

Содержание

Введение	5
1 Постановка задачи	6
1.1 Общее описание приложения	6
1.2 Описание функций приложения	6
1.3 Прототипа приложения	7
2 Диаграмма вариантов использования	16
3 Структура классов и методов	19
Заключение	24
Список используемой источников	25
ПРИЛОЖЕНИЕ А – Антиплагиат	26

Введение

В ходе данного курсового проекта будет разработан проект модели программного приложения «Конструктор игровых сценариев».

Для решения задачи проектирования программного обеспечения будет использовано UML (унифицированный язык моделирования). Который позволит решить такие кейсы в проектировании как визуализация, структурирование приложения, построить поведенческую модель.

Основным результатом проектирования системы будет визуальная и описательная части проекта, которая позволит на ее основе построить прототип приложения для дальнейшего принятия решений в развитии разрабатываемого проекта

1 Постановка задачи

1.1 Общее описание приложения

Требуется разработать проект программного приложения «Конструктор игровых сценариев». Приложение предназначено для упрощения разработки игровых сценариев и взаимодействия сценаристов и разработчиков игр, где есть сюжет и диалоги. В особенности если это касается ветвления сюжета.

Приложение представляет из себя простое оконное приложение, где можно создавать сценарии при помощи графического взаимодействия путем перетаскивания элементов логики или объектов сценария соединяя их между собой. Узлами в этом представлении будут логические объекты, диалог, фраза, либо иное действие касаемое изменение/перехода переменных (завязаны на логике).

1.2 Описание функций приложения

В приложении должен быть реализован следующий перечень функций необходимых в приложении:

- Создание проекта сценария
- Создание первого акта в сценарии
- Логические блоки для нелинейного ветвления сюжета
- Узел диалога (ведется последовательное построение фраз персонажей)
- Создание логических переменные для ветвления
- Модуль создание персонажей
- Контрольные точки для перехода между актами

1.3 Прототипа приложения

На рисунке 1 изображен макет первоначальной формы. На первой форме есть список уже существующих, либо по нажатию кнопки добавления нового сценария меняется как на рисунке 2, где заводится проект с именем и местом хранения.

Меню

Список последних сценариев

The image shows a wireframe of a menu form. It consists of a large outer rectangle. At the top, there is a horizontal header bar containing the word 'Меню'. Below this header, centered within the main area, is a smaller rectangular box. Inside this box, the text 'Список последних сценариев' is centered.

Рисунок 1 – Прототип формы «Меню»

Создание сценария

Путь к проекту

Назваоние

The image shows a wireframe for a 'Create Scenario' form. It consists of a main container with a title bar at the top labeled 'Создание сценария'. Inside the container, there are two input fields stacked vertically. The first field is labeled 'Путь к проекту' (Path to project) and the second field is labeled 'Назваоние' (Name, with a typo). The rest of the container is empty.

Рисунок 2 – Прототип формы «Меню» при создании сценария

На рисунке 3 отражен макет формы редактирования сценария. Основными его элементами является список актов. Актам задается последовательность перехода в сценарии (сверху в низ), задается путем перетаскивания актов на форме.

Кнопка «Редактирование персонажей» выдет на одноименную форму, где создаются персонажи которые взаимодействуют в сюжетной линии.

«Редактирование переменных» тоже кнопка ведущая на отдельную форму, где создаются переменные.

Меню								
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="text-align: center; padding: 10px;">Акт 1</td></tr> <tr><td style="text-align: center; padding: 10px;">Акт 2</td></tr> <tr><td style="text-align: center; padding: 10px;">Акт 3</td></tr> <tr><td style="text-align: center; padding: 10px;">Акт 4</td></tr> <tr><td style="text-align: center; padding: 10px;">Акт 5</td></tr> </table>	Акт 1	Акт 2	Акт 3	Акт 4	Акт 5	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="text-align: center; padding: 5px;">Редактирование персонажей</td></tr> <tr><td style="text-align: center; padding: 5px;">Редактирование переменных</td></tr> </table>	Редактирование персонажей	Редактирование переменных
Акт 1								
Акт 2								
Акт 3								
Акт 4								
Акт 5								
Редактирование персонажей								
Редактирование переменных								

Рисунок 3 – Прототип формы «Сценарий»

На рисунке 4 отражена форма редактирования акта. Форма представляет из себя полотно, на котором расставляются и связываются в порядке при помощи графических элементов таких как выбор, условие, переходный узел, узел диалога.

Каждая из нод выполняет собственные функции макеты и описание узлов ниже будет далее

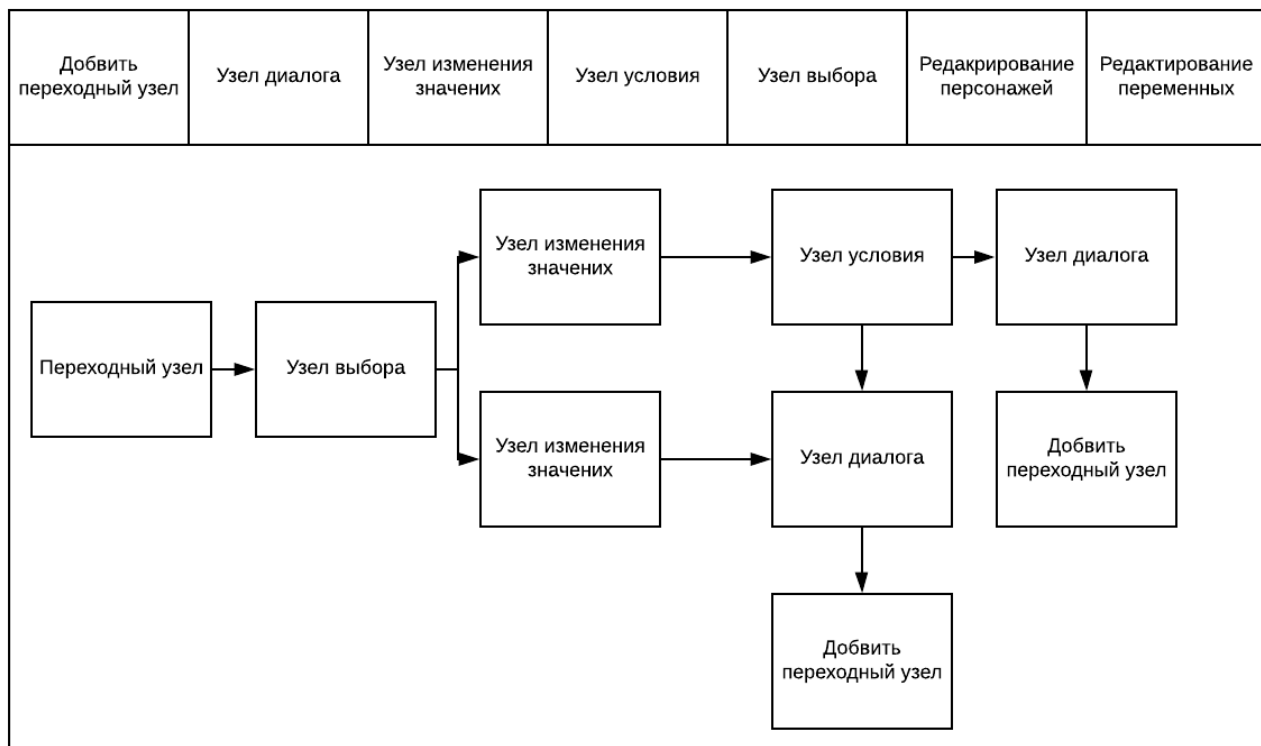


Рисунок 4 – Прототип формы «Акт»

Перходный узел представляет из себя элемент начала в акте. Ставится только в начале акта. Исполняет функцию направления сюжета при переходе из пердыдущих атов сценария. Имеет параметры для прохода по узлу и напращение на следующую ноду в дереве (рисунок 5).

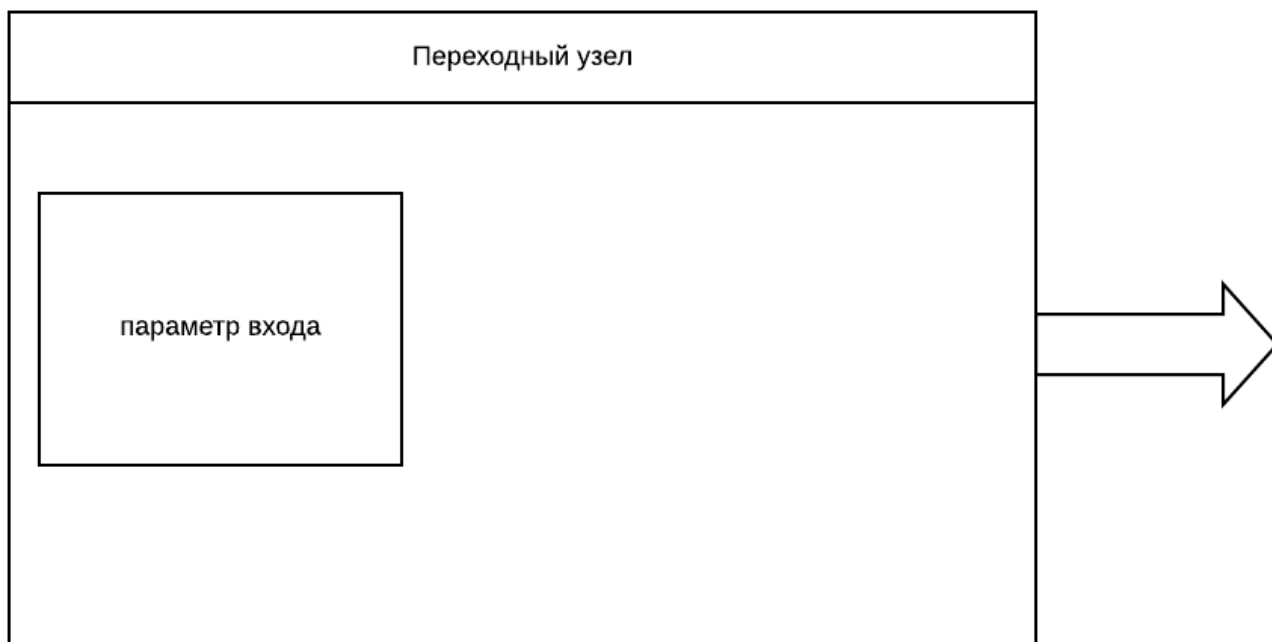


Рисунок 5 – Прототип формы «Переходный узел»

Узел диалога состоит из списка фраз и соответствующие персонажа и эмоции (рисунок 6). Пользователь на данном узле составляет диалог между двумя или группой персонажей. Сами эмоции привязаны к персонажам и выбирается из выпадающего списка.

У ноды диалога нет ветвлений и выбора, является как факт обработки логики и отражает уже выбранный путь в нелнейном сюжете игры, либо повествовании графической новеллы.

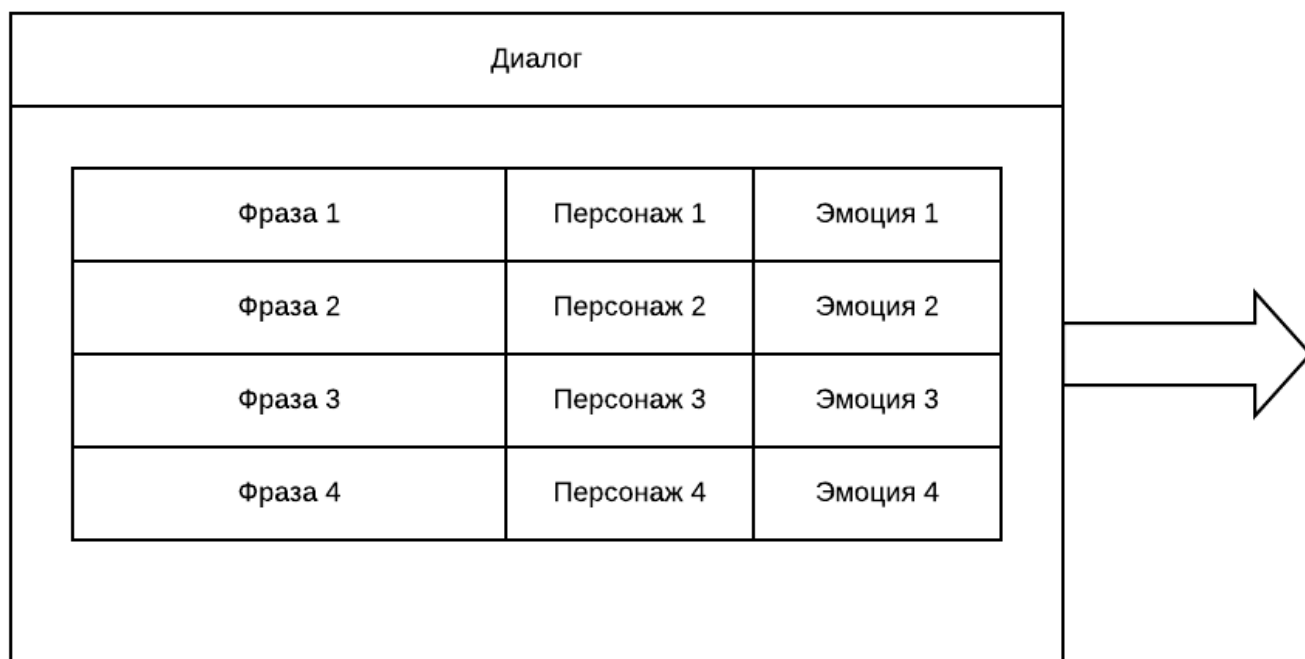


Рисунок 6 – Прототип формы «Диалог»

Узел условие представляет собой графический блок в котором есть список условий которые уходят в связанный маршрут (рисунок 7). Маршрут указывает на следующую ноду, которая может быть как диалогом, так и выбором.

Условия в ноде указывается как обычное сравнение переменных внесенных в соответствующей форме

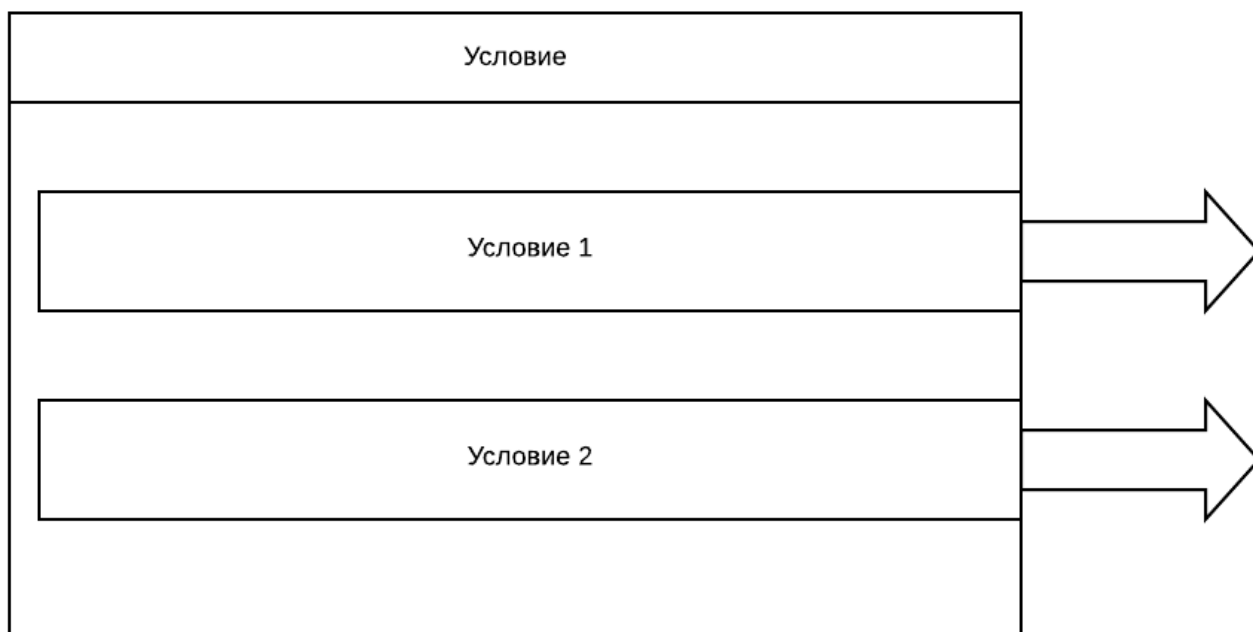


Рисунок 7 – Прототип формы «Условие»

Форма редактора персонажей (рисунок 8). На данной форме заносятся информация о субъектах участвующих в сценарии. Вход на форму есть из сценария и редактора акта.

У персонажей есть основные характеристики для их отличия это в повествования сюжета:

- Имя
- Цвет
- Список эмоций
- Дополнительные параметры

Редактирование персонажей

Имя

Цвет

Эмоции

Дополнительные параметры

Рисунок 8 – Прототип формы «Персонажи»

Форма заведения переменных. Используются в действиях выборах, то что используется в проверке.

Основные параметры для переменной это ее название, тип и начальное значение. 3 вида типов это счетчики, значения, флаги.

Переменные	
Тип переменной	начальное значение

Рисунок 9 – Прототип формы «Переменные»

2 Диаграмма вариантов использования

Конструкция или стандартный элемент языка UML вариант использования применяется для спецификации общих особенностей поведения системы без рассмотрения внутренней структуры этой сущности.

Каждый вариант использования определяет последовательность действий, которые должны быть выполнены проектируемой системой при взаимодействии ее с соответствующим актером.

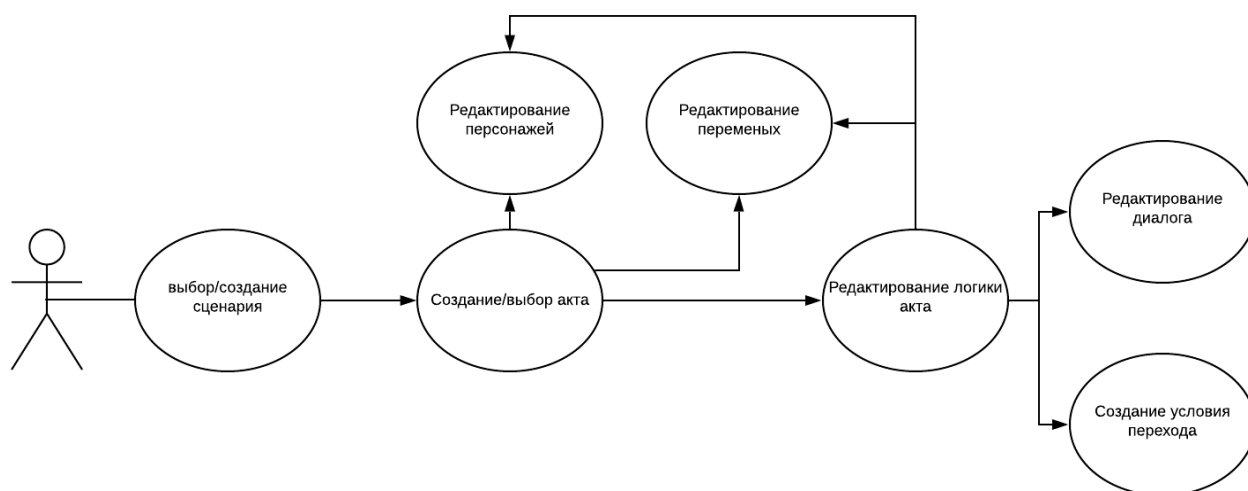


Рисунок 10 - Диаграмма вариантов использования

Поведение и использование форм пользователем практически линейно единственное доступ к редактору персонажей и переменных есть как с формы сценария, так и с редактора акта.

Переходы продемонстрированы на рисунке 11.

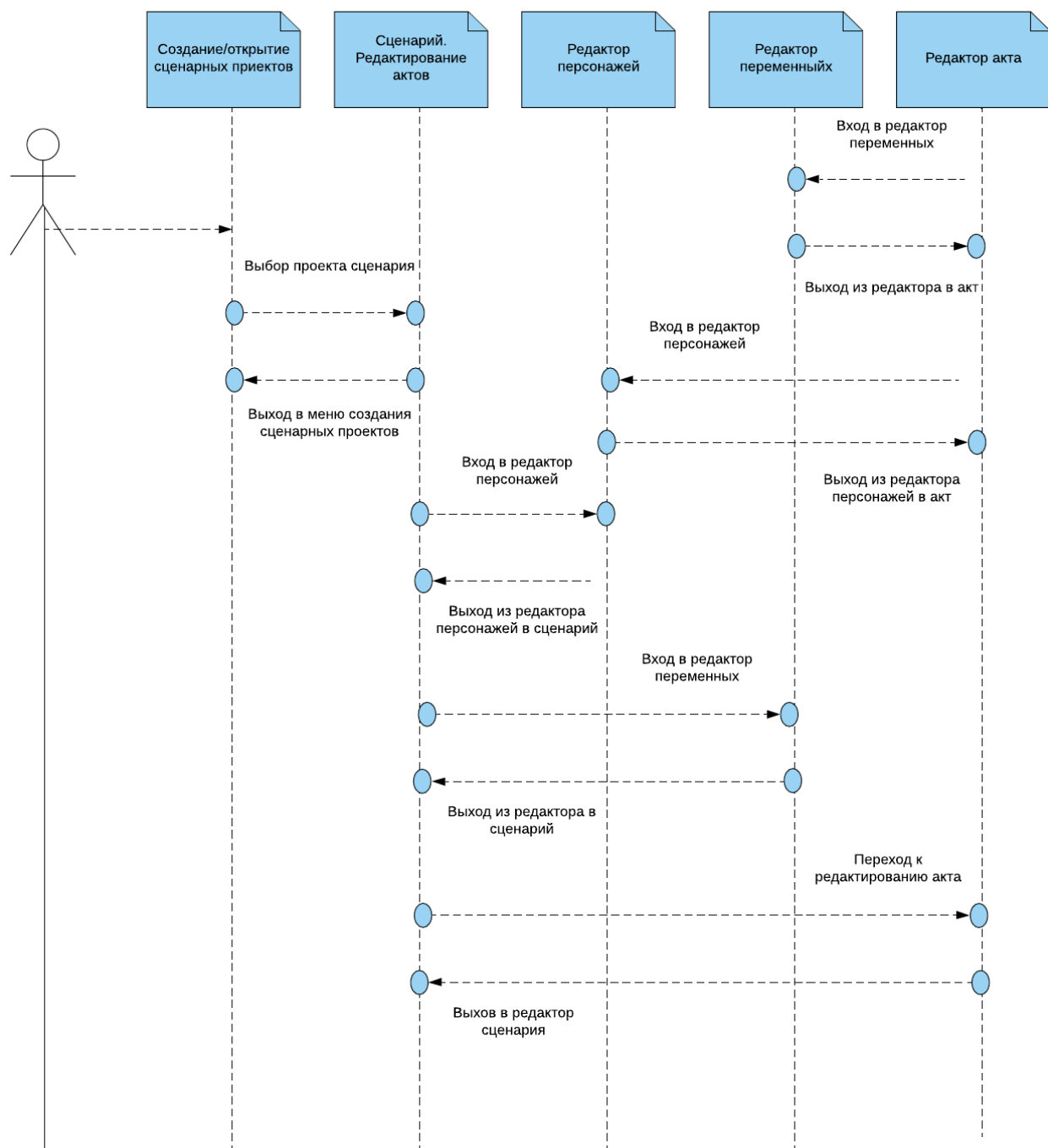


Рисунок 11 – Последовательность действий перехода по основным формам

Большая часть действий пользователя происходит на форме редактора акт. Пользователь выставляет узлы в зависимости от поведения сюжета и условий протекания тех или иных сюжетных поворотов.

Узлы соединяются между собой линиями которая показывает на последовательность действий сценария.

В разных типах узлах есть различное количество входящих выходящих связей. Типы узлов:

- Начало акта
- Условие
- Выбор
- Диалог

3.1 Схема классов

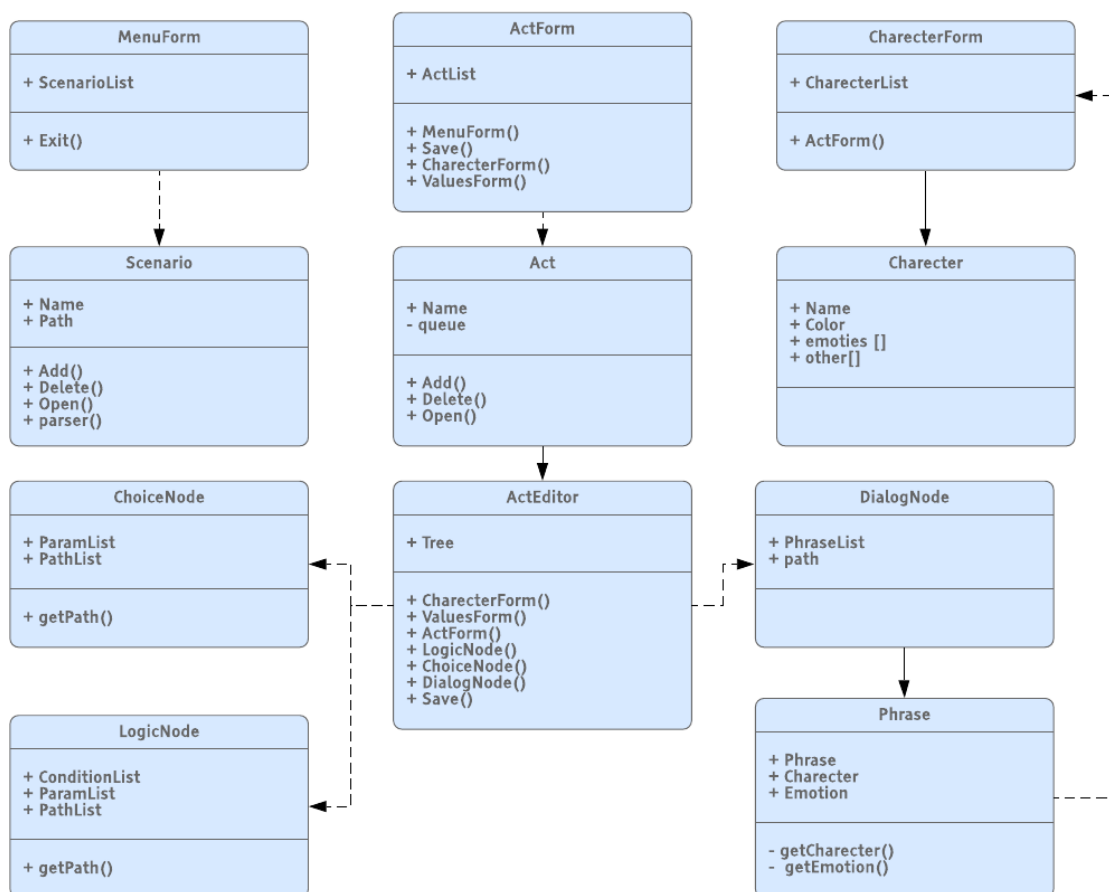


Рисунок 9 – «Структура классов и методов»

3.2 Описание классов методов

MenuForm – Форма меню. Основные действия это создание, удаление, выбор проектов сценария.

ScenarioList – список сценариев.

Exit() – выход из приложения.

Scenario – класс олицетворяющий сущность класса.

Name – название сценария.

Path – путь хранения проекта.

Delete() - удаление объекта.

Open() – открытие сценария в форме сценариев ActForm

parser() – метод сохранения проекта для переноса данных в игровой проет.

ActForm – форма сценария.

ActList – список актов в сценарии.

MenuForm() - переход на форму меню.

Save() - сохранение проекта.

CharecterForm() – переход на форму редактор персонажей.

ValuesForm() – переход на форму редактор перчёных.

Act – Класс описывающий сущность акта.

Name – Название акта.

Queue – номер в очередьт.

Delete() - удаление объекта.

Open() – открытие сценария в форме сценариев ActEditor.

ActEditor

CharecterForm() – переход на форму редактор персонажей.

ValuesForm() – переход на форму редактор перчёных.

ActForm() – переход на форму редактор сценария.

LogicNode() – обработка логического блока.

ChoiceNode() – обработка блока выбора.

DialogNode() – обработка ноды диалога.

Save() – Сохранение акта.

CharecterForm – Форма редактирования персонажей.

CharecterList – список персонажей.

ActForm() – переход на форму сценария.

Charecter – класс описывающий объект персонажа.

Name – имя персонажа.

Color – цвет текста персонажа.

emoties [] – эмоции персонажа.

other[] – дополнительные параметры.

ValueForm – форма редактирования переменных.

ValueList – список значений.

ActForm() – переход на форму сценария.

Value класс описывающий объект переменных.

Name – имя переменной.

Value – первоначальное значение переменной.

Type – тип переменной (флаг, значение, счетчик).

ChoiceNode – Элемент услвия.

ParamList – список параметров.

PathList – соответствующие параметрам лики до следующих нод.

DialogNode – Элемент диалога.

PhraseList – список фраз составляющих диалог.

Phrase – класс описывающий объект фразы.

Phrase – текст, фраза одного из персонажей.

Character – персонаж кому принадлежит фраза.

Emotion – эмоция персонажа в момент фразы.

getCharacter() – подбирает список персонажей.

getEmotion() - подбирает список эмоций соответствующих
выбранному персонажу персонажей.

LogicNode

ConditionList – список условий

PathList – соответствующие условиям логики до следующих нод.

getPath() – метод выбора пути.

3.3 Выбор среды для дальнейшей разработки

В качестве среды для разработки выбран игровой движок Godot Engine. Он подходит не только для создания 2D и 3D игр, но и оконных приложений. У Godot есть очень проработанная система создания элементов управления и написание обработчиков событий.

Основными причинами выбора такой среды разработки в том, что:

Приложение разрабатывается для дальнейшей обработки созданных сценарных проектов в игровых движках.

Удобного создания графических форм.

Из коробки заложена система связывания графических узлов в граф/дерево

Заключение

В ходе выполнения курсового проекта, был разработан проект модели программного приложения «Планировщик бюджета». Была описана общая концепция использования приложения.

Определены основные пользовательские сценарии. Разработана черновая схема работы классов и их взаимодействия, которая будет улучшаться и дополняться в процессе разработки прототипа по данному проекту. Разработанных материалов достаточно для начала реализации полноценного приложения.

Данных материалов достаточно для разработки прототипа приложения и дальнейшего изучения слабых мест в проекте, а также их последующего улучшения.

Список используемой источников

- 1 Арлоу Д., Нейштадт И. UML 2 и Унифицированный процесс. Практический объектно-ориентированный анализ и проектирование. 2-е изд. Пер. с англ. – СПб.: Символ-Плюс, 2007. – 624 с.
- 2 Гагарина Л.Г., Кокорева Е.В., Виснадул Б.Д. Технология разработки программного обеспечения: учебное пособие / Под ред. Л.Г. Гагариной. – М.: ИД «Форум»: Инфра-М, 2013. – 400 с.
- 3 Мартин Фаулер, «UML. Основы. Краткое руководство по стандартному языку объектного моделирования», 2018г - Символ-Плюс, 192 стр.
- 4 Крэг Ларман, «Применение UML и шаблонов проектирования. Третье издание», 2018г - Вильямс, 736 стр.
- 5 Робрет Максимчук, «UML для простых смертных», 2016г - Лори, 400 стр.

ПРИЛОЖЕНИЕ А – Антиплагиат