**Имя документа:**

Техническое задание на разработку системы Разнесение Безналичных Платежей от Корпоративных Клиентов

**Название проекта:**

Noncash Payments

**Обозначение проекта:**

NCP

**Заказчик:**

Зураб Галумашвили

**Версия документа:**

1.0

**Автор документа:**

Р. Вологин

**Дата создания:**

12.06.2015

**Дата изменения:**

14.07.2015

**Цель документа:**

В данном документе описано техническое задание на разработку системы Разнесение безналичных платежей от Корпоративных клиентов. Приведены требования к программе, требования к программной документации, стадии разработки, порядок контроля и приемки.

Оглавление

[1. Обозначения 4](#_Toc424824764)

[2. Назначение и цели создания системы 4](#_Toc424824765)

[3. Концепция Системы автоматизации разнесения БНП 4](#_Toc424824766)

[Сущности БД 4](#_Toc424824767)

[Программное обеспечение – модули Системы 5](#_Toc424824768)

[Бизнес-процессы в рамках проекта 5](#_Toc424824769)

[NPC OB – Получение информации о платежах 5](#_Toc424824770)

[NPC AM – Работа с учётной записью Клиента 6](#_Toc424824771)

[NPC Client – Создание Реестра Клиентом 6](#_Toc424824772)

[NPC Client – Создание Реестра Региональным специалистом 6](#_Toc424824773)

[4. Требования к системе 6](#_Toc424824774)

[Требования к системному окружению 6](#_Toc424824775)

[Требования к языковой поддержке: 7](#_Toc424824776)

[Требования к формату файла-источника данных 7](#_Toc424824777)

[Требования к информационной и программной совместимости 7](#_Toc424824778)

[4.1. Требования к функциональным характеристикам NPC AM 7](#_Toc424824779)

[4.2.1. Требования к данным системы безопасности 7](#_Toc424824780)

[4.2.2. Требования к пользовательскому интерфейсу системы безопасности 8](#_Toc424824781)

[4.2.3. Процедура создания нового пользователя 9](#_Toc424824782)

[4.2.4. Вход в NCP Client 10](#_Toc424824783)

[4.1.4.1. JBoss Security Realm 10](#_Toc424824784)

[4.1.4.2. Страница Логина 10](#_Toc424824785)

[4.1.4.3. Страница восстановления пароля 11](#_Toc424824786)

[4.1.4.4. Смена пароля по желанию пользователя 11](#_Toc424824787)

[4.2. Требования к функциональным характеристикам NPC Client 11](#_Toc424824788)

[4.2.1. Просмотр списка клиентов 11](#_Toc424824789)

[4.2.2. Профиль корпоративного клиента 13](#_Toc424824790)

[4.2.3. Просмотр списка шаблонов клиента 15](#_Toc424824791)

[4.2.4. Страница просмотра/создания/редактирования шаблона 16](#_Toc424824792)

[4.2.5. Просмотр списка платежей клиента 17](#_Toc424824793)

[4.2.6. Страница разнесения платежа 19](#_Toc424824794)

[5. Стадии и этапы разработки 23](#_Toc424824795)

[6. Требования к документированию 23](#_Toc424824796)

[7. Порядок контроля и приема и системы 23](#_Toc424824797)

[8. Вопросы 23](#_Toc424824798)

# Обозначения

**Система** - Система Загрузка безналичных платежей от Корпоративных клиентов в Беркут

**Клиент** – Корпоративный клиент

**БНП** – Безналичный платёж

**СРК** – Специалист по работе с клиентами

**СВР** – Специалист по взаиморасчетам

**Перечень номеров** – список MSISDN, либо лицевых счетов с указанием сумм для пополнения. Список может содержать только MSISDN или только лицевые счета. Вместе MSISDN и лицевые счета в одном списке не допускаются.

**Реестр** – основная сущность Системы, включающая идентификацию Клиента, сумму платежа и перечень номеров.

**ККБ** – Казкоммерцбанк

**БК** – Банк Клиент – программное обеспечение ККБ, установленное в Фин. Блоке Компании

**БД** – База Данных Компании.

# **Назначение и цели создания системы**

Объемы загружаемых по безналичному расчету платежей постоянно растут, процесс по загрузке действующий на данный, момент не позволяет выполнять быструю обработку, в виду задержек возникает недовольство наших корпоративных клиентов.

Система предназначена для автоматизации процесса разнесения безналичных платежей в биллинге.

Необходимо создать сервис, позволяющий Клиентам самостоятельно разносить платежи по своим номерам. В случае если Клиент не захочет выполнять эту процедуру самостоятельно, необходимо предусмотреть возможность разнесения платежа Региональным Специалистом от имени Клиента.

# **Концепция Системы автоматизации разнесения БНП**

## Сущности БД

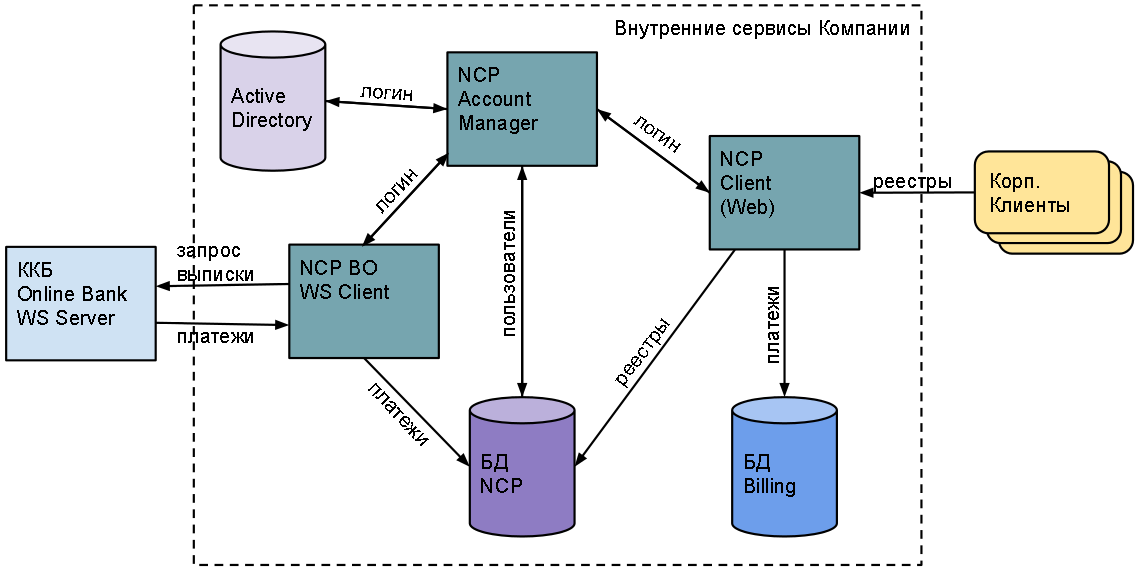
1. В БД создаётся сущность «Профиль Клиента». Профиль клиента в БД должен быть связан с профилем Юридического Лица в Биллинге. Атрибуты Профиля Клиента:
   * Название организации Клиента
   * ИИН
   * БИН
   * Контактное лицо
   * Контактный телефон
   * Email
   * Юридический адрес
   * Перечень номеров
   * Логин/пароль
2. В БД создаётся сущность «Платёж». Атрибуты Платежа:
   * Идентификатор
   * Дата/время создания платежа в БД
   * Дата/время совершения платежа клиентом
   * Клиент (БИН/ИИН или идентификатор)
   * № платежа
   * Сумма платежа
   * Агент – логин пользователя, разносившего платёж
   * Статус (не разнесён / разнесён / ошибка разнесения)
   * Перечень номеров. Номер представляет собой отдельную сущность – отдельную таблицу. Атрибуты номера:
     + MSISDN
     + Лицевой счёт
     + Сумма
     + Статус (не начислен/успех/ошибка)
3. В БД создаётся сущность «Шаблон платежа». Атрибуты Шаблона:
   * Идентификатор
   * Дата/время создания шаблона
   * Клиент (БИН/ИИН или идентификатор)
   * Перечень номеров. Номер представляет собой отдельную сущность – отдельную таблицу. Атрибуты номера:
     + MSISDN
     + Лицевой счёт
     + Сумма

## Программное обеспечение – модули Системы

Необходимо разработать следующие модули:

1. NCP Account Manager (NCP AM) - Управление учётными записями пользователей.
2. NCP Client – открытый в мир пользовательский интерфейс для самостоятельного разнесения безналичных платежей клиентами.
3. NCP Online Bank (NCP OB) – сервис интеграции с Казкоммерцбанком для оперативного получения информации о БНП.
4. Отчёт для сверки платежей.

Взаимодействие компонентов системы представлено на схеме:



## Бизнес-процессы в рамках проекта

### NPC OB – Получение информации о платежах

1. Выполняется опрос сервера на периодической основе.
2. Сервер присылает информацию о платежах, поступивших на счета Компании.
3. Платежи сохраняются в Базе Данных
4. Администратор со специальной web-страницы может принудительно выполнить опрос сервера за любую выбранную дату. В этом случае администратору должны быть показаны результаты выполнения запроса – все движения по счёту за указанную дату в виде таблицы. Платежи за выбранную дату должны быть сохранены в БД с отсечением дубликатов.

### NPC AM – Работа с учётной записью Клиента

Для того чтобы клиент или сотрудник Компании смог работать в Системе, для него должна быть заведена учётная запись в NCP AM.

Для клиентов должна быть выполнена привязка учётной записи к информации о Клиенте в Биллинге.

Для сотрудников компании должна быть выполнена проверка на корректность логина в LDAP.

Для региональных специалистов должна быть выполнена привязка к региону. Регион должен быть взят из биллинга. Для клиентов привязка к региону уже есть в биллинге.

Клиенту должны быть высланы его логин и пароль.

### NPC Client – Создание Реестра Клиентом

1. Клиент логинится в приложение NCP Client.
2. Клиент может просмотреть свои платежи-реестры (таблица с разбиением на страницы).
   * Платежи отсортированы в порядке убывания даты платежа
   * Для платежа указан его статус (не разнесён / разнесён / ошибка разнесения)
3. Клиент находит в списке платежей свой недавний платёж (без реестра) и создаёт для него реестр
4. При создании реестра клиент может заполнить перечень номеров из заранее заданного шаблона
5. Реестр может быть сохранён только в случае если сумма платежа совпадает с итоговой суммой по номерам
6. В случае успешного сохранения реестра, посылается команда на разнесение платежа в Биллинг.
7. Клиенту выдаётся сообщение о результате выполнения операции
8. Клиенту на почту высылается уведомление о результате выполнения операции

### NPC Client – Создание Реестра Региональным специалистом

1. Специалист логинится в приложение NCP Client
2. Специалист просматривает платежи от конкретного клиента
3. Далее алгоритм действий аналогичен описанному выше.

# **Требования к системе**

## Требования к системному окружению

Веб браузер

* Chrome
* Firefox
* IE(опционально)

## Требования к языковой поддержке:

Система должна поддерживать следующие языки

* Русский (по умолчанию)
* Казахский (опционально)

## Требования к формату файла-источника данных

Система принимает следующие форматы данных для заполнения списка абонентов

* Excel(с заданной структурой)
* Текстовый документ

## Требования к информационной и программной совместимости

Программа должна быть разработана на языке Java, база данных Oracle. Серверная часть должна представлять собой web приложение. Приложение должно разворачиваться под управлением web сервера JBoss EAP.

## **Требования к функциональным характеристикам NPC AM**

Модуль NCP AM представляет собой подсистему безопасности. Модуль должен обеспечить разграничение доступа к функциям приложения.

### Требования к данным системы безопасности

В базе данных должны быть предусмотрены таблицы под:

* **Учётные записи пользователей**
  + Идентификатор – уникальное числовое значение
  + Дата начала периода валидности учётной записи
  + Дата конца периода валидности учётной записи
  + Тип пользователя (Клиент / региональный специалист / специалист по взаиморасчётам / Администратор)
  + Флаги
    - Email подтверждён
    - MSISDN подтверждён
    - Согласован
    - Заблокирован
    - Сменить пароль при следующем входе
  + Логин – для клиентов содержит то же значение, как поле Email;
  + Email;
  + MSISDN;
  + Наименование (Наименовение Компании Клиента либо ФИО сотрудника Tele2)
  + Идентификатор Клиента – связь с информацией о клиенте в биллинге;
  + Reference to IT Event – номер IT Event, на основании которого был заведён пользователь.
* **Пароли пользователей**
  + Дата начала периода валидности пароля
  + Дата окончания периода валидности пароля
  + Идентификатор пользователя, кому принадлежит пароль
  + Пароль в зашифрованном виде
* **Действия пользователей в системе** – таблица для хранения любых действий, выполняемых пользователем в системе.
  + Дата, когда было выполнено действие;
  + Логин пользователя;
  + Идентификатор операции;
  + Текстовое поле что делал;

Для всех сущностей у актуальной версии дата конца валидности должна быть выставлена в 01.01.2200.

Никакие из сущностей (пользователи, логи) не должны удаляться из системы в течение 3х лет (настраиваемый параметр).

На сервере приложений JBoss EAP 6.3 должен быть настроен Realm системы безопасности, работающий с перечисленными выше таблицами.

В системе должен быть создан пользователь «administrator», который должен иметь доступ ко всем функциям системы безопасности.

Остальные пользователи, роли и доступы могут быть созданы с помощью этого пользователя.

Пользователь «administrator» не может быть удалён и заблокирован.

### Требования к пользовательскому интерфейсу системы безопасности

Должен быть разработан интерфейс пользователя, позволяющий выполнять:

* **Просмотр списка зарегистрированных в системе пользователей** – таблица, каждая строка которой представляет одного пользователя. Клик по строке – переход на форму редактирования пользователя.
  + Дата создания пользователя
  + Дата закрытия пользователя
  + Наименование
  + Логин
  + Тип пользователя (клиент / региональный сотрудник / администратор / специалист по взаиморасчётам)
* **Редактирование учётной записи пользователя** – форма, позволяющая просматривать и редактировать информацию о пользователе:
  + Идентификатор – нередактируемое поле – однозначно идентифицирует одного пользователя
  + Тип пользователя
  + Фамилия Имя пользователя.
  + Логин
  + Пароль пользователя – для клиентов обязательный, для сотрудника Tele2 не обязательный.

Правила создания пароля:

* + - допускаются только латинские буквы и цифры
    - содержит минимум одну букву в нижнем регистре, одну букву в верхнем регистре, одну цифру.
    - минимальная длина 8 символов
    - максимальная длина 20 символов
    - пароль чувствителен к регистру
    - срок валидности пароля 90 дней (настраиваемый параметр)
    - пароль не должен повторять 3 предыдущих пароля (настраиваемый параметр)
  + Email – является логином при входе в систему.
    - не допускается существование двух пользователей с одинаковыми Email
  + MSISDN.
  + Флаг «Сменить пароль при следующем входе». Флаг выставляется вручную. Если флаг выставлен, пользователь должен будет сменить пароль при попытке логина в приложение. При смене пароля флаг должен сбрасываться автоматически.
  + Флаг «Заблокирован». Флаг выставляется вручную. При выставлении данного флага пользователь не должен иметь возможности пользоваться каким-либо функционалом системы. Флаг доступен только администраторам.
  + Флаг «Email подтверждён». Флаг выставляется автоматически при подтверждении пользователем email. Изменение состояния флага вручную недоступно.
  + Флаг «MSISDN подтверждён». Флаг выставляется автоматически при подтверждении пользователем MSISDN. Изменение состояния флага вручную недоступно.
  + Кнопка «Закрыть учётную запись» - по нажатию на кнопку, текущая дата/время записывается в поле «Дата конца периода валидности». Пользователь считается удалённым.
  + Кнопка «Восстановить» - доступна администратору в режиме просмотра исторических данных. По нажатию на кнопку, информация учётной записи на просматриваемую дату становится актуальной на текущую дату. Пользователю на почту автоматически высылается уведомление о внесённых изменениях.

Для клиентов (Тип пользователя) должно быть предусмотрено поле, связывающее учётную запись с информацией о корпоративном клиенте в биллинге.

Для региональных специалистов должно быть предусмотрено поле «Регион». Значение должно выбираться из перечня регионов биллинга.

На странице приложения должен быть предусмотрен элемент управления, позволяющий просмотреть состояние базы данных на любой момент времени в прошлом – календарь. При выборе в календаре определённой даты/времени, на странице должны отображаться данные на выбранную дату.

Должна быть предусмотрена кнопка возврата к просмотру актуальных данных.

Пользовательский интерфейс системы безопасности должен быть защищён специальным доступом.

Все действия пользователя по редактированию учётных записей, ролей и доступов должны логироваться.

Отчёты по логам должны быть реализованы в системе **Reporting**.

### Процедура создания нового пользователя

Новый пользователь может быть создан только в пользовательском интерфейсе системы безопасности.

Доступ к интерфейсу должны иметь администраторы.

Администраторы могут видеть абсолютно всех пользователей, заведённых в системе.

При создании у нового пользователя по умолчанию должен быть выставлен флаг «Сменить пароль при следующем входе».

После создания пользователю на почту автоматически отправляется письмо – уведомление о том, что он может приступать к работе. В письме должны содержаться логин, пароль и ссылка на страницу подтверждения email.

### Вход в NCP Client

Вход в NCP Client должен быть доступен только для зарегистрированных в NCP AM пользователей.

### JBoss Security Realm

Для **аутентификации** и **авторизации** пользователей должен быть разработан специальный Security Realm.

При **аутентификации** Realm должен по логину определять тип пользователя. В случае если пользователь является сотрудником Компании (специалист или администратор), необходимо выполнять аутентификацию через LDAP. Если пользователь является клиентом, пароль должен сверяться с паролем в БД.

**Авторизация** должна выполняться в зависимости от типа пользователя:

* Клиент – возвращать роль «client»;
* Региональный Специалист – возвращать роль «regional»;
* Специалист по взаиморасчётам – возвращать роль «financial»;
* Администратор – возвращать роль «administrator»;

### Страница Логина

При входе пользователь должен попадать на страницу ввода логина и пароля.

На ввод логина/пароля должно даваться 5 попыток (настраиваемый параметр).

Если пользователь за 5 попыток не смог ввести корректный пароль, но вводил корректный логин (логин существующего неудалённого пользователя), у пользователя должен автоматически выставляться флаг «Заблокирован».

Если пользователь ввёл логин и пароль учётной записи с выставленным флагом «Заблокирован», пользователю должно выдаваться сообщение о том, что он заблокирован и ему следует обратиться к региональному специалисту.

Если пользователь ввёл логин и пароль учётной записи, у которой истёк срок валидности пароля, он должен быть перенаправлен на страницу смены пароля.

Блокировка пользователя и истечение срока валидности не должно касаться сотрудников Tele2.

### Страница восстановления пароля

На странице логина должна быть предусмотрена ссылка на страницу восстановления пароля.

Восстановление пароля должно быть реализовано только для клиентов.

На странице восстановления пароля должна быть реализована форма, в которой пользователь должен ввести свой логин (email).

После нажатия на кнопку «Восстановить», должен выполниться поиск по базе данных существующих неудалённых пользователей, и если было найдено совпадение логина, то на email пользователя должно быть выслано письмо со ссылкой на страницу смены пароля.

### Смена пароля по желанию пользователя

В NCP Client должна быть предусмотрена страница смены пароля.

Смена пароля должна быть реализована только для клиентов.

На странице должны быть предусмотрены поля для ввода старого пароля, нового пароля, подтверждения нового пароля и ввод проверочного кода (капча).

При смене пароля должны действовать правила, описанные в п. 4.1.2.

Все действия пользователя по входу, смене и восстановлению пароля, должны логироваться.

## **Требования к функциональным характеристикам NPC Client**

Модуль NCP Client представляет собой первый вариант портала Tele2 для корпоративных клиентов.

В рамках текущего проекта должен быть реализован следующий функционал:

* Просмотр списка клиентов
* Просмотр профиля корпоративного клиента
  + Информация о корпоративном клиенте из Биллинга
  + Информация об учётной записи Клиента из NPC AM
* Просмотр списка шаблонов реестров
* Просмотр и редактирование шаблонов реестров
* Просмотр списка платежей клиента
  + Просмотр разнесённых платежей
  + Просмотр новых не разнесённых платежей
* Просмотр и разнесение платежа

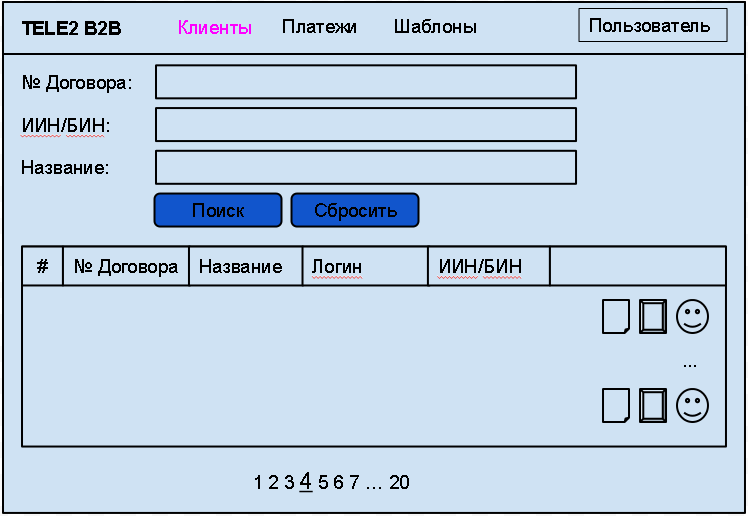
### Просмотр списка клиентов

В модуле NCP Client должна быть разработана страница, отображающая список всех зарегистрированных в системе Клиентов.

Страница должна быть доступна только для сотрудников Компании.

Для администратора должен отображаться полный список корпоративных клиентов. Администратор должен иметь возможность выбрать регион.

Для региональных сотрудников должен отображаться список корпоративных клиентов соответствующего региона.



На странице в табличном виде должен отображаться список всех клиентов, зарегистрированных в NCP AM.

Список должен быть разбит на страницы по N клиентов на странице.

Столбцы таблицы:

* ID клиента – порядковый номер, присвоенный при регистрации в NCP AM
* № Договора
* Название организации
* Логин
* ИИН/БИН
* Кнопки действий по клиенту
  + Кнопка просмотра платежей клиента
  + Кнопка просмотра шаблонов клиента
  + Кнопка просмотра профиля клиента

Список по умолчанию должен быть отсортирован по названию по алфавиту.

Заголовки таблицы должны быть кликабельные. По клику на заголовке список клиентов должен сортироваться по соответствующему столбцу.

На странице должен быть реализован поиск клиентов по полям

* № Договора
* ИИН/БИН
* Название организации

По нажатию кнопки «Поиск» в таблице должны отображаться результаты поиска.

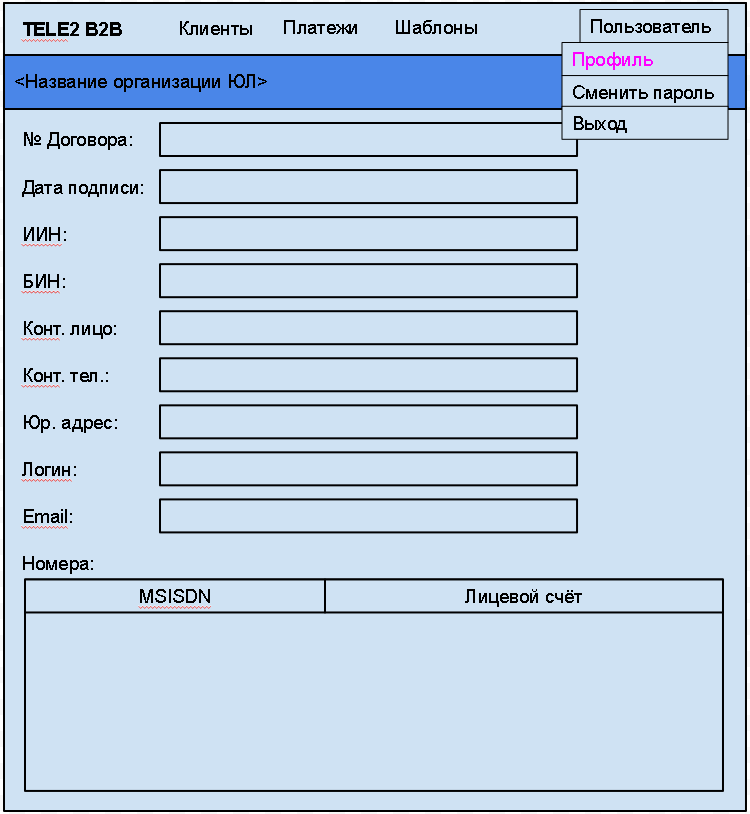
По нажатию кнопки «Сбросить» в таблице должен отображаться полный список клиентов. Поля поиска должны очищаться.

### Профиль корпоративного клиента

В модуле NCP Client должна быть разработана страница, отображающая информацию о корпоративном клиенте.

Пользователи-клиенты должны иметь доступ на просмотр собственного профиля.

Пользователи – сотрудники Tele2 должны иметь доступ на просмотр профиля любого корпоративного клиента.



Страница профиля клиента должна позволять просматривать следующую информацию:

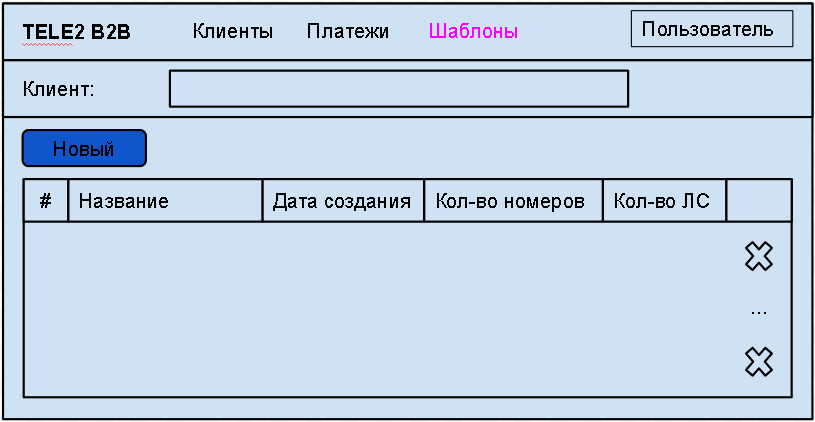
* Название организации юридического лица
* Номер договора
* Дата подписи
* ИИН
* БИН
* Контактное лицо
* Контактный телефон
* Юридический адрес
  + Страна
  + Город
  + Улица
  + Почтовый индекс
  + Дом
  + № Квартиры
* Логин
* Email
* Номера GSM. Если номеров у корпоративного клиента много, список должен быть разбит на страницы.

На странице должна быть предусмотрена ссылка на страницу смены пароля (только для самого пользователя).

### Просмотр списка шаблонов клиента

В модуле NCP Client должна быть разработана web-страница, отображающая шаблоны клиента.

Шаблоны должны отображаться на странице в табличном виде отсортированные по названию в алфавитном порядке.



Столбцы таблицы:

* ID Шаблона
* Название
* Дата создания
* Кол-во номеров
* Кол-во лицевых счетов
* Кнопка удаления шаблона

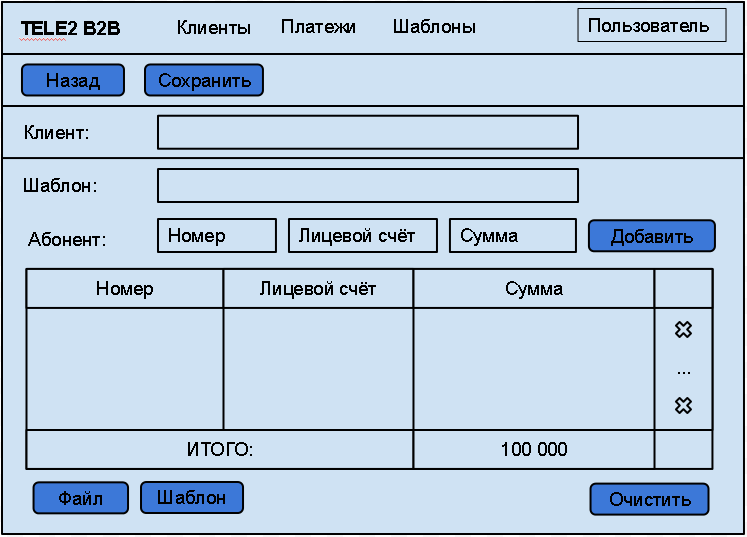
Шаблоны в таблице должны быть кликабельные. По клику на шаблоне должен осуществляться переход на страницу просмотра/редактирования шаблона.

На странице должна быть предусмотрена кнопка «Новый». По нажатию на кнопку должен осуществляться переход на страницу просмотра/редактирования шаблона для создания нового шаблона.

### Страница просмотра/создания/редактирования шаблона

В модуле NCP Client должна быть разработана web-страница, позволяющая просматривать, создавать и редактировать шаблон.

Страница доступна всем.



На странице должно быть предусмотрено поля редактирования названия шаблона.

На странице должны быть предусмотрены элементы управления для формирования списка номеров.

Формирование списка номеров должно быть реализовано аналогично описанному в п. 4.2.6.

Сохранить шаблон можно по нажатию на кнопку «Сохранить».

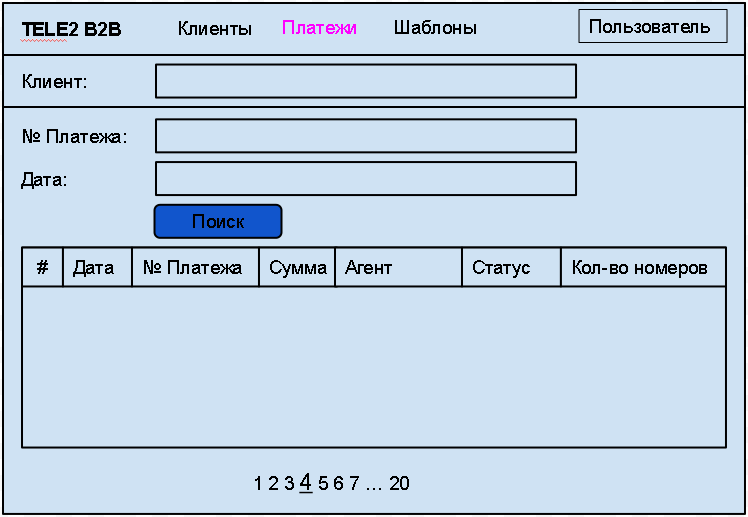
Шаблон может быть сохранён только в случае если у шаблона заполнено поле «Название» и список номеров не пустой.

Номера должны соответствовать формату 7071234567.

### Просмотр списка платежей клиента

В модуле NCP Client должна быть разработана web-страница, отображающая платежи клиента.

Платежи должны отображаться на странице в табличном виде отсортированные в порядке убывания даты.



Должен быть реализован поиск платежа по номеру и по дате.

Должен быть предусмотрен фильтр по статусу (не разнесён / разнесён / ошибочный).

По платежу в таблице должна отображаться информация:

* Идентификатор платежа
* Дата совершения платежа
* № платежа
* Сумма платежа
* Агент – логин пользователя, разносившего платёж
* Статус (не разнесён / разнесён / ошибка разнесения)
* Количество номеров

Платежи в таблице должны быть кликабельные.

По клику мышкой на разнесённом платеже должен осуществляться переход на форму разнесения платежа без возможности изменения полей и выполнения активных действий по платежу.

По клику мышкой на не разнесённом платеже должен осуществляться переход на форму разнесения платежа для ввода перечня номеров и выполнения разнесения.

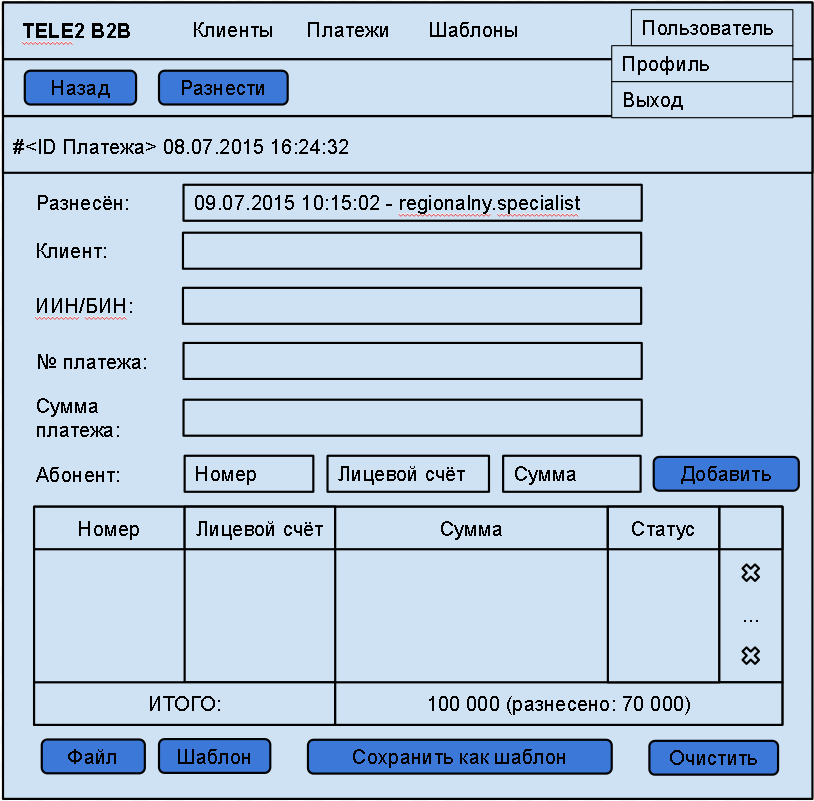
По клику мышкой платеже в статусе «ошибка разнесения» должен осуществляться переход на форму разнесения платежа. На форме должна отображаться детальная информация, поясняющая причины ошибки. Список номеров не может быть отредактирован. Кнопка «Разнести» должна быть доступна. В случае если платёж был разнесён частично (на некоторые номера были начислены средства, на некоторые нет), должна быть предусмотрена возможность начислить средства на оставшиеся номера – доразнести платёж.

### Страница разнесения платежа

Страница разнесения платежа должна быть доступна любому пользователю.

Разграничение доступа происходит на предыдущем шаге – на странице просмотра платежей.

Страница разнесения платежа должна выглядеть следующим образом:



Редактирование и разнесение доступно только для новых неразнесённых платежей и платежей в статусе «ошибка разнесения».

В случае если платёж находится в статусе «ошибка разнесения», на форме должна отображаться детализированная информация об ошибке.

Для редактирования доступен только список номеров.

Добавление номеров в список можно сделать следующими способами:

1. Добавление вручную по одному номеру:
   * Ввести информацию в поля «Номер», «Лицевой счёт», «Сумма»
   * Нажать кнопку «Добавить»
   * Номер добавляется в список.
2. Добавление списка номеров из файла:
   * Нажать кнопку «Файл»
   * Должен открыться диалог для выбора файла. Доступны файлы Excel или текстовые файлы.
   * Файл должен содержать список номеров по одному номеру в строке в формате «Номер» - «Сумма» либо «Лицевой счёт» - «Сумма»
   * Информация из файла должна добавиться в список номеров.
3. Добавление списка номеров из шаблона:
   * Нажать кнопку «Шаблон»
   * Должен открыться диалог выбора шаблона
   * В диалоге выбрать нужный шаблон и нажать кнопку «Ok»
   * Информация из шаблона должна добавиться в список номеров.

При добавлении номеров в список (равно как и лицевых счетов), Система должна проверять номера на принадлежность корпоративному клиенту.

В последней строке таблицы в графе «ИТОГО» должна отображаться итоговая сумма по номерам и сумма по номерам, для которых разнесение прошло успешно.

Дублирование номеров в списке допускается. Каждая строка в списке означает начисление средств либо на номер, либо на лицевой счёт.

В одной строке должен присутствовать либо номер, либо лицевой счёт. Одновременное присутствие и номера и лицевого счёта недопустимо. Сумма в строке должна быть проставлена обязательно.

В последней колонке списка номеров для каждого номера должна быть предусмотрена кнопка удаления номера из списка. Удаление номера выполняется без подтверждения. При удалении номера, информация о номере копируется в поля «Номер», «ЛС», «Сумма».

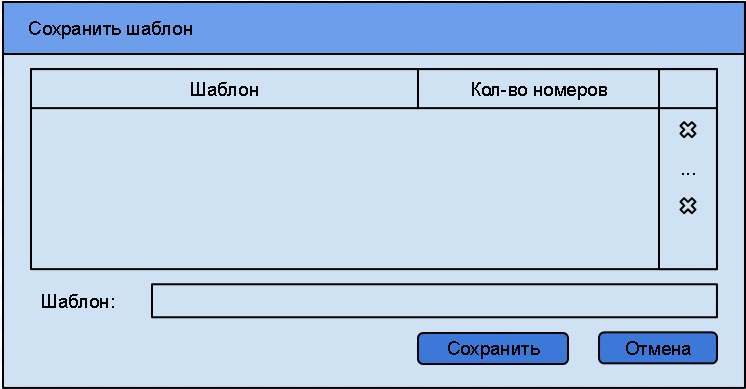
Можно удалить все номера из списка нажав кнопку «Очистить».

Номера, по которым суммы были успешно разнесены, не могут быть удалены из списка.

Можно сохранить сформированный список номеров как шаблон. Для этого необходимо нажать кнопку «Сохранить как шаблон». Сохранение списка номеров в виде шаблона должно быть доступно всегда независимо от статуса платежа. Пустой список сохранять как шаблон нельзя.

По нажатию на кнопку «Сохранить как шаблон»:

* Открывается диалог сохранения шаблона:



* В диалоге отображается список всех шаблонов текущего клиента
* В диалоге отображается поле ввода наименования шаблона
* В диалоге должна быть предусмотрена возможность удаления шаблонов. Удаление шаблонов должно выполняться без подтверждения.
* Пользователь может выбрать любой шаблон из списка. В этом случае имя шаблона отображается в поле «Шаблон»
* Пользователь может ввести новое имя шаблона
* Пользователь нажимает кнопку «Сохранить»
* Если пользователь выбрал существующий шаблон, должно появиться подтверждение на перезапись существующего шаблона.
* Если имя шаблона не пересекается с существующими шаблонами, список номеров сохраняется как новый шаблон.

Проверки на форме

1. Список номеров не может быть пустым
2. При добавлении номеров в таблицу они проверяются на валидность. Номер должен быть в формате 7071234567. Префикс (707) может быть префиксом любого мобильного оператора РК (707, 747, 701, 777, …).
3. Система должна проверять на валидность лицевые счета

На странице разнесения платежа должна отображаться история платежа.

В истории должны отображаться действия, выполненные по платежу:

* Приём платежа от ККБ
* Разнесение платежа с отображением результата разнесения
* Доразнесение платежа после ошибки

Для каждого действия должна отображаться информация:

* Дата/время
* Автор
* Вид действия (приём/разнесение/доразнесение)
* Примечания

#### Разнесение платежа

Разнесение платежа выполняется по кнопке «Разнести».

Платёж может быть разнесён если список номеров не пуст и сумма средств по номерам равняется сумме платежа.

По нажатию на кнопку «Разнести» вызывается Berkut API начисление средств для каждого номера.

Для каждого номера прописывается результат начисления средств (статус).

Если для какого-то номера начисление средств завершилось неуспешно, помимо выставления статуса для соответствующего номера, в истории платежа должно отобразиться подробное описание, поясняющее неуспех операции.

Платёж считается успешно разнесён (статус), если на все номера были успешно начислены средства.

Результат разнесения платежа необходимо занести в историю платежа.

Пользователю должно быть выдано сообщение о результате.

Клиенту на почту должно быть выслано уведомление на почту по результатам разнесения платежа.

Если платёж был разнесён с ошибками, уведомление должно высылаться так же и на email региональных специалистов соответствующего региона.

В тексте письма должна присутствовать прямая ссылка на платёж.

## **Требования к функциональным характеристикам NPC OB**

Модуль NCP OB представляет собой web сервис клиент, который подключается к web сервису Online Bank Казкоммерцбанка.

Приложение должно периодически запрашивать у сервера информацию по платежам, поступившим на счета компании

* IBAN №KZ029261802133052000 – основной
* IBAN №KZ189261802133052003 – для абонентов

На основании информации, получаемой от Online Bank, приложение должно создавать в Базе Данных платежи.

В приложении должен быть реализован пользовательский web интерфейс, обеспечивающий:

* Настройку периодичности запросов к серверной части
* Остановку / запуск периодического опроса сервера
* Контроль работоспособности модуля
* Просмотр исключительных ситуаций, возникающий в процессе работы сервиса
* Выполнение запросов на сервер по команде пользователя за произвольный период времени.

Доступ к пользовательскому интерфейсу должны иметь только администраторы системы.

## **Требования к Сверке платежей**

В рамках проекта должен быть разработан отчёт для сверки платежей.

Ежедневно отправлять отчёт на Специалиста по Взаиморасчётам.

# **Стадии и этапы разработки**

В таблице 1 приведены этапы разработки и примерное время, необходимое для их реализации. Время указано в рабочих часах.

**Таблица 1.**

|  |  |  |
| --- | --- | --- |
| **№ п.п.** | **Наименование этапа** | **Время, ч** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  | **Итого разработка** |  |
|  | **Итого документация** |  |

# **Требования к документированию**

Необходимо в wiki.tele2 создать статьи:

* Руководство пользователя;
* Описание программы (JavaDoc)

# **Порядок контроля и приема и системы**