**«Точки»**

## Описание игры

Данная игра проводится между двумя игроками на клетчатом поле с шириной 39 и высотой 32 клетки. На каждом шаге игрок должен поставить точку своего цвета на одно из свободных пересечений линий на поле (пункт). Пункт считается свободным в том случае, если он не занят другой точкой или не находится в зоне окружения. Программа должна образовывать зону окружения при появлении возможности соединения точек одного цвета непрерывной замкнутой линией и наличии точек другого цвета в замыкаемой области. Точки возможно соединить линией, если они соседствуют на одной клетке (находятся в разных ее углах). Точки соперника, попавшие в зону окружения далее не участвуют в образовании новых зон окружения. На каждом ходе программа должна подсчитывать в каждой замкнутой зоне одного игрока количество точек другого игрока и вывести суммарное количество в качестве очков этого игрока. Зоны, окруженные другой зоной в подсчете не участвуют. Игра заканчивается либо при отсутствии свободных пунктов, либо при взаимном согласии двух игроков. Побеждает игрок с наибольшим количеством очков.

## Архитектура

Программа должна состоять из клиентской части, написанной на языке программирования JavaScript с использованием языка разметки HTML5, загружающейся в браузере пользователя, и серверной части, написанной на языке программирования PHP, запускаемой на сервере, и к которой подсоединяется клиентская часть.

## Клиентская часть

Клиентская часть должна быть представлена в виде web-приложения. При первом заходе в игру программа должна предоставить пользователю возможность зарегистрироваться на сайте, введя имя пользователя и пароль. После регистрации пользователю назначается начальный рейтинг 1600. Рейтинг характеризует опытность пользователя (чем больше – тем лучше) и пересчитывается по итогам каждой игры с участием этого пользователя. Далее программа должна позволять пользователю войти под своими данными в систему путем ввода в окне авторизации имени пользователя и пароля, указанного при регистрации.

После входа игрок попадает в режим ожидания соперника. В этом случае программа должна отобразить список игроков вместе с их рейтингом, также ожидающих игру. Игроки отсортированы по рейтингу в возрастающем порядке. Также программа должна позволять игроку применить фильтр к данному списку путем нажатия кнопки «фильтр» сверху списка пользователей. После нажатия кнопки программа должна отобразить панель, в котором можно выбрать способ фильтрования. Фильтровать позволяется либо по рейтингу (вводится диапазон значений с помощью двух ползунков на одной линии, задающих минимальный и максимальный рейтинг), либо по имени. Чтобы применить фильтр, пользователь должен нажать на кнопку «Применить», чтобы сбросить фильтр, пользователь должен нажать на кнопку «Сброс». После того, как список удовлетворяющих соперников был сформирован, игрок должен иметь возможность либо самостоятельно выбрать понравившегося игрока, нажав на него в списке, либо позволить программе сделать это случайно из сформированного списка, для этого пользователь должен нажать на кнопку «Случайный выбор», находящуюся под кнопкой «Фильтр». После нажатия кнопки программа должна отобразить панель, содержащую надпись «ожидание ответа», кнопку, позволяющую отменить выбор и таймер обратного отсчета, имеющий начальное значение 15 секунд. По окончании времени, либо при отказе соперника программа должна отобразить сообщение о том, что соперник отказался от игры. У выбранного соперника программа должна вывести подтверждение, содержащее информацию об игроке, выразившем желание сразиться с ним, кнопку, с помощью которой он может начать игру, кнопку, с помощью которой он может отказаться от игры и аналогичный таймер обратного отсчета, как и у инициатора игры.

После начала игры программа должна обоим игрокам вывести одинаковое пустое поле, имя пользователя и счет первого игрока слева от поля, имя пользователя и счет второго игрока справа от поля, и состояние хода («ваш ход» или «ход соперника») сверху. Первым считается игрок, инициализировавший партию. Чтобы совершить ход, игрок должен нажать левой кнопкой мыши на пункт, который он хочет занять. После каждого хода программа должна отправлять ход на сервер и запросить у него новый счет и поле, на котором красным цветом выделяются точки и замкнутые линии первого игрока, а синим цветом точки и замкнутые линии второго игрока. В случае некорректности ход должен игнорироваться, а пользователю должна предоставиться возможность сделать ход заново. Также программа должна предоставить возможность либо предложить сопернику ничью, либо завершить игру с текущим счетом путем нажатия на специальные кнопки, расположенные под полем. После нажатия кнопки программа должна отобразить панель, содержащую надпись «ожидание ответа», кнопку, позволяющую отменить предложение и таймер обратного отсчета, имеющий начальное значение 15 секунд. По окончании времени, либо при отказе соперника программа должна отобразить сообщение о том, что соперник отказался от предложения. У выбранного соперника программа должна вывести подтверждение, содержащее кнопку, с помощью которой он может принять предложение, кнопку, с помощью которой он может отказаться от предложения и аналогичный таймер обратного отсчета, как и у предложившего игрока. Также должна быть возможность выйти из игры без подтверждения, закрыв окно браузера либо нажав на специальную кнопку, тем самым признав поражение. После окончания игры пользователь возвращается на экран ожидания.

## Серверная часть

Серверная часть должна обрабатывать регистрации пользователей и хранить в базе данных информацию о них (имя пользователя, пароль, рейтинг). Также она должна принимать ходы игроков, проверять их корректность, находить и обводить окружающие зоны, производить пересчет очков игроков, синхронизировать поля между игроками. Синхронизация должна выполняться с помощью pub/sub каналов: для каждой игры должен организовываться собственный канал, на который подписываются клиентские части обоих игроков, на данном канале серверная часть должна публиковать события, такие как ход игрока, предложения ничьи и завершение игры. Клиентская часть в реальном времени обрабатывает эти события и отображает на экране.

После окончания игры программа должна подсчитывать новый рейтинг игроков в зависимости от исхода игры по системе Эло. Сначала программа дожна вычислить вероятность выигрыша игрока A против игрока B. Эта вероятность одновременно равна наиболее вероятному количеству очков, которое наберёт игрок A в партии с B:

10.9.1:private:var:folders:x0:ygtnchn915v6s_vqtvq6c4pr0000gn:T:TemporaryItems:elo_formula1.png

где:

EA – ожидаемое количество очков, которое наберёт игрок A в партии с B,

RA – рейтинг игрока A,

RB – рейтинг игрока B.

Новый рейтинг игрока A рассчитывается по формуле:

10.9.1:private:var:folders:x0:ygtnchn915v6s_vqtvq6c4pr0000gn:T:TemporaryItems:elo_formula2.png

где:

K – коэффициент, значение которого равно 10 для сильнейших игроков (рейтинг 2400 и выше), 15 — для игроков с рейтингом меньшим чем 2400 и 25 — для игроков с рейтингом меньше чем 1700,

SA – фактически набранное игроком A количество очков (1 очко за победу, 0.5 — за ничью и 0 — за поражение),

RA’ – новый рейтинг игрока A.