

Национальный исследовательский университет ИТМО

Факультет ПИиКТ

Лабораторная работа по веб-программированию №2

Вариант 30003

Работу подготовил:

Асташин С. С.

Группа:

Р3230

Преподаватель:

Каюков И. А.

Санкт-Петербург

2021

Задание

Разработать веб-приложение на базе сервлетов 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:

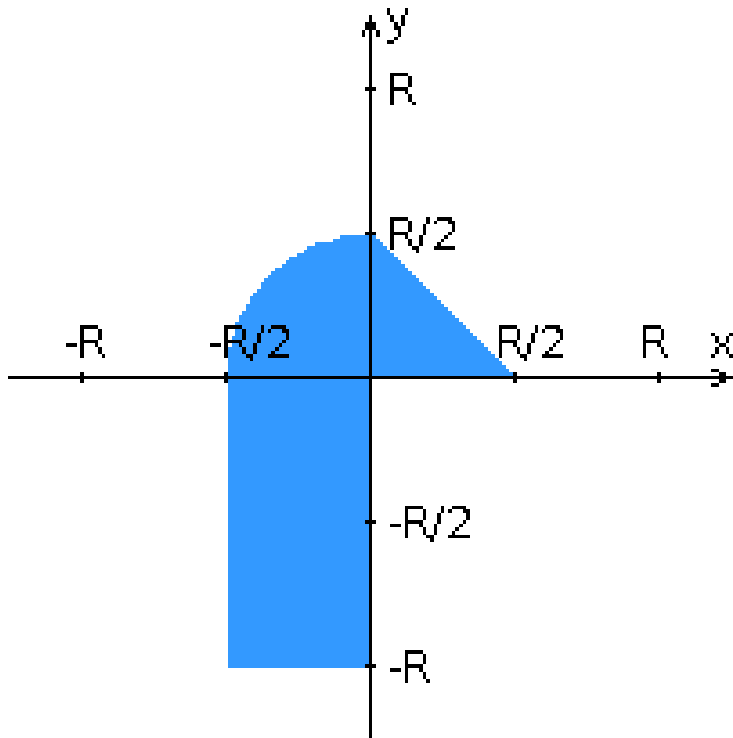
1. "Шапку", содержащую ФИО студента, номер группы и номер варианта.
2. Форму, отправляющую данные на сервер.
3. Набор полей для задания координат точки и радиуса области в соответствии с вариантом задания.
4. Сценарий на языке JavaScript, осуществляющий валидацию значений, вводимых пользователем в поля 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 непадения 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.

Область



изменение X: Button{'-4','-3','-2','-1','0','1','2','3','4'}

изменение Y: Text {-5 ... 5}

изменение R: Select {'1','1.5','2','2.5','3'}

метод HTTP: GET

список результатов сохранять в Bean-компоненте

Исходный код

https://github.com/Gramdel/web_lab2

Вывод

В ходе этой работы я познакомился с «новейшими» технологиями: JSP и сервлетами; узнал, что такое Bean-компонент и с чем его едят; научился использовать Canvas для рисования графиков (и не только) на JS. Кроме того, пришлось разбираться с WildFly (будь он неладен, никогда не думал, что можно настраивать веб-сервер дольше часа). В любом случае, лабораторная работа была достаточно интересной, особенно относительно первой – код на джаве писать гораздо приятней, чем на php, ну и в ней есть куча удобных фишек, которых там нет.