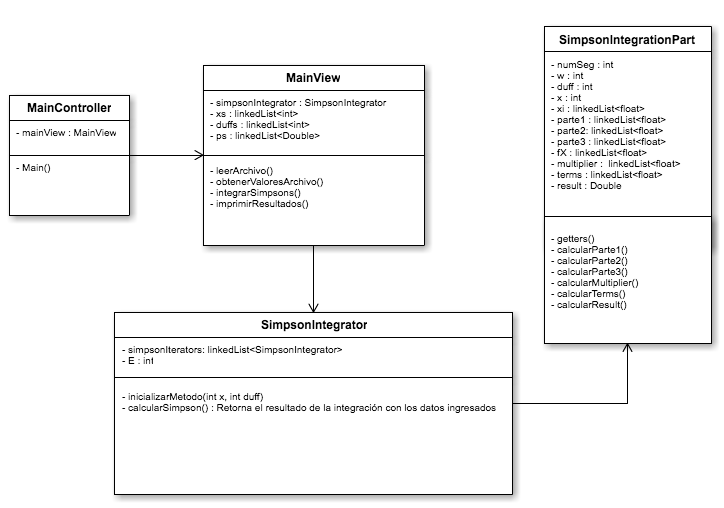
**Metaphor/Architecture Specification Template**

|  |  |  |  |
| --- | --- | --- | --- |
| Student | Ian Sarasty Medina | Date | 01/04/17 |
| Program | Program 5 | Program # | 5 |
| Instructor |  | Language | Java |

|  |  |
| --- | --- |
| **Design** | Functional Specification.rtf |
| **References** | Operational Scenarios.rtf |
|  |  |
|  |  |
|  |  |

**Graphical representation of the metaphor**

****

**Textual representation of metaphor**

The MVC pattern is used in this design. There are four Classes: one MainController that read the input data from the user, MainView that manage the integration operation, a Class that calculate all the Integration and a Class that contain all the data for one iteration for the integration.

The only parameter that the program waits is the file with the data needed for make the integration.

Metaphor/Architecture Specification Template Instructions

|  |  |
| --- | --- |
| Purpose | * To contain the metaphor for a program, component, or system * To enable precise, rapid and complete design understanding * To facilitate thorough design and implementation reviews and inspections |
| General | * Use this template to document the program’s high-level metaphor. * The metaphor could be based in common programming patterns as MVC, or architectural styles as tree layer design, client-server, or inversion of control frameworks * After implementation and testing, update the template to reflect the actual implemented product. * Use plain language and avoid using programming instructions wherever practical. |
| Header | * Enter your name and the date. * Enter the program name and number. * Enter the instructor’s name and the programming language you are using. |
| Design References | List the references used to produce the program’s design.   * the Operational, Functional, and State templates * the program’s requirements * any other pertinent source |
| Graphical representation of the metaphor/Architecture | * Create a graphical representation of the main program parts and its interactions * Use clear names for each part * Use edges with arrows to show interactions * Use descriptive names for the interactions |
| Textual representation of metaphor | * Use text to describe the main idea and metaphor used in your design * Describe the graphical representation using common language |