Software Documentation

Generated by Doxygen 1.9.4

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

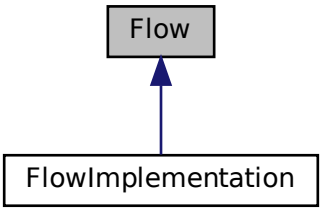
Chapter 4

Class Documentation

4.1 Flow Class Reference

`#include <Flow.h>`

Inheritance diagram for Flow:



Public Member Functions

- virtual ~Flow ()=default
- virtual std::string getName () const =0
- virtual void setName (std::string)=0
- virtual double getValue () const =0
- virtual void setValue (double)=0
- virtual void expression ()=0
- virtual System * getSystemBegin () const =0
- virtual void setSystemBegin (System *)=0
- virtual System * getSystemEnd () const =0
- virtual void setSystemEnd (System *)=0

4.1.1 Constructor & Destructor Documentation

4.1.1.1 ~Flow()

```
virtual Flow::~Flow ( )  [virtual], [default]
```

4.1.2 Member Function Documentation

4.1.2.1 expression()

```
virtual void Flow::expression ( )  [pure virtual]
```

Implemented in FlowImplementation.

4.1.2.2 getName()

```
virtual std::string Flow::getName ( ) const  [pure virtual]
```

Implemented in FlowImplementation.

4.1.2.3 getSystemBegin()

```
virtual System * Flow::getSystemBegin ( ) const  [pure virtual]
```

Implemented in FlowImplementation.

4.1.2.4 getSystemEnd()

```
virtual System * Flow::getSystemEnd ( ) const  [pure virtual]
```

Implemented in FlowImplementation.

4.1.2.5 getValue()

```
virtual double Flow::getValue ( ) const  [pure virtual]
```

Implemented in FlowImplementation.

4.1.2.6 setName()

```
virtual void Flow::setName (
            std::string  )  [pure virtual]
```

Implemented in FlowImplementation.

4.1.2.7 setSystemBegin()

```
virtual void Flow::setSystemBegin (
            System *  )  [pure virtual]
```

Implemented in FlowImplementation.

4.1.2.8 setSystemEnd()

```
virtual void Flow::setSystemEnd (
            System *  )  [pure virtual]
```

Implemented in FlowImplementation.

4.1.2.9 setValue()

```
virtual void Flow::setValue (
            double  )  [pure virtual]
```

Implemented in FlowImplementation.

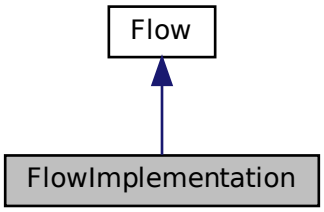
The documentation for this class was generated from the following file:

- src/lib/Flow.h

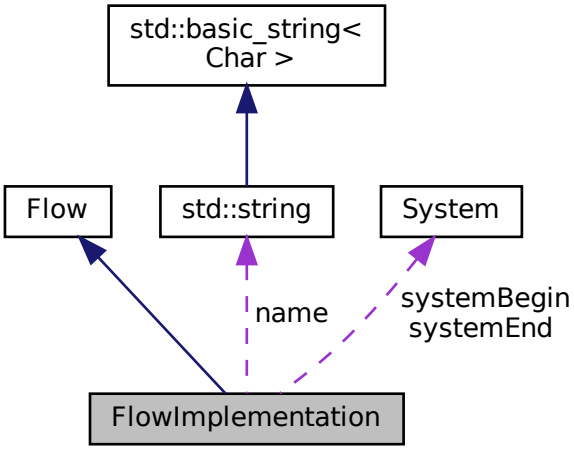
4.2 FlowImplementation Class Reference

`#include <FlowImplementation.h>`

Inheritance diagram for FlowImplementation:



Collaboration diagram for FlowImplementation:



Public Member Functions

- ~FlowImplementation () override
- FlowImplementation (std::string name, double value, System *systemBegin, System *systemEnd)
- FlowImplementation & operator= (const Flow &flow)
- void expression () override=0
- std::string getName () const override
- void setName (std::string n) override
- double getValue () const override

- void setValue (double v) override
- System * getSystemBegin () const override
- void setSystemBegin (System *system) override
- System * getSystemEnd () const override
- void setSystemEnd (System *system) override

Protected Attributes

- std::string name
- double value
- System * systemBegin
- System * systemEnd

4.2.1 Detailed Description

Flow that converges energy from a model to another

4.2.2 Constructor & Destructor Documentation

4.2.2.1 ~FlowImplementation()

```
FlowImplementation::~FlowImplementation ( )  [override], [default]
```

Default destructor

4.2.2.2 FlowImplementation()

```
FlowImplementation::FlowImplementation (
            std::string name,
            double value,
            System * systemBegin,
            System * systemEnd )
```

Default constructor

Parameters

name	Inital flow name
value	Inital flow value
systemBegin	Inital system where the flow comes from
systemEnd	Inital system where the flow goes to

Returns

Flow with initial name, value, systemBegin and systemEnd

4.2.3 Member Function Documentation

4.2.3.1 expression()

```
void FlowImplementation::expression ( )  [override], [pure virtual]
```

Sets the expression of the flow

Implements Flow.

4.2.3.2 getName()

```
std::string FlowImplementation::getName ( ) const  [override], [virtual]
```

Get system name

Implements Flow.

4.2.3.3 getSystemBegin()

```
System * FlowImplementation::getSystemBegin ( ) const  [override], [virtual]
```

Get systemBegin

Implements Flow.

4.2.3.4 getSystemEnd()

```
System * FlowImplementation::getSystemEnd ( ) const  [override], [virtual]
```

Get systemEnd

Implements Flow.

4.2.3.5 getValue()

```
double FlowImplementation::getValue ( ) const  [override], [virtual]
```

Get system value

Implements Flow.

4.2.3.6 operator=()

```
FlowImplementation & FlowImplementation::operator= (
            const Flow & flow )
```

Copy Assignment Operator

Parameters

flow	Flow to copy from

Returns

Copied flow

4.2.3.7 setName()

```
void FlowImplementation::setName (
            std::string n )  [override], [virtual]
```

Set system name

Parameters

n	Name for the flow

Implements Flow.

4.2.3.8 setSystemBegin()

```
void FlowImplementation::setSystemBegin (
            System * system )  [override], [virtual]
```

Set systemBegin

Parameters

system	SystemBegin for the flow

Implements Flow.

4.2.3.9 setSystemEnd()

```
void FlowImplementation::setSystemEnd (
            System * system )  [override], [virtual]
```

Set systemBegin

Parameters

system	SystemEnd for the flow

Implements Flow.

4.2.3.10 setValue()

```
void FlowImplementation::setValue (
            double v )  [override], [virtual]
```

Set system value

Parameters

v	Value for the flow

Implements Flow.

4.2.4 Member Data Documentation

4.2.4.1 name

```
std::string FlowImplementation::name  [protected]
```

4.2.4.2 systemBegin

```
System* FlowImplementation::systemBegin  [protected]
```

4.2.4.3 systemEnd

```
System* FlowImplementation::systemEnd  [protected]
```

4.2.4.4 value

```
double FlowImplementation::value  [protected]
```

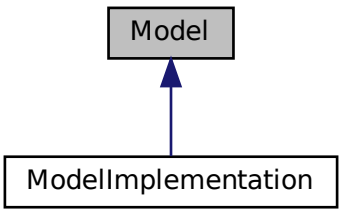
The documentation for this class was generated from the following files:

- src/lib/FlowImplementation.h
- src/lib/FlowImplementation.cpp

4.3 Model Class Reference

```
#include <Model.h>
```

Inheritance diagram for Model:



Public Member Functions

- virtual ~Model ()=default
- virtual void simulate (double, double, double)=0
- virtual std::string getName () const =0
- virtual void setName (std::string)=0
- virtual double getTime () const =0
- virtual void setTime (std::string)=0
- virtual void add (System *)=0
- virtual void add (Flow *)=0

4.3.1 Constructor & Destructor Documentation

4.3.1.1 ~Model()

```
virtual Model::~Model ( )  [virtual], [default]
```

4.3.2 Member Function Documentation

4.3.2.1 add() [1/2]

```
virtual void Model::add (
            Flow *  )  [pure virtual]
```

Implemented in ModelImplementation.

4.3.2.2 add() [2/2]

```
virtual void Model::add (
            System *  )  [pure virtual]
```

Implemented in ModelImplementation.

4.3.2.3 getName()

```
virtual std::string Model::getName ( ) const  [pure virtual]
```

Implemented in ModelImplementation.

4.3.2.4 getTime()

```
virtual double Model::getTime ( ) const  [pure virtual]
```

Implemented in ModelImplementation.

4.3.2.5 setName()

```
virtual void Model::setName (
            std::string  )  [pure virtual]
```

Implemented in ModelImplementation.

4.3.2.6 setTime()

```
virtual void Model::setTime (
            std::string  )  [pure virtual]
```

Implemented in ModelImplementation.

4.3.2.7 simulate()

```
virtual void Model::simulate (
            double ,
            double ,
            double  )  [pure virtual]
```

Implemented in ModelImplementation.

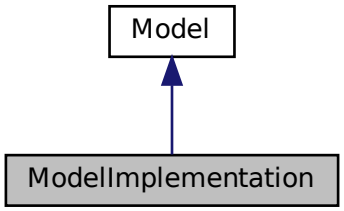
The documentation for this class was generated from the following file:

- src/lib/Model.h

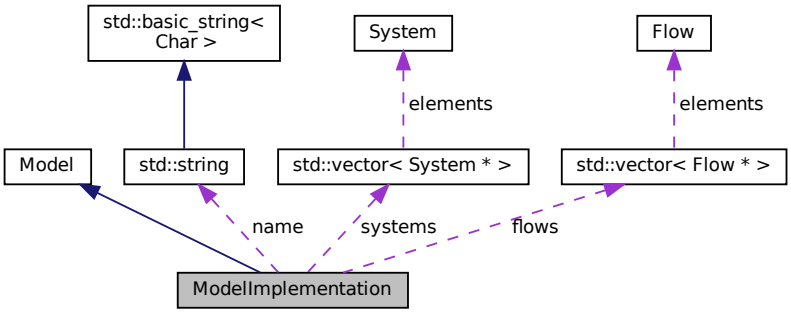
4.4 ModelImplementation Class Reference

`#include <ModelImplementation.h>`

Inheritance diagram for ModelImplementation:



Collaboration diagram for ModelImplementation:



Public Member Functions

- ~ModelImplementation () override
- ModelImplementation (std::string name, double time)
- ModelImplementation & operator= (const ModelImplementation &model)
- void simulate (double start, double end, double timestep) override
- std::string getName () const override
- void setName (std::string n) override
- double getTime () const override
- void setTime (std::string t) override
- void add (System *system) override
- void add (Flow *flow) override

Protected Attributes

- std::string name
- double time
- std::vector< System * > systems
- std::vector< Flow * > flows

4.4.1 Detailed Description

Model that simulates the energy flow through models

4.4.2 Constructor & Destructor Documentation

4.4.2.1 ~ModelImplementation()

```
ModelImplementation::~ModelImplementation ( )  [override]
```

Default destructor

4.4.2.2 ModelImplementation()

```
ModelImplementation::ModelImplementation (
            std::string name,
            double time )
```

Default constructor

Parameters

name	Inital model name
time	Inital model time

Returns

Model with initial name and time

4.4.3 Member Function Documentation

4.4.3.1 add() [1/2]

```
void ModelImplementation::add (
            Flow * flow )  [override], [virtual]
```

Add a flow to the model

Parameters

flow	Flow to be added to the model

Implements Model.

4.4.3.2 add() [2/2]

```
void ModelImplementation::add (
            System * system )  [override], [virtual]
```

Add a system to the model

Parameters

system	System to be added to the model

Implements Model.

4.4.3.3 getName()

```
std::string ModelImplementation::getName ( ) const  [override], [virtual]
```

Get model name

Implements Model.

4.4.3.4 getTime()

```
double ModelImplementation::getTime ( ) const  [override], [virtual]
```

Get model time

Implements Model.

4.4.3.5 operator=()

```
ModelImplementation & ModelImplementation::operator= (
            const ModelImplementation & model )
```

Copy Assignment Operator

Parameters

model	Model to copy from

Returns

Copied model

4.4.3.6 setName()

```
void ModelImplementation::setName (
            std::string n )  [override], [virtual]
```

Set model name

Parameters

n	Name for the system

Implements Model.

4.4.3.7 setTime()

```
void ModelImplementation::setTime (
            std::string t )  [override], [virtual]
```

Set model time

Parameters

t	Name for the system

Implements Model.

4.4.3.8 simulate()

```
void ModelImplementation::simulate (
            double start,
            double end,
            double timestep )  [override], [virtual]
```

Simulates the model during a period of time between start and end time values with a specified timestep

Parameters

start	Time where the simulation starts
end	Time where the simulation ends
timestep	Timestep value to simulate with

Implements Model.

4.4.4 Member Data Documentation

4.4.4.1 flows

```
std::vector<Flow*> ModelImplementation::flows  [protected]
```

4.4.4.2 name

```
std::string ModelImplementation::name  [protected]
```

4.4.4.3 systems

```
std::vector<System*> ModelImplementation::systems  [protected]
```

4.4.4.4 time

```
double ModelImplementation::time  [protected]
```

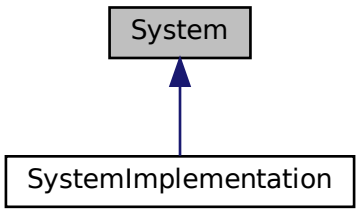
The documentation for this class was generated from the following files:

- src/lib/ModelImplementation.h
- src/lib/ModelImplementation.cpp

4.5 System Class Reference

```
#include <System.h>
```

Inheritance diagram for System:



Public Member Functions

- virtual ~System ()=default
- virtual std::string getName () const =0
- virtual void setName (std::string)=0
- virtual double getValue () const =0
- virtual void setValue (double)=0

4.5.1 Constructor & Destructor Documentation

4.5.1.1 ~System()

```
virtual System::~System ( )  [virtual], [default]
```

4.5.2 Member Function Documentation

4.5.2.1 getName()

```
virtual std::string System::getName ( ) const  [pure virtual]
```

Implemented in SystemImplementation.

4.5.2.2 getValue()

```
virtual double System::getValue ( ) const  [pure virtual]
```

Implemented in SystemImplementation.

4.5.2.3 setName()

```
virtual void System::setName (
            std::string  )  [pure virtual]
```

Implemented in SystemImplementation.

4.5.2.4 setValue()

```
virtual void System::setValue (
            double  )  [pure virtual]
```

Implemented in SystemImplementation.

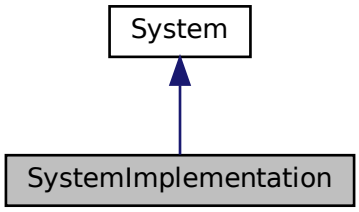
The documentation for this class was generated from the following file:

- src/lib/System.h

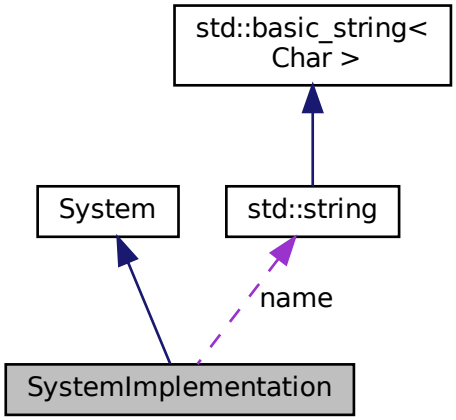
4.6 SystemImplementation Class Reference

`#include <SystemImplementation.h>`

Inheritance diagram for SystemImplementation:



Collaboration diagram for SystemImplementation:



Public Member Functions

- ~SystemImplementation () override
- SystemImplementation (std::string name, double value)
- SystemImplementation & operator= (const SystemImplementation &system)
- std::string getName () const override
- void setName (std::string n) override
- double getValue () const override
- void setValue (double v) override

Protected Attributes

- std::string name
- double value

4.6.1 Detailed Description

System that stores energy

4.6.2 Constructor & Destructor Documentation

4.6.2.1 ~SystemImplementation()

```
SystemImplementation::~SystemImplementation ( )  [override], [default]
```

Default destructor

4.6.2.2 SystemImplementation()

```
SystemImplementation::SystemImplementation (
            std::string name,
            double value )
```

Default constructor

Parameters

name	Inital system name
value	Inital system value

Returns

System with initial name and value

4.6.3 Member Function Documentation

4.6.3.1 getName()

```
std::string SystemImplementation::getName ( ) const  [override], [virtual]
```

Get system name

Implements System.

4.6.3.2 getValue()

```
double SystemImplementation::getValue ( ) const  [override], [virtual]
```

Get system value

Implements System.

4.6.3.3 operator=()

```
SystemImplementation & SystemImplementation::operator= (
            const SystemImplementation & system )
```

Copy Assignment Operator

Parameters

system	System to copy from

Returns

Copied system

4.6.3.4 setName()

```
void SystemImplementation::setName (
            std::string n )  [override], [virtual]
```

Set system name

Parameters

n	Name for the system

Implements System.

4.6.3.5 setValue()

```
void SystemImplementation::setValue (
            double v )  [override], [virtual]
```

Set system value

Parameters

v	Value for the system

Implements System.

4.6.4 Member Data Documentation

4.6.4.1 name

```
std::string SystemImplementation::name  [protected]
```

4.6.4.2 value

```
double SystemImplementation::value  [protected]
```

The documentation for this class was generated from the following files:

- src/lib/SystemImplementation.h
- src/lib/SystemImplementation.cpp

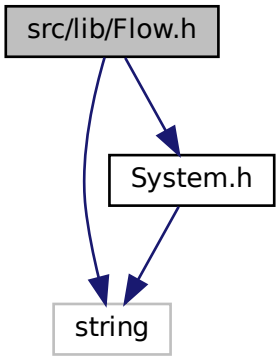
Chapter 5

File Documentation

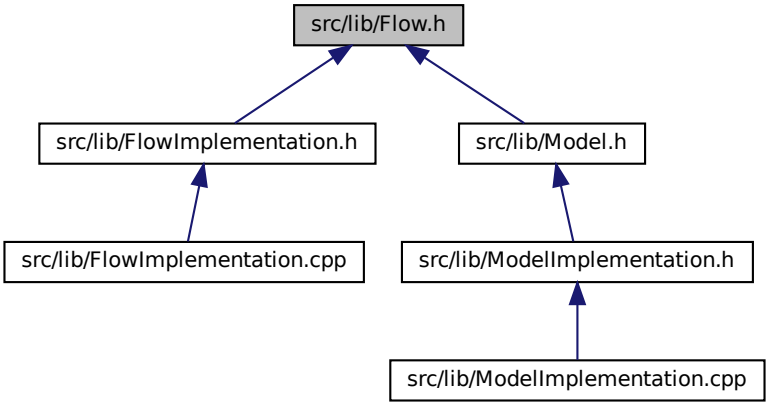
5.1 src/lib/Flow.h File Reference

```
#include <string>
#include "System.h"
```

Include dependency graph for Flow.h:



This graph shows which files directly or indirectly include this file:



Classes

- class Flow

5.2 Flow.h

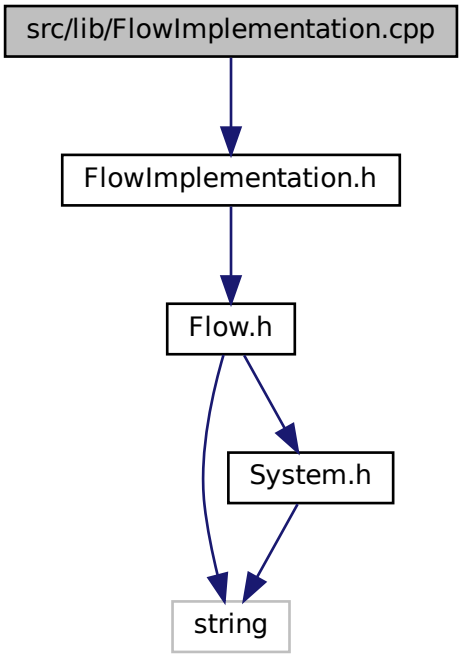
Go to the documentation of this file.

```
1 //
2 // Created by joaozenobio on 27/04/2022.
3 //
4
5 #ifndef ENG1_FLOW_H
6 #define ENG1_FLOW_H
7
8 #include <string>
9
10 #include "System.h"
11
12 class Flow {
13 public:
14     virtual ~Flow() = default;
15     virtual std::string getName() const = 0;
16     virtual void setName(std::string) = 0;
17     virtual double getValue() const = 0;
18     virtual void setValue(double) = 0;
19     virtual void expression() = 0;
20     virtual System* getSystemBegin() const = 0;
21     virtual void setSystemBegin(System*) = 0;
22     virtual System* getSystemEnd() const = 0;
23     virtual void setSystemEnd(System*) = 0;
24 };
25
26
27 #endif //ENG1_FLOW_H
```

5.3 src/lib/FlowImplementation.cpp File Reference

```
#include "FlowImplementation.h"
```

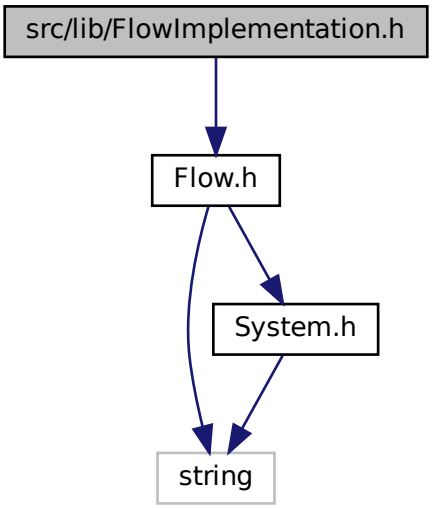
Include dependency graph for FlowImplementation.cpp:



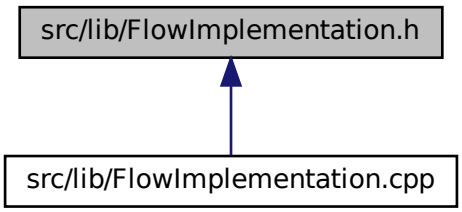
5.4 src/lib/FlowImplementation.h File Reference

```
#include "Flow.h"
```

Include dependency graph for FlowImplementation.h:



This graph shows which files directly or indirectly include this file:



Classes

- class FlowImplementation

5.5 FlowImplementation.h

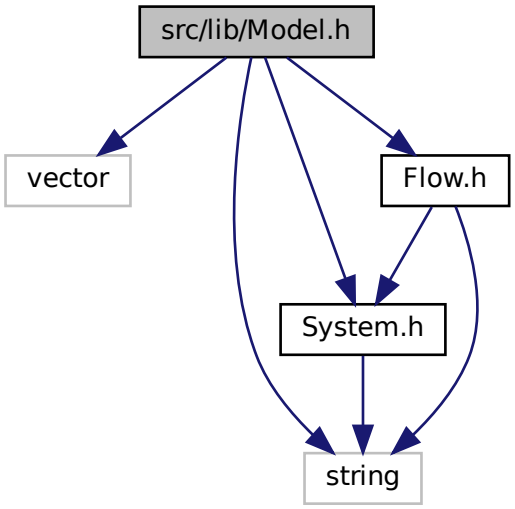
Go to the documentation of this file.

```
1 //
2 // Created by joaozenobio on 28/04/2022.
3 //
4
5 #ifndef ENG1_FLOWIMPLEMENTATION_H
6 #define ENG1_FLOWIMPLEMENTATION_H
7
8
9 #include "Flow.h"
10
14 class FlowImplementation : public Flow {
15
16 private:
22     FlowImplementation(const FlowImplementation& flow);
23
24 protected:
25     std::string name;
26     double value;
27     System* systemBegin;
28     System* systemEnd;
29
30 public:
34     ~FlowImplementation() override;
35
44     FlowImplementation(std::string name, double value, System* systemBegin, System* systemEnd);
45
51     FlowImplementation& operator=(const Flow& flow);
52
56     void expression() override = 0;
57
61     std::string getName() const override;
62
67     void setName(std::string n) override;
68
72     double getValue() const override;
73
78     void setValue(double v) override;
79
83     System* getSystemBegin() const override;
84
89     void setSystemBegin(System* system) override;
90
94     System* getSystemEnd() const override;
95
100     void setSystemEnd(System* system) override;
101 };
102
103
104 #endif //ENG1_FLOWIMPLEMENTATION_H
```

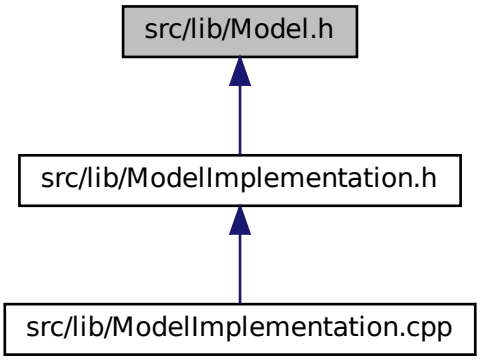
5.6 src/lib/Model.h File Reference

```
#include <vector>
#include <string>
#include "System.h"
#include "Flow.h"
```

Include dependency graph for Model.h:



This graph shows which files directly or indirectly include this file:



Classes

- class Model

5.7 Model.h

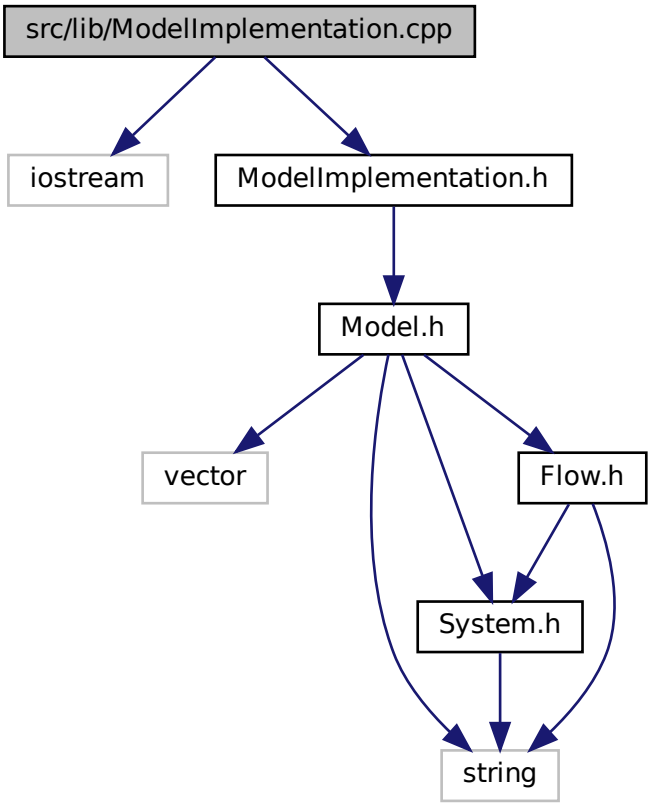
Go to the documentation of this file.

```
1 //
2 // Created by joaozenobio on 27/04/2022.
3 //
4
5 #ifndef ENG1_MODEL_H
6 #define ENG1_MODEL_H
7
8
9 #include <vector>
10 #include <string>
11
12 #include "System.h"
13 #include "Flow.h"
14
15 class Model {
16 public:
17     virtual ~Model() = default;
18     virtual void simulate(double, double, double) = 0;
19     virtual std::string getName() const = 0;
20     virtual void setName(std::string) = 0;
21     virtual double getTime() const = 0;
22     virtual void setTime(std::string) = 0;
23     virtual void add(System*) = 0;
24     virtual void add(Flow*) = 0;
25 };
26
27
28 #endif //ENG1_MODEL_H
```

5.8 src/lib/ModelImplementation.cpp File Reference

```
#include <iostream>
#include "ModelImplementation.h"
```

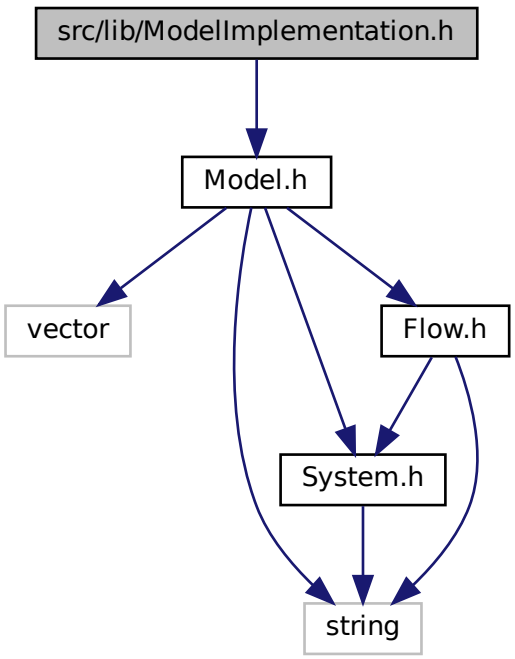
Include dependency graph for ModelImplementation.cpp:



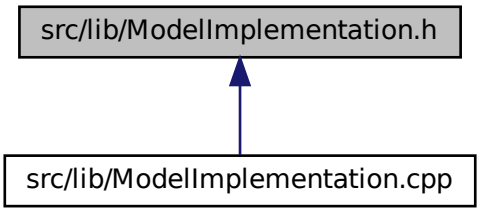
5.9 src/lib/ModelImplementation.h File Reference

```
#include "Model.h"
```

Include dependency graph for ModelImplementation.h:



This graph shows which files directly or indirectly include this file:



Classes

- class ModelImplementation

5.10 ModelImplementation.h

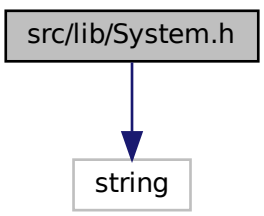
Go to the documentation of this file.

```
1 //
2 // Created by joaozenobio on 28/04/2022.
3 //
4
5 #ifndef ENG1_MODELIMPLEMENTATION_H
6 #define ENG1_MODELIMPLEMENTATION_H
7
8
9 #include "Model.h"
10
14 class ModelImplementation: public Model {
15
16 private:
22     ModelImplementation(const ModelImplementation& model);
23
24 protected:
25     std::string name;
26     double time;
27     std::vector<System*> systems;
28     std::vector<Flow*> flows;
29
30 public:
34     ~ModelImplementation() override;
35
42     ModelImplementation(std::string name, double time);
43
49     ModelImplementation& operator=(const ModelImplementation& model);
50
57     void simulate(double start, double end, double timestep) override;
58
62     std::string getName() const override;
63
68     void setName(std::string n) override;
69
73     double getTime() const override;
74
79     void setTime(std::string t) override;
80
85     void add(System* system) override;
86
91     void add(Flow* flow) override;
92 };
93
94
95 #endif //ENG1_MODELIMPLEMENTATION_H
```

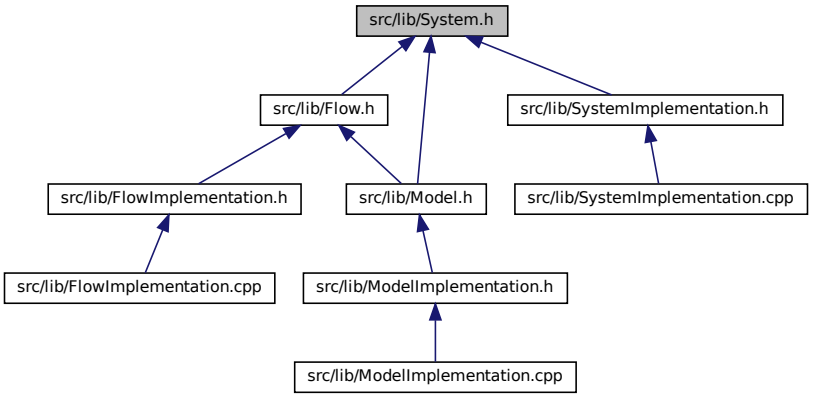
5.11 src/lib/System.h File Reference

`#include <string>`

Include dependency graph for System.h:



This graph shows which files directly or indirectly include this file:



Classes

- class System

5.12 System.h

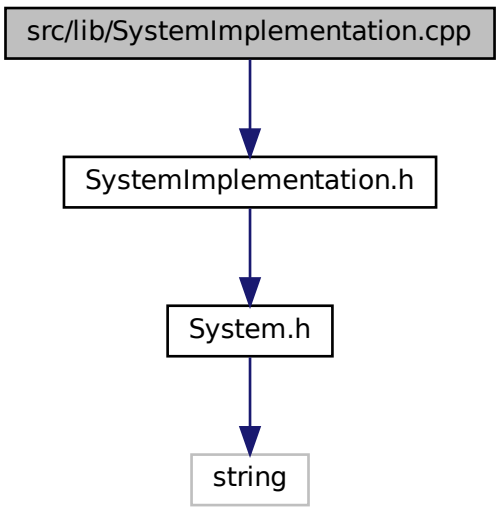
Go to the documentation of this file.

```
1 //
2 // Created by joaozenobio on 27/04/2022.
3 //
4
5 #ifndef ENG1_SYSTEM_H
6 #define ENG1_SYSTEM_H
7
8 #include <string>
9
10 class System {
11 public:
12     virtual ~System() = default;
13     virtual std::string getName() const = 0;
14     virtual void setName(std::string) = 0;
15     virtual double getValue() const = 0;
16     virtual void setValue(double) = 0;
17 };
18
19
20 #endif //ENG1_SYSTEM_H
```

5.13 src/lib/SystemImplementation.cpp File Reference

```
#include "SystemImplementation.h"
```

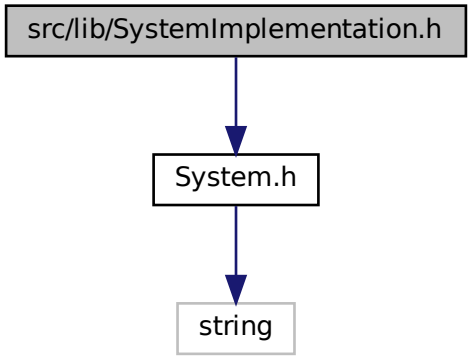
Include dependency graph for SystemImplementation.cpp:



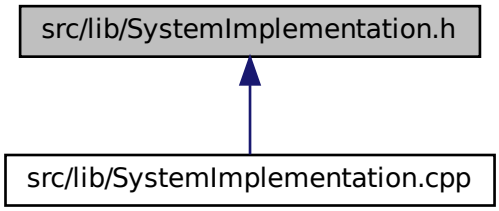
5.14 src/lib/SystemImplementation.h File Reference

```
#include "System.h"
```

Include dependency graph for SystemImplementation.h:



This graph shows which files directly or indirectly include this file:



Classes

- class SystemImplementation

5.15 SystemImplementation.h

Go to the documentation of this file.

```
1 //
2 // Created by joaozenobio on 28/04/2022.
3 //
4
5 #ifndef ENG1_SYSTEMIMPLEMENTATION_H
6 #define ENG1_SYSTEMIMPLEMENTATION_H
7
8 #include "System.h"
9
13 class SystemImplementation : public System {
14
15 private:
21     SystemImplementation(const SystemImplementation& system);
22
23 protected:
24     std::string name;
25     double value;
26
27 public:
31     ~SystemImplementation() override;
32
39     SystemImplementation(std::string name, double value);
40
46     SystemImplementation& operator=(const SystemImplementation& system);
47
51     std::string getName() const override;
52
57     void setName(std::string n) override;
58
62     double getValue() const override;
63
68     void setValue(double v) override;
69 };
70
71
72 #endif //ENG1_SYSTEMIMPLEMENTATION_H
```

Index