

Отборочный чемпионат ФГБОУ ВО РЭУ им. Г.В. Плеханова

Компетенция веб-дизайн и разработка Примерное конкурсное задание

Задание состоит из четырех обязательных (на 2 полных дня) модулей. При этом в течении дня конкурсант выполняет одно задание, состоящее из двух модулей. Нарушать порядок выполнения модулей в рамках одного дня нельзя.

Задания построены с учетом WSSSS:

- Организация работы и управление (WSSSS 6%)
- Коммуникационные и межличностные навыки (WSSSS 6%)
- Графический дизайн веб-страниц (WSSSS 22%)
- Верстка страниц (WSSSS 22%)
- Программирование на стороне клиента (WSSSS 22%)
- Программирование на стороне сервера (WSSSS 14%)
- Системы управления контентом (Content management systems) (WSSSS 8%)

Задание первого дня (Дизайн и верстка + JavaScript + PHP)

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. 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. Если по рыбке не кликнули в течение 10 секунд, она должна уплывать. Одновременно на одном игровом поле может находиться не более 10 рыбок. Существуют рыбки 3 размеров. При клике на маленькую зачисляется 30 очков, на среднюю – 20 очков, на большую – 10 очков.

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 быть 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 имени игрока, а также кнопку “Начать игру”. Имя не может быть пустым. Кнопка “Начать игру” активна, если только введено имя пользователя.
На стартовом экране представлена инструкция к игре.
- **экран игры** - интерфейс должен содержать:
 - ✓ **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 “tester” то имя пользователя должно иметь иное визуальное оформление, 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 результатов.

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

Вы можете использовать по своему усмотрению любые медиаданные, доступные в папке media1_1 и/или создавать свои.

Модуль 1 (3 часа)

В данном модуле необходимо решить следующие задачи:

- Разработка дизайна интерфейса игры, учитывая удобство использования интерфейса. В результате выполнения задания необходимо создать psd и png файлы для каждого из экранов.

Используйте следующие имена файлов:

- для стартового экрана – X_start.psd, X_start.png;
- для экрана игры – X_play.psd, X_play.png;
- для экрана окончания игры – X_results.psd, X_results.png, где X-номер машины участника.

При проверке в первую очередь экспертами будут просматриваться png-файлы. Однако в ряде случаев эксперты могут обратиться к psd-файлам.

Созданные файлы следует сохранить в каталоге *X_day1_1/images/*.

Участники также могут представить дополнительные графические файлы на свое усмотрение с целью подчеркнуть разного рода эффекты, элементы и т.д. В этом случае дополнительные изображения сохраняются в папке *X_day1_1/images/additional/*.

- Верстка стартового экрана, игрового поля и экрана окончания игры. Вам следует использовать не менее 1 из предоставленных вам 4 шрифтов (*.ttf).
- Анимация элементов экранов. Участнику необходимо реализовать анимацию кнопок, а также всех интерактивных элементов игры (кнопок, рыбок, изменение значение таймера и др.).

Работу следует представить в виде, доступном для проверки верстки и реализованной анимации.

Все созданные и используемые файлы следует сохранить в папке *X_day1_1*, где X-номер машины участника. Структуру каталогов внутри указанной папки можете создать на свое усмотрение.

Модуль 2 (3 часа)

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

- Пауза игрового процесса - останавливается время на таймере, запрещается воздействие на игровое поле. Допустима только анимация gif-файлов. Режим паузы также может быть инициирован по нажатию на клавишу пробел. Возобновление игры также возможно по нажатию на клавишу “пробел” или по нажатию на кнопку “Пауза” на игровом поле.
- Таймер обратного отсчета - начинает обратный отсчет с началом игры, начальное значение 00:30. Как только таймер обратного отсчета доходит до значения 00:00, игра заканчивается.
- Таблица результатов - показывает 10 лучших результатов игры. Если игрок не вошел в 10 лучших, его результат отображается вместо 10 результата с указанием его места в таблице результатов.
- Сохранение итоговых результатов в базу данных на стороне сервера.
- Режим теста – режим, в котором таймер обратного отсчета не запускается, и игра не останавливается при взаимодействиях, которые подразумевают конец игры.
- Реализация логики работы игрового поля, например, изменение количества баллов, уменьшение оставшегося времени, а также обработка соответствующих событий – подсчет баллов, конец игры и т.д. должна быть представлена с анимацией.

Работа должна быть доступна с машины участника по адресу http://X_day1_2, где X-номер машины участника.

В приведенной ниже таблице описывается, как распределяются баллы относительно разделов WSSS:

WSSS SECTION	Описание	Баллы
1	Work organization and management	2
2	Communication and interpersonal skills	2
3	Website design	11
4	Website layout	13
5	Client side development	11
6	Server side development	2
7	Content management systems	0
Итого		41

Задание второго дня (PHP + JavaScript)

В данном задании участнику необходимо реализовать сервис «Будь здоров!», предоставляющий возможности записи к врачу. Использование PHP Framework по желанию участника. Также разрешено использование JQuery, JQuery UI.

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

Работа будет проверяться в браузере Google Chrome.

Модуль 3 (3 часа)

Доступ к сайту происходит по адресу: <http://X-day-2/>. Вопросы дизайна в данном модуле не рассматриваются. Вам следует обратить внимание только на расположение полей ввода.

Необходимо реализовать указанный функционал:

Регистрация - Содержит следующие обязательные поля:

- ФИО –содержит только кириллицу, пробелы, без цифр и знаков препинания – валидация на стороне клиента.
- E-mail – валидируется на соответствие шаблону e-mail адресов – на стороне клиента. Валидация на уникальность e-mail происходит на стороне сервера.
- Пароль – должен содержать не менее 6 символов английской раскладки, верхнего и нижнего регистра – валидация на стороне клиента.
Пароль хранится в БД в хешированном виде.
- Подтверждение пароля - должно совпадать с полем Пароль – валидация на стороне клиента.
- Поле для загрузки изображения пользователя (аватара) в формате png и размером до 1 Мб - валидация происходит на стороне сервера.
- кнопка Зарегистрироваться.
E-mail адрес является логином для последующей авторизации.

В случае несоответствия требованиям выводится соответствующее сообщение, поля с ошибками выделяются.

Авторизация

При вводе невалидной пары логин-пароль пользователю отображается сообщение об ошибке.

Логин администратора: admin, пароль: WwSsRr.

При успешной авторизации пользователь должен перенаправляться в личный кабинет с возможностью просмотра своих записей на прием к врачу. Личный кабинет доступен только авторизованному пользователю.

Панель управления сайтом - страница доступная по адресу <http://X-day-2/admin>, где X – номер участника. Панель управления сайтом доступна только администратору.

Возможности неавторизованных пользователей

- Вход в личный кабинет по логину и паролю
- Регистрация
- Просмотр списка специальностей врачей

Возможности пользователя после авторизации:

- После выбора специальности врача и даты планируемого посещения, на экране отображается список специалистов данного профиля(специальности) с указанием количества свободных номерков на выбранную дату.
Пользователь может записаться на прием к любому врачу на указанную дату только при наличии свободных номерков (Максимальное количество свободных номерков на любой день для каждого специалиста – 5).
- Просмотр своих записей на прием с фильтром (активные, прошедшие)
- Отмена записи (не позднее 1 дня до приема). Например, 20 сентября нельзя отменить записи на прием 20 сентября и более ранние сроки. Освободившийся номерок доступен для других пользователей.

Разделы администратора:

- Добавление врача с указанием ФИО, специальности.

Все авторизованные пользователи могут выйти из профиля по нажатию на ссылку/кнопку «Выход».

Модуль 4 (3 часа)

Перед участником ставится задача улучшения работы сервиса путем изменения взаимодействия пользователя с интерфейсом. Для этого необходимо реализовать следующий функционал:

- Пагинация списка специальностей. На одной странице представлено не более 4 специальностей. Остальные специальности представлены на следующих страницах. Возможен переход к следующей и предыдущей странице по ссылкам «Вперед» и «Назад».
- Усовершенствовать возможность администратора сайта добавлять врача: при добавлении нового врача поиск специальности происходит с помощью автодополнения (подсказки при наборе текста) слов в поиске (ajax).
- Импорт данных при добавлении врача из XML файла - участнику предоставляется файл, ему необходимо написать страницу для его импорта на сайт. Вы можете загрузить данные из файла, указав путь к нему в программном коде, или создать форму для загрузки файла. В случае необходимости вы можете внести изменения в XML-файл.
- Улучшение дизайна и удобства использования сервиса, в т.ч. учитывая целевую аудиторию – русскоговорящие пользователи старше 60 лет.
- Создание логотипа. Логотип должен отображать специфику сервиса и соответствовать целевой аудитории.
- Отмена записи к врачу происходит после подтверждения.
- На сайте используется не менее 4 иконок (пиктограмм).

Разрешенные PHP-фреймворки: Yii2 (basic), Laravel 5

В приведенной ниже таблице описывается, как распределяются баллы относительно разделов WSSS:

WSSS SECTION	Описание	Баллы
1	Work organization and management	2
2	Communication and interpersonal skills	2
3	Website design	9
4	Website layout	7
5	Client side development	9
6	Server side development	10
7	Content management systems	0
Итого		39