

Санкт-Петербургский политехнический университет
Институт компьютерных наук и технологий
Высшая школа «Информационных систем и супер компьютерных
технологий»

КУРСОВОЙ ПРОЕКТ
Разработка игры "Виселица"
по дисциплине «Технологии программирования»

Выполнил студент
гр. 3530901/2

Васихин Н.А.

Преподаватель

Степанов Д.С.

22 мая 2021 г.

Санкт-Петербург

2021

ЗАДАНИЕ
НА ВЫПОЛНЕНИЕ КУРСОВОГО ПРОЕКТА

студенту группы 3530901/2 Васихину Нииколаю Андреевичу

1. Тема проекта: создание игры Виселица с графическим интерфейсом.
2. Срок сдачи законченного проекта: 22 мая
3. Исходные данные к проекту: требования к реализовываемому проекту
4. Содержание пояснительной записки: введение с описанием правил игры, основная часть (технологии JavaFX, MVC и их применение в приложении), описание автоматических тестов для бизнес-логики, заключение, список использованных техноогий

Дата получения задания: «22» апреля 2021 г.

Руководитель

Степанов Д.С.

Задание принял к исполнению

Васихин Н.А.

22 апреля 2021

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	4
ОПИСАНИЕ ПРЕДЛОЖЕННОГО РЕШЕНИЯ.....	5
ТЕСТИРОВАНИЕ ПРОГРАММЫ.....	6
ЗАКЛЮЧЕНИЕ.....	7
СПИСОК ИСПОЛЪУЕМЫХ ИСТОЧНИКОВ	8

ВВЕДЕНИЕ

Цель работы: создать и протестировать игру Виселица с графическим интерфейсом.

Правила игры:

Один из игроков (или компьютер) загадывает слово — пишет на бумаге любые две буквы слова и отмечает места для остальных букв, например чертами (существует также вариант, когда изначально все буквы слова неизвестны). Также рисуется виселица с петлёй.

Согласно традиции русских лингвистических игр, слово должно быть именем существительным, нарицательным в именительном падеже единственного числа, либо множественного числа при отсутствии у слова формы единственного числа.

Второй игрок предлагает букву, которая может входить в это слово. Если такая буква есть в слове, то первый игрок пишет её над соответствующими этой букве чертами — столько раз, сколько она встречается в слове. Если такой буквы нет, то к виселице добавляется круг в петле, изображающий голову. Второй игрок продолжает отгадывать буквы до тех пор, пока не отгадает всё слово. За каждый неправильный ответ первый игрок добавляет одну часть туловища к виселице (обычно их 6: голова, туловище, 2 руки и 2 ноги, существует также вариант с 8 частями — добавляются ступни, а также самый длинный вариант, когда сначала за неотгаданную букву рисуются части самой виселицы).

Если туловище в виселице нарисовано полностью, то отгадывающий игрок проигрывает, считается повешенным. Если игроку удаётся угадать слово, он выигрывает и может загадывать слово.

ОПИСАНИЕ ПРЕДЛОЖЕННОГО РЕШЕНИЯ

Для создания графического пользовательского интерфейса (GUI) использовалась библиотека JavaFX, которая обладает большим числом заготовок элементов интерфейса. В частности были использованы следующие элементы: кнопка (Button), текст (Text), ссылка (Hyperlink), холст (Canvas), сетка/таблица (Gridpane), а также вертикальные и горизонтальные ряды (VBox и HBox). Взаимодействие пользователя с графическим элементом описывается в событиях для этого элемента. Существует множество различных событий. Например, на нажатие, на наведение или на уход мыши.

Программа была написана с использованием концепция MVC (model- view- controller) для отделения бизнес-логики от визуализации, поэтому весь код разбит на три файла App(View), Controller и Logic(Model).

В классе App содержится main функция программы, задание параметров окна и расположения внутри него всех элементов интерфейса, а также прописаны события на действия мыши.

Класс Controller содержит объявление всех объектов графического интерфейса, а также методы для работы с ними.

Класс Logic содержит переменные и методы, предназначенные для работы бизнес-логики. Все переменные здесь объявлены как приватные, а для возможности обращения к ним из Controller имеются геттеры.

В соответствии с выбранным шаблоном разработки, пользователь взаимодействует с view, все команды от пользователя обрабатывает controller, который в свою очередь обращается к model и, если это необходимо, перерисовывает view.

ТЕСТИРОВАНИЕ ПРОГРАММЫ

Для тестирования бизнес-логики было написано 5 автоматических тестов, использующих возможности библиотеки JUnit.

Первый тест проверяет вхождение введённой буквы в загаданное слово. Второй тест проверяет вхождение введенной буквы два раза в загаданное слово. Третий тест проверяет функцию корректного хода с проверкой статуса игры. Четвертый тест симулирует ситуацию, при которой игрок проигрывает с выдачей информации об этом. Пятый тест симулирует ситуацию, при которой игрок выигрывает с выдачей информации об этом.

ЗАКЛЮЧЕНИЕ

Было создано приложение с графическим интерфейсом, предназначенное для игры двух человек в Виселица. Также были разработаны автоматические тесты для проверки работоспособности кода. В ходе выполнения этого задания мною были изучены библиотека JavaFX и шаблон MVC.

Исходные файлы приложения лежат в репозитории на GitHub:

https://github.com/robot-dol-end-eb/javafx_mvc_game_viselica.

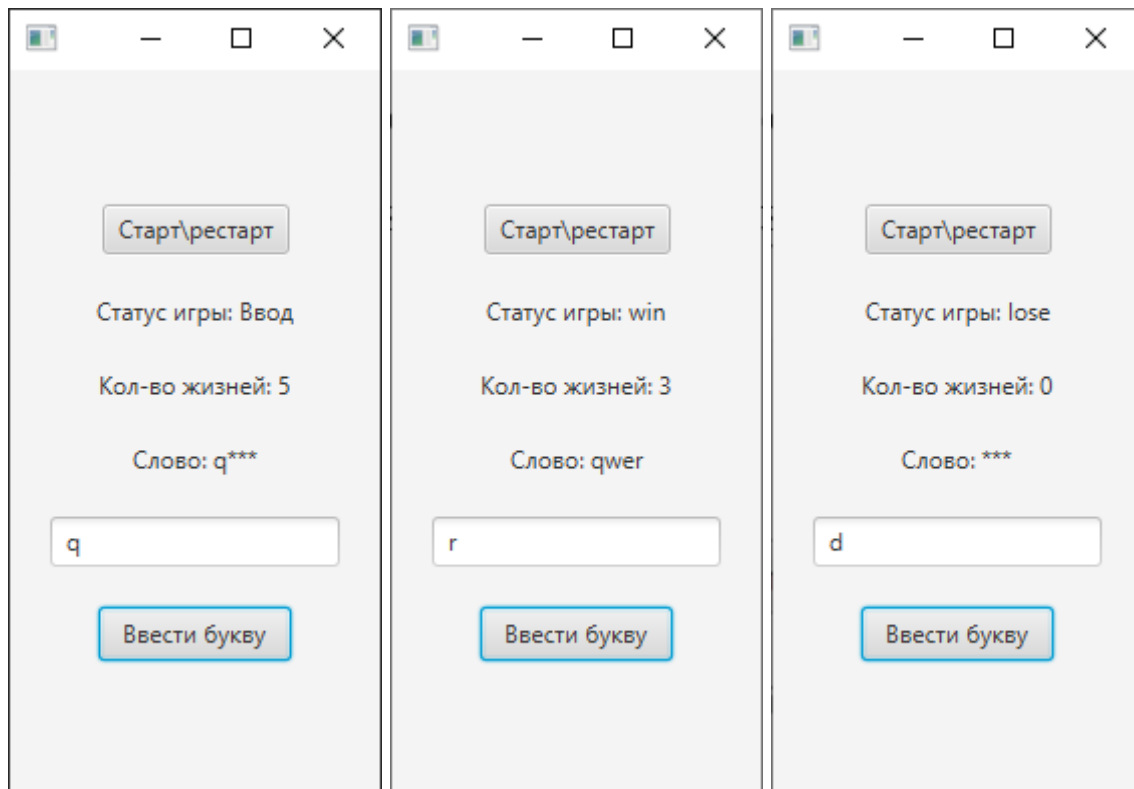


Рис 1. Скриншот приложения

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. <https://metanit.com/java/tutorial/> – описание языка Java
2. <https://metanit.com/java/javafx/> – описание JavaFX
3. <https://docs.oracle.com/javafx/2/canvas/jfxpub-canvas.htm> – описание работы с компонентом Canvas в JavaFX
4. <https://refactoring.guru/ru/design-patterns> – сборник шаблонов разработки приложений
5. <https://logic-games.spb.ru/othello/> – онлайн игра Виселица