**Глава 3. Разработка программного средства**

**3.1. Концепция проекта**

В качестве языка программирования был выбран язык Java. Для реализации графической части используется Swing - библиотека для создания графического интерфейса. Выбор основывается на том, что на программирование на языке Java не накладывается никаких ограничений связанных с авторскими правами и лицензиями. Плюс технологии Swing в ориентированности на создание пользовательских интерфейсов.

Программное средство разрабатывается для демонстрации работы предложенных методов анализа цветных сетей Петри. Программа должна наглядно демонстрировать возможности применения подобного анализа в моделировании динамических дискретных систем.

Для выполнения поставленной задачи требуется создать программу, которая удовлетворяет следущим требованиям:

1. Позволяет создавать модели динамических дискретных систем.
2. Использует цветные сети Петри для создания моделей.
3. Использует цветные сети Петри для создания моделей.

На программную систему наложены следующие ограничения и предположения:

1. Для храниения данных используется их предствление в текстовом виде.
2. Одновременно в программе допускается работа только с одной моделью.
3. Результаты анализа представляются в виде текстовых файлов.

Сценарии использования (функциональность решения):

1. Пользователь может создать модель используя цветную сеть Петри.
2. Модель можно сохранить в файл и загрузить из файла.
3. Вместе с моделью создается база токенов – перечисление всех токенов в модели и описание их свойств. Базу токенов можно сохранить и загрузить отдельно.
4. При загруженной/созданной модели можно совершить следующие действия: выполнить один шаг сети Петри, или запустить автоматическое выполнение.
5. Выполнение можно остановить и внести изменения в структуру сети.
6. Перед выполнением анализа, пользователь вводит данные о параметрах изменения и параметрах последствий.
7. Анализ запускается и его нельзя остановить до полного завершения.
8. В системе нелья отменять внесенные изменения: удалять созданные элементы сети и описанные параметры анализа.
9. Результаты анализа сохраняются в виде текстовых файлов.

Стратегия архитектурного дизайна решения подразумевает расделение программной системы на компоненты (разделение представлено объединением «пакетов»):

1. “core, fileSavers ...” - отвечает за представление всех элементов цветной сети Петри, сохранение их в файлы
2. “constructor, graphicalElements” – отвечает за графическое представление конструктора для сети Петри
3. “staticNet, liveNet” – отвечает за выполнение сети Петри
4. “dynamic” – отвечает за анализ сети петри, включая формирование параметров анализа

**3.2. Описание функций программы**

Функции программы можно разделить на две группы: функции моделирования и функции проведения анализа. Моделирование позволяет описывать динамических дискретных систем с использованием цветных сетей Петри. Основные функции моделирования:

- Создание токена. При создании токена указывается его название и свойства. Свойство может быть числом или строкой. При задании свойства указывается его тип, и значение. Созданный токен попадает в базу токенов. Далее, для использования этого токена в описании параметров сети или анализа реализована функциональность выбора токена из базы. Если создается токен и его имя уже встречалось ранее – к имени добавляется символ “#”. Если создается токен который уже был в базе (имя и свойства полностью совпадают) – токен в базу не добавляется.

- Создание состояния. Состояние определяется только своим названием.Учет состояний не ведется, проверки на совпадения имен не выполняются. Все имена должны быть уникальны – это ограничение программы.

- Создание перехода. Переход определяется только своим названием. Учет переходов не ведется и проверок на совпадение названий нет. Также как и в случае с состоянием – необходимо чтобы все переходы имели уникальные названия. Параметр перехода – задержка выполнения. Реализована возможность задать переход, который сраборает только один раз – на первом шаге системы. Это позволяет произвести начальное маркирование сети используя только переходы и функции следования.

- Создание функций перехода. Функции предшествования и следования имеют разную релизацию. Создание функции начинается с указания состояния и перехода (порядок означает тип функции). При создании функции указывается следующий параметры: токен (есть возможность взять из базы токенов или создать новый) и количество таких токенов. Для функции предшествования указывается возможность сохранения токена в состоянии.

- Запуск сети. Реализовано два варианта запуска сети – запуск выполнения только следующего шага и запуск выполнения с указанием временной задержки между шагами. Также реализована возможность остановить выполнение сети. Далее, можно опять запускать сеть.

Функции проведения анализа представляют следующие возможности – описать параметры изменений, описать параметры последствий, выбрать сеть для анализа, выбрать папку для сохранения результатов анализа и запустить анализ.

Реализованы следующие параметры изменения:

- параметры изменения для перехода: название перехода, диапазон изменения задержки перехода, параметры временного прекращения работы перехода – диапазон шагов сети.

- параметры изменения для функций перехода: диапазон количества токенов.

- параметры изменения состояний: потеря или появление токена, указание количества таких токенов и интервал времени (в шагах сети) через котороый изменение повторяется.

Реализованы следующие параметры последствий:

- параметры последствий для переходов: название перехода, критическое число его срабатываний и общее число его срабатываний.

- параметры последствий для состояний: указание токена, текущего количества заданных токенов в состоянии, максимальное количество одновременно присутствующих токенов заданного типа в указанном состоянии и критическое число токенов в данном состоянии, присутствующих одновременно.

При запуске анализа указывается количество шагов для запуска сети с примененными параметрами. Статистика анализа сохраняется в текстовых файлах и состоит из описания всех параметров изменений, описания примененных параметров изменений и всех данных о последствиях. Если были достигнуты критические показатели из параметров последствий – они помечаются ключевым словом «!Критично».

Алгоритм применения изменений работает следующим образом: перебираются все сочетания изменений и применяются по очереди.

**3.3. Описание эксприментов**

Для проведения экспериментов, была выбрана модель занятия рабочей станции.

Описание модели:

...

Параметры изменения:

....

Параметры последствий:

....

Результаты анализа:

...