**Министерство науки и высшего образования Российской Федерации**

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«Владимирский государственный университет**

**имени Александра Григорьевича и Николая Григорьевича Столетовых»**

**(ВлГУ)**

Кафедра информационных систем и программной инженерии

**Лабораторная работа №1**

**ОПРЕДЕЛЕНИЕ СТРУКТУРЫ XML-ДОКУМЕНТА СРЕДСТВАМИ DTD И XML-СХЕМ**

Выполнил: студент группы ПРИм-121

Евстигнеев Андрей Васильевич

Проверил: преподаватель кафедры ИСПИ

Спирин Иван Вячеславович

Владимир 2022

**ЦЕЛЬ РАБОТЫ**

Разработать модель предметной области в виде диаграммы классов UML. Описать средствами DTD и в виде XML-схемы, привести пример XML-документов, соответствующих DTD-описанию и схеме.

**ХОД РАБОТЫ**

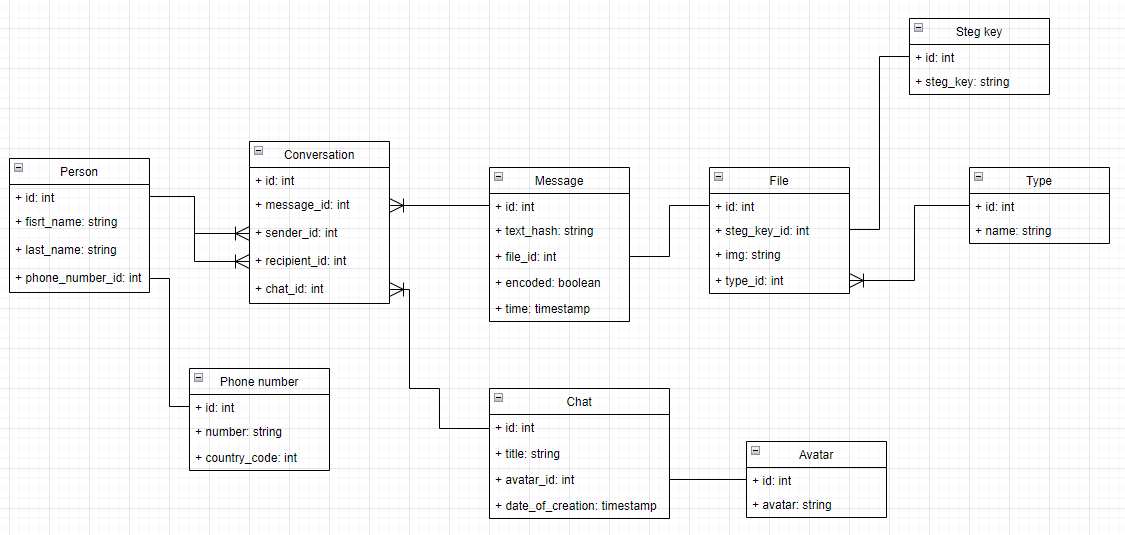
1. **Предметная область**:

Стеганография – это наука о скрытой передаче информации, в которой скрывается сам факт её существования. Целью ВКР является теоретическое и экспериментальное исследование методов и алгоритмов анализа цифровых изображений для задач стеганографии. В качестве главной задачи является исследование методов внедрения информации в цифровые изображения, а также модификация для затруднения их обнаружения.

Если у юзера стоит цель закодировать информацию в изображение, то графический файл попадает к нему любым способом (хоть на флешке). Потом юзер обрабатывает его своим ПО. В дальнейшем он может передать это изображение либо через чат в мессенджере, либо по почте, либо также на флешке. ПО могут воспользоваться люди из структуры, чтобы скрыть информацию от посторонних лиц.

Если рассмотреть на примере чата, то чат собирает в себе массив всех сообщений между двумя лицами: закодированных и не закодированных. У каждого сообщения есть автор и получатель. Если сообщение закодированное, то это будет изображение.

2. **UML-диаграмма**:



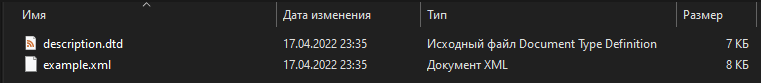
Модель данных содержит 5 взаимосвязанных сущностей (документов): Person, Conversation, Message, File и Chat. Кроме сущностей присутствуют 4 вспомогательных класса, описывающих комплексные типы данных: Phone number, Steg key, Type и Avatar.

Между классами Person и Message определено отношение типа «многие-ко-многим» через промежуточную таблицу Conversation.

3. **DTD-описание**:

Технические требования для DTD-описания:

1) должно быть выполнено в отдельном внешнем файле, а не непосредственно в рамках документа-примера



2) документ-пример должен содержать ссылку на DTD-описание

<!DOCTYPE System SYSTEM "description.dtd">

3) DTD-описание должно демонстрировать использование конструкций описания элементов с вложенными элементами и применением следования, альтернативы, необязательности и повторяемости этих вложенных элементов

<!ELEMENT person (firstName,(lastName)?,pnoneNumber)>

<!ELEMENT personList (person)+>

<!ELEMENT content ((file)|(text))>

<!ELEMENT messageList (message)\*>

4) DTD-описание должно демонстрировать использование конструкций описания атрибутов, как обязательных, так и необязательных, в том числе со значением по умолчанию для атрибутов следующих типов: ID, IDREF или IDREFS, CDATA, а также одного из перечисляемых типов

<!ATTLIST person id ID #REQUIRED>

<!ATTLIST file

    type (bmp|jpeg|gif|png) #REQUIRED

<!ATTLIST sender idref IDREF #REQUIRED>

<!ATTLIST System xmlns CDATA #REQUIRED>

5) DTD-описание должно демонстрировать использование параметризованных сущностей и ссылок на них

<!ENTITY % title "<!ELEMENT title (#PCDATA)>">

%title;

6) DTD-описание должно демонстрировать использование общих сущностей и ссылок на них в документе-примере

<!ENTITY andreyName "Andrey">

В документе-примере: <firstName>&andreyName;</firstName>

4. **XML-схема**:

Технические требования для XML-схемы:

1) схеме должно быть сопоставлено определенное пространство имен, при этом должна быть предусмотрена квалификация элементов, определенных схемой, в документе, построенном на ее основе

<xs:schema xmlns:sy="http://namespaceGrandvil.com/System" targetNamespace="http://namespaceGrandvil.com/System" xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

2) документ-пример должен содержать ссылку на XML-схему

<System xmlns="http://namespaceGrandvil.com/System" xsi:schemaLocation="http://namespaceGrandvil.com/System XMLScheme.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

3) схема должна демонстрировать использование конструкций для описания сложного типа, образованного вложенными элементами и атрибутами; при этом должен быть хотя бы один сложный тип, хотя бы один вложенный элемент которого был бы также сложного типа; при описании сложных типов необходимо продемонстрировать использование не менее двух видов модельных групп (sequence, choice и all)

<xs:complexType name="PersonRef">

        <xs:attribute name="idref" type="xs:unsignedByte" use="required" />

    </xs:complexType>

    <xs:complexType name="PersonsRef">

        <xs:sequence>

            <xs:element maxOccurs="unbounded" name="personRef" type="sy:PersonRef" />

        </xs:sequence>

    </xs:complexType>

<xs:element name="System">

        <xs:complexType>

            <xs:sequence>

                <xs:element name="personList" type="sy:PersonList" />

                <xs:element name="chatHistory" type="sy:ChatHistory" />

            </xs:sequence>

        </xs:complexType>

<xs:complexType name="PersonList">

        <xs:sequence>

            <xs:element maxOccurs="unbounded" name="person" type="sy:Person" />

        </xs:sequence>

    </xs:complexType>

<xs:complexType name="Content">

        <xs:choice minOccurs="1" maxOccurs="1">

            <xs:element name="file" type="sy:File" />

            <xs:element name="text" type="xs:string" />

        </xs:choice>

    </xs:complexType>

<xs:complexType name="Message">

        <xs:all>

            <xs:element name="sender" type="sy:PersonRef" />

            <xs:element name="receivers" type="sy:PersonsRef" />

            <xs:element name="content" type="sy:Content" />

            <xs:element name="time" type="xs:dateTime" />

        </xs:all>

    </xs:complexType>

4) схема должна содержать описание абстрактного сложного типа и элемента этого типа, а пример документа - демонстрировать использование последнего

<xs:complexType name="AbstractChat" abstract="true">

        <xs:sequence>

            <xs:element name="messageList" type="sy:MessageList" />

        </xs:sequence>

    </xs:complexType>

    <xs:complexType name="Chat">

        <xs:complexContent>

            <xs:extension base="sy:AbstractChat">

                <xs:sequence>

                    <xs:element name="type" type="sy:Type" />

                </xs:sequence>

                <xs:attribute name="id" type="xs:unsignedByte" use="required" />

            </xs:extension>

        </xs:complexContent>

    </xs:complexType>

5) схема должна демонстрировать возможность описания нового сложного типа на базе как простого типа, так и сложного как путем расширения, так и путем ограничения

<xs:complexType name="File">

        <xs:simpleContent>

            <xs:extension base="xs:string">

                <xs:attribute name="type" type="sy:MIMEType" use="required" />

                <xs:attribute name="messageEncoded" type="xs:boolean" use="required" />

            </xs:extension>

        </xs:simpleContent>

    </xs:complexType>

<xs:simpleType name="MIMEType">

        <xs:restriction base="xs:string">

            <xs:enumeration value="image/bmp" />

            <xs:enumeration value="image/jpeg" />

            <xs:enumeration value="image/gif" />

            <xs:enumeration value="image/png" />

        </xs:restriction>

    </xs:simpleType>

6) схема должна демонстрировать возможности описания новых простых типов на базе существующих (в том числе описание перечислимого типа) с использованием не менее трех видов фасетов

<xs:simpleType name="MIMEType">

        <xs:restriction base="xs:string">

            <xs:enumeration value="image/bmp" />

            <xs:enumeration value="image/jpeg" />

            <xs:enumeration value="image/gif" />

            <xs:enumeration value="image/png" />

        </xs:restriction>

    </xs:simpleType>

7) схема должна содержать описание ограничения уникальности, ключа и ссылки на ключ, применение чего должно иметь смысл в предметной области

<xs:key name="personID">

            <xs:selector xpath=".//person" />

            <xs:field xpath="@id" />

</xs:key>

<xs:keyref name="personIdref" refer="sy:personID">

    <xs:selector xpath=".//personRef" />

    <xs:field xpath="@idref" />

</xs:keyref>