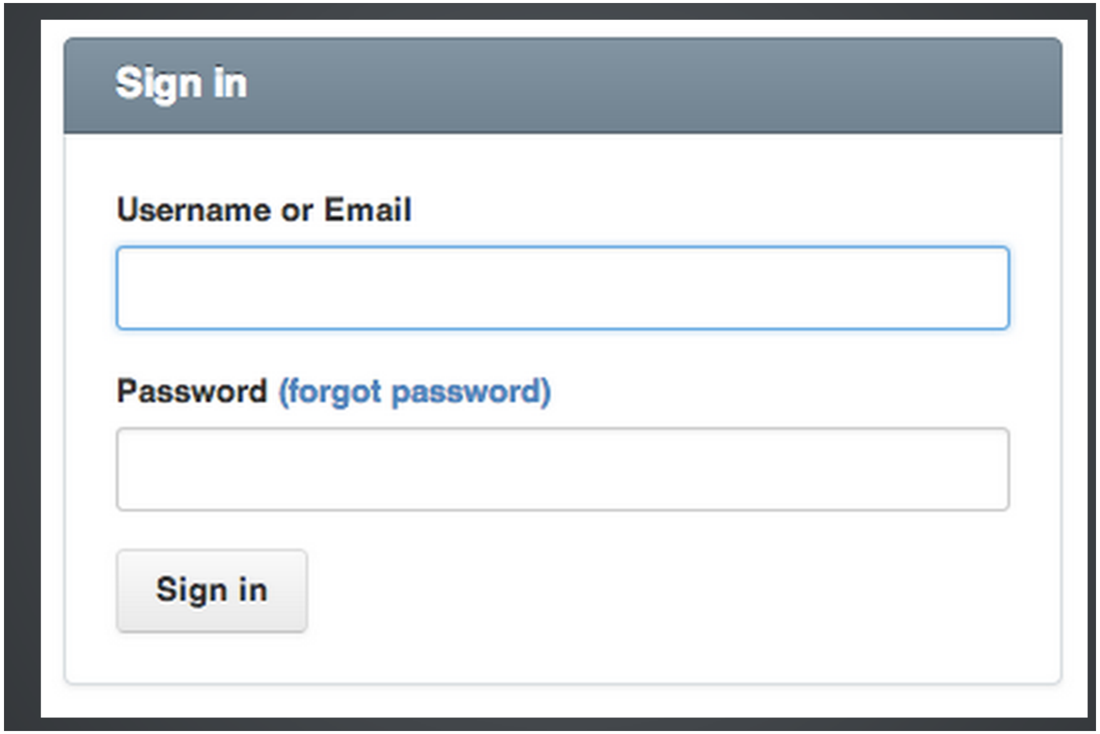
BACK END WEB DEVELOPMENT

CLASS 15- AUTHENTICATION USING DEVISE

AGENDA

- REVIEW SYLLABUS
- PRACTICE READING DOCUMENTATION
- DEVISE INTEGRATION
- FINAL PROJECT LAB


Sign in
Username or Email
Password (forgot password)
Sign in

AUTHENTICATION

Who you are ?

AUTHORIZATION

WHAT YOU HAVE ACCESS TO

HOW AUTHENTICATION DOES IT WORK?

- Validating user identity using a username/password
- Tracking user's identity using a *session*
- Keeping user's identity between visits with a *cookie*.
- Protecting the site with security using *encryption*
  - Using secure cryptographic methods!!!
- Providing additional features based on identity

STORING PASSWORDS

- Passwords should *never* be stored in "plain text", (human readable). If your site is hacked, someone can get everyone's password.
- Always use a "hashing encryption" on the passwords before storing them in the database.
    - This is done for us automatically by auth gems!
- We'll use a gem called 'devise' for today's class.

MANAGING USERS: SESSIONS

- *Sessions* are a special browser feature that hold per-site data for as long as the browser is open.
- *Sessions* go away when they close the browser.
- When the user is authenticated, we store their user_id in the *session* .
- The user_id allows us to look up the user's info, like their shopping cart, name, blog posts, etc.

MANAGING USERS: COOKIES

- Used for storing small bits of data about users.
- Cookies allow web servers to provide a temporary unique identifier to a browser.
  - Browser storage is not secure.
  - Sensitive data (credit cards, passwords) should never be stored in a cookie.

SESSIONS & COOKIES

- Sessions are used to store user identifiers.
  - Expire when a browser is closed
  - Or whenever the browser wants to
- Cookies can store a small amount of any data.
  - Expire when you tell them to
  - Or when a user manually deletes their cookies

AUTHENTICATION GEMS

Creating authentication from scratch is a complex and challenging process, especially for security.

Rails developers have created gems for authentication. Here are the most popular:

- **Devise**
- Pundit

DEVISE

By far, the most popular authentication gem is devise.

https://github.com/plataformatec/devise

- Straight-forward to implement, integrate and customize.
- Handles complex security, so you don't have to.
- Provides controller filters and view helpers
- Uses bcrypt by default .

DEVISE INSTALLATION

- This was very fast and simple!
- It's also the *very* basics of devise, but that's quite a bit of useful code!
- View helpers:
  - user_signed_in?
  - current_user
  - user_session

```
gem 'devise' # Add the gem

rails g devise:install
rails g devise User

rake db:migrate

# Add sign in/out helpers to application.html.erb
```

DEVISE INTEGRATION

TRY TO DETERMINE THE STEPS
DOCUMENTATION PRACTICE!

FINAL PROJECT

LAB TIME