

SATURN 2017

13th Software Engineering Institute Architecture
Technology User Network Conference

Django & Twitter bootstrap in the workplace: build 'em fast and furious

Eliezer Kanal





Copyright 2017 Carnegie Mellon University. All Rights Reserved.

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

[DISTRIBUTION STATEMENT A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

Internal use:* Permission to reproduce this material and to prepare derivative works from this material for internal use is granted, provided the copyright and "No Warranty" statements are included with all reproductions and derivative works.

External use:* This material may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other external and/or commercial use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

* These restrictions do not apply to U.S. government entities.

Carnegie Mellon® and CERT® are registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

DM17-0178

Outline

* Quick note on Microservices
and API Endpoints

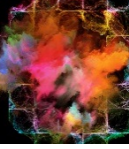
- Framework Selection
- Set up for success
- Django
- Django extras
- Bootstrap



- Too many decisions
- Too many options
- Too many variables



Don't...



- x Roll your own
- x Choose “no framework”
- x Just choose your personal favorite
- ❖ Technical debt
- ❖ Framework/base development vs. Actual development
- ❖ Security bugs!!!
- ❖ You are on a *team*
- ❖ Each project is unique

Web Framework Benefits

Backend Framework will make you:

- **Faster**
- **Capable**
- **Secure**



Frontend Framework (layout) will make you:

- **Look Professional**

Frontend Framework (interactivity) will make you:

- **Interactive**



Framework Selection

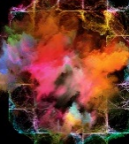
Critical:

- What will the organization support?
- What does your team already know?
- What best suits your needs?
 - Size of community, support

Less critical:

- Ability to scale?
- Framework speed?





Set up for Success: Minimal training

- Django:
 1. Do the tutorial
 2. Google EVERYTHING
- Bootstrap:
 1. Explain the grid
 2. Scan through [docs](#)
Resize the window while browsing!

Time investment: 1 day

Webmail Client

Folder

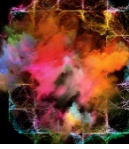
Mail

mail list

app's

Mail
Todo
Calendar

mail view



Django – projects, apps, URLs

Terminal:

```
$ django-admin startproject mail_clone
$ django-admin startapp email
$ django-admin startapp calendar
$ django-admin startapp todo
```

- Namespaced database
- Test webserver
- `manage.py` file
- Site settings file
- Skeleton folder/file structure

settings.py:

```
INSTALLED_APPS = (
    'email',
    'calendar',
    'todo',
)
```

urls.py:

```
urlpatterns = [
    url(r'^email/', include('email.urls')),
    url(r'^calendar/', include('calendar.urls')),
    url(r'^todo/', include('todo.urls')),
]
```

Django – Database, ORM

models.py:

```
from django.db import models

class Email(models.Model):
    to = models.CharField()
    from = models.CharField()
    date = models.DateField()
    subject = models.CharField()
    body = models.TextField()
```

Terminal:

```
$ python manage.py makemigrations
$ python manage.py migrate
```

◦ Field types

- AutoField
- BigAutoField
- BigIntegerField
- BinaryField
- BooleanField
- CharField
- CommaSeparatedIntegerField
- DateField
- DateTimeField
- DecimalField
- DurationField
- EmailField
- FileField
 - FileField and FieldFile
- FilePathField
- FloatField

– Create database changes file

– Update database structure

Django – Database, ORM



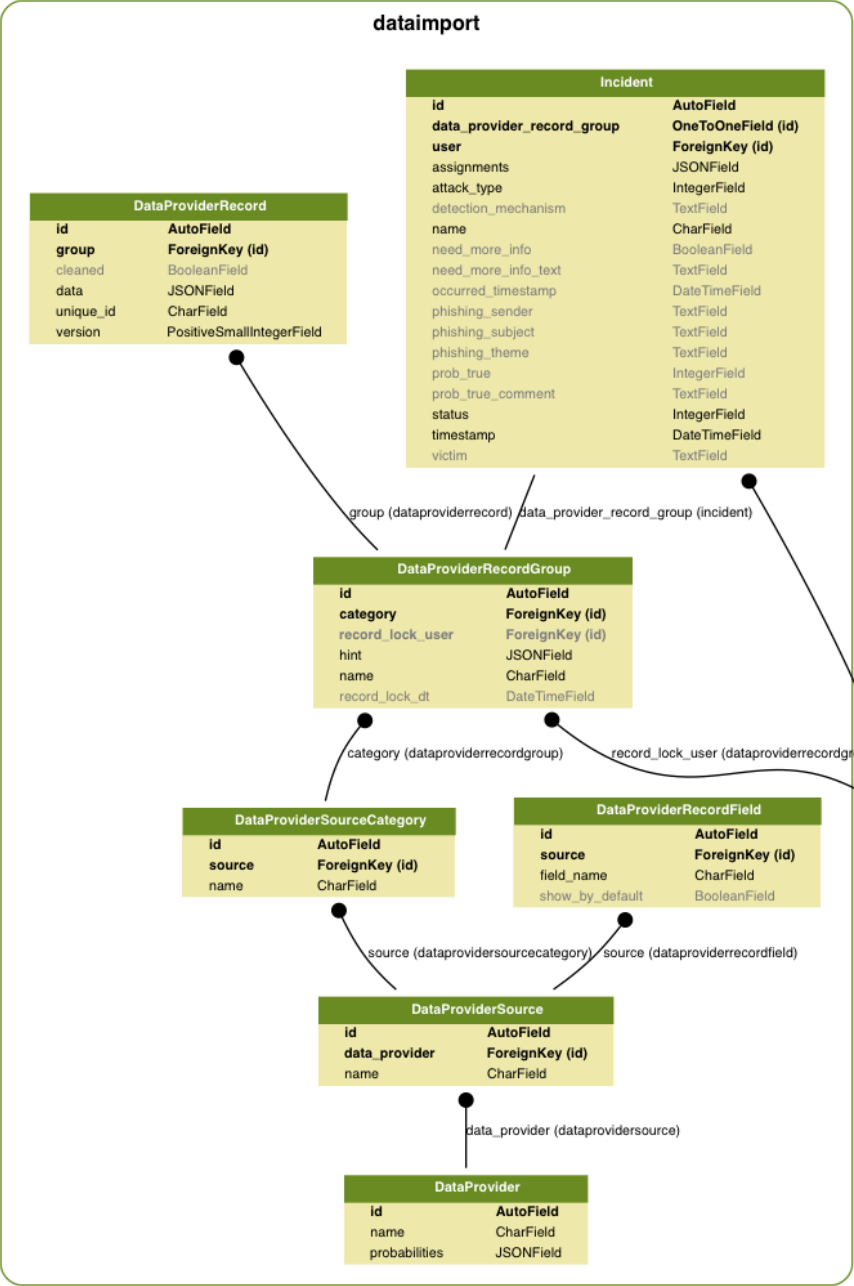
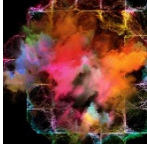
models.py:

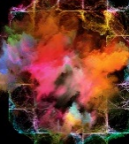
```
from django.db import models
```

```
class Program(models.Model):  
    name = models.CharField(max_length=500)  
    description = models.TextField()  
    date_added = models.DateField(auto_now_add=True)
```

```
>> Email.objects.create(  
    to="alice@example.com",  
    from="bob@example.com",  
    date="2017-04-01 13:00:00",  
    subject="Howdy!",  
    body="Created an email!")
```

```
>> e = Email.objects.filter(to="alice@example.com")  
>> e.subject = "Hello!"  
>> e.save()  
>> e.delete()
```





Django – Other fun stuff

`views.py`

- Logic for each page
- Submitted form processing

`forms.py`

- Forms framework... cleans, error checks, etc. for you

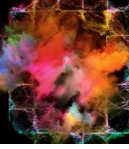
Templates

- HTML, with custom syntax
- very similar to most other template languages

`tests.py`

- Built-in testing framework

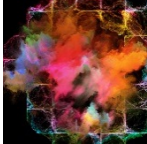
Django – Admin tools



The screenshot displays the Django Admin interface. At the top, the header bar includes the title 'Django administration' and a navigation menu with links: 'WELCOME, EKANAL.', 'VIEW SITE', 'CHANGE PASSWORD', and 'LOG OUT'. Below the header, a breadcrumb trail shows 'Home > Todo > Td lists'. A green success message states: 'The td list "TDList object" was added successfully.' The main content area is titled 'Select td list to change' and features an 'ADD TD LIST +' button. Below this, there is an 'Action:' dropdown menu with a 'Go' button and a selection count of '0 of 1 selected'. A table lists the available objects:

<input type="checkbox"/>	TD LIST
<input type="checkbox"/>	TDList object

Below the table, it indicates '1 td list'.



ReportWriter

Dashboard

Recent Actions

ekanal

[Home](#) / [Report](#) / [Reports](#) / Add Report

Add Report

Name

/

Status

Draft

Title*

Description*

CWEs*

Either click 'Suggest CWEs' to get the suggest CWE based on your description or select a CWE from the list

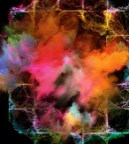
Suggest CWEs

Write my own Misuse Case and Use Case

Suggest Misuse Cases and Use Cases

Save and continue editing

Save



Django – DRF

Django Rest Framework (DRF)

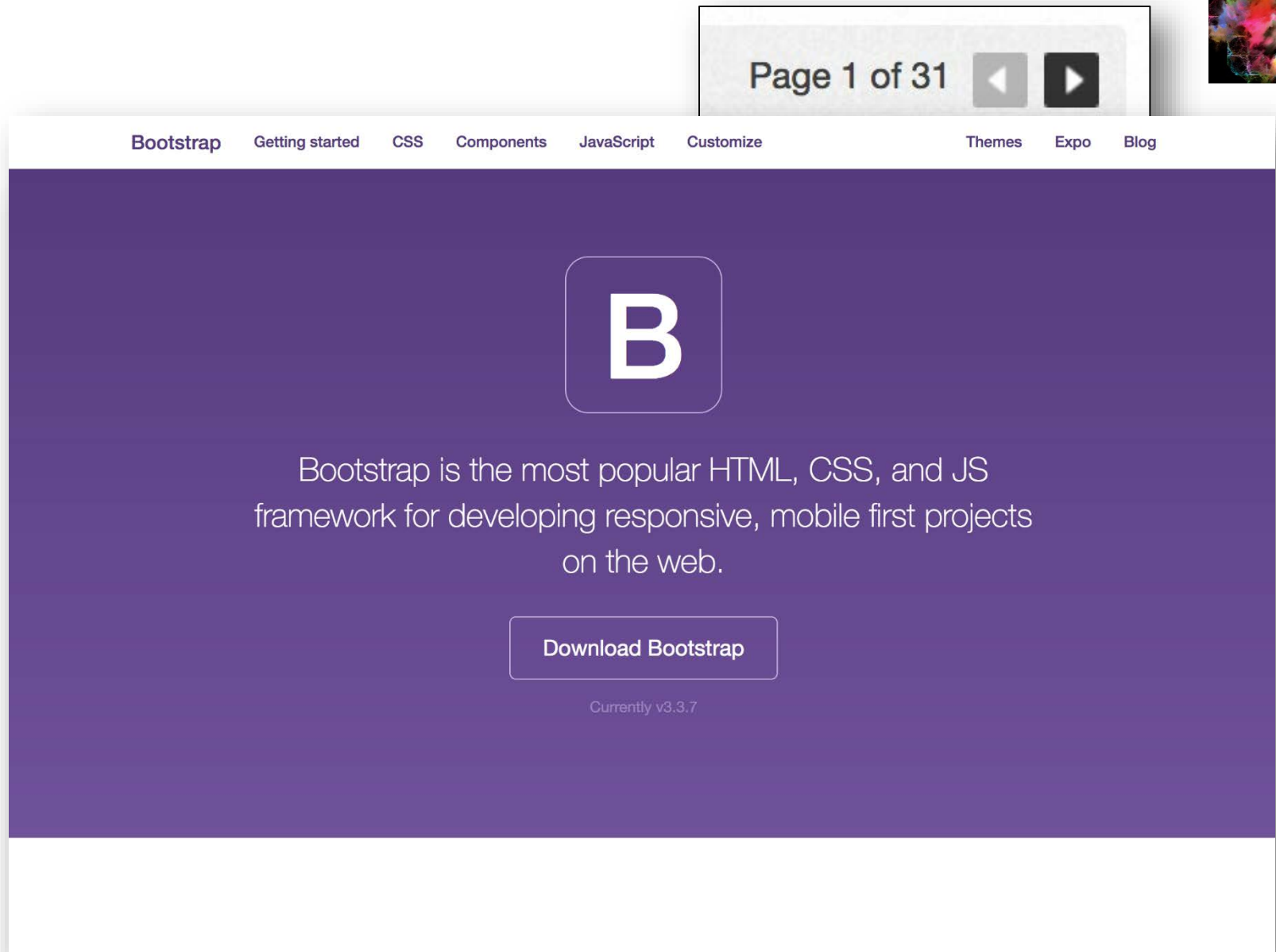
- Sophisticated API design framework
- From their website:

Some reasons you might want to use REST framework:

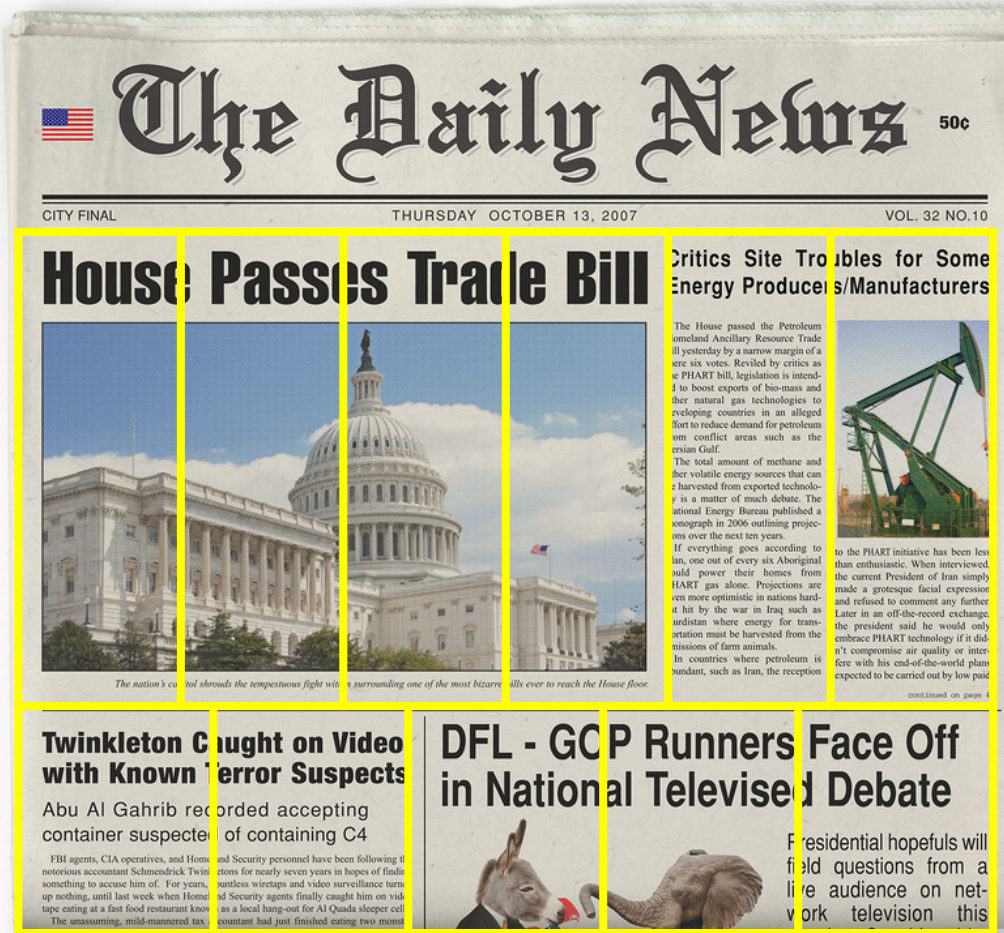
- The [Web browsable API](#) is a huge usability win for your developers.
- [Authentication policies](#) including packages for [OAuth1a](#) and [OAuth2](#).
- [Serialization](#) that supports both [ORM](#) and [non-ORM](#) data sources.

Bootstrap – Why

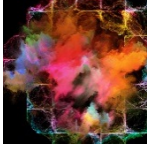
- UI Consistency
- Proven UI concepts
- Professional look
- Excellent community support:
 - Themes
 - Extensions



Bootstrap – Grid



- Think newspapers
- Columnar content
- Bootstrap deals with padding, margin
- “Just Add Classes”



.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1
.col-md-8								.col-md-4			
.col-md-4				.col-md-4				.col-md-4			
.col-md-6						.col-md-6					

```
<div class="row">
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
  <div class="col-md-1">.col-md-1</div>
</div>
<div class="row">
  <div class="col-md-8">.col-md-8</div>
  <div class="col-md-4">.col-md-4</div>
</div>
<div class="row">
  <div class="col-md-4">.col-md-4</div>
  <div class="col-md-4">.col-md-4</div>
  <div class="col-md-4">.col-md-4</div>
</div>
<div class="row">
  <div class="col-md-6">.col-md-6</div>
  <div class="col-md-6">.col-md-6</div>
</div>
```

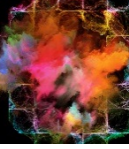
Copy

Bootstrap – Everything else



Bootstrap UI components demonstration:

- Navigation:** Title, Home, Link, Link, Dropdown, Search, Link, Dropdown.
- Buttons:** Action, Action, Danger, Warning, Success, Info, Inverse, Search.
- Buttons (Size/Style):** Large button, Large button, Left, Middle, Right, «, 1, 2, 3, 4, 5, ».
- Buttons (Size):** Default button, Default button, Small button, Small button, Mini button, Mini button.
- Dropdown Menu:** Action, Another action, Something else here, Separated link.
- Modal:** Modal header, One fine body..., Close, Save changes.
- Footer:** Home, Help, Dropdown, «, 1, 2, 3, 4, 5, ».



Bootstrap – Everything else

- Forms
- Buttons
- Wells
- Tables
- Images
- Typgraphy
- Navbars
- Breadcrumbs
- Alerts
- Thumbnails
- Progress bars
- Labels
- Badges
- Panels
- Lists
- Dropdowns
- List groups
- Glyphicons
- Tooltips
- Carousels
- Responsive menus
- Tabbed lists
- Modal boxes
- Popovers
- Transitions
- Plugins
- Themes

Bootstrap – Implementation

- Almost all through `<div>`'s and `classes`
- Encourages valid HTML
- Note: Poor bug handling (just markup, no “debugger”)

Amount (in dollars)

\$

.00

EXAMPLE

\$ Amount .00

```
<form class="form-inline">
  <div class="form-group">
    <label class="sr-only" for="exampleInputAmount">Amount (in dollars)</label>
    <div class="input-group">
      <div class="input-group-addon">$</div>
      <input type="text" class="form-control" id="exampleInputAmount"
placeholder="Amount">
      <div class="input-group-addon">.00</div>
    </div>
  </div>
  <button type="submit" class="btn btn-primary">Transfer cash</button>
</form>
```

Copy

AdminLTE

Alexander Pierce

Online

Search...

MAIN NAVIGATION

Dashboard

Dashboard v1

Dashboard v2

Layout Options

Widgets

Charts

UI Elements

Forms

Tables

Calendar

Mailbox

Examples

Multilevel

Ready to sell your theme? Submit your theme to our new marketplace now and let over 200k visitors see it! [Let's Do It!](#)

Dashboard

Version 2.0

Home > Dashboard

CPU TRAFFIC

90%

LIKES

41,410

SALES

760

NEW MEMBERS

2,000

Monthly Recap Report

Sales: 1 Jan, 2014 - 30 Jul, 2014

90

80

70

60

50

40

30

20

10

0

January

February

March

April

May

June

July

Goal Completion

Add Products to Cart

160/200

Complete Purchase

310/400

Visit Premium Page

480/800

Send Inquiries

250/500

▲ 17%

\$35,210.43

TOTAL REVENUE

▼ 0%

\$10,390.90

TOTAL COST

▲ 20%

\$24,813.53

TOTAL PROFIT

▼ 18%

1200

GOAL COMPLETIONS

Visitors Report

INVENTORY

5,200

24

eykanal



Search...



MAIN NAVIGATION

Dashboard <

Layout Options 4

Widgets new

Charts <

UI Elements <

Forms ▾

General Elements

Advanced Elements

Editors

Tables <

Calendar 17 3

Mailbox 5 16 12

Examples <

Advanced Form Elements Preview

Home > Forms > Advanced Elements

Select2



Minimal

Alabama ▼

Disabled

Alabama ▼

Multiple

Select a State

Disabled Result

Alabama ▼

Visit [Select2 documentation](#) for more examples and information about the plugin.

Input masks

Date masks:



dd/mm/yyyy



US phone mask:



Intl US phone mask:



__-__-__X__

IP mask:



Date picker

Date:



Date range:



04/30/2017 - 04/30/2017

Date and time range:



04/30/2017 - 04/30/2017

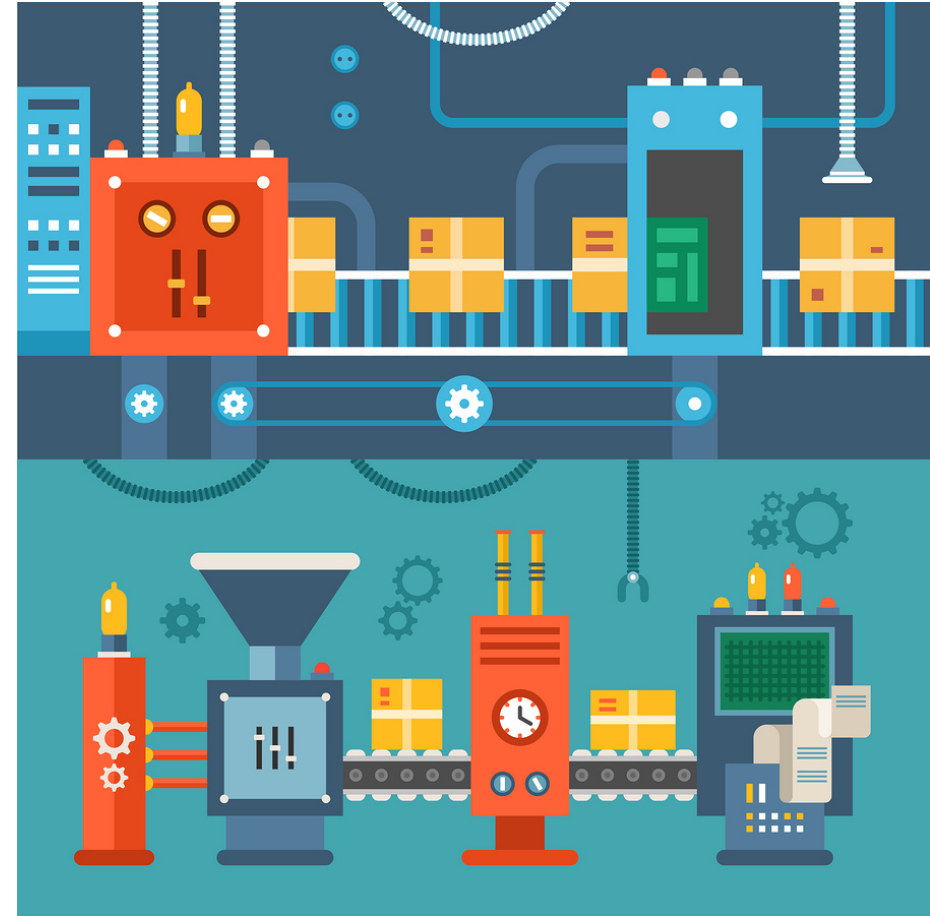
Date range button:



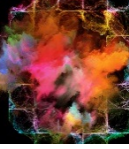
Date range picker ▼

Django – Admin tools

- Built-in interface for oft-used functionality
- Needs only model definition
- Highly customizable UI
- Limited functionality extensibility



Takeaway



- Frameworks save time, effort
- Roll-your-own is bad
- Django (and other modern tools) have powerful features
- Bootstrap is a really cheap way to look super professional



Contact Info

Eliezer Kanal

Technical Manager & Principle Researcher

Telephone: +1 412.268.5204

Email: ekanal@cert.org

blog.erikdev.com

@eykanal

