*Федеральное государственное бюджетное образовательное учреждение* *высшего профессионального образования*

|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | ***«Московский государственный технический университет  имени Н.Э. Баумана»***  ***(МГТУ им. Н.Э. Баумана)*** |

ФАКУЛЬТЕТ Информатика и системы управления

КАФЕДРА Компьютерные системы и сети

**Отчет**

**по лабораторной работе №3**

**Разработка контроллера backend для приложения “Интернет-магазин компьютеров”**

Студент гр. ИУ6-13М\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Макаров А.В.

(Подпись, дата) (Фамилия И.О.)

Студент гр. ИУ6-12М\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Разуваев К.А.

(Подпись, дата) (Фамилия И.О.)

Студент гр. ИУ6-12М\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Колесников В.С.

(Подпись, дата) (Фамилия И.О.)

Студент гр. ИУ6-12М\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Пирмамедов М.Э.

(Подпись, дата) (Фамилия И.О.)

Преподаватель **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** Еремин О.Ю.

(Подпись, дата) (Фамилия И.О.)

Москва, 2019

**Задание:** разработать контроллер для приложения.

**Контроллер (Controller)** интерпретирует действия пользователя, оповещая модель о необходимости изменений.

Контроллер обеспечивает «связь» между пользователем и системой. Контролирует и направляет данные от пользователя к системе и наоборот. Использует модель и представление для реализации необходимого действия.

В Django контроллер это встроенный URL-маршрутизатор, который обеспечивает логику запрос-ответ.

Контроллер определяет представление в зависимости от указаний пользователя, следуя настройкам URLconf и вызывает соответствующую функцию Python для заданного URL.

Ниже представлен исходный код контроллера приложения:

**from** django.conf.urls **import** url, include  
**from** django.contrib **import** admin  
**from** app.views **import** \*  
**from** django.conf.urls.static **import** static  
**from** django.conf **import** settings  
  
urlpatterns = [  
 url(**r'^admin/'**, admin.site.urls),  
 url(**r'^$'**, main, name=**"main"**),  
 url(**r'^registration/'**, registration, name=**'registration\_url'**),  
 url(**r'^logout/'**, logout, name=**'logout\_url'**),  
 url(**r'^auth/'**, authorization, name=**'auth\_url'**),  
 url(**r'^login/'**, loginn, name=**"login"**),  
 url(**r'^order/'**, order\_add, name=**"order"**),  
 url(**r'^orderna/'**, order\_non\_auth, name=**"order\_non\_auth"**),  
 url(**r'^computers/$'**, ComputersList.as\_view(), name=**'computers\_url'**),  
 url(**r'^computer/(?P<computer\_id>\d+)/$'**, computer, name=**'computer\_url'**),  
 url(**r'^orders/'**, OrdersView.as\_view(), name=**"orders"**),  
 url(**r'^add/'**, computer\_add, name=**"computer\_add"**),  
]  
  
  
**if** settings.DEBUG:  
 urlpatterns += static(settings.STATIC\_URL, document\_root=settings.STATIC\_ROOT)  
 urlpatterns += static(settings.MEDIA\_URL, document\_root=settings.MEDIA\_ROOT)

Данный контроллер работает с HTTP-запросами. Примеры таких запросов представлены на рисунках 1,2:

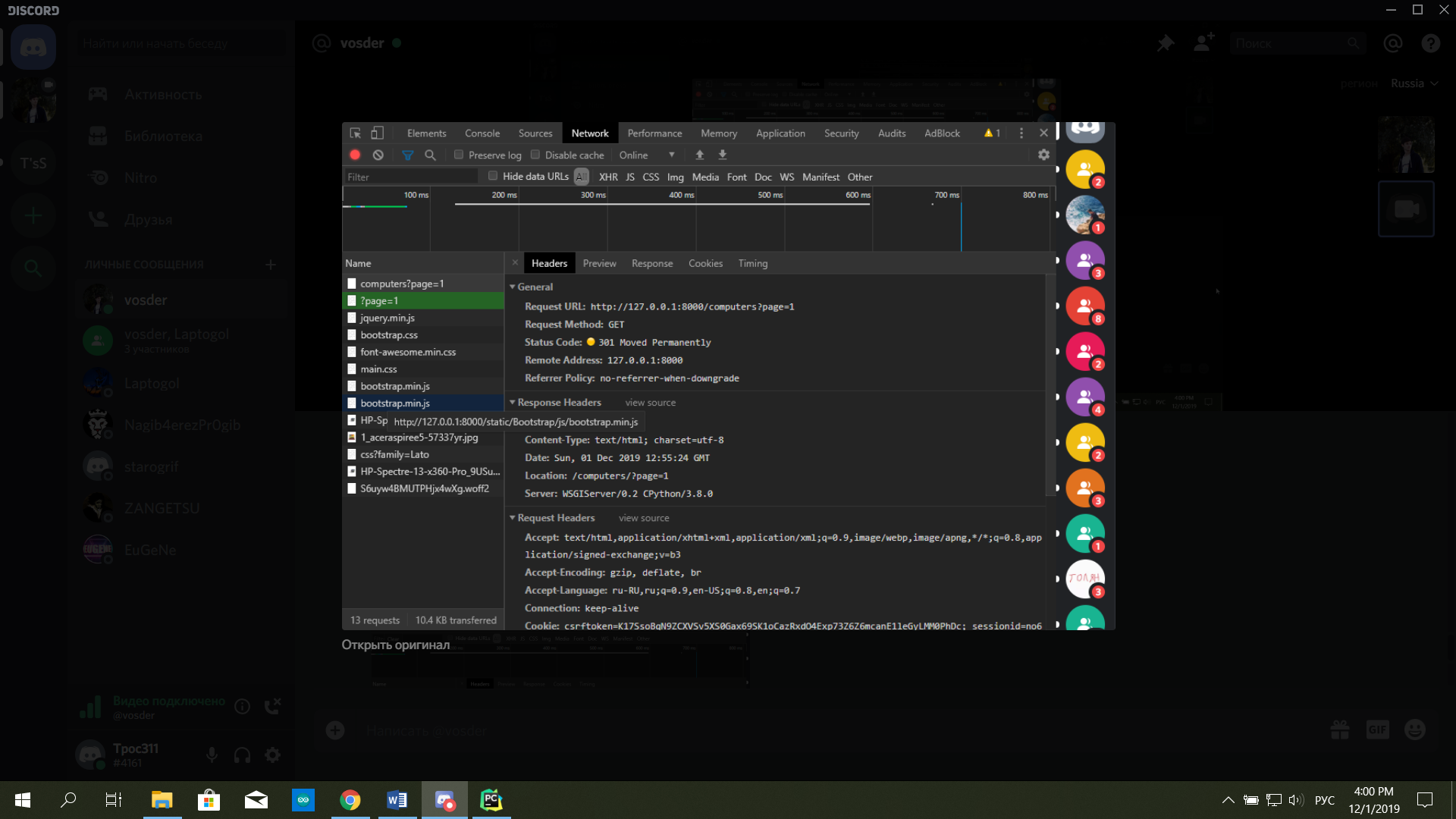


Рисунок 1 - Пример работы контроллера.

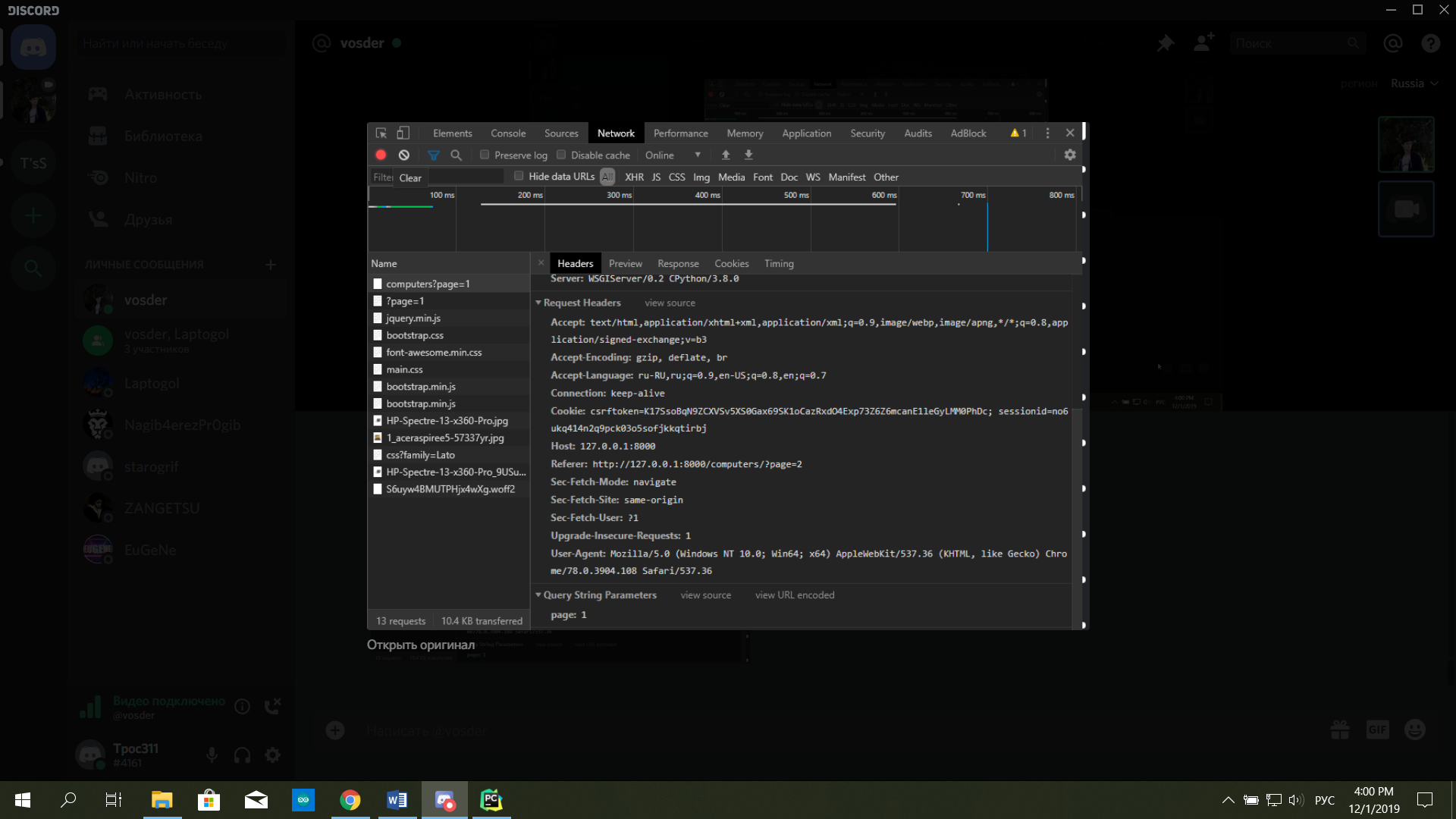


Рисунок 2 - Пример работы контроллера.

**Вывод:** в данной работе был разработан контроллер для приложения.