

# Users and Authentication

Reindert-Jan Ekker  
[nl.linkedin.com/in/rjekker/](https://nl.linkedin.com/in/rjekker/)  
@rjekker



**pluralsight**   
hardcore dev and IT training

# In This Module



## Authentication with [Flask-Login](#)

Configuration

Adapting the User class for login



## Securing views

Using the [@login\\_required](#) decorator

## Adding a login view, form and template

And don't forget logout

## Adding signup for new users

# Setting up Flask-Login

`pip install flask-login`

Import and configure

```
from flask_login import LoginManager  
login_manager = LoginManager()  
login_manager.session_protection = "strong"  
login_manager.login_view = "login"  
login_manager.init_app(app)
```

# Setting Up the User Class

- **Declare a `user_loader`:**

- Tells Flask-login how to retrieve a user by id
- id is stored in the HTTP session

```
@login_manager.user_loader
def load_user(userid):
    return User.query.get(int(userid))
```

- **Inherit from `UserMixin`:**

- default implementations for `is_authenticated`, `get_id`, etc

```
class User(db.Model, UserMixin):
```

# Using Flask-Login

Mark views with `@login_required`

The `current_user` variable holds the currently logged in user

Log a user in with `login_user(user)`

Optional argument: `remember (bool)`

Log them out with `logout_user()`

In a view:

```
{% if current_user.is_authenticated() %}
```



# Resources



<http://flask-login.readthedocs.org/en/latest/>  
(<http://goo.gl/QufHY1>)

# Summary

- **Flask-Login**
  - `user_loader`
  - `UserMixin`
  - `login_required`
  - `current_user`
  - `login_user()`
  - `logout_user()`
- **Password hashing**
- **Signup**