Факультет ИУ "Информатика и системы управления"

Кафедра ИУ-3 "Информационные системы и телекоммуникации"

Отчет к лабораторной работе №2

по курсу "**Разработка программного обеспечения**"

направления 2304000062

" **Работа с плагинами и Apache Jena**"

Продолжительность 4 часа.

Выполнил:

студент группы ИУ3-62

Даварашвили И.Б.

Проверил:

Иванов А.М.

# Цель работы

Цель работы — формирование у студентов ряда навыков:

* Использование передовых технологий разработки GUI толстых клиентов
* Использование DSL языков на примере Xtend2
* Основы работы в системе контроля версий Git

# Задание

Создать Eclipse-проект с векторной фигуркой  в соответствии со своим вариантом задания и требованиями, добавить его в репозитарий системы контроля версий.

# Настройка инструментов

Предварительные требования:

* Eclipse с установленным e(fx)clipse с предыдущей лабораторной работы

Последовательность настройки:

1. Создать аккаунт на Github и выслать логин на [amivanoff@gmail.com](mailto:amivanoff@gmail.com) для предоставления прав на запись

2. Настроить подключение Eclipse к Github

* Использование руководство [Настройка Eclipse IDE для доступа к репозитарию GitHub по протоколу HTTP](http://www.agentlab.ru/confluence/pages/viewpage.action?pageId=41058472)
  + Указать другой адрес репозитария, скопировав его со страницы <https://github.com/agentlab/jfxeditor>

3. Установить в Eclipse поддержку языка Xtend2

* См. <https://www.eclipse.org/xtend/download.html>
  + там выбрать "Latest Release"



Исходный код программы:

**package ru.agentlab.jfxed.figures.ass**

**import de.fxdiagram.core.XNode**

**import javafx.scene.Group**

**import javafx.scene.layout.GridPane**

**import javafx.scene.shape.Circle**

**import javafx.scene.shape.Polyline**

**import javafx.scene.text.Text**

**import ru.agentlab.jfxed.IFigure**

**class Assessment extends XNode implements IFigure {**

**var GridPane root**

**val nameTextBox = new Text**

**var String name**

**new(String name) {**

**super(name)**

**this.name = name**

**}**

**//String name = "Assessment";**

**new (){**

**super("Class");**

**name = "Class";**

**root = new GridPane**

**node = root**

**val s = Assessment.getResource("ClassFigure.css").toExternalForm()**

**root.stylesheets += s**

**root.children += new Group => [**

**children += new Polyline => [**

**points += 60d**

**points += 40d**

**points += 40d**

**points += 60d**

**points += 40d**

**points += 120d**

**points += 60d**

**points += 140d**

**points += 200d**

**points += 140d**

**points += 220d**

**points += 120d**

**points += 220d**

**points += 60d**

**points += 200d**

**points += 40d**

**points += 60d**

**points += 40d**

**styleClass += "Prem"**

**]**

**children += new Polyline => [**

**points += 185d**

**points += 80d**

**points += 200d**

**points += 60d**

**]**

**children += new Circle => [**

**centerX = 200**

**centerY = 60**

**radius = 8**

**styleClass += "Krug"**

**]**

**// root.styleClass += "ClassFigure";**

**children += nameTextBox => [**

**text = name**

**layoutX = 75**

**layoutY = 95**

**styleClass += "ClassFigure-Text"**

**]**

**]**

**}**

**override setName(String name) {**

**nameTextBox.text = name**

**}**

**def getName() {**

**nameTextBox.text**

**}**

**override getRoot() {**

**this**

**}**

**}**