Q1

Расскажите, чем, на ваш взгляд, отличается хорошее клиентское приложение от плохого с точки зрения

* пользователя;
* менеджера проекта;
* дизайнера;
* верстальщика;
* серверного программиста.

Пользователь

Для пользователя приложение будет хорошим, если оно будет отвечать следующим критериям:

* простое и понятное в использовании: с интерактивным подсказками, пошаговой инструкцией по работе, с отсутствием избыточных опций - приложение должно прежде всего выполнять свой основной функционал;
* быстрое и дающее качественный результат;
* надёжное, то есть не подверженное сбоям или техническим неисправностям;
* доступное на различных устройствах - от ноутбуков до телефонов - и разных операционных системах;
* безопасное, то есть сохраняющее конфиденциальность личных данных пользователя, защищающее свои базы данных и каналы передачи информации современными методами шифрования;
* визуально привлекательное - сам как пользователь обращаю внимание на стиль!

Отсустствие соответствия какому-либо из вышеперечисленных критериев будет признаком плохого приложения для пользователя.

Менеджер проекта

Для менеджера проектов главное - удовлетворённость клиента. Следовательно, если клиенты довольны, то о приложении будут хороше отзывы, что привлечёт новых клиентов и поднимет его популярность. Таким образом, хорошее приложение для пользователя - это хорошее приложение для менеджера проектов. Также можно отметить, что приложение будут хорошим, если объём усилий, затрачиваемых на его поддержание, будет небольшим. Отсутствие ошибок технического характера (багов) приведёт к тому, что отпадёт необходимость в том, чтобы их исправлять, выпускать патчи и т.д.

Дизайнер

Приложение хорошо, если имеет презентабельный вид, выполнено в едином стиле и не оскорбляет эстетических чувств ценителей прекрасного. У приложения должна быть удачно подобрана цветовая гамма, а вся информация, содержащаяся в нём, должна быть легко считываемой, что обеспечивается правильно подобранными размером и типом шрифта, контрасностью, визаульными методами выделения особо важных данных и другими приёмами. Симметричность расположения элементов приложения также будет влиять на его качество с точки зрения дизайнера.

Верстальщик

Вёрстка должна быть адаптивной, то есть приспосабливающейся под размеры экранов разных устройств, и кроссбраузерной, то есть одинаково отображающейся в разных браузерах. Элементы приложения должны быть сгруппированы таким образом, чтобы при необходимости стили и порчядок отображения любых из них можно было легко и быстро массово изменить.

Серверный программист

Приложение должно предоставлять данные на сервер в качеством формате: не допускается предоставление неполных или неправильно заполненных данных. Для этого в клиентской части приложения (front-end) должна иметься “защита от дурака” : например, данные формы не будут отправлены, если имеются незаполненные поля, или значение одного из полей превышает допустимое количество знаков (номер банковской карты не может превышать 1 знаков и тому подобное). Также каналы передачи данных с приложения на сервер должны быть защищены шифрованием.

Q2

Опишите основные особенности разработки крупных многостраничных сайтов, функциональность которых может меняться в процессе реализации и поддержки. Расскажите о своём опыте работы над подобными сайтами: какие подходы инструменты и технологии вы применяли на практике, с какими проблемами сталкивались и как их решали.

Опыта работы по работе с многостраничными сайтами я не имею, однако мне известно, что структура (HTML) и стилевые компоненты сайта (CSS) должны быть разделены на отдельные файлы. Стилевые компоненты целесообразно хранить в одном файле и давать на него ссылку во всех файлах HTML. На всех страницах сайта будут присутствовать общие для них элементы: панель навигации, футер (footer) и так далее. Храня стили на все страницы в одном документе, можно легко их менять при необходимости, то есть не придётся заходить в каждый файл HTML и вносить в нём изменения - выполнять одну и ту же операцию десять раз. Отредактировав единый файл CSS, можно одновременно внести изменения в стилевые компоненты всех HTML страниц.

Q3

При разработке интерфейсов с использованием компонентной архитектуры часть используются термины Presentational Components и Container Components. Что означают данные термины? Зачем нужно такое разделение, какие у него есть плюсы и минусы?

Затрудняюсь ответить на этот вопрос, так как незнаком с данными терминами.

Q4

Как устроено наследование в JS? Расскажите о своём опыте реализации JS-наследования без использования фреймворков.

В JavaScript имеется прототипное наследование. Каждый объект имеет свойство, которое указывает на другой объект - прототип объекта. Прототип - это объект, свойства и методы которого наследует текущий объект. В свою очередь, текущий объект может являться прототипом для другого объекта.

Наследование в JS реализуется посредством классов (class) и их экземпляров (instance). Например, объект “Шарик” наследует свойста класса “Собака”. Кроме того, на основе одного объекта можно сделать его модифированную копию посредством ключевого слова extends. Например, “Снежок” extends “Шарик”. Допустим, мы хотим перенести все свойства “Шарика” в объект “Снежок”, плюс добавить “Снежку” новый метод, присущий только ему: “ходить на задних лапах”.

Не имею опыта работы над проектами, задействующими реализацию наследования.

Q5

Какие библиотеки можно использовать для написания тестов end-to-end во фронтенде? Расскажите о своём опыте тестирования веб-приложений.

Не имею опыта по тестированию веб-приложений.

Q6

Вам нужно реализовать форму для отправки данных на сервер, состоящую из нескольких шагов. В вашем распоряжении дизайн формы и статичная вёрстка, в которой не показано, как форма должна работать в динамике. Подробного описания, как должны вести себя различные поля в зависимости от действий пользователя, в требованиях к проекту нет. Ваши действия?

Подготовить свои предложения и представить их на согласование заказчику. В случае его одобрения приступить к технической реализации представленных предложений.

Q7

Расскажите, какие инструменты помогают вам экономить время в процессе написания, проверки и отладки кода.

DevTools в браузере Google Chrome, редактор кода Atom.

Q8

Какие ресурсы вы используете для развития в профессиональной сфере? Приведите несколько конкретных примеров (сайты, блоги и так далее).

Какие ещё области знаний, кроме тех, что непосредственно относятся к работе, вам интересны?

<https://teamtreehouse.com/>

<https://www.codecademy.com/>

<https://stackoverflow.com/>

Мне также интересны математика, астрономия, другие языки программирования.

Q9

Расскажите нам немного о себе и предоставьте несколько ссылок на последние работы, выполненные вами.

По образованию я экономист. После окончания университета работал в различных организациях: коммерческом банке, фирме по оценке стоимости жилой недвижимости, ВУЗе.

Заинтересовался областью IT и web-разработкой. В 2018 году окончил курсы по вёрстке сайтов на Codecademy, где изучил HTML, CSS. В том же году записался на дипломную программу “Full Stack JavaScript” на teamtreehouse, которую осваиваю по настоящее время.

Владею английским языком на уровне Intermediate.

Мои учебные проекты можно посмотреть на GitHub:

<https://github.com/enamarat/techdegree-project-5>

<https://github.com/enamarat/techdegree-project-4>

<https://github.com/enamarat/techdegree-project-3>

<https://github.com/enamarat/techdegree-project-2>

<https://github.com/enamarat/techdegree-project-1>