МИНОБРНАУКИ РОССИИ

федеральное государственное автономное образовательное учреждение   
высшего образования

**«Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В.И. Ульянова (Ленина)»**

**(СПбГЭТУ «ЛЭТИ»)**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кафедра МОЭВМ

**Человеко-машинный интерфейс**

Выполнили:

Диогенова К.А.

Казачкова А.Д.

Сабынин Е.С.

Факультет: КТИ

Группа: 3304

Преподаватель: Первицкий А.Ю.

Санкт-Петербург

2016

1. **Предметная область**

Молекулы органической химии.

1. **Видение виджета**

Мы будем разрабатывать виджет для графического отображения строения молекул органической химии. Виджет будет иметь поле, на котором будут отображаться молекулы и поле с отдельными атомами.

Поле с атомами хранит основные атомы, из которых состоят молекулы органических соединений, рассматриваемых в курсе химии за 10-11 классы. Атомы перетаскиваются из поля с атомами на поле для отображения молекул. Каждый атом имеет свой размер, зависящий от его относительной атомной массы. Таким образом, атом водорода меньше атома углерода. Так же, каждый атом имеет подпись, обозначающую элемент таблицы Менделеева, и цвет. Из каждого атома исходят отрезки, обозначающие его валентность. Соединенные между собой палочки разных атомов означают соединение.

На поле для отображения молекул выстраиваются химические соединения. На этом поле может находится одновременно несколько молекул. Как только в молекуле не осталось атомов, имеющих не закрытые связи, т.е. молекула «собрана», под молекулой пишется ее формула. Если собранная молекула относится к одной из рассматриваемых в курсе химии за 10-11 классы, под молекулой пишется так же ее название.

1. **Цели создания виджета**

Цели создания виджета следующие:

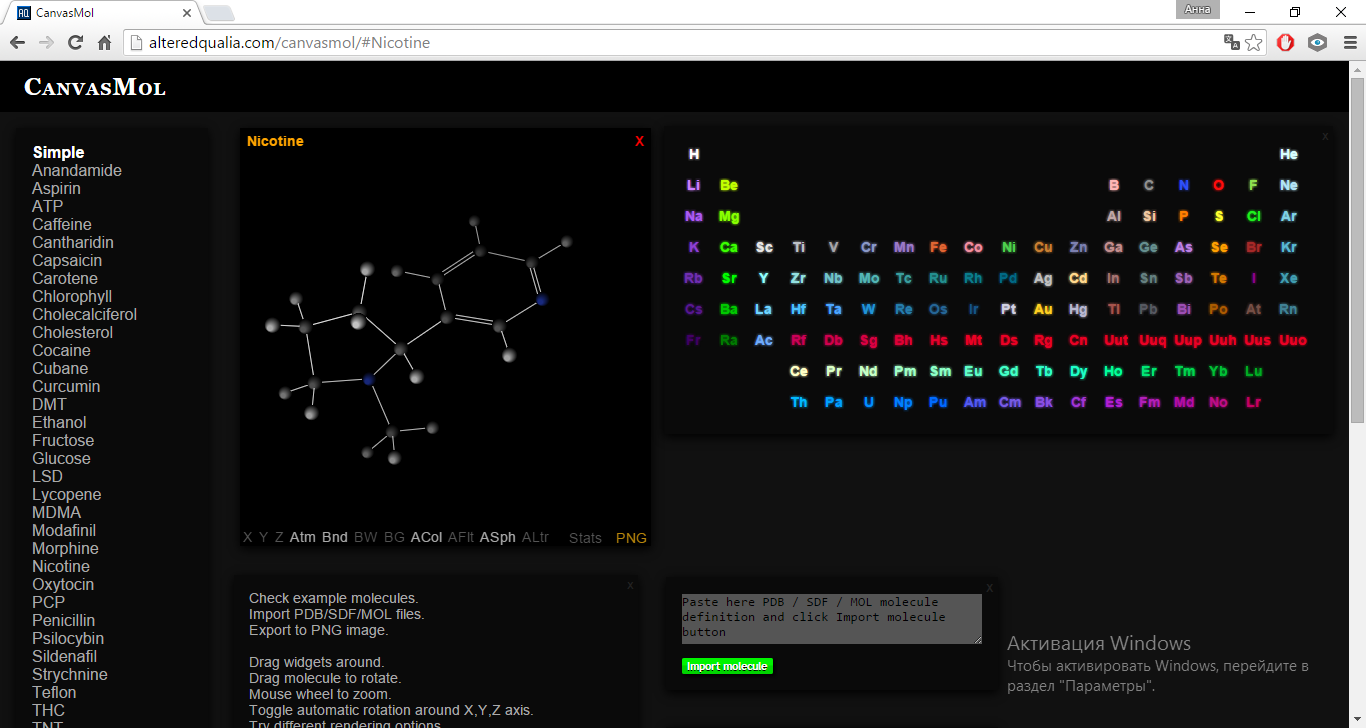
* Создать инструмент, позволяющий наглядно продемонстрировать строение молекул.
* Создать инструмент, позволяющий наглядно демонстрировать отдельные атомы, показывая размер и валентность.
* Выполнить задание по курсу «Проектирование человеко-машинного интерфейса»

1. **Примеры существующих приложений**

**CanvasMol**

<http://alteredqualia.com/canvasmol/>

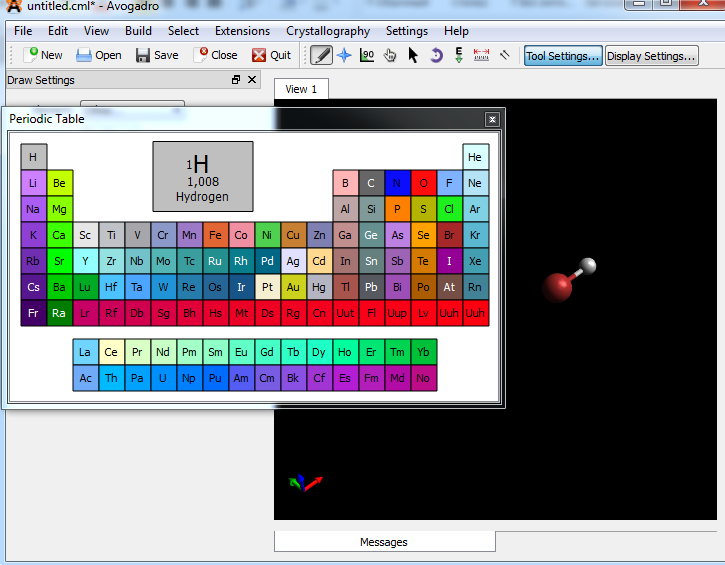
Приложение демонстрирует строение некоторых сложных органических молекул.



Можно рассмотреть молекулы из списка(слева). Так же представлено поле, на котором в отдельном окошке открывается молекула. Каждая молекула имеет настройки – она может вращаться вдоль осей X, Y, Z, могут отображаться или не отображаться молекулы в узлах, соединения между ними. Так же можно настроить чтобы атомы отображались как цветные шарики или как буквы. Молекулы так же можно вращать просто мышкой.  
В отдельном окне открыта таблица Менделеева, на которой отображены цветовые представления атомов, используемые в программе.

**Avogadro**

http://avogadro.cc/wiki/Main\_Page

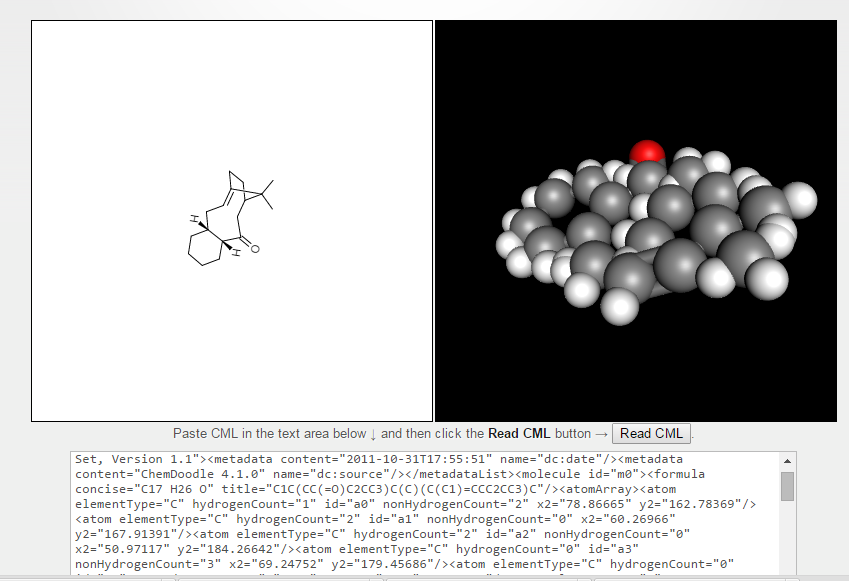


Приложение позволяет выбрать и визуализировать любой элемент из таблицы Менделеева, присутствует область для редактирования. Есть возможность сохранения и загрузки CML-файлов.

**Отрисовка молекулы по CML**

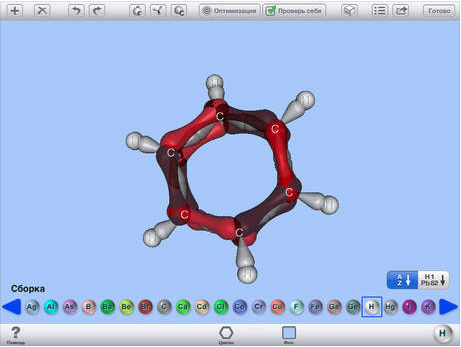
<https://web.chemdoodle.com/demos/chemical-markup-language-cml/>

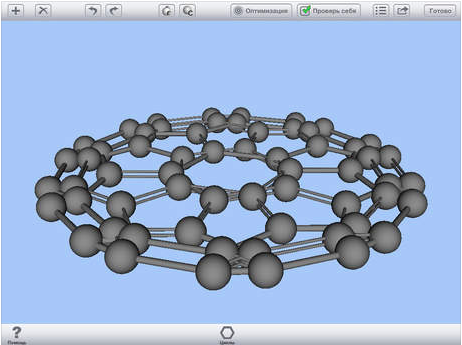
С помощью данного сервиса, можно получить схему молекулы и наглядно увидеть устройство молекулы, передав предварительно строение молекулы в формате CML.

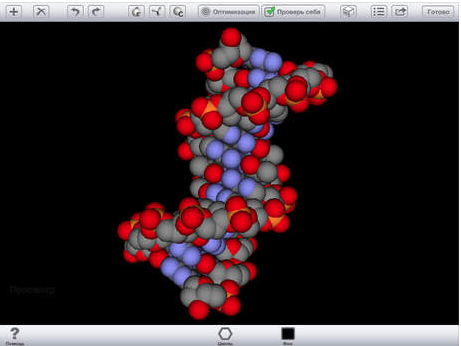


**Приложение для телефонов и планшетов Mols Viewer**

Приложение позволяет конструировать из атомов 3D модели молекул органических и неорганических соединений. Главной особенностью этого редактора является режим «Проверь себя» , в котором можно оценить знания 3D структуры химических соединений.

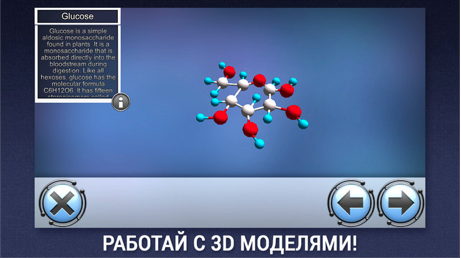


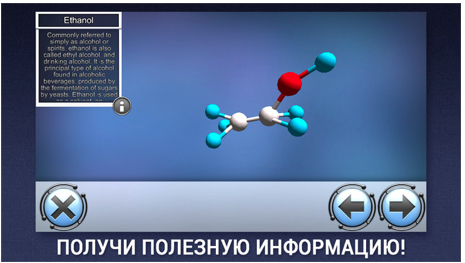


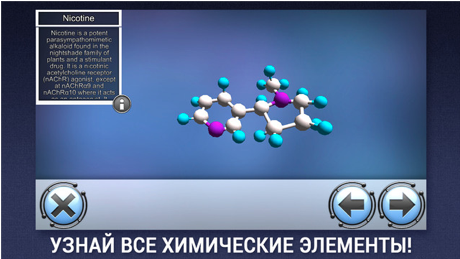


**Приложение для телефонов и планшетов Chemistry Species 3D**

Приложение позволяет создавать различные химические соединения прикладывая элементы друг к другу.







1. **Возможности нашего приложения**

* Выбрать молекулу из списка
* Построить ее схему из заданных элементов
* Просмотреть молекулу в 3D
* Просмотреть выбранные молекулы в 3D без построения схемы

1. **Что мы не будем делать**

* Построение молекулы сразу в 3D
* Состовление пользовательских молекул
* Реакции между атомами не рассматриваются

1. **Язык программирования**

Мы будем использовать для разработки язык программирования Java. Это обусловлено следующими причинами:

* Все участники команды владеют языком Java на необходимом для разработки уровне.
* Язык Java является кроссплатформенным. У нас не возникнет проблем с разработкой под разными операционными системами (Windows 10, Mac OS)
* Язык Java содержит большое количество разнообразных библиотек, что упростит разработку. Можно будет использовать готовые решения, например, контейнеры.
* Возможности языка позволят упростить разработку.

1. **Библиотека для разработки графического интерфейса.**

Для разработки графического интерфейса мы будем использовать платформу JavaFX по следующим причинам:

* JavaFX имеет графический редактор и множество готовых компонент интерфейса (кнопки, панели, разделители и т.д.)
* Форма на JavaFX может быть описана на fxml, что уменьшает количество кода для создания интерфейса.
* Приложение с графическим интерфейсом, сделанным с помощью JavaFX, может быть запущено непосредственно из-под операционной системы, в браузере, на мобильных устройствах.

1. **Интегрированная среда разработки**

Мы будем использовать IDE NetBeans, т.к. считаем ее достаточно удобной и обладающей всеми необходимыми для разработки возможностями.

1. **Средство контроля версий**

При разработке планируется использовать Git — это распределённая [система управления версиями](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%83%D0%BF%D1%80%D0%B0%D0%B2%D0%BB%D0%B5%D0%BD%D0%B8%D1%8F_%D0%B2%D0%B5%D1%80%D1%81%D0%B8%D1%8F%D0%BC%D0%B8), используемая при разработке программ. В нашем случае она удобна, т.к. в IDE NetBeans присутствует плагин для работы с Git, что позволит нашей команде без затруднений коммитить изменения, возвращаться к предыдущей версии и т.д. Еще большой плюс Git в том, что мы можем разрабатывать в параллельных ветвях, не беспокоясь о том, что кто-то напишет код, который может конфликтовать с кодом, который редактируем в текущий момент.

1. **Средство описания строения молекул**

**CML** (Chemical Markup Language) — это язык описания химических соединений, основанный на универсальном языке разметки [XML](https://ru.wikipedia.org/wiki/XML). Использование CML - общепризнанная практика при разработке ПО в сфере химии.

Причины использования CML:

* Большая база знаний: словари, конвертеры, конвенции построения CML.
* Сторонние сервисы и программы, а также компоненты, которые мы можем использовать, работают в большей части только с CML.
* Удобный формат для описания молекулы.

1. **Средство визуализации молекулы**

**Jmol** — программа для просмотра структуры [молекул](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%BB%D0%B5%D0%BA%D1%83%D0%BB%D0%B0) в трёх измерениях.

Jmol используется как для учебных целей, так и при проведении научных исследований в области [молекулярной биологии](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%BB%D0%B5%D0%BA%D1%83%D0%BB%D1%8F%D1%80%D0%BD%D0%B0%D1%8F_%D0%B1%D0%B8%D0%BE%D0%BB%D0%BE%D0%B3%D0%B8%D1%8F), [химии](https://ru.wikipedia.org/wiki/%D0%A5%D0%B8%D0%BC%D0%B8%D1%8F) и [биохимии](https://ru.wikipedia.org/wiki/%D0%91%D0%B8%D0%BE%D1%85%D0%B8%D0%BC%D0%B8%D1%8F). Программа является [свободной](https://ru.wikipedia.org/wiki/%D0%A1%D0%B2%D0%BE%D0%B1%D0%BE%D0%B4%D0%BD%D0%B0%D1%8F_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0) и [открытой](https://ru.wikipedia.org/wiki/%D0%9E%D1%82%D0%BA%D1%80%D1%8B%D1%82%D0%B0%D1%8F_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0). Она написана на языке [Java](https://ru.wikipedia.org/wiki/Java) и потому является [кроссплатформенной](https://ru.wikipedia.org/wiki/%D0%9A%D1%80%D0%BE%D1%81%D1%81%D0%BF%D0%BB%D0%B0%D1%82%D1%84%D0%BE%D1%80%D0%BC%D0%B5%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D0%B5). Существует как отдельная программа, так и средства для интеграции в другое Java-приложение. Чаще всего программа используется как [аплет](https://ru.wikipedia.org/wiki/%D0%90%D0%BF%D0%BB%D0%B5%D1%82), встраиваемый в [веб-страницу](https://ru.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%81%D1%82%D1%80%D0%B0%D0%BD%D0%B8%D1%86%D0%B0). Программа позволяет строить изображения молекул несколькими способами. Jmol поддерживает большое количество форматов файлов, включая:

* [Protein Data Bank](https://ru.wikipedia.org/wiki/Protein_Data_Bank) (pdb)
* [Crystallographic Information File](https://ru.wikipedia.org/wiki/Crystallographic_Information_File) (cif)
* MDL Molfile (mol)
* Chemical Markup Language (CML)
* Chemical File Format (XYZ).

1. **Модель**

В предметной области выделяются простые сущности – атомы и составные сущности – молекулы.

Молекула с точки зрения структуры представляет собой граф. В программе молекула будет храниться как список смежности этого графа. Для этого будут использоваться стандартные Java коллекции TreeMap.

В одном TreeMap будут храниться id атома и соответствующий этому id класс атома. Во втором TreeMap будут храниться id атома и список атомов, с которыми он связан. Таким образом в памяти программы будет храниться информация об атомах и связях между ними.

Каждый атом характеризуется :

* Элементом, который он представляет
* Символом, обозначающий элемент
* Размером атома (относительной атомной массой)
* Количеством связей, которые он может устанавливать
* Коодинатами x,y,z в трехмерном пространстве

Каждая молекула характеризуется:

* Входящими в ее состав атомами
* Связями между этими атомами

Каждая молекула, схема которой используется в программе, описана в отдельном сml файле. Пример:

<molecule>  
 <atomArray>  
 <atom id="a1" elementType="O" x3="-0.230656" y3="2.711862" z3="0.024653"/>  
 <atom id="a2" elementType="H" x3="0.735610" y3="2.673645" z3="-0.031217"/>  
 <atom id="a3" elementType="H" x3="-0.530164" y3="2.191793" z3="-0.735659"/>  
 </atomArray>  
 <bondArray>  
 <bond atomRefs2="a1 a2" order="1"/>  
 <bond atomRefs2="a1 a3" order="1"/>  
 </bondArray>  
</molecule>

Где:

elementType – название атома

x3,y3,z3 – координаты атом в трехмерном пространстве

atomRefs – связи между атомами

order – количество связей(валентность)

x3,y3,z3 – координаты атом в 3-ех мерном пространстве

1. **Вид**

Описанная модель должна отображаться на экране.

Графическое представление каждого атома следующее:

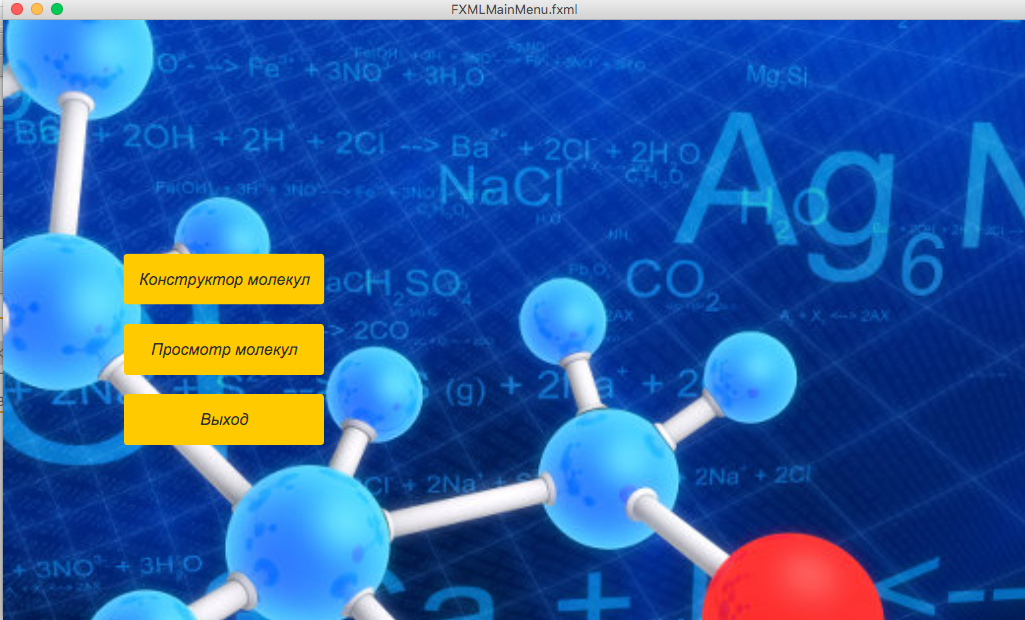
Атом представляется в виде кружка соответствующего размера, в котором написана буква, обозначающая химический элемент. Из кружка выходят линии, обозначающие связи. Между двумя атомами не может быть больше связей, чем минимальное число связей в одном из этих атомов.

Молекула представляется как атомы (кружки), связанные между собой линиями.

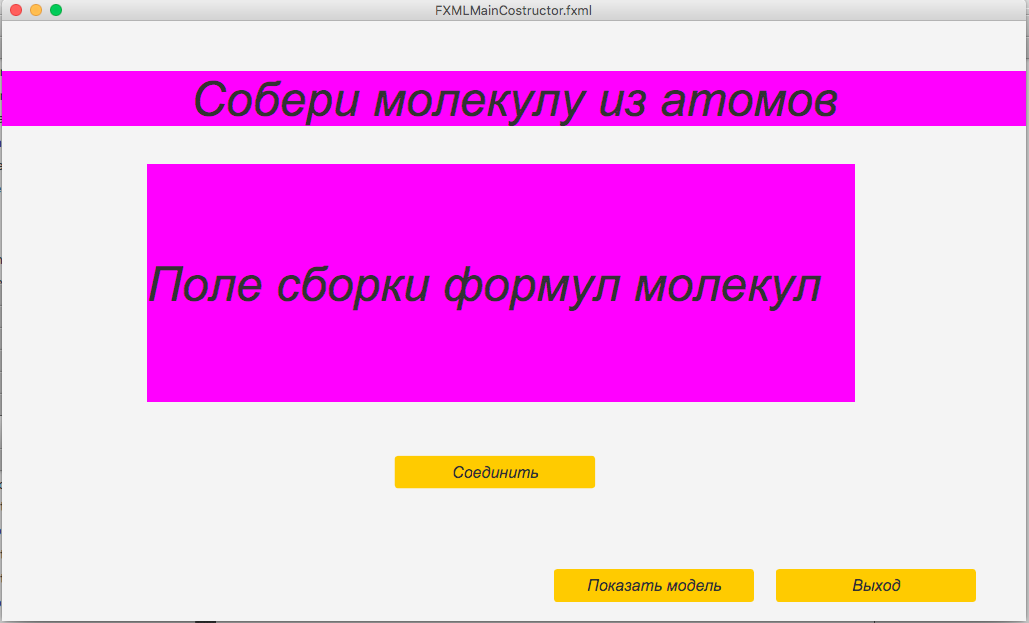
Не связанные между собой атомы рисуются с выходящими из них линиями, не входящими ни в какие другие атомы.

Молекула так же может изображаться трехмерной моделью.

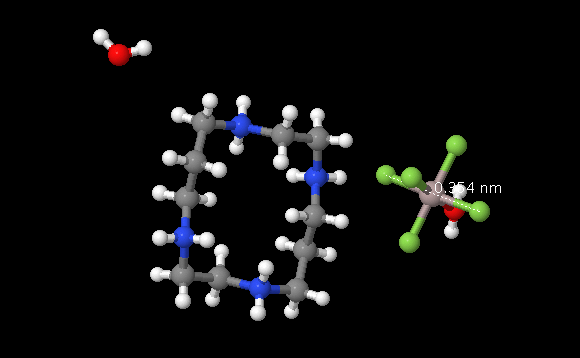
1) Главное меню



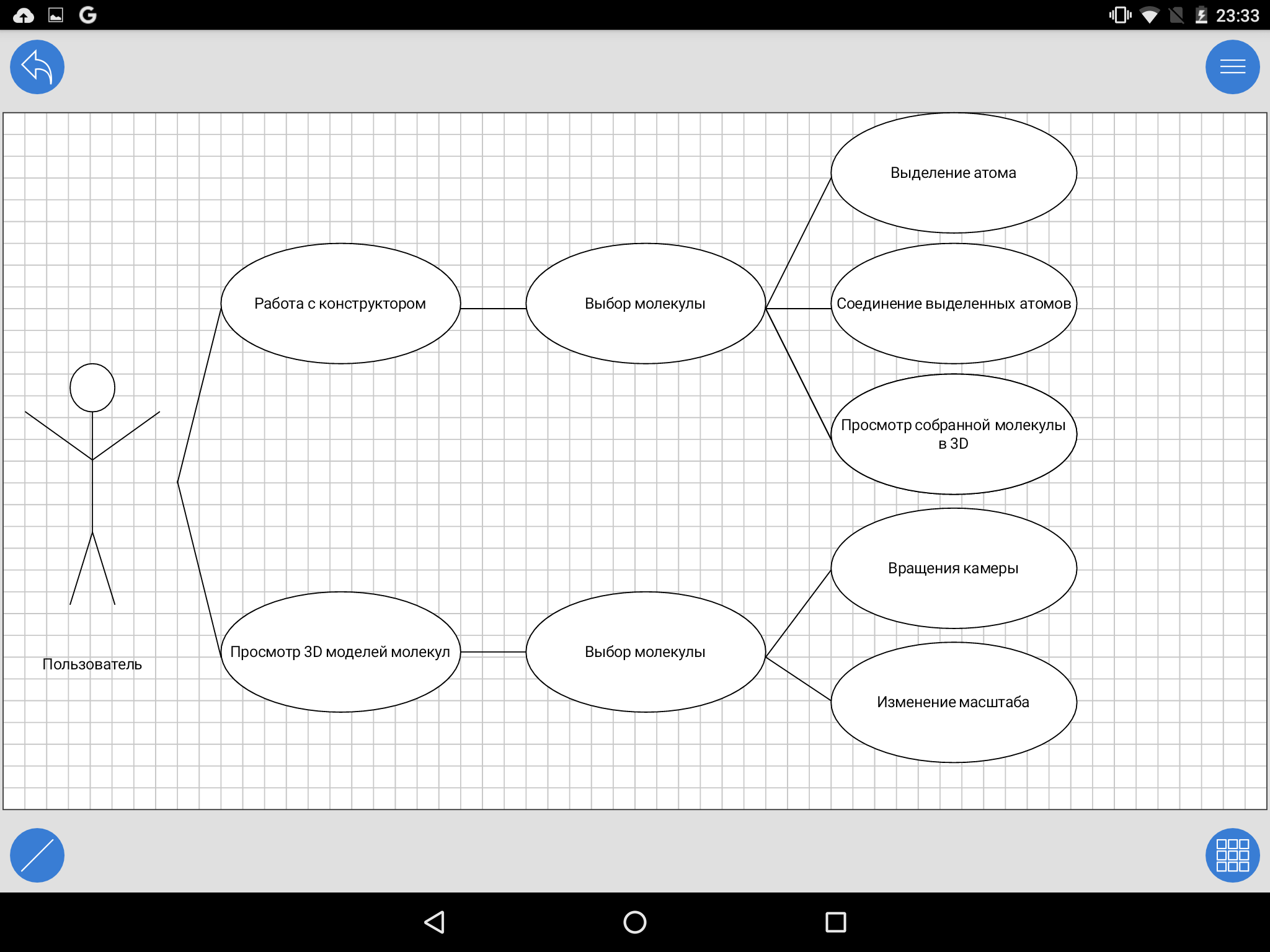
2) Конструктор



3) Просмотр молекул



1. **Use-case диаграмма**



1. **Граф диалога**