Министерство образования и науки Российской Федерации

федеральное государственное автономное образовательное учреждение

высшего образования

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Инженерная школа энергетики

Направление 09.04.03 – Прикладная информатика

Отделение электроэнергетики и электротехники

**Проектная документация**

**Отчет по лабораторной работе № 6**

по дисциплине: «Основы объектно-ориентированного программирования»

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Студент гр. | О-5КМ91 |  |  |  | А. А. Петров |
|  | (номер группы) |  | (подпись) |  |  |
|  |  |  | 20.06.2021 |  |  |
|  |  |  | (дата) |  |  |
|  |  |  |  |  |  |
| Преподаватель | |  |  |  | А. А. Калентьев |
| доцент, канд. техн. наук | |  | (подпись) |  |  |
|  |  |  |  |  |  |
|  |  |  | (дата) |  |  |

Томск 2021 г.

**Оглавление**

[Введение 3](#_Toc75017519)

[1 UML диаграмма вариантов использования 4](#_Toc75017520)

[2 UML диаграмма классов 4](#_Toc75017521)

[3 Описание элементов, образующих связь типа «общее-частное» 6](#_Toc75017522)

[Список использованных источников 8](#_Toc75017523)

[Приложение А 9](#_Toc75017524)

Введение

Проектная документация представляет из себя набор документов, которые будут сопровождать любой программный продукт как в процессе разработки, так и на этапе его реализации на месте использования. Написание проектной документации является обязательным критерием разработки и последующей поддержки проекта [1].

Полный объем документов для любого программного продукта позволяет обеспечить информационную поддержку при его изучении, а также защиту интересов его заказчика и разработчика. Описанное обстоятельство достигается путем разработки следующих составляющих проектной документации:

– UML диаграммы вариантов использования;

– UML диаграммы классов;

– описания классов, образующих связь типа «общее – частное» (наследование. реализация);

– технического задания на разработку информационной системы.

1 UML диаграмма вариантов использования

Поскольку пользователи будут взаимодействовать с разработанной информационной системой, то необходимо составить UML диаграмму вариантов ее использования (рис. 1), которая будет показывать. какое поведение можно ожидать пользователю от программы при реализации тех или иных действий [1].

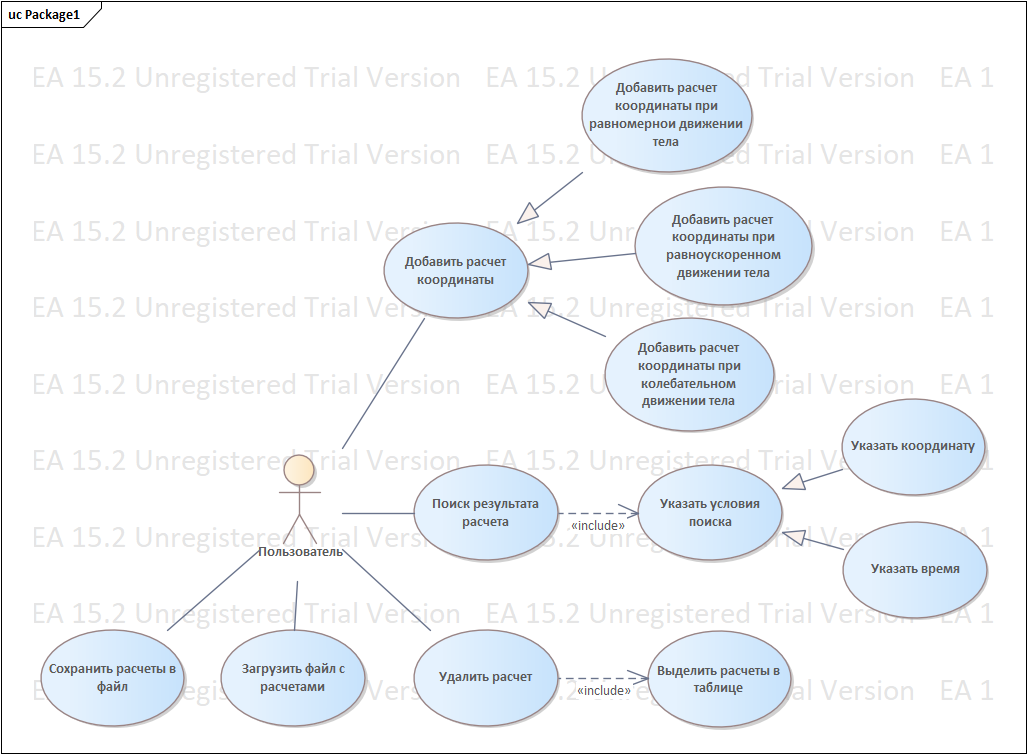


Рисунок 1 – UML диаграмма вариантов использования

2 UML диаграмма классов

UML диаграмма классов, отображающая взаимодействие между классами внутри разработанного программного продукта. представлена на рисунке 2.

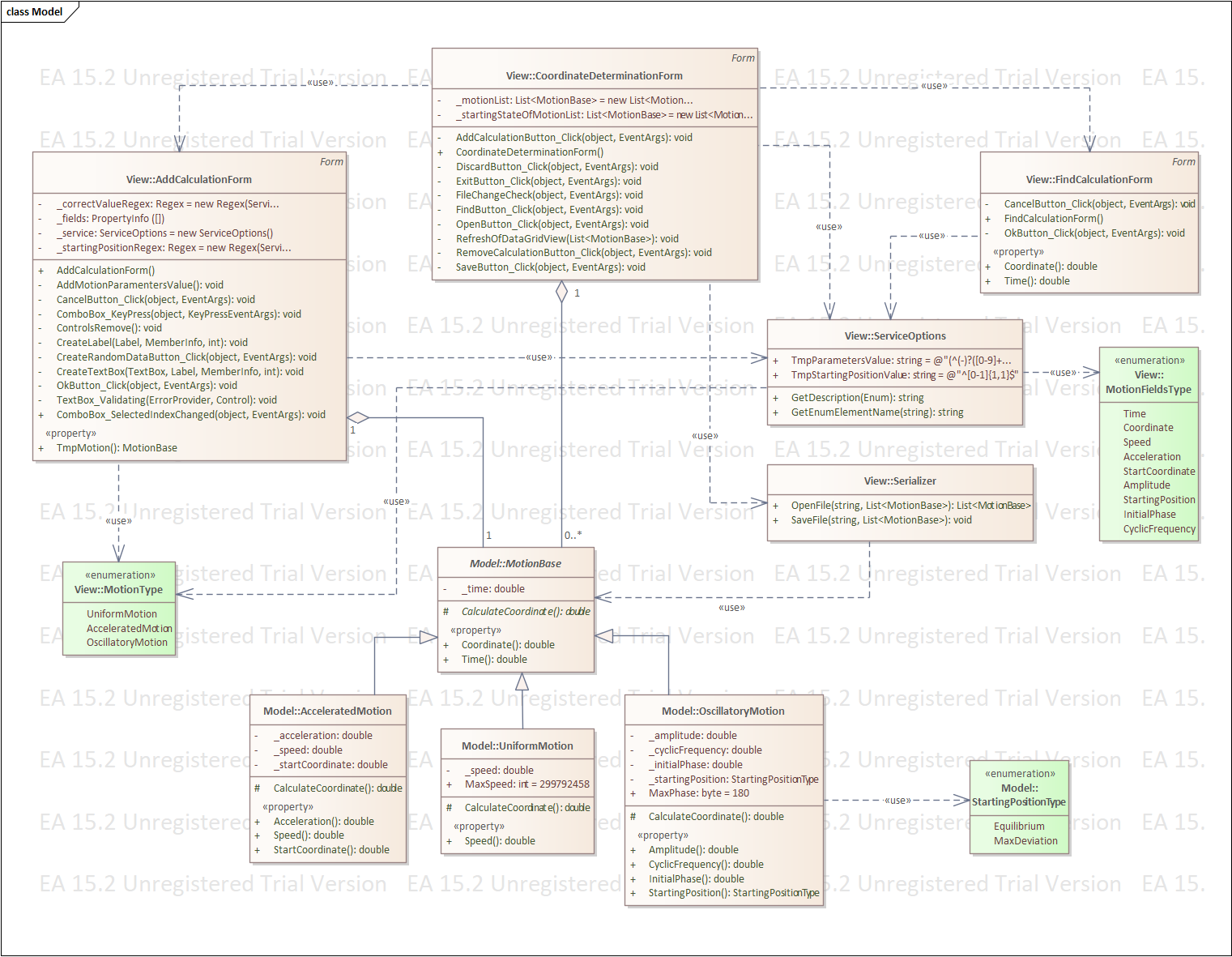


Рисунок 2 – UML диаграмма классов

3 Описание элементов, образующих связь типа «общее-частное»

Связь типа «общее-частное» в разработанном программном продукте образует абстрактный класс MotionBase и такие варианты его реализации, как UniformMotion, AcceleratedMotion и OscillatoryMotion. Описание приведенных классов представлено в таблице 1 – 4.

Таблица 1 – Описание класса MotionBase

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| Описание класса | | |
| Класс *MotionBase* – сущность для описания абстрактного движения физического тела в программе | | |
| Поля | | |
| – \_time | double | Время движения тела |
| Свойства | | |
| + Time | double | Время движения тела |
| + Coordinate | double | Координата тела |
| Методы | | |
| # CalculateCoordinate | double | Расчет координаты тела при его движении. Абстрактный метод переопределяемый в производных классах в зависимости от типа движения. |

Таблица 2 – Описание класса UniformMotion

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| Описание класса | | |
| Класс *UniformMotion* – сущность для описания равномерного движения физического тела в программе | | |
| Поля | | |
| – \_speed | double | Скорость движения тела |
| + MaxSpeed | int | Константа, определяющая максимально возможную скорость движения тела |
| Свойства | | |
| + Speed | double | Скорость движения тела |
| Методы | | |
| # CalculateCoordinate | double | Расчет координаты тела при его равномерном движении. |

Таблица 3 – Описание класса AcceleratedMotion

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| Описание класса | | |
| Класс *AcceleratedMotion* – сущность для описания равноускоренного движения физического тела в программе | | |
| Поля | | |
| – \_acceleration | double | Ускорение тела при его движении |
| – \_speed | double | Скорость движения тела |
| – \_startCoordinate | double | Начальная координате тела |
| Свойства | | |
| + Acceleration | double |  |
| + Speed | double | Скорость движения тела |
| + StartCoordinate | double |  |
| Методы | | |
| # CalculateCoordinate | double | Расчет координаты тела при его равноускоренном движении. |

Таблица 4 – Описание класса OscillatoryMotion

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| Описание класса | | |
| Класс *AcceleratedMotion* – сущность для описания колебательного движения физического тела в программе | | |
| Поля | | |
| – \_amplitude | double | Амплитуда колебаний |
| – \_cyclicFrequency | double | Циклическая частота колебаний |
| – \_initialPhase | double | Начальная фаза колебаний |
| – \_startingPosition | StartingPositionType | Начальное положение тела во время колебаний. Может принимать значения:  – положение равновесия (Equilibrium);  – положение максимального отклонения (MaxDeviation). |
| + MaxPhase | byte | Максимальное значение начальной фазы колебаний |
| Свойства | | |
| + Amplitude | double | Амплитуда колебаний |
| + CyclicFrequency |  | Циклическая частота колебаний |
| + InitialPhase | double | Начальная фаза колебаний |
| + StartingPosition | StartingPositionType | Начальное положение тела во время колебаний. Может принимать значения:  – положение равновесия (Equilibrium);  – положение максимального отклонения (MaxDeviation). |
| Методы | | |
| # CalculateCoordinate | double | Расчет координаты тела при его колебательном движении. |

Список использованных источников

1. Калентьев, А. А. Новые технологии в программировании : учебное пособие / А. А. Калентьев, Д. В. Гарайс, А. Е. Гориянов. – Томск : Эль Контент, 2014. – 176 с. – ISBN 978-5-4332-0185-9.

Приложение А

**Техническое задание на создание автоматизированной системы**

Программа для расчеты координаты тела при различных видах его движения

Разработчик: студент гр. О-5КМ91 ИШЭ ТПУ Петров А.А.

Заказчик: канд. техн. наук, доцент кафедры КСУП ТУСУР Калентьев А. А.

Томск, 2021 г.

**1 Общие сведения**

**1.1 Наименование информационной системы и ее условное обозначение**

Полное наименование: «Программа расчета координаты тела при различных видах его движения».

Условное наименование: Система.

**1.2 Сведенья об организациях заказчика и разработчика**

Заказчик: канд. техн. наук, доцент кафедры КСУП ТУСУР Калентьев А. А.

Разработчик: студент гр. О-5КМ91 ИШЭ ТПУ Петров А.А.

1.3 Плановые сроки начала и окончания работы по созданию системы

Начало разработки Системы – 30.04.2021 г.

Окончание разработки Системы – 12.06.2021 г.

2 Назначение и цели создания системы

**2.1 Назначение системы**

Система предназначена для автоматизации процесса расчета координаты тела при его равномерном, равноускоренном или колебательном движении.

**2.2 Цели создания системы**

Система создается с целью сокращения трудозатрат персонала, занимающегося анализом движения объектов, и снижения ошибок при выполнении большого количества однотипных операций.

3 Характеристика объекта автоматизации

Задача определения координаты движущегося тела является довольно распространенной в обыденной жизни (движение человека, автомобиля, маятника и прочих тел). Для выполнения анализа движения тел порой необходимо выполнение большого количества однотипных расчетов. С целью экономии времени и трудозатрат, а также снижения ошибок при проведении этих расчетов целесообразно автоматизировать процесс их выполнения.

**4 Требования к Системе**

Требования к разрабатываемой системе перечисляются с учетом идентификаторов, представленных в таблице 1.

Таблица 1 – Идентификаторы требований к Системе

|  |  |
| --- | --- |
| Идентификатор | Тип требования |
| А | Требование к архитектуре |
| С | Требование к программной или аппаратной совместимости |
| D | Требование к структуре данных |
| F | Требование к функциональности |
| U | Требование к пользовательскому интерфейсу |

**4.1 Требования к архитектуре**

А01. Система должна быть реализована в виде настольного приложения.

**4.2 Требование к структуре данных**

D01. Исходные данные для работы системы должны содержаться в XML-файле с расширением MTN.

D02. Структура XML-файла с исходными данными должна соответствовать следующей XSD-схеме:

<xsd:schema xmlns:xs= "http://www.w3.org/2001/XMLSchema">

<xsd:element name= "ArrayOfMotionBase">

<xsd:complexType>

<xsd:all>

<xsd:element name="UniformMotion" maxOccurs="Unbounded" minOccurs=0>

<xsd:complexType>

<xsd:sequence>

<xsd:element name="Time" type="xsd:double" >

<xsd:element name="Speed" type="xsd:double">

</xsd:sequence>

</xsd:complexType>

</xsd:element>

<xsd:element name="AcceleratedMotion" maxOccurs="Unbounded" minOccurs=0>

<xsd:complexType>

<xsd:sequence>

<xsd:element name="Time" type="xsd:double" >

<xsd:element name="Speed" type="xsd:double">

<xsd:element name="StartCoordinate" type="xsd:double">

<xsd:element name="Acceleration" type="xsd:double">

</xsd:sequence>

</xsd:complexType>

</xsd:element>

<xsd:element name="OscillatoryMotion" maxOccurs="Unbounded" minOccurs=0>

<xsd:complexType>

<xsd:sequence>

<xsd:element name="Time" type="xsd:double" >

<xsd:element name="Amplitude" type="xsd:double">

<xsd:element name="StartingPosition" type="xsd:StartingPositionType">

<xsd:element name="CyclicFrequency" type="xsd:double">

<xsd:element name="IntialPhase" type="xsd:double">

</xsd:sequence>

</xsd:complexType>

</xsd:element>

</xsd:all>

</xsd:complexType>

</xsd:element>

</xsd:schema>

**4.3 Функциональные требования**

F01. Система должна позволять выполнять расчет координаты тела для следующих типов его движения:

– равномерное;

– равноускоренное;

– колебательное.

F02. Расчет координаты тела при равномерном движении должен выполняться по следующей формуле:

|  |
| --- |
| , |

где – скорость движения тела, м/с; – время движения тела, с.

F03. Расчет координаты тела при равноускоренном движении должен выполняться по следующей формуле:

|  |
| --- |
| , |

где – начальная координата тела, м; – скорость движения тела, м/с; – время движения тела, с; – ускорение тела, м/с2.

F04. Расчет координаты тела при колебательном движении должен выполняться по следующим формулам:

– при начале движения из положения равновесия:

|  |
| --- |
| ; |

– при начале движения из положения максимального отклонения:

;

где– амплитуда колебаний, м; – циклическая частота колебаний, рад/с; – время колебаний, с; – начальная фаза колебаний, рад.

F05. Система должна позволять добавлять параметры и результаты расчетов в общий список расчетов.

F06. Запись расчета координаты движении тела в общем списке расчетов должна иметь следующие параметры:

1. равномерное движение:

– время;

– скорость;

2. равноускоренное движение:

– время;

– скорость;

– начальную координату;

– ускорение;

3. колебательное движение:

– время колебаний;

– амплитуду колебаний;

– начальную фазу колебаний;

– циклическую частоту колебаний.

F07. Система должна позволять удалять параметры и результаты расчетов из общего списка расчетов.

F08. Система должна позволять производить поиск результата расчета координаты движения тела по любому из следующих параметров или их совокупности:

– время;

– координата.

F09. Система должна позволять сохранять подготовленный список расчетов в файл, требования к которому представлены в разделе 4.2.

F10. Система должна позволять открывать подготовленный список расчетов из файла, требования к которому представлены в разделе 4.2.

F11. В системе должна быть реализована обработка значений, вводимых пользователем.

**4.4 Требования к пользовательскому интерфейсу**

U01. Система должна иметь графический интерфейс пользователя, отображающий следующие параметры расчетов:

– тип движения;

– время;

– координату.

U02. Представление расчетов и их параметров должно быть выполнено в табличном виде.

U03. Графический интерфейс должен позволять выбирать расчеты, которые необходимо удалить из общего списка.

U04. Графический интерфейс должен информировать пользователя о возникающих в процессе работы Системы ошибках.

U05. При добавлении параметров нового расчета графический интерфейс должен информировать пользователя о некорректно введенных значениях.

4.5 Требования к программному обеспечению

С01. Система должна работать на компьютере с операционной системой Windows 10 и совместимых с ней.

**4.6 Требования к аппаратному обеспечению**

С02. Процессор компьютера должен иметь тактовую частоту не менее 1 ГГц.

С03. ОЗУ Компьютера должно иметь не менее 2 ГБ памяти.

С04. Компьютер должен иметь не менее 500 МБ свободного места на жестком диске.