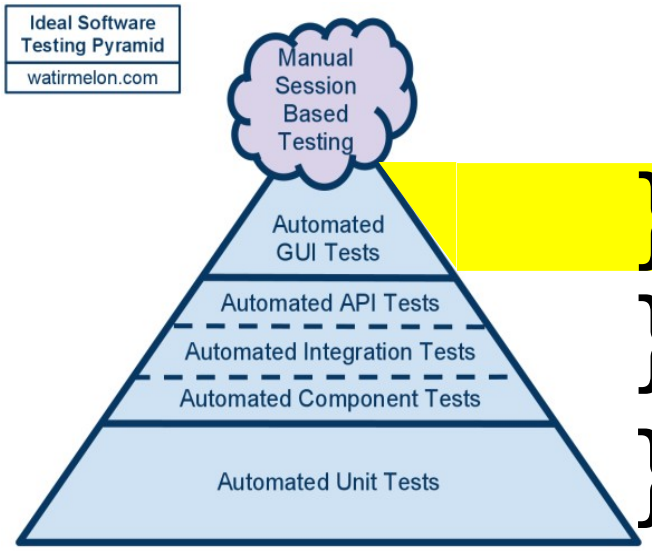

} User Interface Tests

} Service Layer Tests

} Unit Tests

Ideal Automated Testing Pyramid & Ice-cream Cone Antipattern - http://watirmelon.com/2012/01/31/introducing-the-software-testing-ice-cream-cone/

# Ideal Testing Pyramid




} User Interface Tests

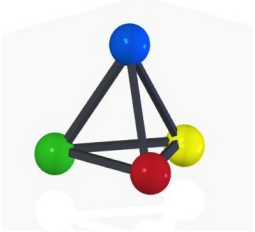
} Service Layer Tests

} Unit Tests

Ideal Automated Testing Pyramid & Ice-cream Cone Antipattern - http://watirmelon.com/2012/01/31/introducing-the-software-testing-ice-cream-cone/

# Automated Testing Tools

## Web Crawlers



CrawlJax

IBM AppScan

### Pros:

- Little Developer Coordination
- Semi directed exploration
- Web App is a black box

### Cons:

- Time consuming to run
- No authority to judge results

# Automated Testing Tools

## Test Recorders

Selenium

QuickTest Pro

Sahi Pro

### Pros:

- Very customizable
- Produces specific targeted tests
- Programmable in a script

### Cons:

- Time consuming to build tests
- Allows for brittle test designs

# Automated Testing Tools

## Test Recorders

Selenium

- Developed By Jason Huggins and Simon Stewart at ThoughtWorks

QuickTest Pro

- Browser vendor support through W3C spec

Sahi Pro

- Automation written in “your favorite language”...
- Worldwide support Community

# Automated Testing Tools

## Test Recorders

Selenium

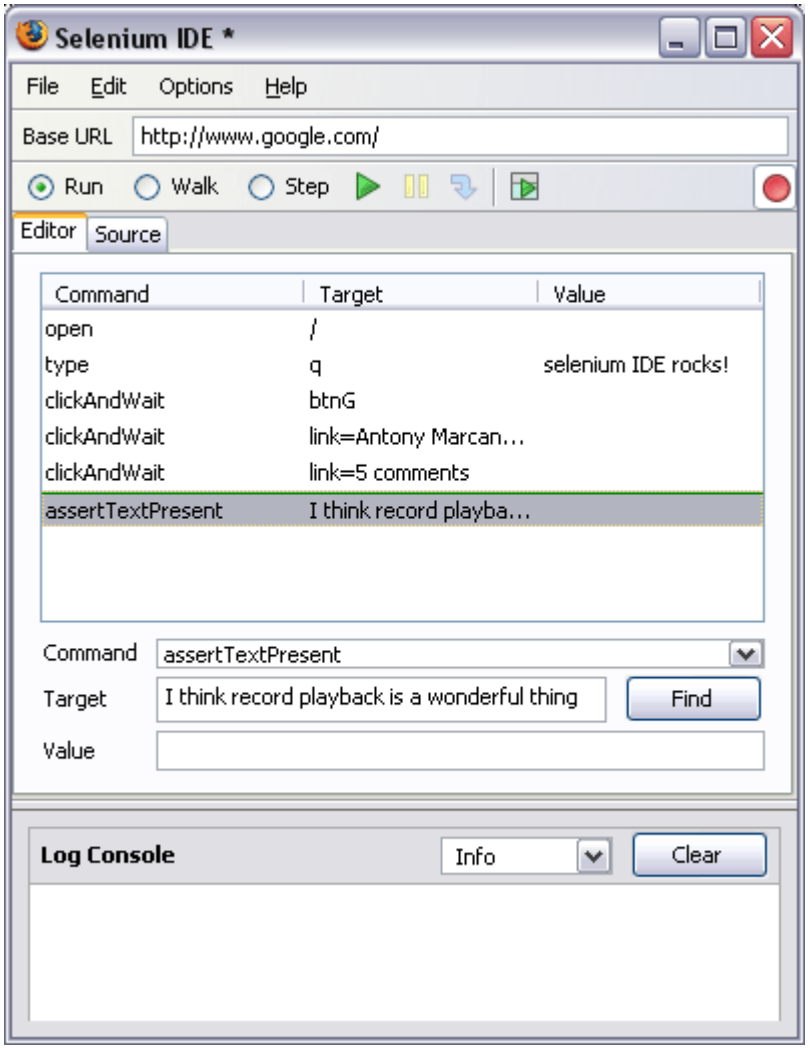
QuickTest Pro

Sahi Pro

**What can it do?**

- Open web browser
- Locate elements on page
- Type in or click elements
- Wait for triggered events
- Simulate moving the mouse

# A brief word about Selenium IDE...

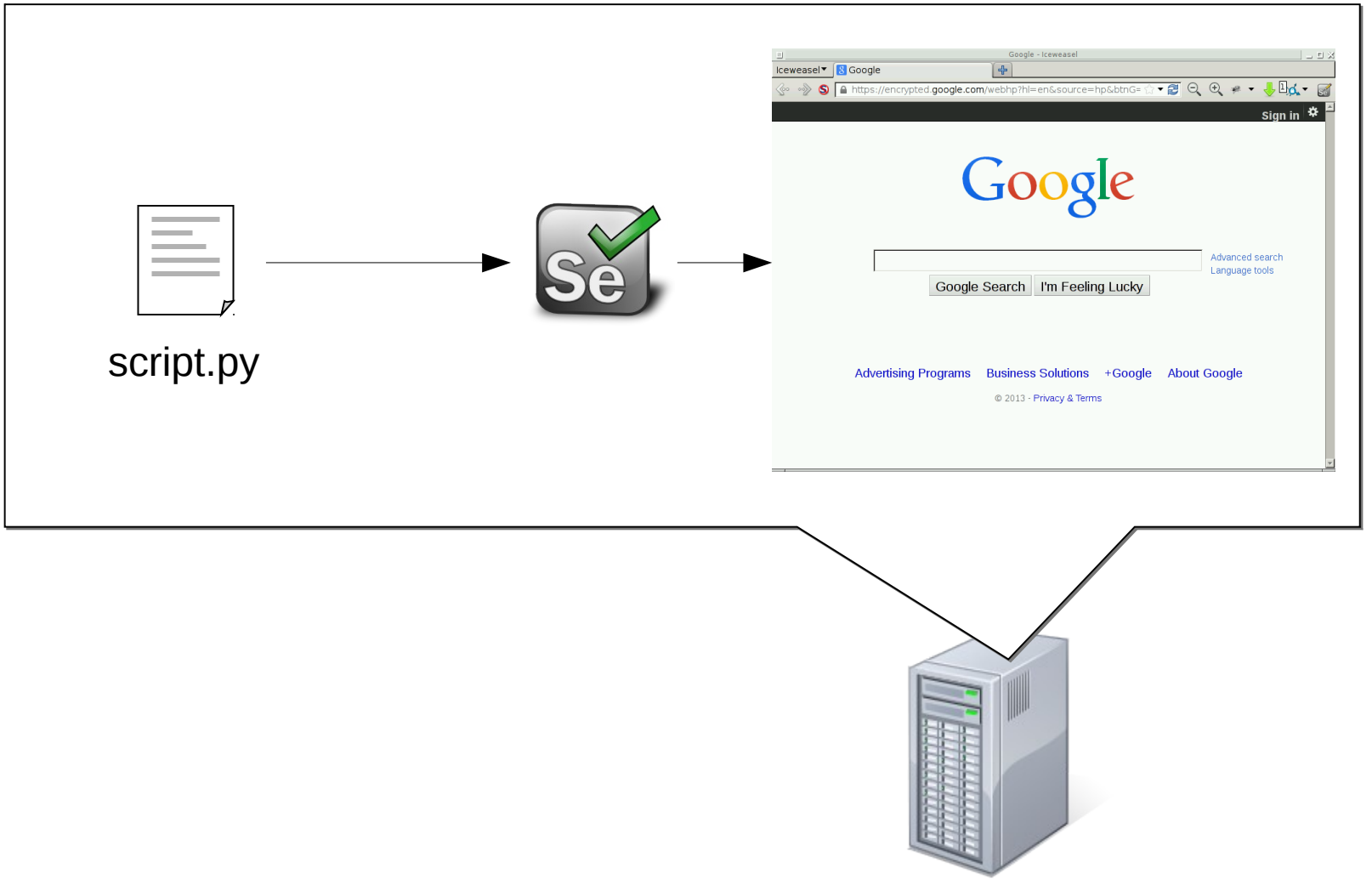



**What can it do?**

- Record actions, replay as tests
- Works in Firefox only
- Selenium 1.0 style tests
- Tests can be brittle

**Try it out, then move on**

# How Selenium Works



# Login

**Log in with your Hub account.**

Username:

Forgot your Username?

Password:

Forgot your Password?

Remember Me

Login

**No account?**

Register. It's free!

**Is this really free?**

Yes! Use of HUBzero resources and tools is *free* for registered users. There are no hidden costs or fees.

**Why is registration required for parts of HUBzero?**

Our sponsors ask us who uses HUBzero and what they use it for. Registration helps us answer these questions. Usage statistics also focus our attention on improvements, making the HUBzero experience better for *you*.

# Login

**Log in with your Hub account.**

Username:

hctest

Forgot your Username?

Password:

Forgot your Password?

Remember Me

Login

**No account?**

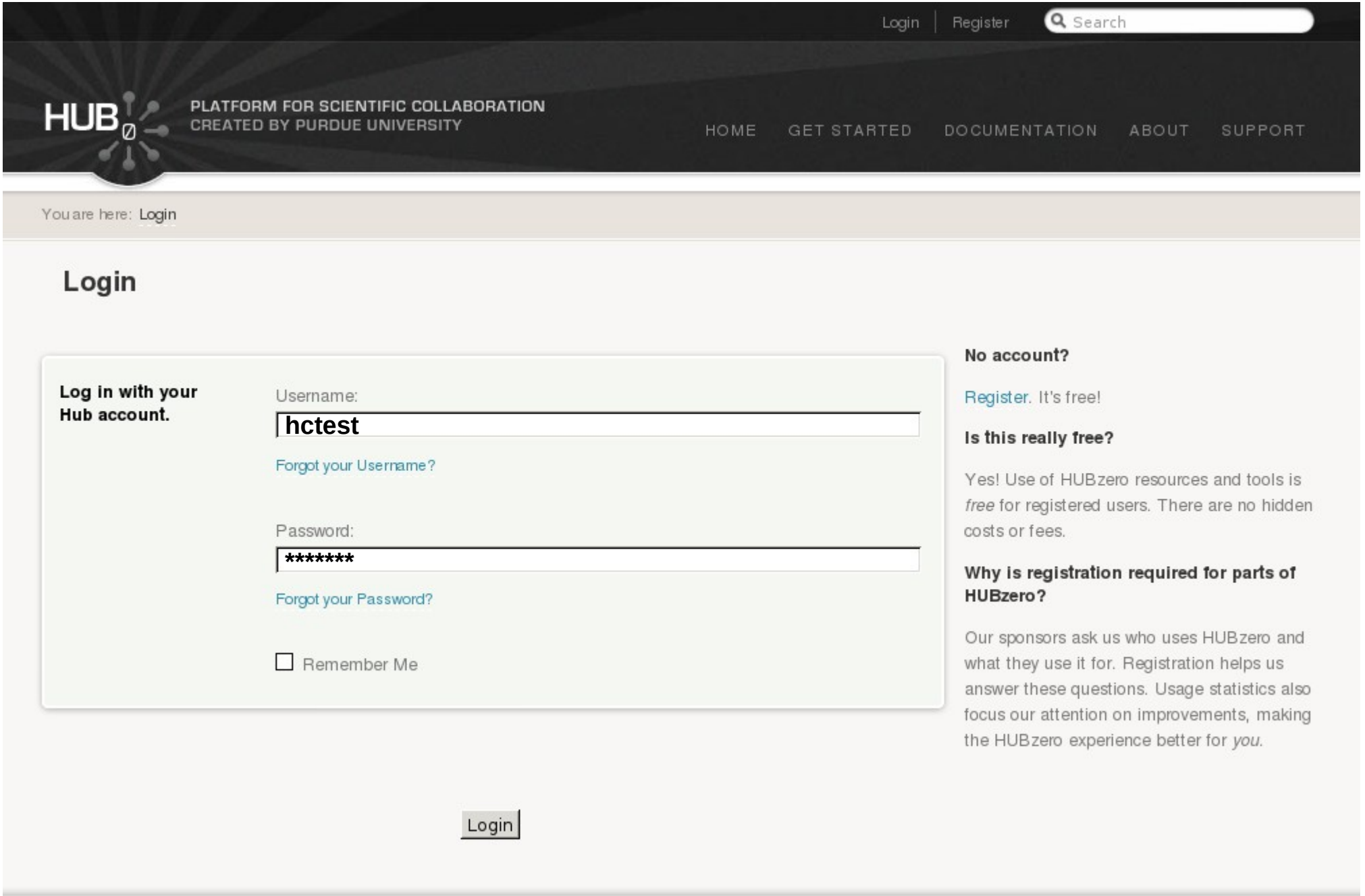
Register. It's free!

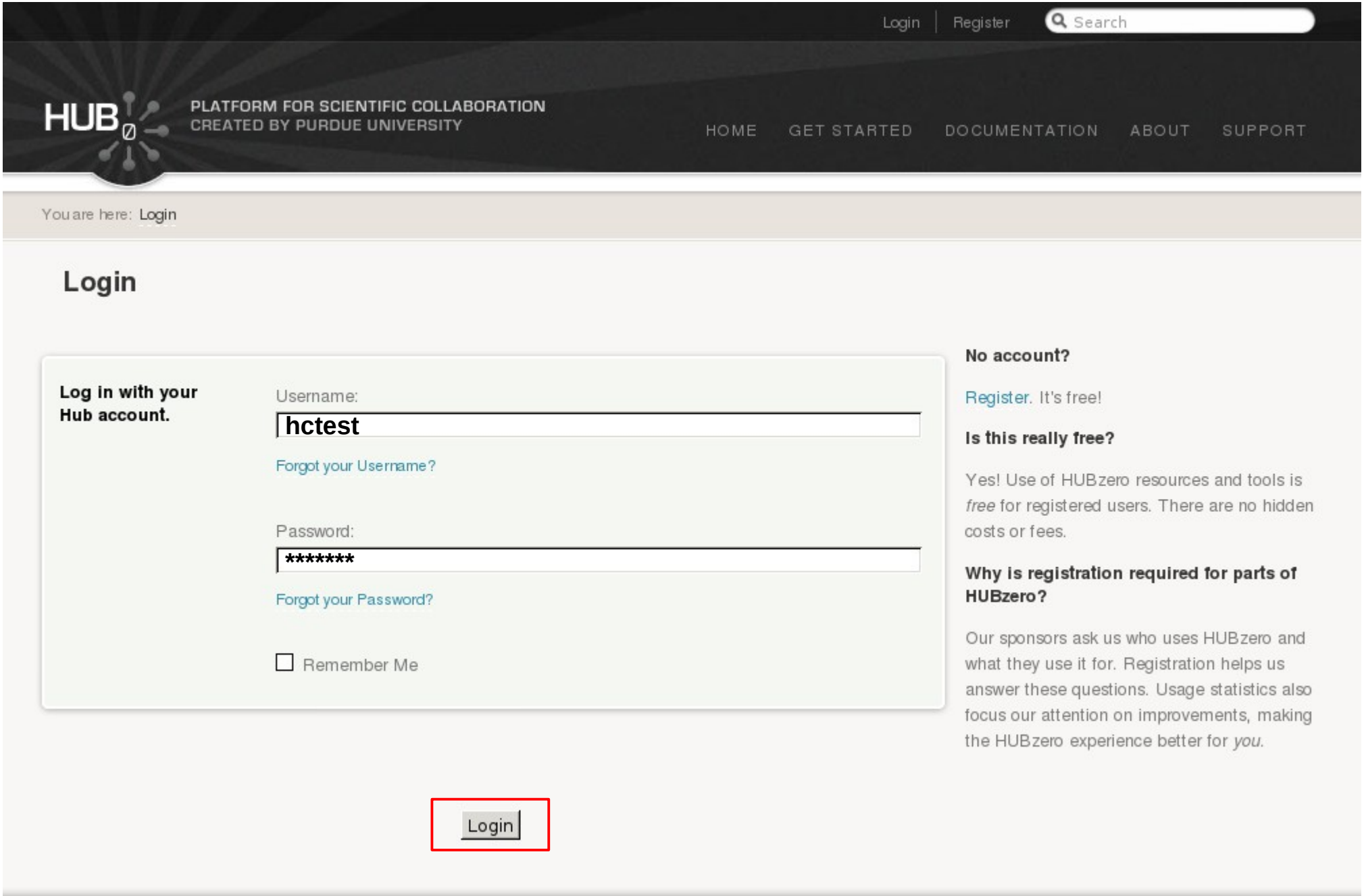
**Is this really free?**

Yes! Use of HUBzero resources and tools is *free* for registered users. There are no hidden costs or fees.

**Why is registration required for parts of HUBzero?**

Our sponsors ask us who uses HUBzero and what they use it for. Registration helps us answer these questions. Usage statistics also focus our attention on improvements, making the HUBzero experience better for *you*.


Login
Register
Search
HUB
PLATFORM FOR SCIENTIFIC COLLABORATION
CREATED BY PURDUE UNIVERSITY
HOME
GET STARTED
DOCUMENTATION
ABOUT
SUPPORT
You are here: Login
Login
Log in with your Hub account.
Username:
hctest
Forgot your Username?
Password:
*******
Forgot your Password?
Remember Me
No account?
Register. It's free!
Is this really free?
Yes! Use of HUBzero resources and tools is free for registered users. There are no hidden costs or fees.
Why is registration required for parts of HUBzero?
Our sponsors ask us who uses HUBzero and what they use it for. Registration helps us answer these questions. Usage statistics also focus our attention on improvements, making the HUBzero experience better for you.
Login


Login
Register
Search
HUB
PLATFORM FOR SCIENTIFIC COLLABORATION
CREATED BY PURDUE UNIVERSITY
HOME
GET STARTED
DOCUMENTATION
ABOUT
SUPPORT
You are here: Login
Login
Log in with your Hub account.
Username:
hctest
Forgot your Username?
Password:
*******
Forgot your Password?
Remember Me
No account?
Register. It's free!
Is this really free?
Yes! Use of HUBzero resources and tools is free for registered users. There are no hidden costs or fees.
Why is registration required for parts of HUBzero?
Our sponsors ask us who uses HUBzero and what they use it for. Registration helps us answer these questions. Usage statistics also focus our attention on improvements, making the HUBzero experience better for you.
Login

# Loading the Selenium Webdriver module:

```
from selenium import webdriver
```

# Start a web browser:

```
from selenium import webdriver

url = 'http://hubzero.org'

browser = webdriver.Firefox()
```

# Navigating to a web page:

```
from selenium import webdriver

url = 'http://hubzero.org'

browser = webdriver.Firefox()
browser.get(url)
```

# Closing a web browser:

```
from selenium import webdriver

url = 'http://hubzero.org'

browser = webdriver.Firefox()
browser.get(url)
...
browser.quit()
```

# Finding elements on a web page:

```
from selenium import webdriver

url = 'http://hubzero.org'

browser = webdriver.Firefox()
browser.get(url)

browser.find_element(...)
browser.find_element_by_*(...)

browser.quit()
```

# Login

**Log in with your Hub account.**

Username:

Forgot your Username?

Password:

Forgot your Password?

Remember Me

Login

**No account?**

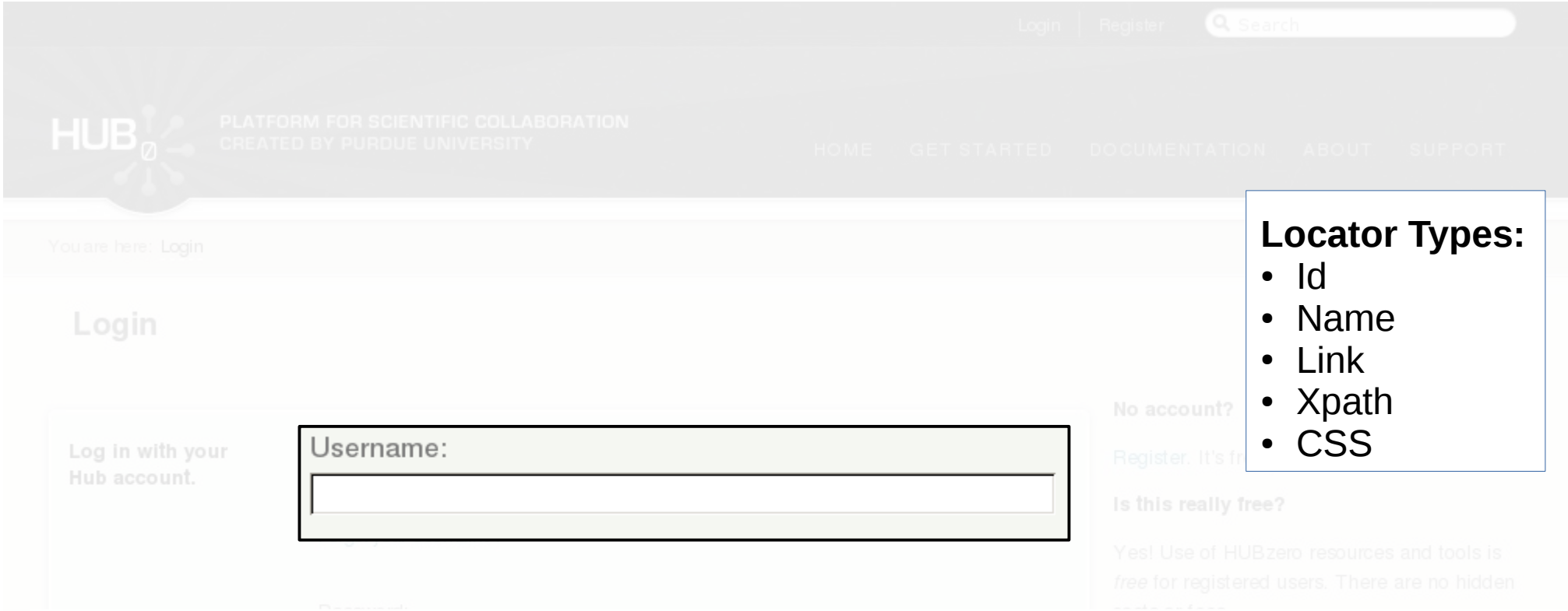
Register. It's free!

**Is this really free?**

Yes! Use of HUBzero resources and tools is *free* for registered users. There are no hidden costs or fees.

**Why is registration required for parts of HUBzero?**

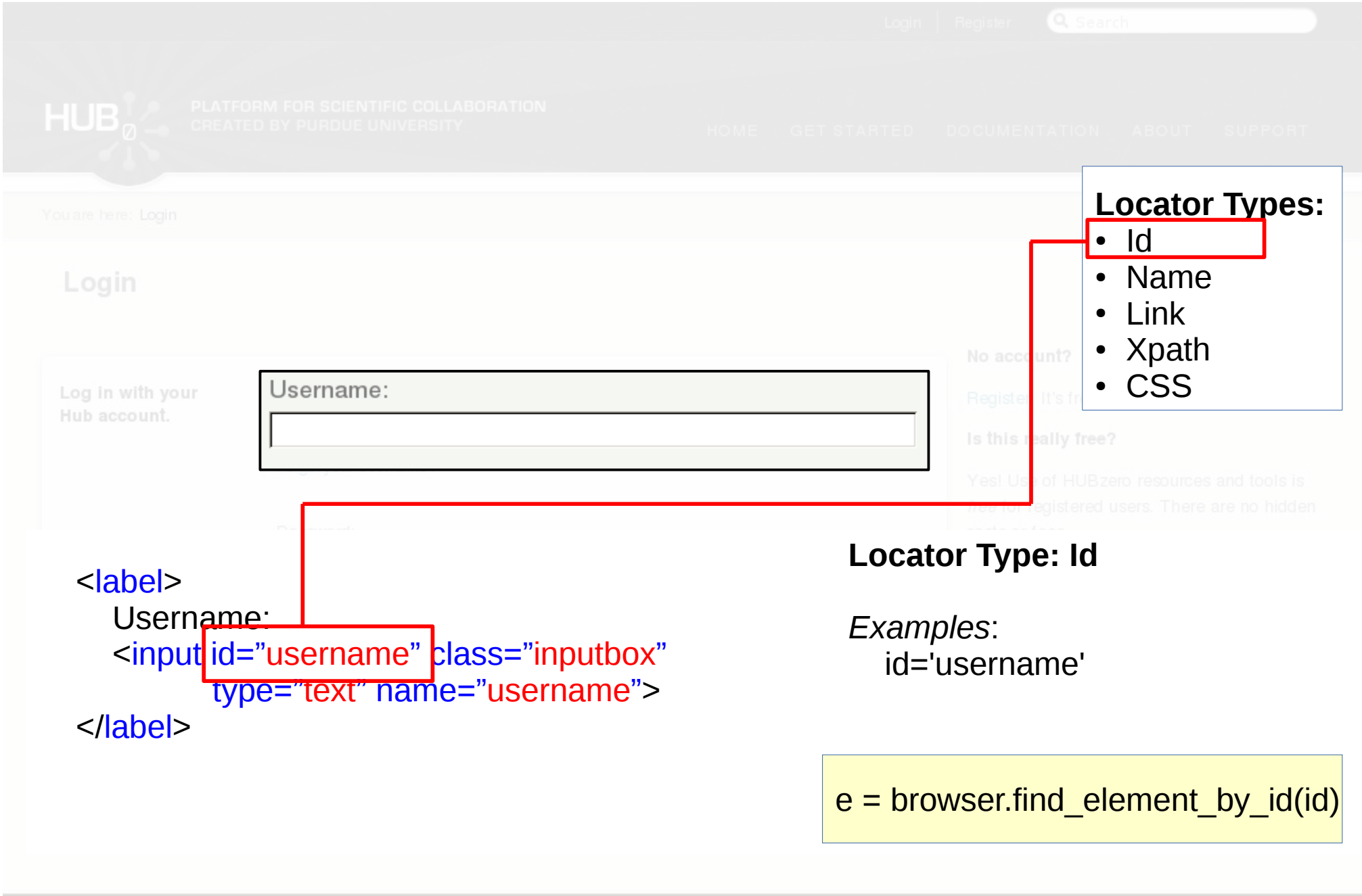
Our sponsors ask us who uses HUBzero and what they use it for. Registration helps us answer these questions. Usage statistics also focus our attention on improvements, making the HUBzero experience better for *you*.




**Locator Types:**

- Id
- Name
- Link
- Xpath
- CSS

```
<label>
   Username:
   <input id="username" class="inputbox"
          type="text" name="username">
</label>
```



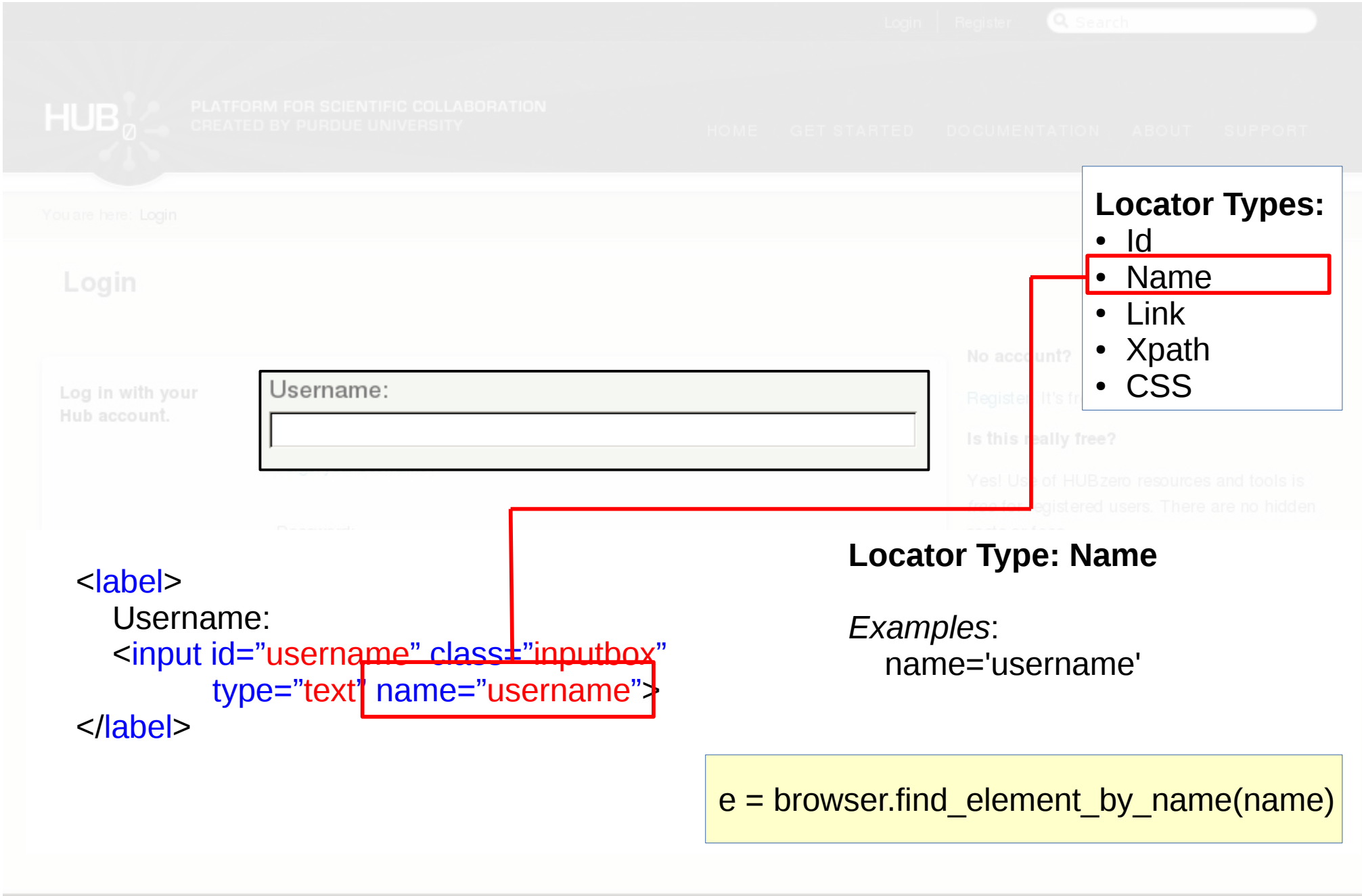

```
<label>
    Username:
    <input id="username" class="inputbox"
           type="text" name="username">
</label>
```

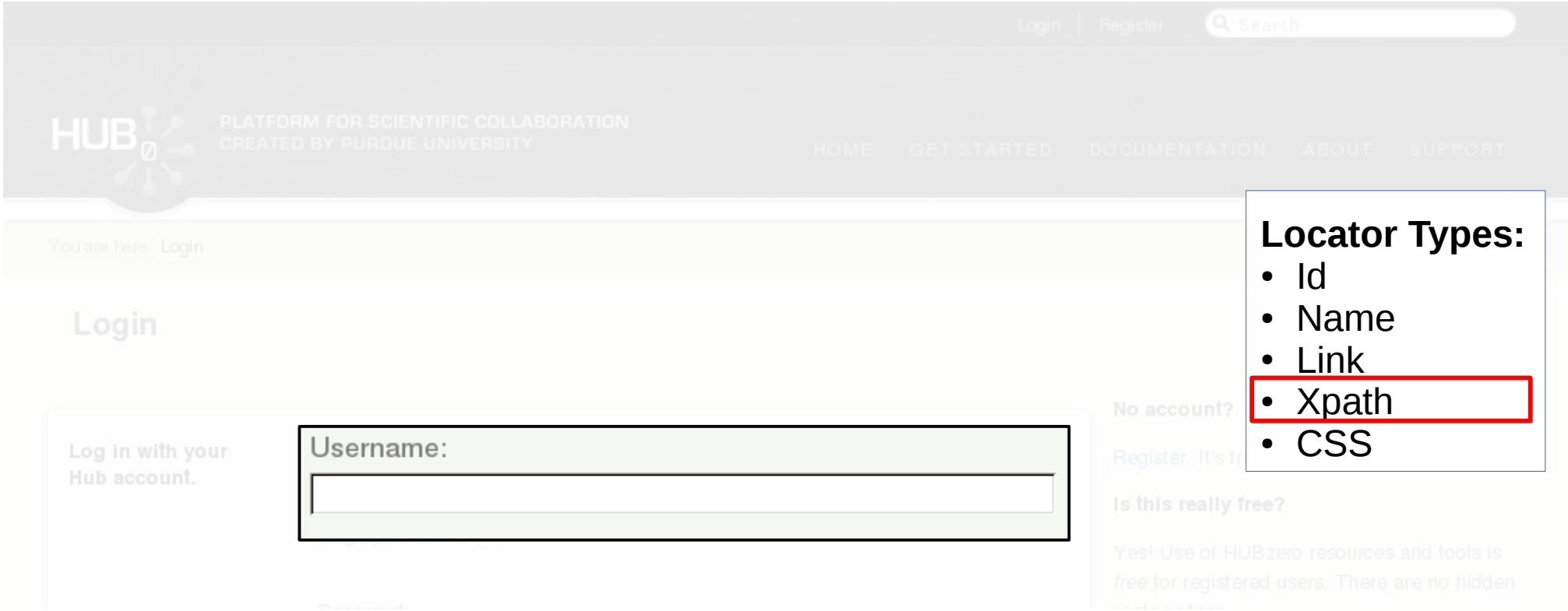
**Locator Type: Id**

*Examples*:

id='username'

```
e = browser.find_element_by_id(id)
```


Login
Register
Search
HUB
PLATFORM FOR SCIENTIFIC COLLABORATION
CREATED BY PURDUE UNIVERSITY
HOME
GET STARTED
DOCUMENTATION
ABOUT
SUPPORT
You are here: Login
Login
Log in with your Hub account.
Username:
No account?
Is this really free?
Locator Types:
• Id
• Name
• Link
• Xpath
• CSS
<label>
Username:
<input id="username" class="inputbox"
type="text" name="username">
</label>
Locator Type: Name
Examples:
name='username'
e = browser.find_element_by_name(name)




**Locator Types:**

- Id
- Name
- Link
- Xpath
- CSS

```
<label>
   Username:
   <input id=”username” class=”inputbox”
          type=”text” name=”username”>
</label>
```

**Locator Type: Name**

*Examples*:

```
loc = “//input[@id='username']”
loc = “//input[contains(@class,'inputbox')]”
```

```
e = browser.find_element_by_xpath(loc)
```

**Locator Types:**

- Id
- Name
- Link
- Xpath
- CSS

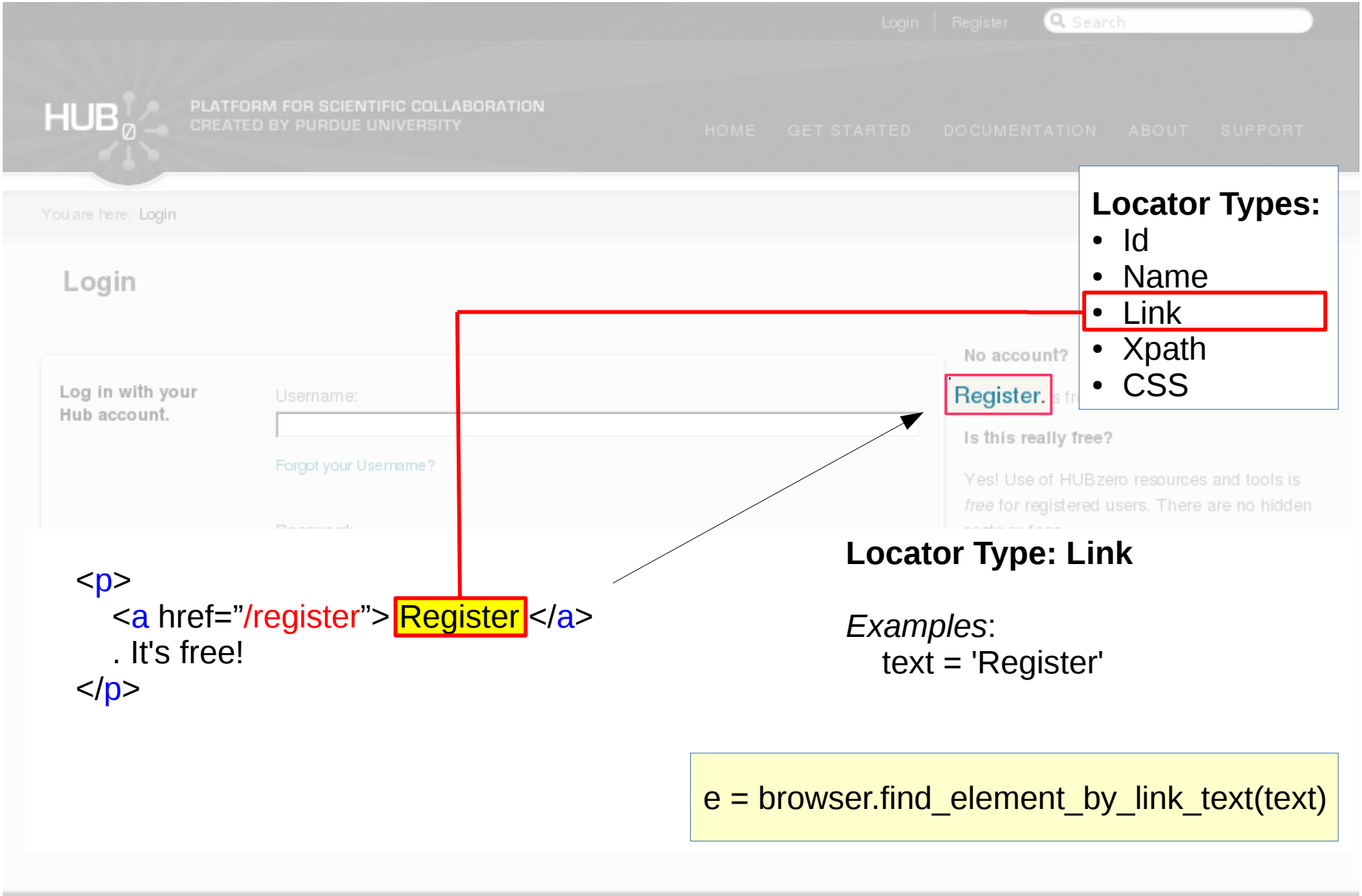
```
<label>
   Username:
   <input id="username" class="inputbox"
          type="text" name="username">
</label>
```

**Locator Type: Name**

*Examples*:

```
loc = "input[id='username']"
loc = "#username"
loc = ".inputbox"
```

```
e = browser.find_element_by_css_selector(loc)
```



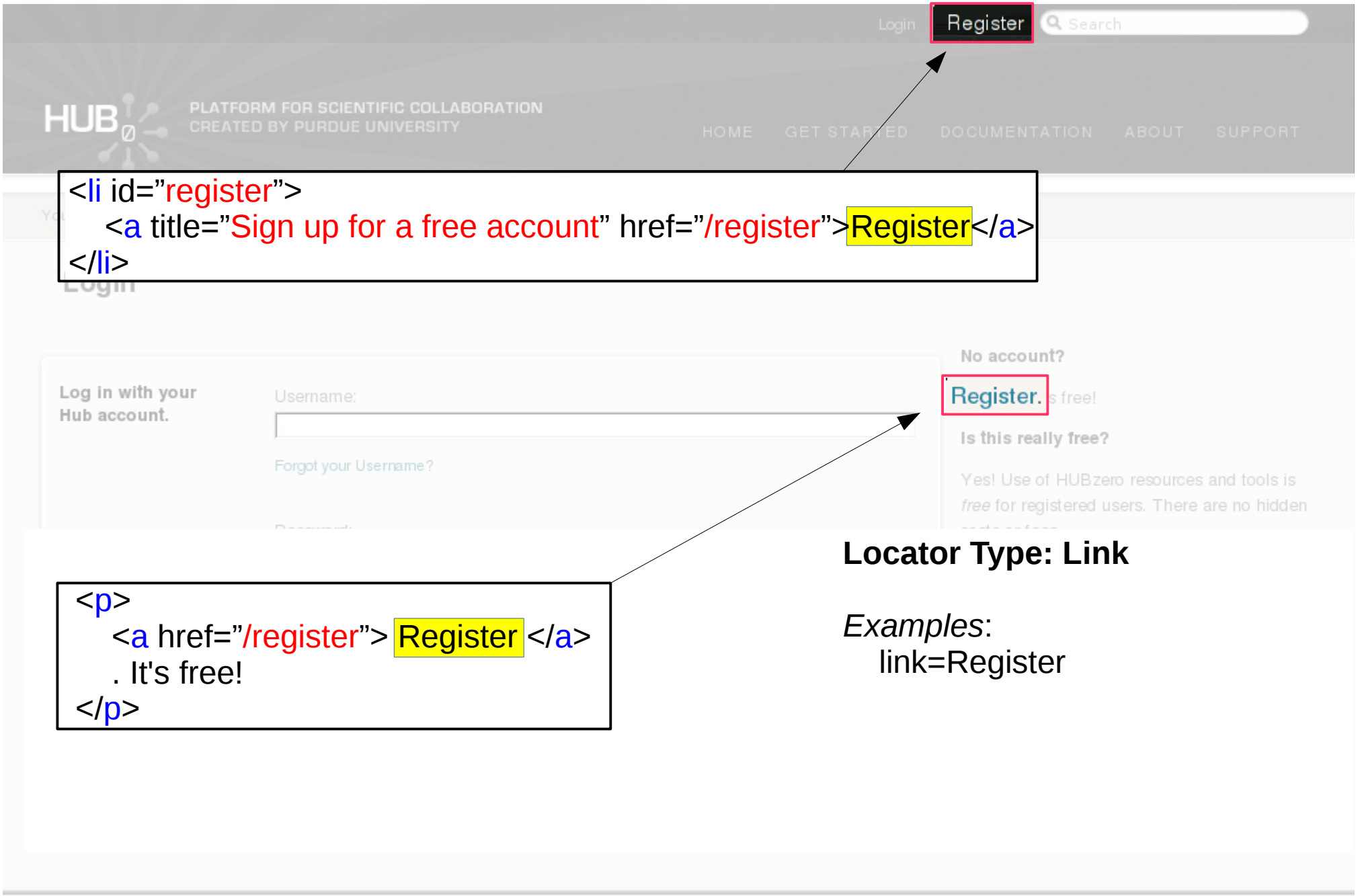

```
<p>
    <a href="/register"> Register </a>
    . It's free!
</p>
```

**Locator Type: Link**

*Examples*:

text = 'Register'

```
e = browser.find_element_by_link_text(text)
```




```
<li id="register">
   <a title="Sign up for a free account" href="/register">Register</a>
</li>
```

```
<p>
   <a href="/register"> Register </a>
   . It's free!
</p>
```

**Locator Type: Link**

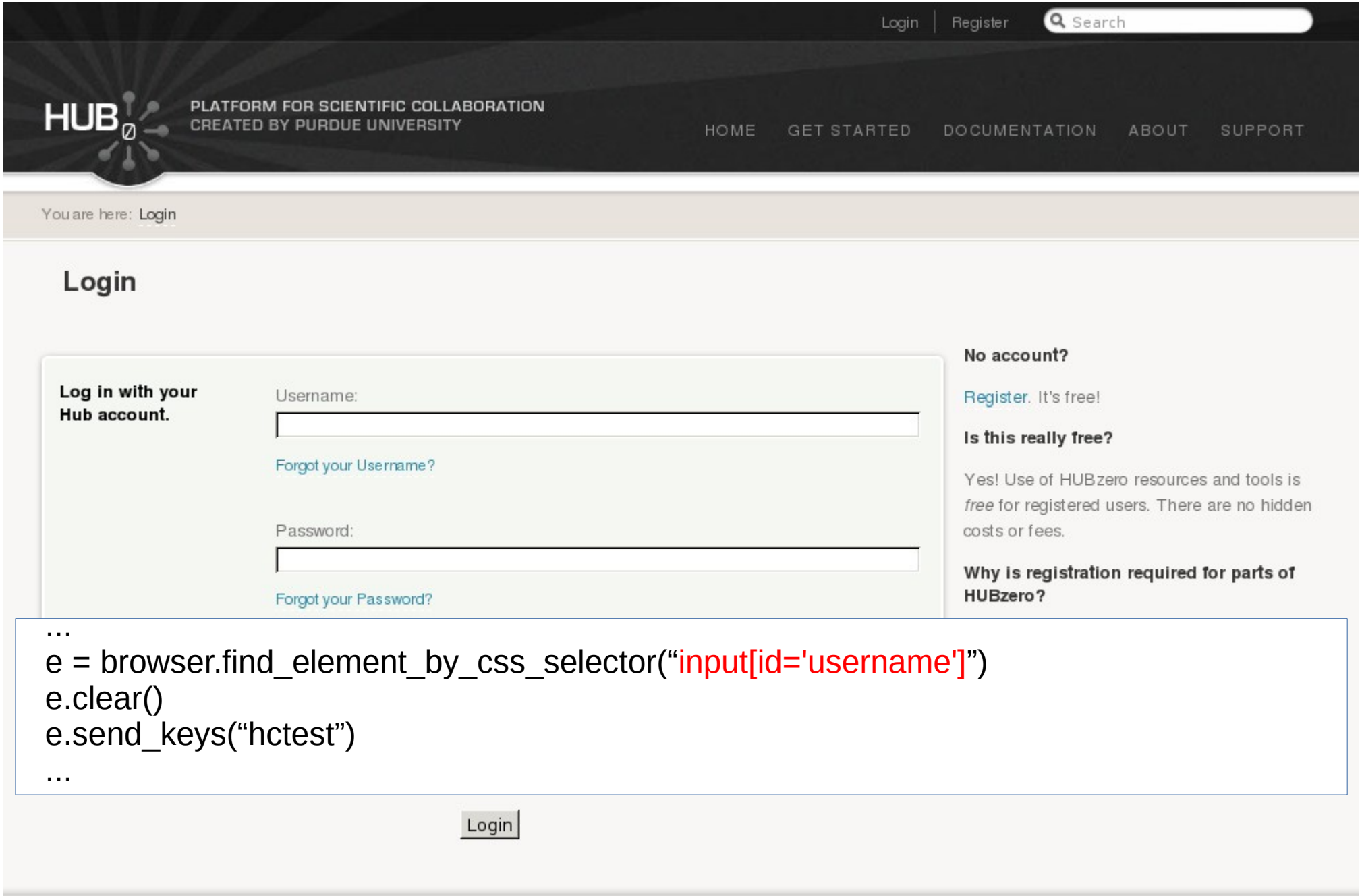
*Examples*:

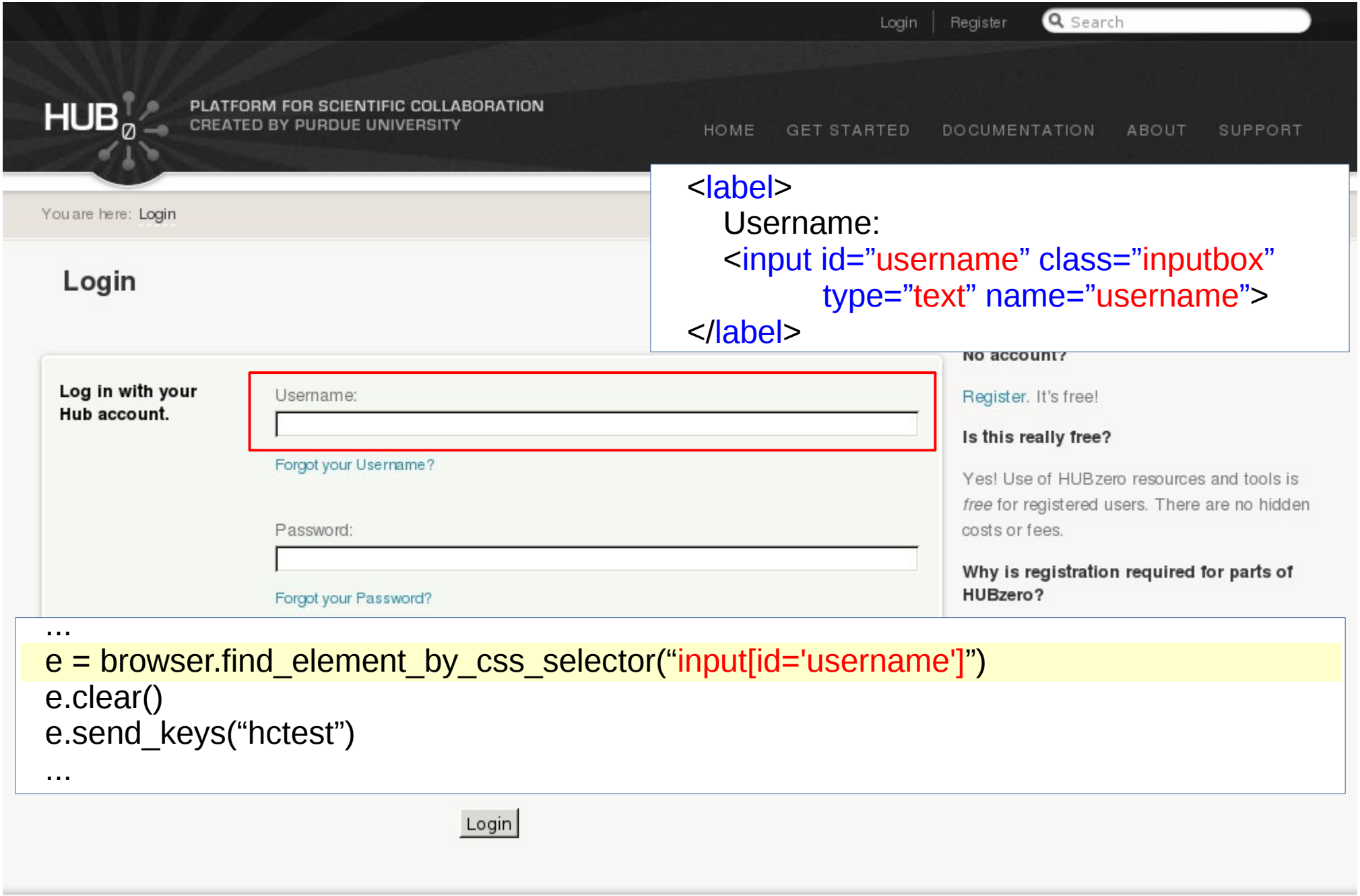
link=Register

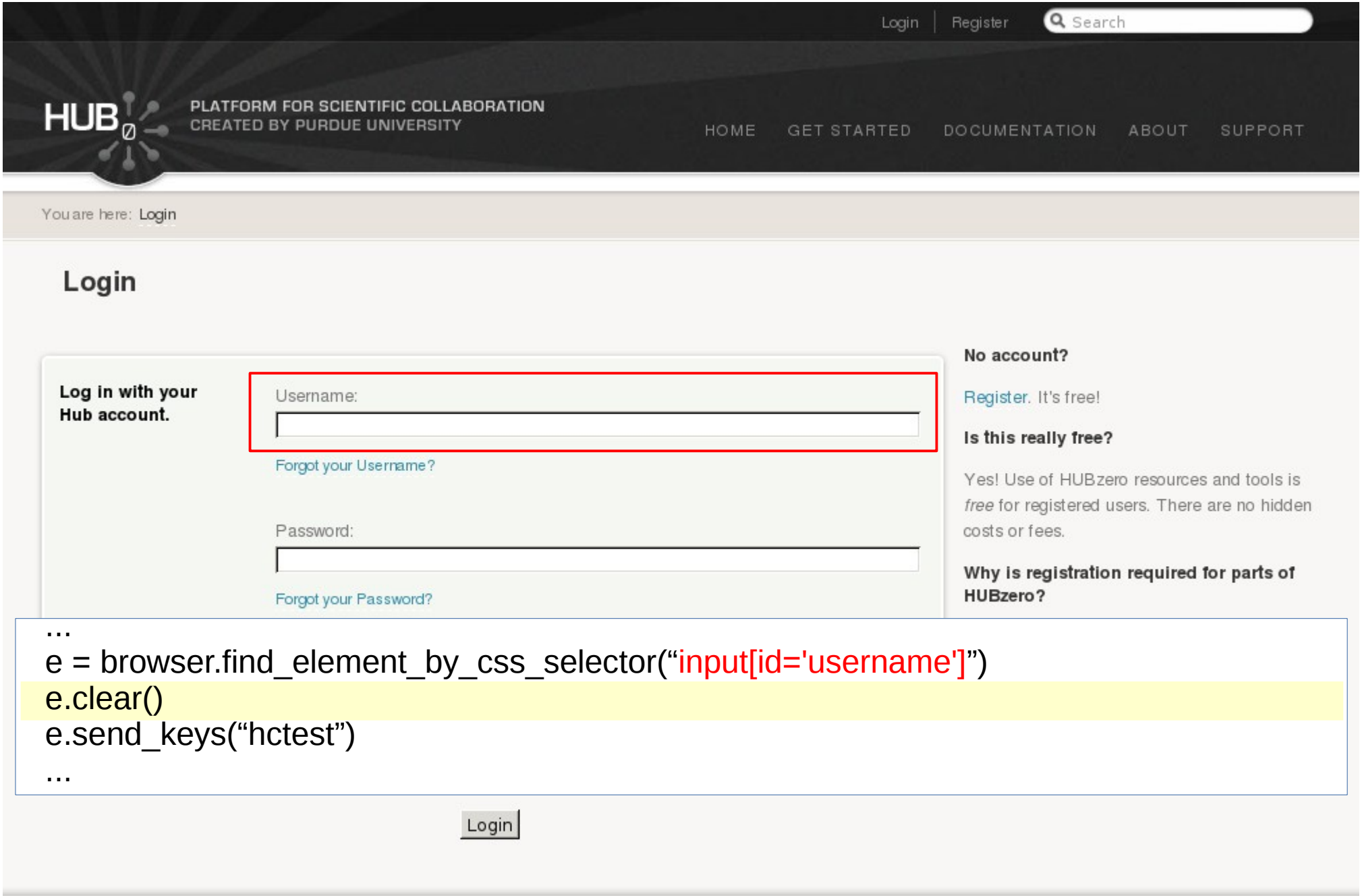
```
<p>
    <a href="/register"> Register </a>
    . It's free!
</p>
```

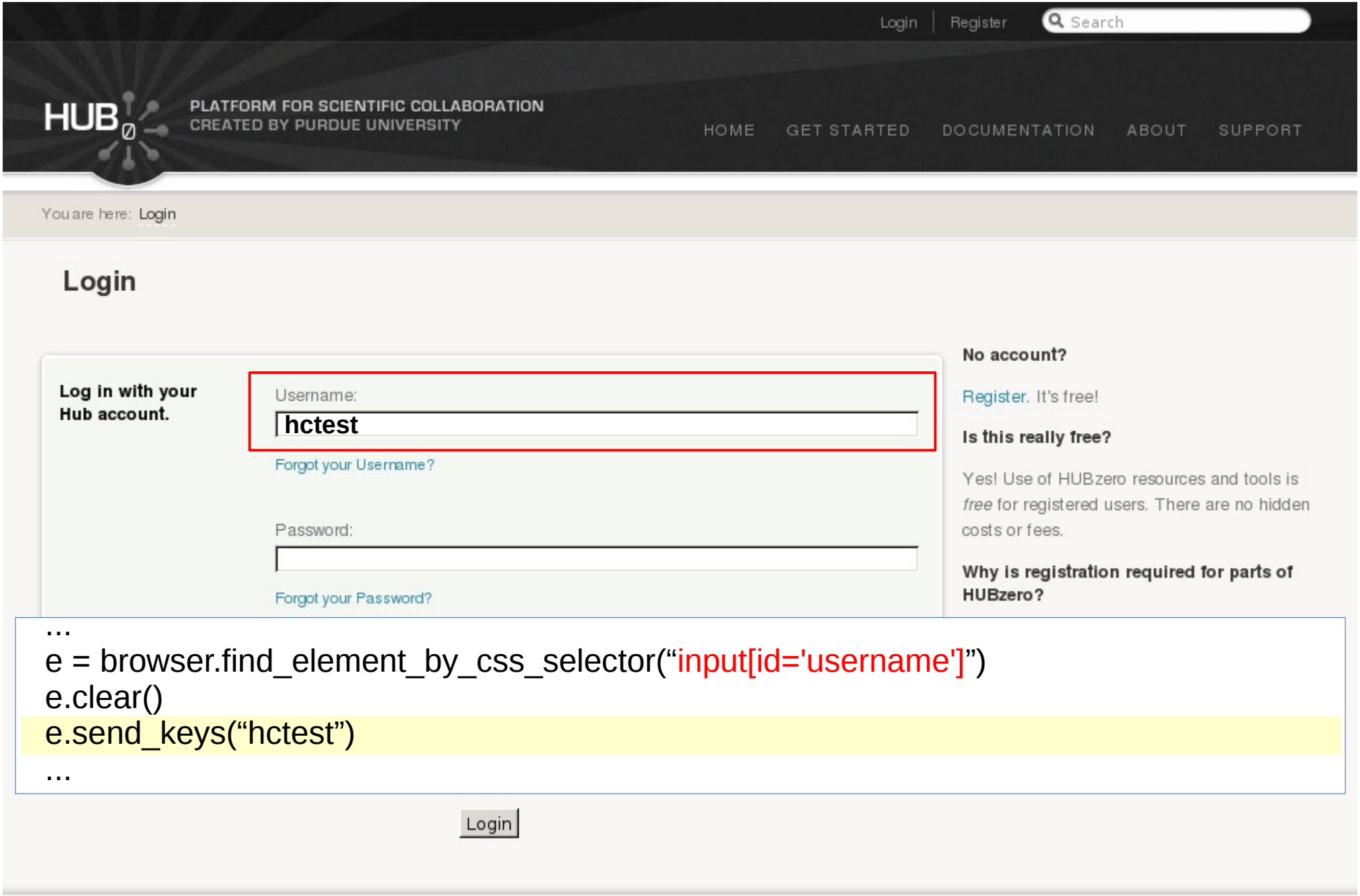
```
css="li[id='register'] > a"
```

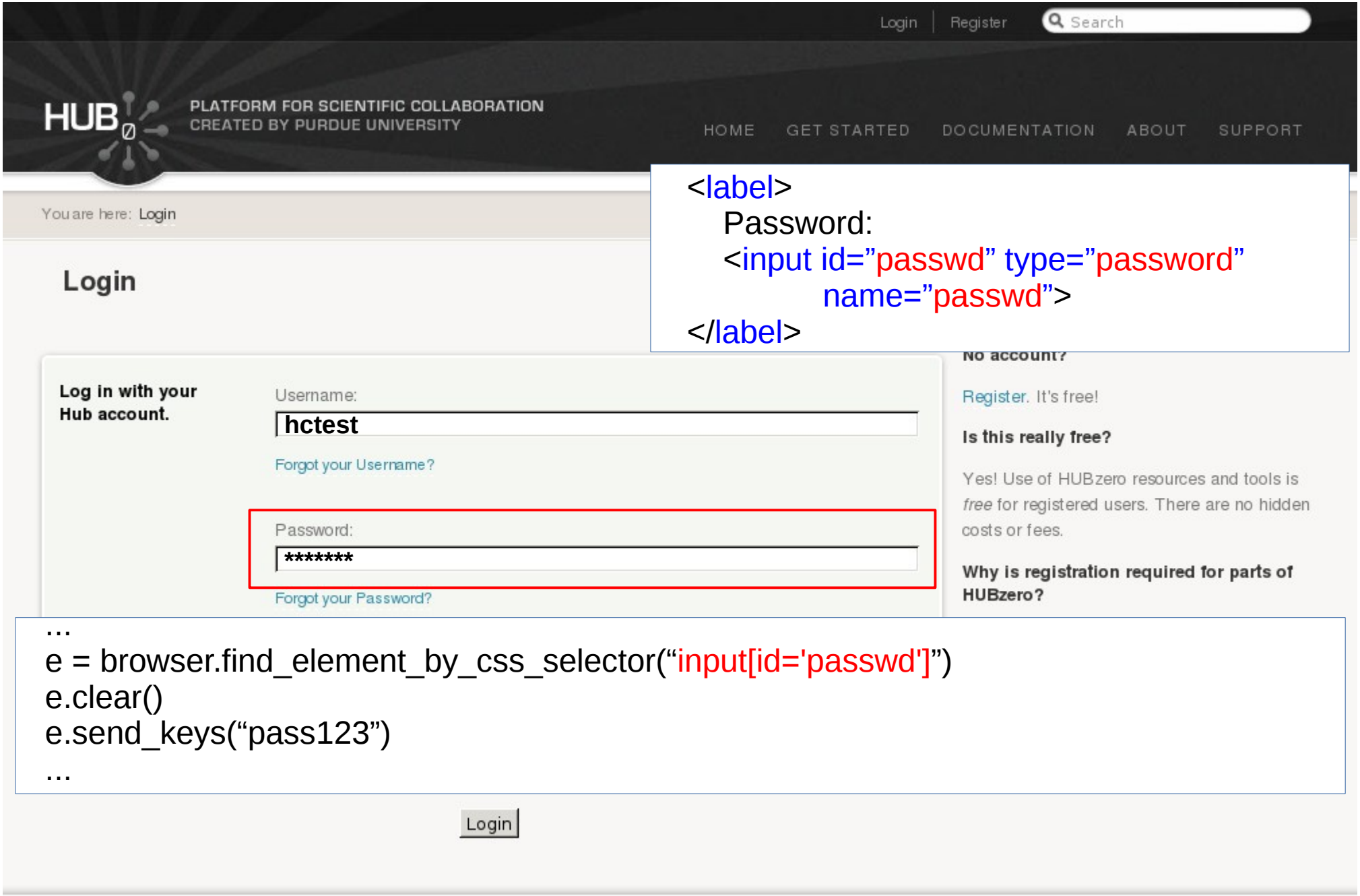
```
css="a[title='Sign up for a free account']"
```

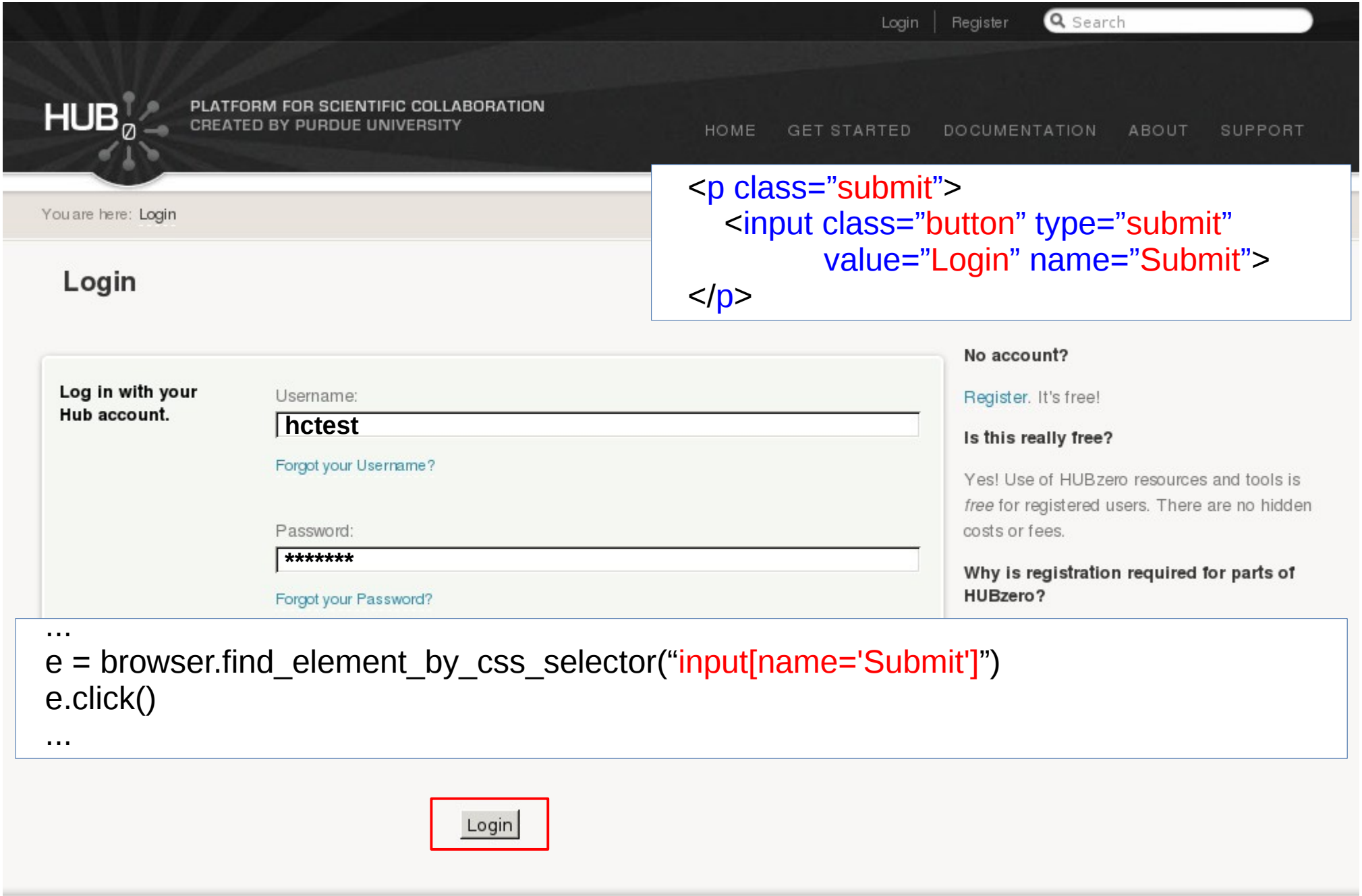

Login
Register
Search
HUB
PLATFORM FOR SCIENTIFIC COLLABORATION
CREATED BY PURDUE UNIVERSITY
HOME
GET STARTED
DOCUMENTATION
ABOUT
SUPPORT
You are here: Login
Login
Log in with your Hub account.
Username:
Forgot your Username?
Password:
Forgot your Password?
No account?
Register. It's free!
Is this really free?
Yes! Use of HUBzero resources and tools is free for registered users. There are no hidden costs or fees.
Why is registration required for parts of HUBzero?
...
e = browser.find_element_by_css_selector("input[id='username']")
e.clear()
e.send_keys("hctest")
...
Login


Login
Register
Search
HUB
PLATFORM FOR SCIENTIFIC COLLABORATION
CREATED BY PURDUE UNIVERSITY
HOME
GET STARTED
DOCUMENTATION
ABOUT
SUPPORT
You are here: Login
Login
<label>
Username:
<input id="username" class="inputbox"
type="text" name="username">
</label>
Log in with your Hub account.
Username:
Forgot your Username?
Password:
Forgot your Password?
No account?
Register. It's free!
Is this really free?
Yes! Use of HUBzero resources and tools is free for registered users. There are no hidden costs or fees.
Why is registration required for parts of HUBzero?
...
e = browser.find_element_by_css_selector("input[id='username']")
e.clear()
e.send_keys("hctest")
...
Login


Login
Register
Search
HUB
PLATFORM FOR SCIENTIFIC COLLABORATION
CREATED BY PURDUE UNIVERSITY
HOME
GET STARTED
DOCUMENTATION
ABOUT
SUPPORT
You are here: Login
Login
Log in with your Hub account.
Username:
Forgot your Username?
Password:
Forgot your Password?
No account?
Register. It's free!
Is this really free?
Yes! Use of HUBzero resources and tools is free for registered users. There are no hidden costs or fees.
Why is registration required for parts of HUBzero?
...
e = browser.find_element_by_css_selector("input[id='username']")
e.clear()
e.send_keys("hctest")
...
Login

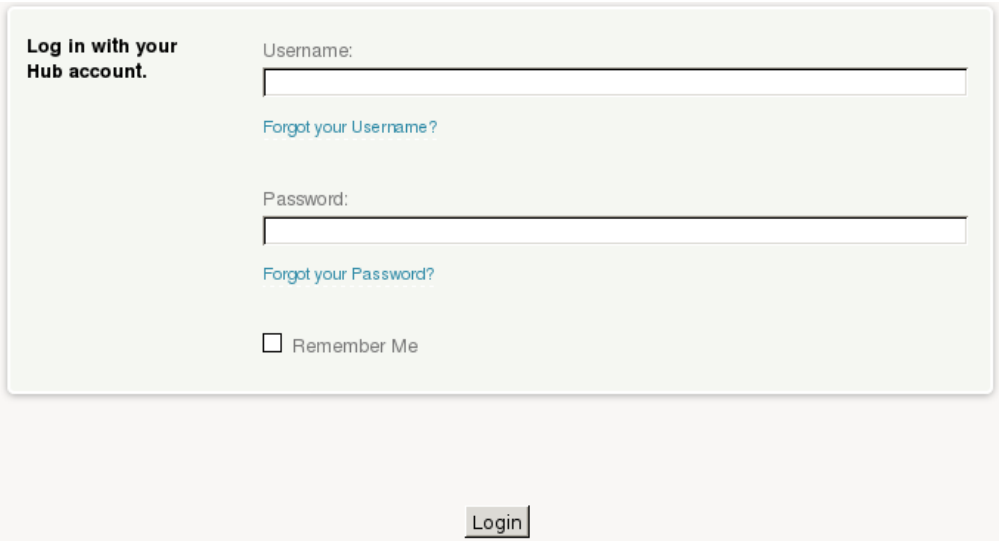

Login
Register
Search
HUB
PLATFORM FOR SCIENTIFIC COLLABORATION
CREATED BY PURDUE UNIVERSITY
HOME
GET STARTED
DOCUMENTATION
ABOUT
SUPPORT
You are here: Login
Login
Log in with your Hub account.
Username:
hctest
Forgot your Username?
Password:
Forgot your Password?
No account?
Register. It's free!
Is this really free?
Yes! Use of HUBzero resources and tools is free for registered users. There are no hidden costs or fees.
Why is registration required for parts of HUBzero?
…
e = browser.find_element_by_css_selector("input[id='username']")
e.clear()
e.send_keys("hctest")
…
Login


Login
Register
Search
HUB
PLATFORM FOR SCIENTIFIC COLLABORATION
CREATED BY PURDUE UNIVERSITY
HOME
GET STARTED
DOCUMENTATION
ABOUT
SUPPORT
You are here: Login
Login
<label>
Password:
<input id="passwd" type="password"
name="passwd">
</label>
Log in with your Hub account.
Username:
hctest
Forgot your Username?
Password:
*******
Forgot your Password?
No account?
Register. It's free!
Is this really free?
Yes! Use of HUBzero resources and tools is free for registered users. There are no hidden costs or fees.
Why is registration required for parts of HUBzero?
...
e = browser.find_element_by_css_selector("input[id='passwd']")
e.clear()
e.send_keys("pass123")
...
Login


Login
Register
Search
HUB
PLATFORM FOR SCIENTIFIC COLLABORATION
CREATED BY PURDUE UNIVERSITY
HOME
GET STARTED
DOCUMENTATION
ABOUT
SUPPORT
You are here: Login
Login

<input class="button" type="submit"
value="Login" name="Submit">

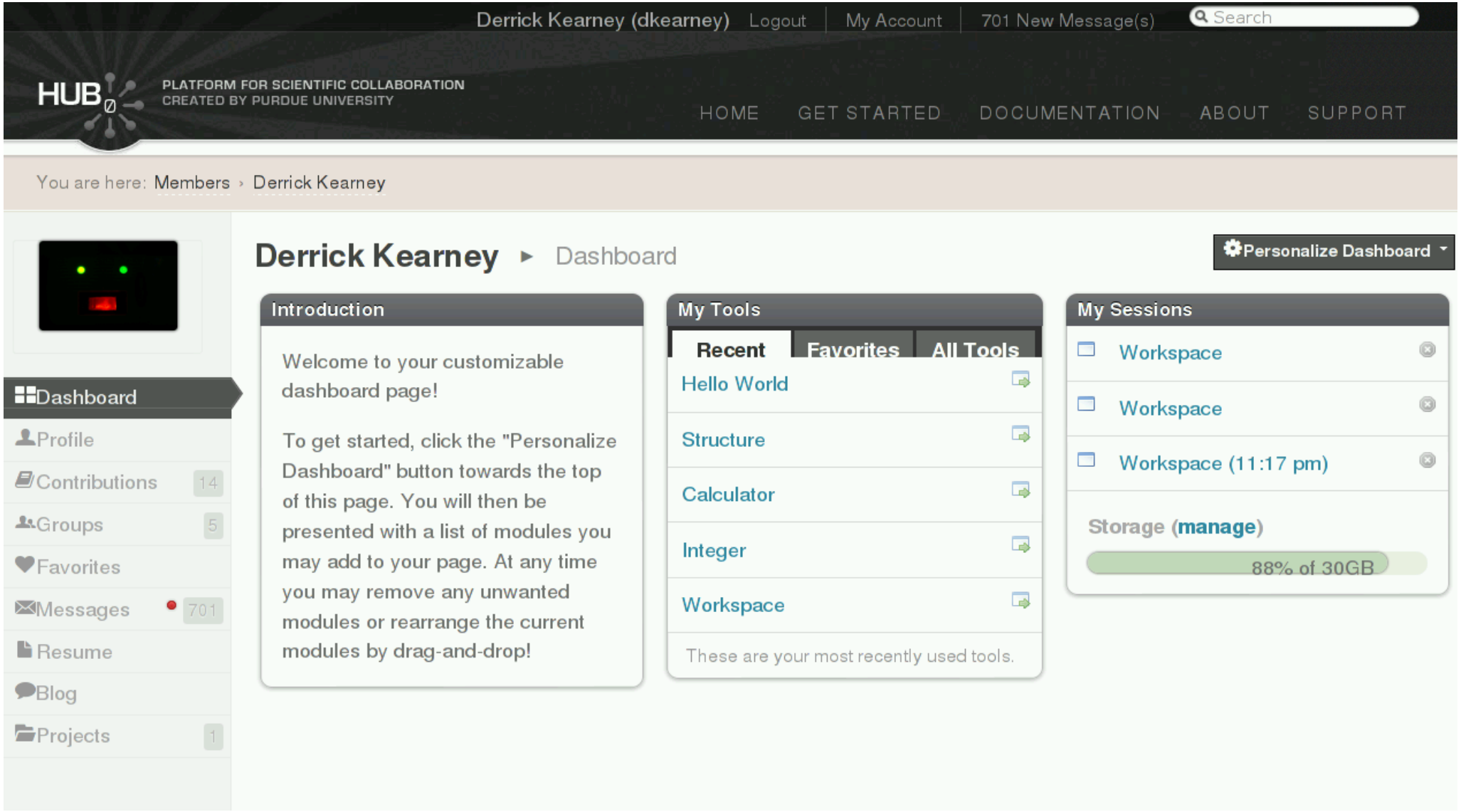
Log in with your Hub account.
Username:
hctest
Forgot your Username?
Password:
*******
Forgot your Password?
No account?
Register. It's free!
Is this really free?
Yes! Use of HUBzero resources and tools is free for registered users. There are no hidden costs or fees.
Why is registration required for parts of HUBzero?
...
e = browser.find_element_by_css_selector("input[name='Submit']")
e.click()
...
Login

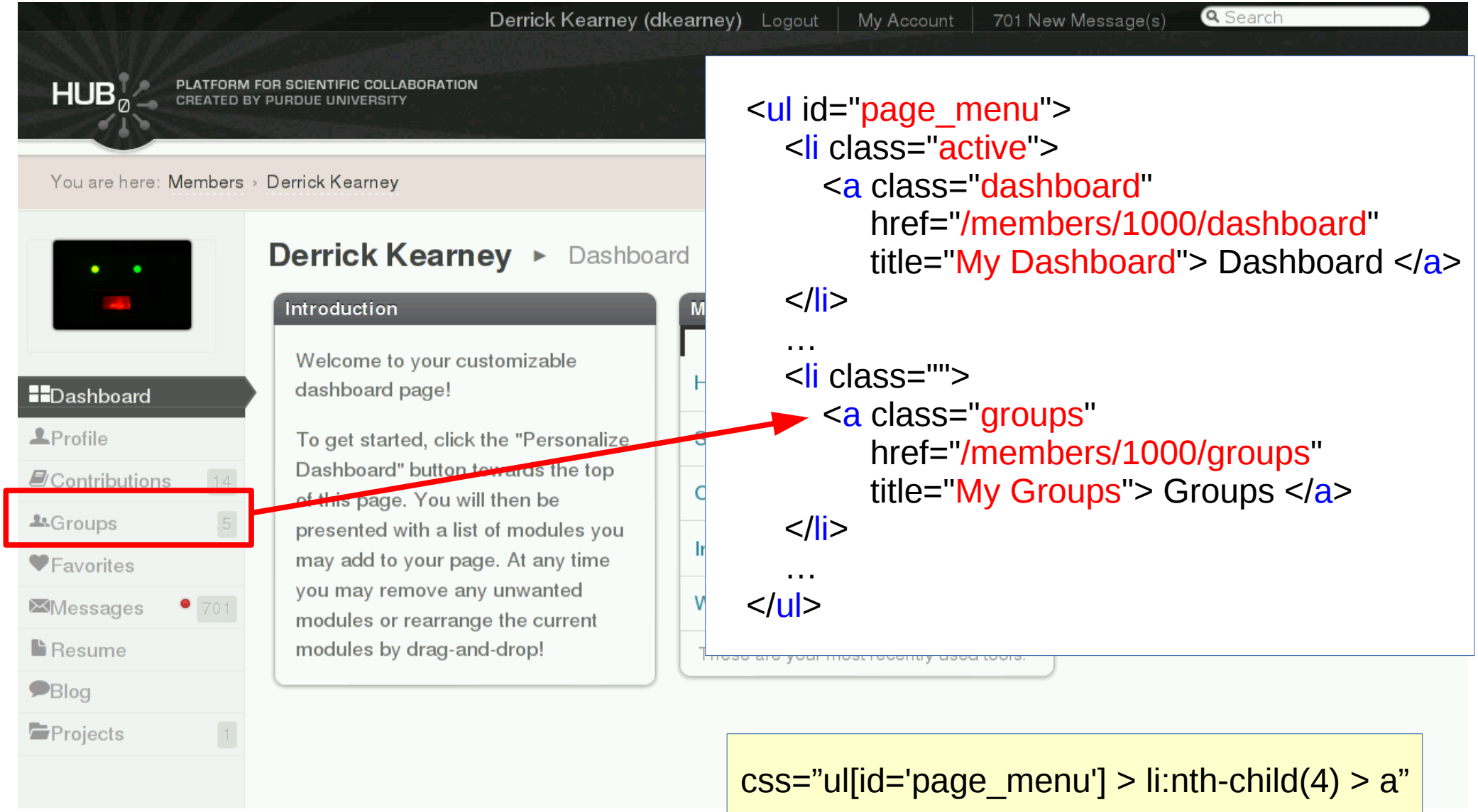



```
...
e = browser.find_element_by_css_selector("input[id='username']")
e.clear()
e.send_keys("hctest")

e = browser.find_element_by_css_selector("input[id='passwd']")
e.clear()
e.send_keys("pass123")

e = browser.find_element_by_css_selector("input[name='Submit']")
e.click()
...
```

# Login automation script

```
from selenium import webdriver

url = 'http://hubzero.org'

browser = webdriver.Firefox()
browser.get(url)

e = browser.find_element_by_css_selector("input[id='username']")
e.clear()
e.send_keys("hctest")

e = browser.find_element_by_css_selector("input[id='passwd']")
e.clear()
e.send_keys("pass123")

e = browser.find_element_by_css_selector("input[name='Submit']")
e.click()

assert(...)

browser.quit()
```


Derrick Kearney (dkearney)
Logout
My Account
701 New Message(s)
Search
HUB
PLATFORM FOR SCIENTIFIC COLLABORATION
CREATED BY PURDUE UNIVERSITY
HOME
GET STARTED
DOCUMENTATION
ABOUT
SUPPORT
You are here: Members › Derrick Kearney
Dashboard
Profile
Contributions 14
Groups 5
Favorites
Messages 701
Resume
Blog
Projects 1
Derrick Kearney ▸ Dashboard
Personalize Dashboard
Introduction
Welcome to your customizable dashboard page!
To get started, click the "Personalize Dashboard" button towards the top of this page. You will then be presented with a list of modules you may add to your page. At any time you may remove any unwanted modules or rearrange the current modules by drag-and-drop!
My Tools
Recent
Favorites
All Tools
Hello World
Structure
Calculator
Integer
Workspace
These are your most recently used tools.
My Sessions
Workspace
Workspace
Workspace (11:17 pm)
Storage (manage)
88% of 30GB




css=”ul[id='page_menu'] > li:nth-child(4) > a”

css=”ul[id='page_menu'] a[class='groups']”

css=”#page_menu .groups”

# Traversing tables and lists:

***# Printing the text of the first link, from all list items:***

```
...
elist = browser.find_elements_by_css_selector('#page_menu li')

for e in elist:
    link = e.find_element_by_css_selector('a')
    print 'link text = %s' % (link.text())
...
```

```
<ul id="page_menu">
    <li class="active">
        <a class="dashboard"
            href="/members/1000/dashboard"
            title="My Dashboard"> Dashboard </a>
    </li>
    …
    <li class="">
        <a class="groups"
            href="/members/1000/groups"
            title="My Groups"> Groups </a>
    </li>
    …
</ul>
```

# Traversing tables and lists:

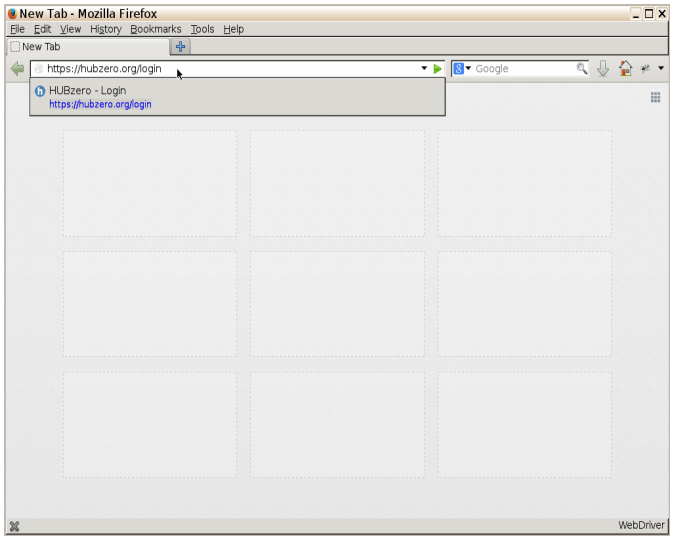
***# Printing the title attribute of the first link, from all list items:***

```
...
elist = browser.find_elements_by_css_selector('#page_menu li')

for e in elist:
    link = e.find_element_by_css_selector('a')
    print 'link text = %s' % (link.get_attribute('title'))
...
```

```
<ul id="page_menu">
    <li class="active">
        <a class="dashboard"
            href="/members/1000/dashboard"
            title="My Dashboard"> Dashboard </a>
    </li>
    …
    <li class="">
        <a class="groups"
            href="/members/1000/groups"
            title="My Groups"> Groups </a>
    </li>
    …
</ul>
```

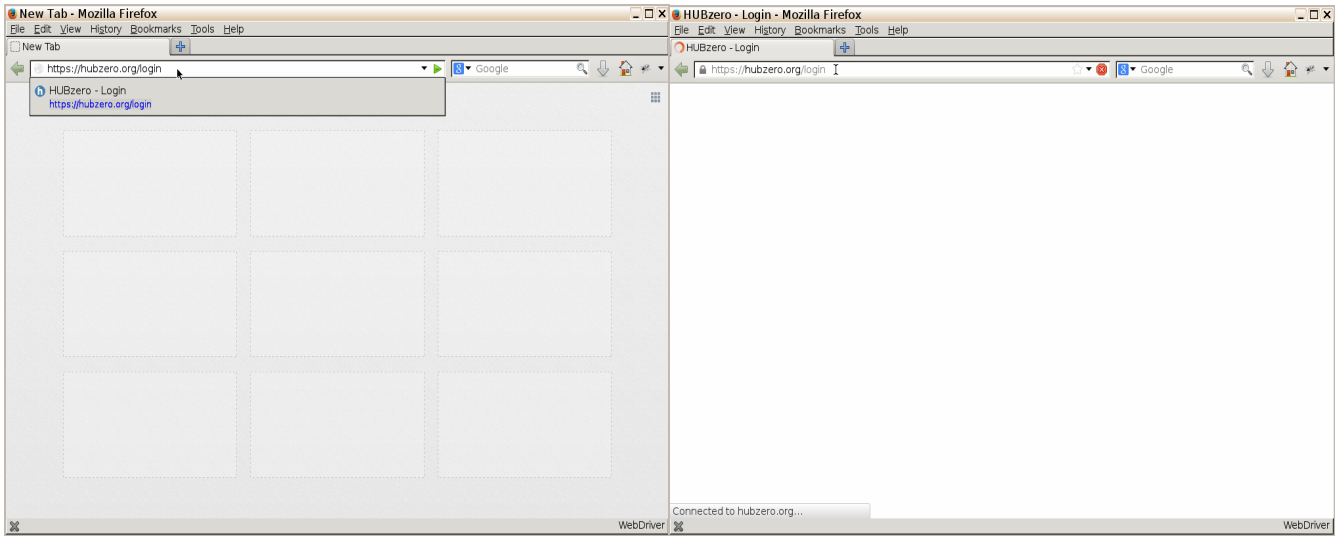
# Implicit waits:

```
from selenium import webdriver

url = 'https://hubzero.org/login'

browser = webdriver.Firefox()

browser.get(url)

e = browser.find_element_by_css_selector(“input[id='username']”)
e.clear()
e.send_keys('hctest')

browser.quit()
```



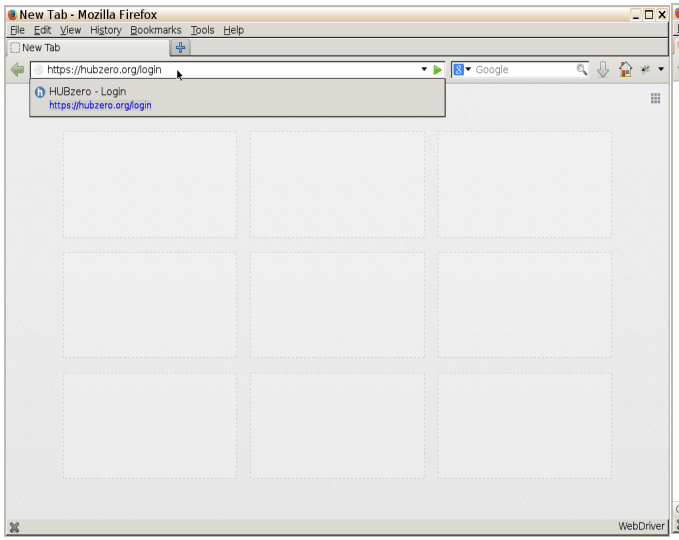
# Implicit waits:

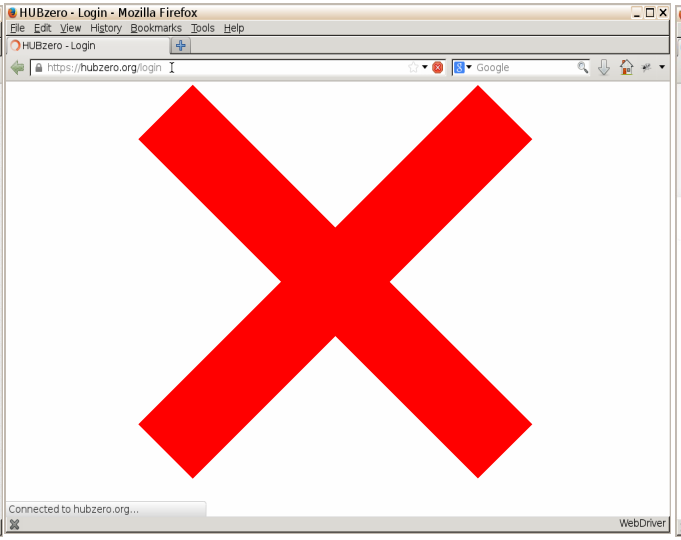
```
from selenium import webdriver

url = 'https://hubzero.org/login'

browser = webdriver.Firefox()

browser.get(url)

e = browser.find_element_by_css_selector(“input[id='username']”)
e.clear()
e.send_keys('hctest')

browser.quit()
```

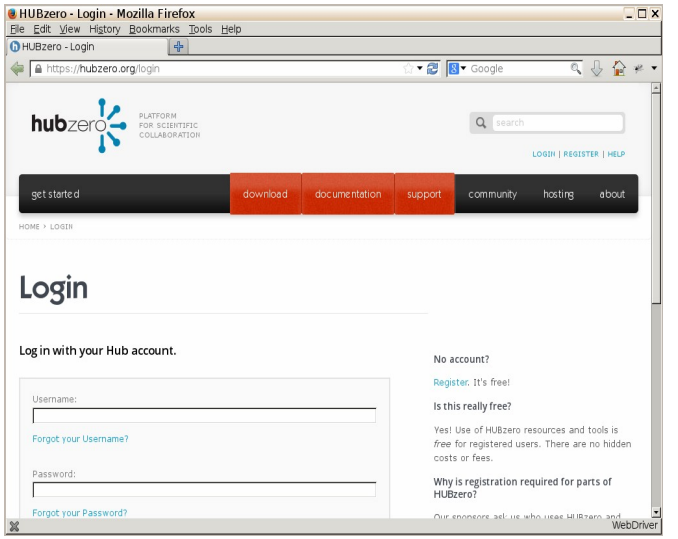


# Implicit waits:

```
from selenium import webdriver

url = 'https://hubzero.org/login'

browser = webdriver.Firefox()

browser.get(url)

e = browser.find_element_by_css_selector("input[id='username']")
e.clear()
e.send_keys('hctest')

browser.quit()
```

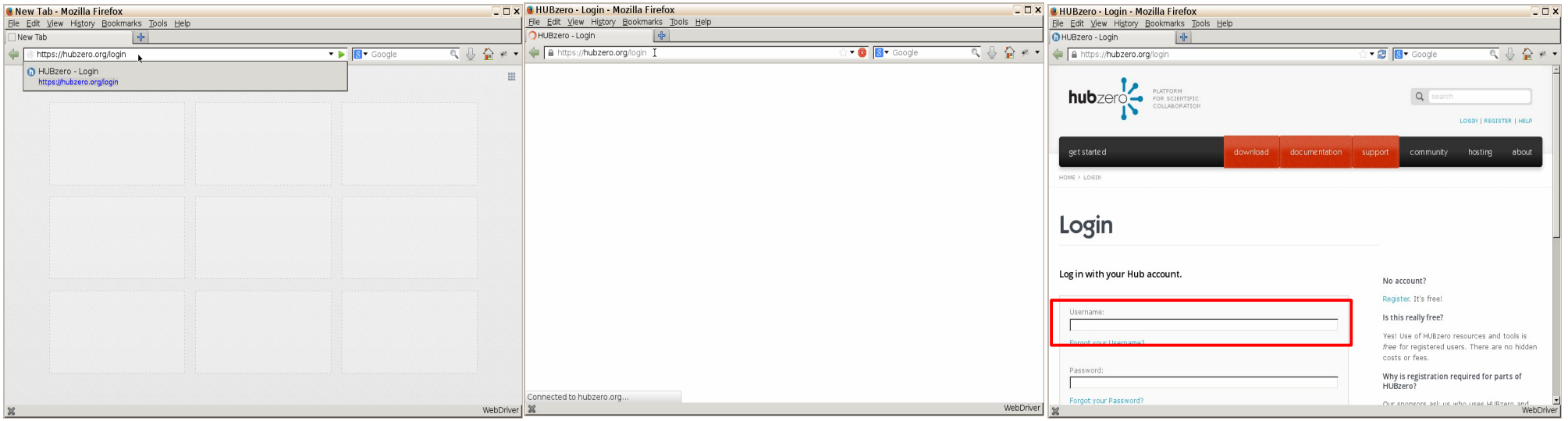


# Implicit waits:

```
from selenium import webdriver

url = 'https://hubzero.org/login'

browser = webdriver.Firefox()
browser.implicitly_wait(30)
browser.get(url)

e = browser.find_element_by_css_selector("input[id='username']")
e.clear()
e.send_keys('hctest')

browser.quit()
```



# Explicit waits:

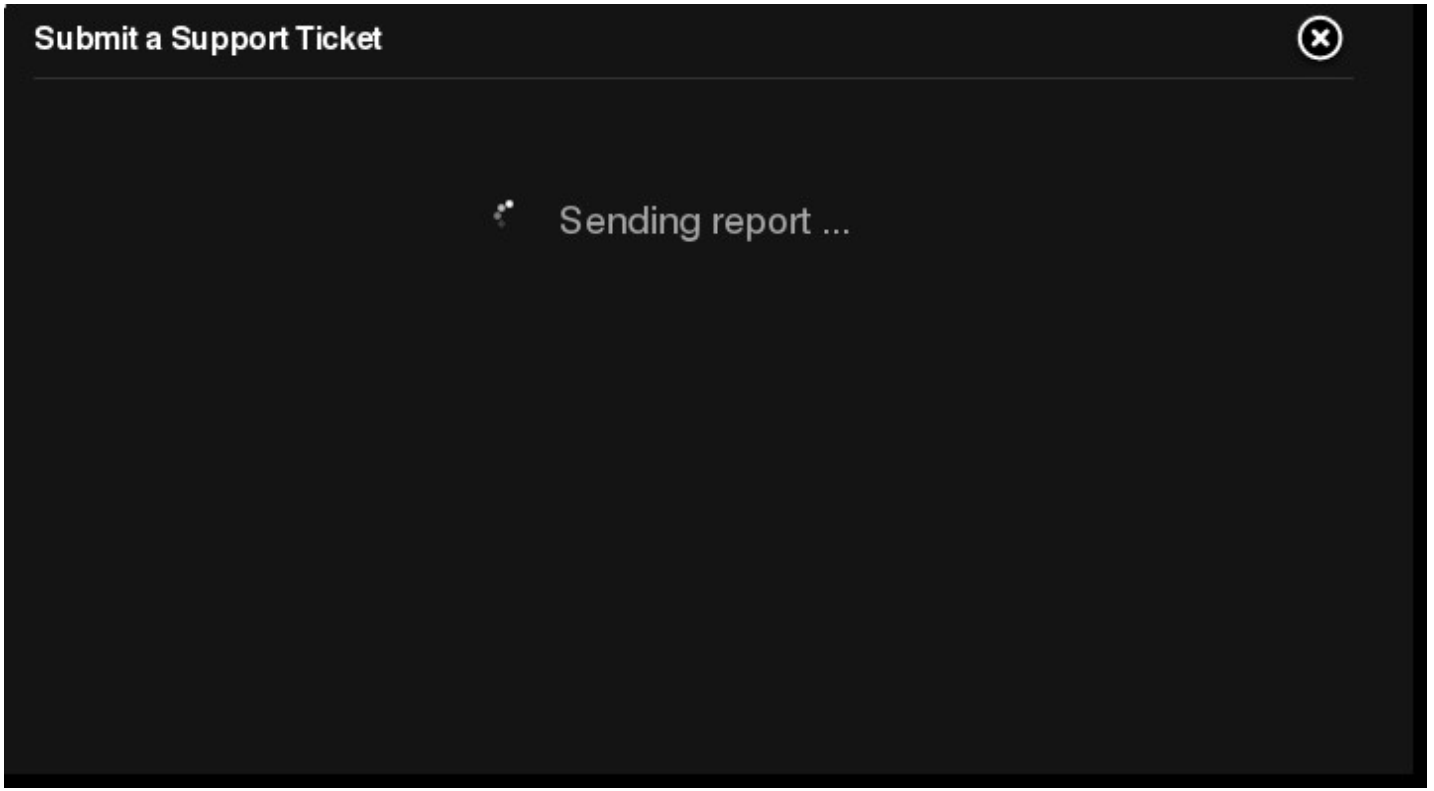
```
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC

url = 'https://hubzero.org/login'

browser = webdriver.Firefox()
browser.get(url)

wait = WebDriverWait(browser, 10)
wait.until(CONDITION)

e = browser.find_element_by_css_selector("input[id='username']")
e.clear()
e.send_keys('hctest')

browser.quit()
```

# Explicit waits:

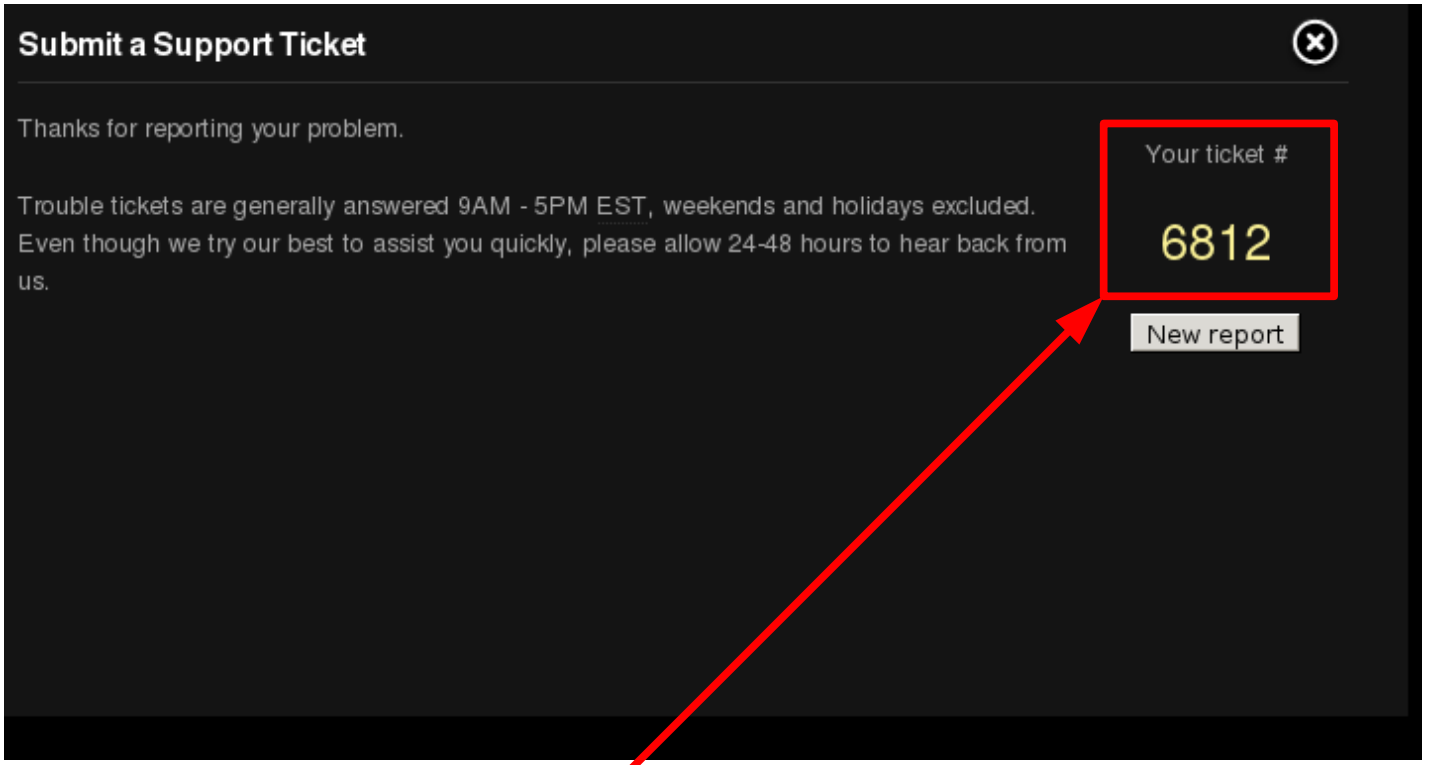
```
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC

url = 'https://hubzero.org/login'

browser = webdriver.Firefox()
browser.get(url)

wait = WebDriverWait(browser, 10)
wait.until(EC.visibility_of_element_located(
                    (By.CSS_SELECTOR,'#username')))

e = browser.find_element_by_css_selector("input[id='username']")
e.clear()
e.send_keys('hctest')

browser.quit()
```

# Dealing with AJAX:

# Dealing with AJAX:



# Dealing with AJAX:




```
<div id="trSuccess">
   <div>
      <p>Your ticket #
         <span>
            <a title=”View ticket” href="/support/ticket/6812">6812</a>
         </span>
      </p>
   </div>
</div>
```

# Dealing with AJAX:

```
…

submit_ticket()

# wait for the page to refresh
wait = WebDriverWait(browser, 60)
e = wait.until(lambda browser :
                    browser.find_element_by_id("trSuccess").is_displayed())

ticket_number = e.find_element_by_css_selector('a').text()
...
```

```
<div id="trSuccess">
    <div>
        <p>Your ticket #
            <span>
                <a title=”View ticket” href="/support/ticket/6812">6812</a>
            </span>
        </p>
    </div>
</div>
```

# QUESTIONS?
# MOVIES?
# MORE?

# THE PAGE OBJECT DESIGN PATTERN

hubzero

PLATFORM
FOR SCIENTIFIC
COLLABORATION

search

LOGIN | REGISTER | HELP

get started | download | documentation | support | community | hosting | about

# HUBzero® is a powerful, open source software platform for creating dynamic web sites that support scientific research and educational activities

habricentral.org

Resources for the Study of the Human-Animal Bond.

HABRI Central exists to advance the interdisciplinary understanding and study of human-animal interaction.

see more hubs

GET STARTED

DOWNLOAD

## Support Options

**Knowledge Base**
Find information on common issues.

**Ask the Community**
Ask questions and find answers from other users.

**Wish List**
Suggest a new site feature or improvement.

**Support Tickets**
Check on status of your tickets.

## Submit a Support Ticket

Username: OPTIONAL

Name: REQUIRED

E-mail: REQUIRED

Please answer the question: REQUIRED
Eye, ankle or arm: which is part of the head?

Problem: REQUIRED

Attach a screenshot: OPTIONAL
Browse...
(.jpg, .jpeg, .jpe, .bmp, .tif, .tiff, .png, .gif)

Submit

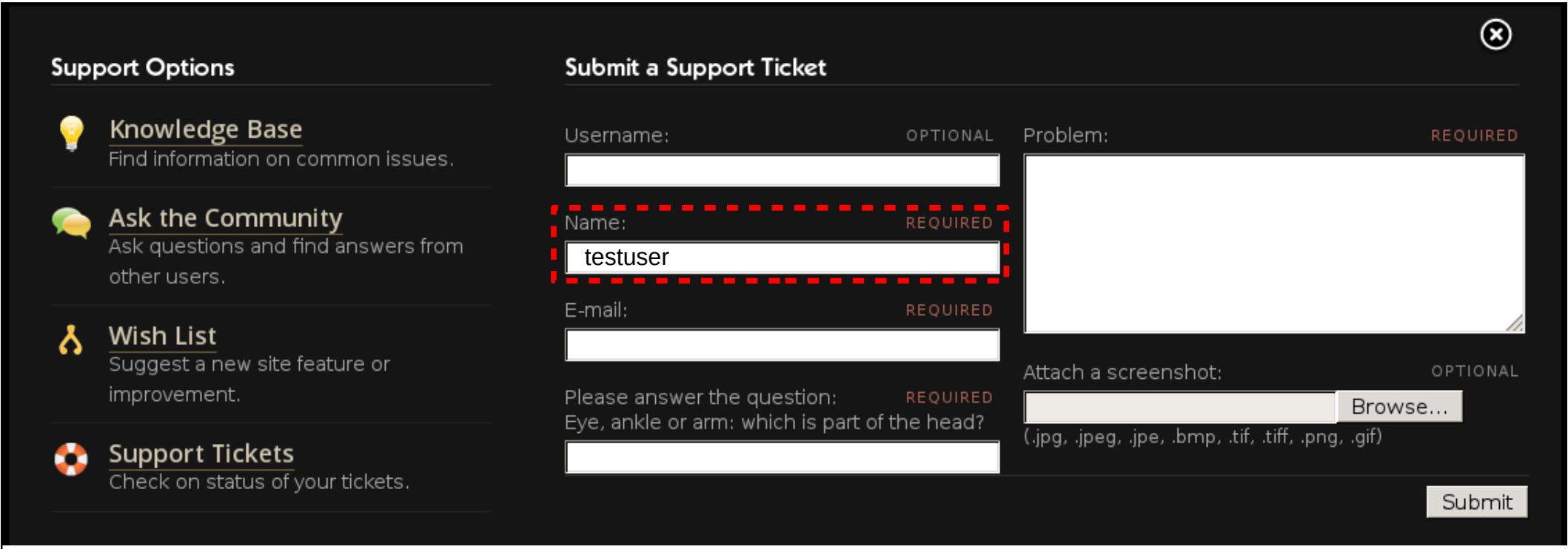
hubzero
PLATFORM FOR SCIENTIFIC COLLABORATION

search

LOGIN | REGISTER | HELP

get started | download | documentation | support | community | hosting | about

HUBzero® is a powerful, open source software platform for creating dynamic web sites that support scientific research and educational activities



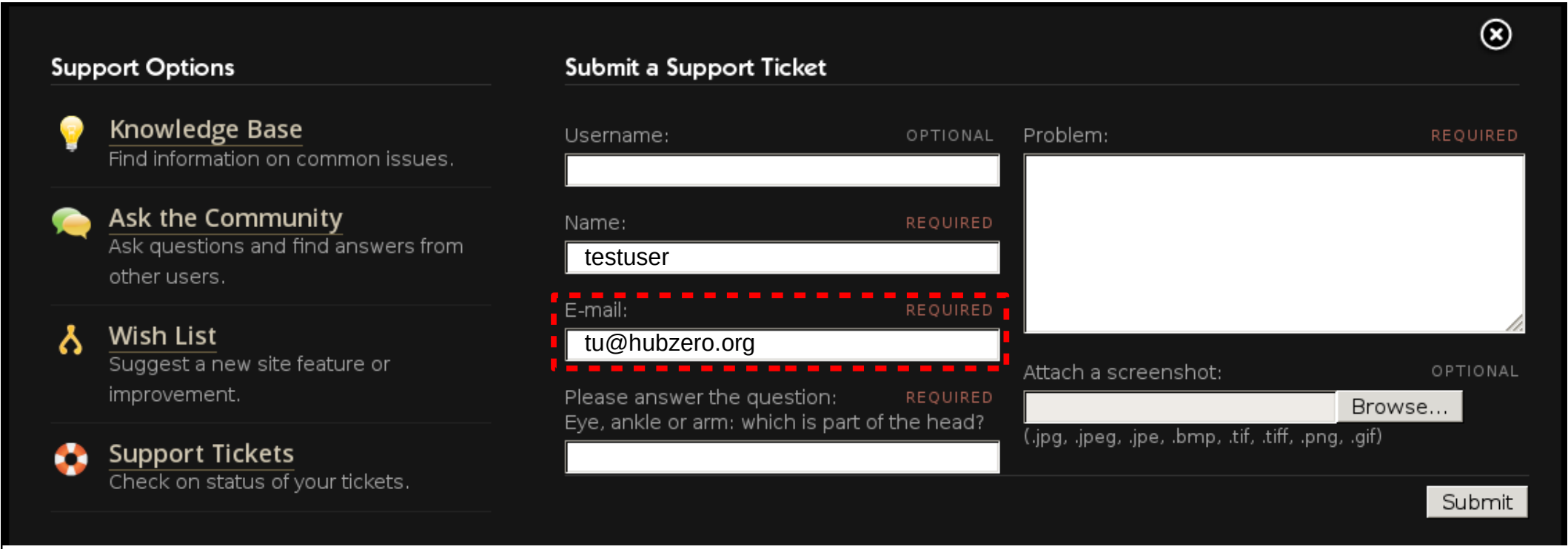

Writing to a Text Input:

1. Find the element
2. Clear the element
3. Send keys to the element

```
<label for="trName">
    Name:
    <span class="required">required</span>
    <input id="trName" type="text" value=""
        name="reporter[name]">
</label>
```

---

```
...
e = browser.find_element(by=BY.CSS, value="input[id='trName']")
e.clear()
e.send_keys("testuser")
...
```



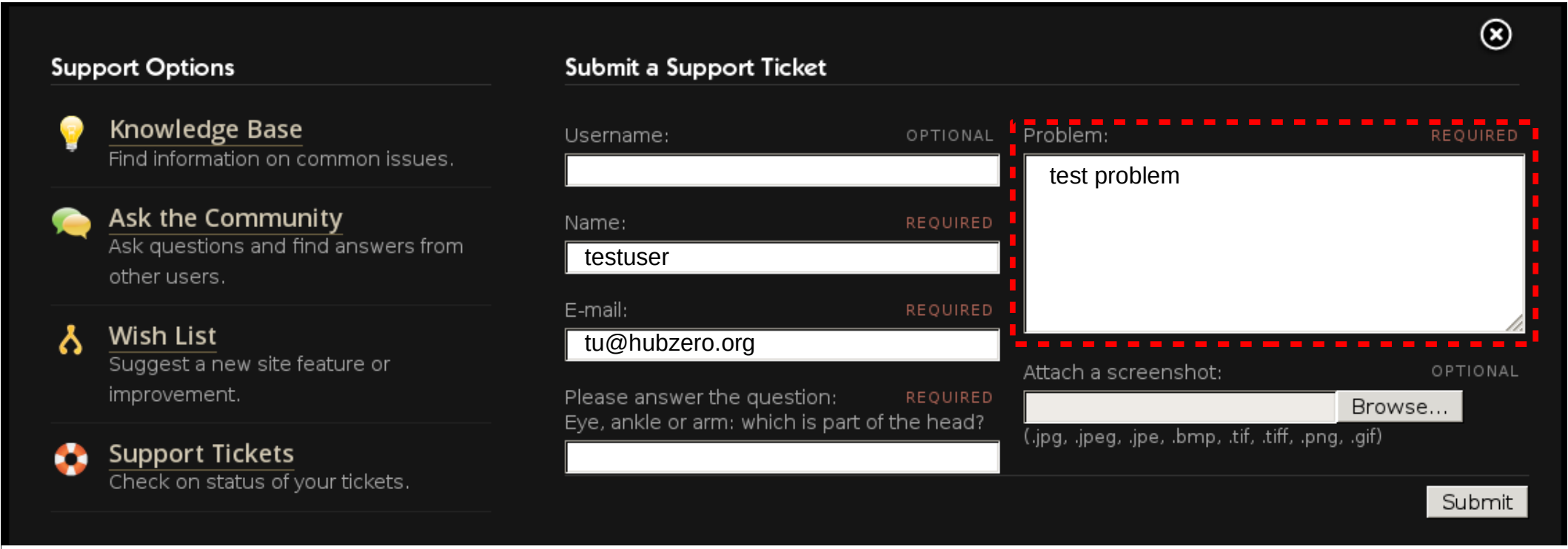

Writing to a Text Input:

1. Find the element
2. Clear the element
3. Send keys to the element

```
<label for="trEmail">
    E-mail:
    <span class="required">required</span>
    <input id="trEmail" type="text" value=""
        name="reporter[email]">
</label>
```

---

```
...
e = browser.find_element(by=BY.CSS, value="input[id='trEmail']")
e.clear()
e.send_keys("tu@hubzero.org")
...
```



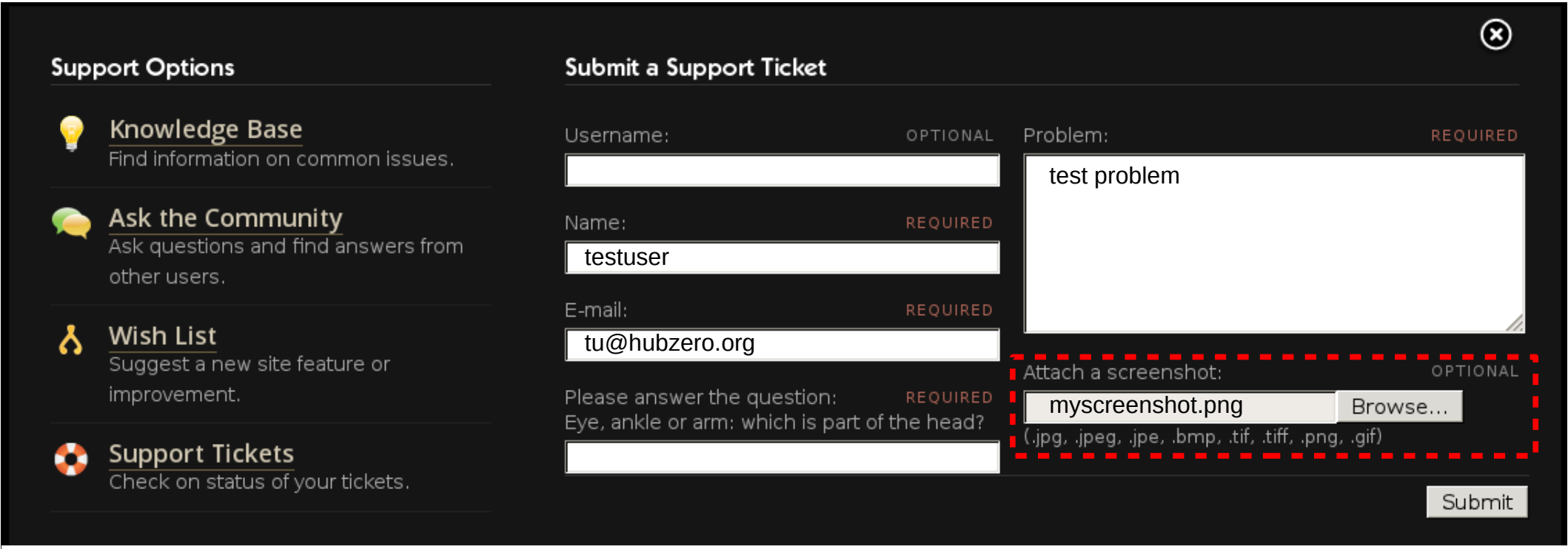

Writing to a Text Input:

1. Find the element
2. Clear the element
3. Send keys to the element

```
<label for="trProblem">
   Problem:
   <span class="required">required</span>
   <textarea id="trProblem"
      name="problem[long]"> </textarea>
</label>
```

---

```
...
e = browser.find_element(by=BY.CSS, value="textarea[id='trProblem']")
e.clear()
e.send_keys("test problem")
...
```



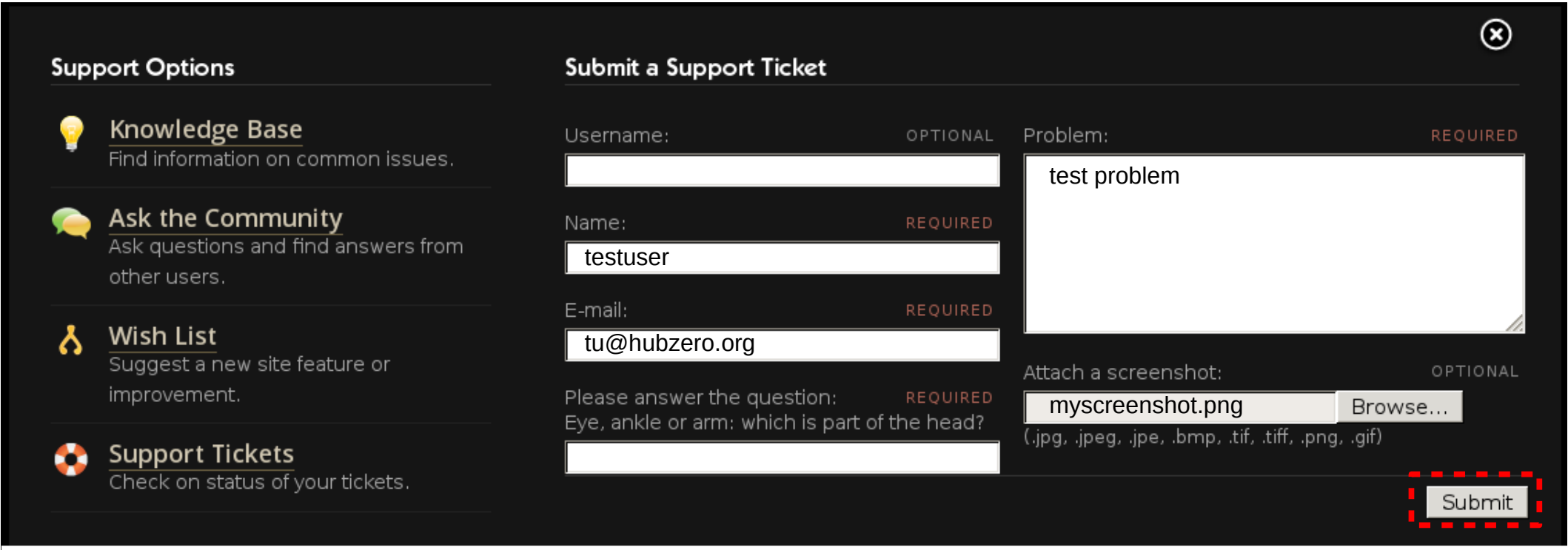

Uploading a file:

1. Find the element
2. Send file path to element

```
<label for="trUpload">
    Attach a screenshot:
    <span class="optional">optional</span>
    <input id="trUpload" type="file"
        name="upload">
</label>
```

---

```
...
e = browser.find_element(by=BY.CSS,value="input[id='trUpload']")
e.send_keys("$HOME/myscreenshot.png")
...
```




Pushing a button:

1. Find the element
2. Call the element's “click()” method

```
<div class=”submit”>
   <input id=”send-form” type=”submit”
      value=”Submit”>
</div>
```

---

```
...
e = browser.find_element(by=BY.CSS,value=”input[id='send-form']”)
e.click()
...
```

```
…
# File a Support Ticket

# Populate the Name field
e = browser.find_element(by=BY.CSS, value="input[id='trName']")
e.clear()
e.send_keys("testuser")

# Populate the Email field
e = browser.find_element(by=BY.CSS, value="input[id='trEmail']")
e.clear()
e.send_keys("tu@hubzero.org")

# Populate the Problem field
e = browser.find_element(by=BY.CSS, value="textarea[id='trProblem']")
e.clear()
e.send_keys("test problem")

# Populate the Upload field
e = browser.find_element(by=BY.CSS,value="input[id='trUpload']")
e.send_keys("$HOME/myscreenshot.png")

# Click the Submit button
e = browser.find_element(by=BY.CSS,value="input[id='send-form']")
e.click()

...
```

```
...
# File a Support Ticket

# Populate the Name field
e = browser.find_element(by=BY.CSS, value="input[id='trName']")
e.clear()
e.send_keys("testuser")

# Populate the Email field
e = browser.find_element(by=BY.CSS, value="input[id='trEmail']")
e.clear()
e.send_keys("tu@hubzero.org")

# Populate the Problem field
e = browser.find_element(by=BY.CSS, value="textarea[id='trProblem']")
e.clear()
e.send_keys("test problem")

# Populate the Upload field
e = browser.find_element(by=BY.CSS,value="input[id='trUpload']")
e.send_keys("$HOME/myscreenshot.png")

# Click the Submit button
e = browser.find_element(by=BY.CSS,value="input[id='send-form']")
e.click()

...
```

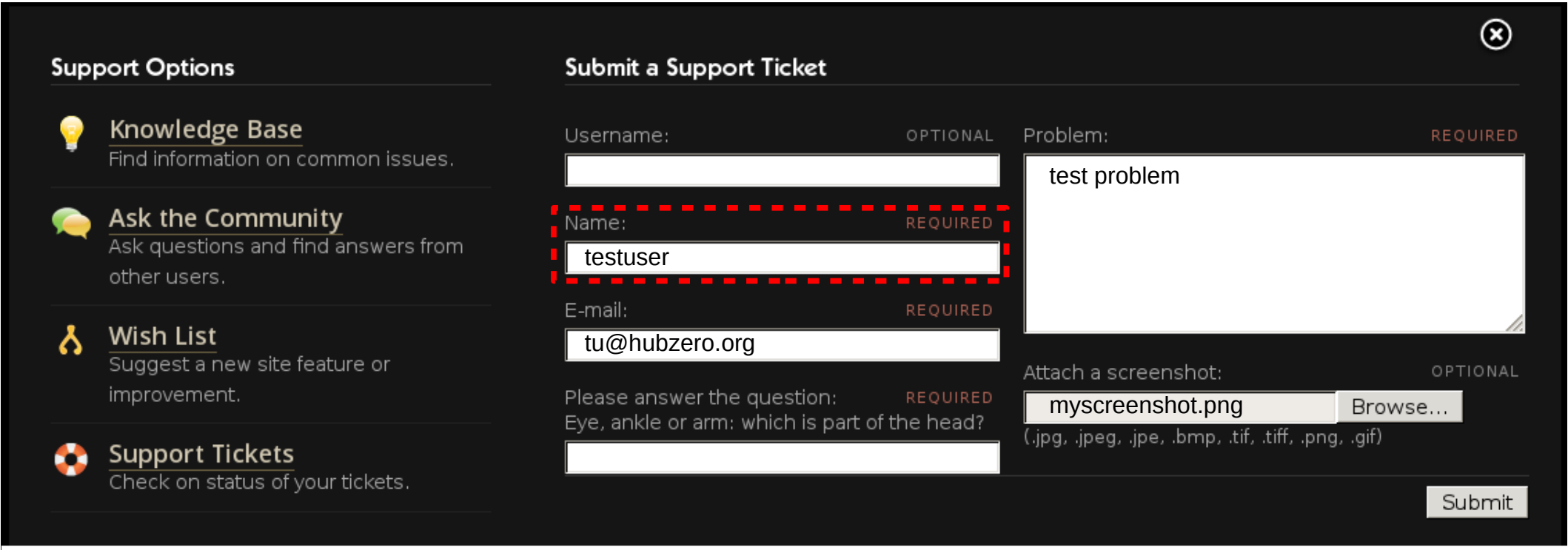
**Text Input**

**Text Input**

**Text Input**

**Text Input**

**Button**



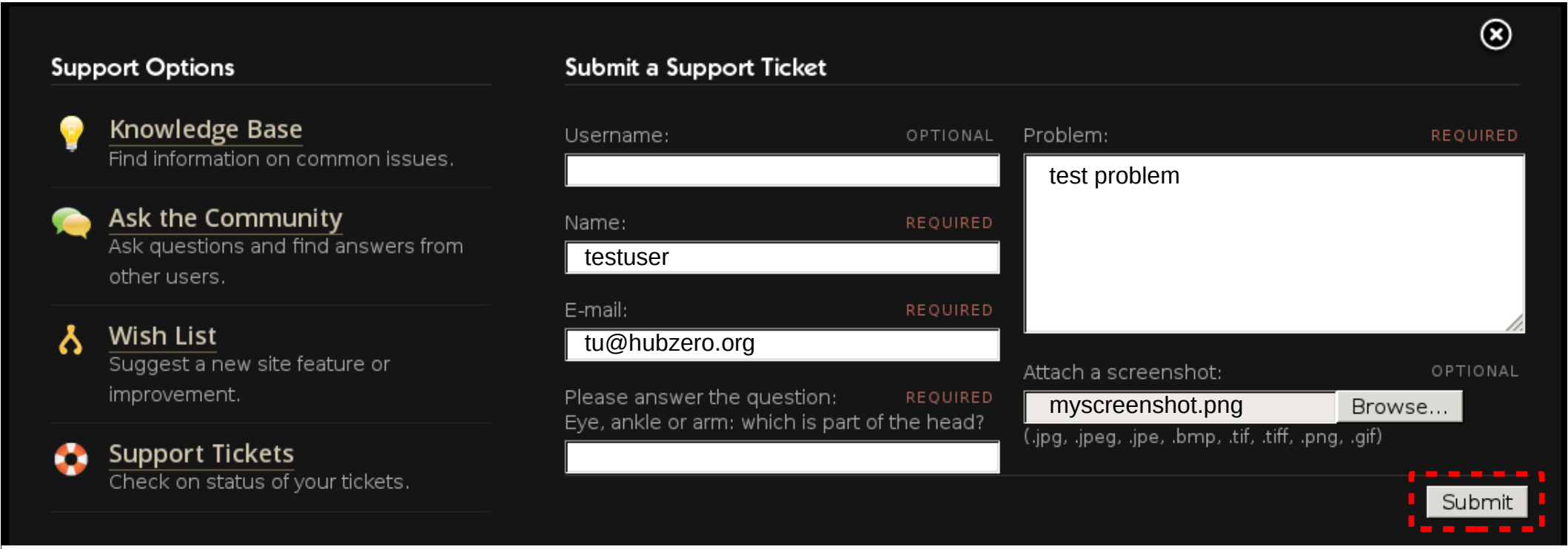

```
class Text(BasePageWidget):
    def __init__(self, locator):
        …
    def value(self, text):
        e = self.find_element(self.locator)
        e.clear()
        e.send_keys(text)
```

```
<label for="trName">
    Name:
    <span class="required">required</span>
    <input id="trName" type="text" value=""
        name="reporter[name]">
</label>
```

```
…
e = browser.find_element( by=BY.CSS,
        value="input[id='trName']")
e.clear()
e.send_keys("testuser")
…
```

→

```
…
name = Text("input[id='trName']")
name.value('testuser')
…
```




```
class Button(BasePageWidget):
    def __init__(self, locator):
        …
    def click(self):
        e = self.find_element(self.locator)
        e.click()
```

```
<div class="submit">
    <input id="send-form" type="submit"
        value="Submit">
</div>
```

---

```
…
e = browser.find_element( by=BY.CSS,
        value="input[id='send-form']")
e.click()
…
```

→

```
…
submit = Text("input[id='send-form']")
submit.click()
…
```

```
...
# File a Support Ticket

# Populate the Name field
name = Text("input[id='trName']")
name.value("testuser")

# Populate the Email field
email = Text("input[id='trEmail']")
email.value("tu@hubzero.org")

# Populate the Problem field
problem = Text("textarea[id='trProblem']")
problem.value("test problem")

# Populate the Upload field
upload = Text("input[id='trUpload']")
upload.value("$HOME/myscreenshot.png")

# Click the Submit button
submit = Button("input[id='send-form']")
submit.click()

...
```

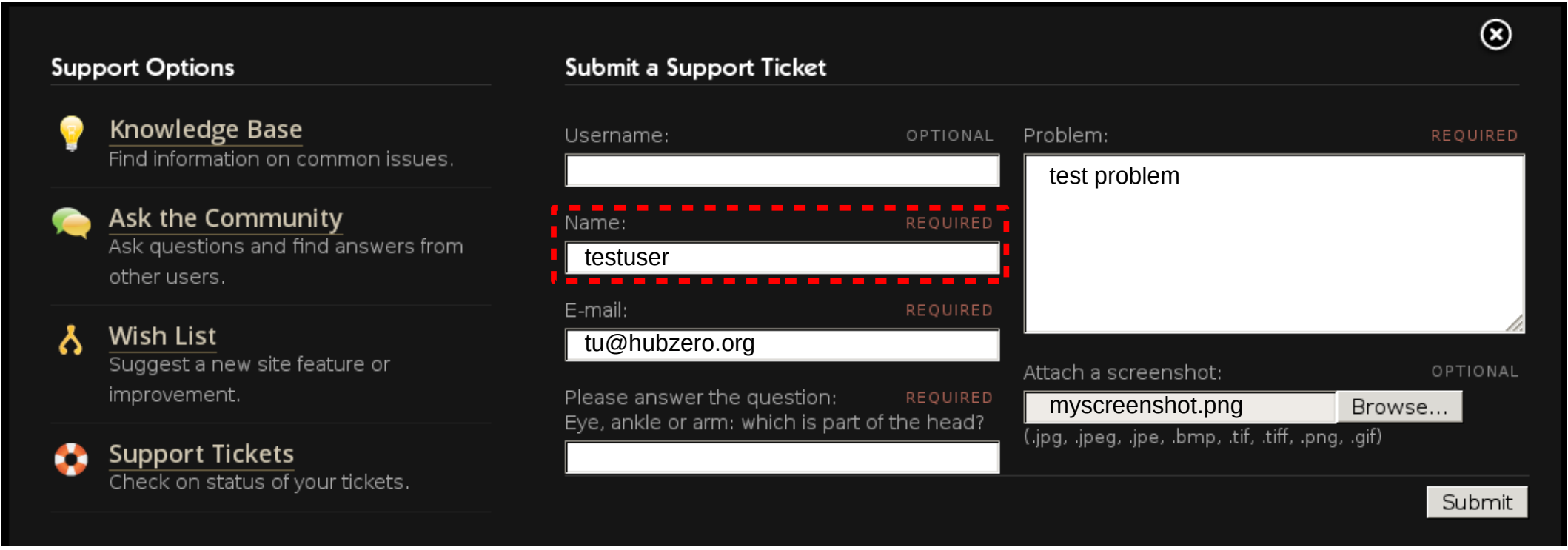
**Text Input**

**Text Input**

**Text Input**

**Text Input**

**Button**




```
class Text(BasePageWidget):
    def __init__(self, locator):
        …
    def value(self):
        e = self.find_element(self.locator)
        return e.get_attribute('value')
```

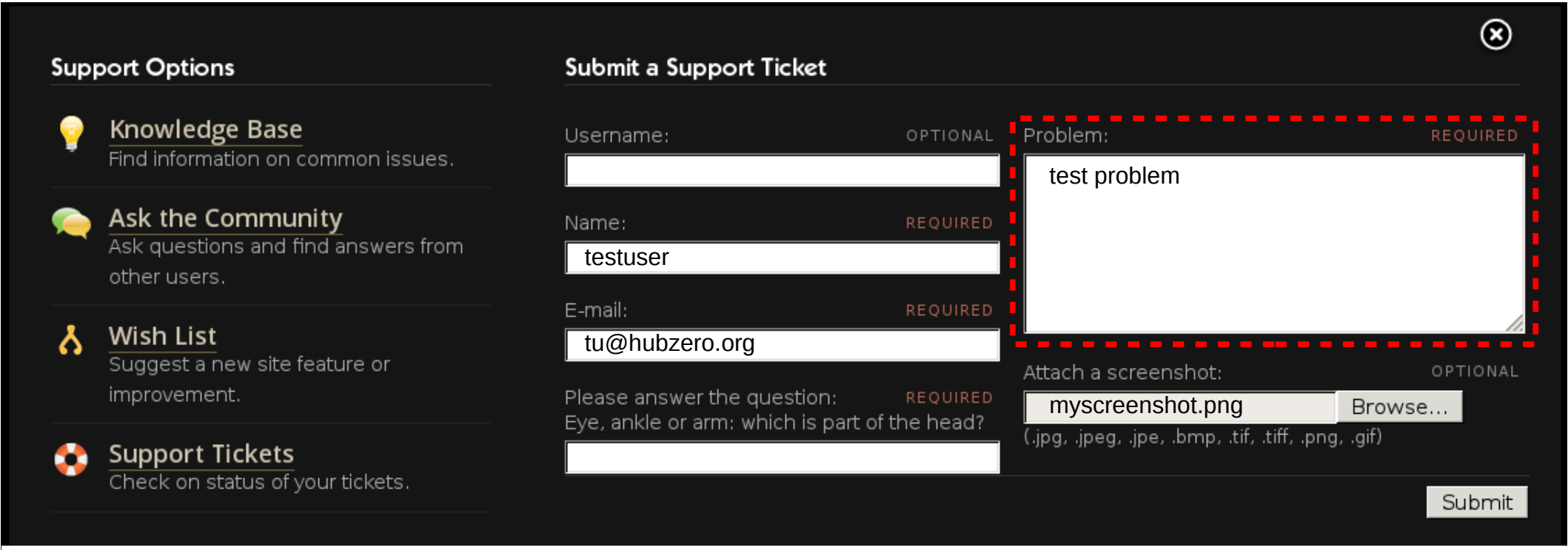
```
<label for="trName">
    Name:
    <span class="required">required</span>
    <input id="trName" type="text" value=""
        name="reporter[name]">
</label>
```

---

```
…
e = browser.find_element( by=BY.CSS,
        value="input[id='trName']")
e.get_attribute('value')

…
```

→

```
…
name = Text("input[id='trName']")
x = name.value()
…
```




```
class Text(BasePageWidget):
    def __init__(self, locator):
        …
    def value(self):
        e = self.find_element(self.locator)
        return e.get_attribute('value')
```

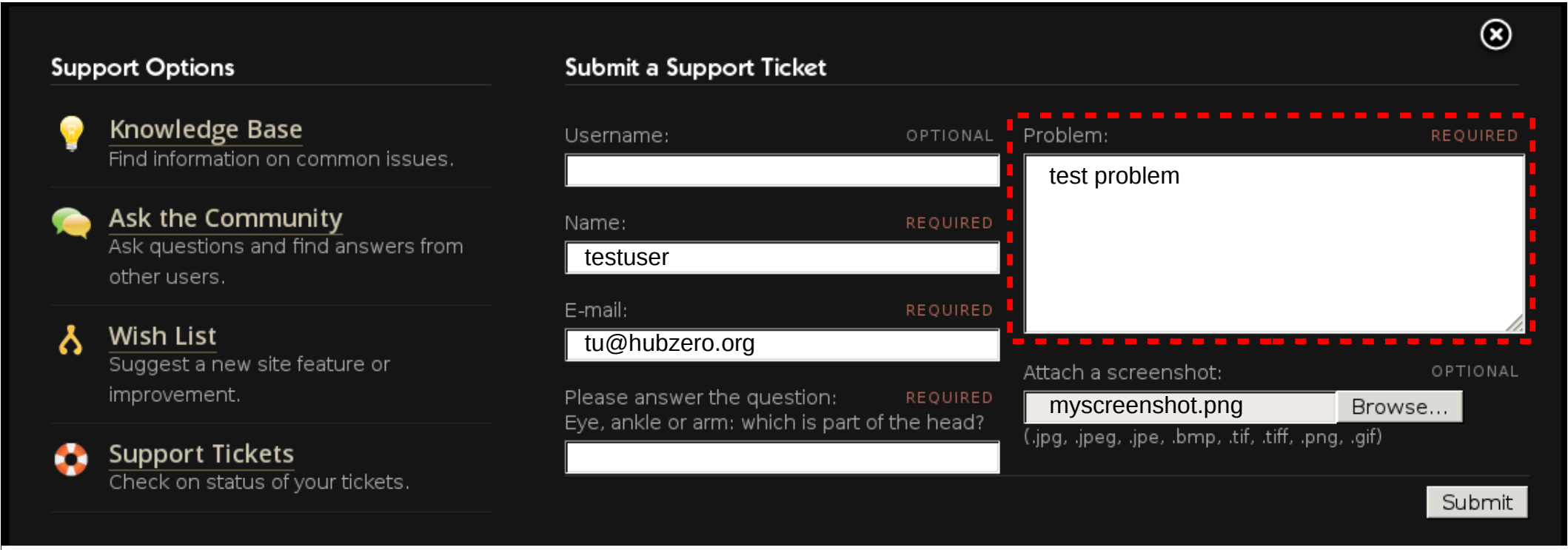
```
<label for="trProblem">
    Problem:
    <span class="required">required</span>
    <textarea id="trProblem"
        name="problem[long]"> </textarea>
</label>
```

---

```
…
e = browser.find_element( by=BY.CSS,
        value="input[id='trProblem']")
e.text

…
```

→

```
…
problem = Text("input[id='trProblem']")
x = problem.value()
…
```




```
class TextArea(BasePageWidget):
    def __init__(self, locator):
        …
    def value(self):
        e = self.find_element(self.locator)
        return e.text
```

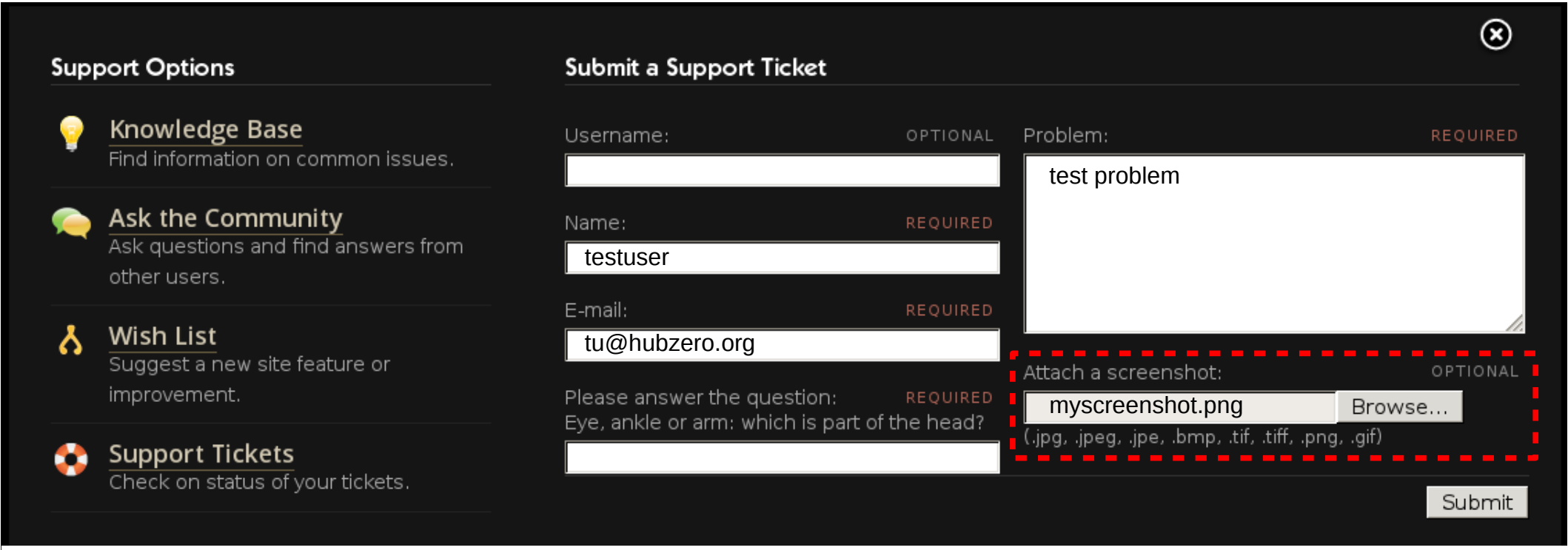
```
<label for="trProblem">
    Problem:
    <span class="required">required</span>
    <textarea id="trProblem"
        name="problem[long]"> </textarea>
</label>
```

---

```
…
e = browser.find_element( by=BY.CSS,
        value="input[id='trProblem']")
e.text

…
```

→

```
…
problem = Text("input[id='trProblem']")
x = problem.value()
…
```




```
class Upload(BasePageWidget):
    def __init__(self, text_loc,browse_loc):
        …
    def value(self,value):
        e = self.find_element(self.browse_loc)
        e.get_attribute('value')
```

```
<label for="trUpload">
    Attach a screenshot:
    <span class="optional">optional</span>
    <input id="trUpload" type="file"
        name="upload">
</label>
```

---

```
…
e = browser.find_element( by=BY.CSS,
        value="input[id='trUpload']")
e.send_keys('$HOME/myscreenshot.png')

…
```

→

```
…
problem = Upload("input[id='trProblem']")
problem.value('$HOME/myscreenshot.png')
…
```

```
...
# File a Support Ticket

# Populate the Name field
name = Text("input[id='trName']")
name.value("testuser")

# Populate the Email field
email = Text("input[id='trEmail']")
email.value("tu@hubzero.org")

# Populate the Problem field
problem = TextArea("textarea[id='trProblem']")
problem.value("test problem")

# Populate the Upload field
upload = Upload("input[id='trUpload']")
upload.value("$HOME/myscreenshot.png")

# Click the Submit button
submit = Button("input[id='send-form']")
submit.click()

...
```

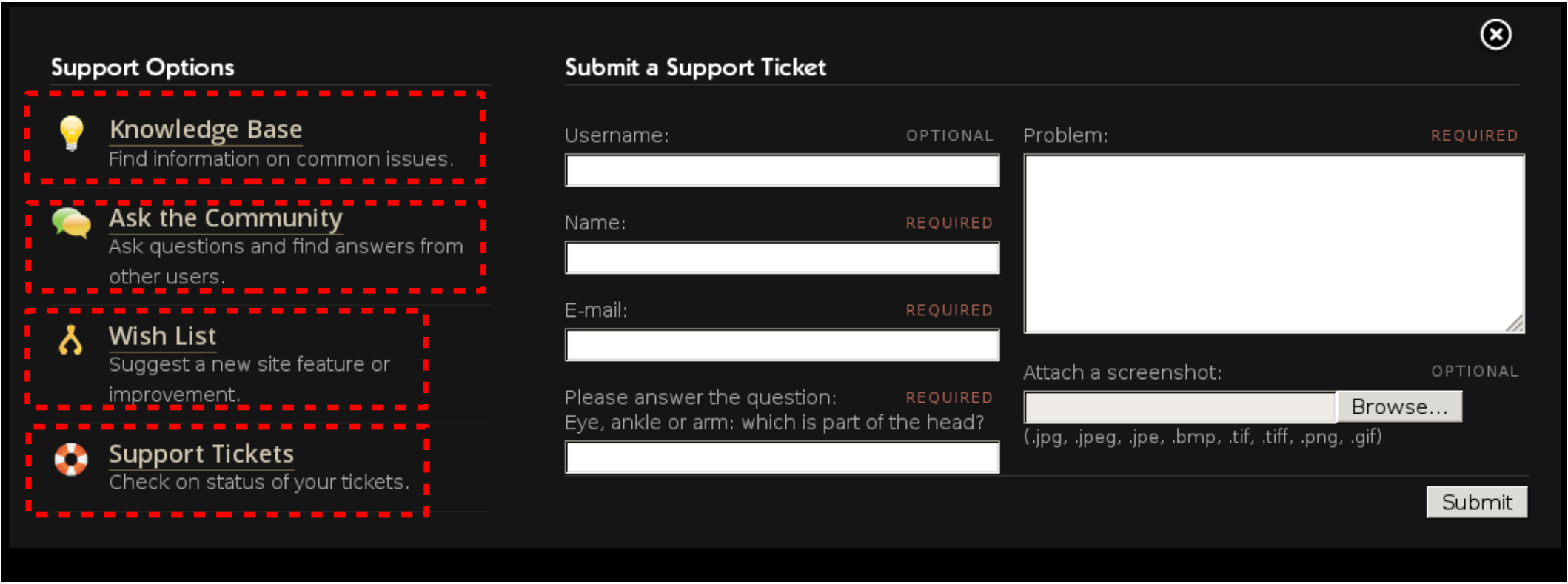
**Text Input**

**Text Input**

**TextArea**

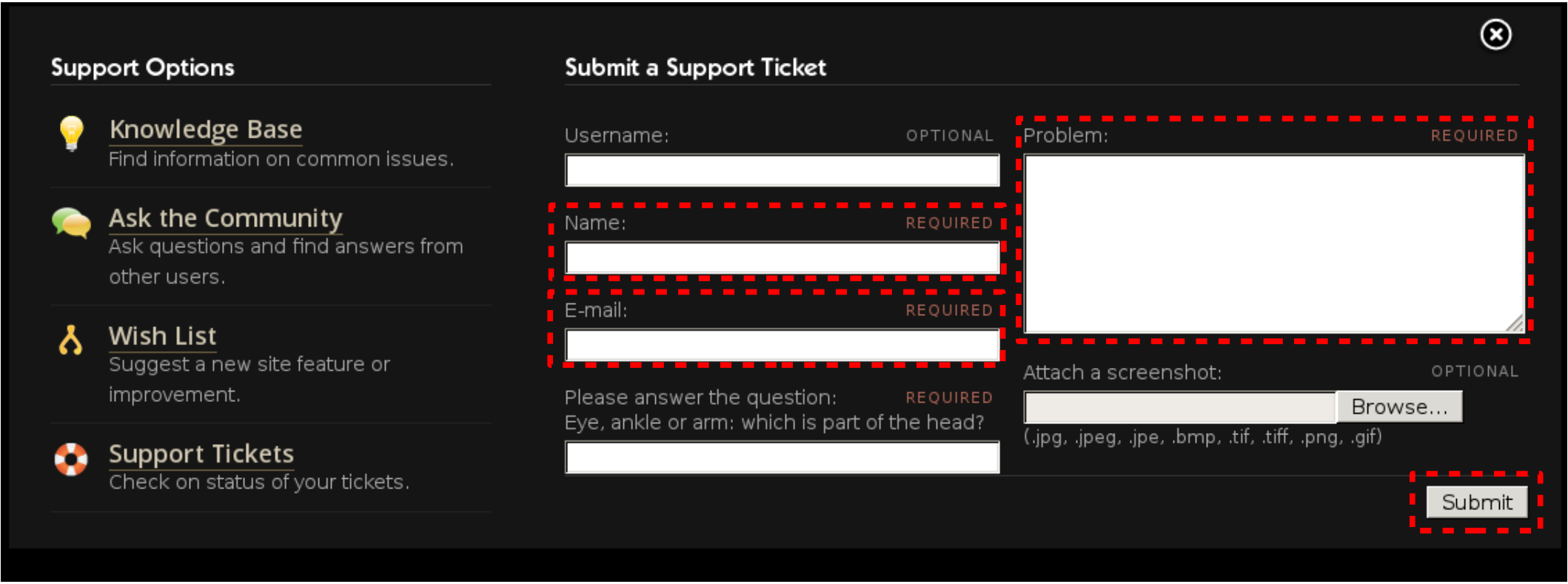
**Upload**

**Button**

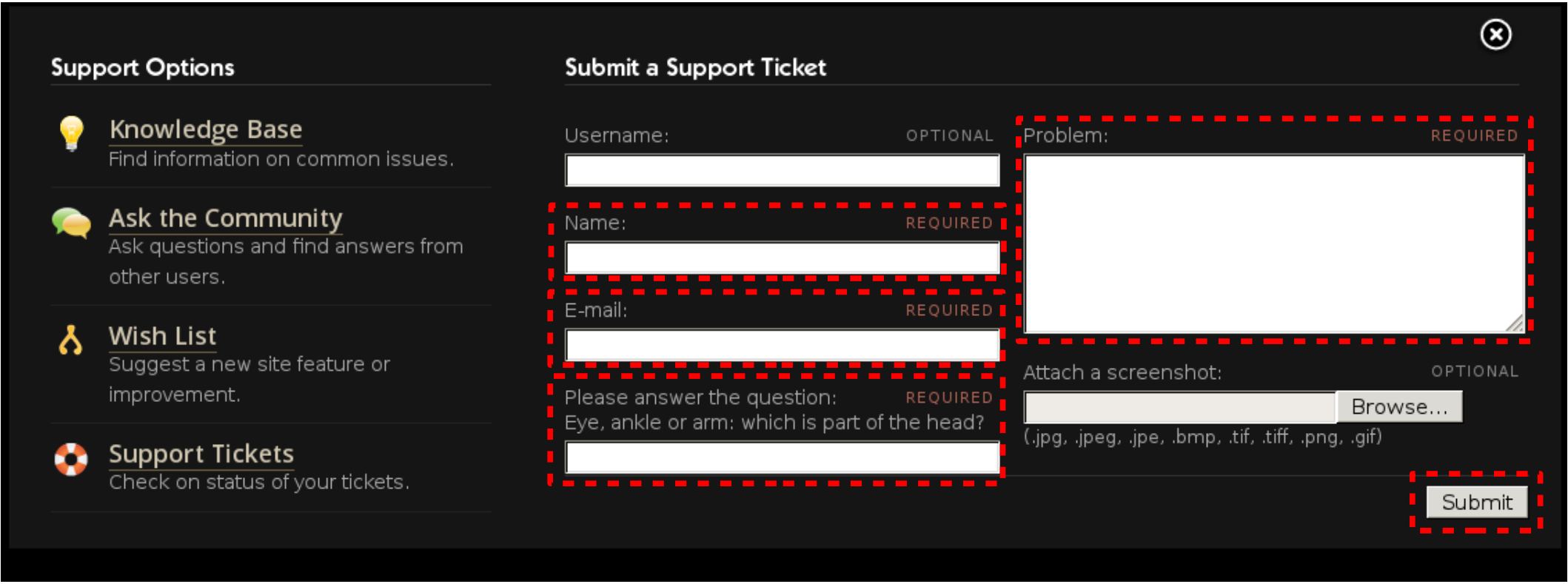



# Go to Support Webpages

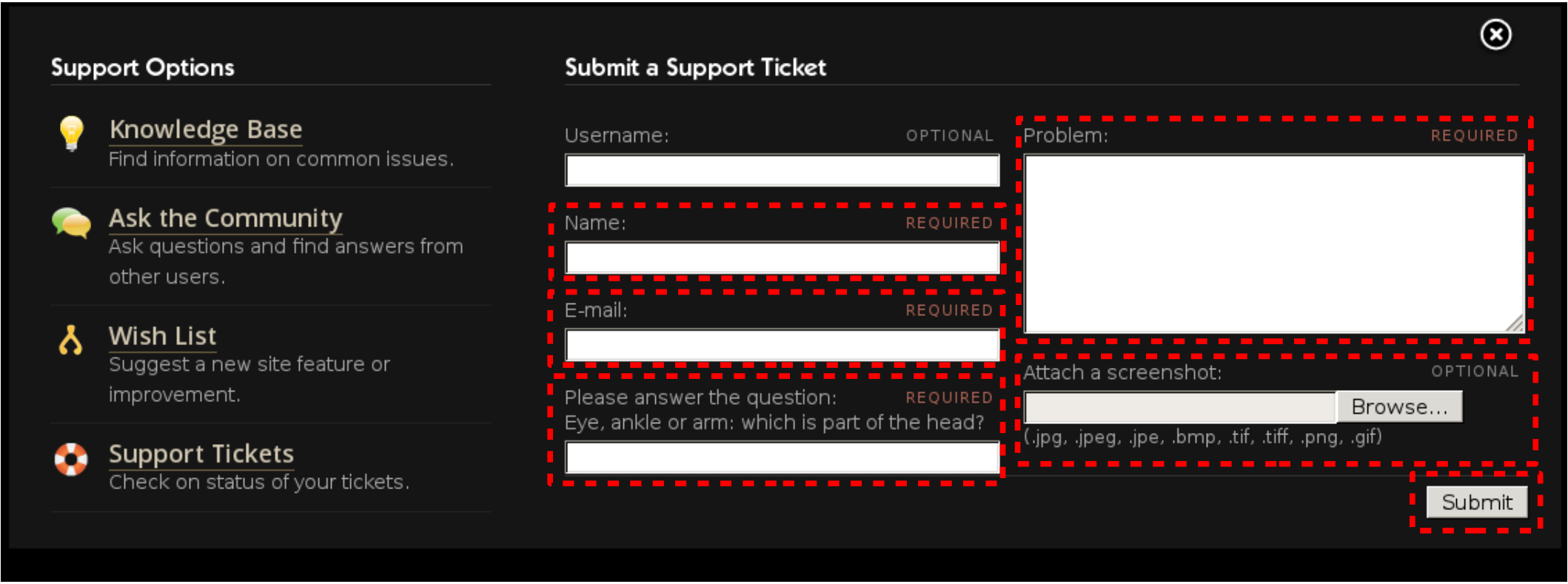
* Knowledge Base
* Questions/Answers
* Wish List
* Support Tickets




# Submit New Ticket
# * Name, Email, Problem




# Submit New Ticket
# * Name, Email, Problem, Captcha



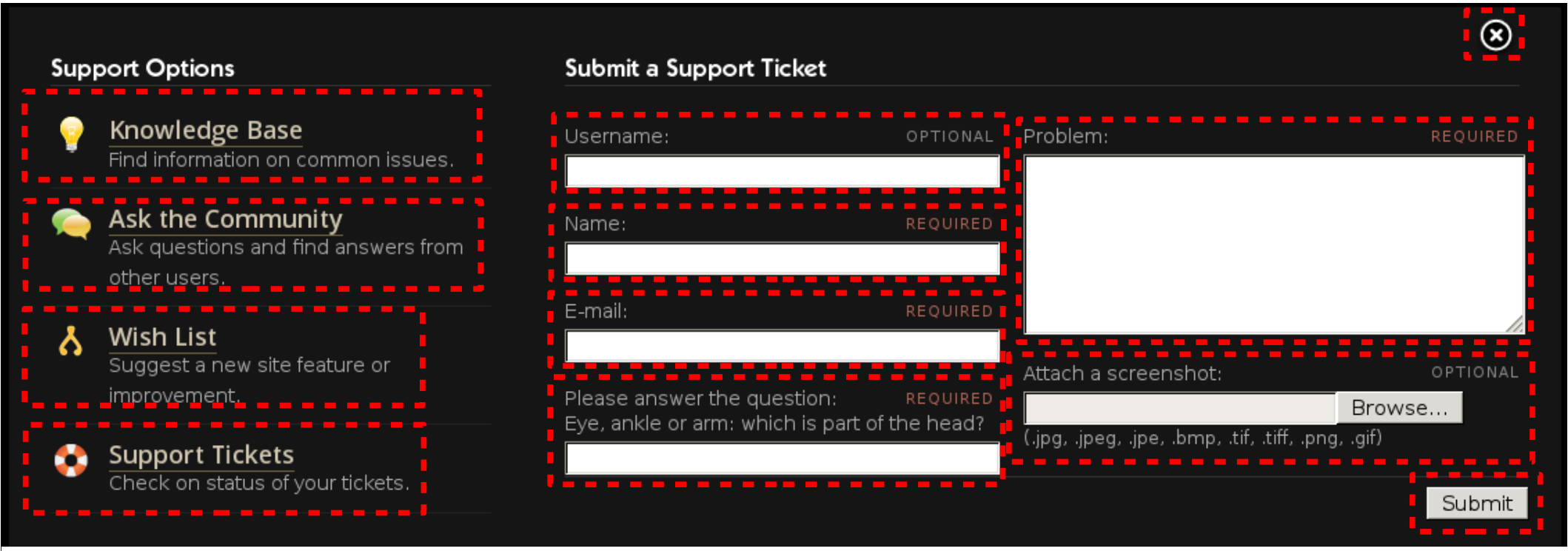

# Submit New Ticket
# * Name, Email, Problem, Captcha, Upload

```
...
# File a Support Ticket

# Populate the Name field
e = browser.find_element(by=BY.CSS, value="input[id='trName']")
e.clear()
e.send_keys("testuser")

# Populate the Email field
e = browser.find_element(by=BY.CSS, value="input[id='trEmail']")
e.clear()
e.send_keys("tu@hubzero.org")

# Populate the Problem field
e = browser.find_element(by=BY.CSS, value="textarea[id='trProblem']")
e.clear()
e.send_keys("test problem")

# Populate the Upload field
e = browser.find_element(by=BY.CSS,value="input[id='trUpload']")
e.send_keys("$HOME/myscreenshot.png")

# Click the Submit button
e = browser.find_element(by=BY.CSS,value="input[id='send-form']")
e.click()

...
```

```
...
# File a Support Ticket

# Populate the Name field
name = Text("input[id='trName']")
name.value("testuser")


# Populate the Email field
email = Text("input[id='trEmail']")
email.value("tu@hubzero.org")


# Populate the Problem field
problem = TextArea("textarea[id='trProblem']")
problem.value("test problem")


# Populate the Upload field
upload = Upload("input[id='trUpload']")
upload.value("$HOME/myscreenshot.png")

# Click the Submit button
submit = Button("input[id='send-form']")
submit.click()

...
```

```
...
# File a Support Ticket

# Populate the Support Ticket
po = TroubleReportForm("#troublereport")

data = {
    'username'    :    'hctest1',
    'name'        :    'hctest1 hctest1',
    'email'       :    'hctest1@hubzero.org',
    'problem'     :    'test problem text',
    'captcha'     :    True,
    'upload'      :    '/home/hctest1/myscreenshot.png',
}

po.submit_ticket(data)

...
```

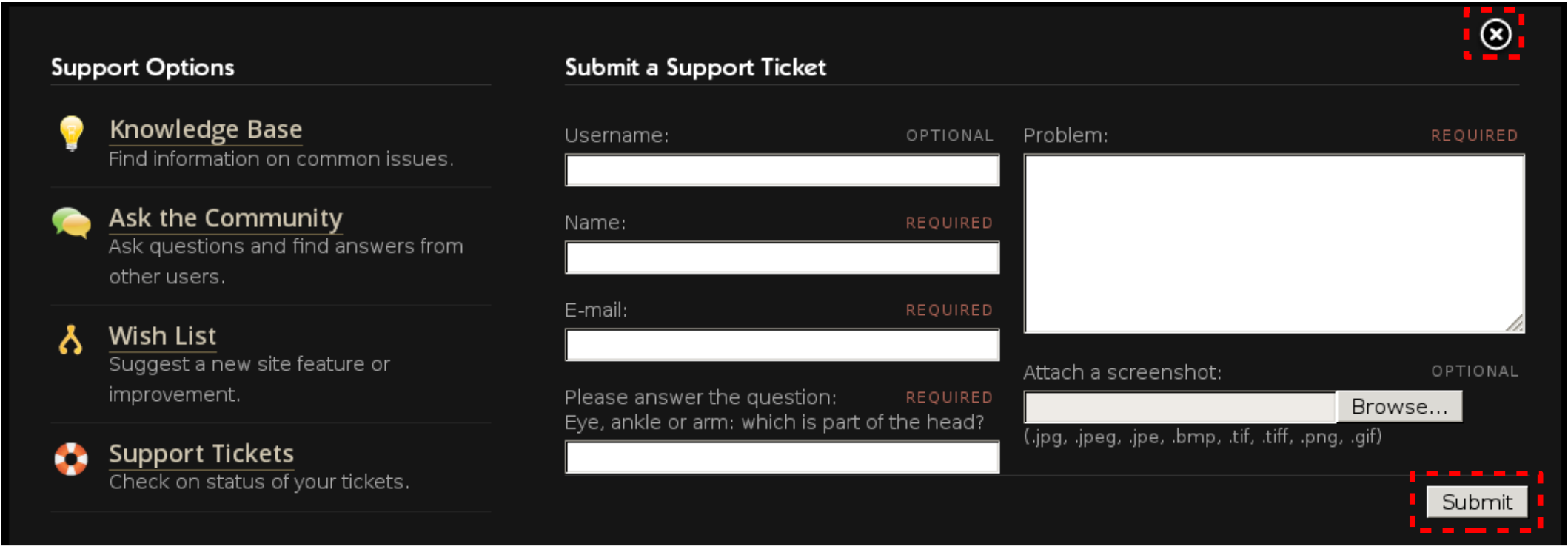
```
...
# File a Support Ticket

# Populate the Support Ticket
po = TroubleReportForm("#troublereport")

data = {
    'name'          :    'hctest1 hctest1',
    'email'         :    'hctest1@hubzero.org',
    'problem'       :    'test problem text',
}

po.submit_ticket(data)

...
```




```
class TroubleReportForm(BasePageWidget):
    def __init__(self):
        ...
        self.username = Text(self,'username')
        self.captcha  = Captcha(self,'captcha')
        self.problem  = Text(self,'problem')
        self.upload   = Upload(self,'upload')
        self.submit   = Button(self,'submit')
        …
    def submit_ticket(self,data):
        …
    def open(self):
        …
    def close(self):
        …
```

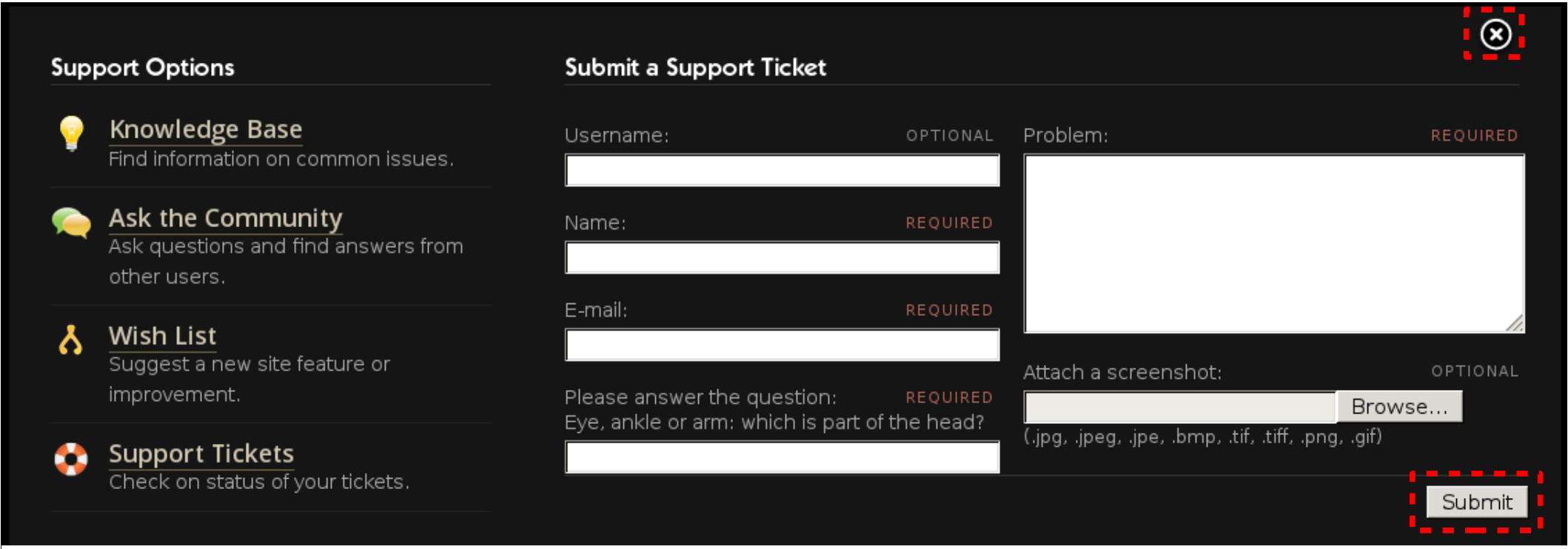
**Page Description**

**Page Services**



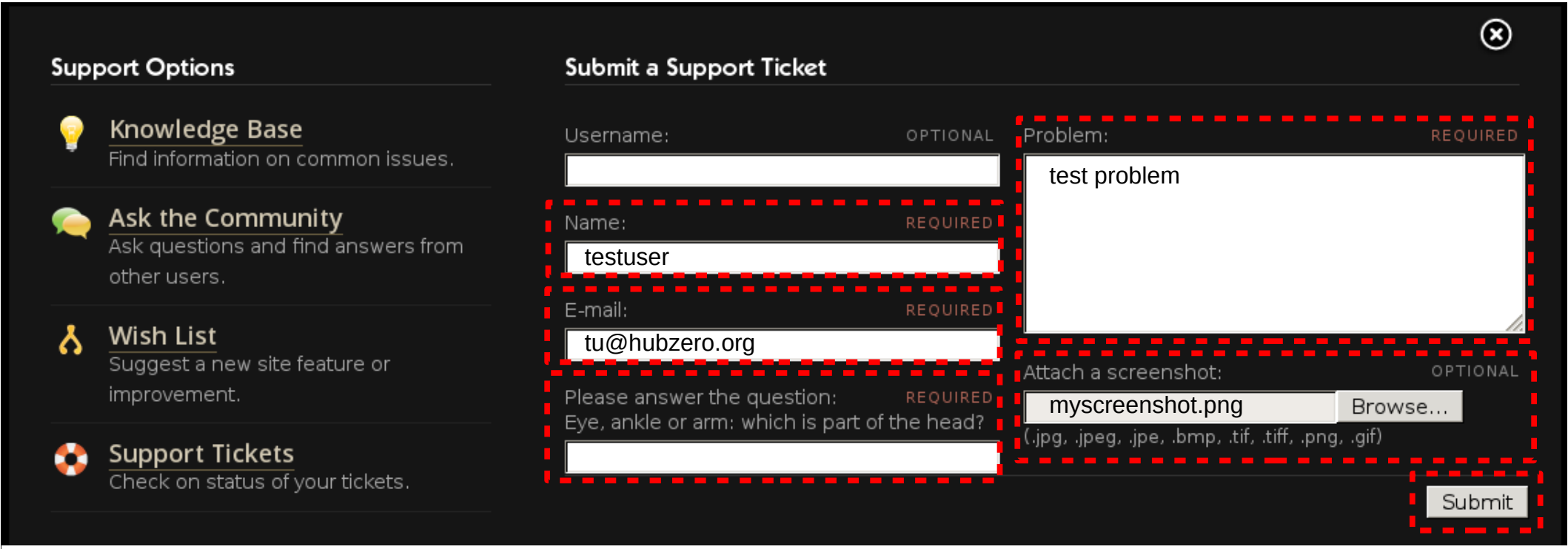

```
class TroubleReportForm(BasePageWidget):
    def __init__(self):
        ...
        self.username = Text(self,'username')
        self.captcha  = Captcha(self,'captcha')
        self.problem  = Text(self,'problem')
        self.upload   = Upload(self,'upload')
        self.submit   = Button(self,'submit')
        …
    def submit_ticket(self,data):
        …
    def open(self):
        …
    def close(self):
        …
```

```
# open and close the need help pane

po = TroubleReportForm()
po.open()
po.close()
```



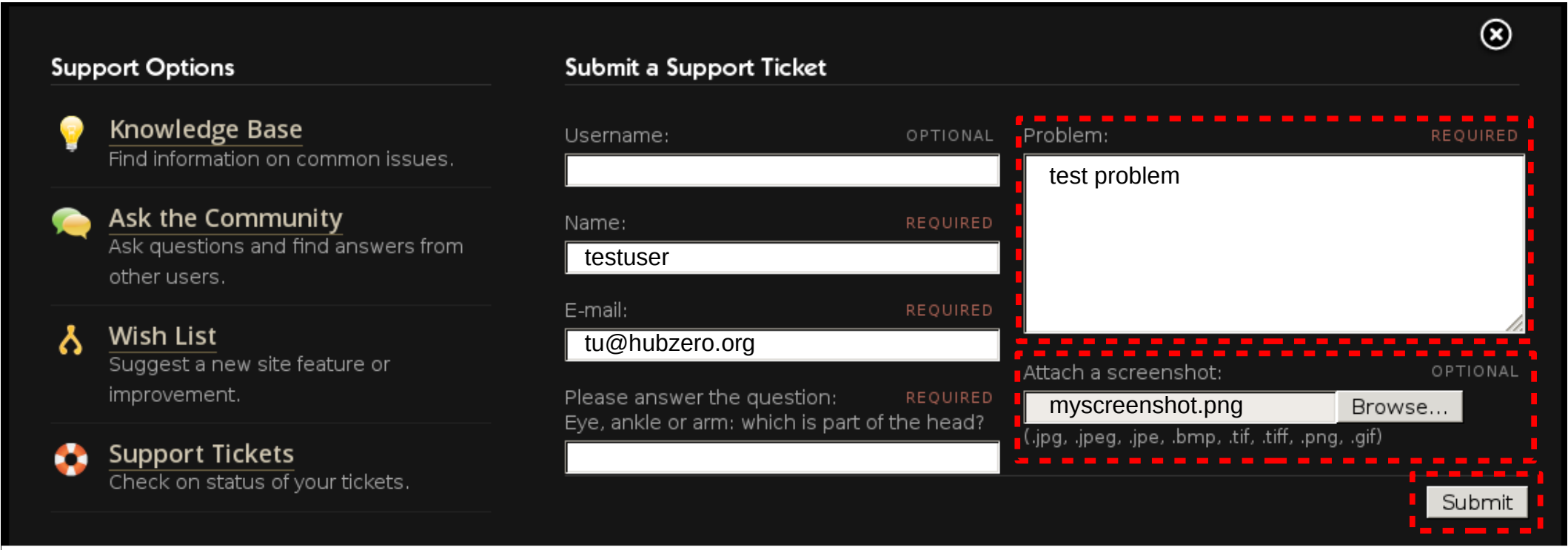

```
class TroubleReportForm(BasePageWidget):
    def __init__(self):
        ...
        self.username = Text(self,'username')
        self.captcha  = Captcha(self,'captcha')
        self.problem  = Text(self,'problem')
        self.upload   = Upload(self,'upload')
        self.submit   = Button(self,'submit')
        …
    def submit_ticket(self,data):
        …
    def open(self):
        …
    def close(self):
        …
```

```
# submit a blank ticket

po = TroubleReportForm()
po.open()
data = {}
po.submit_ticket(data)
```




```
class TroubleReportForm(BasePageWidget):
    def __init__(self):
        ...
        self.username = Text(self,'username')
        self.captcha  = Captcha(self,'captcha')
        self.problem  = Text(self,'problem')
        self.upload   = Upload(self,'upload')
        self.submit   = Button(self,'submit')
        …
    def submit_ticket(self,data):
        …
    def open(self):
        …
    def close(self):
        …
```

```
# fill in the name, email, problem,
# upload a file, then submit the ticket

po = TroubleReportForm()
po.open()
data = {
    'name'    : 'testuser',
    'email'   : 'tu@hubzero.org',
    'problem' : 'test problem',
    'upload'  : 'myscreenshot.png',
}
po.submit_ticket(data)
```




```
class TroubleReportForm(BasePageWidget):
    def __init__(self):
        ...
        self.username = Text(self,'username')
        self.captcha  = Captcha(self,'captcha')
        self.problem  = Text(self,'problem')
        self.upload   = Upload(self,'upload')
        self.submit   = Button(self,'submit')
        …
    def submit_ticket(self,data):
        …
    def open(self):
        …
    def close(self):
        …
```

```
# login to the website,
# fill in problem, upload file,
# then submit ticket

po = Login()
po.login_as('testuser','tupass')

po = TroubleReportForm()
po.open()
data = {
    'problem' : 'test problem',
    'upload' : 'myscreenshot.png',
}
po.submit_ticket(data)
```

# Login

## Log in with your Hub account.

Username:

Forgot your Username?

Password:

Forgot your Password?

Remember Me

Login

### No account?

Register. It's free!

### Is this really free?

Yes! Use of HUBzero resources and tools is *free* for registered users. There are no hidden costs or fees.

### Why is registration required for parts of HUBzero?

Our sponsors ask us who uses HUBzero and what they use it for. Registration helps us answer these questions. Usage statistics also focus our attention on improvements, making the HUBzero experience better for *you*.

# Login

## Log in with your Hub account.

Username:

Forgot your Username?

Password:

Forgot your Password?

Remember Me

Login

### No account?

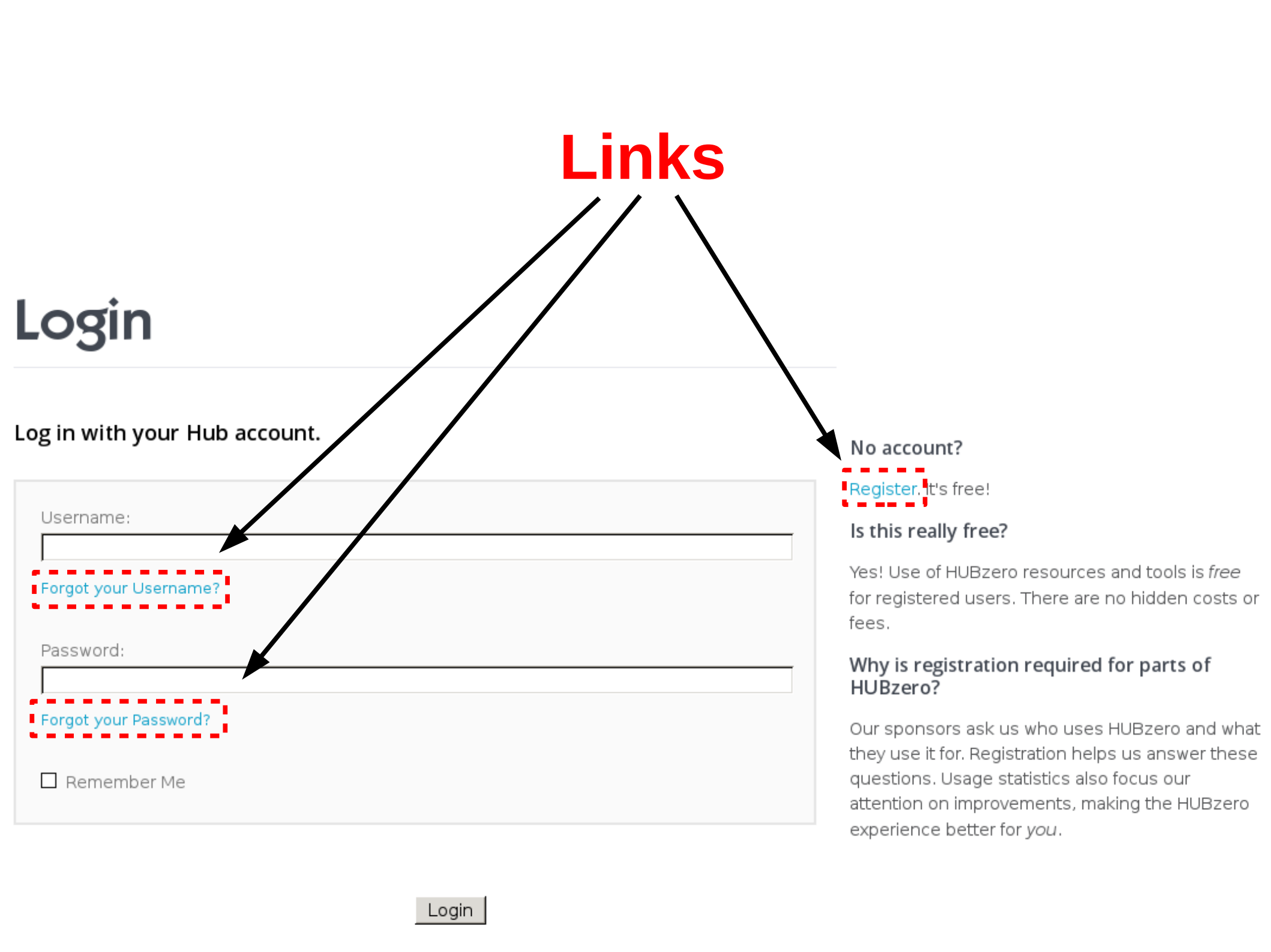
Register. It's free!

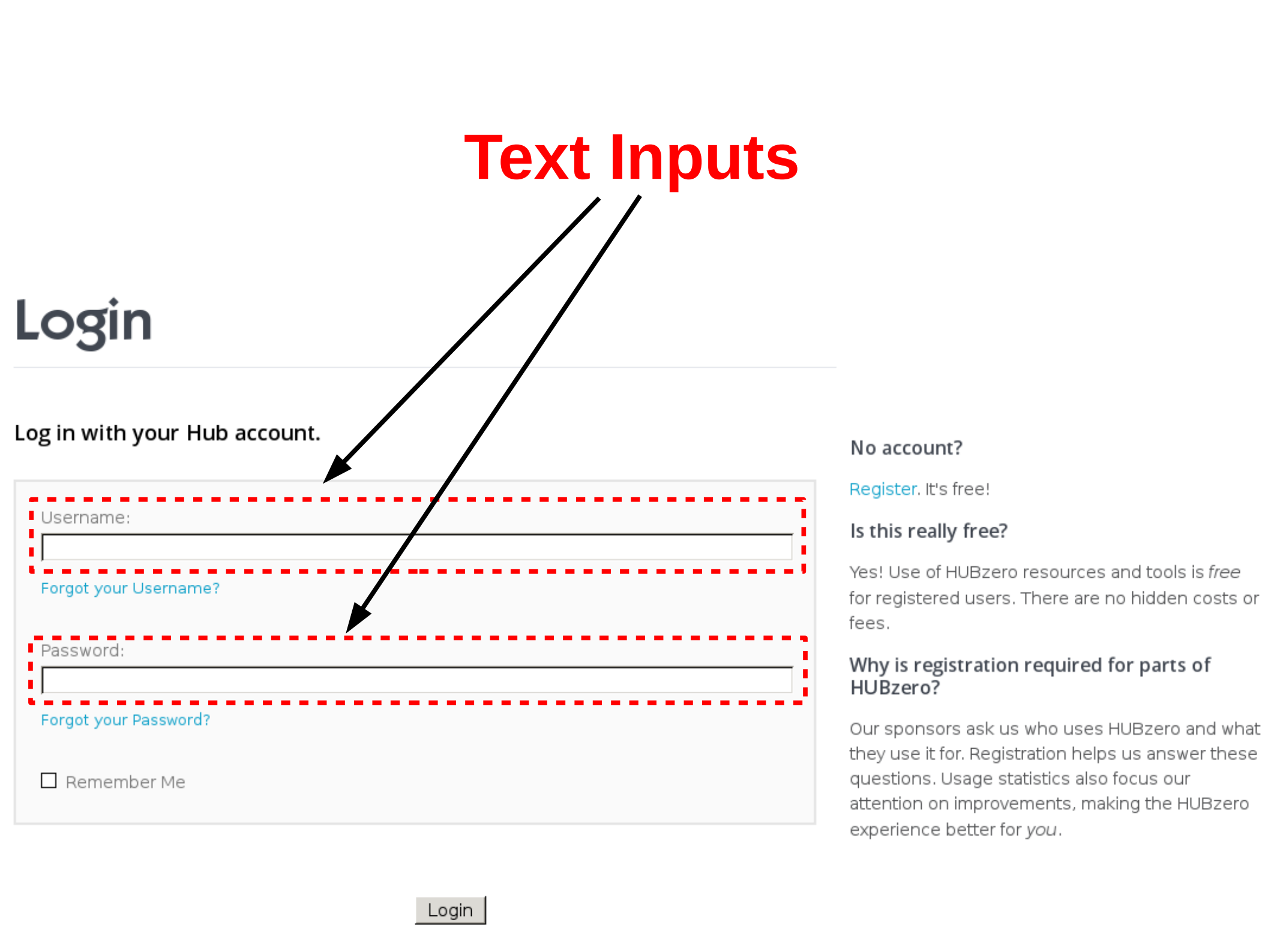
### Is this really free?

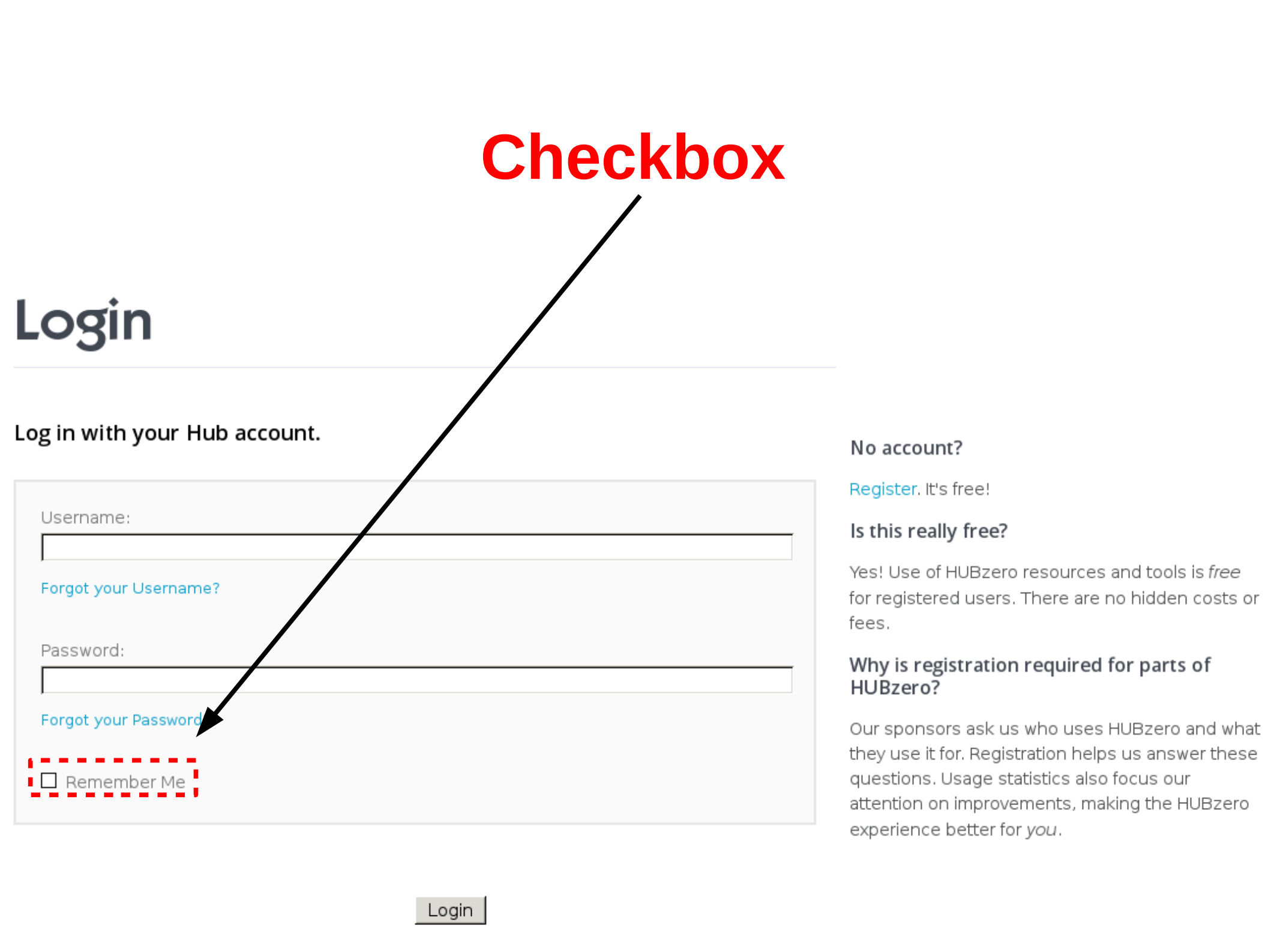
Yes! Use of HUBzero resources and tools is *free* for registered users. There are no hidden costs or fees.

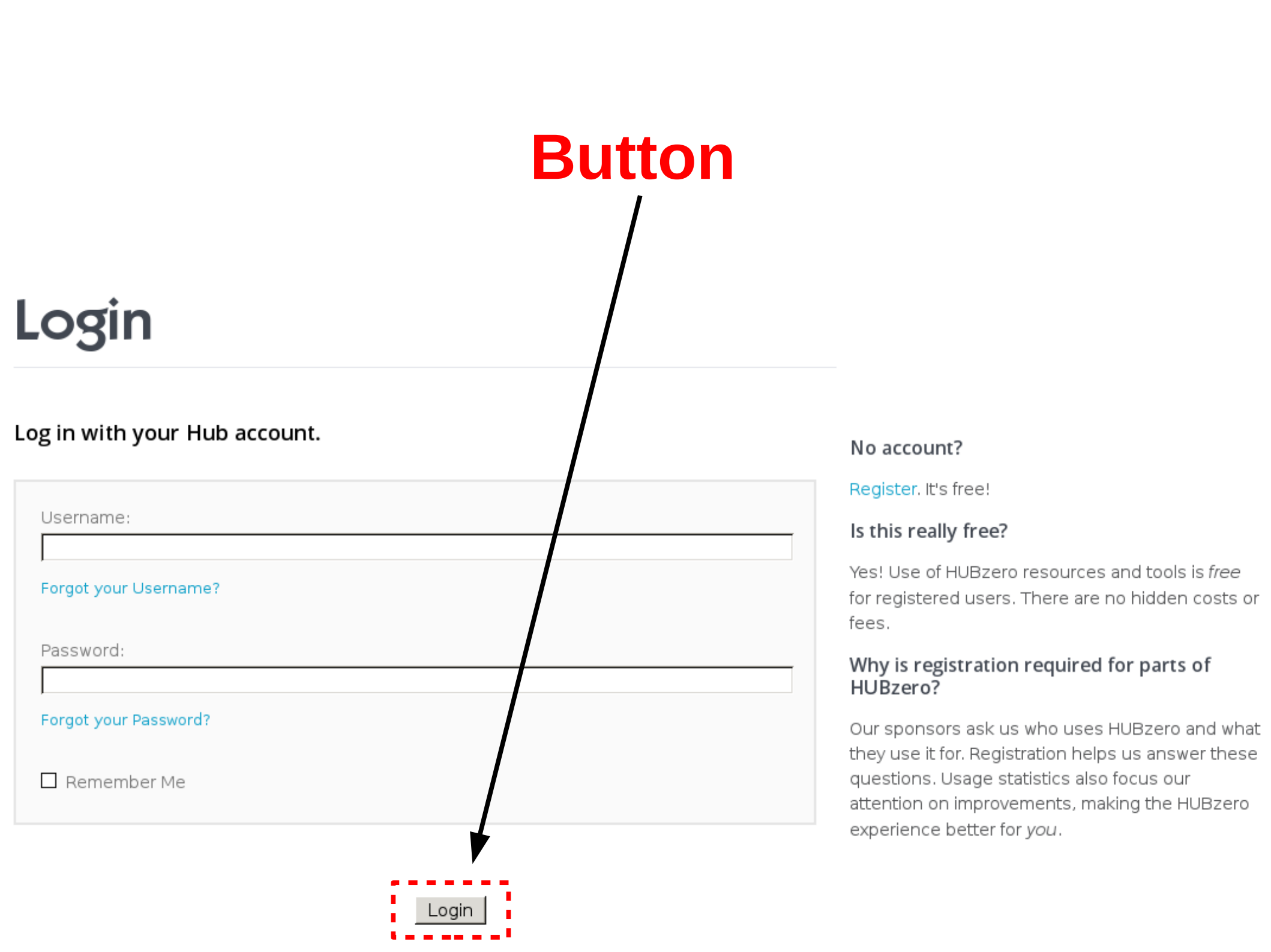
### Why is registration required for parts of HUBzero?

Our sponsors ask us who uses HUBzero and what they use it for. Registration helps us answer these questions. Usage statistics also focus our attention on improvements, making the HUBzero experience better for *you*.


Links
Login
Log in with your Hub account.
Username:
Forgot your Username?
Password:
Forgot your Password?
Remember Me
Login
No account?
Register. It's free!
Is this really free?
Yes! Use of HUBzero resources and tools is free for registered users. There are no hidden costs or fees.
Why is registration required for parts of HUBzero?
Our sponsors ask us who uses HUBzero and what they use it for. Registration helps us answer these questions. Usage statistics also focus our attention on improvements, making the HUBzero experience better for you.


Text Inputs
Login
Log in with your Hub account.
Username:
Forgot your Username?
Password:
Forgot your Password?
Remember Me
Login
No account?
Register. It's free!
Is this really free?
Yes! Use of HUBzero resources and tools is free for registered users. There are no hidden costs or fees.
Why is registration required for parts of HUBzero?
Our sponsors ask us who uses HUBzero and what they use it for. Registration helps us answer these questions. Usage statistics also focus our attention on improvements, making the HUBzero experience better for you.


Checkbox
Login
Log in with your Hub account.
Username:
Forgot your Username?
Password:
Forgot your Password?
Remember Me
No account?
Register. It's free!
Is this really free?
Yes! Use of HUBzero resources and tools is free for registered users. There are no hidden costs or fees.
Why is registration required for parts of HUBzero?
Our sponsors ask us who uses HUBzero and what they use it for. Registration helps us answer these questions. Usage statistics also focus our attention on improvements, making the HUBzero experience better for you.
Login


Button
Login
Log in with your Hub account.
Username:
Forgot your Username?
Password:
Forgot your Password?
Remember Me
Login
No account?
Register. It's free!
Is this really free?
Yes! Use of HUBzero resources and tools is free for registered users. There are no hidden costs or fees.
Why is registration required for parts of HUBzero?
Our sponsors ask us who uses HUBzero and what they use it for. Registration helps us answer these questions. Usage statistics also focus our attention on improvements, making the HUBzero experience better for you.

```
class Login(BasePageWidget):
    def __init__(self, locator):
        ...
        self.username = Text(self,'username')
        self.password  = Text(self,'password')
        self.remember = Checkbox(self,'remember')
        self.remind      = Link(self,'remind')
        self.reset         = Link(self,'reset')
        self.register     = Link(self,'register')
        self.submit      = Button(self,'submit')

    def login_as(self,username,password):
        self.username.value = username
        self.password.value = password
        self.submit.click()
```

**Page Description**

**Page Services**

```
class Login(BasePageWidget):
    def __init__(self, locator):
        ...
        self.username = Text(self,'username')
        self.password  = Text(self,'password')
        self.remember = Checkbox(self,'remember')
        self.remind      = Link(self,'remind')
        self.reset         = Link(self,'reset')
        self.register     = Link(self,'register')
        self.submit      = Button(self,'submit')

    def login_as(self,username,password):
        self.username.value = username
        self.password.value = password
        self.submit.click()
```

```
po = Login('#hubForm')
po.login_as('testuser','testpassword')
```

---

```
po = Login('#hubForm')
po.remember.value = True
po.login_as('testuser','testpassword')
```

---

```
po = Login('#hubForm')
po.remind.click()
```

---

```
po = Login('#hubForm')
po.username.value = 'testuser'
po.submit.click()
```

---