

CRED-2200 SA. ALI. Отправка детализации платежа на email

Содержание

- [Содержание](#)
- [Ссылки](#)
- [Описание задачи](#)
 - [Цели](#)
- [Целевые процессы](#)
- [Функциональные требования](#)
- [Требования к UI](#)
- [Содержание файла Детализация платежа](#)
 - [Новый метод](#)

Ссылки

Epic: [CRED-2197 - Getting issue details...](#)

Analysis task: [CRED-2200 - Getting issue details...](#)

Документация сервиса Sender: [POST /send/email](#)

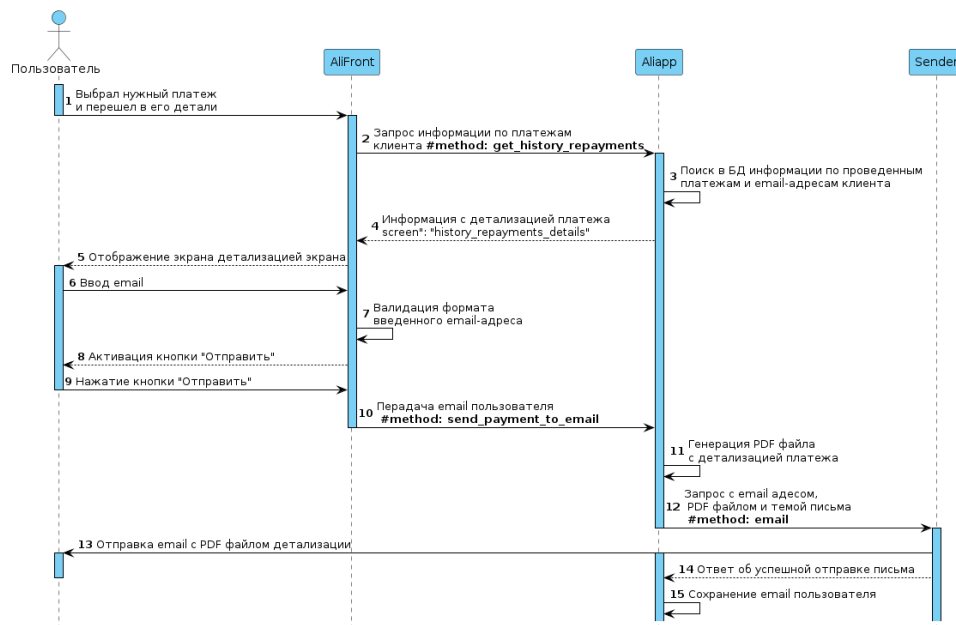
Описание задачи

Цели

Предоставить клиентам продукта Рассрочка возможность отправлять email с файлом детализации платежа.

Целевые процессы

Description	Content	Task
CRED.UC-137 Отправка детализации платежа на email	Акторы <ul style="list-style-type: none">• Пользователь• Ali-front• Ali-app• Sender - сервис отправки email DBS Предусловия <ul style="list-style-type: none">• У пользователя была в прошлом или есть на текущий момент действующая рассрочка;• У пользователя есть ≥1 успешное погашение;• Пользователь вошел в ЛК рассрочки; Триггер <p>Пользователь открыл раздел "История оплат" в ЛК Рассрочки.</p> Схема	CRED-2200 - G issue details... <input type="button" value="STATUS"/>



Сценарий

1. Пользователь переходит на экран платежа;
2. Ali-front запрашивает у Ali-app информацию по платежам клиента через метод `get_history_repayments`;
3. Ali-app ищет в БД информацию по проведенным платежам и проверяет наличие сохраненного ранее email клиента;
4. Ali-app направляет в Ali-front ответ с экраном `screen: "history_repayments_details"`;
5. + Ali-front отображает клиенту экран с детализацией платежа. Input поле "Почта" пустое, для пользователей, которые не отправляли файл с детализацией на email ранее;
6. Пользователь вводит свой email адрес в input поле; [CRED.FR-139](#)
7. + Ali-front успешно валидирует формат введенного email-адреса согласно [CRED.FR-143](#) ;
8. Ali-front снимает блокировку с кнопки "Отправить"
9. Пользователь запрашивает отправку файла с детализацией на email, нажав на кнопку "Отправить";
10. Ali-front передает в Ali-app email пользователя из input поля "Почта" используя метод `send_payment_to_email`;
11. Ali-app генерирует PDF файл с детализацией платежа; [CRED.FR-140](#)
12. Ali-app отправляет запрос в Sender, включающий в себя PDF файл с детализацией, email адресом и темой письма, используя метод `#email`;
13. + Сервис Sender отправляет email клиенту;
14. После получения пользователем email-а, Sender отправляет в Ali-app ответ об успешной отправке email-а;
15. Ali-app сохраняет email адрес пользователя; [CRED.FR-141](#)

Расширения

5a. Ali-front отображает клиенту экран с детализацией платежа. Пользователь ранее направлял файл с детализацией на свой email;

1. В input поле предзаполнен последний успешно валидированный и отправленный пользователем email адрес, который доступен для редактирования;
2. Переход к шагу №7;

7a. Ali-front оставляет кнопку "Отправить" заблокированной из-за неуспешной валидации формата email адреса;

1. Пользователь редактирует введенный email адрес согласно валидируемым стандартам;
2. Переход к шагу №7;

13a. Ali-app получает в ответ от Sender код ошибки 500 (Сервис не отвечает).

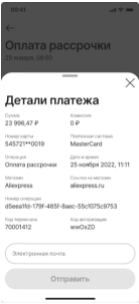
1. Ali-app направляет в Ali-front запрос с ошибкой 500 и текстом "Не смогли отправить детали платежа, сервис не доступен";
2. Ali-front изменяет состояние input поля и отображает текст ошибки который передал Ali-app.
3. Ali-front отображает на экране текст ошибки [CRED.UI-056](#)
4. Завершение сценария.

Функциональные требования

Description	Task
<p>CRED.FR-139</p> <p>Ali-front должен предзаполнять в input поле последний использованный пользователем email адрес согласно CRED.FR-141</p>	<p>CRED-2200 - Getting issue details...</p> <p>STATUS</p>
<p>CRED.FR-140</p> <p>Ali-app должен генерировать PDF файл с детализацией платежа (см. таблицу Содержание файла Детализация платежа). После получения ответа от сервиса Sender об успешной отправке email-а, PDF файл должен удаляться.</p>	<p>CRED-2200 - Getting issue details...</p> <p>STATUS</p>
<p>CRED.FR-141</p> <p>Ali-app должен сохранять и перезаписывать успешно использованный пользователем email адрес в колонку user_email согласно #send_payment_to_email.</p>	<p>CRED-2200 - Getting issue details...</p> <p>STATUS</p>
<p>CRED.FR-142</p> <p>Ali-front должен передавать в Ali-app введенный пользователем email адрес используя метод #send_payment_to_email.</p>	<p>CRED-2200 - Getting issue details...</p> <p>STATUS</p>
<p>CRED.FR-143</p> <p>Ali-front должен валидировать формат введенного пользователем email адреса в формате xxxx@yyyy.zz. Кнопка "Отправить" доступна для нажатия при успешном результате валидации.</p>	<p>CRED-2200 - Getting issue details...</p> <p>STATUS</p>

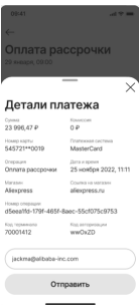
Требования к UI

Макеты [здесь](#).

Description	Task
<p>CRED.UI-054</p> <p>Первая попытка пользователя отправить детализацию на email</p> 	

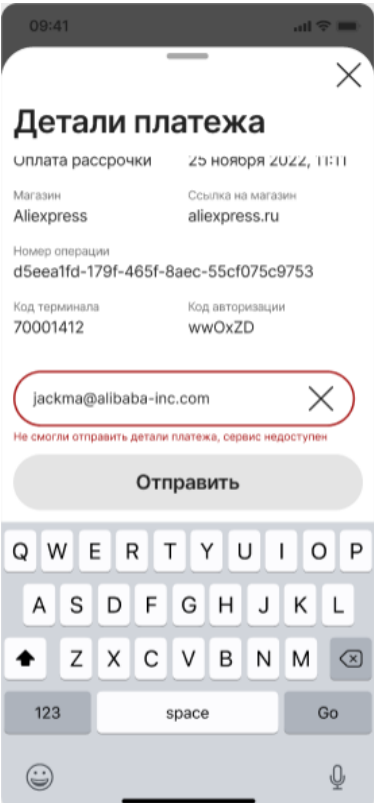
CRED.UI-055

Состояние с предзаполненным email-ом в input поле



CRED.UI-056

Состояние с ошибкой "Сервис не доступен"



CRED.UI-057

Состояние с загрузкой запроса



Содержание файла Детализация платежа

Сумма	Комиссия
-------	----------

X	Y
Номер карты	Платежная система
Маскированный PAN	Название МПС
Операция	Дата и время
Оплата рассрочки	дд месяц гггг, чч:мм
Магазин	Ссылка на магазин
Aliexpress	aliexpress.ru
Номер операции	
uuid	
Код терминала	Код авторизации
70001412	wwOxZD

Новый метод

Запрос раздела Выбор карты

Запрос				
Поле		Обязательность	Тип	Описание
method		Да	string	Название запрашиваемого метода: payment_to_email
params		Да		
	user_email	Да	string	email пользователя
	operation_id	Да	string	ID услуги
Ответ - Ок (200)				