

**Министерство науки и высшего образования Российской Федерации**  
федеральное государственное автономное образовательное учреждение высшего  
образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**Факультет программной инженерии и компьютерной техники (ФПИ и КТ)**

**Направление подготовки (специальность) – 09.03.04  
(Нейротехнологии и программная инженерия 2024)**

**Отчет**

по лабораторной работе дисциплины  
“Веб-программирование”

Лабораторная работа №2

вариант 58999

**Автор:** Немыкин Ярослав Алексеевич

**Группа:** Р3222

**Преподаватель:** Кулинич Ярослав Вадимович

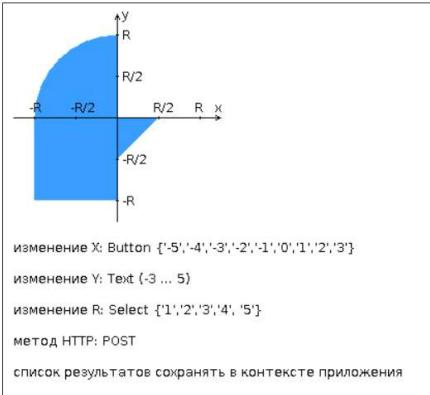
Санкт-Петербург, 2025

# Задание варианта 405271.

## Лабораторная работа #2

Вариант **5899**

Чтобы получить задание, введите свой номер варианта.



Разработать веб-приложение на базе серверов и JSP, определяющее попадание точки на координатной плоскости в заданную область.

Разработать веб-приложение на базе серверов и JSP, определяющее попадание точки на координатной плоскости в заданную область.

Приложение должно быть реализовано в соответствии с [шаблоном MVC](#) и состоять из следующих элементов:

- **ControllerServlet**, определяющий тип запроса, и, в зависимости от того, содержит ли запрос информацию о координатах точки и радиусе, делегирующий его обработку одному из перечисленных ниже компонентов. Все запросы внутри приложения должны передаваться этому сервлету (по методу GET или POST в зависимости от варианта задания), остальные сервлеты с веб-страниц напрямую вызываться не должны.
- **AreaCheckServlet**, осуществляющий проверку попадания точки в область на координатной плоскости и формирующий HTML-страницу с результатами проверки. Должен обрабатывать все запросы, содержащие сведения о координатах точки и радиусе области.
- **Страница JSP**, формирующая HTML-страницу с веб-формой. Должна обрабатывать все запросы, не содержащие сведений о координатах точки и радиусе области.

### Разработанная страница JSP должна содержать:

1. "Шапку", содержащую ФИО студента, номер группы и номер варианта.
2. Форму, отправляющую данные на сервер.
3. Набор полей для задания координат точки и радиуса области в соответствии с вариантом задания.
4. Сценарий на языке JavaScript, осуществляющий валидацию значений, вводимых пользователем в поля формы.
5. Интерактивный элемент, содержащий изображение области на координатной плоскости (в соответствии с вариантом задания) и реализующий следующую функциональность:
  - Если радиус области установлен, клик курсором мыши по изображению должен обрабатываться JavaScript-функцией, определяющей координаты точки, по которой кликнул пользователь и отправляющей полученные координаты на сервер для проверки факта попадания.
  - В противном случае, после клика по картинке должно выводиться сообщение о невозможности определения координат точки.
  - После проверки факта попадания точки в область изображение должно быть обновлено с учётом результатов этой проверки (т.е., на нём должна появиться новая точка).
6. Таблицу с результатами предыдущих проверок. Список результатов должен браться из контекста приложения, HTTP-сессии или Bean-компонента в зависимости от варианта.

### Страница, возвращаемая AreaCheckServlet, должна содержать:

1. Таблицу, содержащую полученные параметры.
2. Результат вычислений - факт попадания или непопадания точки в область.
3. Ссылку на страницу с веб-формой для формирования нового запроса.

Разработанное веб-приложение необходимо развернуть на сервере [WildFly](#). Сервер должен быть запущен в standalone-конфигурации, порты должны быть настроены в соответствии с выданным portbase, доступ к http listener'у должен быть открыт для всех IP.

## Код программы.

Код расположен на гит-хаб: <https://github.com/yargames1/ITMO/tree/main/web/lab2>

## Вывод программы.

Немыкин Ярослав Алексеевич; группа Р3222; Вариант 58999

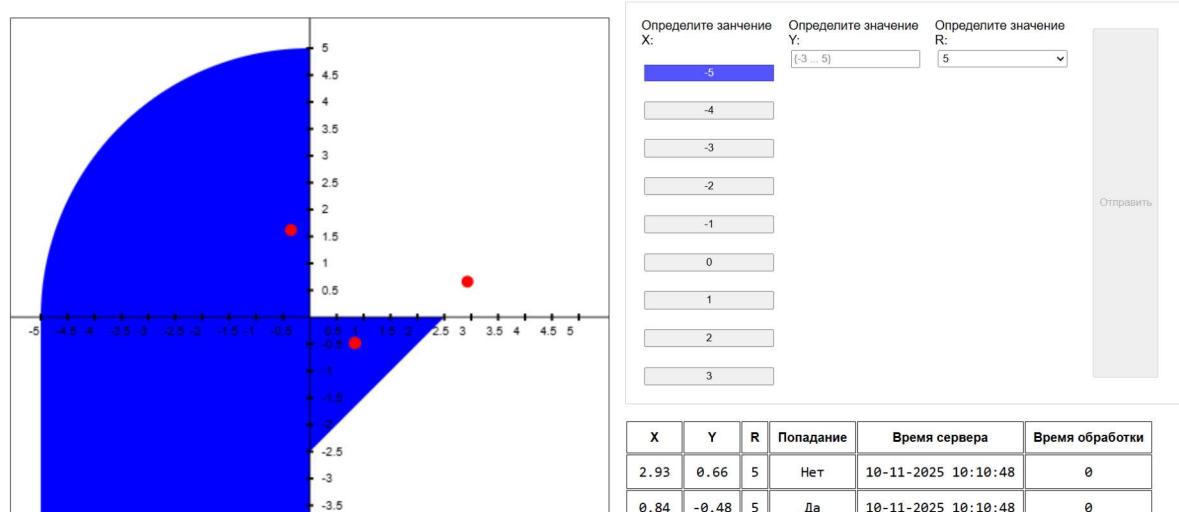


Рис. 1 главная страница

Результаты:			
X	Y	R	Попадание
-5.0	2.0	5.0	Нет

[Вернуться на главную](#)

Рис. 2 страница с обработкой результатов

## **Выход.**

В ходе выполнения лабораторной работы были получены практические навыки развертывания и настройки сервера приложений WildFly, создания и конфигурации сервлетов и JSP-страниц.

Было реализовано веб-приложение, определяющее попадание точки в заданную область координатной плоскости, с использованием архитектурного шаблона MVC. Настроена передача данных между компонентами и сохранение истории проверок через контекст приложения.

В результате была разработана и успешно развернута работоспособная версия веб-приложения на сервере WildFly, демонстрирующая корректное взаимодействие JSP, Servlet и Java-классов.