**Выделение Задач из Тестового задания**

**Оценку затрат времени в таком стиле делаю впервые, исходил из разных соображений, насколько я понимаю важно фиксировать время чтобы частично симулировать рабочий процесс. Первая оценка - оптимистичная, вторая оценка - средняя, третья оценка - пессимистичная.**

**1.Главная страница (заглушка) Оценка затрат времени: 1**

**2.Пользователю доступна авторизация через данные Google-почты**

**2.1 дивайз гем и его конфигурирование = модель юзера + авторизация Оценка затрат времени: 1 2 3**

**2.2 Google OAuth2 установка и конфигурирование = возможность зайти используя данные Gmail почты. Оценка затрат времени: 3 6 9**

(могут возникнуть затруднения, исходя из документации это очень гибкая билиотека и нужно будет копаться в "мясе")

**2.3 Убираем всё лишнее от дивайз гема, должна остаться только возможность зайти используя данные Gmail почты Оценка затрат времени: 1 2 3**

(насколько я понял встроенной авторизации (devise) в приложении быть не должно? Должна быть только авторизация через данные гугл почты?)

**3.Создание лейаута приложения = базовый лейаут, шапка + кнопка входа - выхода Оценка затрат времени: 1 2 3**

**4.Создание системы сохранения новой почты из базы данных гугла в бд приложения после входа пользователя**

(я думаю будет лучше если новая почта сразу сохранится в бд после захода пользователя, чтобы не показывать ему пустую страницу, а затем уже можно будет обновлять кнопкой)

**4.1 Установка и конфигурирование gmailgem (могут возникнуть трудности) Оценка затрат времени: 3 6 9**

(в справке есть другой гем по google api но я документации хорошей по нему не нашел)

**4.2 Создание и конфигурирование модели incoming\_message.rb Оценка затрат времени: 1 2 3**

**4.3 Создание и конфигурирование модели attachment Оценка затрат времени: 1 2 3**

**4.4 Добавление ассоциаций между моделями user, incoming\_message, attachment Оценка затрат времени: 1 2 3**

**4.5 Настройка нестед ресурсес роутотов Оценка затрат времени: 1 2 3**

**4.6 Создаём контроллер с действием которое сохранит новую почту с гмейл базы данных в базу данных приложения Оценка затрат времени: 2 4 6**

(пока трудно оценить что там будет точно, и будет ли это отдельным контроллером, но action точно будет кастомным)

**4.7 Создаём Колбек в контроллере для почты который запросит все новые письма**

**и сохранит сразу как пользователь зайдёт в приложение Оценка затрат времени: 2 4 6**

(4.2 - 4.7 уже спроектировал модели, эти задачи могут занять 1-2 часа если всё пройдёт гладко, сохранение файлов и список адресатов в виде массива буду делать впервые)

**5.Пользователь может просматривать список сообщений (которые ранее были сохранены в бд) Оценка затрат времени: 2 4 6**

**6.Минимальный стайлинг страницы списка сообщений Оценка затрат времени: 1**

**7.Список сообщений пользователя отображается с пагинацией 50 на странице Оценка затрат времени: 1 2 3**

**8.Пользователь может просматривать одно сообщение (которое ранее было сохранено в бд) Оценка затрат времени: 2 4 6**

**9.Добавление ссылок "назад" Оценка затрат времени: 1**

**10.Добавление еррорс в виде партиалов для представлений Оценка затрат времени: 1 2 3**

**11.Минимальный стайлинг страницы показа одного сообщения Оценка затрат времени: 1**

**12.Переход на страницы сообщения со страницы списка сообщений Оценка затрат времени: 1 2 3**

**13.Кнопка Refresh которая забирает новую почту из API на странице Списка сообщений и сохраняет в базу данных Оценка затрат времени: 2 4 6**

(здесь может возникнуть трудность с использованием методов для google api, не было опыта раньше, если получится создать метод то здесь вопрос только в кнопке)

**14.Установка двуязычности и установка переключателя локали на главной Оценка затрат времени: 2 4 6 (суммарно на все подзадачи)**

**14.1 Составление английской локали**

**14.2 Составление русской локали**

**14.3 Редактирование представлений**

**15.Тесты для модели user.rb Оценка затрат времени: 2 4 6**

**16.Тесты для модели incoming\_message.rb Оценка затрат времени: 2 4 6**

**17.Тесты для модели attachment.rb Оценка затрат времени: 2 4 6**

**18.Тесты для контроллера users\_controller.rb Оценка затрат времени: 2 4 6**

**19.Тесты для контроллера incoming\_messages\_controller.rb Оценка затрат времени: 2 4 6**

**20.Тесты для контроллера attachments\_controller.rb Оценка затрат времени: 2 4 6**

**21.Тесты для контроллера static\_pages.rb Оценка затрат времени: 2 4 6**

**(есть шаблоны по тестам, могут возникнуть трудности тестов только для контроллеров + возможно где-то я пропустил контроллер, Agile подход =)).**

**Оптимистичная - 46 (1 неделя), средняя - 88 (2 недели). пессимистичная - 134 (3 недели)**