МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное

учреждение высшего образования

«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

ИНСТИТУТ ВЫСОКИХ ТЕХНОЛОГИЙ И ПЬЕЗОТЕХНИКИ

Кафедра прикладной информатики и инноватики

КУРСОВАЯ РАБОТА

по дисциплине

«Архитектура информационных систем»

на тему

«Модуль для анализа результатов и показателей циклических видов спорта»

Выполнил

Студент 3 курса 6 группы

Кочетов Даниил Вячеславович

Проверил

Профессор, д.т.н.

Жмайлов Борис Борисович

Ростов-на-Дону

2022

Оглавление

[Техническое задание 3](#_Toc97742671)

[Диаграмма вариантов использования 4](#_Toc97742672)

[Диаграмма классов 7](#_Toc97742673)

[Диаграмма компонент 8](#_Toc97742674)

[Тестирование 9](#_Toc97742675)

# Техническое задание

Наименование проекта: Модуль для анализа результатов и показателей циклических видов спорта

Функции:

1. Определение вероятности победы в гонке, исходя из победы на выбранной дисциплине.
2. Определение вероятности победы в гонке, исходя из любого места в выбранной дисциплине, кроме первого.
3. Нахождение среднего отставания по всем гонкам выбранного участника.
4. Нахождение среднего отставания по всем гонкам в выбранной дисциплине выбранного участника.

# Диаграмма вариантов использования

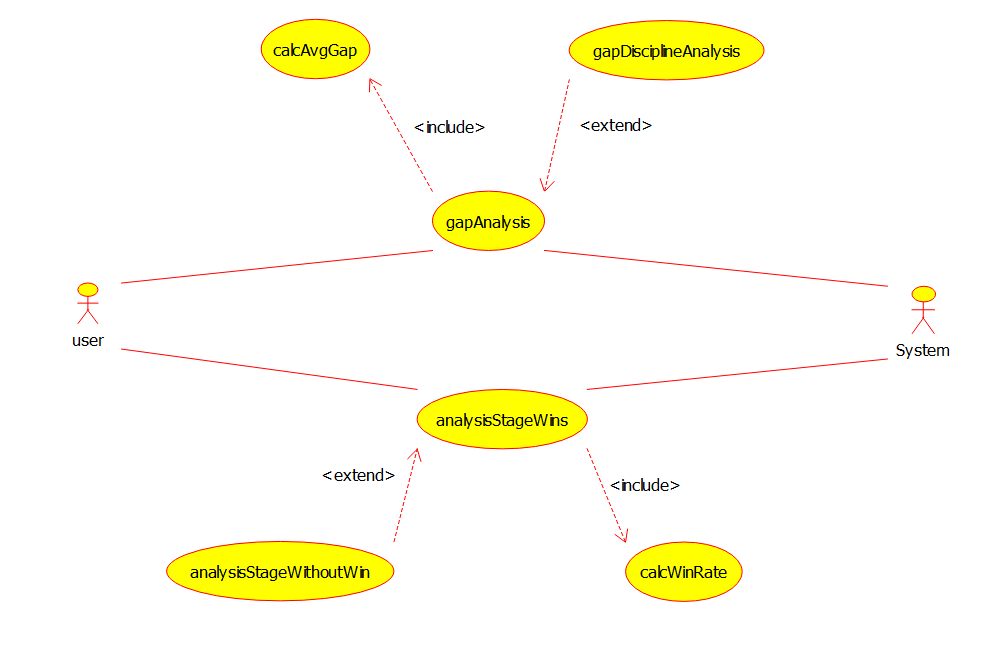


Рисунок 1 Диаграмма вариантов использования

|  |  |
| --- | --- |
| **Вариант использования** | **Детализация** |
| analysisStageWins | Основной метод, позволяющий выбрать вид дисциплины для нахождения процента побед по итогам гонки, при победе на выбранной дисциплине  **Входные данные:**  Переменная типа RaceType – тип гонки.  Переменная типа DisciplineType – тип дисциплины  String переменная, является флагом для использования дополнительного метода  **Выходные данные:**  Список, содержащий посчитанный процент побед, а также процент побед из дополнительного метода (при использовании) |
| analysisStageWithoutWins | Метод, вызывающийся в случае, если третий параметр в методе analysisStageWins эквивалентен “y”, и находящий колличество побед по итогам гонки, при любом месте кроме первого в выбранной дисциплине.  **Входные данные:**  Список результатов по выбранной дисциплине  **Выходные данные:**  Переменная Integer – количество побед. |
| calcWinRate | Обязательный метод, вызывающийся в analysisStageWins, считает процент побед  **Входные данные:**  Две числа Integer, количество побед всего по выбранной гонке (равно количеству гонок) и количество побед по дисциплине при победе во всей гонке соответственно.  **Выходные данные:**  Число Float – процент побед |
| gapAnalysis | Основной метод, позволяющий выбрать участника и посчитать его среднее отставание от лидера по всем его гонкам.  **Входные данные:**  String строка – ID выбранного участника  Переменная типа DisciplineType(необязательный параметр для использования дополнительного метода)  **Выходные данные:**  Список, содержащий посчитанное среднее отставание, а также отставание из дополнительного метода (при использовании). |
| gapDisciplineAnalysis | Дополнительный метод, вызывается при присутствии второго параметра в методе gapAnalysis, и считает среднее отставание участника от лидера по выбранной дисциплине  Входные данные:  Список типа Results, содержащий все результаты участника  Объект типа DisciplineType – выбранная дисциплина  Выходные данные:  Число Float – среднее отставание участника от лидера по выбранной дисциплине. |
| calcAvgGap | Обязательный метод, вызывающийся в gapAnalysis, считает среднее отставание  **Входные данные:**  Две числа Float and Integer, общее отставание от лидера и количество гонок.  **Выходные данные:**  Число Float – среднее отставание |

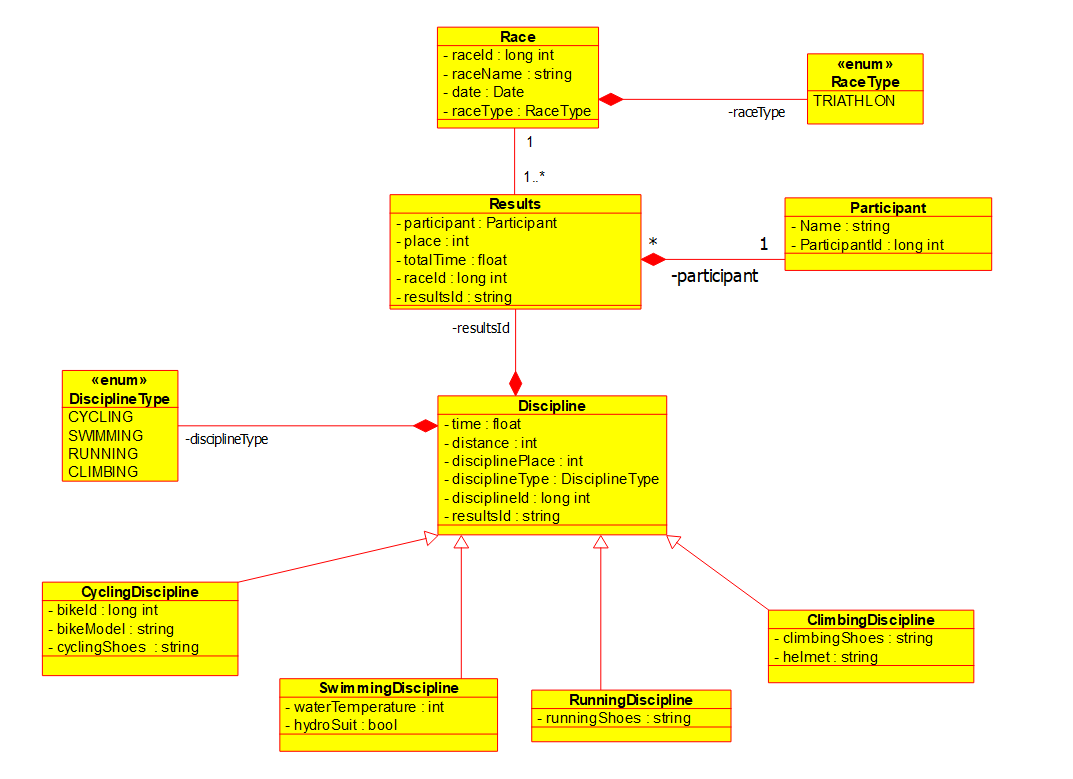
**Диаграмма классов**

Рисунок 2 диаграмма классов

# Диаграмма компонент

Рисунок 3 диаграмма компонент

# Тестирование

|  |  |  |  |
| --- | --- | --- | --- |
| **Тестовый метод** | **Тип сценария** | **Тестируемый метод** | **Что контролируется?** |
| analysisStageWinPositive | Позитивный | Data.provider.analysisStageWin (RaceType raceType, DisciplineType disciplineType, String analysisWithoutWin) | Соответствие результата с заранее заготовленными данными  assertEquals(expected, resultList) |
| analysisStageWinNegative | Негативный | Data.provider.analysisStageWin (RaceType raceType, DisciplineType disciplineType, String analysisWithoutWin) | Существование результата (указание несуществующей дисциплины)  assertFalse(result.isPresent) |
| gapAnalysisPositive | Позитивный | Data.provider.gapAnalysis(String participantId, DisciplineType disciplineType) | Соответствие результата с заранее заготовленными данными  assertEquals(expected, resultList) |
| gapAnalysisNegative | Негативный | Data.provider.gapAnalysis(String participantId, DisciplineType disciplineType) | Существование результата (указание несуществующего id участника)  assertFalse(result.isPresent) |