Some best practices:

https://codingkilledthecat.wordpress.com/2012/09/04/some-best-practices-for-web-app-authentication/

Stack overflow guide:

https://stackoverflow.com/questions/549/the-definitive-guide-to-form-based-website-authentication

PHP login page:

http://form.guide/php-form/php-login-form.html

The big question is PHP or javascript?

From: https://www.upwork.com/hiring/development/php-vs-javascript/

Even with the advent of Node.js, the short answer to this question is still: JavaScript for the client-side, and PHP for the server-side. But, if you're trying to choose between back-end technologies or are building an entire solution stack, it helps to go into a little more detail. You should consider PHP if your project involves…

- CMS's like WordPress, Drupal, or Joomla etc.
- Servers like MySQL, SQL, MariaDB, Oracle, Sybase, and Postgresql etc.
- Solution Stacks like LAMP (Linux, Apache, MySQL, PHP)

You should consider JavaScript if your project involves…

- Dynamic single page applications (SPAs)
- Front-end technologies like jQuery, AngularJS, Backbone.js, Ember.js, ReactJS etc.
- Server-side technologies like Node.js, MongoDB, Express.js, etc.
- Solution Stacks like MEAN (MongoDB, Express.js, AngularJS)

a guide to some of the aspects of using nodejs vs roll your own Auth
https://stormpath.com/blog/choosing-nodejs-authentication-strategy