**День 1 (Дизайн и верстка + JavaScript + PHP)**

**Общее время выполнения 4 часов**

В данном задании участнику необходимо разработать браузерную игру.

**Описание игры**

Игровое поле представляет собой подводный мир с плавающими рыбками. Игрок должен набрать максимальное количество баллов, кликая на них пока не закончилось время таймера. После клика по рыбке она исчезает, а игрок получает очки. Каждая рыбка движется со случайными скоростью и траекторией. Рыбка не может находиться на игровом поле постоянно, со временем она должна уплывать, если по ней так и не кликнули. Одновременно на одном игровом поле может находиться не более 10 рыбок, но не менее 5. Существуют рыбки 3 размеров. При клике на маленькую начисляется 30 очков, на среднюю – 20 очков, на большую – 10 очков. Размер игрового поля должен быть 800 на 800 пикселей. Скорость рыбок от 10 до 50 пикселей.

Участнику необходимо разработать дизайн, реализовать указанный функционал игры, а также разработать понятный и удобный интерфейс игры (менять игровую логику запрещено).

HTML и CSS код должны быть валидными. Создаваемые файлы должны структурированы и содержать комментарии. PHP и JS код должен выполняться без отображения ошибок, в т.ч. в консоли браузера.

Браузером для проверки основных функциональных возможностей является Mozilla Firefox Developer Edition. Однако работа приложения будет также проверена в браузере Google Chrome для проверки кроссбраузерности программы.

* **стартовый экран** - содержит наименование игры, поле ввода своего имени, а также кнопку “Начать игру”. Имя не может быть пустым. Кнопка “Начать игру” активна, если только введено имя пользователя.
* **экран игры** - интерфейс должен содержать игровое поле, таблицу рекордов, справочную информацию, а также дополнительную информацию. Для новичков есть всплывающие подсказки.
* **блок с таймером** - содержит таймер обратного отсчета с указанием минут, секунд в формате мм:сс. Таймер должен обновляться без перезагрузки страницы. В этом же блоке располагается кнопка “пауза”
* **блок с заработанными баллами** - количество заработанных игроком баллов за текущую игру. Обнуляется при начале новой игры.
* **блок с именем игрока** - если игрок на стартовом экране в имени ввел “terter” то имя пользователя должно отображаться по другому, так как это “кодовое” слово для игры в тестовом режиме.
* **игровое поле** - процесс игры должен включать в себя взаимодействия пользователя и компьютера и протекать без обновления страницы
* **экран окончания игры** - содержит таблицу результатов с первыми 10 лучшими результатами. Если игрок по баллам не вошел в 10 лучших, то вместо 10 места показывается его результат с указанием места в таблице рекордов. Таблица рекордов должна содержать все необходимые поля (Номер, Имя игрока, количество заработанных баллов, время, дата).

**Модуль 1**

**Время выполнения 2 часа**

В данном модуле необходимо решить следующие задачи:

* Разработка дизайна интерфейса игры, учитывая удобство использования интерфейса.
* Верстка стартового экрана, игрового поля и экрана окончания игры
* Анимация игрового поля, таблицы рекордов, окончания игры, таймера.
* Разработка анимации элементов управления и переходов от одного этапа игры к другому.

Также вам необходимо разработать страничку, в которая объясняет правила и порядок игры.

Также вам необходимо разработать и создать Базу данных для хранения необходимой информации.

**Модуль 2**

**Время выполнения 2 часа**

Реализация логики, в состав которой должны быть включены следующие функции:

* Пауза игрового процесса - останавливается время на таймере, запрещается воздействие на игровое поле. Режим паузы также может быть инициирован по нажатию на клавишу пробел. Возобновление игры так же возможно по нажатию на клавишу “пробел” или по нажатию на кнопку “Пауза” на игровом поле
* Таймер обратного отсчета - начинает обратный отсчет с началом игры (01:00), как только доходит до значения 00:00 игра заканчивается
* Таблица рекордов - показывает 10 лучших результатов игры. Если игрок не вошел в 10 лучших его результат показывает вместо 10 результата, с указанием его места в таблице рекордов.
* Сохранение итоговых результатов в базу данных на стороне сервера.
* Режима теста - режим в котором таймер обратного отсчета не запускается, и игра не останавливается при взаимодействиях, которые подразумевают проигрыш или конец игры
* Реализация логики работы игрового поля, например, изменение количества собранных объектов, уменьшение оставшегося времени и т.д., а также обработка соответствующих событий - выигрыш при наступлении 00:00 на таймере.

**Разрешенные JS-библиотеки: jQuery, jQuery UI. JavaScript фреймворки запрещены.**