

Г2. PSAD, Система трансляции псевдокода в диаграмму деятельности

Золин Иван¹, Кузнецов Даниил¹

¹группа 5030102/00201

1 Назначение и область применения

Назначение

Проект предназначен для визуализации алгоритмов, написанных на псевдокоде, в наглядные диаграммы деятельности.

Область применения

1. Образование и обучение. Система может быть использована в учебных целях для визуализации материалов по алгоритмам и программированию.
2. Программирование и разработка ПО. Разработчики ПО могут применять эту систему для визуализации алгоритмов перед реализацией, что помогает им лучше понять структуру и логику своего кода.

2 Цели и задачи

1. Разработать синтаксис псевдокода.
2. Понять процесс трансляции псевдокода в диаграмму деятельности.
3. Определить основные сущности, которые будут представлены на диаграмме деятельности.
4. Разработать транслятор для преобразования псевдокода в абстрактное синтаксическое дерево (AST).
5. Создать Python файл с классами, описывающими сущности псевдокода.
6. Разработать парсер для преобразования AST в классы Python.
7. Описать figures и bridges для каждой сущности на языке DiaDel.

3 Программа и методика испытаний.

1. Тестирование транслятора псевдокода в AST.
 - **Задача:** Проверить, что транслятор корректно преобразует псевдокод в абстрактное синтаксическое дерево (AST), отражая все сущности диаграммы.
 - **Методика:**
 - (а) Создать набор тестовых алгоритмов на псевдокоде с различными сущностями.

- (b) Запустить транслятор и проверить на каждом примере результаты.
- (c) Убедиться, что недочётов нет. При наличии исправить.

2. Тестирование парсера AST в Python классы.

- **Задача:** Убедиться, что парсер правильно интерпретирует AST и преобразует его в соответствующие классы Python,
- **Методика:**
 - (a) Создать набор тестовых AST, полученных из различных алгоритмов на псевдокоде.
 - (b) Запустить парсер на каждом AST и проверить, что каждый узел соответствует своему Python классу.
 - (c) Убедиться, что недочётов нет. При наличии исправить.

3. Тестирование системы с использованием языка DiaDel

- **Задача:** Убедиться, что система успешно визуализирует псевдокод в диаграмму деятельности, используя языки PSAD и DiaDel.
- **Методика:**
 - (a) Создать набор тестовых алгоритмов на псевдокоде с различными сущностями.
 - (b) Запустить систему и проверить, что каждый псевдокод корректно отображается в диаграмму.
 - (c) Добавить несколько новых сущностей из псевдокода.
 - (d) Протестировать на новом наборе алгоритмов.
 - (e) Убедиться, что недочётов нет. При наличии исправить.

4. Тестирование системы на реальных примерах использования

- **Задача:** Проверить работоспособность системы на реальных практических задачах, которые могут встретиться пользователям.
- **Методика:**
 - (a) Собрать набор реальных алгоритмов на псевдокоде из учебных пособий, книг, онлайн-ресурсов и практических задач.
 - (b) Протестировать систему, удостоверившись, что получаются корректные диаграммы деятельности.

4 Понедельный план-график

01.03:

- Разобрать постановку задачи

- Написать одностраничное описание
- Написать диаграмму использования UML
- Создать метамодель
- Согласовать описание, диаграммы использования, метамодели и планов с Новиковым Ф.А.

15.03:

- Разработать синтаксис псевдокода.
- Определить основные сущности, которые будут представлены на диаграмме деятельности.
- Разработать транслятор для преобразования псевдокода в абстрактное синтаксическое дерево (AST).
- Протестировать полученный транслятор на некоторых примерах.

22.03:

- Создать Python файл с классами, описывающими сущности псевдокода.
- Начать разрабатывать парсер для преобразования AST в классы Python.

29.03:

- Описать figures и bridges для каждой сущности на языке DiaDel.
- Закончить разрабатывать парсер для преобразования AST в классы Python.

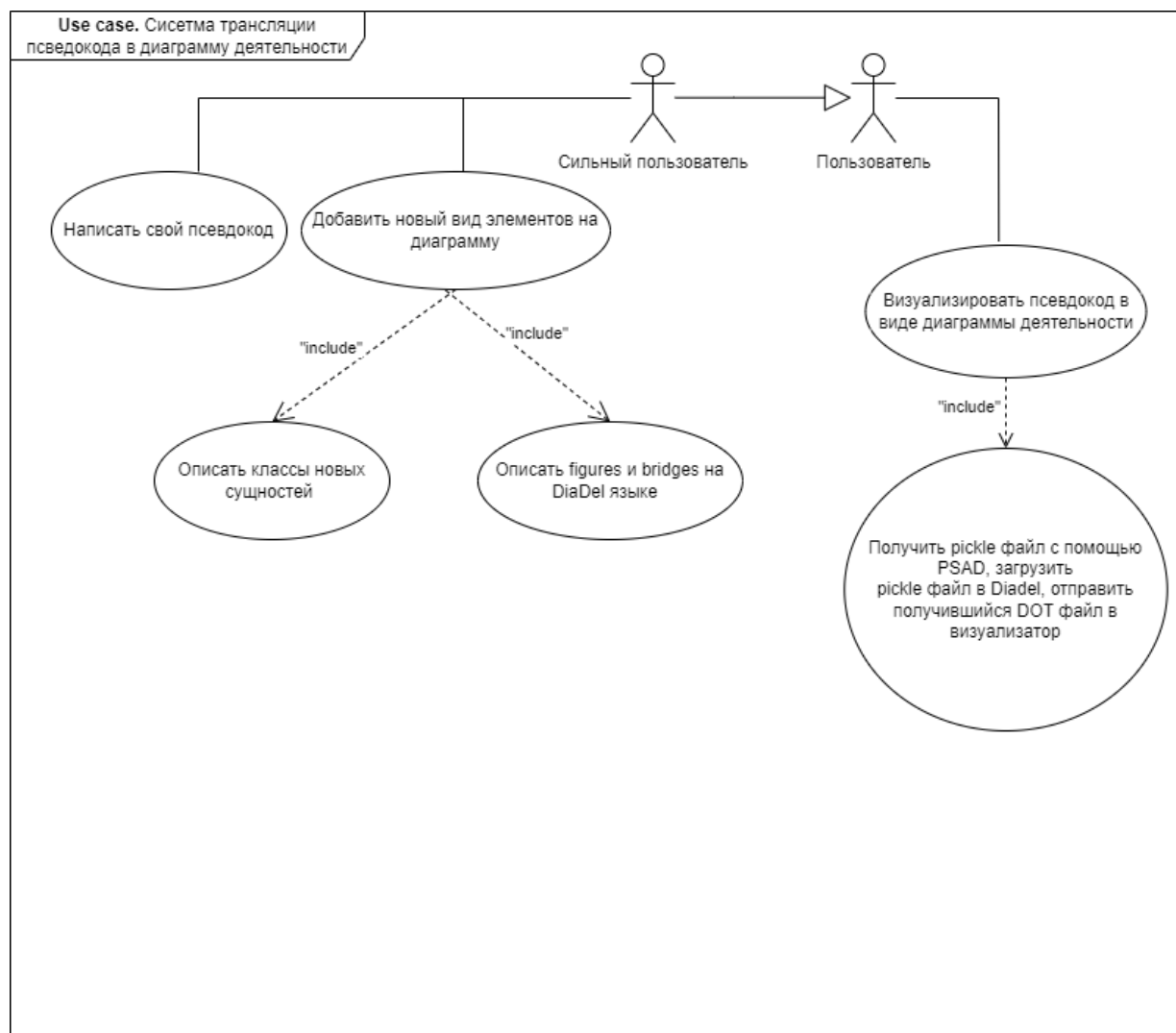
05.04:

- Протестировать парсер.
- Протестировать систему PSAD в целом на некоторых примерах из пособий.

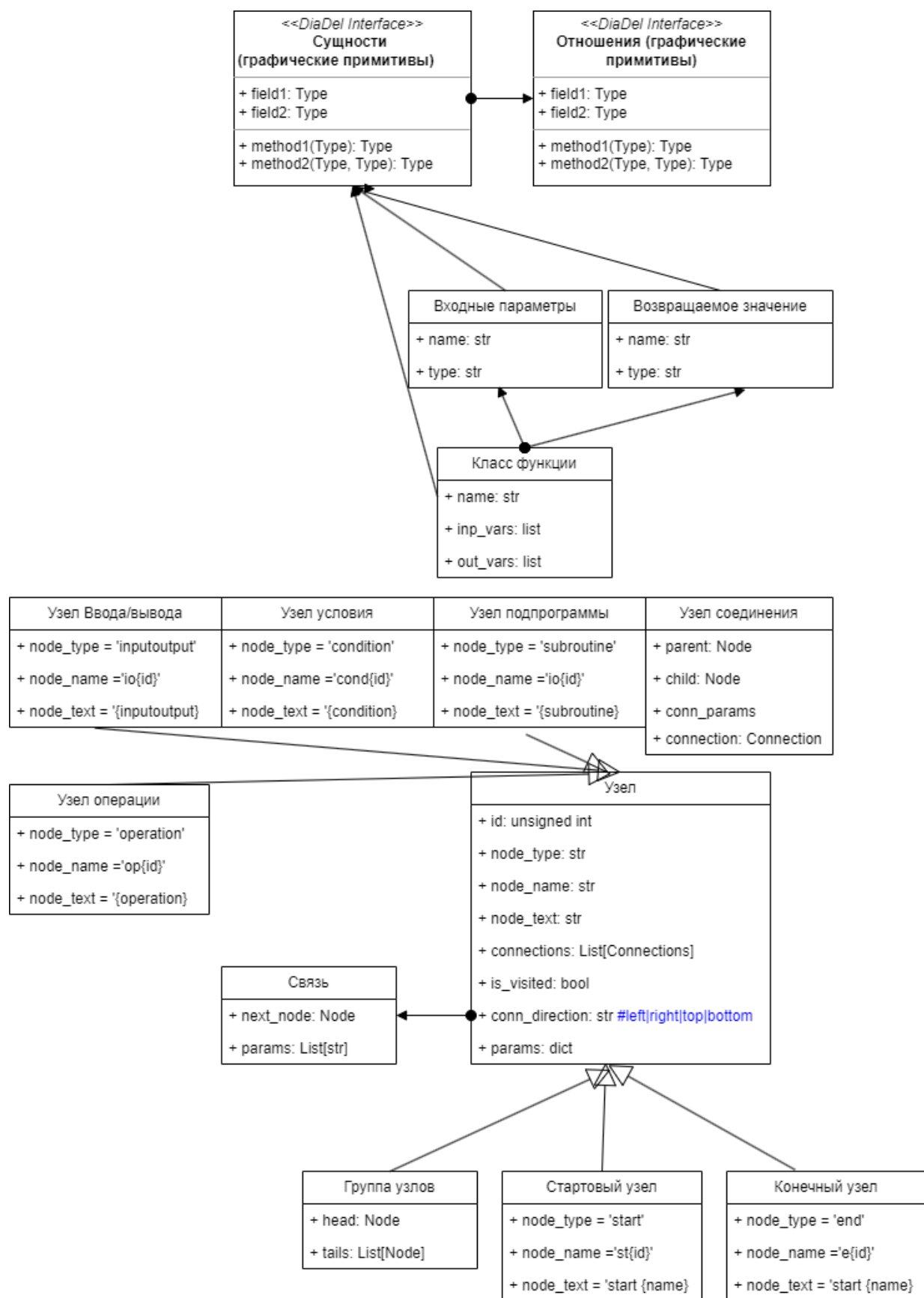
12.04:

- Проверить отрисовку диаграмм с помощью языка DiaDel
- Оформить финальный отчёт
- Защитить проект

5 Диаграмма использования



6 Мета модель



Github репозиторий: <https://github.com/IMZolin/Python-Flowchart-Dsl>