Пермский филиал федерального государственного автономного образовательного учреждения высшего образования   
«Национальный исследовательский университет   
«Высшая школа экономики»

*Факультет экономики, менеджмента и бизнес-информатики*

Чепоков Елизар Сергеевич

**МАШИНА СОСТОЯНИЙ**

*Отчет по лабораторной работе*

студента образовательной программы «Программная инженерия»

по направлению подготовки *09.03.04 Программная инженерия*

Руководитель:

преподаватель кафедры

информационных технологий

в бизнесе

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

В.П. Куприн

Пермь, 2021 год

**Оглавление**

[**Постановка задачи** 3](#_Toc74508053)

[**Построение Use case диаграммы** 4](#_Toc74508054)

[**Формирование требований** 7](#_Toc74508055)

# **Постановка задачи**

В ходе лабораторной работы необходимо сформулировать требования к проектируемой системе – машине состояний, предназначение которого заключается в отслеживании состояния заданного пользователем цикла и вывода изменений в данном цикле. Кроме того, требуется определить основные варианты использования и ограничения к системе.

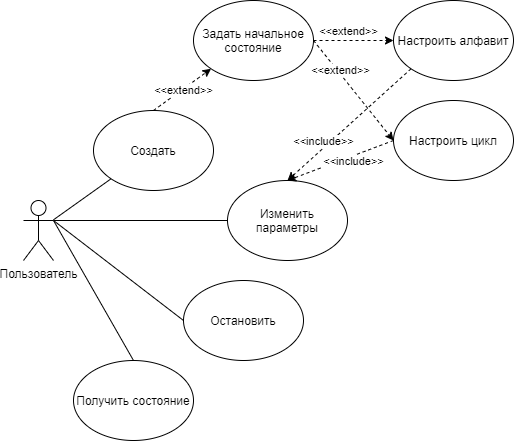
Цель работы: Научиться анализировать поставленную задачу, формулировать требования к проектируемой системе. Определять основные варианты использования и ограничения.

Ожидаемый результат:

1. USE CASE диаграмма;
2. Текстовое описание решаемой задачи;
3. Описание функциональных требований;
4. Описание нефункциональные требований;

# **Построение Use case диаграммы**

Машина состояний (Конечный автомат) устройство, имеющее один вход, один выход и в каждый момент времени находящегося в одном состоянии из множества возможных, также в зависимости от переданных или заранее установленных данных машина состояний вычисляет и изменяет текущее состояние объекта.

Как пример автоматы Мура и автоматы Мили принимают входные значения в виде команд и формируют выходные сигналы, так же к машинам состояний стоит отнести любые триггеры, устройства или программы выполняющие задачи путем цикличного повторения (можно отнести анимации персонажей, устройства по отслеживанию каких-либо данных, например, датчики движения или светофоры и другие программы или устройства)

***Рисунок 1 – Use case диаграмма***

*Таблица 1 – Прецедент «Создать»*

|  |  |  |  |
| --- | --- | --- | --- |
| Идентификатор и название | UC-1. Создать. | | |
| Автор |  | Дата | 02.02.2021 |
| Акторы | Пользователь. | | |
| Описание | Прецедент позволяет пользователю создать новую машину состояний для последующей задачи правил и циклов. | | |
| Триггер | Пользователь создает машину состояний. | | |
| Предварительные условия | PRE-1. Наличие на диске исполняемого файла. | | |
| Выходные условия | POST-1. Машина состояний запущена.  POST-2. Пользователь переходит к задаче правил для машины состояний. | | |
| Основной поток | 1.0 Запуск машины состояний:   1. Выполняется прецедент «Задать начальное состояние» (см 1.0.Е1) | | |
| Альтернативные потоки | - | | |
| Исключения | 1.0.Е1. Файлы программы повреждены.   1. Система выводит сообщение о том, что файлы программы повреждены. 2. Система возвращается к главному экрану. | | |
| Приоритет | Высокий | | |
| Точка расширения | UC-5. Задать начальное состояние. | | |

*Таблица 2 – Прецедент «Изменить параметры»*

|  |  |  |  |
| --- | --- | --- | --- |
| Идентификатор и название | UC-2. Изменить параметры. | | |
| Автор |  | Дата | 02.02.2021 |
| Акторы | Пользователь. | | |
| Описание | Прецедент позволяет изменять правила работы машины состояний. | | |
| Триггер | Пользователь взаимодействует с программой. | | |
| Предварительные условия | PRE-1. Машина состояний запущена. | | |
| Выходные условия | POST-1. Машина состояний работает в заданном цикле. | | |
| Основной поток | 1.0 Изменение параметров:   1. Выполняется прецедент «Настроить алфавит» 2. Выполняется прецедент «Настроить цикл» | | |
| Альтернативные потоки | * 1. Пользователь оставил параметры неизменными:  1. Машина состояний продолжает выполнение работы. | | |
| Исключения | - | | |
| Приоритет | Высокий | | |
| Точка расширения | UC-6. Настроить алфавит.  UC-7. Настроить цикл. | | |

*Таблица 3 – Прецедент «Остановить»*

|  |  |  |  |
| --- | --- | --- | --- |
| Идентификатор и название | UC-3. Остановить. | | |
| Автор |  | Дата | 02.02.2021 |
| Акторы | Пользователь. | | |
| Описание | Прецедент позволяет пользователю завершить или приостановить выполнение машины состояний. | | |
| Триггер | Пользователь завершает или приостанавливает машину состояний. | | |
| Предварительные условия | PRE-1. Машина состояний запущена. | | |
| Выходные условия | POST-1. Программа закрыта. | | |
| Основной поток | 1.0 Завершение работы:   1. Программа завершает работу | | |
| Альтернативные потоки | - | | |
| Исключения | - | | |
| Приоритет | Высокий | | |
| Точка расширения | - | | |

*Таблица 4 – Прецедент «Получить состояние»*

|  |  |  |  |
| --- | --- | --- | --- |
| Идентификатор и название | UC-4. Получить состояние. | | |
| Автор |  | Дата | 02.02.2021 |
| Акторы | Пользователь. | | |
| Описание | Пользователь получает состояние цикла. | | |
| Триггер | - | | |
| Предварительные условия | PRE-1. Машина состояний запущена.  PRE-2. Правила машины состояний заданы. | | |
| Выходные условия | POST-1. Машина состояний выводит состояние. | | |
| Основной поток | 1.0 Вывод состояния:   1. Программа оповещает о состоянии. | | |
| Альтернативные потоки | - | | |
| Исключения | - | | |
| Приоритет | Высокий | | |
| Точка расширения | - | | |

*Таблица 5 – Прецедент «Задать начальное состояние»*

|  |  |  |  |
| --- | --- | --- | --- |
| Идентификатор и название | UC-5. Задать изначальное состояние. | | |
| Автор |  | Дата | 02.02.2021 |
| Акторы | Пользователь. | | |
| Описание | Прецедент позволяет изменить параметры машины состояний. | | |
| Триггер | Пользователь вносит изменения. | | |
| Предварительные условия | PRE-1. Машина состояний запущена. | | |
| Выходные условия | POST-1. Параметры изменены. | | |
| Основной поток | 1.0 Изменение параметров:   1. Выполняется прецедент «Настроить алфавит» 2. Выполняется прецедент «Настроить цикл» | | |
| Альтернативные потоки | - | | |
| Исключения | - | | |
| Приоритет | Высокий | | |
| Точка расширения | UC-6. Настроить алфавит.  UC-7. Настроить цикл. | | |

*Таблица 6 – Прецедент «Настроить алфавит»*

|  |  |  |  |
| --- | --- | --- | --- |
| Идентификатор и название | UC-6. Настроить алфавит. | | |
| Автор |  | Дата | 02.02.2021 |
| Акторы | Пользователь. | | |
| Описание | Прецедент позволяет настроить алфавит работы машины состояний. | | |
| Триггер | Пользователь задает значения. | | |
| Предварительные условия | PRE-1. Машина состояний запущена. | | |
| Выходные условия | POST-1. Алфавит задан. | | |
| Основной поток | 1.0 Настройка алфавита:   1. Пользователь вводит список правил. | | |
| Альтернативные потоки | - | | |
| Исключения | - | | |
| Приоритет | Высокий | | |
| Точка расширения | - | | |

*Таблица 7 – Прецедент «Настроить цикл»*

|  |  |  |  |
| --- | --- | --- | --- |
| Идентификатор и название | UC-7. Настроить цикл. | | |
| Автор |  | Дата | 02.02.2021 |
| Акторы | Пользователь. | | |
| Описание | Прецедент позволяет пользователю задавать или изменять цикл работы. | | |
| Триггер | Выполняется прецедент «Задать начальное значение» или «Изменить параметры» | | |
| Предварительные условия | PRE-1. Машина состояний запущена. | | |
| Выходные условия | POST-1. Новый уикл задан. | | |
| Основной поток | 1.0 Изменение цикла:   1. Пользователь вводит новый цикл. | | |
| Альтернативные потоки | - | | |
| Исключения | - | | |
| Приоритет | Высокий | | |
| Точка расширения | - | | |

# **Формирование требований**

|  |  |
| --- | --- |
| Задача | Описание |
| Создание машины состояний | Создание машины состояний пользователем |
| Изменение параметров | Изменение входных данных пользователем |
| Назначение алфавита | Назначение алфавита для работы машины |
| Назначение цикла | Назначение входного цикла |
| Получение результата | Получение результата пользователем |
| Завершение работы машины состояний | Завершение работы |

Функциональные требования:

* Возможность запустить машину состояний;
* Возможность завершить процесс машины состояний;
* Возможность изменять параметры машины состояний;
* Отображение результата работы программы;
* Возможность изменять цикл;
* Возможность изменять алфавит;

Нефункциональные требования:

* Производительность – система должна быть производительной и не занимать большое количество ресурсов;
* Надежность – машина состояний должна обходить возможные ошибки и выдавать предупреждения;
* Расширяемость – машина состояний должна поддерживать возможность будущих улучшений или пере использования в других системах;
* Гибкость – машина состояний должна иметь возможность изменения параметров под разные задачи;