* Программист Front-End

1. Уточните все непонятные вам детали приложения, сформулировав вопросы в письменном или устном виде (можно на русском, английский приветствуется).
2. Если Вы не пишете серверную часть, то мы вышлем Вам документацию на существующие веб-сервисы.
3. Создайте веб приложение.

Попугая крылья. Обзор приложения.

Назначение приложения Попугая крылья переводить валюту PW, внутреннюю валюту между пользователями системы.

Приложение будет очень вежливым и будет информировать пользователя о любой проблеме (т.е. логин не удачен не достаточно PW чтобы освободить транзакцию и т.д.)

Регистрация пользователя

Любой человек на земле может зарегистрироваться в сервисе бесплатно, предоставив его имя валидный email и пароль.

Когда новый пользователь регистрируется, система проверит, что у email предоставленного пользователем является уникальным и так же предоставлено человеческое имя и пароль. Эти 3 поля обязательные. Для того, чтобы быть уверенными в правильности пароль должен быть напечатан дважды. Не требуется какой-либо проверки подлинности email.

По удачной регистрации каждый пользователь будет вознаграждён 500 (пятьюстами) PW на начальном балансе.

Вход в систему

Пользователи входят в систему, используя их email и пароль.

Пользователи будут иметь возможность выйти.

Никакого восстановления пароля или изменения пароля, и тому подобных функций не требуется.

PW

Система позволяет пользователям выполнять следующие операции.

1. Постоянно наблюдать ваше имя и текущий PW баланс на экране
2. Создавать новую сделку. Чтобы сделать новую транзакцию (PW платеж) пользователь должен
   1. Выбрать получателя по средствам запроса к списку пользователей по имени (автокомплит)
   2. Когда пользователь выбран, ввести количество PW для сделки. Система проверит, что объем сделки не выше чем текущий баланс пользователя.
   3. Свершение сделки. Как только транзакция завершается успехом, аккаунт получателя будет приписан (PW++) в соответствии с введенным количеством PW, а акаунт плательщика плательщика будет дебитирован (PW--) соответствующим размером PW. Системе следует отобразить PW изменение баланса пользователю мгновенно.
3. (По желанию) Создать новую сделку, как копию из листа тех существующих сделок: создать удобное UI для пользователя, чтобы просматривать их текущие сделки и выбирать сделку, как основу для новой сделки. Как только старая транзакция выбрана, все её детали (получатель, количество PW) ,будут скопированы в новую сделку.
4. Просмотр списка (истории) таких сделок. Список транзакций покажет наиболее последние транзакции в верхней части этого списка и отобразит следующую информацию для каждой сделки:
   1. Дата/Время сделки
   2. Имя корреспондента
   3. Размер транзакции (Дебит/Кредит для PW переноса средств)
   4. Итоговый баланс

(По желанию) Реализовать фильтрацию и/или сортировку списка сделок по дате в соответствии с именем и объёмом сделки.

**Требования архитектуры.**

1. WEB API приложение, на стороне сервера

WEB API должно реализовывать сервис на REST архитектуке.

1. HTML frontend приложение
2. Мобильное приложение

HTML frontend и мобильное приложение реализует полную функциональность и общается с WEB API используя JSON.

**Технические требования**

Используйте следующее программное обеспечение:

1. Серверная сторона
   1. MS SQL Server 2012 or higher
   2. IIS
   3. Web API
   4. Entity framework 6
2. WEB HTML клиент
   1. HTML5
   2. Javascript or TypeScript
   3. AngularJS or ReactJS
   4. (optional) primeNG, material, bootstrap
3. Мобильное приложение
   1. iOs or Android (at your choice, or both platforms)
   2. C# with Xamarin development environment

**Требования к дизайну пользовательского интрфейса**

Заказчик любит дизайн Material приблизительно так же как родное Mac OS ощущение от внешнего восприятия.

Обязательно придумать и реализовать уникальное ощущение от пользовательского интерфейса для frontend приложений для каждого UI элементов и (по желанию) применить его в frontend приложении.