



Отчёт по лабораторной работе №4

«Шаблонизация»

По курсу «Разработка интернет приложений»

Выполнил:

Студент группы РТ5-51

Разуваев К. А.

Москва, 2017

---

### Задание:

- Реализовать view, в которых генерируются html-страницы
- В шаблонах должны быть использованы рассмотренные конструкции: переменные, вложенные значения, циклы, условия
- Все шаблоны должны расширять базовый шаблон
- Для элементов списка использовать тег `include`
- По нажатии на элемент списка должны открываться страница информации об элементе
- Для верстки необходимо использовать Bootstrap

### Исходный код:

#### MyApp/urls.py:

```
from django.conf.urls import url, include
from django.contrib import admin
from app.views import main, concerts, TicketsView, hello
from django.conf.urls.static import static
from django.conf import settings

urlpatterns = [
    url(r'^admin/', admin.site.urls),
    url(r'^$', main, name="main"),
    url(r'^(?P<id>\d+)', concerts, name='concerts'),
    url(r'^tickets/', include('app.urls')),
    url(r'^tickets/', TicketsView.as_view(), name='tickets_url'),
    url(r'^hello/', hello, name="hello"),
]

if settings.DEBUG:
    urlpatterns += static(settings.STATIC_URL,
document_root=settings.STATIC_URL)
```

#### app/urls.py:

```
from django.conf.urls import url
from . import views
from app.views import TicketView

urlpatterns = [
    url(r'^(?P<id>\d+)', TicketView.as_view(), name='ticket_url'),
]
```

#### views.py:

```
from datetime import datetime
from django.shortcuts import render
from django.views import View

def main(request):
    return render(request, 'main.html')

def tickets(request):
    return render(request, 'tickets.html')

def hello(request):
    return render(request, 'hello.html')
```

```

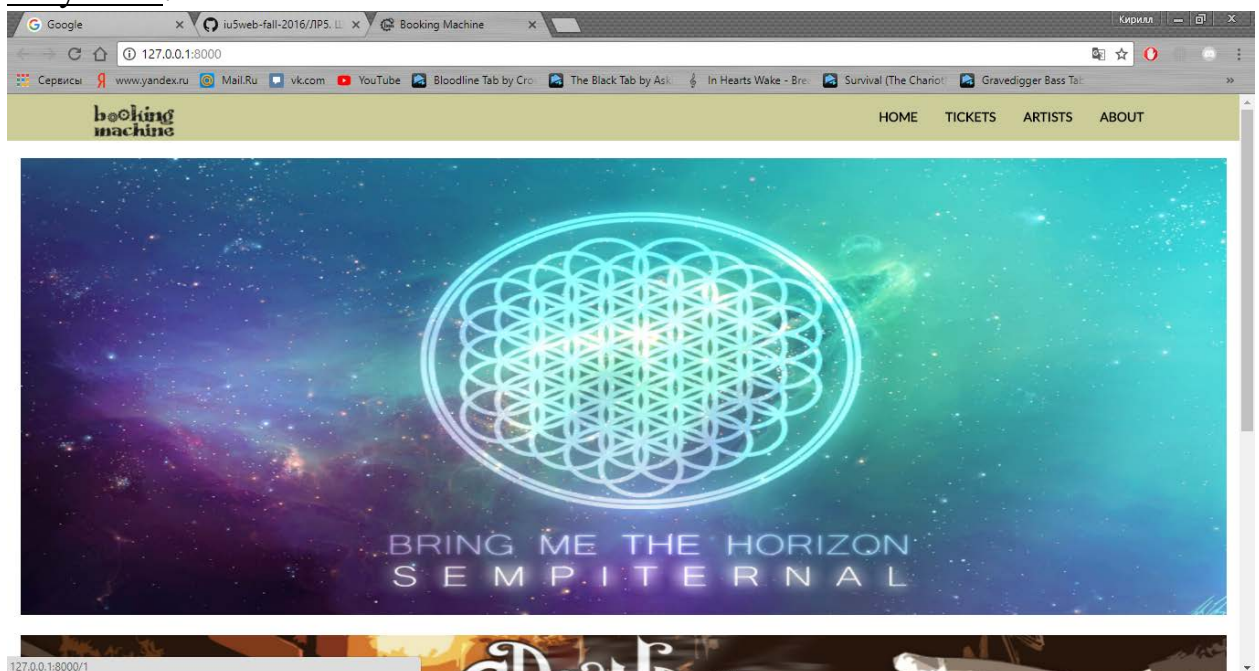
def concerts(request, id):
    name = ['BMTH_info', 'CTE_info']
    BMTH_info = 'BMTH bla bla'
    CTE_info = 'Crown the empire bla bla'
    info = [BMTH_info, CTE_info]
    data1 = {'concert': {'id': id}}
    data2 = {'concerts': [{'id': '1', 'concert_name': 'Bring Me The Horizon',
'info': BMTH_info},
                        {'id': '2', 'concert_name': 'Crown The Empire',
'info': CTE_info}]}
    return render(request, 'concerts.html', locals())

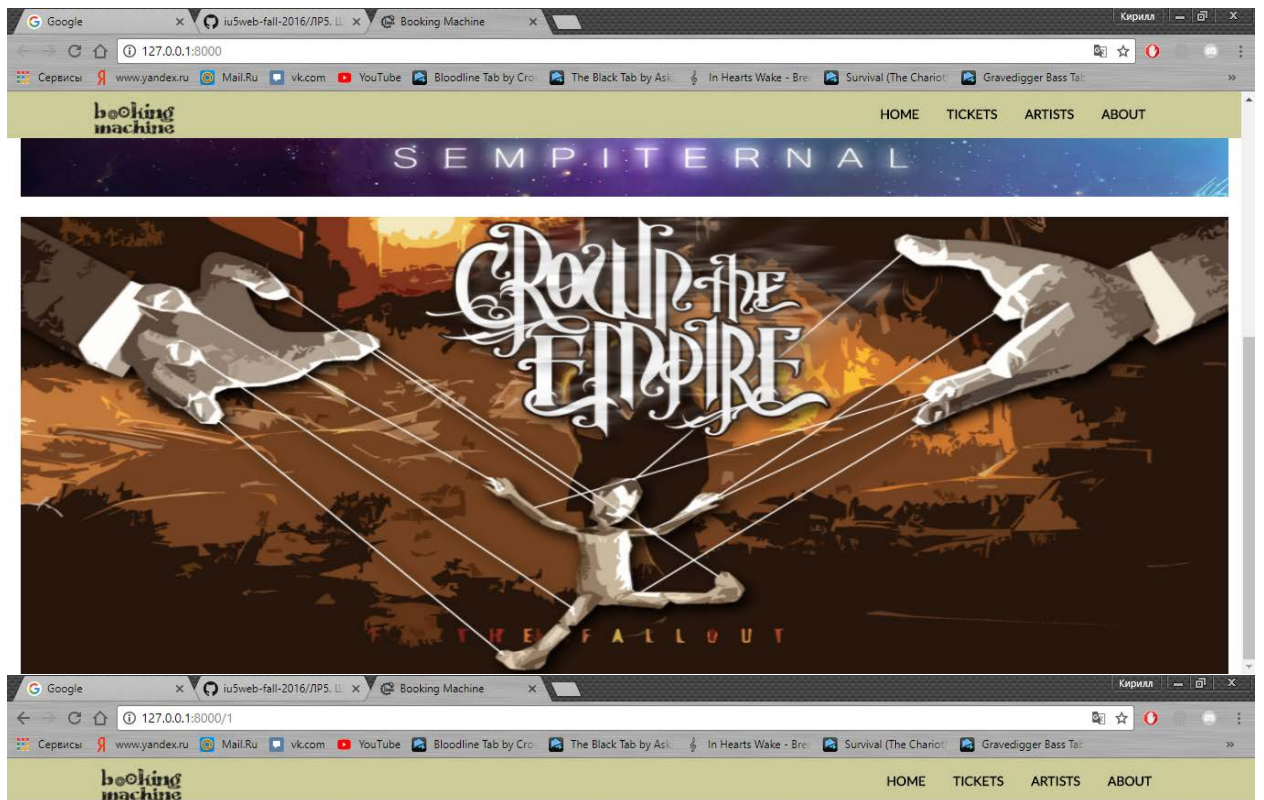
class TicketsView(View):
    def get(self, request):
        #variable = 'Django'
        today_date = datetime.now()
        data = {
            'tickets': [
                {'title': 'Первый билет', 'id': 1},
                {'title': 'Второй билет', 'id': 2}
            ]
        }
        return render(request, 'tickets.html', locals())

class TicketView(View):
    def get(self, request, id):
        #variable = 'Django'
        today_date = datetime.now()
        data = {
            'ticket': {
                'id': id
            }
        }
        return render(request, 'ticket.html', locals())

```

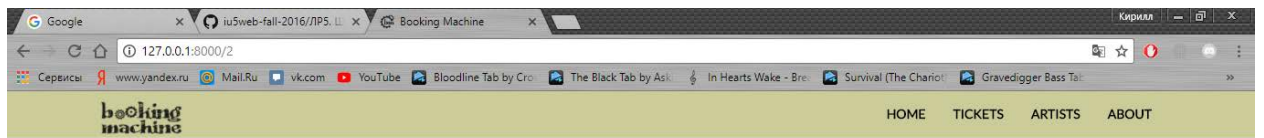
## Результат:





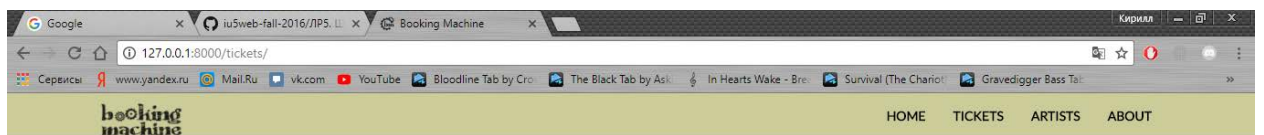
Bring Me The Horizon

BMTH bla bla



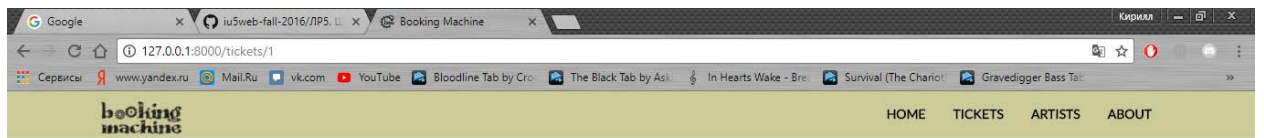
## Crown The Empire

Crown the empire bla bla



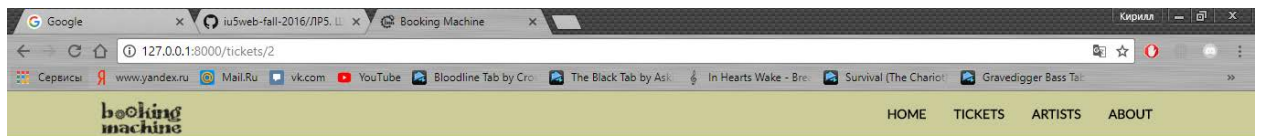
## Билеты

- Первый билет
- Второй билет



Билет №1

Страница билета №1



Билет №2

Страница билета №2

