Кафедра МО ЭВМ

**Лабораторная работа №2**

По дисциплине

**«Инструментарий разработки»**

**Преподаватель:** Первицкий А.Ю.

**Выполнили:** Диогенова К.А.

Казачкова А.Д.

Сабынин Е.

**Факультет**: КТИ

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. **Выводы**

Все описанные инструменты разработки обладают существенными для нас свойствами. Инструменты выбраны таким образом, чтобы сделать разработку простой и удобной.