

Тестовое задание для кандидата на вакансию C# (ASP.NET/Xamarin) программиста

Предлагаем Вам продемонстрировать компании Сайберс Ваши знания и решить приведённую ниже задачу. Дополнительные задания 1 и 2 расположены в порядке возрастания сложности и выполняются по желанию (при этом можно выполнить любое из них).

Пожалуйста, обратите внимание на требования к технологиям, которые необходимо использовать, и качеству кода. Соблюдение всех требований не обязательно, но несоблюдение их будет минусом.

Если у Вас возникнут вопросы, пишите нам или звоните по телефону +7 (383) 330-6626.

Выполненное тестовое задание ждем на почту work@sibers.ru.

Желаем успехов!

ОБЯЗАТЕЛЬНЫЕ ТРЕБОВАНИЯ

Код выполненного задания должен быть структурирован, легко читаем и содержать необходимые комментарии. Написанная Вами программа должна являться законченным программным продуктом, т.е. должна легко устанавливаться, предусматривать обработку нестандартных ситуаций, быть устойчивой к некорректным действиям пользователей и т.д.

Задания можно выполнять:

- на Xamarin: .NET C# (.NET Standard 2 (iOS или Android), SQLite),
- на ASP.NET: .NET C# (EntityFramework, ASP.NET MVC 5/Web API 2/Core 2).

ТРЕБОВАНИЯ К ТЕХНОЛОГИЯМ

Нефункциональные требования:

- Трёхуровневая архитектура проекта (уровень доступа к данным, уровень логики, уровень представления).
- В качестве хранилища данных желательно использовать MSSQL или SQLite. Базу предоставить в виде SQL script-ов.
- При использовании сторонних фреймворков и пакетов — они должны быть общедоступными, и желательно описать используемые версии в пояснительной записке.
- Рекомендовано использовать units test для отладки уровня логики.
- Для выполнения задания на ASP.NET желательно продемонстрировать использование клиентского скрипта на JavaScript.
- Использовать как минимум версию MVC 5.

ТРЕБОВАНИЯ К ФОРМАТУ ПРЕДСТАВЛЕНИЯ РЕЗУЛЬТАТА

Выполненное ТЗ должно быть собрано в архив и включать в себя:

1. Обязательно: с готовой программой предоставить все исходные файлы вместе с проектными файлами.
2. В случае необходимости — предоставить описание конфигурации системы, процесса запуска в пояснительной записке.

Бинарные файлы отправлять не нужно.

ВРЕМЯ НА ВЫПОЛНЕНИЕ ТЕСТОВОГО ЗАДАНИЯ

В среднем выполнение трёх подобных задач у разработчика соответствующего уровня может занять до 8–16 часов (на Xamarin) и 12–16 часов (на ASP.NET).

Задача: Программа для занесения данных о проекте в БД

Реализовать программу для занесения данных о проектах в базу данных и дальнейшей работы с этими данными.

Функциональные требования:

- Возможность создавать/просматривать/редактировать/удалять информацию о проектах.
- Возможность создавать/просматривать/редактировать/удалять информацию о работниках.
- Возможность добавлять и удалять работников с проекта (один сотрудник может работать одновременно на нескольких проектах, и на одном проекте может работать несколько человек).
- Для просмотра проектов предусмотреть различные способы фильтрации (по диапазону даты начала, по приоритетам, и т.п.) и сортировки (по основным полям).

Необходимая для хранения информация:

- название проекта,
- название компании-заказчика,
- название компании-исполнителя,
- данные о сотруднике (имя, фамилия, отчество, e-mail человека),
- данные о руководителе проекта,
- данные об исполнителях проекта,
- даты начала и окончания проекта,
- приоритет проекта (целочисленный).

Дополнительное задание 1:

1) Добавить сущность «Задача», используя Entity Framework Code First DB Migrations.

Функциональные требования:

- Возможность создавать/просматривать/редактировать/удалять информацию о задачах.
- Возможность добавлять и удалять задачи из проекта (один проект может содержать несколько задач).
- Возможность добавлять и менять работников (исполнителей проекта) на задачи (у задачи может быть только один работник, у одного работника может быть много задач).
- Для просмотра задач предусмотреть различные способы фильтрации (например, по статусу) и сортировки (по основным полям).

Необходимая для хранения информация:

- название задачи,
- автор задачи (связь с сущностью «Работник»),
- исполнитель задачи (связь с сущностью «Работник»),
- статус задачи (перечисление ToDo/InProgress/Done),
- комментарий,
- приоритет задачи (целочисленный).

2) Используя ASP.NET Identity 2.0, добавить в проект авторизацию и контроль прав доступа.

Функциональные требования:

- Добавить три типа ролей: руководитель, менеджер проекта, сотрудник.
- Руководитель может видеть все страницы и управлять всеми сущностями.
- Менеджер проекта может: а) просматривать свои проекты и назначать/убирать сотрудников с проекта, б) управлять задачами на проекте, назначать исполнителя, менять статус. Менеджер проекта не может добавлять в систему новых сотрудников.
- Сотрудник может: а) просматривать свои проекты, б) просматривать свои задачи и менять их статус.

Дополнительное задание 2:

Используя один из JavaScript MVVM фреймворков (например, Knockout.js или Vue.js), реализовать визард создания проекта.

Функциональные требования:

- Визард состоит следующих шагов:
 1. Заполнение названия проекта, дат начала и окончания проекта, приоритета проекта.
 2. Заполнение компании-заказчика и компании-исполнителя.
 3. Выбор руководителя проекта из списка всех сотрудников.
 4. Выбор исполнителей для проекта из списка всех сотрудников.
 5. Загрузка документов проекта.
- На шагах выбора руководителя и исполнителя проектов использовать выпадающий список сотрудников с возможностью частичного ввода текста для последующего поиска сотрудников на сервере через AJAX-запрос и ASP.NET Web API.
- На пятом шаге визарда реализовать HTML5 file uploader с возможностью drag & drop.