

Latvijas Universitāte
Fizikas un matemātikas fakultāte
Datorikas nodaļa

JasperReports un iReport Designer lietojumi IS sistēmas izstrādē

Bakalaura darbs

Autors

Aleksandrs Dubrovskis

Stud. apl. Nr. DatZ020092

Vadītājs

Kalvis Apsītis

*analītiķis programmētājs, Accenture
Technology Solution, Dr. sc. comp.*

Rīga, 2006

Anotācija

Šis darbs ir veltīts JasperReports bibliotēkai un iReport Designer programmai. Šie produkti tiek izmantoti informatīvas sistēmas izstrādē Java vidē un ar to palīdzību var veidot jebkuras atskaites un dokumentus.

Darba tiek aprakstīti un izskatīti visi atskaites pamatjēdzieni, tiek salīdzināts tiešais darbs ar JasperReports un caur iReport Designer un piedāvāti risinājumi problēmām, kuras rādās izstrādes gaitā. Ņemot vērā to, ka šiem produktiem nav bezmaksas dokumentācijas, šo darbu var izmantot kā mācību līdzekli iesacījām.

Annotation

This thesis devoted to JasperReports library and iReport Designer. This two products are used in Java informative system development and using it developer is able to create any kind of reports and ready to print documents.

Thesis includes description of basic report definitions. Compared two ways of report development: directly through JasperReports and through iReport Designer. Are presented solutions to common problems which developer usually meets during the report development process. Due to these products don't have free documentation and then this thesis could be used as user guide for beginners.

Аннотация

Эта работа посвящена библиотеке JasperReports и программе iReportDesigner. Эти два продукта используются в разработке информационных систем для создания различных отчетов и готовых к распечатке документов.

В работе рассмотрены и описаны основные понятия отчетов, проведено сравнение работы с JasperReports напрямую и через iReport Designer и предложены решения проблем, которые возникают в процессе разработки. Принимая во внимание то, что у продуктов нет бесплатной документации, то данную работу можно использовать как учебное пособие для начинающих.

Atoreferāts

Darba galvenais mērķis ir JasperReports un iReport Designer pielietojuma pētīšana informatīvas sistēmas atskaites izstrādē un īsas loģiski pareizas mācību līdzekļa veidošana.

Darba gaitā bija izskatīts:

- atskaites dzīves cikls
- atskaites XML struktūra
- atskaites izsaušanas no Java aplikācijas
- izstrādes process tieši ar JasperReports bibliotēku bez kādas grafiskas redaktora palīdzības un caur iReport Designer
- visi atskaites pamatjēdzieni un tas atribūti
- problēmas, kuras rādījās reālas sistēmas izstrādes gaitā un piedāvāti iespējami risinājumi.

Saturs

1. <u>Ievads</u>	8
1.1 JasperReports	8
1.2 iReport Designer	9
2. <u>JasperReports pamatjēdzieni</u>	11
2.1 Atskaites dzīves cikls	11
2.2 JRXML izejas koda un Jasper faili	11
2.3 Datu avoti un drukas formāti	19
2.4 Izteiksmes	20
2.5 Atskaites izsaukšana Javā	21
3. <u>Atskaites struktūra</u>	23
3.1 Sekcijas	23
3.2 Atskaites opcijas	24
4. <u>Atskaites elementi</u>	26
4.1 Atskaites elementa definīcija	26
4.2 Elementa atribūti	26
4.3 Grafiski elementi	29
4.4 Teksta elementi	29
4.5 Apakšatskaites	30
5. <u>Lauki, parametri, mainīgie</u>	31
5.1 Datu glabāšanas objekti	31
5.2 Lauki	32
5.3 Mainīgie	33
5.4 Parametri	34
6. <u>Datu avoti</u>	34
6.1 Datu avoti	34
6.2 JDBC savienojums	35
6.3 JRDataSource interfeiss	36
6.4 JavaBean datu avots	36
6.5 XML datu avots	37
6.6 Tukšais datu avots	39
7. <u>Atskaites klase (skriptlets)</u>	39
7.1 JRAbstractScriptlet klase	39

8. <u>Risinājumi vispārējām problēmām</u>	40
8.1 Procentu apstrāde	40
8.2 Daudzlapaspušu atskaites	41
8.3 Hierarhiskas atskaites	44
8.4 Apakšatskaite un skriptlets	45
8.5 Lauku reģistrācija	45
9. <u>Noslēgums</u>	46

1. Ievads

1.1 JasperReports

JasperReports tā ir ļoti jaudīga Javas bibliotēka, kura parasti tiek izmantota informatīvas sistēmas izstrādē, lai veidot dažādus datu atskaites. Šis instruments ļauj piegādāt atskaites saturu tādos formātos kā PDF, HTML, RTF, XLS, CVS un XML.

Bibliotēka ir uzrakstīta pilnībā Javas valodā un varbūt izmantota Java aplikācijās, tai skaitā J2EE vai tīmekļa programmās, lai ģenerēt dinamisko saturu. Tā galvenais mērķis ir palīdzēt veidot dokumentus, kuri ir gatavi tiešai drukāšanai.

Kad veidojas atskaite JasperReports izmanto šablonus, kuri ir strukturēti sekcijās, piemēram, virsraksts, kopsavilkums, detalizēti dati un lapas vai grupas galvene un kājene. Katras sekcijas kārtojums ir brīvā formā, kur var ievietot vairākus un dažādus elementa tipus: statiskus un dinamiskus teksta laukus, attēlus, līnijas un taisnstūrus. JasperReports izmanto šo šablonu, lai organizētu datus XML failā (JRXML). Datus jāņem no kāda datu avota: vai no relāciju datu bāzes, vai no kolekcijas, vai tas varētu būt XML datu fails.

Lai aizpildīt atskaiti ar datiem, izstrādātajam pirmkārt jānokompilē XML šablonu. Kompilācijas laika notiek šablona validācija un pievieno pie tā kompilācijas informāciju. Šī informācija tiek izmantota lai novērtēt atskaites izteiksmes izpildes laiku. Nokompilētu atskaites šablonu var saglabāt uz diska vai nosūtīt to pa tīklu. Šis saglabātais objekts pēc tam tiek lietots, kad programma aizpilda atskaiti ar datiem. Faktiski kompilējot šablonu, kompilējas visas atskaites izteiksmes. Kompilācijas laika notiek dažādas pārbaudes, lai novērtēt atskaites dizaina secību. Rezultātā izstrādātājs saņem šablonu, kuru jau var aizpildīt ar datiem un noģenerēt dokumentu kādā pieļaujamā formātā.

Lai aizpildīt šablonu ar datiem, programmai ir jāsaņem atskaites datus. Šie dati iespējams saņemt dažādās formās. Daļa no tiem var būt padota kā atskaites parametrs, bet pamatdaļa nāc no datu avota. JasperReports var saņemt no datu avota speciālu objektus, kuri apgāda atskaiti ar informāciju vai ja dati atrodas relāciju datu bāzē, tad tas izmantos JDBC savienojumu.

Rezultātā parādās jauns objekts – dokuments, kurš jau ir var droši izdrukāt. Šo objektu tapī pati var saglabāt uz diska vai nosūtīt to pa tīklu. Rezultātu var apskatīt ar iebūvēto bibliotēkā skatītāju JasperReports viewer vai to var eksportēt kādā citā populārā formātā, kā PDF, HTML, RTF, XLS vai XML.

Visa JasperReports funkcionalitāte koncentrēta vienā JAR failā (jasperreports-x.x.x.jar), to var izmantot jebkura platformā, kura atbalsta Java. Bibliotēku var atrast <http://jasperreports.sourceforge.net/>. Lai pilnība varētu strādāt ar to vēl nepieciešami dažas lietas: XML parsētājs, PDF un XLS ģenerācija.

Kā jau bija pateikts JasperReports, tā ir bibliotēka, tātad lai to izmantot izstrādes procesā to nevajag instalēt. Vienkārši jāielādē tas JAR failu no interneta un nolikt to programmas classpath'ā.

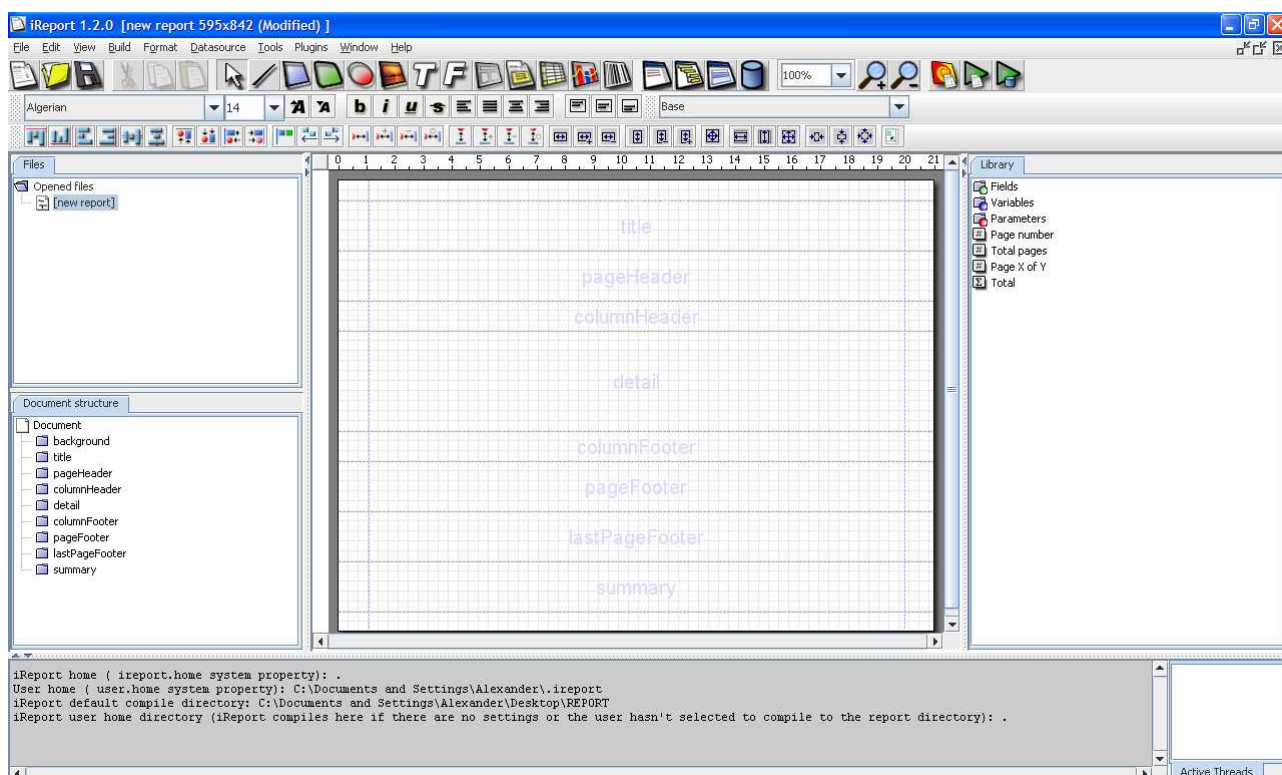
JasperReports atbild tikai par atskaitēm un par XML pārsēšanu, savienojumu ar datu bāzi, PDF, XLS izvadi citas bibliotēkas, piemēram: JAXP 1.1 XML Parser (XML), JDBC 2.0 Driver (savienojums), iText – free Java – PDF library(PDF), Jakarta Poi 2.0 (XLS).

Lai strādāt ar JasperReports nepieciešams Java JRE 1.3, vai augstāka versija, JDK 1.3, Eclipse JDT Java compiler, Jikes, Groovy, BeanShell.

Vēl būs nepieciešams Ant Build Tool kompilācijai, datu aizpildei un skatīšanai, arī to var izmantot lai nosūtīt rezultātu uz PDF vai HTML.

1.2 iReport Designer

iReport Designer ir atklātā koda programmatūra, ar kuras palīdzību var taisīt sarežģītas atskaites, izmantojot JasperReports bibliotēku. Caur grafisku interfeisu iespējams izstrādāt atskaites šablonus ļoti ātri un viegli. Strādāt ar šo programmu var izstrādātājs kur saprot atskaišu realizācijas procesu, kurš negrib tērēt laiku uz XML sintakses studēšanu un izstrādātājs, kurš jau pazīst šo sintaksi un grib paātrināt savu darbību (Attēls 2.).



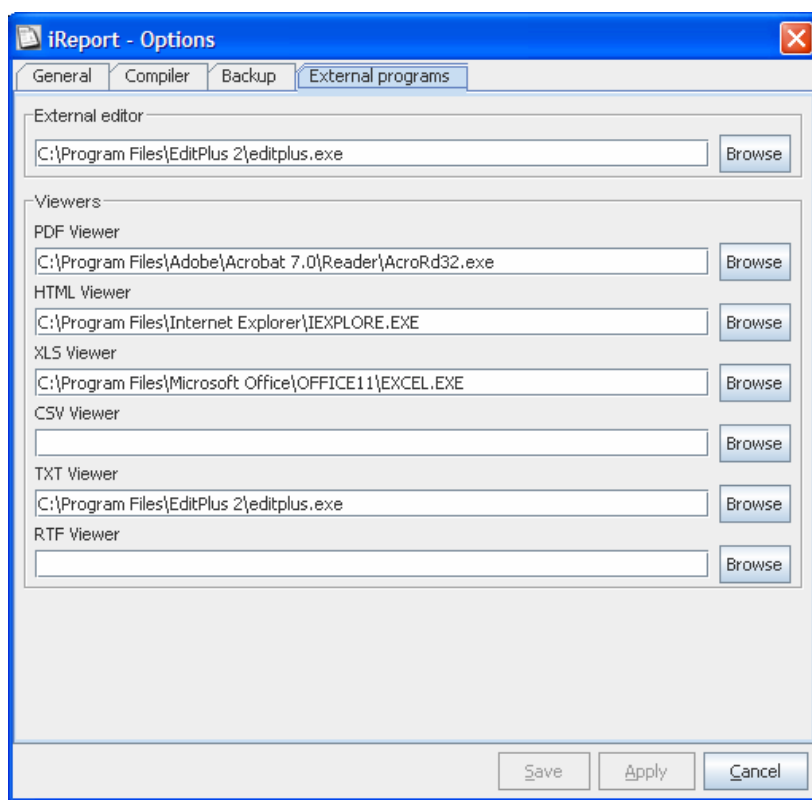
iReport pilnīgi atbalsta JasperReports XML birkas. Programma ietver sevī WYSIWYG redaktoru atskaites veidošanai, kur var izmantot taisnstūrus, līnijas, elipses, teksta laukus, apakšatskaites. iReportā eksistē iebūvēts izgaismošanas redaktors, kurš domāts izteiksmes

rakstīšanai, ka arī pārlūks, lai kontrolēt dokumenta struktūru, integrēti atskaites kompilators un eksportētājs. Programma atbalsta visas datu bāzes pieejamas caur JDBC un skriptletus.

No JasperReports iReports atšķiras ar to, ka tā ir programma - izstrādes vidē, visas nepieciešamas bibliotēkas jau ir iebūvēti, vienīgo ko vajag uzinstalēt ir Ant Build Tool un arī nepieciešama Java 2 (JDK).

Kad palaida programmu pirmo reizi, rekomendējas uzreiz to nokonfigurēt. Vajag tikt opcijas formā nospiežot galvenajā izvēlnē Tools, Options. Tā nav sarežģīta, vienīgi kam vajag pievērst uzmanību ir 'External Programms': šeit ir jānorāda ceļu pie programmām kurus izmanto ar dažādiem izvada formātiem (Attēls 2.).

Pēc visam aprakstītām darbība var sākt darbu ar iReport.

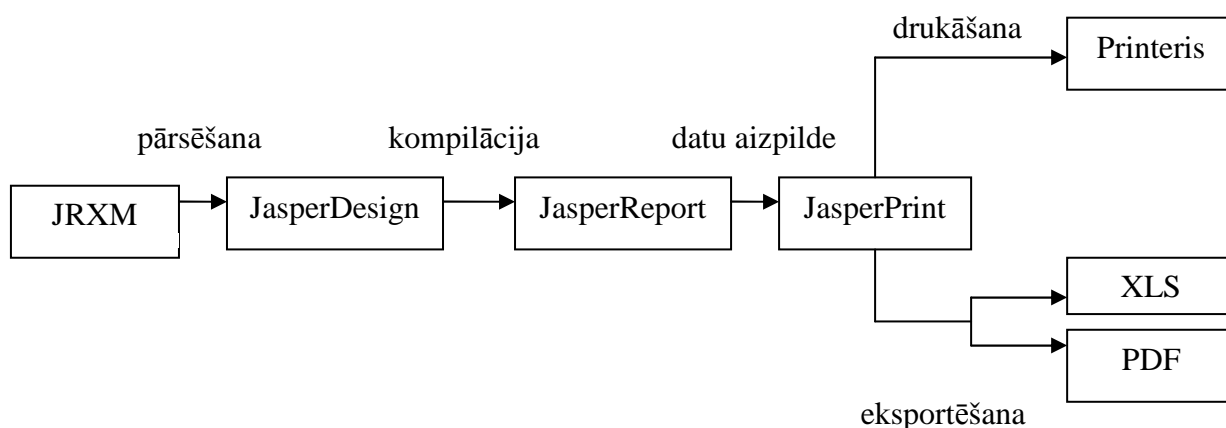


Attēls 2.

2. JasperReports pamatjēdzieni

2.1 Atskaites dzīves cikls

JasperReports atskaites dzīves cikls ļoti līdzīgs Java klases dzīves ciklam. Javā tas ir aprakstīts failā ar izejas kodu un ar paplašinājumu *.java*, kurš uzrakstīts pēc Javas sintakses likumiem. Kods kompilējas ar kompilatora palīdzību un veido klases failu ar paplašinājumu *.class*. Kad klase tiek izmantota, tā ielādas atmiņā un veido tā eksemplāru ar Javas interpretatora palīdzību. Tāpat, atskaite ir aprakstīta failā ar izejas kodu, XML formātā, kurš ir definēts ar DTD (*jasperreport.dtd* var apskatīt pielikumā). Izejas koda faila paplašinājums ir *.jrxml* (t.i. JasperReports XML). Nokompilējot tādu failu, radās *jasper* fails (*.jasper* paplašinājums). Bet pirms tam notiek XML pārsēšana un šis process veido JasperDesign objektu, kurš pēc tam kompilācijas laikā atgriež Jasper failu. Šis fails ir analogisks Java klases failam. Jasper fails ielādējas programmā izpildes laikā un savienojas ar datiem no datu avota, lai izveidot JasperPrint objektu, kuru var izdrukāt uz printera vai eksportēt to kāda formātā, piemēram PDF vai XLS. Vizuāli to var aprakstīt sekojoši:



Attēls 3.

2.2 JRXML izejas koda un Jasper faili

Kā jau bija pateikts, atskaite ir aprakstīta ar XML failu saskaņā ar definīcijām DTD (*jasperreports.dtd*). Šis fails ir dalīts sekcijās: daži no tiem apraksts atskaiti fiziski, piemēram, lapaspuses izmērs, lauku pozicionēšana, grupu augstums, utt.; un daži loģiski: parametru un mainīgu deklarācijas, pieprasījuma definēšana datu izvēlei utt.

Šeit ir JRXML piemērs :

```
<?xml version="1.0" encoding="UTF-8" ?>
<!-- Created with iReport - A designer for JasperReports -->
<!DOCTYPE jasperReport PUBLIC "-//JasperReports//DTD Report Design//EN"
"http://jasperreports.sourceforge.net/dtds/jasperreport.dtd">
<jasperReport
    name="Rep39"
    columnCount="1"
    printOrder="Horizontal"
```

```

orientation="Portrait"
pageWidth="595"
pageHeight="842"
columnWidth="535"
columnSpacing="0"
leftMargin="30"
rightMargin="30"
topMargin="20"
bottomMargin="20"
whenNoDataType="NoPages"
isTitleNewPage="false"
isSummaryNewPage="false"
resourceBundle="Rep39">
<property name="ireport.scriptlethandling" value="0" />
<property name="ireport.encoding" value="UTF-8" />
<import value="java.util.*" />
<import value="net.sf.jasperreports.engine.*" />
<import value="net.sf.jasperreports.engine.data.*" />

<parameter name="dateFrom" isForPrompting="true" class="java.lang.String"/>
<parameter name="dateTo" isForPrompting="true" class="java.lang.String"/>
<parameter name="code" isForPrompting="true" class="java.lang.String"/>
<queryString><![CDATA[select unit.name vdeak,
iisp_rep39_certificate.get_count_all(unit.id,'${dateFrom}','${dateTo}') total,
iisp_rep39_certificate.get_count_firsttime(unit.id,'${dateFrom}','${dateTo}') firsttime,
iisp_rep39_certificate.get_count_by_status(3,unit.id,'${dateFrom}','${dateTo}') policija,
iisp_rep39_certificate.get_count_by_status(2,unit.id,'${dateFrom}','${dateTo}') terminsh,
iisp_rep39_certificate.get_count_by_status(5,unit.id,'${dateFrom}','${dateTo}') sabojata
from
iis_k01_vdeak_units unit
where
unit.CODE=decode('${code}',0,unit.CODE,'${code}')]]></queryString>

<field name="VDEAK" class="java.lang.String"/>
<field name="TOTAL" class="java.lang.Long"/>
<field name="FIRSTTIME" class="java.lang.Long"/>
<field name="POLICIJA" class="java.lang.Long"/>
<field name="TERMINSH" class="java.lang.Long"/>
<field name="SABOJATA" class="java.lang.Long"/>

<background>
<band height="0" isSplitAllowed="true" >
</band>
</background>
<title>
<band height="20" isSplitAllowed="true" >
<textField
isBlankWhenNull="false" evaluationTime="Now" hyperlinkType="None" hyperlinkTarget="Self" >
<reportElement
mode="Transparent"
x="0"
y="0"
width="535"
height="15"
forecolor="#000000"
backcolor="#FFFFFF"
key="textField-17"
stretchType="NoStretch"
positionType="FixRelativeToTop"
isPrintRepeatedValues="true"
isRemoveLineWhenBlank="false"
isPrintInFirstWholeBand="false"
isPrintWhenDetailOverflows="false" />
<box
topBorder="None" topBorderColor="#000000"
leftBorder="None" leftBorderColor="#000000" rightBorder="None" rightBorderColor="#000000"
bottomBorder="None" bottomBorderColor="#000000" />
<textElement
textAlignment="Center"
verticalAlignment="Middle" rotation="None" lineSpacing="Single">
<font fontName="Times New Roman" pdfFontName="Times-
Roman" size="12" isBold="true" isItalic="false" isUnderline="false" isPdfEmbedded="true"
pdfEncoding="Cp1257" isStrikeThrough="false" />
</textElement>
<textFieldExpression
class="java.lang.String"><![CDATA[{$R{Report39.Title}]]></textFieldExpression>
</textField>
</band>
</title>
<pageHeader>
<band height="20" isSplitAllowed="true" >

```

```

        <textField          isStretchWithOverflow="false"          pattern=""
isBlankWhenNull="false"  evaluationTime="Now"  hyperlinkType="None"  hyperlinkTarget="Self" >
        <reportElement
            mode="Transparent "
            x="0"
            y="0"
            width="535"
            height="15"
            forecolor="#000000"
            backcolor="#FFFFFF"
            key="textField-1"
            stretchType="NoStretch"
            positionType="FixRelativeToTop"
            isPrintRepeatedValues="true"
            isRemoveLineWhenBlank="false"
            isPrintInFirstWholeBand="false"
            isPrintWhenDetailOverflows="false" />
        <box              topBorder="None"          topBorderColor="#000000"
leftBorder="None"      leftBorderColor="#000000"      rightBorder="None"      rightBorderColor="#000000"
bottomBorder="None"  bottomBorderColor="#000000" />
        <textElement          textAlignment="Center"
verticalAlignment="Middle"  rotation="None"  lineSpacing="Single">
            <font  fontName="Times New Roman"  pdfFontName="Times-
Roman"  size="10"  isBold="false"  isItalic="false"  isUnderline="false"  isPdfEmbedded  ="true"
pdfEncoding  ="Cp1257"  isStrikeThrough="false"  />
        </textElement>
        <textFieldExpression
class="java.lang.String"><![CDATA[{$R{Report39.PageHeader1}+" "+$P{dateFrom}
+" "+$R{Report39.PageHeader2}+" "+$P{dateTo}]]></textFieldExpression>
        </textField>
    </band>
</pageHeader>
<columnHeader>
    <band height="20"  isSplitAllowed="true" >
        <textField          isStretchWithOverflow="false"          pattern=""
isBlankWhenNull="false"  evaluationTime="Now"  hyperlinkType="None"  hyperlinkTarget="Self" >
        <reportElement
            mode="Transparent "
            x="0"
            y="0"
            width="135"
            height="20"
            forecolor="#000000"
            backcolor="#FFFFFF"
            key="textField-2"
            stretchType="NoStretch"
            positionType="FixRelativeToTop"
            isPrintRepeatedValues="true"
            isRemoveLineWhenBlank="false"
            isPrintInFirstWholeBand="false"
            isPrintWhenDetailOverflows="false" />
        <box              topBorder="Thin"          topBorderColor="#000000"
leftBorder="Thin"      leftBorderColor="#000000"      rightBorder="Thin"      rightBorderColor="#000000"
bottomBorder="Thin"  bottomBorderColor="#000000" />
        <textElement          textAlignment="Center"
verticalAlignment="Middle"  rotation="None"  lineSpacing="Single">
            <font  fontName="Times New Roman"  pdfFontName="Times-
Roman"  size="8"  isBold="true"  isItalic="false"  isUnderline="false"  isPdfEmbedded  ="true"
pdfEncoding  ="Cp1257"  isStrikeThrough="false"  />
        </textElement>
        <textFieldExpression
class="java.lang.String"><![CDATA[{$R{Report39.Head01}]]></textFieldExpression>
        </textField>
        <textField          isStretchWithOverflow="false"          pattern=""
isBlankWhenNull="false"  evaluationTime="Now"  hyperlinkType="None"  hyperlinkTarget="Self" >
        <reportElement
            mode="Transparent "
            x="135"
            y="0"
            width="80"
            height="20"
            forecolor="#000000"
            backcolor="#FFFFFF"
            key="textField-3"
            stretchType="NoStretch"
            positionType="FixRelativeToTop"
            isPrintRepeatedValues="true"
            isRemoveLineWhenBlank="false"
            isPrintInFirstWholeBand="false"
            isPrintWhenDetailOverflows="false" />

```

```

                <box                topBorder="Thin"                topBorderColor="#000000"
leftBorder="Thin"    leftBorderColor="#000000"    rightBorder="Thin"    rightBorderColor="#000000"
bottomBorder="Thin" bottomBorderColor="#000000"/>
                <textElement                textAlignment="Center"
verticalAlignment="Middle" rotation="None" lineSpacing="Single">
                    <font fontName="Times New Roman" pdfFontName="Times-
Roman"    size="8"    isBold="true"    isItalic="false"    isUnderline="false"    isPdfEmbedded    ="true"
pdfEncoding    ="Cp1257"    isStrikeThrough="false"    />
                </textElement>
                <textFieldExpression
class="java.lang.String"><![CDATA[{$R{Report39.Head02}}]></textFieldExpression>
                </textField>
                <textField                isStretchWithOverflow="false"                pattern=""
isBlankWhenNull="false"    evaluationTime="Now"    hyperlinkType="None"    hyperlinkTarget="Self"    >
                    <reportElement
                        mode="Transparent "
                        x="215"
                        y="0"
                        width="80"
                        height="20"
                        forecolor="#000000"
                        backcolor="#FFFFFF"
                        key="textField-4"
                        stretchType="NoStretch"
                        positionType="FixRelativeToTop"
                        isPrintRepeatedValues="true"
                        isRemoveLineWhenBlank="false"
                        isPrintInFirstWholeBand="false"
                        isPrintWhenDetailOverflows="false"/>
                    <box                topBorder="Thin"                topBorderColor="#000000"
leftBorder="Thin"    leftBorderColor="#000000"    rightBorder="Thin"    rightBorderColor="#000000"
bottomBorder="Thin" bottomBorderColor="#000000"/>
                        <textElement                textAlignment="Center"
verticalAlignment="Middle" rotation="None" lineSpacing="Single">
                            <font fontName="Times New Roman" pdfFontName="Times-
Roman"    size="8"    isBold="true"    isItalic="false"    isUnderline="false"    isPdfEmbedded    ="true"
pdfEncoding    ="Cp1257"    isStrikeThrough="false"    />
                        </textElement>
                        <textFieldExpression
class="java.lang.String"><![CDATA[{$R{Report39.Head03}}]></textFieldExpression>
                        </textField>
                        <textField                isStretchWithOverflow="false"                pattern=""
isBlankWhenNull="false"    evaluationTime="Now"    hyperlinkType="None"    hyperlinkTarget="Self"    >
                            <reportElement
                                mode="Transparent "
                                x="295"
                                y="0"
                                width="80"
                                height="20"
                                forecolor="#000000"
                                backcolor="#FFFFFF"
                                key="textField-5"
                                stretchType="NoStretch"
                                positionType="FixRelativeToTop"
                                isPrintRepeatedValues="true"
                                isRemoveLineWhenBlank="false"
                                isPrintInFirstWholeBand="false"
                                isPrintWhenDetailOverflows="false"/>
                            <box                topBorder="Thin"                topBorderColor="#000000"
leftBorder="Thin"    leftBorderColor="#000000"    rightBorder="Thin"    rightBorderColor="#000000"
bottomBorder="Thin" bottomBorderColor="#000000"/>
                                <textElement                textAlignment="Center"
verticalAlignment="Middle" rotation="None" lineSpacing="Single">
                                    <font fontName="Times New Roman" pdfFontName="Times-
Roman"    size="8"    isBold="true"    isItalic="false"    isUnderline="false"    isPdfEmbedded    ="true"
pdfEncoding    ="Cp1257"    isStrikeThrough="false"    />
                                </textElement>
                                <textFieldExpression
class="java.lang.String"><![CDATA[{$R{Report39.Head04}}]></textFieldExpression>
                                </textField>
                                <textField                isStretchWithOverflow="false"                pattern=""
isBlankWhenNull="false"    evaluationTime="Now"    hyperlinkType="None"    hyperlinkTarget="Self"    >
                                    <reportElement
                                        mode="Transparent "
                                        x="375"
                                        y="0"
                                        width="80"
                                        height="20"
                                        forecolor="#000000"
                                        backcolor="#FFFFFF"

```

```

        key="textField-6"
        stretchType="NoStretch"
        positionType="FixRelativeToTop"
        isPrintRepeatedValues="true"
        isRemoveLineWhenBlank="false"
        isPrintInFirstWholeBand="false"
        isPrintWhenDetailOverflows="false" />
    <box topBorder="Thin" topBorderColor="#000000"
leftBorder="Thin" leftBorderColor="#000000" rightBorder="Thin" rightBorderColor="#000000"
bottomBorder="Thin" bottomBorderColor="#000000" />
    <textElement textAlignment="Center"
verticalAlignment="Middle" rotation="None" lineSpacing="Single">
        <font fontName="Times New Roman" pdfFontName="Times-
Roman" size="8" isBold="true" isItalic="false" isUnderline="false" isPdfEmbedded ="true"
pdfEncoding ="Cp1257" isStrikeThrough="false" />
    </textElement>
    <textFieldExpression
class="java.lang.String"><![CDATA[{$R{Report39.Head05}}]></textFieldExpression>
    </textField>
    <textField isStretchWithOverflow="false" pattern=""
isBlankWhenNull="false" evaluationTime="Now" hyperlinkType="None" hyperlinkTarget="Self" >
    <reportElement
mode="Transparent"
x="455"
y="0"
width="80"
height="20"
forecolor="#000000"
backcolor="#FFFFFF"
key="textField-7"
stretchType="NoStretch"
positionType="FixRelativeToTop"
isPrintRepeatedValues="true"
isRemoveLineWhenBlank="false"
isPrintInFirstWholeBand="false"
isPrintWhenDetailOverflows="false" />
    <box topBorder="Thin" topBorderColor="#000000"
leftBorder="Thin" leftBorderColor="#000000" rightBorder="Thin" rightBorderColor="#000000"
bottomBorder="Thin" bottomBorderColor="#000000" />
    <textElement textAlignment="Center"
verticalAlignment="Middle" rotation="None" lineSpacing="Single">
        <font fontName="Times New Roman" pdfFontName="Times-
Roman" size="8" isBold="true" isItalic="false" isUnderline="false" isPdfEmbedded ="true"
pdfEncoding ="Cp1257" isStrikeThrough="false" />
    </textElement>
    <textFieldExpression
class="java.lang.String"><![CDATA[{$R{Report39.Head06}}]></textFieldExpression>
    </textField>
    </band>
</columnHeader>
<detail>
    <band height="30" isSplitAllowed="true" >
        <textField isStretchWithOverflow="false" pattern=""
isBlankWhenNull="true" evaluationTime="Now" hyperlinkType="None" hyperlinkTarget="Self" >
    <reportElement
mode="Transparent"
x="0"
y="0"
width="135"
height="30"
forecolor="#000000"
backcolor="#FFFFFF"
key="textField-8"
stretchType="NoStretch"
positionType="FixRelativeToTop"
isPrintRepeatedValues="true"
isRemoveLineWhenBlank="false"
isPrintInFirstWholeBand="false"
isPrintWhenDetailOverflows="false" />
    <box topBorder="Thin" topBorderColor="#000000"
leftBorder="Thin" leftBorderColor="#000000" rightBorder="Thin" rightBorderColor="#000000"
bottomBorder="Thin" bottomBorderColor="#000000" />
    <textElement textAlignment="Center"
verticalAlignment="Middle" rotation="None" lineSpacing="Single">
        <font fontName="Times New Roman" pdfFontName="Times-
Roman" size="8" isBold="false" isItalic="false" isUnderline="false" isPdfEmbedded ="true"
pdfEncoding ="Cp1257" isStrikeThrough="false" />
    </textElement>
    <textFieldExpression
class="java.lang.String"><![CDATA[{$F{VDEAK}}]></textFieldExpression>

```

```

        </textField>
        <textField          isStretchWithOverflow="false"          pattern=""
isBlankWhenNull="true"  evaluationTime="Now"  hyperlinkType="None"  hyperlinkTarget="Self"  >
        <reportElement
            mode="Transparent "
            x="135"
            y="0"
            width="80"
            height="30"
            forecolor="#000000"
            backcolor="#FFFFFF"
            key="textField-9"
            stretchType="NoStretch"
            positionType="FixRelativeToTop"
            isPrintRepeatedValues="true"
            isRemoveLineWhenBlank="false"
            isPrintInFirstWholeBand="false"
            isPrintWhenDetailOverflows="false" />
        <box          topBorder="Thin"          topBorderColor="#000000"
leftBorder="Thin"  leftBorderColor="#000000"  rightBorder="Thin"  rightBorderColor="#000000"
bottomBorder="Thin"  bottomBorderColor="#000000" />
        <textElement          textAlignment="Center"
verticalAlignment="Middle"  rotation="None"  lineSpacing="Single">
            <font  fontName="Times New Roman"  pdfFontName="Times-
Roman"  size="8"  isBold="false"  isItalic="false"  isUnderline="false"  isPdfEmbedded  ="true"
pdfEncoding  ="Cp1257"  isStrikeThrough="false"  />
        </textElement>
        <textFieldExpression
class="java.lang.Long"><![CDATA[ ${F{TOTAL}} ]]></textFieldExpression>
        </textField>
        <textField          isStretchWithOverflow="false"          pattern=""
isBlankWhenNull="true"  evaluationTime="Now"  hyperlinkType="None"  hyperlinkTarget="Self"  >
        <reportElement
            mode="Transparent "
            x="215"
            y="0"
            width="80"
            height="30"
            forecolor="#000000"
            backcolor="#FFFFFF"
            key="textField-10"
            stretchType="NoStretch"
            positionType="FixRelativeToTop"
            isPrintRepeatedValues="true"
            isRemoveLineWhenBlank="false"
            isPrintInFirstWholeBand="false"
            isPrintWhenDetailOverflows="false" />
        <box          topBorder="Thin"          topBorderColor="#000000"
leftBorder="Thin"  leftBorderColor="#000000"  rightBorder="Thin"  rightBorderColor="#000000"
bottomBorder="Thin"  bottomBorderColor="#000000" />
        <textElement          textAlignment="Center"
verticalAlignment="Middle"  rotation="None"  lineSpacing="Single">
            <font  fontName="Times New Roman"  pdfFontName="Times-
Roman"  size="8"  isBold="false"  isItalic="false"  isUnderline="false"  isPdfEmbedded  ="true"
pdfEncoding  ="Cp1257"  isStrikeThrough="false"  />
        </textElement>
        <textFieldExpression
class="java.lang.Long"><![CDATA[ ${F{FIRSTTIME}} ]]></textFieldExpression>
        </textField>
        <textField          isStretchWithOverflow="false"          pattern=""
isBlankWhenNull="true"  evaluationTime="Now"  hyperlinkType="None"  hyperlinkTarget="Self"  >
        <reportElement
            mode="Transparent "
            x="295"
            y="0"
            width="80"
            height="30"
            forecolor="#000000"
            backcolor="#FFFFFF"
            key="textField-11"
            stretchType="NoStretch"
            positionType="FixRelativeToTop"
            isPrintRepeatedValues="true"
            isRemoveLineWhenBlank="false"
            isPrintInFirstWholeBand="false"
            isPrintWhenDetailOverflows="false" />
        <box          topBorder="Thin"          topBorderColor="#000000"
leftBorder="Thin"  leftBorderColor="#000000"  rightBorder="Thin"  rightBorderColor="#000000"
bottomBorder="Thin"  bottomBorderColor="#000000" />

```

```

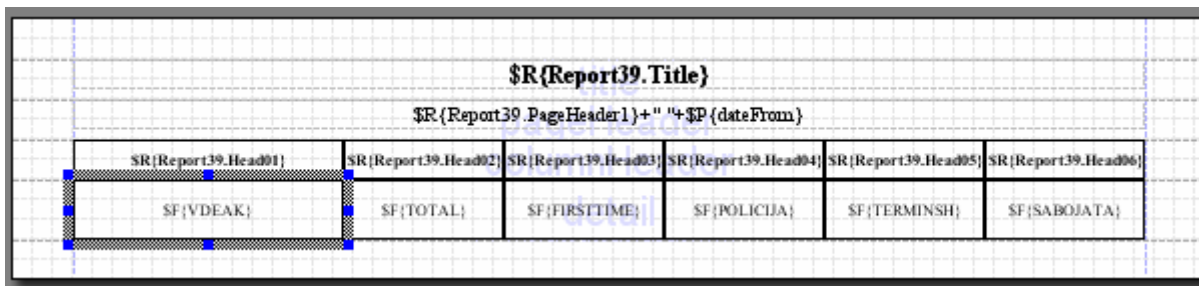
                                <textElement                                textAlignment="Center"
verticalAlignment="Middle" rotation="None" lineSpacing="Single">
                                <font fontName="Times New Roman" pdfFontName="Times-
Roman" size="8" isBold="false" isItalic="false" isUnderline="false" isPdfEmbedded ="true"
pdfEncoding ="Cp1257" isStrikeThrough="false" />
                                </textElement>
                                <textFieldExpression
class="java.lang.Long"><![CDATA[ ${POLICIJA} ]]></textFieldExpression>
                                </textField>
                                <textField                                isStretchWithOverflow="false"                                pattern=""
isBlankWhenNull="true" evaluationTime="Now" hyperlinkType="None" hyperlinkTarget="Self" >
                                <reportElement
                                mode="Transparent "
                                x="375"
                                y="0"
                                width="80"
                                height="30"
                                forecolor="#000000"
                                backcolor="#FFFFFF"
                                key="textField-12"
                                stretchType="NoStretch"
                                positionType="FixRelativeToTop"
                                isPrintRepeatedValues="true"
                                isRemoveLineWhenBlank="false"
                                isPrintInFirstWholeBand="false"
                                isPrintWhenDetailOverflows="false" />
                                <box                                topBorder="Thin"                                topBorderColor="#000000"
leftBorder="Thin" leftBorderColor="#000000" rightBorder="Thin" rightBorderColor="#000000"
bottomBorder="Thin" bottomBorderColor="#000000" />
                                <textElement                                textAlignment="Center"
verticalAlignment="Middle" rotation="None" lineSpacing="Single">
                                <font fontName="Times New Roman" pdfFontName="Times-
Roman" size="8" isBold="false" isItalic="false" isUnderline="false" isPdfEmbedded ="true"
pdfEncoding ="Cp1257" isStrikeThrough="false" />
                                </textElement>
                                <textFieldExpression
class="java.lang.Long"><![CDATA[ ${TERMINSH} ]]></textFieldExpression>
                                </textField>
                                <textField                                isStretchWithOverflow="false"                                pattern=""
isBlankWhenNull="true" evaluationTime="Now" hyperlinkType="None" hyperlinkTarget="Self" >
                                <reportElement
                                mode="Transparent "
                                x="455"
                                y="0"
                                width="80"
                                height="30"
                                forecolor="#000000"
                                backcolor="#FFFFFF"
                                key="textField-13"
                                stretchType="NoStretch"
                                positionType="FixRelativeToTop"
                                isPrintRepeatedValues="true"
                                isRemoveLineWhenBlank="false"
                                isPrintInFirstWholeBand="false"
                                isPrintWhenDetailOverflows="false" />
                                <box                                topBorder="Thin"                                topBorderColor="#000000"
leftBorder="Thin" leftBorderColor="#000000" rightBorder="Thin" rightBorderColor="#000000"
bottomBorder="Thin" bottomBorderColor="#000000" />
                                <textElement                                textAlignment="Center"
verticalAlignment="Middle" rotation="None" lineSpacing="Single">
                                <font fontName="Times New Roman" pdfFontName="Times-
Roman" size="8" isBold="false" isItalic="false" isUnderline="false" isPdfEmbedded ="true"
pdfEncoding ="Cp1257" isStrikeThrough="false" />
                                </textElement>
                                <textFieldExpression
class="java.lang.Long"><![CDATA[ ${SABOJATA} ]]></textFieldExpression>
                                </textField>
                                </band>
                                </detail>
                                <columnFooter>
                                <band height="0" isSplitAllowed="true" >
                                </band>
                                </columnFooter>
                                <pageFooter>
                                <band height="0" isSplitAllowed="true" >
                                </band>
                                </pageFooter>
                                <summary>
                                <band height="0" isSplitAllowed="true" >
                                </band>

```

```
</summary>  
</jasperReport>
```

Izdruka 1.

Kā ir redzams tas ir salīdzinoši garš fails, bet ar ļoti labi saprotamu struktūru. Diemžēl lai viss to pareizi uzrakstīt un neko ne palaist garam vajadzēs daudz laika. Un šeit izstrādātājam palīdzes iReport Designer. Tāda pati atskaite, kura bija apskatīta XML formā, sekojoši grafiski izskatās iReporta:



The screenshot shows a report layout in iReport Designer. At the top, there is a title field: `$_R{Report39.Title}`. Below it is a page header field: `$_R{Report39.PageHeader1}+" "+$_P{dateFrom}`. The main body of the report is a table with six columns. The first column is labeled `SF{VDEAK}`. The other columns are labeled `SF{TOTAL}`, `SF{FIRSTTIME}`, `SF{POLICIJA}`, `SF{TERMINSH}`, and `SF{SABOJATA}`. The table is surrounded by a grid and has a blue border.

Attēls 4.

Ja palaist šo atskaiti bez datu bāzes, rezultāts būs sekojošs:

Izsniegtās invalīda apliecības

Atskaite par izsniegtām invalīda apliecībām laikā no līdz

VDEĀK	Izsniegtas kopā	t.sk. izsniegtas pirmo reizi	Nozaudētas, ir policijas izziņa	Nozaudētas pirms termiņa beigām	Sabojātas pirms termiņa beigām

Attēls 5.

Jau apskatot un salīdzinot abus atskaites veidošanas metodes var secināt, ka ar iReport Designer atskaites izstrādes laiks paātrinās, jo rakstot tīru XML kodu izstrādātājs nevarēs vērot visu atskaiti uzreiz, to var pielaist tikai tad ja tā nav sarežģīta atskaite. Bet ja novērtēt dotu piemēru, grūti pateikt vai tā ir grūta atskaiti vai nav, jo XML fails satur 400 rindas. No šīs puses iReport izskatās kā ļoti pievilcīgs instruments, bet diemžēl, kā jebkura Javas programma tā prasa daudz resursu no izstrādātājas datora, savukārt strādājot ar JasperReports bibliotēku tiešā veidā var izmantot jebkuru teksta redaktoru.

Vēl viena patīkama lieta, kura saistīta ar iReport ir tas, ka pabeidzot dizaina procesu un nospiežot



vienu no podziņām, var dabūt gatavo *jasper* failu direktoriņā, kuru izstrādātājs nedefinēja opcijās. Katra no tiem podziņām izpilda sekojošus procesus:



- izpilda vienkāršu kompilāciju, nerādot nekādu druku, tikai veido *jasper* failu.



- izpilda kompilāciju, veido *jasper* failu un parāda druku bez datiem (tāpat, ka attēlā

5.).



- izpilda kompilāciju, veido *jasper* failu un parāda druku ar datiem no datu avota.

Bet strādājot ar JasperReports bez iReport palīdzības, vispirms jāizveido build.xml failu kur būs aprakstīti visi šie procesi. Piemēram, atskaites kompilācijas uzdevums build.xml varētu izskatīties sekojoši:

```
<taskdef name="jrc"
  classname="net.sf.jasperreports.ant.JRAntCompileTask">
  <classpath>
    <fileset dir="./lib">
      <include name="**/*.jar"/>
    </fileset>
  </classpath>
</taskdef>
```

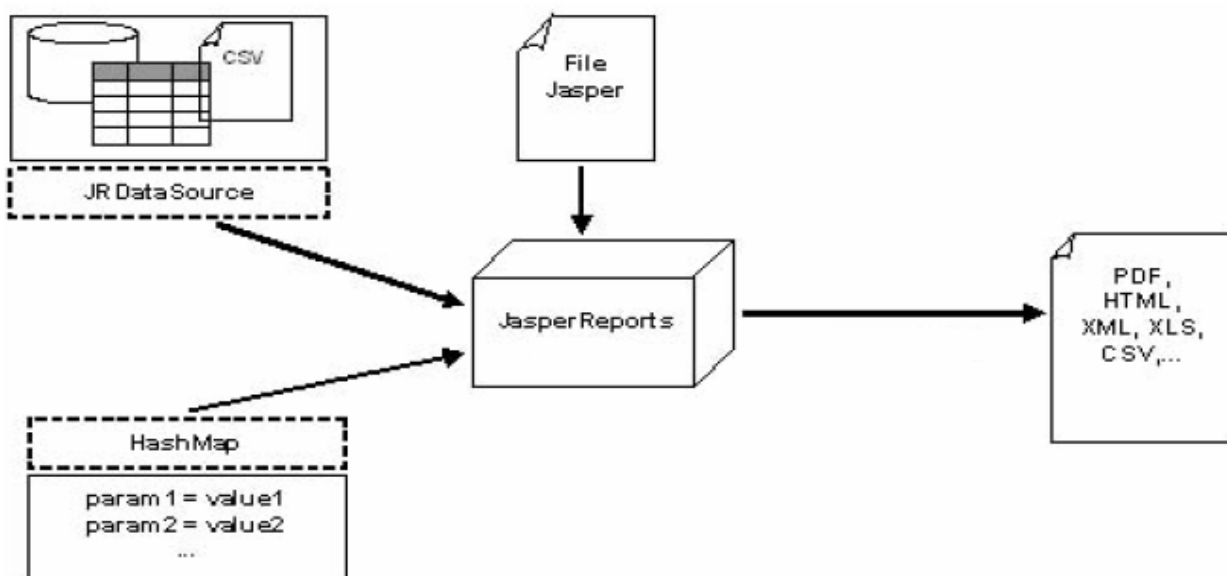
Izdruka 2.

Piemērā lib jāsatur *jasperreports-x.x.x.jar* failu, kur *x.x.x* ir bibliotēkas versija. Šis izstrādes veids der izstrādātājam ar pieredzi, bet iesācējam labāk sākt strādāt ar iReport.

2.3 Datu avoti un drukas formāti

JasperReports definē datus izdrukai divos veidos: kā parametri un kā dati no datu avota, kuri aprakstīti JRDataSource interfeisā. Bibliotēka ļauj aizpildīt atskaites ar datiem lietojot JDBC savienojumu pie kādas relāciju datu bāzes, no kuras šie dati būs paņemti ar SQL pieprasījumu: JRDataSource ļauj nolasīt datus, kuri organizēti tabulās(rindas un kolonas).

Ja vajadzēs saņemt datus ar kaut kādu nosacījumu, tad savlaicīgi jānedefinē atskaites elementus 'vārds/vērtība'. Šos elementus sauc par parametriem. Caur interfeisu fillManger iespējams savienot *jasper* failu un datu avotu *JasperPrint* objektā. Šis objekts var radīt reālu izdrucku, pēc eksportēšanas vēlamā formātā ar atbilstošas klases palīdzību, kurus īsteno JRExporter interfeiss. JasperReports piedāvā dažādus eksportētājus ar kuram var radīt pdf, xls, cvs, xml, html, utt. failus (attēls 4.). Ar JRViewer klasi iespējams uzreiz apskatīt gatavu atskaiti uz ekrānā vai izdrukāt to.



Attēls 6.

2.4 Izteiksmes

Visas JasperReports formulas definēti, kā izteiksmes. Izteiksme nozīme Java instrukciju, kuras rezultāts ir objekts, piemēram:

- “Izteiksme”
- `new Boolean(true)`
- `new Integer(3)`
- `((${Parameter}.equals(“Y”)) ? “Yes” : “No”)`

Nepareizi izteiksmes piemēri ir sekojoši:

- `3+2*5`
- `true`
- `((${Parameter} == 1) ? “Yes” : “No”)`

Pirmās divas izteiksmes ir nepareizas, jo atgriež, kā rezultātu primitīvu datu tipu (`int`), trešais ir nepareizs, jo `${Parameter}` ir primitīvs datu tips, jo to var salīdzināt ar `int` tipu caur `==` operatoru.

Ar izteiksmes palīdzību iespējams strādāt ar teksta laukiem, mainīgām, parametriem un ar atskaites lokalizāciju:

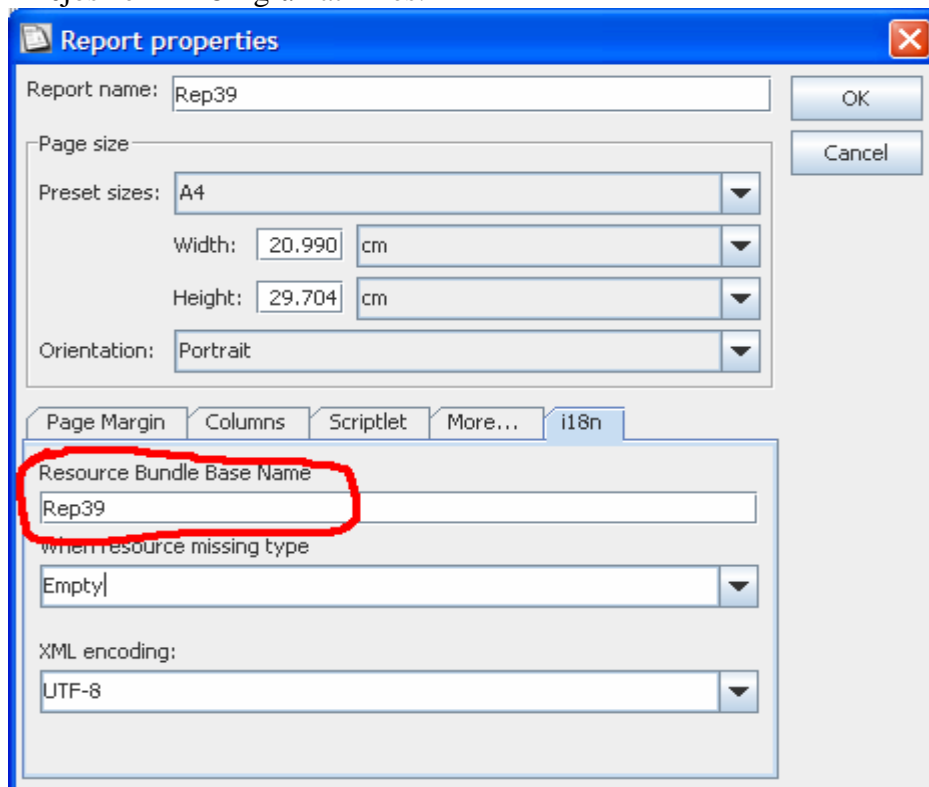
- `#{lauka_nosaukums}` – *lauka_nosaukums* norāda lauku
- `#{mainīgas_nosaukums}` – *mainīgas_nosaukums* norāda mainīgo
- `#{parametra_nosaukums}` – *parametra_nosaukums* norāda parametru
- `#{!parametra_nosaukums}` – *parametra_nosaukums* speciāla sintakse, tiek izmatota SQL pieprasījumos, lai parādīt, ka `#{!parametra_nosaukums}` nav vērtība, bet tikai norāde uz vērtību, kura būs izmantota pieprasījumā nosacījumos
- `#{resursa_nosaukums}` – *resursa_nosaukums* norāda resursu

Teksta lauki, mainīgie un parametri vienmēr pārstāve objektus (tiem varbūt piešķirta `null` vērtība) un tas tips definējas tikai deklarācijas laikā. Runājot par atskaites lokalizāciju, jāpievērst uzmanību, ka *resursa_nosaukums* ir atslēga tekstam fragmentam, kurš glabājas failā ar *.properties* paplašinājumu. Piemēram, priekš attēla 4. šis fails izskata sekojoši:

```
Report39.Head01=VDE\u0100K
Report39.Head02=Izsniegta kop\u0101
Report39.Head03=t.sk. izsniegta pirmo reizi
Report39.Head04=Nozaud\u0113tas, ir policijas izzi\u0146a
Report39.Head05=Nozaud\u0113tas pirms termi\u0146a beig\u0101m
Report39.Head06=Saboj\u0101tas pirms termi\u0146a beig\u0101m
Report39.PageHeader1=Atskaite par izsniegt\u0101m inval\u012Bda
apliec\u012Bb\u0101m laik\u0101 no
Report39.PageHeader2=l\u012Bdz
Report39.Title=Izsniegt\u0101s inval\u012Bda apliec\u012Bbas
```

Izdruka 3.

Šo failu iz jāapzīmē zem *jasperReport* birkas, kā *resourceBundle="Rep39"*, vai iReportā atskaites apzīmējos zem *i18n* grāmatzīmes:



Attēls 7.

Lai dod vēl skaidrāku priekšstatu, parādīsim mazliet sarežģītākus izteiksmes piemērus:

- `new Double($F{f1}.doubleValue() + $F{f2}.doubleValue())`
- `new Boolean($F{f}.intValue()==1)`
- `new Boolean($F{f}!=null && $F{f}.equals("test"))`

2.5 Atskaites izsaukšana Javā

Par svarīgu lietu tiek uzskatīta atskaites izsaukšana Javā. Kā piemēru paņemsim atskaiti, kuru mēs apskatījām iepriekš (Rep39), un izveidosim Java klasi, kuru izsauks šo atskaiti, kāda Javas programma:

```
import net.sf.jasperreports.engine.*;
import net.sf.jasperreports.engine.export.*;
import java.util.*;

public class JasperTest
{
    public static void main(String[] args)
    {
        String fileName = "Rep39.jasper";
        String outFileName = "Rep39.pdf";
        HashMap hm = new HashMap();
        hm.put("dateFrom", "01/01/2005");
        hm.put("dateTo", "01/01/2005");
        hm.put("code", "11");

        try
        {
```

```

        JasperPrint print = JasperFillManager.fillReport(
            fileName,
            hm,
            new JREmptyDataSource());

        JRExporter exporter = new net.sf.jasperreports.engine.export.JRPdfExporter();

        exporter.setParameter(JRExporterParameter.OUTPUT_FILE_NAME,
            outFileName);
        exporter.setParameter(JRExporterParameter.JASPER_PRINT, print);

        exporter.exportReport();

        System.out.println("Created file: " + outFileName);
    }
    catch (JRException e)
    {
        e.printStackTrace();
        System.exit(1);
    }
    catch (Exception e)
    {
        e.printStackTrace();
        System.exit(1);
    }
}

```

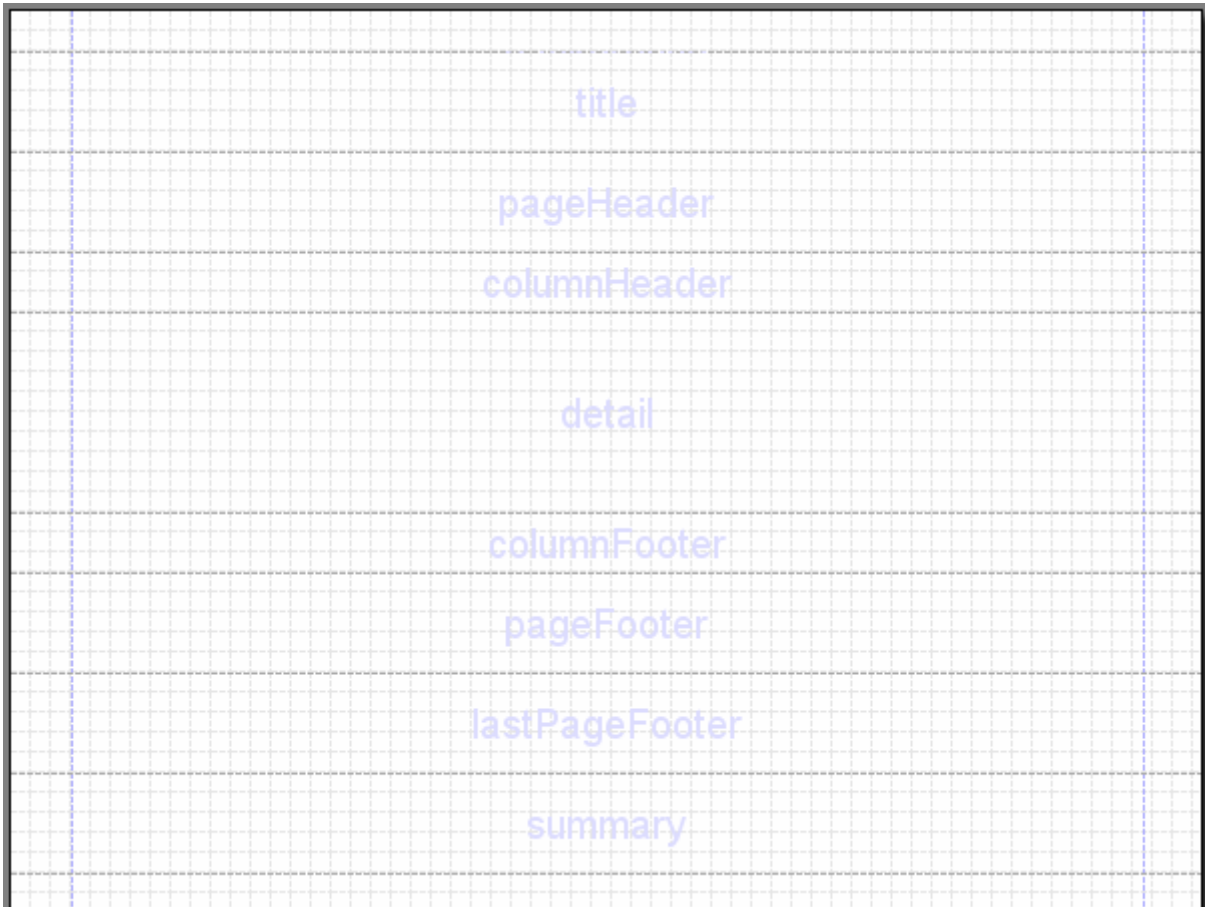
Izdruka 4.

Atskaites parametrus jāpadod, kā objektu HashMap, ar atslēgām, kuri ir atskaites *parametru_nosaukumi* JRXML failā. JREmptyDataSource padots *fillReport* metodē, kā tukšais datu avots. Rezultātā mēs saņemsim attēlu 5, failā Rep39.pdf.

3. Atskaites struktūra

3.1 Sekcijas

Atskaite ir sadalīta dažādās horizontālas sekcijās. Kad tā ir palaista un savienojas ar datiem, šīs sekcijas tiek drukāti vairākas reizes un saskaņā ar likumiem, kurus nodefinējis izstrādātājs. Piemēram, lapaspuses galvene atkārtos uz katras lapaspuse, bet detaļu sekcija tiek palaista ar katru ierakstu. Pēc noklusēšanas šablons tiek dalīts deviņas sekcijās un eksistē iespēja pielikt klāt grupas, kuriem sastāv no grupas galvenes un grupas kājenes.



Attēls 8.

Sekcija vienmēr tik plaša, cik ir plaša pati lapaspuse, neieskaitot malas. Ja sekcijas augstums ir nodefinēts dizaina laikā, nav obligāti, ka tā būs tik augsta, kā bija noteikts, tā var būt palielināta uz apakšu saskaņā ar sekcijas elementiem. Tā parasti notiek, kad tiek izmantoti apakšatskaites, vai teksta laukiem nepietiek vietas, lai pilnībā parādīt saturu. Tātad var secināt, ka dizaina laikā izstrādātājs var noteikt minimālo sekcijas augstumu. Bet ne visas sekcijas varētu būt dināmiski organizēti saskaņā ar saturu, īpaši kolonas kājene, lapaspuses kājene un pēdējas lapaspuses kājene.

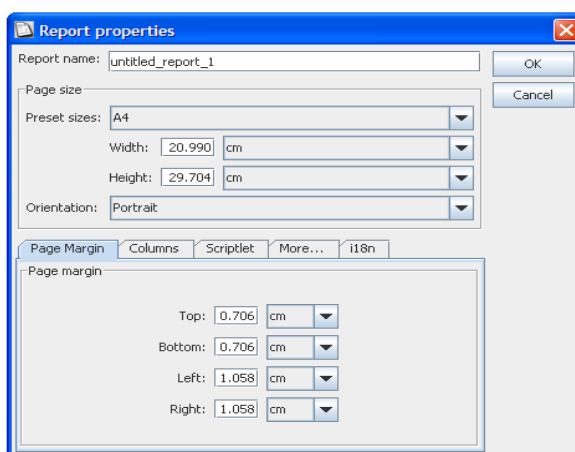
Summējot visas sekcijas augstumus(izņemot fonu) ir jāsaņem skaitļi, kurš būtu mazāk, vai vienāds ar lapaspuses augstumu, mīnus augšas un lejas malas.

Katra atskaite iekļauj sevi sekojošas sekcijas:

- Virsraksts – ir pirmā lieta, ko ieraudzīs informatīvas sistēmas lietotājs, palaižot kādu no atskaitēm. Šī sekcija veidojas tikai vienu reizi, bet ir iespēja drukāt to uz katras lapaspuses.
- Lapaspuses galvene – ļauj nodēfinēt lapaspuses galveni, kura būs drukāta uz katras lapaspuses, tādā pašā pozīcijā, kura bija padota tam dizaina laikā.
- Kolonas galvene – šī sekcija tiek drukāta pirms katras kolonas, kuras atrodas detaļu sekcijā.
- Grupas galvene – atskaite var saturēt nulle, vai vairāk grupas sekcijas, kuras dod iespēju operēt ar detalizēto ierakstu kolekcijām. Ja eksistē grupas galvene, tad eksistē arī grupas kājene. Iespējams noteikt tā, lai grupas galvene būtu drukāta jaunajā lapaspusē.
- Detaļu sekcija – atrodas atbilstībā ar katru ierakstu no datu avota. Ar lielo varbūtību var pateikt, ka visvairāk elementu būs drukāti šeit.
- Grupas kājene – beidz grupas sekciju. Parasti satur laukus ar starp aprēķinus, vai kādu grafisko elementu, piemēram: līnijas.
- Kolonas kājene – šī sekcija parādās katras kolonas beigās.
- Lapaspuses kājene – tiek drukāta uz katras lapaspuses.
- Pēdējas lapaspuses kājene – dod iespēju uztaisīt pēdējas lapaspuses kājeni atšķirīgo no citas lapaspuses kājeni.
- Kopsavilkums – analogiski lapaspuses kājenēm.
- Fons – dod iespēju izdarīt atskaiti ar ūdenszīmēm.

3.2 Atskaites opcijas

Sākot strādāt ar iReport, vispirms jāizveido jauno šablonu. Lai to izdarītu jānospiež galvenajā izvēlnē zem *File* opcijas sarakstā *New Document*. Parādīsies sekojošs logs:



Attēls 9.

Caur opcijas logu iReport ņem datus un piešķir tos atbilstošiem atribūtiem JRXML failā.

Piemēram:

```
<jasperReport
  name="Rep39"
  columnCount="1"
  printOrder="Horizontal"
  orientation="Portrait"
  pageWidth="595"
  pageHeight="842"
  columnWidth="535"
  columnSpacing="0"
  leftMargin="30"
  rightMargin="30"
  topMargin="20"
  bottomMargin="20"
  whenNoDataType="NoPages"
  isTitleNewPage="false"
  isSummaryNewPage="false"
  resourceBundle="Rep39">
```

Izdruka 5.

Šeit uzreiz var pamanīt iReporta plusu: tam eksistē iespēja nodefinēt lapas izmērus caur lapas formātu(A4, A3 utt.). Arī šeit var norādīt kolonas skaitu un tās izmērus, ceļu uz atskaites klasi (skriptletu), kura var izpildīt izstrādātāja aprakstītas darbības, mazliet nodefinēt drukāšanas metodi, piemēram, kā izdrukāt virsrakstu, kopsavilkumu, kolonas kājeni nākamajā lapaspusē un izdrukāšanas principu: vertikāli vai horizontāli. Šeit arī var norādīt resursa failu, pār kuru jau bija runāts 2.4 Izteiksmes.

4. Atskaites elementi

4.1 Atskaites elementa definīcija

Par atskaites elementu sauc grafisko elementu, piemēram teksts vai taisnstūris. Ir septiņi tā saukti „galveni” elementi, ko piedāvā JasperReports bibliotēka:

- Līnija
- Taisnstūris
- Elipse
- Statisks teksts
- Teksta lauks (vai vienkārši lauks)
- Attēls
- Apakšatskaite

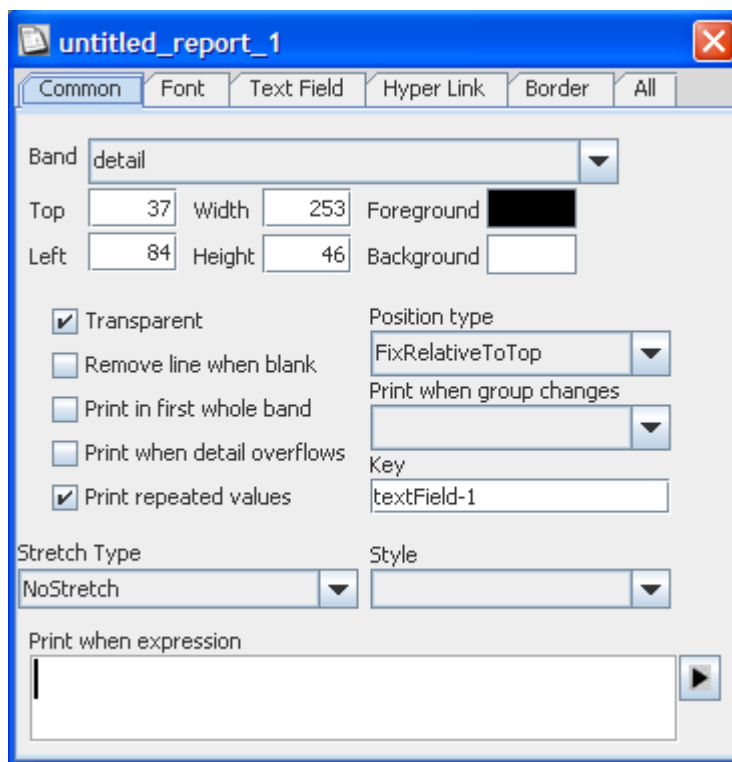
Kombinējot šos elementus, rādās iespēja izstrādāt ļoti sarežģītas atskaites. Katram elementam ir pamatatribūti, piemēram, augstums, platums, koordinātas, grupa vai sekcija, kurai tas pieder un citas īpašības (piemēram, šrifts, vai ja elements ir taisnstūra malas biezums). Atskaites elementus var sadalīt divās grupās: grafiski elementi(līnija, taisnstūris, elipse, attēlošana) un teksta lauki. Apakšatskaite pārstāv atsevišķu elementu grupu, jo šis elements diezgan sarežģīts.

Katrs elements atrodas vienā sekcijā, nevar notikt tā, ka viena elementa daļa atrodas vienā sekcijā un otrā citā. Ja tāda situācija gadījusies, tad kompilators atgriezis izziņu, kura informēs par kļūdu atskaites dizainā. Atskaite jebkura gadījuma būs nokompilēta, bet elements, kurš radījis kļūdu nebūs izdrukāts.

Atskaites elementu pozicionēšana iReportā, atgādina darbu ar jebkuru grafisko redaktoru, jo tur izstrādātājas var izvēlēties nepieciešamo elementu no rīkjolas un viegli nodefinēt tam visas nepieciešamas atribūtas. Savukārt, atskaites veidošana bez iReportas palīdzības, tas ir tāds pats, kā taisīt tīmekļa lapas primitīvā teksta redaktorā, ar vienu atšķirību: internēt lapas mēs aprakstām vairāk ar HTML, bet atskaites tikai ar XML.

4.2 Elementa atribūti

Visiem elementiem ir vienāda īpašību kopa. Šo kopu var apskatīt iReportā galvenajā izvēlnē zem *View* → *Element Properties* (Attēls 10.):



Attēls 10.

Šī paša kopa sekojoši aprakstās ar XML:

```

<textField isStretchWithOverflow="false" pattern="
isBlankWhenNull="false" evaluationTime="Now" hyperlinkType="None" hyperlinkTarget="Self" >
  <reportElement
    mode="Transparent "
    x="84"
    y="37"
    width="253"
    height="46"
    forecolor="#000000"
    backcolor="#FFFFFF"
    key="textField-1"
    stretchType="NoStretch"
    positionType="FixRelativeToTop"
    isPrintRepeatedValues="true"
    isRemoveLineWhenBlank="false"
    isPrintInFirstWholeBand="false"
    isPrintWhenDetailOverflows="false"/>
    <box topBorder="Thin" topBorderColor="#000000"
leftBorder="Thin" leftBorderColor="#000000" rightBorder="Thin" rightBorderColor="#000000"
bottomBorder="Thin" bottomBorderColor="#000000"/>
      <textElement textAlignment="Center"
verticalAlignment="Middle" rotation="None" lineSpacing="Single">
        <font fontName="Times New Roman" pdfFontName="Times-
Roman" size="8" isBold="true" isItalic="false" isUnderline="false" isPdfEmbedded = "true"
pdfEncoding = "Cp1257" isStrikeThrough="false" />
      </textElement>
      <textFieldExpression
class="java.lang.String"><![CDATA[{$R{Report39.Head06}}]></textFieldExpression>
    </textField>

```

Izdruka 6.

Koordinātes un izmēri vienmēr skaitās pikseļos.

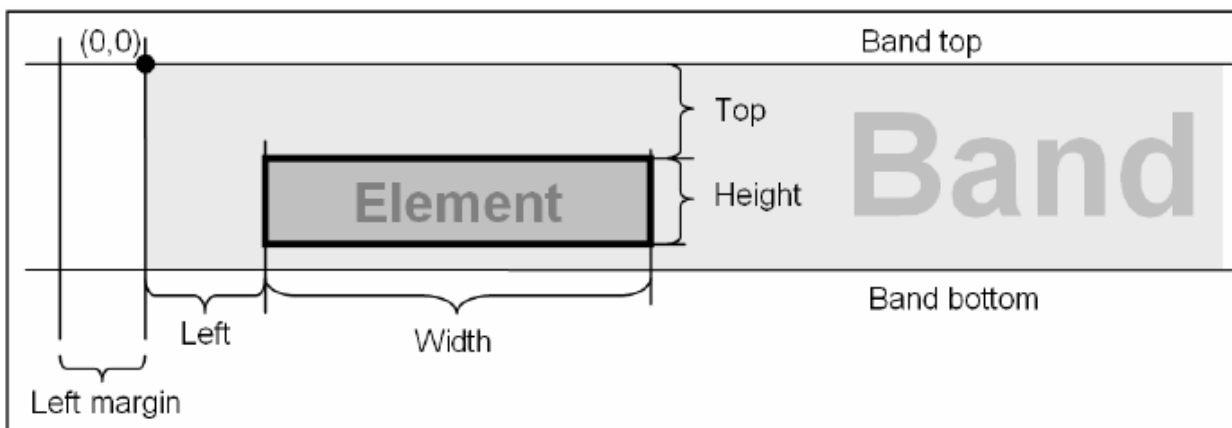
Band – šī opcija atbild par sekciju, kurai elements ir piešķirts, katrs elements atrodas tikai vienā sekcijā un ne vairāk;

Top – tas ir distance starp augšas labas elementa stūres un sekcijas augstāko punktu;

Left – tas ir distance starp augšas kreisas elementa stūres un atskaites kreiso malu;

Width – elementa platums;

Height – elementa augstums, nodrošina minimālo augstumu, kura var palielināties atskaites izpildes laikā saskaņā ar datiem šajā elementā;



Attēls 11.

Foreground – teksta krāsa;

Background – fona krāsa;

Transparent – ļauj padarīt elementu par caurspīdīgu;

Remove Line When Blank – kad elements ir tukšs, opcija ļauj paslēpt elementu, tā ka var rasties doma, ka tas vispār neeksistēja;

Print in the first whole band – ļauj drukāt elementu jaunajā lapaspusē, ja tam nepietiek vietas dotajā;

Print when detail overflows – ļauj drukāt elementu nākamajā lapaspusē, ja sekcijai nepietiek vietas dotajā;

Position type – definē koordinātes uzvedību, ja mainīsies atskaites augstums:

- *FixRelatieToTop* – distance starp elementa un sekcijas augstākiem punktiem ne mainās
- *Float* – ja iepriekšējs elements palielināja savu augstumu, tad elements ar *Float* ies uz lēju;
- *FixRelatieToBottom* – pretēji *FixRelatieToTop*;

Print when group changes – šeit piedāvās visas atskaites grupas. Elements būs izdrukāts tikai tad, kad būs izveidota jauna atskaites izteiksme izvēlētā grupā;

Key – elementa nosaukums;

Stretch Type – ļauj nedefinēt elementa uzvedību drukāšanas procesā:

- *No Stretch* – nozīme fiksēto augstumu;
- *RelativeToBandHeight* – elementa augstums palielinās kopa ar sekcijas augstumu;
- *RelativeToTallestObject* – elementa augstums būs tik augsts, cik ir augsts pats augstākais elements sekcijā;

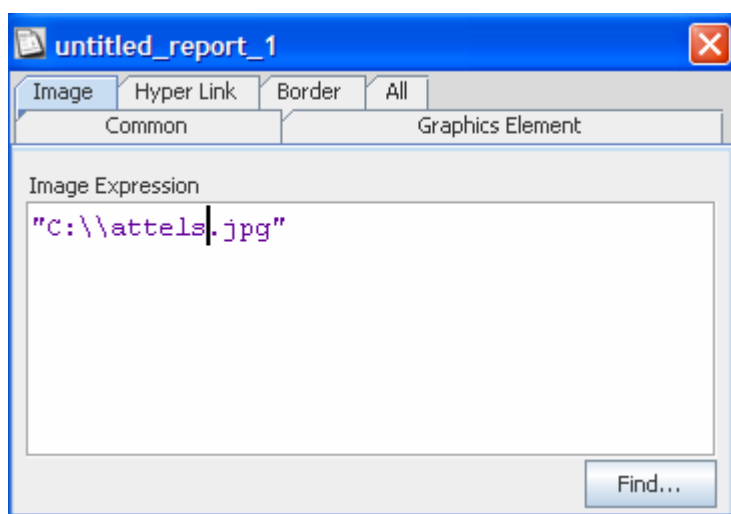
Print when expression – tā ir Java izteiksme, kura atgriež objektu Boolean, ja objekta vērtība ir true tad elements būs izdrukāts, ja false, tad nebūs.

4.3 Grafiski elementi

Grafiski elementi(līnijas, taisnstūri) neattēlo nekādu informācija, tie palīdz palielināt atskaites lasāmību.

Zīmējot taisnstūru, elipsi, līniju jānodēfinē tikai līnijas platumu. Tā varētu būt no 0 līdz 4 pikseļiem, vai uzzīmēta ar punktiem.

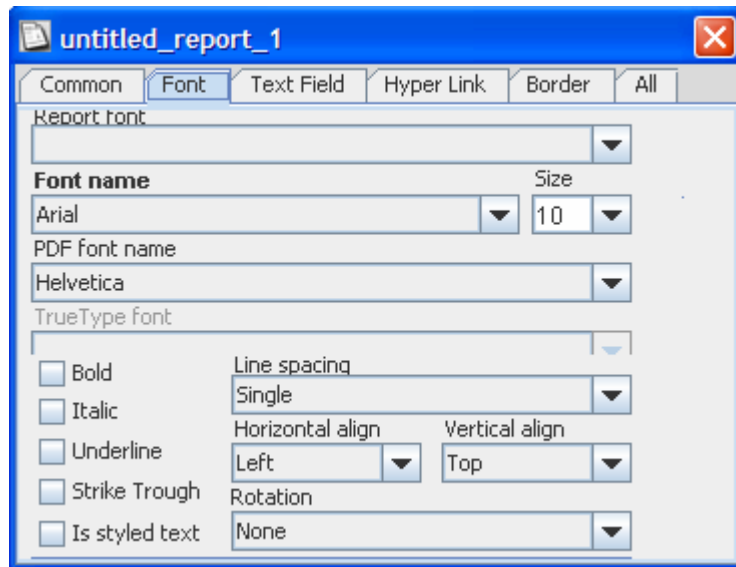
Šai grupai pieder arī attēli (GIF, PNG, JPEG, utt.). Galvenais nosacījums attēla radīšanai atskaitē ir definēts ceļš līdz grafiskam failam:



Attēls 12.

4.4 Teksta elementi

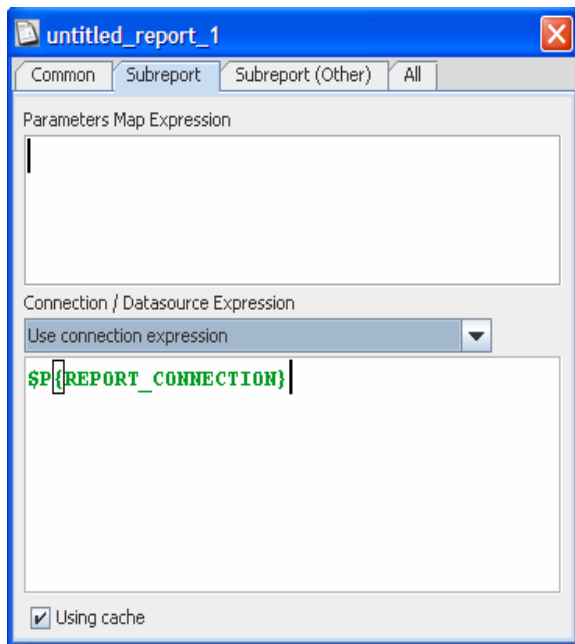
Teksta elementi tiek uzskatīti par atskaites pamatelementiem. Ir divi teksta laukuma tipi: statisks teksta laukums un teksta laukums. Statiska teksta laukuma definēta vērtība nebūs mainīta. Savukārt teksta laukums domāts, lai drukāt Javas izteiksmes rezultātus, kuri var atšķirties viens no otra(piemēram dati no datu avota). Teksta laukiem var nodēfinēt šrifta izmēru, attālumu starp rindām, teksta rotāciju un pozīciju elementā(pa kreisi, pa labi, centrā):



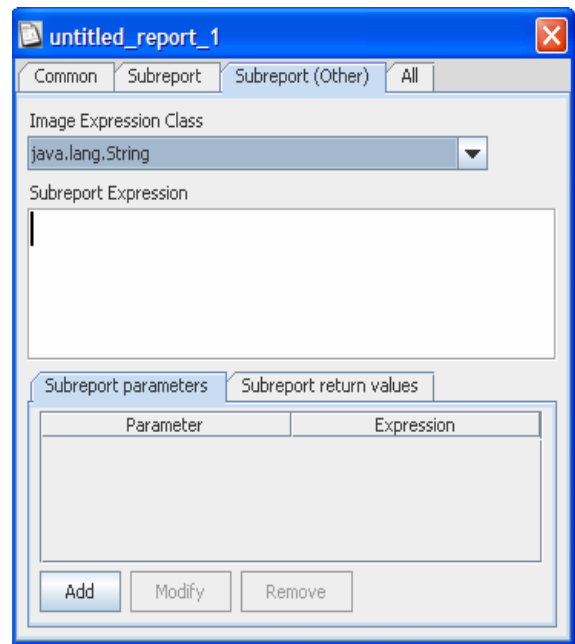
Attēls 13.

4.5 Apakšatskaites

Apakšatskaite tas ir elements, kuram ir savs JRXML un *jasper* faili ar savu datu avotu. Šis elements pārstāv visprogresīvāko tehnoloģiju un dod iespēju ražot sarežģītas atskaites. Galvenā ideja ir ievietot atskaiti atskaitē. Elementa atribūti ir:



Attēls 14.



Attēls 15.

- *Parameters Map Expression* – definē izteiksmi, kura tiek novērtēta izpildes laikā; izteiksmei jāatgriež `java.util.Map` – tā satur pārus nosaukums/objekts, kuri būs padoti apakšatskaitei, kā parametri;
- *Connection/Datasource expression* – definē izteiksmi, kura izpildes atgriezīs JDBC savienojumu vai `JRDataSource`, kurš būs izmantots, lai aizpildīt atskaiti ar datiem;

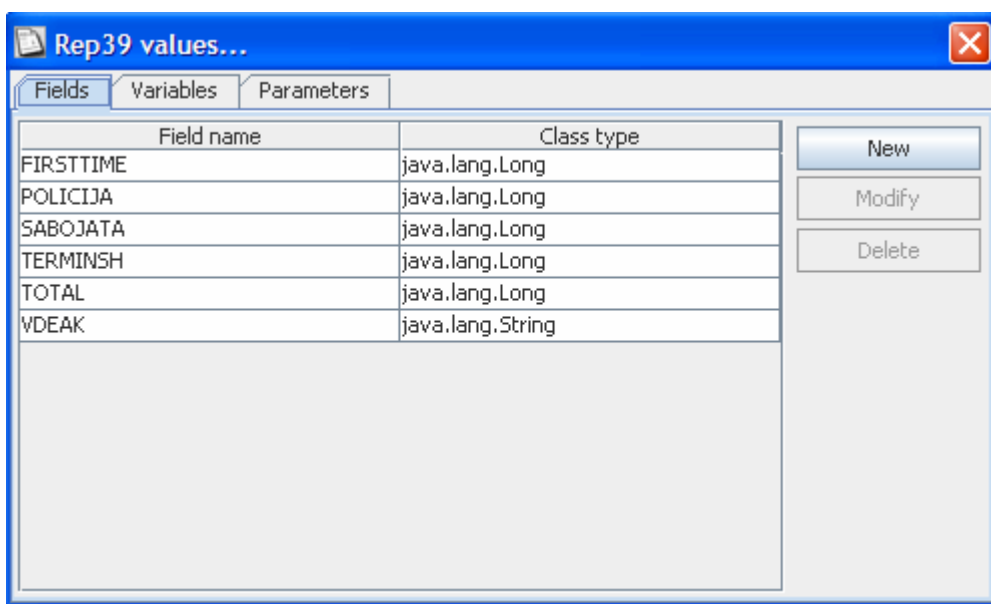
- *Using cache* – ļauj paturēt atmiņā atskaites datu struktūru, lai padarīt nākamo ielādēšanu ātrāk.
- *Subreport expression* – tiek domāta lai norādīt ceļu līdz apakšatskaites *jasper* failam.
- *Subreports parameters* – tabula satur pārus nosaukums/izteiksme, kuri varbūt padoti apakšatskaitē, kā parametri.

5. Lauki, parametri, mainīgie

5.1 Datu glabāšanas objekti

Atskaitēm ir trīs grupas, kuri var saturēt vērtības: lauki, parametri un mainīgie. Šie objekti tiek izmantoti izteiksmēs, tie var mainīt savu vērtību kad atskaite tiek drukāta un katrs objekts var reprezentēt kādu datu tipu (piemēram *String*, *Double* utt.).

Laukus, parametrus un mainīgus jādefinē atskaitē tādā secībā, kā viņi būs izmantoti. iReport piedāvā iespēju apstrādāt šos objektus speciālajā izvēlnē:



Attēls 16.

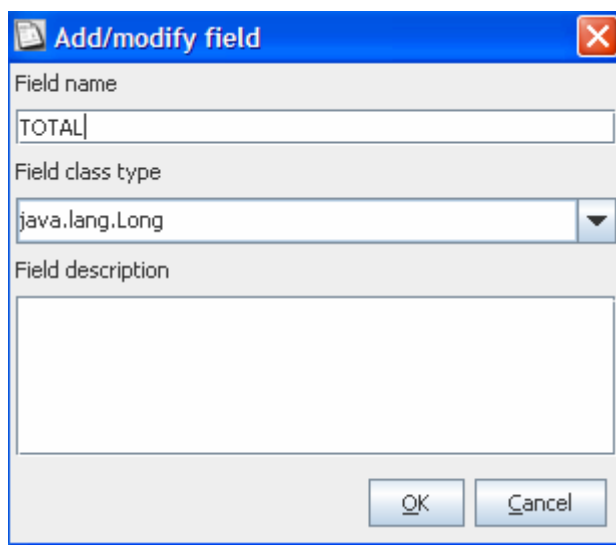
XML implementācija tas izskatās sekojoši:

```
<field name="VDEAK" class="java.lang.String"/>
<field name="TOTAL" class="java.lang.Long"/>
<field name="FIRSTTIME" class="java.lang.Long"/>
<field name="POLICIJA" class="java.lang.Long"/>
<field name="TERMINSH" class="java.lang.Long"/>
<field name="SABOJATA" class="java.lang.Long"/>
<variable name="var" class="java.lang.Integer" resetType="Report" calculation="Count">
  <variableExpression><![CDATA[ ${POLICIJA} ]]></variableExpression>
</variable>
<parameter name="dateFrom" isForPrompting="true" class="java.lang.String"/>
<parameter name="dateTo" isForPrompting="true" class="java.lang.String"/>
```

Izdruka 7.

5.2 Lauki

Atskaites dati rādās no datu avota. JasperReports bibliotēkā atskaites informācija organizēta ierakstos, kuri līdzīgi SQL pieprasījuma rezultātiem. Katrs ieraksts var saturēt vienu, vai vairāk laukus. Lai nodefinēt lauku iReportā, vajag izsaukt *Add/Modify Field* lodziņu:

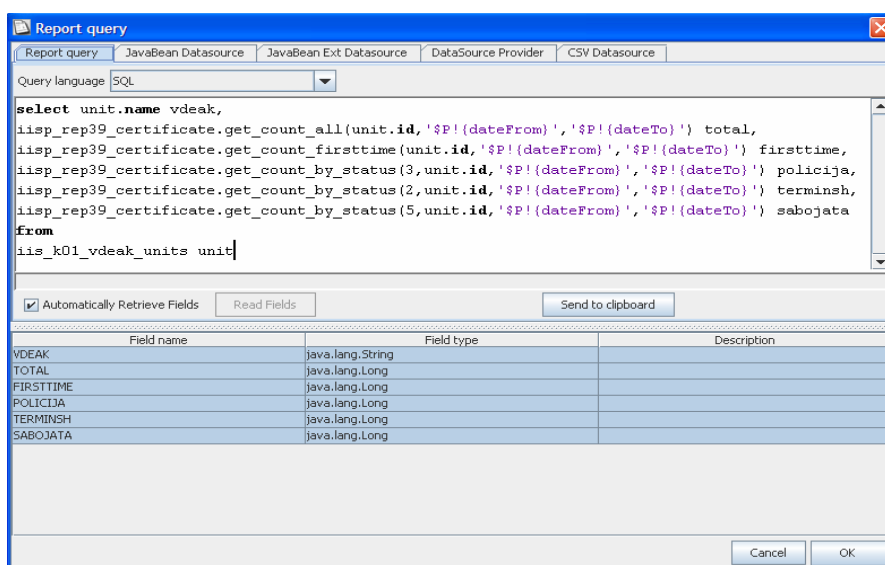


Attēls 17.

Lauks tiek aprakstīts ar lauka vārdu, lauka tipu un lauka aprakstu, kurš ir opcionāls. Lai izsaukt lauku atskaites drukāšanas procesā ir izmantots speciālā sintakse:

$\$F\{<lauka_nosaukums>\}$

Atskaite var saturēt daudz lauku (pat vairākus simtus). Lai padarīt darbu vieglāk iReport piedāvā speciālus instrumentus. Viens no tam ir SQL pieprasījuma lauku reģistrācija. Tā ir vissvarīgākā opcija, kura ļoti plaši tiek izmantota. Piemēram, ir SQL pieprasījums, kurš atgriež ierakstus, kuri sadalīti laukos(tie varētu būt, jau nodefinēti lauki, vai funkcijas rezultāti, Attēls 18.):



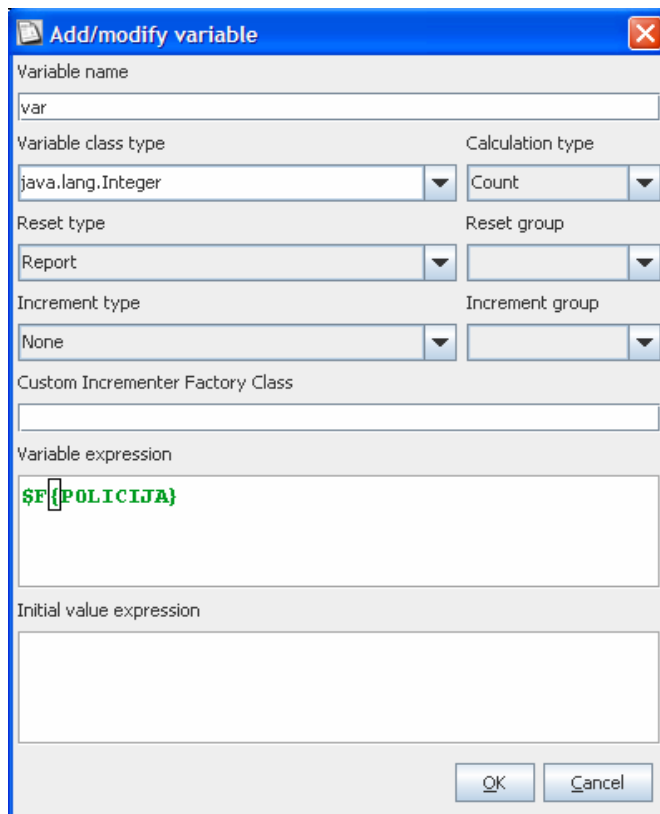
Field name	Field type	Description
VDEAK	java.lang.String	
TOTAL	java.lang.Long	
FIRSTTIME	java.lang.Long	
POLICIJA	java.lang.Long	
TERMINSH	java.lang.Long	
SABOJATA	java.lang.Long	

Attēls 18.

Kad pieprasījums ir ierakstīts logā, tad tas uzreiz tiek palaists un iReport analīzes saņemto rezultātu, lai nodefinēt laukus automātiski, kurus var redzēt logā apakšdaļā.

5.3 Mainīgie

Mainīgie tiek domāti, lai saglabāt informāciju par kalkulācijām, kā summas un apakš summas:



Attēls 19.

Mainīga, tāpat, kā lauks tiek aprakstīta ar vārdu un tipu, bet kā ir redzams attēlā 19., šeit ir atšķirības:

- Kalkulācijas tips – definē, kāda veidā mainīgas vērtība būs mainīta (nebūs mainīta, skaitīs cik ieraksti bija izdrukāti, skaitīs vienas kolonas summu, vidējo aritmētisko, minimālo un maksimālo vērtību)
- Atjaunošanas tipu – kad mainīgai jāpiešķir null. To var izdarīt vai nu atskaites beigās, vai lapaspuses, kolonas, grupas

iReport piedāvā arī iebūvētos mainīgas, piemēram, lapaspuses numurēšanai ($\$V\{PAGE_NUMBER\}$).

5.4 Parametri

Parametri, tas ir objekti, kuri tiek padoti atskaitē no Java aplikācijas, lai vadīt procesus pareizā veidā (piemēram SQL pieprasījumā, to izmanto nosacījumu daļā). Parametrs tiek definēts ar vārdu, tipu, opcionālo aprakstu un vērtību pēc noklusēšanas. Izsaukšanas sintakse ir sekojoši:

```
$P<parametra_nosaukums> }
```

Ja vajag izsaukt parametru SQL nosacījumu daļā, tad pēc '\$P' jāieliek izsaukuma zīmi:

```
$P!<parametra_nosaukums> }
```

Parametriem, varbūt piešķirta vērtība pēc noklusēšanas, tā var notikt tikai gadījumā, ja no aplikācijas nebija nekas padots.

Parametri tiek padoti atskaitēm ar HashMap palīdzību, piemēram:

```
HashMap hm = new HashMap();  
...  
JasperPrint print = JasperFillManager.fillReport(  
    fileName,  
    hm,  
    new JREmptyDataSource());  
...
```

Izdruka 8.

6. Datu avoti

6.1 Datu avoti

Datu avots tas ir avots, no kura JasperReports ņem datus lai aizpildīt atskaiti. Datu avoti sastāv no divām grupām: JDBC savienojumi ar relāciju datubāzēm, kur SQL pieprasījumi tiek palaisti, un objekti, kuri paplašina JRDataSource interfeisu, kurš ļauj strādāt ar XML dokumentiem un JavaBean masīviem.

JasperReports spēj strādāt ar jebkuru datubāzi, kurai ir JDBC draiveris. Šī īpatnība kopā ar SQL lauku reģistrāciju dod iespādu par to, ka darbam ar JDBC jānotiek ļoti ātri un kvalitatīvi.

Ja nav iespējas strādāt ar datu bāzi caur JDBC, tad var lietot JRDataSource interfeisu (tas nozīme JasperReports Data Source), kurš ļauj tikt pie datiem, kuri noorganizēti tabulās, kolonās un rindās.

JDBC savienojums un JRDataSource netiek veidoti ar JasperReports palīdzību, tie veidoti ar aplikāciju, kura izsauc atskaites. Programma sūta *fillReport* metodēm (šī metode atbild par atskaites aizpildīšanu ar datiem) atvērto JDBC savienojumu, vai JRDataSource instanci, lai aizpildīt atskaites šablonu ar datiem.

JasperReports bibliotēka dod iespēju strādāt dažādiem datu avotiem:

- JDBC savienojumiem
- XML datu avotiem

- JavaBean datu avotiem
- CVS datu avotiem
- JRDataSourceProvider
- Hibernate datu avotiem

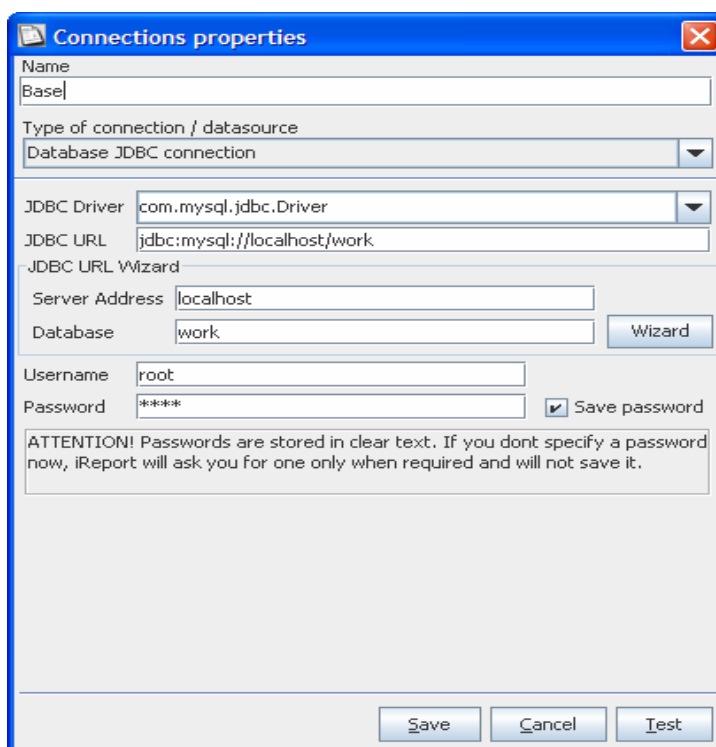
JDBC savienojumi tiek atvērti un padoti tieši JasperReports atskaites ģenerācijas laikā. XML datu avots ļauj ņemt datus no XML dokumentiem. CSV datu avots ļauj atvērt CSV (comma separated values – vērtības atdalītas ar komatiem) failu un izmantot atskaitē. JavaBean un JRDataSourceProvider ļauj drukāt datus no java klases un Hibernate datu avots dod iespēju izpildīt HQL pieprasījumus.

6.2 JDBC savienojums

JDBC savienojums ļauj izmantot relāciju datu bāzi, kā datu avotu, vai ja runājot vienkāršā valodā jebko, kas ir pieejams caur JDBC draiveri.

Kad atskaite ir izveidota ar JDBC savienojuma palīdzību, izstrādātājs definē SQL pieprasījumu, lai saņemtu datus no datu bāzes. Šis savienojums varbūt izmantots arī apakšatskaitē, tāpēc JasperReports novieto speciālo parametru *REPORT_CONNECTION java.sql.Connection* datu tipa, kurš varbūt izmantots izteiksmēs ar sintaksi *\$P{ REPORT_CONNECTION }*. Šis parametrs satur *java.sql.Connection* klasi, kurš bija sūtīts JasperReports no aplikācijas.

Tā kā iReport programma ir atskaites izstrādes vide, tad tā ļauj nedefinēt savienojumu caur interfeisu:



Attēls 20.

Šeit ir jānorāda savienojuma vārdu, tipu, JDBC draiveri, URL, lietotāju vārdu un paroli. Ja nospiežot pogu *Test* parādīsies izziņa, ka viss ir kārtība un šis savienojums eksistē, tad var turpināt darbu, pretēja gadījumā konfigurācijā ir kļūda.

6.3 JRDataSource interfeiss

Katrs JRDataSource implementē divas metodes:

```
public boolean next()
public Object getFieldValue(JRField jrField)
```

Pirmā metode pārvieto virtuālo kursoru vienmēr uz nākamo ierakstu, jo kā bija jau pateikts JRDataSource ir dati, kuri organizēti tabulās. Metode atgriež *patiesu* ja kursors bija tiek pārvietots nākamajā ieraksta un *aplamu*, ja ierakstu vairāk nav.

Katru reizi, kad JasperReports palaid *next*, visi lauki, kuri bija nedefinēti, aizpildās un izteiksmes tiek aprēķināti no jauna. Kad *next* atgriež *aplams* atskaite druka savas nobeiguma sekcijas (kājene). *Next* metode tiek izsaukta tik daudz reizes, cik ierakstu bija saņemts no datu avota.

Metode *getFieldValue* izsaukas uzreiz pēc *next*, ja tā rezultāts bija *paties* un atgriež vērtību, kura būs ierakstīta laukā.

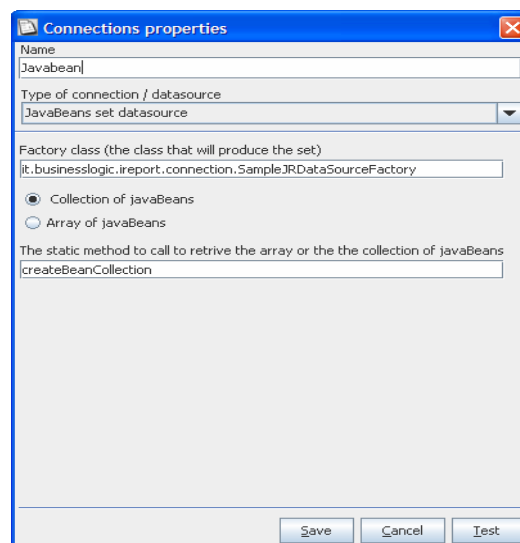
6.4 JavaBean datu avots

Šis datu avots ļauj izmantot JavaBeans, lai aizpildīt atskaiti ar datiem. JavaBean ir Java klasi, kura rāda „*getter*” metožu sērijas, kā:

```
public <returnType> getXXX(),
```

kur <returnType> (vērtība, kura būs atgriezta) ir primitīvs datu tips(int, double, utt.), vai objekts(String).

iReports arī piedāvā iespēju nedefinēt šo savienojumu, caur savu interfeisu:



Attēls 21.

JavaBean datu avots izmanto ārējo klasi (*Factory*), lai veidot objektus, kuri sastāda datus, kuri būs nosūtīti atskaitē. Klasē, kura būs izmantota, kā datu avots jāsaturs sevī statisko metodi, kura var veidot JavaBeans un atgriezt tos, kā kolekciju, vai masīvu.

Piemēram, dati tiek reprezentēti kā *Person* objekti:

```
public class Person
{
    private String name = "";
    private int age = 0;
    public Person(String name, int age)
    {
        this.name = name;
        this.age = age;
    }
    public int getAge()
    {
        return age;
    }
    public String getName()
    {
        return name;
    }
}
```

Izdruka 9.

Šajā gadījumā *Factory* klase izskatīsies sekojoši:

```
public class TestFactory
{
    public static java.util.Collection generateCollection()
    {
        java.util.Vector collection = new java.util.Vector();
        collection.add(new Person("Ted", 20) );
        collection.add(new Person("Jack", 34) );
        collection.add(new Person("Bob", 56) );
        collection.add(new Person("Alice", 12) );
        collection.add(new Person("Robin", 22) );
        return collection;
    }
}
```

Izdruka 10.

Tātad datu avots reprezentēs piecus JavaBeans *Person* tipa.

Šai konfigurācijai būs sekojoši parametri:

Vārds: Javabeen

Factory klase: TestFactory

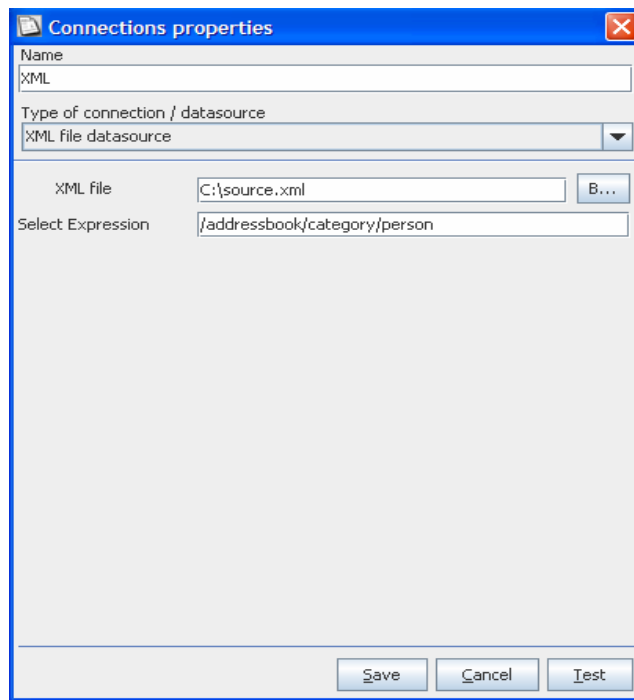
Metodi izsaukumam: generateCollection

Rezultāta tips: Collection of JavaBean

6.5 XML datu avots

XML dokuments atbilst koka struktūrai, tāpēc nepieciešams zināt kā un kādus mezglus būs jāizvēlas no tā un attēlot tos ierakstos. Lai to izdarīt tiek izmantota XPath izteiksme (XML Path valoda). XPath specifikācijas var atrast pēc adreses <http://www.w3.org/TR/xpath>.

Lai nodefinēt XML datu avotu, iReportā jāizpilda sekojošo formu:



Attēls 22.

Šeit vajag norādīt ceļu pie XML faila un XPath izteiksmi.

Piemēram, ir source.xml fails, kuram ir sekojoša struktūra:

```

<addressbook>
  <category name="home">
    <person id="1">
      <lastname>Davolio</lastname>
      <firstname>Nancy</firstname>
    </person>
    <person id="2">
      <lastname>Fuller</lastname>
      <firstname>Andrew</firstname>
    </person>
    <person id="3">
      <lastname>Leverling</lastname>
    </person>
  </category>
  <category name="work">
    <person id="4">
      <lastname>Peacock</lastname>
      <firstname>Margaret</firstname>
    </person>
  </category>
</addressbook>

```

Izruka 11.

Šis fails attēlo piezīmju grāmatiņu. Lai izvēlēties visas personas būs jāizmanto **/addressbook/category/person** izteiksmi. Rezultāts būs sekojošs:

```

<person id="1">
  <lastname>Davolio</lastname>
  <firstname>Nancy</firstname>
</person>
<person id="2">
  <lastname>Fuller</lastname>
  <firstname>Andrew</firstname>
</person>

```

```
<person id="3">
  <lastname>Leverling</lastname>
</person>
<person id="4">
  <lastname>Peacock</lastname>
  <firstname>Margaret</firstname>
</person>
```

Izdruka 12.

Ja, piemēram, būs nepieciešams izvēlēties personas no darba, būs jāizmanto `/addressbook/category[@name = "work"]/person` un rezultāts būs:

```
<person id="4">
  <lastname>Peacock</lastname>
  <firstname>Margaret</firstname>
</person>
```

Izdruka 13.

6.6 Tukšais datu avots

JasperReports piedāvā arī tukšo datu avotu. Šis avots atgriež *patiess* uz *next* metodi un *null* uz *getFieldValue*. Vienkārši sakot šis avots derīgs pirmajos izstrādes posmos, kad datu bāzes, vai datu avota vēl nav, bet ļoti vajag apskatīt darba rezultātus. Pēc izpildes parādās atskaite bez kādiem datiem, tāpat, kā attēlā 5.

7. Atskaites klase (skriptlets)

7.1 JRAbstractScriptlet klase

Skriptlets ir Java klase, kura izmantota, lai izpildīt kādus darbības atskaites drukāšanas laikā. Skriptlets sastāv no metodēm, kuri tiek palaisti kopā ar specialām atskaites notikumiem, tādiem kā jaunas lapaspuses radīšana, vai detaļas rindas apstrādāšanas beigas. JasperReports skriptlets paplašina *net.sf.jasperreports.engine.JRAbstractScriptlet* klasi. Šī klasi iekļauj sevī visus metodes, kuri palīdz pārvaldīt notikumus, kuri notiek atskaites izpildes laikā. Pēc noklusēšanas JasperReports skriptlets (JRDefaultScriptlet) izskatīsies sekojoši:

```
package net.sf.jasperreports.engine;

public class JRDefaultScriptlet extends JRAbstractScriptlet
{
    public JRDefaultScriptlet() { }
    public void beforeReportInit() throws JRScriptletException
    {}
    public void afterReportInit() throws JRScriptletException
    {}
    public void beforePageInit() throws JRScriptletException
    {}
    public void afterPageInit() throws JRScriptletException
    {}
    public void beforeColumnInit() throws JRScriptletException
    {}
}
```

```

    public void afterColumnInit() throws JRScriptletException
    {}
    public void beforeGroupInit(String groupName) throws
    JRScriptletException
    {}
    public void afterGroupInit(String groupName) throws
    JRScriptletException
    {}
    public void beforeDetailEval() throws JRScriptletException
    {}
    public void afterDetailEval() throws JRScriptletException
    {}
}

```

Izdruka 14.

Kā redzams tas sastāv no metožu salikuma ar vārdiem, kuri sastāv no atslēgvārdiem **after** (pēc) un **before** (pirms), pēc kuriem seko notikuma, vai darbības nosaukums (DetailEval vai PageInit).

Katra metode varbūt palaista tieša laikā:

- *Before Report Init* Metode būs izsaukta pirms atskaites inicializācijas
- *After Report Init* Metode būs izsaukta pēc mainīgas inicializācijas
- *Before Page Init* Metode būs izsaukta pirms jaunas lapaspuses inicializācijas
- *After Page Init* Metode būs izsaukta pēc jaunas lapaspuses inicializācijas
- *Before Column Init* Metode būs izsaukta pirms jaunas kolonas inicializācijas
- *After Column Init* Metode būs izsaukta pēc jaunas kolonas inicializācijas
- *Before Group <x> Init* Metode būs izsaukta pirms *x* grupas inicializācijas
- *After Group <x> Init* Metode būs izsaukta pēc *x* grupas inicializācijas
- *Before Detail Eval* Metode būs izsaukta pirms detaļu sekcijas drukāšanas
- *After Detail Eval* Metode būs izsaukta pēc detaļu sekcijas drukāšanas

8. Risinājumi vispārējam problēmām

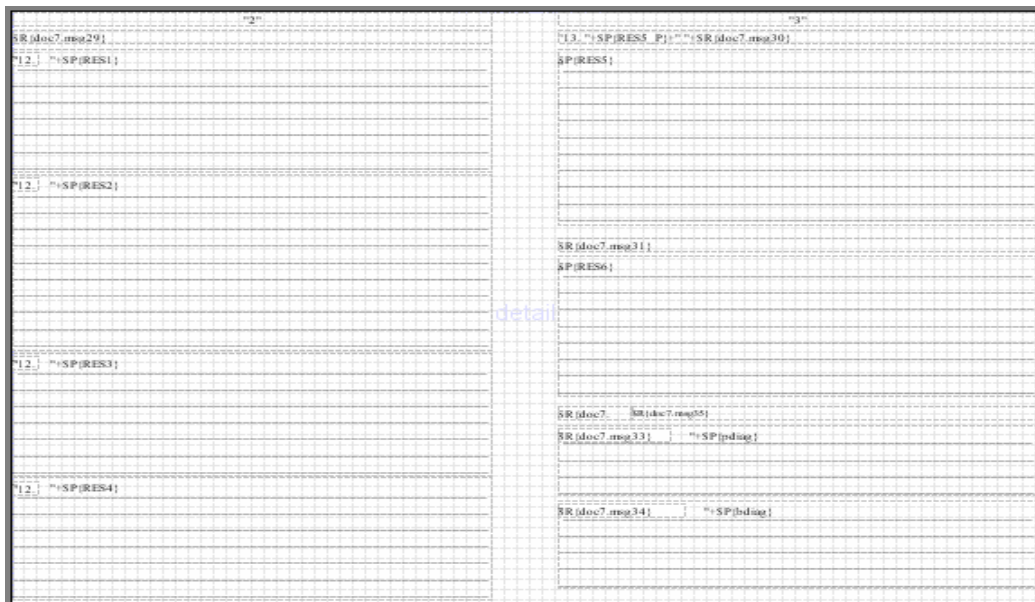
8.1 Procentu apstrāde

Bieži veidojot atskaites vajag skaitīt procentus. Piemēram, ir tabula:

A	B
10	20%
25	50%
15	30%

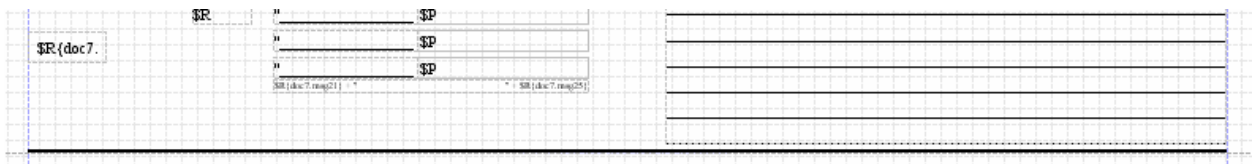
Tabula 1.

A kolona satur skaitļus un **B** cik procentus no kopsummas katrs reprezentē. No datu avota iespējams saņemt tikai **A** kolonas rezultātus, bet **B** ir jāaprēķina. To var izdarīt pēc formulas $B = (\Sigma A * 100) / A$, kur ΣA ir kolonas **A** visu vērtību summa. To nav iespējams izdarīt



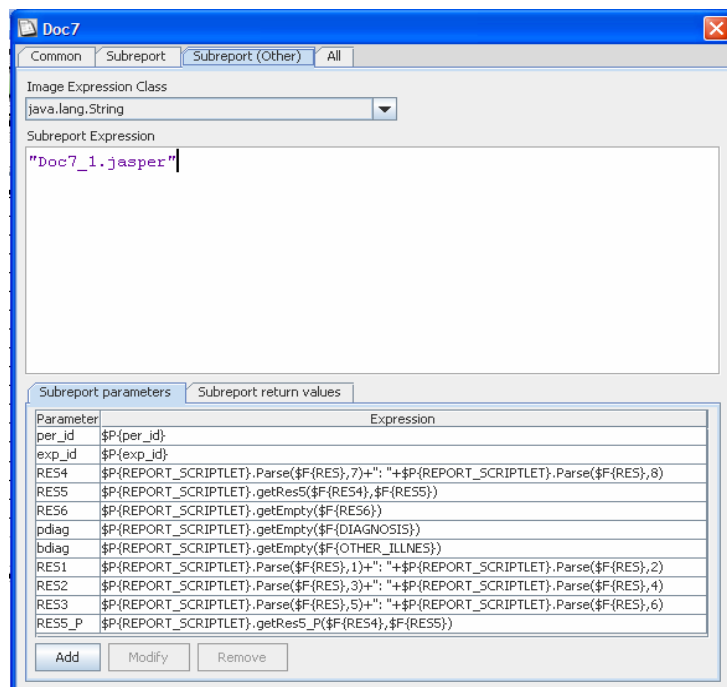
Attēls 24.

Nodefinējot un nopozicionējot visus laukus, pirmajā lapaspuses leņā (galvenā atskaite) bija izveidots apakšatskaite elements, kurš atbilst otrai lapaspusei (apakšatskaite):



Attēls 25.

Galvenā atskaitē bija SQL pieprasījums, kurš atgriezta visas nepieciešamo informāciju, ar kuras daļu bija aizpildīta pirmā lapa, un ar otro - otra lapa. Informācija līdz apakšatskaitei bija nosūtīta, kā parametri:



Attēls 26.

Vajag pievērst uzmanību, ka apakšatskaitei nav malu, tas ir tāpēc, ka tā tiek ielikta iekša galvenā atskaitē, tātad otrā lappusē malas būs no pirmās.

8.3 Hierarhiskas atskaites

Izstrādājot „Invaliditātes informatīvu sistēmu ekspertīžu un invaliditātes apliecību uzskaiti” vajadzēja izveidot atskaiti, kā attēlā 27:

Pirmreizējā/Atkārtotā invaliditāte pēc vecuma, slimību diagnozes, lokalizācijas, invaliditātes cēloņa, sociālajām kategorijām, dzimuma un invaliditātes grupām 2003.g. - 2004.g.par periodu no 01/01/2004 līdz 01/01/2004

Vecums: 16-39;

Invaliditātes cēlonis: Saslimšana (sakropojums), saistīta ar dienestu Nacionālajos bruņotajos spēkos vai Aizsardzības ministrijā;

Operēti pamatsimības dēļ

Npk.	Slimības diagnoze	Statistiskās klasifikācijas kods	Kopā	Strādājošie						Netrādājošie					
				Vīrieši	Sievietes	I grupa	II grupa	III grupa	Vīrieši	Sievietes	I grupa	II grupa	III grupa		
				2003	2004	2003	2004	2003	2004	2003	2004	2003	2004	2003	2004
Kopā															
1	Tuberkuloze	A15-A19;B90													
	-- Elpošanas orgānu tuberkuloze	A15-A16													
	-- Tuberkulozes sekas	B90													
2	Laundabīgie audzēji	C00-C80; D00-D09													
	-- Kuņģa laundabīgais audzējs	C16													
	-- Balsenes laundabīgais audzējs	C32													
	-- Melnoma un citi laundabīgi ādas audzēji	C43-C44													
5	Endokrīnās, uzturiskās un vielmaiņas slimības	E00-E90													
	-- Vairoģdziedzeru funkcijas traucējumi	E00-E07													
	-- Insulīnkarīgs cukura diabēts	E10													
	-- Vielmaiņas traucējumi	E70-E90													

o

Attēls 27.

Analizējot attēlu 27, var secināt, ka vajag divas grupas: viena pirmajam hierarhijas līmenim un otra otrajam. Otrā hierarhijas līmeni vajag nopozicionēt detaļu sekcijā, jo redzams, ka drukāšana atkārtas vairākas reizes. Priekš pirmās vajag izveidot jauno grupu:

Attēls 28.

Jaunajā grupā norādīja, ka tā būs inicializēta ar katru jaunu $\$F\{ID\}$, šis lauks atbilst katram pirmā hierarhijas līmeņa ierakstam. Rezultāta bija izveidots sekojošs šablons:

$\$P\{REPORT_SCRIPTLET\} Inv(\$P\{34\})+'' ''+$ $\$R\{Report38.Header01\}+'' ''+$																					
$SR\{SP:REPORT_SCRIPTLET\} Vecums(SP\{age\})$ $SR\{Report38.Header06\} SF\{126\}$ $SR\{Report38.Header07\} SP$																					
SR Report38 - Head	SR Report38. Head02	SR Report38. Head03	SR Report38. Head05												SR Report38. Head06						
			SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP
SR Report38. Head04																					
SF {CO}	''+SF{NAME1}	SF{SSK1}	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP
	''+SF{NAME2}	SF{SSK2}	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP	SP
SV {PAGE NUMBER}																					

Attēls 29.

Izpildot šo atskaiti izvada parādījās tāda paša tabula, kura bija programmatūras prasībās (Attēls 27).

8.4 Apakšatskaite un skriptlets

Izstrādājot „Invaliditātes informatīvu sistēmu ekspertīžu un invaliditātes apliecību uzskaitē” bija atrasta kļūda JasperReports bibliotēkā. Ja, piemēram, ir galvenā atskaite, apakšatskaite un skriptlets. Situācija prasa nosūtītus datus apakšatskaitē, kā parametrus apstrādāt ar kādu metodi no klases un rezultātu izdrukāt. Bet izrādījās, ka apakšatskaite nekā negrib strādāt ar skriptletu. Visas darbības ar klasi var notikt tikai galvenā atskaitē. Problēma bija risināta sekojoši: apakšatskaites parametriem bija piešķirti metožu darbības rezultāti, tāpat, ka attēla 26.

8.5 Lauku reģistrācija

Kā bija pateikts iReport piedāvā SQL pieprasījuma automātisku lauku reģistrāciju. Tas notiek tad, kad šis pieprasījums tiek sekmīgi palaist. Problēma radās, kad vajag strādāt ar SQL pieprasījumu, kur ir parametrs ($\$P\{parametra_nosaukums\}$). Ar to pieprasījums netiek palaists, jo parametram nav nekādas vērtības. Problēma tiek risināta ar to, ka parametram vajag piešķirt kādu vērtību pēc noklusēšanas, tad viss strādās tāpat, kā attēlā 18.

9. Noslēgums

Var secināt, ka JasperReports ir patiešām ļoti jaudīgs instruments informatīvas sistēmas atskaites izstrādē. Pozitīvo iespaidi dod tas, ka šis produkts ir bezmaksas un to var lietot jebkur gribas. Pie negatīvam pusēm var atnest to, ka JasperReports, tāpat kā iReport Designer ir maksas dokumentācija. Tā informācija, kas ir oficiālajā lapaspusēs bieži nav vērtīga, lai norisināt sarežģītus problēmus. Protams, var atrast alternatīvu ar to, ka kopā ar bibliotēku ielādas arī atskaites piemēri, bet joprojām tas nav tas pats kā dokumentācija. Tāpēc bakalaura darba veidošanas gaitā bija izmantots tik maz literatūras, viss, kas bija aprakstīts balstas uz autora pieredzes un praktiskas zināšanas.

Tāpat kopā ar bibliotēku iet arī izējas kods un ja ir nepieciešams, to var modificēt pēc savam vajadzībam. Bet strādāt ar to teksta redaktora praktiski nav iespējams, jo tas aizņems daudz laika. Šeit ļoti derīga ir iReport Designer programma, kura arī ir bezmaksas, ar izejas kodu un maksas dokumentāciju, bet ar ļoti sapratamo interfeisu, bet kad vajag veidot sarežģītas atskaites, piemēram, 1000 šūņu tabulu, tas strāda ļoti lēni, bet tā kā tas ir vienīgais bezmaksas redaktors, kurš strāda ar JasperReports bibliotēku tad uz to var aizvērt acis.

Latvijā šie produkti ir izmantoti „Invaliditātes informatīvās sistēmas ekspertīžu un invaliditātes apliecību uzskaitē” un „Zālū valsts aģentūras informatīvajā sistēmā” izstrādē, kurus veido SIA Mebius IT. Šī kompānija izmantos šo bakalaura darbu, ka mācību līdzekļi savos nākamajos projektos.

Izmantotas literatūras saraksts

1. JasperReports – Tutorial [tiešsaiste]. Pieejams Internetā:
<http://jasperreports.sourceforge.net/tutorial/index.html>
2. SourceForge.net: Forums for JasperReports - Java Reporting [tiešsaiste]. Pieejams Internetā:
http://sourceforge.net/forum/?group_id=36382
3. Introduction to iReport (free HTML manual) [tiešsaiste]. Pieejams Internetā:
http://ireport.sourceforge.net/index.php?p_module=page&pageid=11
4. A Tutorial on Generating Reports by iReport Designer for Jasper Reports and Reporting them in Java [tiešsaiste]. Pieejams Internetā:
<http://www.cise.ufl.edu/~otopsaka/CIS4301/ReportDemo/>
5. Open Source Java Reporting with JasperReports and iReport [tiešsaiste]. Pieejams Internetā:
<http://www.devx.com/Java/Article/29309>
6. Reports made easy with JasperReports [tiešsaiste]. Pieejams Internetā:
<http://www.javaworld.com/javaworld/jw-09-2002/jw-0920-opensourceprofile.html>
7. Ноутон П., Шилдт Г. Java™ 2, «БХВ-Петербург», Санкт – Петербург, 2005, 1050 стр.

Apliecinājums

Ar šo es apliecinu, ka šodien iesniegto bakalaura darbu es esmu veicis pašrocīgi un esmu izmantojis tikai tajā norādītos palīglīdzekļus.

Rīgā,

Paraksts:

Pielikums JasperReports DTD

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT jasperReport (property*, import*, reportFont*, style*, subDataset*, parameter*,
queryString?, field*, variable*, group*, background?, title?, pageHeader?, columnHeader?, detail?,
columnFooter?, pageFooter?, lastPageFooter?, summary?)>
<!ATTLIST jasperReport
    name NMTOKEN #REQUIRED
    language NMTOKEN "java"
    columnCount NMTOKEN "1"
    printOrder (Vertical | Horizontal) "Vertical"
    pageWidth NMTOKEN "595"
    pageHeight NMTOKEN "842"
    orientation (Portrait | Landscape) "Portrait"
    whenNoDataType (NoPages | BlankPage | AllSectionsNoDetail) "NoPages"
    columnWidth NMTOKEN "555"
    columnSpacing NMTOKEN "0"
    leftMargin NMTOKEN "20"
    rightMargin NMTOKEN "20"
    topMargin NMTOKEN "30"
    bottomMargin NMTOKEN "30"
    isTitleNewPage (true | false) "false"
    isSummaryNewPage (true | false) "false"
    isFloatColumnFooter (true | false) "false"
    scriptletClass NMTOKEN #IMPLIED
    resourceBundle CDATA #IMPLIED
    whenResourceMissingType (Null | Empty | Key | Error) "Null"
>

<!ELEMENT property EMPTY>
<!ATTLIST property
    name CDATA #REQUIRED
    value CDATA #IMPLIED
>

<!ELEMENT import EMPTY>
<!ATTLIST import
    value CDATA #REQUIRED
>

<!ELEMENT reportFont EMPTY>
<!ATTLIST reportFont
    name NMTOKEN #REQUIRED
    isDefault (true | false) "false"
    fontName CDATA "sansserif"
    size NMTOKEN "10"
    isBold (true | false) "false"
    isItalic (true | false) "false"
    isUnderline (true | false) "false"
    isStrikeThrough (true | false) "false"
    pdfFontName CDATA "Helvetica"
    pdfEncoding CDATA "CP1252"
    isPdfEmbedded (true | false) "false"
>

<!ELEMENT style EMPTY>
<!ATTLIST style
    name NMTOKEN #REQUIRED
    isDefault (true | false) "false"
    style NMTOKEN #IMPLIED
    mode (Opaque | Transparent) #IMPLIED
    forecolor CDATA #IMPLIED
    backcolor CDATA #IMPLIED
    pen (None | Thin | 1Point | 2Point | 4Point | Dotted) #IMPLIED
    fill (Solid) #IMPLIED
    radius NMTOKEN #IMPLIED
    scaleImage (Clip | FillFrame | RetainShape) #IMPLIED
    hAlign (Left | Center | Right | Justified) #IMPLIED
    vAlign (Top | Middle | Bottom) #IMPLIED
    border (None | Thin | 1Point | 2Point | 4Point | Dotted) #IMPLIED
    borderColor CDATA #IMPLIED
    padding NMTOKEN #IMPLIED
    topBorder (None | Thin | 1Point | 2Point | 4Point | Dotted) #IMPLIED
    topBorderColor CDATA #IMPLIED
    topPadding NMTOKEN #IMPLIED
    leftBorder (None | Thin | 1Point | 2Point | 4Point | Dotted) #IMPLIED
    leftBorderColor CDATA #IMPLIED
    leftPadding NMTOKEN #IMPLIED
```

```

bottomBorder (None | Thin | 1Point | 2Point | 4Point | Dotted) #IMPLIED
bottomBorderColor CDATA #IMPLIED
bottomPadding NMTOKEN #IMPLIED
rightBorder (None | Thin | 1Point | 2Point | 4Point | Dotted) #IMPLIED
rightBorderColor CDATA #IMPLIED
rightPadding NMTOKEN #IMPLIED
rotation (None | Left | Right) #IMPLIED
lineSpacing (Single | 1_1_2 | Double) #IMPLIED
isStyledText (true | false) #IMPLIED
fontName CDATA #IMPLIED
fontSize NMTOKEN #IMPLIED
isBold (true | false) #IMPLIED
isItalic (true | false) #IMPLIED
isUnderline (true | false) #IMPLIED
isStrikeThrough (true | false) #IMPLIED
pdfFontName CDATA #IMPLIED
pdfEncoding CDATA #IMPLIED
isPdfEmbedded (true | false) #IMPLIED
pattern CDATA #IMPLIED
isBlankWhenNull (true | false) #IMPLIED
>

<!ELEMENT subDataset (parameter*, queryString?, field*, variable*, group*)>
<!ATTLIST subDataset
    name NMTOKEN #REQUIRED
    scriptletClass NMTOKEN #IMPLIED
    resourceBundle CDATA #IMPLIED
    whenResourceMissingType (Null | Empty | Key | Error) "Null"
>

<!ELEMENT parameter (parameterDescription?, defaultValueExpression?)>
<!ATTLIST parameter
    name CDATA #REQUIRED
    class NMTOKEN "java.lang.String"
    isForPrompting (true | false) "true"
>

<!ELEMENT parameterDescription (#PCDATA)>
<!ELEMENT defaultValueExpression (#PCDATA)>

<!ELEMENT queryString (#PCDATA)>

<!ELEMENT field (fieldDescription?)>
<!ATTLIST field
    name CDATA #REQUIRED
    class NMTOKEN "java.lang.String"
>

<!ELEMENT fieldDescription (#PCDATA)>

<!ELEMENT variable (variableExpression?, initialValueExpression?)>
<!ATTLIST variable
    name CDATA #REQUIRED
    class NMTOKEN "java.lang.String"
    resetType (None | Report | Page | Column | Group) "Report"
    resetGroup CDATA #IMPLIED
    incrementType (None | Report | Page | Column | Group) "None"
    incrementGroup CDATA #IMPLIED
    calculation (Nothing | Count | Sum | Average | Lowest | Highest | StandardDeviation |
Variance | System | First) "Nothing"
    incrementerFactoryClass NMTOKEN #IMPLIED
>

<!ELEMENT variableExpression (#PCDATA)>
<!ELEMENT initialValueExpression (#PCDATA)>

<!ELEMENT group (groupExpression?, groupHeader?, groupFooter?)>
<!ATTLIST group
    name NMTOKEN #REQUIRED
    isStartNewColumn (true | false) "false"
    isStartNewPage (true | false) "false"
    isResetPageNumber (true | false) "false"
    isReprintHeaderOnEachPage (true | false) "false"
    minHeightToStartNewPage NMTOKEN "0"
>

<!ELEMENT groupExpression (#PCDATA)>
<!ELEMENT groupHeader (band?)>
<!ELEMENT groupFooter (band?)>

<!ELEMENT background (band?)>
<!ELEMENT title (band?)>
<!ELEMENT pageHeader (band?)>
<!ELEMENT columnHeader (band?)>

```

```

<!ELEMENT detail (band?)>
<!ELEMENT columnFooter (band?)>
<!ELEMENT pageFooter (band?)>
<!ELEMENT lastPageFooter (band?)>
<!ELEMENT summary (band?)>

<!ELEMENT band (printWhenExpression?, (line | rectangle | ellipse | image | staticText | textField
| subreport | pieChart | pie3DChart | barChart | bar3DChart | xyBarChart | stackedBarChart |
stackedBar3DChart| lineChart | xyLineChart | areaChart | xyAreaChart | scatterChart | bubbleChart
| timeSeriesChart | highLowChart | candlestickChart | elementGroup | crosstab | frame)*)>
<!ATTLIST band
    height NMTOKEN "0"
    isSplitAllowed (true | false) "true"
>

<!ELEMENT line (reportElement, graphicElement?)>
<!ATTLIST line
    direction (TopDown | BottomUp) "TopDown"
>

<!ELEMENT reportElement (printWhenExpression?)>
<!ATTLIST reportElement
    key NMTOKEN #IMPLIED
    style NMTOKEN #IMPLIED
    positionType (Float | FixRelativeToTop | FixRelativeToBottom) "FixRelativeToTop"
    stretchType (NoStretch | RelativeToTallestObject | RelativeToBandHeight) "NoStretch"
    isPrintRepeatedValues (true | false) "true"
    mode (Opaque | Transparent) #IMPLIED
    x NMTOKEN #REQUIRED
    y NMTOKEN #REQUIRED
    width NMTOKEN #REQUIRED
    height NMTOKEN #REQUIRED
    isRemoveLineWhenBlank (true | false) "false"
    isPrintInFirstWholeBand (true | false) "false"
    isPrintWhenDetailOverflows (true | false) "false"
    printWhenGroupChanges CDATA #IMPLIED
    forecolor CDATA #IMPLIED
    backcolor CDATA #IMPLIED
>
<!ELEMENT printWhenExpression (#PCDATA)>

<!ELEMENT graphicElement EMPTY>
<!ATTLIST graphicElement
    stretchType (NoStretch | RelativeToTallestObject | RelativeToBandHeight) #IMPLIED
    pen (None | Thin | 1Point | 2Point | 4Point | Dotted) #IMPLIED
    fill (Solid) #IMPLIED
>

<!ELEMENT rectangle (reportElement, graphicElement?)>
<!ATTLIST rectangle
    radius NMTOKEN #IMPLIED
>

<!ELEMENT ellipse (reportElement, graphicElement?)>

<!ELEMENT image (reportElement, box?, graphicElement?, imageExpression?, anchorNameExpression?,
hyperlinkReferenceExpression?, hyperlinkAnchorExpression?, hyperlinkPageExpression?)>
<!ATTLIST image
    scaleImage (Clip | FillFrame | RetainShape) #IMPLIED
    hAlign (Left | Center | Right) #IMPLIED
    vAlign (Top | Middle | Bottom) #IMPLIED
    isUsingCache (true | false) #IMPLIED
    isLazy (true | false) "false"
    onErrorType (Error | Blank | Icon) "Error"
    evaluationTime (Now | Report | Page | Column | Group | Band) "Now"
    evaluationGroup CDATA #IMPLIED
    hyperlinkType (None | Reference | LocalAnchor | LocalPage | RemoteAnchor | RemotePage)
"None"
    hyperlinkTarget (Self | Blank) "Self"
    bookmarkLevel NMTOKEN "0"
>
<!ELEMENT imageExpression (#PCDATA)>
<!ATTLIST imageExpression
    class (java.lang.String | java.io.File | java.net.URL | java.io.InputStream |
java.awt.Image | net.sf.jasperreports.engine.JRRenderable) "java.lang.String"
>

<!ELEMENT box EMPTY>
<!ATTLIST box
    border (None | Thin | 1Point | 2Point | 4Point | Dotted) #IMPLIED

```

```

borderColor CDATA #IMPLIED
padding NMTOKEN #IMPLIED
topBorder (None | Thin | 1Point | 2Point | 4Point | Dotted) #IMPLIED
topBorderColor CDATA #IMPLIED
topPadding NMTOKEN #IMPLIED
leftBorder (None | Thin | 1Point | 2Point | 4Point | Dotted) #IMPLIED
leftBorderColor CDATA #IMPLIED
leftPadding NMTOKEN #IMPLIED
bottomBorder (None | Thin | 1Point | 2Point | 4Point | Dotted) #IMPLIED
bottomBorderColor CDATA #IMPLIED
bottomPadding NMTOKEN #IMPLIED
rightBorder (None | Thin | 1Point | 2Point | 4Point | Dotted) #IMPLIED
rightBorderColor CDATA #IMPLIED
rightPadding NMTOKEN #IMPLIED
>

<!ELEMENT anchorNameExpression (#PCDATA)>
<!ELEMENT hyperlinkReferenceExpression (#PCDATA)>
<!ELEMENT hyperlinkAnchorExpression (#PCDATA)>
<!ELEMENT hyperlinkPageExpression (#PCDATA)>

<!ELEMENT staticText (reportElement, box?, textElement?, text?)>
<!ELEMENT text (#PCDATA)>

<!ELEMENT textElement (font?)>
<!ATTLIST textElement
    textAlignment (Left | Center | Right | Justified) #IMPLIED
    verticalAlignment (Top | Middle | Bottom) #IMPLIED
    rotation (None | Left | Right) #IMPLIED
    lineSpacing (Single | 1_1_2 | Double) #IMPLIED
    isStyledText (true | false) #IMPLIED
>

<!ELEMENT font EMPTY>
<!ATTLIST font
    reportFont NMTOKEN #IMPLIED
    fontName CDATA #IMPLIED
    size NMTOKEN #IMPLIED
    isBold (true | false) #IMPLIED
    isItalic (true | false) #IMPLIED
    isUnderline (true | false) #IMPLIED
    isStrikeThrough (true | false) #IMPLIED
    pdfFontName CDATA #IMPLIED
    pdfEncoding CDATA #IMPLIED
    isPdfEmbedded (true | false) #IMPLIED
>

<!ELEMENT textField (reportElement, box?, textElement?, textFieldExpression?,
anchorNameExpression?, hyperlinkReferenceExpression?, hyperlinkAnchorExpression?,
hyperlinkPageExpression?)>
<!ATTLIST textField
    isStretchWithOverflow (true | false) "false"
    evaluationTime (Now | Report | Page | Column | Group | Band) "Now"
    evaluationGroup CDATA #IMPLIED
    pattern CDATA #IMPLIED
    isBlankWhenNull (true | false) #IMPLIED
    hyperlinkType (None | Reference | LocalAnchor | LocalPage | RemoteAnchor | RemotePage)
"None"
    hyperlinkTarget (Self | Blank) "Self"
    bookmarkLevel NMTOKEN "0"
>
<!ELEMENT textFieldExpression (#PCDATA)>
<!ATTLIST textFieldExpression
    class (java.lang.Boolean | java.lang.Byte | java.util.Date | java.sql.Timestamp |
java.sql.Time | java.lang.Double | java.lang.Float | java.lang.Integer | java.lang.Long |
java.lang.Short | java.math.BigDecimal | java.lang.Number | java.lang.String) "java.lang.String"
>

<!ELEMENT subreport (reportElement, parametersMapExpression?, subreportParameter*,
(connectionExpression | dataSourceExpression)?, returnValue*, subreportExpression?)>
<!ATTLIST subreport
    isUsingCache (true | false) #IMPLIED
>
<!ELEMENT parametersMapExpression (#PCDATA)>
<!ELEMENT subreportParameter (subreportParameterExpression?)>
<!ATTLIST subreportParameter
    name NMTOKEN #REQUIRED
>
<!ELEMENT subreportParameterExpression (#PCDATA)>

```

```

<!ELEMENT returnValue EMPTY>
<!ATTLIST returnValue
    subreportVariable NMTOKEN #IMPLIED
    toVariable NMTOKEN #IMPLIED
    calculation (Nothing | Count | Sum | Average | Lowest | Highest | StandardDeviation |
Variance | First) "Nothing"
    incrementerFactoryClass NMTOKEN #IMPLIED
>

<!ELEMENT connectionExpression (#PCDATA)>
<!ELEMENT dataSourceExpression (#PCDATA)>
<!ELEMENT subreportExpression (#PCDATA)>
<!ATTLIST subreportExpression
    class (java.lang.String | java.io.File | java.net.URL | java.io.InputStream |
net.sf.jasperreports.engine.JasperReport | dori.jasper.engine.JasperReport) "java.lang.String"
>

<!ELEMENT elementGroup (line | rectangle | ellipse | image | staticText | textField | subreport |
pieChart | pie3DChart | barChart | bar3DChart | xyBarChart | stackedBarChart | stackedBar3DChart |
lineChart | xyLineChart | areaChart | xyAreaChart | scatterChart | bubbleChart | timeSeriesChart |
highLowChart | candlestickChart | elementGroup | crosstab | frame)*>

<!ELEMENT chart (reportElement, box?, chartTitle?, chartSubtitle?, anchorNameExpression?,
hyperlinkReferenceExpression?, hyperlinkAnchorExpression?, hyperlinkPageExpression?)>
<!ATTLIST chart
    isShowLegend (true | false) "true"
    evaluationTime (Now | Report | Page | Column | Group | Band) "Now"
    evaluationGroup CDATA #IMPLIED
    hyperlinkType (None | Reference | LocalAnchor | LocalPage | RemoteAnchor | RemotePage)
"None"
    hyperlinkTarget (Self | Blank) "Self"
    bookmarkLevel NMTOKEN "0"
    customizerClass NMTOKEN #IMPLIED
>
<!ELEMENT chartTitle (font?, titleExpression?)>
<!ATTLIST chartTitle
    position (Top | Bottom | Left | Right) "Top"
    color CDATA #IMPLIED
>
<!ELEMENT titleExpression (#PCDATA)>
<!ELEMENT chartSubtitle (font?, subtitleExpression?)>
<!ATTLIST chartSubtitle
    color CDATA #IMPLIED
>
<!ELEMENT subtitleExpression (#PCDATA)>

<!ELEMENT pieChart (chart, pieDataset, piePlot)>
<!ELEMENT pieDataset (dataset?, keyExpression?, valueExpression?, labelExpression?)>
<!ELEMENT keyExpression (#PCDATA)>
<!ELEMENT valueExpression (#PCDATA)>
<!ELEMENT labelExpression (#PCDATA)>
<!ELEMENT piePlot (plot)>

<!ELEMENT dataset (datasetRun?)>
<!ATTLIST dataset
    resetType (None | Report | Page | Column | Group) "Report"
    resetGroup CDATA #IMPLIED
    incrementType (None | Report | Page | Column | Group) "None"
    incrementGroup CDATA #IMPLIED
>

<!ELEMENT datasetRun (parametersMapExpression?, datasetParameter*, (connectionExpression |
dataSourceExpression?)>
<!ATTLIST datasetRun
    subDataset NMTOKEN #REQUIRED
>

<!ELEMENT datasetParameter (datasetParameterExpression?)>
<!ATTLIST datasetParameter
    name NMTOKEN #REQUIRED
>

<!ELEMENT datasetParameterExpression (#PCDATA)>

<!ELEMENT plot EMPTY>
<!ATTLIST plot
    backcolor CDATA #IMPLIED
    orientation (Horizontal | Vertical) "Vertical"
    backgroundAlpha NMTOKEN "1"
    foregroundAlpha NMTOKEN "1"

```

```

>
<!ELEMENT pie3DChart (chart, pieDataset, pie3DPlot)>
<!ELEMENT pie3DPlot (plot)>
<!ATTLIST pie3DPlot
    depthFactor CDATA "0.2"
>

<!ELEMENT barChart (chart, categoryDataset, barPlot)>
<!ELEMENT categoryDataset (dataset?, categorySeries*)>
<!ELEMENT categorySeries (seriesExpression?, categoryExpression?, valueExpression?,
labelExpression?)>
<!ELEMENT seriesExpression (#PCDATA)>
<!ELEMENT categoryExpression (#PCDATA)>

<!ELEMENT barPlot (plot, categoryAxisLabelExpression?, valueAxisLabelExpression?)>
<!ATTLIST barPlot
    isShowLabels (true | false ) "false"
    isShowTickMarks (true | false) "true"
    isShowTickLabels (true | false) "true"
>
<!ELEMENT categoryAxisLabelExpression (#PCDATA)>
<!ELEMENT valueAxisLabelExpression (#PCDATA)>

<!ELEMENT bar3DChart (chart, categoryDataset, bar3DPlot)>
<!ELEMENT bar3DPlot (plot, categoryAxisLabelExpression?, valueAxisLabelExpression?)>
<!ATTLIST bar3DPlot
    isShowLabels (true | false ) "false"
    xOffset CDATA #IMPLIED
    yOffset CDATA #IMPLIED
>

<!ELEMENT xyBarChart (chart, (timePeriodDataset | timeSeriesDataset | xyDataset ), barPlot)>

<!ELEMENT timePeriodDataset (dataset?, timePeriodSeries*)>
<!ELEMENT timePeriodSeries (seriesExpression?, startDateExpression?, endDateExpression?,
valueExpression?, labelExpression?)>
<!ELEMENT startDateExpression (#PCDATA)>
<!ELEMENT endDateExpression (#PCDATA)>

<!ELEMENT stackedBarChart (chart, categoryDataset, barPlot)>

<!ELEMENT stackedBar3DChart (chart, categoryDataset, bar3DPlot)>

<!ELEMENT lineChart (chart, categoryDataset, linePlot)>
<!ELEMENT linePlot (plot, categoryAxisLabelExpression?, valueAxisLabelExpression?)>
<!ATTLIST linePlot
    isShowLines (true | false) "true"
    isShowShapes (true | false) "true"
>

<!ELEMENT xyLineChart (chart, xyDataset, linePlot)>
<!ELEMENT xyDataset (dataset?, xySeries*)>
<!ELEMENT xySeries (seriesExpression?, xValueExpression?, yValueExpression?, labelExpression?)>
<!ELEMENT xValueExpression (#PCDATA)>
<!ELEMENT yValueExpression (#PCDATA)>

<!ELEMENT areaChart (chart, categoryDataset, areaPlot)>
<!ELEMENT areaPlot (plot, categoryAxisLabelExpression?, valueAxisLabelExpression?)>

<!ELEMENT xyAreaChart (chart, xyDataset, areaPlot)>

<!ELEMENT scatterChart (chart, xyDataset, scatterPlot)>
<!ELEMENT scatterPlot (plot, xAxisLabelExpression?, yAxisLabelExpression?)>
<!ATTLIST scatterPlot
    isShowLines (true | false) "true"
    isShowShapes (true | false) "true"
>
<!ELEMENT xAxisLabelExpression (#PCDATA)>
<!ELEMENT yAxisLabelExpression (#PCDATA)>

<!ELEMENT bubbleChart (chart, xyzDataset, bubblePlot)>
<!ELEMENT xyzDataset (dataset?, xyzSeries*)>
<!ELEMENT xyzSeries (seriesExpression?, xValueExpression?, yValueExpression?, zValueExpression?)>
<!ELEMENT bubblePlot (plot, xAxisLabelExpression?, yAxisLabelExpression?)>
<!ATTLIST bubblePlot
    scaleType (BothAxes | DomainAxis | RangeAxis) "RangeAxis"
>
<!ELEMENT zValueExpression (#PCDATA)>

```

```

<!ELEMENT timeSeriesDataset (dataset?, timeSeries*)>
<!ATTLIST timeSeriesDataset
    timePeriod (Year | Quarter | Month | Week | Day | Hour | Minute | Second | Milisecond )
    "Day"
>
<!ELEMENT timeSeries (seriesExpression?, timePeriodExpression?, valueExpression?,
labelExpression?)>
<!ELEMENT timePeriodExpression (#PCDATA)>

<!ELEMENT timeSeriesChart (chart, timeSeriesDataset, timeSeriesPlot)>
<!ELEMENT timeSeriesPlot (plot, timeAxisLabelExpression?, valueAxisLabelExpression?)>
<!ATTLIST timeSeriesPlot
    isShowLines (true | false) "true"
    isShowShapes (true | false) "true"
>
<!ELEMENT timeAxisLabelExpression (#PCDATA)>

<!ELEMENT highLowChart (chart, highLowDataset, highLowPlot)>
<!ELEMENT highLowDataset (dataset?, seriesExpression?, dateExpression?, highExpression?,
lowExpression?, openExpression?, closeExpression?, volumeExpression?)>
<!ELEMENT highLowPlot (plot, timeAxisLabelExpression?, valueAxisLabelExpression?)>
<!ATTLIST highLowPlot
    isShowCloseTicks (true | false) "true"
    isShowOpenTicks (true | false) "true"
>
<!ELEMENT dateExpression (#PCDATA)>
<!ELEMENT highExpression (#PCDATA)>
<!ELEMENT lowExpression (#PCDATA)>
<!ELEMENT openExpression (#PCDATA)>
<!ELEMENT closeExpression (#PCDATA)>
<!ELEMENT volumeExpression (#PCDATA)>

<!ELEMENT candlestickChart (chart, highLowDataset, candlestickPlot)>
<!ELEMENT candlestickPlot (plot, timeAxisLabelExpression?, valueAxisLabelExpression?)>
<!ATTLIST candlestickPlot
    isShowVolume (true | false) "true"
>

<!ELEMENT crosstab (reportElement, crosstabParameter*, parametersMapExpression?, crosstabDataset,
rowGroup*, columnGroup*, measure*, crosstabCell*, whenNoDataCell?)>
<!ATTLIST crosstab
    name NMTOKEN #REQUIRED
    isRepeatColumnHeaders (true | false) "true"
    isRepeatRowHeaders (true | false) "true"
    columnBreakOffset NMTOKEN "10"
>

<!ELEMENT crosstabParameter (parameterValueExpression?)>
<!ATTLIST crosstabParameter
    name CDATA #REQUIRED
    class NMTOKEN "java.lang.String"
>

<!ELEMENT parameterValueExpression (#PCDATA)>

<!ELEMENT crosstabDataset (dataset)>
<!ATTLIST crosstabDataset
    isDataPreSorted (true | false) "false"
>

<!ELEMENT rowGroup (bucket, crosstabRowHeader?, crosstabTotalRowHeader?)>
<!ATTLIST rowGroup
    name NMTOKEN #REQUIRED
    width NMTOKEN #REQUIRED
    totalPosition (Start | End | None) "None"
    headerPosition (Top | Middle | Bottom | Stretch) "Top"
>

<!ELEMENT crosstabRowHeader (cellContents?)>

<!ELEMENT crosstabTotalRowHeader (cellContents?)>

<!ELEMENT columnGroup (bucket, crosstabColumnHeader?, crosstabTotalColumnHeader?)>
<!ATTLIST columnGroup
    name NMTOKEN #REQUIRED
    height NMTOKEN #REQUIRED
    totalPosition (Start | End | None) "None"
    headerPosition (Left | Center | Right | Stretch) "Left"
>

```

```

<!ELEMENT crosstabColumnHeader (cellContents?)>

<!ELEMENT crosstabTotalColumnHeader (cellContents?)>

<!ELEMENT bucket (bucketExpression, comparatorExpression?)>
<!ATTLIST bucket
    order (Ascending | Descending) "Ascending"
>

<!ELEMENT bucketExpression (#PCDATA)>
<!ATTLIST bucketExpression
    class NMTOKEN #REQUIRED
>

<!ELEMENT comparatorExpression (#PCDATA)>

<!ELEMENT measure (measureExpression)>
<!ATTLIST measure
    name NMTOKEN #REQUIRED
    class NMTOKEN #REQUIRED
    calculation (Nothing | Count | Sum | Average | Lowest | Highest | StandardDeviation |
Variance | First) "Nothing"
    incrementerFactoryClass NMTOKEN #IMPLIED
    percentageOf (None | GrandTotal) "None"
    percentageCalculatorClass NMTOKEN #IMPLIED
>

<!ELEMENT measureExpression (#PCDATA)>

<!ELEMENT crosstabCell (cellContents?)>
<!ATTLIST crosstabCell
    width NMTOKEN #IMPLIED
    height NMTOKEN #IMPLIED
    rowTotalGroup NMTOKEN #IMPLIED
    columnTotalGroup NMTOKEN #IMPLIED
>

<!ELEMENT cellContents (box?, (line | rectangle | ellipse | image | staticText | textField |
elementGroup | frame)*)>
<!ATTLIST cellContents
    bgcolor CDATA #IMPLIED
>

<!ELEMENT whenNoDataCell (cellContents)>

<!ELEMENT frame (reportElement, box?, (line | rectangle | ellipse | image | staticText | textField
| subreport | pieChart | pie3DChart | barChart | bar3DChart | xyBarChart | stackedBarChart |
stackedBar3DChart| lineChart | xyLineChart | areaChart | xyAreaChart | scatterChart | bubbleChart
| timeSeriesChart | highLowChart | candlestickChart | elementGroup | crosstab | frame)*)>

```

Bakalaura darbs izstrādāts

LU Datorikas nodaļā

Autors:

Fizikas un matemātikas

fakultātes students

St. apl. Nr. DatZ 020092

.....

Aleksandrs Dubrovskis

2006. g. jūnijā.

Darba vadītājs:

analītiskis programmētājs,

Accenture Technology Solution, Dr. sc. comp.

Kalvis Apsītis

LU MII

.....

Recenzents:

asistents, Mg. sc. comp Dainis Dosbergs

LU Fizikas un matemātikas fakultāte

.....

Darbs iesniegts Datorikas nodaļā

2006. g. jūnijā.

Pieņēma sekretāre

Aizstāvēts datorzinātņu bakalaura pārbaudījumu komisijas sēdē

2006.g.ar atzīmi.

Protokols Nr. _____

Bakalaura pārbaudījumu komisijas sekretārs:

doc. Meds Mediņš