

Тестовое техническое задание

1. Цель технического задания

Показать знания в разработке ios приложений, посредством написания тестового мобильного приложения.

2. Основные требования к разработке

- Проект на Swift 5, верстка кодом uikit или swiftui, не используя storyboard, только iphone и портретная ориентация экрана.
- Шрифт Roboto (<https://fonts.google.com/specimen/Roboto>), иконки любые.
- Вёрстка по своему усмотрению.
- Использовать Наше API (<https://plannerok.ru/docs#/>).
- Использование JWT (<https://jwt.io/>).
- Обработка ошибок и состояние загрузки.
- Можно использовать любые сторонние библиотеки.
- Приветствуются дополнения, не описанные в задании.
- Проект опубликовать на GitHub, качественное ведение git.

3. Структура проекта

Экраны:

1. Авторизация.
 - 1.1. Номер телефона.
 - 1.2. Код подтверждения СМС.
2. Регистрация.
3. Чаты. (Только верстка самого простого чата, без функционала)
 - 3.1. Чат. (Только верстка самого простого чата, без функционала)
4. Профиль.
 - 4.1. Редактирование профиля.

4. Запросы

Все запросы кроме </api/v1/users/register/>, </api/v1/users/send-auth-code/>, </api/v1/users/check-auth-code/>, должны содержать значение заголовка авторизации с помощью токена доступа (value: "Bearer **access token**", forHTTPHeaderField: "Authorization"). Access token (время жизни 10 минут) полученный при регистрации или авторизации профиля. По истечению 10 минут access token будет недействителен, ответ запроса будет 401, Вам следует сделать refresh token, запросом POST - </api/v1/users/refresh-token/> и повторить предыдущий запрос. Не забывайте, что одновременно может быть более одного запроса.

5. Требования к экрану «Авторизация»

- Ввод номера телефона с маской, флагом и кодом страны. Страну можно выбрать, вручную кликнув по флагу или ввести код страны. Должен быть placeholder. Дефолтное значение: currentRegion.
- Отправка номера телефона (phone: String) запросом POST - </api/v1/users/send-auth-code/>, пример номера телефона: +79219999999.
- При удачном запросе, пользователь должен ввести код подтверждения, смс не придет, для теста используем код: 133337. Ввод кода подтверждения должен состоять из 6 символов, только цифры.
- Отправка данных (phone: String, code: String) запросом POST - </api/v1/users/check-auth-code/>.
- При удачном запросе Вы получите refresh token, access token, user id, is user exists.
- Если is user exists == true, авторизуем пользователя, либо отправляем на регистрацию.
- Не забудьте сохранить refresh token, access token.

6. Требование к экрану «Регистрация»

- Показывать номер телефона, без возможности редактирования.
- Ввод имени пользователя.
- Ввод username пользователя:
 - Заглавные латинские буквы: от A до Z (26 символов)
 - Строчные латинские буквы: от a до z (26 символов)
 - Цифры от 0 до 9 (10 символов)
 - Символы: - _ (2 символа)
- Отправка данных (phone: String, name: String, username: String) запросом POST - </api/v1/users/register/>.
- При удачном запросе Вы получите refresh token, access token, user id.
- Авторизуем пользователя, либо выдаем toast с ошибкой.
- Не забудьте сохранить refresh token, access token.

7. Требование к экрану «Чаты»

- Вёрстка по своему усмотрению, должна содержать список чатов.

8. Требование к экрану «Чата с сообщениями»

- Вёрстка по своему усмотрению, должна содержать список сообщений и TextField прикреплена к нижней части экрана и кнопки отправить (Эта панель должна быть всегда видна). При нажатии на TextField открывать клавиатуру, панель с TextField и кнопкой отправки прикреплена кверху клавиатуры. Клавиатура должна быть interactively.

9. Требование к экрану «Профиль»

- Должен содержать аватарку пользователя, номер телефона, никнейм, город проживания, дату рождения, знак зодиака по дате рождения, о себе.
- Все данные пользователя можно получить запросом GET - </api/v1/users/me/>.
- Данные пользователя храним на устройстве, чтобы не делать повторных запросов.

10. Требования к экрану «Редактировать профиль»

- Изменение всех данных, кроме phone и username.
- Изменение аватарки или добавление новой.
- Сохраняем новые данные на устройстве и отправляем на сервер.
- Отправка данных запросом PUT - </api/v1/users/me/>.
- Данные аватарки: название файла и сам файл base 64.

11. Стилизовое оформление

Стиль оформления по своему желанию.

12.Сроки выполнения и проверка

Максимальный срок выполнения задания – 14 суток. При проверке будет учитываться скорость выполнения задания, качество, полнота выполнения и удобство интерфейса. Задание можно сдать частично выполненным.

13.Результат выполнения задания

Результатом задания должно быть работоспособное мобильное приложение. Проект предоставляется в виде ссылки на git-проект.

Максимальный срок выполнения задания - 2 недели.

Преимуществом будет не только качественно выполненное задание, но и минимальные сроки его выполнения.

