

Программное обеспечение  
для визуализации контрольных лент в  
электронной форме.

Руководство по установке и использованию.

ТОВ «ИКС-Маркет»

2015

Программное обеспечение для визуализации контрольных лент в электронной форме (далее в контексте документа принято сокращение КЛЭФ) предназначено для преобразования из внутреннего формата хранения данных РРО в удобную и понятную электронную интерпретацию в виде самодокументируемого формата xml, а также дополнительную визуальную интерпретацию этих данных по аналогии с бумажным носителем.

Программное обеспечение работает под управлением операционных систем Windows и Linux.

## 1. Необходимые требования:

Операционная система Windows XP (Service Pack 3) или выше.

Операционная система семейства Linux: например Ubuntu 14.04 LTS.

Независимо от выбора типа операционной системы на ПК должна быть установлена среда исполнения Java приложений: Java Runtime Environment (JRE). Скачать и установить её можно с сайта компании «**ORACLE**»:

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

## 2. Поставка продукта.

В поставку входят службы созданные на основе приложений “Procrun” от Apache Commons™: <http://commons.apache.org/proper/commons-daemon/procrun.html>

- для ОС Windows:

### 1. Клиентская часть

Имя файла (каталога)	Описание
KsefSvc.jar	Клиентская служба
config.ini	Файл конфигурации клиентской службы
prunsrv.exe	Приложение “procrun” от Apache Commons™
amd64	Каталог с приложением “procrun” для архитектуры процессоров AMD-64
ia64	Каталог с приложением “procrun” для архитектуры процессоров Intel Itanium.
install.bat	batch – файл для установки службы
start.bat	batch – файл для запуска службы
stop.bat	batch – файл остановки службы
uninstall.bat	batch – файл удаления службы

## 2. Серверная часть

Имя файла (каталога)	Описание
<b>KSEFServer.jar</b>	Серверная служба
<b>server.ini</b>	Файл конфигурации серверной службы
<b>prunsrv.exe</b>	Приложение “procrun” от Apache Commons™
<b>amd64</b>	Каталог с приложением “procrun” для архитектуры процессоров AMD-64
<b>ia64</b>	Каталог с приложением “procrun” для архитектуры процессоров Intel Itanium.
<b>install.bat</b>	batch – файл для установки службы
<b>start.bat</b>	batch – файл для запуска службы
<b>stop.bat</b>	batch – файл остановки службы
<b>uninstall.bat</b>	batch – файл удаления службы

- для ОС семейства Linux:

### 1. Клиентская часть

Имя файла (каталога)	Описание
<b>KsefDaemon.jar</b>	Клиентская служба
<b>lib</b>	Каталог с библиотекой Commons Daemon от Apache Commons™
<b>config.ini</b>	Файл конфигурации клиентского приложения
<b>IN</b>	Каталог для исходных двоичных файлов КЛЭФ.
<b>BACKUP</b>	Каталог для резервного хранения двоичных файлов КЛЭФ
<b>OUT</b>	Каталог для преобразованных в xml-формат файлов КЛЭФ.
<b>KsefDaemon.sh</b>	sh – скрипт клиентского приложения

## 2. Серверная часть

Имя файла (каталога)	Описание
<b>KsefSrvNix.jar</b>	Серверное приложение
<b>lib</b>	Каталог с библиотекой Commons Daemon от Apache Commons™
<b>server.ini</b>	Файл конфигурации серверного приложения

<b>KSEFSTORAGE</b>	Каталог для хранения всех данных КЛЭФ в xml-формате.
<b>KsefSrvNix.sh</b>	sh – скрипт серверного приложения

- ПО для визуализации контрольных лент:

Имя файла (каталога)	Описание
<b>KsefView.jar</b>	Приложение визуализации контрольных лент КЛЭФ.

### 3. Описание принципа работы.

Процесс работы служб построен таким образом, чтобы после однократной их настройки и запуска остальной процесс осуществлялся автоматически без участия и контроля со стороны человека.

Клиентские службы устанавливаются непосредственно на ПК, на которых установлен драйвер РРО. Каждая установленная и успешно запущенная клиентская служба раз в несколько секунд просматривает каталог исходных двоичных данных КЛЭФ (по умолчанию каталог "IN"), сформированных драйвером РРО по команде управляющего ПО. Если в каталоге есть файл с расширением ".ksf", то проверяется его формат и целостность.

Далее производится преобразование двоичного файла КЛЭФ в самодокументируемый формат xml с тем же именем. Преобразованные данные в xml-формате записываются в выходной каталог данных. После преобразования исходный двоичный файл переписывается в каталог резервного хранения. Если клиентская служба была дополнительно сконфигурирована на передачу преобразованной информации на сервер, то дополнительно предпринимается попытка подключения к серверу и передачи на сервер преобразованного файла КЛЭФ.

Серверные службы устанавливаются только в случае необходимости централизованного хранения данных КЛЭФ на файловом сервере. Этот вариант великолепно подходит для торговых предприятий от двух и более кассовых мест.

Серверная служба отвечает за приём преобразованных данных КЛЭФ от кассовых мест и сохранение данных на файловом сервере КЛЭФ в соответствующем каталоге хранилища. При подключении клиентской службы и её готовности передать новые данные, сервер принимает подключение, создаёт в хранилище КЛЭФ (KSEFSTORAGE) новый каталог с серийным номером РРО или идентификатором кассового места POS\_ID, если такого ещё не существует и записывает туда принятый файл.

Дополнительно установив программу визуализации КЛЭФ "KSEFView" на кассовое место или на файловый сервер, можно просматривать, находить нужную информацию в чеках и отчётах, а также распечатывать их.

**Внимание!** В программе визуализации КЛЭФ чеки и отчёты представлены с максимальным соответствием оригиналу документа, но не является таковым. Визуальная копия КЛЭФ не является базой для доказательств в юридических спорах, а лишь помогает торговому предприятию с поиском нужной информации за прошедшие в прошлом периоды работы, а также даёт представление об объёме и содержимом информации, которая передаётся на сервер ДПА.

#### 4. Установка и настройка.

##### ОС Windows:

##### 4.1. Клиентская служба Windows

Переписываем в любой из каталогов на системном разделе содержимое подкаталога "client" в каталоге "Windows". В зависимости от архитектуры процессора (AMD 64 или Intel Itanium) переписываем файл "prunsrv.exe" из соответствующего подкаталога и располагаем рядом с файлом "KsefSvc.jar".

Редактируем конфигурацию файла "config.ini" (кодировка ansi).

IN\_DATA\_STRING\_VALUE = C:\\IN

OUT\_DATA\_STRING\_VALUE = C:\\OUT

BACKUP\_DATA\_STRING\_VALUE = C:\\BACKUP

DATA\_TRANSMITTER\_ENABLED = false

SERVER\_IP = localhost

SERVER\_PORT = 2188

POS\_ID = Касса №1

Параметр	Описание	Значение по умолчанию
IN_DATA_STRING_VALUE	Путь к данным с исходными двоичными данными КЛЭФ.	C:\\IN
OUT_DATA_STRING_VALUE	Путь для экспорта преобразованных в xml-формат данных КЛЭФ.	C:\\OUT
BACKUP_DATA_STRING_VALUE	Путь к резервному хранилищу двоичных данных КЛЭФ	C:\\BACKUP
DATA_TRANSMITTER_ENABLED	Параметр, отвечающий за передачу преобразованных данных на сервер ( <b>false</b> – не передавать/ <b>true</b> – передать на сервер)	<b>false</b>

<b>SERVER_IP</b>	IP-адрес или имя хоста сервера	<b>localhost</b>
<b>SERVER_PORT</b>	порт сервера	<b>2188</b>
<b>POS_ID</b>	Идентификатор кассового места	<b>Касса №1</b>

Сохраняем сделанные изменения в файле “config.ini”.

В batch-файле “install.bat” с помощью текстового редактора можно изменить каталог журнала операций клиентской службы. В частности найдите в файле приведенные ниже строки и при необходимости измените их, потом сохраните эти изменения:

```
SET PR_LOGPATH=c:\logs\client
```

```
SET PR_STDOUTPUT=c:\logs\client\stdout.txt
```

```
SET PR_STDERROR=c:\logs\client\stderr.txt
```

Запустите файл “install.bat”. Служба должна установиться. Проверить установку службы можно через меню “Пуск → Панель Управления → Система и безопасность → Администрирование → Службы”. В списке служб должна появиться новая служба с именем “KSEF Service”.

Чтобы запустить службу запустите файл “start.bat” или запустите эту службу прямо из списка служб. В журнале будет сделана запись о запуске службы или ошибке во время её запуска. В окне списка служб нажимаем кнопку «F5» и, если статус службы “KSEF Service” в состоянии «Работает», то служба была успешно запущена и продолжает работать.

#### 4.2. Серверная служба Windows.

Переписываем в любой из каталогов на системном разделе содержимое подкаталога “server” в каталоге “Windows”. В зависимости от архитектуры процессора (AMD 64 или Intel Itanium) переписываем файл “prunsrv.exe” из соответствующего подкаталога и располагаем рядом с файлом “KSEFServer.jar”.

Редактируем конфигурацию файла “server.ini” (кодировка ansi).

```
LISTEN_PORT = 2188
```

```
STORAGE_PATH = C:\\
```

Параметр	Описание	Значение по умолчанию
<b>LISTEN_PORT</b>	TCP- порт сервера для подключения клиентов.	<b>2188</b>
<b>STORAGE_PATH</b>	Каталог расположения хранилища KSEFSTORAGE	<b>C:\\</b>

Сохраняем сделанные изменения в файле "server.ini".

В batch-файле "install.bat" с помощью текстового редактора можно изменить каталог журнала операций серверной службы. В частности найдите в файле приведенные ниже строки и при необходимости измените их, потом сохраните эти изменения:

```
SET PR_LOGPATH=c:\logs\server
```

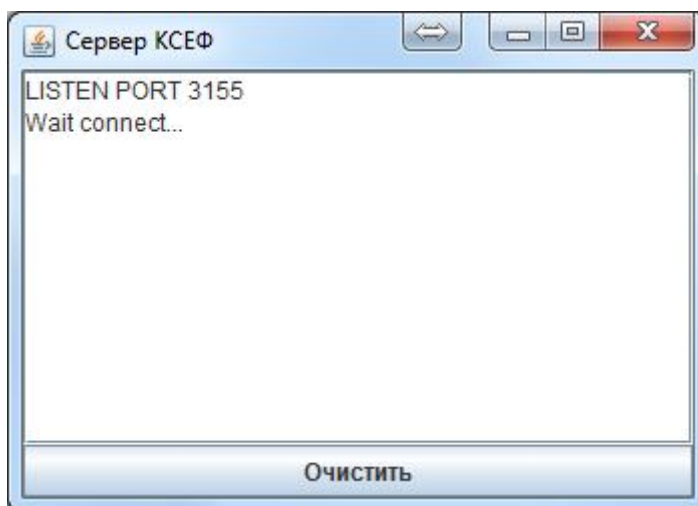
```
SET PR_STDOUTPUT=c:\logs\server\srvout.txt
```

```
SET PR_STDERROR=c:\logs\server\srverr.txt
```

Запустите файл "install.bat". Служба должна установиться. Проверить установку службы можно через меню "Пуск → Панель Управления → Система и безопасность → Администрирование → Службы". В списке служб должна появиться новая служба с именем "KSEF Server".

Чтобы запустить службу запустите файл "start.bat" или запустите эту службу прямо из списка служб. В журнале будет сделана запись о запуске службы или ошибке во время её запуска. В окне списка служб нажимаем кнопку «F5» и, если статус службы "KSEF Server" в состоянии «Работает», то служба успешно была запущена и продолжает работать.

Для отладки сервер можно просто запустить как обычное приложение. Для этого нужно просто запустить приложение "KSEFServer.jar" на выполнение.



**Рис.1. Приложение KSEFServer.jar**

**ОС Linux (Ubuntu 14.04 LTS):**

#### **4.3. Клиентская служба (демон) Linux**

Переписываем в домашнюю папку пользователя содержимое подкаталога "client" в каталоге "Linux"

Создаём из ресурсов исполняемый файл "jsvc.jar". Детальная инструкция по созданию описана тут:

<http://commons.apache.org/proper/commons-daemon/jsvc.html>

1. Скачиваем по адресу [http://commons.apache.org/proper/commons-daemon/download\\_daemon.cgi](http://commons.apache.org/proper/commons-daemon/download_daemon.cgi) архив исходников [commons-daemon-1.0.15-src.tar.gz](http://commons.apache.org/proper/commons-daemon/download_daemon.cgi)
2. Разархивируем скачанный архив:  

```
tar xvfz commons-daemon-native.tar.gz
```
3. Переходим в каталог unix:  

```
cd commons-daemon-1.0.15src/src/native/unix
```
4. Выполняем скрипт конфигурирования.  

```
./configure
```
5. Компилируем командой “make”  

```
make
```

В результате вы должны получить исполняемый файл “jsvc.jar”

Рекомендуется переписать “jsvc.jar” в каталог /usr/local/bin. Если используется другой каталог, то нужно исправить путь к нему в файле скрипта “KsefDaemon.sh”

Далее необходимо установить Java Development Kit (JDK). Руководство по установке можно посмотреть по этой ссылке:

<http://docs.oracle.com/javase/7/docs/webnotes/install/linux/linux-jdk.html#install-64>

После установки JDK, необходимо проверить переменную среды окружения JAVA\_HOME. Она должна указывать на каталог с установленным JDK. Это можно сделать с помощью команды:

```
echo $JAVA_HOME
```

Если JAVA\_HOME не указывает на каталог JDK, то нужно выполнить команду:

```
export JAVA_HOME = jdk-install-dir
```

где ***jdk-install-dir*** – каталог установки JDK.

После этого нужно внести соответствующие изменения в sh-скрипт.

В текстовом редакторе “Gedit” или любом другом, открываем файл sh - скрипта “KsefDaemon.sh” и изменяем переменные на свои, из перечня в следующей таблице:

Переменная	Описание	Значение по умолчанию
EXEC	Переменная, указывающая на JSVC.	<i>/usr/local/bin/jsvc</i>
JAVA_HOME	Переменная среды окружения, должна указывать на каталог с JDK.	<i>/usr/java/jdk1.7.0_71</i>
CLASS_HOME	Переменная, указывающая путь к классу клиентского приложения.	<i>"/home/user/Daemon"</i>

<b>CLASS_PATH</b>	Переменная для подключения класса клиентского приложения.	"\$CLASS_HOME/lib/commons-daemon-1.0.15.jar": "\$CLASS_HOME/KsefDaemon.jar"
<b>CLASS</b>	Имя класса клиентского приложения	KSEFSvcPackage.MainClass
<b>OPTIONS</b>	Опции для указания адреса хоста серверного приложения. Используются, если параметров SERVER_IP и SERVER_PORT нет в файле конфигурации "config.ini".	"--ip=localhost --port=2188"
<b>USER</b>	Пользователь, от имени которого будет запущено клиентское приложение.	user
<b>PID</b>	Файл с "process ids" запущенного демона.	/var/run/KsefDaemon.pid
<b>LOG_OUT</b>	Файл журнала вывода информационных сообщений клиентского приложения.	\$CLASS_HOME/daemon.log
<b>LOG_ERR</b>	Файл журнала информации об ошибках клиентского приложения.	\$CLASS_HOME/daemon.err
<b>PROCNAME</b>	Имя процесса	KsefDaemon

Сохраняем сделанные изменения в файле скрипта.

Редактируем конфигурацию файла "config.ini" (кодировка utf8).

IN\_DATA\_STRING\_VALUE = /home/user/Daemon/IN

OUT\_DATA\_STRING\_VALUE = /home/user/Daemon/OUT

BACKUP\_DATA\_STRING\_VALUE = /home/user/Daemon/BACKUP

DATA\_TRANSMITTER\_ENABLED = true

SERVER\_IP = localhost

SERVER\_PORT = 3155

POS\_ID = KACCA №1

Параметр	Описание	Значение по умолчанию
<b>IN_DATA_STRING_VALUE</b>	Путь к данным с исходными двоичными данными КЛЭФ.	/home/user/Daemon/IN

<b>OUT_DATA_STRING_VALUE</b>	Путь для экспорта преобразованных в xml-формат данных КЛЭФ.	<b>/home/user/Daemon/OUT</b>
<b>BACKUP_DATA_STRING_VALUE</b>	Путь к резервному хранилищу двоичных данных КЛЭФ	<b>/home/user/Daemon/BACKUP</b>
<b>DATA_TRANSMITTER_ENABLED</b>	Параметр, отвечающий за передачу преобразованных данных на сервер ( <b>false</b> – не передавать/ <b>true</b> – передать на сервер)	<b>false</b>
<b>SERVER_IP</b>	IP-адрес или имя хоста сервера	<b>localhost</b>
<b>SERVER_PORT</b>	TCP - порт сервера	<b>2188</b>
<b>POS_ID</b>	Идентификатор кассового места	<b>КАССА №1</b>

Сохраняем сделанные изменения в файле “config.ini”.

Настало время запустить клиентский демон. Для этого сначала переходим в каталог с скриптом “KsefDaemon.sh”:

```
cd home/<user>
```

Запускаем скрипт на выполнение:

```
KsefDaemon.sh start
```

В каталоге “/home/<user>/Daemon” вы должны обнаружить новый файл “daemon.log”. Если его открыть, то в последней строке должна быть запись «Daemon started...». Если так и есть, то демон клиентского приложения успешно установлен и запущен в работу. Если этого не случилось, проверьте наличие файла “daemon.err” в том же каталоге с описанием ошибки.

Для остановки клиентского демона выполняйте команду “stop”:

```
KsefDaemon.sh stop
```

Для перезапуска клиентского демона выполняйте команду “restart”:

```
KsefDaemon.sh restart
```

#### 4.4. Серверная служба (демон) Linux

Переписываем в домашнюю папку пользователя содержимое подкаталога “server” в каталоге “Linux”

Выполняем операции по установке JRE, JDK, конфигурирование и компиляцию JSVC, а также проверку переменной среды окружения JAVA\_HOME описанные в предыдущем подразделе (4.3), если серверное приложение устанавливается на отдельный ПК.

В текстовом редакторе “Gedit” или любом другом, открываем файл sh - скрипта “KsefSrvNix.sh” и изменяем переменные на свои, из перечня в следующей таблице:

Переменная	Описание	Значение по умолчанию
EXEC	Переменная, указывающая на JSVC.	/usr/local/bin/jsvc
JAVA_HOME	Переменная среды окружения, должна указывать на каталог с JDK.	/usr/java/jdk1.7.0_71
CLASS_HOME	Переменная, указывающая путь к классу серверного приложения.	"/home/user/Server "
CLASS_PATH	Переменная для подключения класса серверного приложения.	"\$CLASS_HOME/lib/commons-daemon-1.0.15.jar": "\$CLASS_HOME/KsefSrvNix.jar"
CLASS	Имя класса серверного приложения	KsefSrvPackage.MainClass
OPTIONS	Опции для указания порта подключения к серверу. И включения режима отладки.	"--port=2188 --debug"
USER	Пользователь, от имени которого будет запущено серверное приложение.	user
PID	Файл с “process ids” запущенного демона.	/var/run/KsefSrvNix.pid
LOG_OUT	Файл журнала вывода информационных сообщений серверного приложения.	\$CLASS_HOME/srv.log
LOG_ERR	Файл журнала информации об ошибках серверного приложения.	\$CLASS_HOME/srv.err
PROCNAME	Имя процесса	KsefSrvNix

Сохраняем сделанные изменения в файле скрипта.

Редактируем конфигурацию файла “server.ini” (кодировка utf8).

```
LISTEN_PORT = 2188
```

```
STORAGE_PATH = /home/user/Server
```

Параметр	Описание	Значение по умолчанию
<b>LISTEN_PORT</b>	TCP- порт сервера для подключения клиентов.	<b>2188</b>
<b>STORAGE_PATH</b>	Каталог расположения хранилища KSEFSTORAGE	<b>/home/user/Server</b>

Сохраняем сделанные изменения в файле “server.ini”.

Запускаем серверное приложение - демон. Для этого сначала переходим в каталог с скриптом “KsefDaemon.sh”:

```
cd home/<user>
```

Запускаем скрипт на выполнение:

```
KsefSrvNix.sh start
```

В каталоге “/home/<user>/” вы должны обнаружить новый файл “srv.log”. Если его открыть, то в последней строке должна быть запись «Server started...». Если так и есть, то демон серверного приложения успешно установлен и запущен в работу. Если этого не случилось, проверьте наличие файла “srv.err” в том же каталоге с описанием ошибки.

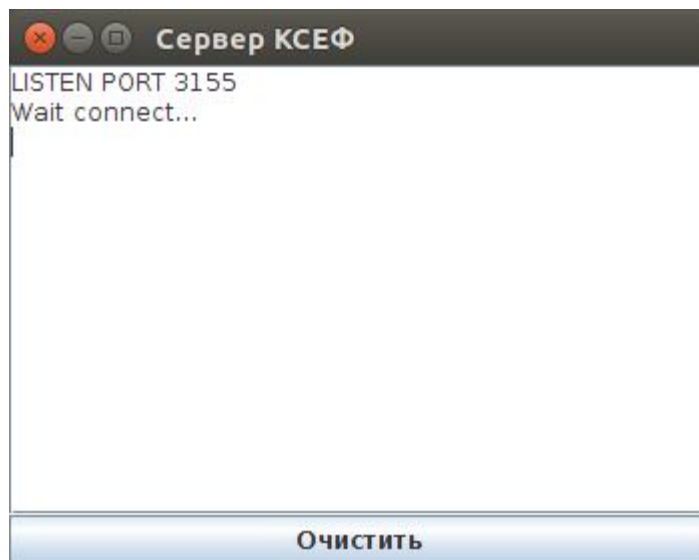
Для остановки серверного демона выполняйте команду “stop”:

```
KsefSrvNix.sh stop
```

Для перезапуска серверного демона выполняйте команду “restart”:

```
KsefSrvNix.sh restart
```

Если серверный sh- скрипт запустить с включенной опцией “debug”, то серверное приложение запустится с окном как на рис.2. В нём можно увидеть процесс подключения клиентских сервисов или демонов и результат передачи данных.



**Рис.2. Приложение KSEFServer.jar**

## **5. Приложение KSEFView.**

Приложение выполнено в виде исполняемого .jar - файла. Работает на ОС Windows и ОС Linux под управлением Java SE Runtime Environment (JRE) версии 7 и выше. Это приложение может быть установлено как на рабочее место кассира, где установлен драйвер РРО "ИКС-Маркет" выше версии 1.1.7, так и на отдельном файловом сервере, куда будут стекаться данные КЛЭФ со всех кассовых мест.

На верхней панели расположены кнопки с окном контекстного поиска. Кнопка «Хранилище» предназначена для выбора каталога хранилища с преобразованными данными КЛЭФ РРО клиентскими службами и демонами в xml-формат. Кнопка печать позволяет напечатать отображения чека на обычном бытовом принтере или распечатать его в pdf-файл. В поле поиска нужно вводить поисковый текст. Кнопка «Поиск» правее от поискового поля осуществляет контекстный поиск по всей контрольной ленте (отдельному файлу). Кнопка «Инфо» выводит информационное окно о приложении.

На левой панели отображается структура каталогов хранилища КЛЭФ.

Тут нужно выбрать нужный подкаталог кассы POS\_ID или подкаталог с серийным номером РРО. Потом нужно выбрать интересующий xml-файл с диапазоном пакетов, например MS80009971p420-440.xml или файлом смены, например MS80009971z60.xml.

В поле справа должен отобразиться список доступных пакетов выбранного файла КЛЭФ. Сразу над ним присутствует поле для быстрого поиска пакета по номеру. Этим удобно пользоваться при длинном списке пакетов. В этом списке ищем и выбираем интересующий номер пакета. Теперь, интересующий вас пакет должен быть построен и отображен на экране в центральной части программы «KSEFView». Всё очень просто. Далее эту построенную модель чека можно распечатать или сохранить в pdf-формат.

Внизу на информационной панели будет отображаться статусная информация по данному файлу КЛЭФ: модель и серийный номер РРО, версия внутреннего программного обеспечения РРО, POS\_ID и дата формирования отчёта.

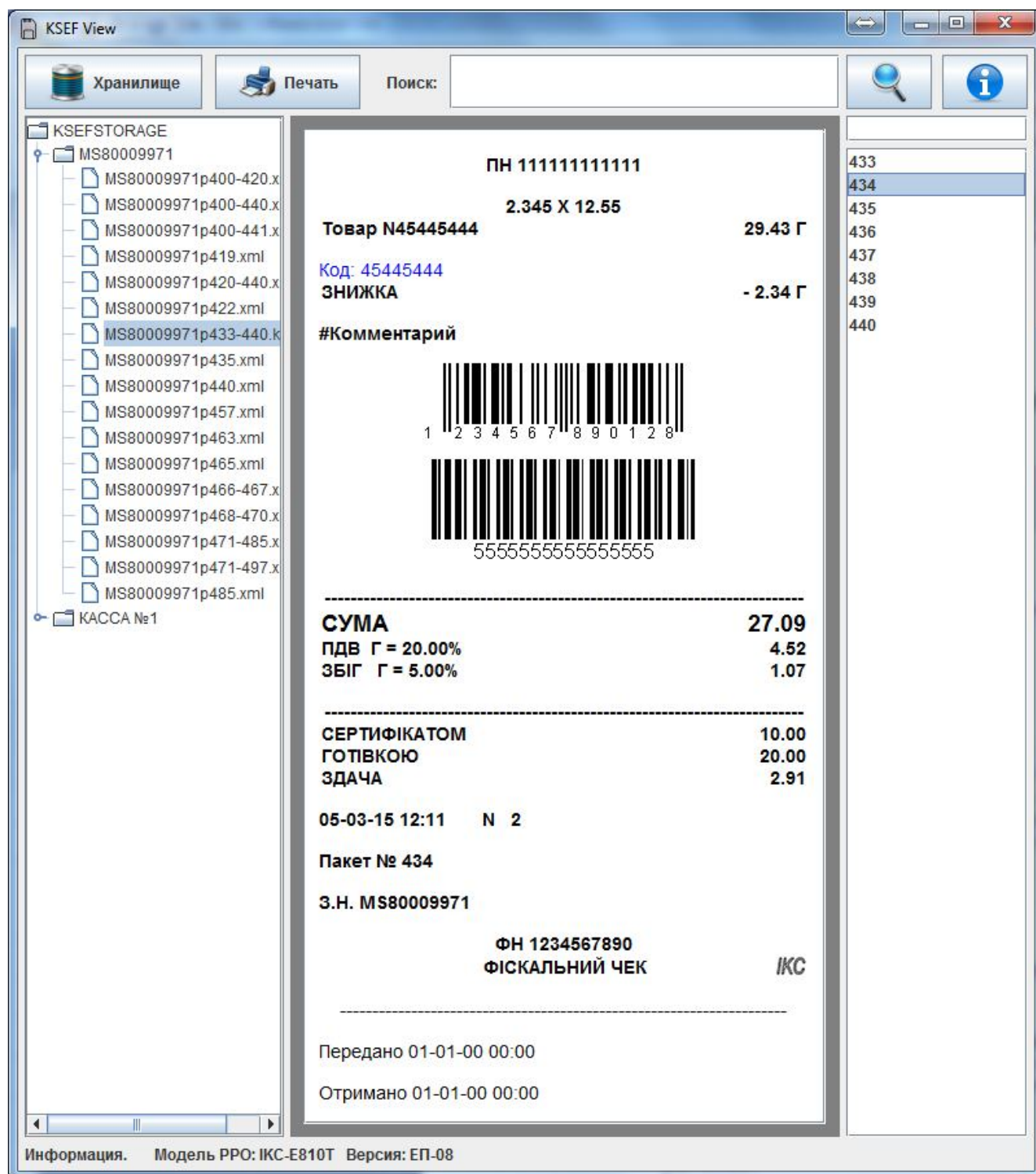


Рис.3. Приложение KSEFView.