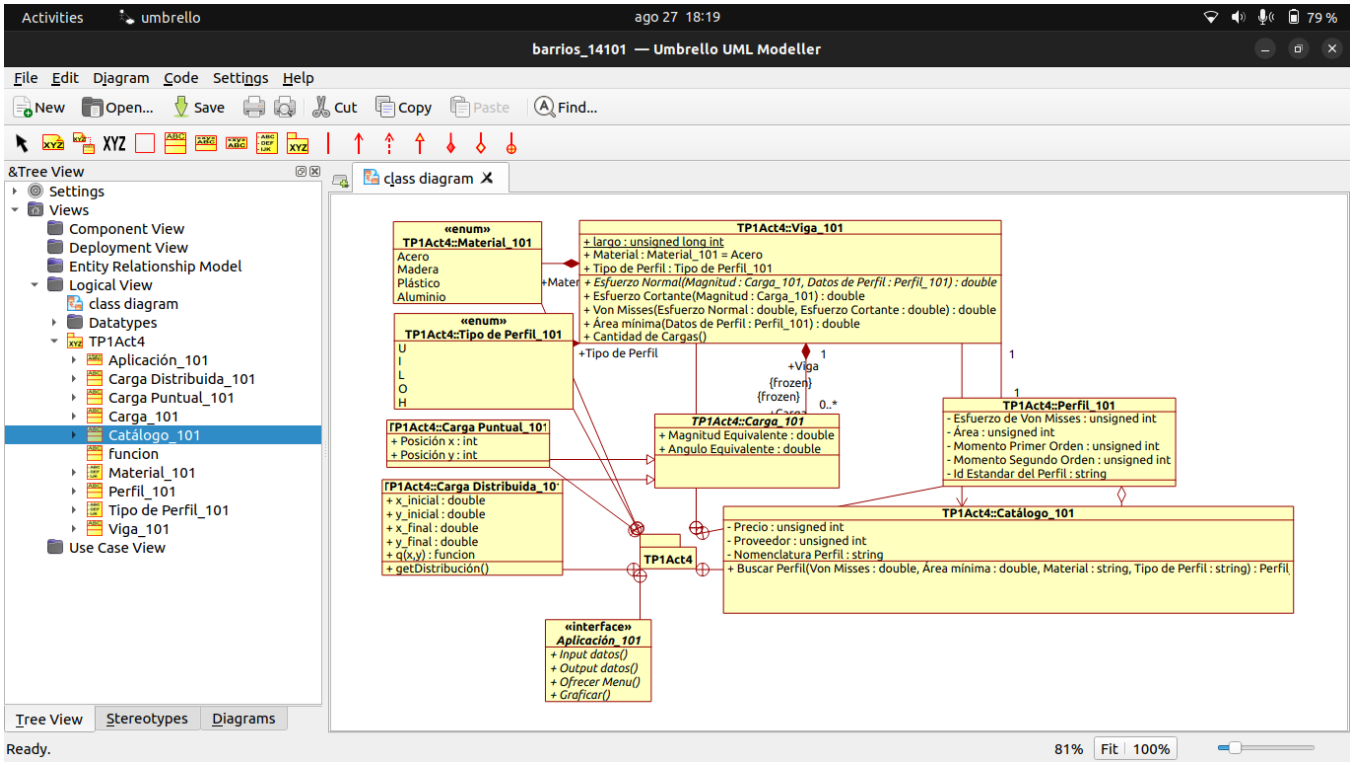
Trabajo Práctico N° 1 – Plataforma de desarrollo

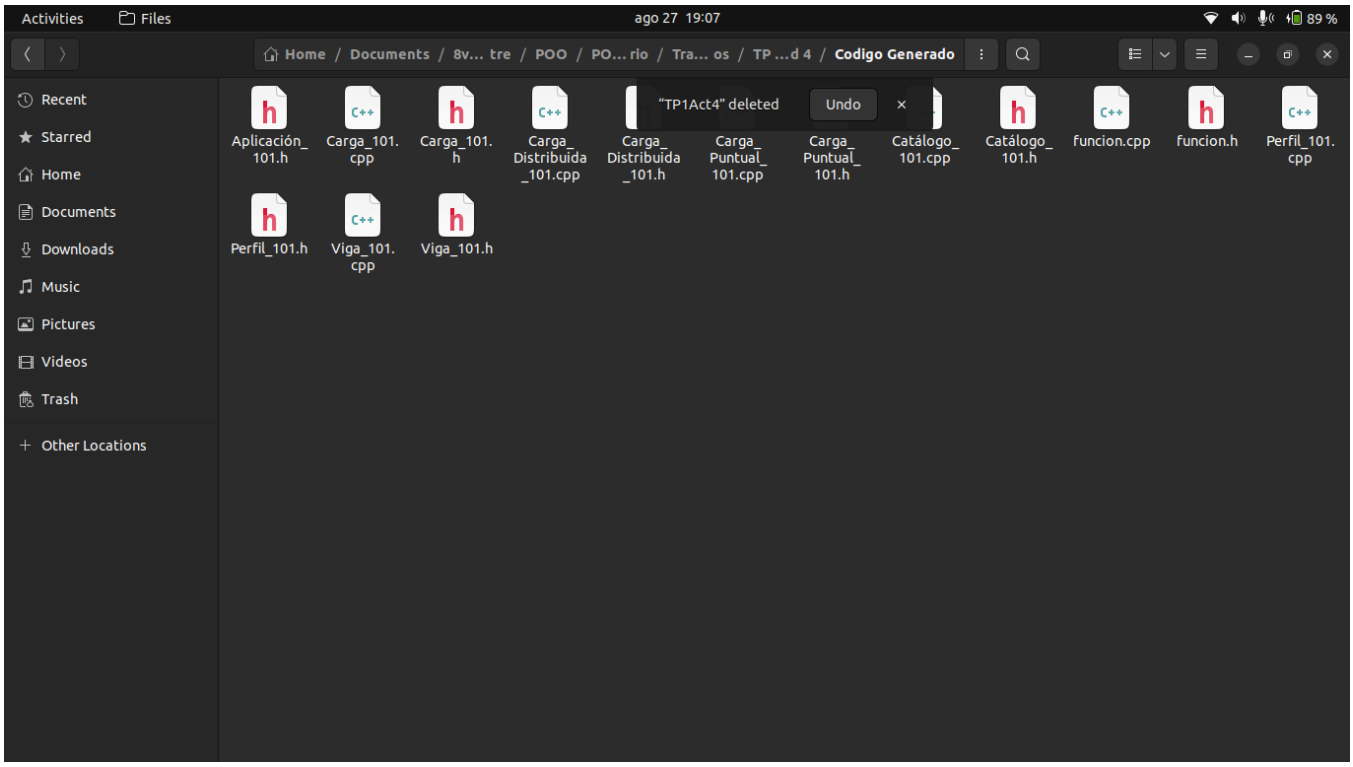
Actividad 4

A continuación, una serie de screenshots que demuestran el uso de una máquina virtual de sistema Ubuntu para programar y ejecutar programas de C++.

Diagrama de clase en Umbrello:

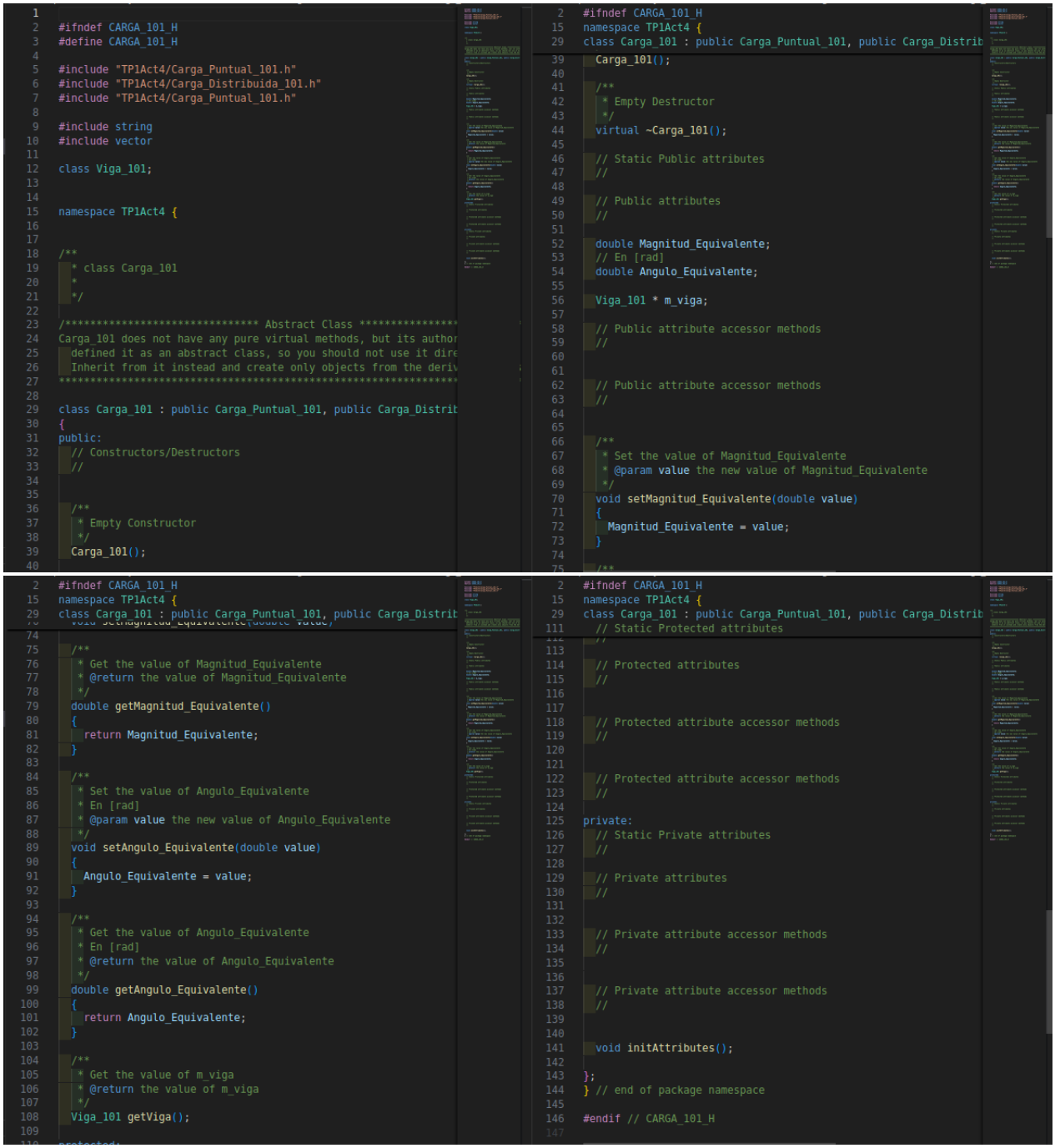


Estructura Física implementada:



Captura de pantalla de los módulos de cada clase:

Carga



```
#ifndef CARGA_101_H
#define CARGA_101_H

#include "TP1Act4/Carga_Puntual_101.h"
#include "TP1Act4/Carga_Distribuida_101.h"
#include "TP1Act4/Carga_Puntual_101.h"

#include string
#include vector

class Viga_101;


namespace TP1Act4 {


/**
 * class Carga_101
 *
 */

/****************************** Abstract Class ****************
Carga_101 does not have any pure virtual methods, but its author
  defined it as an abstract class, so you should not use it dire
  Inherit from it instead and create only objects from the deriv
**************************************************************/

class Carga_101 : public Carga_Puntual_101, public Carga_Distrit
{
public:
  // Constructors/Destructors
  //


  /**
   * Empty Constructor
   */
  Carga_101();

  /**
   * Empty Destructor
   */
  virtual ~Carga_101();

  // Static Public attributes
  //

  // Public attributes
  //

  double Magnitud_Equivalente;
  // En [rad]
  double Angulo_Equivalente;

  Viga_101 * m_viga;

  // Public attribute accessor methods
  //


  // Public attribute accessor methods
  //


  /**
   * Set the value of Magnitud_Equivalente
   * @param value the new value of Magnitud_Equivalente
   */
  void setMagnitud_Equivalente(double value)
  {
    Magnitud_Equivalente = value;
  }

  /**
   * Get the value of Magnitud_Equivalente
   * @return the value of Magnitud_Equivalente
   */
  double getMagnitud_Equivalente()
  {
    return Magnitud_Equivalente;
  }

  /**
   * Set the value of Angulo_Equivalente
   * En [rad]
   * @param value the new value of Angulo_Equivalente
   */
  void setAngulo_Equivalente(double value)
  {
    Angulo_Equivalente = value;
  }

  /**
   * Get the value of Angulo_Equivalente
   * En [rad]
   * @return the value of Angulo_Equivalente
   */
  double getAngulo_Equivalente()
  {
    return Angulo_Equivalente;
  }

  /**
   * Get the value of m_viga
   * @return the value of m_viga
   */
  Viga_101 getViga();

protected:
  // Static Protected attributes
  //

  // Protected attributes
  //


  // Protected attribute accessor methods
  //


  // Protected attribute accessor methods
  //

private:
  // Static Private attributes
  //

  // Private attributes
  //


  // Private attribute accessor methods
  //


  // Private attribute accessor methods
  //


  void initAttributes();

};
} // end of package namespace

#endif // CARGA_101_H
```

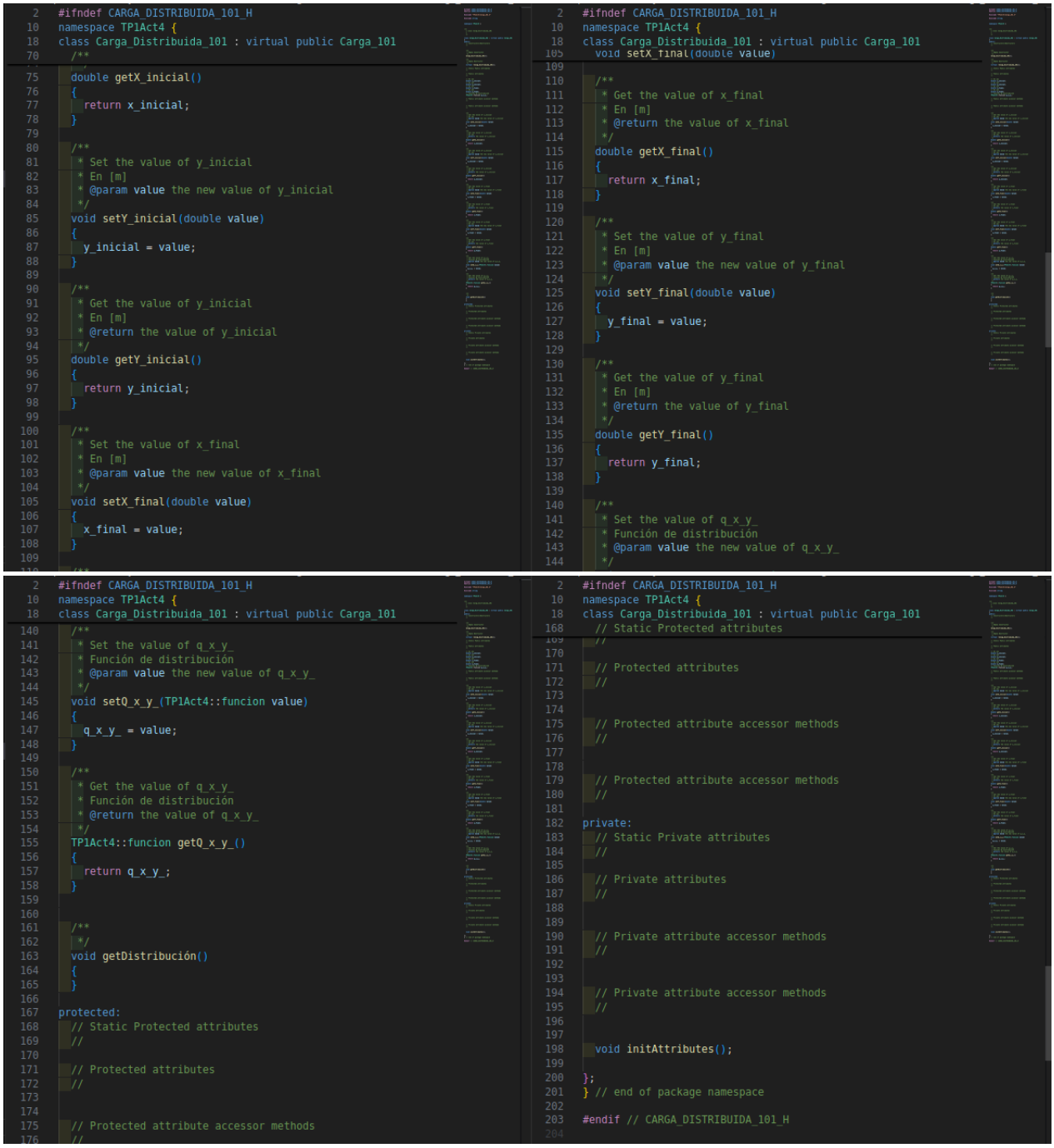
```cpp
#include "Carga_101.h"

// Constructors/Destructors
//

Carga_101::Carga_101()
{
  initAttributes();
}

Carga_101::~Carga_101()
{
}

//
// Methods
//


// Accessor methods
//


/**
 * Get the value of m_viga
 * @return the value of m_viga
 */
Viga_101 * Carga_101::getViga()
{
  return m_viga;
}

// Other methods
//

void Carga_101::initAttributes()
{
  m_viga = new Viga_101();
}

```

Carga Distribuida

```cpp
#include "Carga_Distribuida_101.h"

// Constructors/Destructors
//

Carga_Distribuida_101::Carga_Distribuida_101()
{
  initAttributes();
}

Carga_Distribuida_101::~Carga_Distribuida_101()
{
}

//
// Methods
//


// Accessor methods
//


// Other methods
//

void Carga_Distribuida_101::initAttributes()
{
}
```

```cpp

#ifndef CARGA_DISTRIBUIDA_101_H
#define CARGA_DISTRIBUIDA_101_H

#include "TP1Act4/Carga_101.h"

#include string


namespace TP1Act4 {


/**
 * class Carga_Distribuida_101
 *
 */

class Carga_Distribuida_101 : virtual public Carga_101
{
public:
  // Constructors/Destructors
  //


  /**
   * Empty Constructor
   */
  Carga_Distribuida_101();

  /**
   * Empty Destructor
   */
  virtual ~Carga_Distribuida_101();

  // Static Public attributes
  //

  // Public attributes
  //
```

```cpp
#ifndef CARGA_DISTRIBUIDA_101_H
namespace TP1Act4 {
class Carga_Distribuida_101 : virtual public Carga_101
  // Public attributes

  // En [m]
  double x_inicial;
  // En [m]
  double y_inicial;
  // En [m]
  double x_final;
  // En [m]
  double y_final;
  // Función de distribución
  TP1Act4::funcion q_x_y_;

  // Public attribute accessor methods
  //


  // Public attribute accessor methods
  //


  /**
   * Set the value of x_inicial
   * En [m]
   * @param value the new value of x_inicial
   */
  void setX_inicial(double value)
  {
    x_inicial = value;
  }

  /**
   * Get the value of x_inicial
   * En [m]
   * @return the value of x_inicial
   */
  double getX_inicial()
```



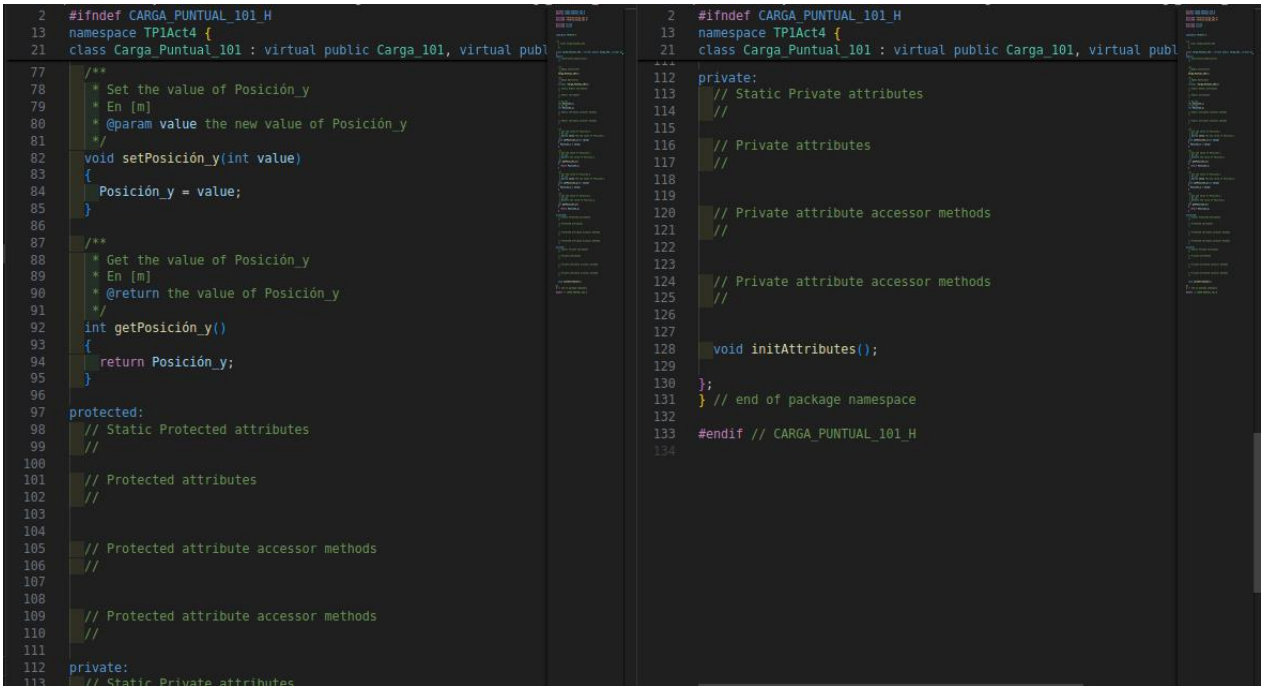
```
#ifndef CARGA_DISTRIBUIDA_101_H
namespace TP1Act4 {
class Carga_Distribuida_101 : virtual public Carga_101
  /**
  double getX_inicial()
  {
    return x_inicial;
  }

  /**
   * Set the value of y_inicial
   * En [m]
   * @param value the new value of y_inicial
   */
  void setY_inicial(double value)
  {
    y_inicial = value;
  }

  /**
   * Get the value of y_inicial
   * En [m]
   * @return the value of y_inicial
   */
  double getY_inicial()
  {
    return y_inicial;
  }

  /**
   * Set the value of x_final
   * En [m]
   * @param value the new value of x_final
   */
  void setX_final(double value)
  {
    x_final = value;
  }

#ifndef CARGA_DISTRIBUIDA_101_H
namespace TP1Act4 {
class Carga_Distribuida_101 : virtual public Carga_101
  void setX_final(double value)

  /**
   * Get the value of x_final
   * En [m]
   * @return the value of x_final
   */
  double getX_final()
  {
    return x_final;
  }

  /**
   * Set the value of y_final
   * En [m]
   * @param value the new value of y_final
   */
  void setY_final(double value)
  {
    y_final = value;
  }

  /**
   * Get the value of y_final
   * En [m]
   * @return the value of y_final
   */
  double getY_final()
  {
    return y_final;
  }

  /**
   * Set the value of q_x_y_
   * Función de distribución
   * @param value the new value of q_x_y_
   */

#ifndef CARGA_DISTRIBUIDA_101_H
namespace TP1Act4 {
class Carga_Distribuida_101 : virtual public Carga_101
  /**
   * Set the value of q_x_y_
   * Función de distribución
   * @param value the new value of q_x_y_
   */
  void setQ_x_y_(TP1Act4::funcion value)
  {
    q_x_y_ = value;
  }

  /**
   * Get the value of q_x_y_
   * Función de distribución
   * @return the value of q_x_y_
   */
  TP1Act4::funcion getQ_x_y_()
  {
    return q_x_y_;
  }


  /**
   */
  void getDistribución()
  {
  }

protected:
  // Static Protected attributes
  //

  // Protected attributes
  //


  // Protected attribute accessor methods

#ifndef CARGA_DISTRIBUIDA_101_H
namespace TP1Act4 {
class Carga_Distribuida_101 : virtual public Carga_101
  // Static Protected attributes
  //

  // Protected attributes
  //


  // Protected attribute accessor methods
  //


  // Protected attribute accessor methods
  //

private:
  // Static Private attributes
  //

  // Private attributes
  //


  // Private attribute accessor methods
  //


  // Private attribute accessor methods
  //


  void initAttributes();

};
} // end of package namespace

#endif // CARGA_DISTRIBUIDA_101_H
```

Carga Puntual

```cpp
#include "Carga_Puntual_101.h"

// Constructors/Destructors
//

Carga_Puntual_101::Carga_Puntual_101()
{
  initAttributes();
}

Carga_Puntual_101::~Carga_Puntual_101()
{
}

//
// Methods
//


// Accessor methods
//


// Other methods
//

void Carga_Puntual_101::initAttributes()
{
}
```

```cpp
#ifndef CARGA_PUNTUAL_101_H
#define CARGA_PUNTUAL_101_H

#include "TP1Act4/Carga_101.h"
#include "TP1Act4/Carga_101.h"

#include string
#include vector


namespace TP1Act4 {


/**
 * class Carga_Puntual_101
 *
 */

class Carga_Puntual_101 : virtual public Carga_101, virtual publ
{
public:
  // Constructors/Destructors
  //


  /**
   * Empty Constructor
   */
  Carga_Puntual_101();

  /**
   * Empty Destructor
   */
  virtual ~Carga_Puntual_101();

  // Static Public attributes
  //
```

```cpp
#ifndef CARGA_PUNTUAL_101_H
namespace TP1Act4 {
class Carga_Puntual_101 : virtual public Carga_101, virtual publ
  // Public attributes


  // En [m]
  int Posición_x;
  // En [m]
  int Posición_y;

  // Public attribute accessor methods
  //


  // Public attribute accessor methods
  //


  /**
   * Set the value of Posición_x
   * En [m]
   * @param value the new value of Posición_x
   */
  void setPosición_x(int value)
  {
    Posición_x = value;
  }

  /**
   * Get the value of Posición_x
   * En [m]
   * @return the value of Posición_x
   */
  int getPosición_x()
  {
    return Posición_x;
  }

  /**
   * Set the value of Posición_y
```

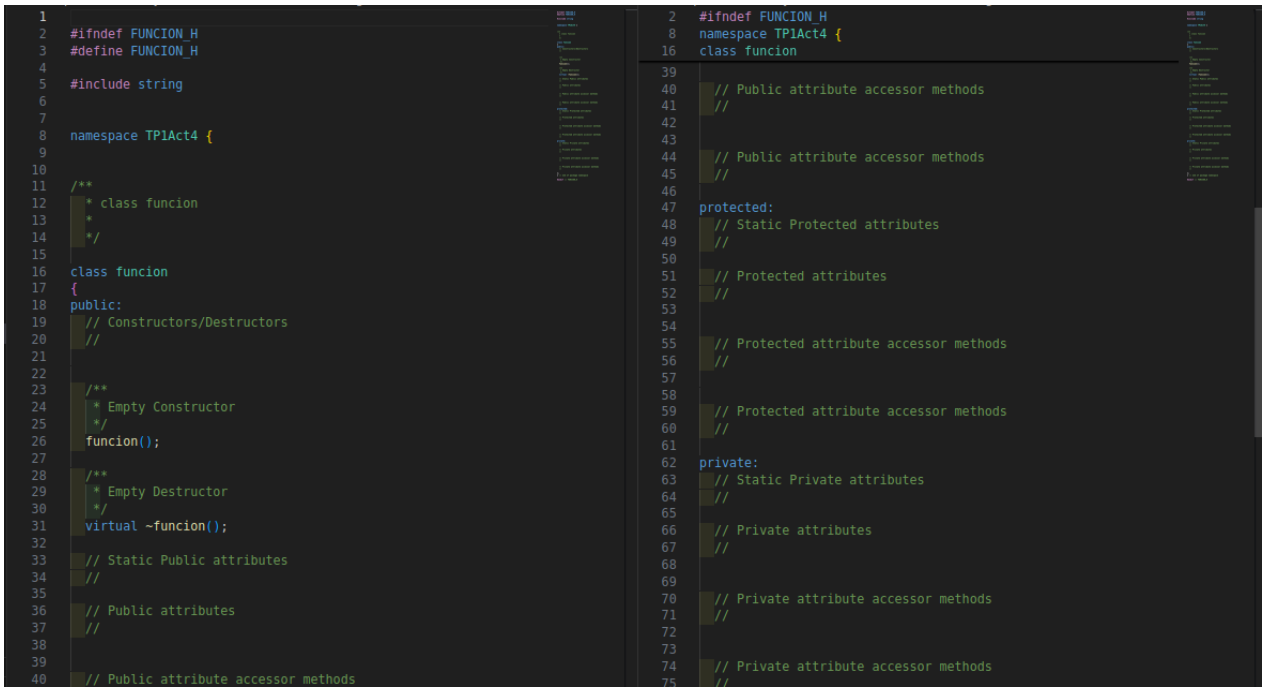


```
#ifndef CARGA_PUNTUAL_101_H
namespace TP1Act4 {
class Carga_Puntual_101 : virtual public Carga_101, virtual publ
  /**
   * Set the value of Posición_y
   * En [m]
   * @param value the new value of Posición_y
   */
  void setPosición_y(int value)
  {
    Posición_y = value;
  }

  /**
   * Get the value of Posición_y
   * En [m]
   * @return the value of Posición_y
   */
  int getPosición_y()
  {
    return Posición_y;
  }

protected:
  // Static Protected attributes
  //

  // Protected attributes
  //


  // Protected attribute accessor methods
  //


  // Protected attribute accessor methods
  //

private:
  // Static Private attributes
```

```
#ifndef CARGA_PUNTUAL_101_H
namespace TP1Act4 {
class Carga_Puntual_101 : virtual public Carga_101, virtual publ
private:
  // Static Private attributes
  //

  // Private attributes
  //


  // Private attribute accessor methods
  //


  // Private attribute accessor methods
  //


  void initAttributes();

};
} // end of package namespace

#endif // CARGA_PUNTUAL_101_H
```

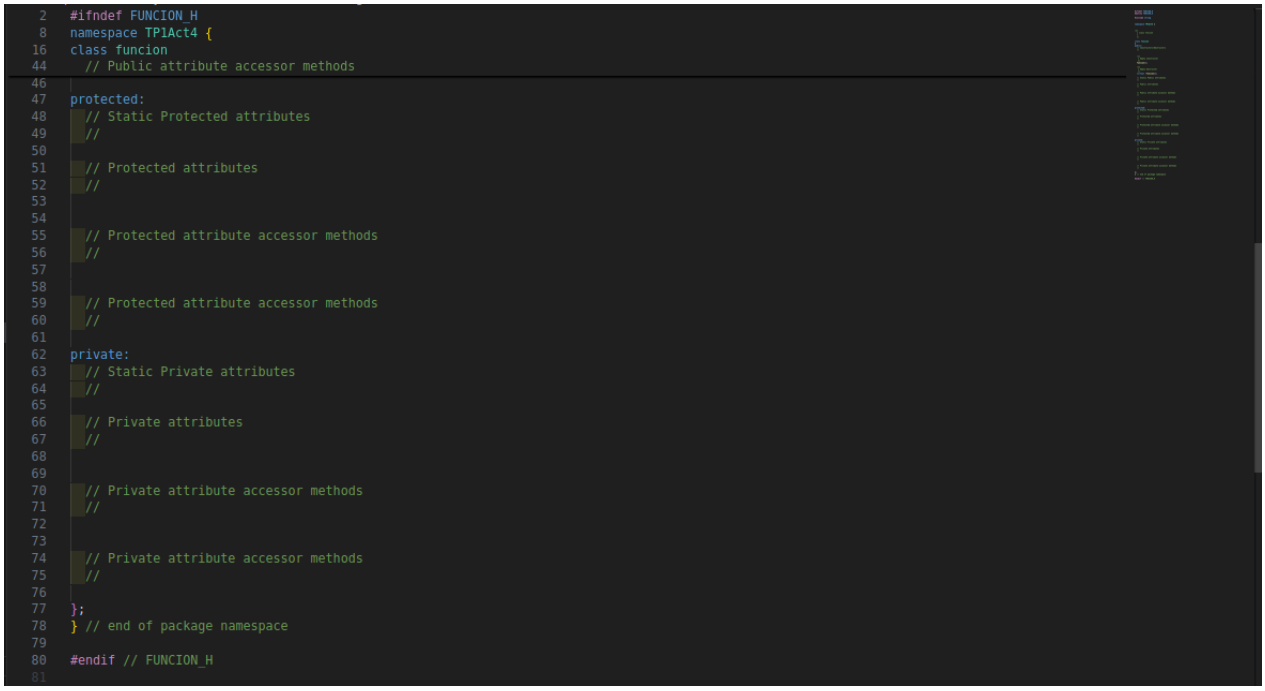
Función

```cpp
#include "funcion.h"

// Constructors/Destructors
//

funcion::funcion()
{
}

funcion::~funcion()
{
}

//
// Methods
//


// Accessor methods
//


// Other methods
//
```



```cpp
#ifndef FUNCION_H
#define FUNCION_H

#include string


namespace TP1Act4 {


/**
 * class funcion
 *
 */

class funcion
{
public:
  // Constructors/Destructors
  //


  /**
   * Empty Constructor
   */
  funcion();

  /**
   * Empty Destructor
   */
  virtual ~funcion();

  // Static Public attributes
  //

  // Public attributes
  //


  // Public attribute accessor methods
```

```cpp
#ifndef FUNCION_H
namespace TP1Act4 {
class funcion

  // Public attribute accessor methods
  //


  // Public attribute accessor methods
  //

protected:
  // Static Protected attributes
  //

  // Protected attributes
  //


  // Protected attribute accessor methods
  //


  // Protected attribute accessor methods
  //

private:
  // Static Private attributes
  //

  // Private attributes
  //


  // Private attribute accessor methods
  //


  // Private attribute accessor methods
```



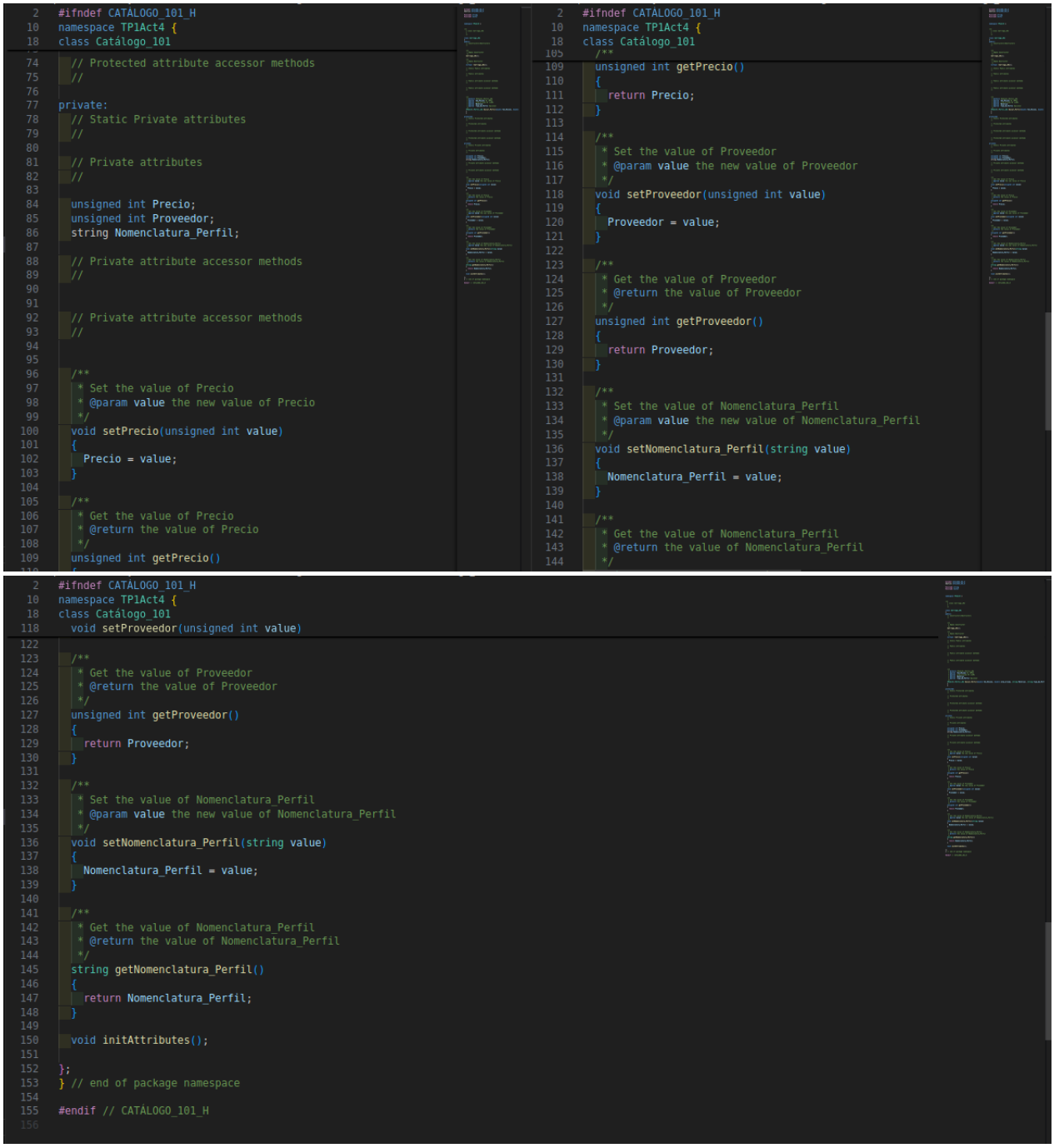
```
#ifndef FUNCION_H
namespace TP1Act4 {
class funcion
  // Public attribute accessor methods

protected:
  // Static Protected attributes
  //

  // Protected attributes
  //


  // Protected attribute accessor methods
  //


  // Protected attribute accessor methods
  //

private:
  // Static Private attributes
  //

  // Private attributes
  //


  // Private attribute accessor methods
  //


  // Private attribute accessor methods
  //

};
} // end of package namespace

#endif // FUNCION_H
```

Catálogo

```cpp
#include "Catálogo_101.h"

// Constructors/Destructors
//

Catálogo_101::Catálogo_101()
{
  initAttributes();
}

Catálogo_101::~Catálogo_101()
{
}

//
// Methods
//


// Accessor methods
//


// Other methods
//

void Catálogo_101::initAttributes()
{
}
```

```cpp
#ifndef CATÁLOGO_101_H
#define CATÁLOGO_101_H

#include string
#include vector



namespace TP1Act4 {


/**
 * class Catálogo_101
 *
 */

class Catálogo_101
{
public:
  // Constructors/Destructors
  //


  /**
   * Empty Constructor
   */
  Catálogo_101();

  /**
   * Empty Destructor
   */
  virtual ~Catálogo_101();

  // Static Public attributes
  //

  // Public attributes
  //
```

```cpp
#ifndef CATÁLOGO_101_H
namespace TP1Act4 {
class Catálogo_101
  // Public attributes



  // Public attribute accessor methods
  //


  // Public attribute accessor methods
  //



  /**
   * @return TP1Act4::Perfil_101
   * @param  Von_Misses En [MPa]
   * @param  Area mínima En [cm²]
   * @param  Material
   * @param  Tipo_de_Perfil Opcional
   */
  TP1Act4::Perfil_101 Buscar_Perfil(double Von_Misses, double Ár
  {
  }

protected:
  // Static Protected attributes
  //

  // Protected attributes
  //


  // Protected attribute accessor methods
  //


  // Protected attribute accessor methods
```



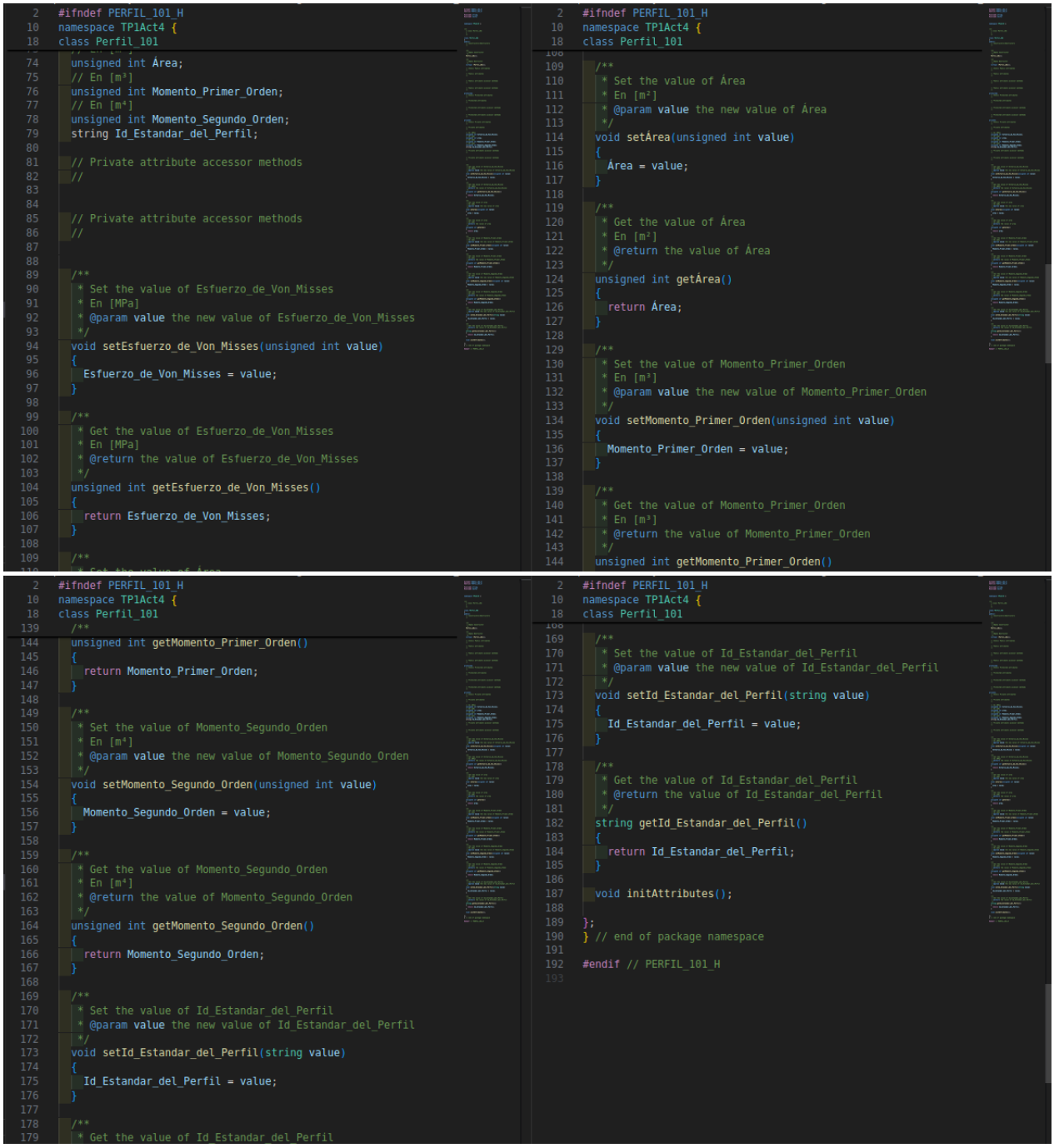
```
#ifndef CATALOGO_101_H
namespace TP1Act4 {
class Catálogo_101

  // Protected attribute accessor methods
  //

private:
  // Static Private attributes
  //

  // Private attributes
  //

  unsigned int Precio;
  unsigned int Proveedor;
  string Nomenclatura_Perfil;

  // Private attribute accessor methods
  //


  // Private attribute accessor methods
  //


  /**
   * Set the value of Precio
   * @param value the new value of Precio
   */
  void setPrecio(unsigned int value)
  {
    Precio = value;
  }

  /**
   * Get the value of Precio
   * @return the value of Precio
   */
  unsigned int getPrecio()
  {
    return Precio;
  }

  /**
   * Set the value of Proveedor
   * @param value the new value of Proveedor
   */
  void setProveedor(unsigned int value)
  {
    Proveedor = value;
  }

  /**
   * Get the value of Proveedor
   * @return the value of Proveedor
   */
  unsigned int getProveedor()
  {
    return Proveedor;
  }

  /**
   * Set the value of Nomenclatura_Perfil
   * @param value the new value of Nomenclatura_Perfil
   */
  void setNomenclatura_Perfil(string value)
  {
    Nomenclatura_Perfil = value;
  }

  /**
   * Get the value of Nomenclatura_Perfil
   * @return the value of Nomenclatura_Perfil
   */
  string getNomenclatura_Perfil()
  {
    return Nomenclatura_Perfil;
  }

  void initAttributes();

};
} // end of package namespace

#endif // CATALOGO_101_H
```

Perfil

```cpp
#include "Perfil_101.h"

// Constructors/Destructors
//

Perfil_101::Perfil_101()
{
  initAttributes();
}

Perfil_101::~Perfil_101()
{
}

//
// Methods
//


// Accessor methods
//


// Other methods
//

void Perfil_101::initAttributes()
{
}
```

```cpp
#ifndef PERFIL_101_H
#define PERFIL_101_H

#include string
#include vector



namespace TP1Act4 {


/**
 * class Perfil_101
 *
 */

class Perfil_101
{
public:
  // Constructors/Destructors
  //


  /**
   * Empty Constructor
   */
  Perfil_101();

  /**
   * Empty Destructor
   */
  virtual ~Perfil_101();

  // Static Public attributes
  //

  // Public attributes
  //
```

```cpp
#ifndef PERFIL_101_H
namespace TP1Act4 {
class Perfil_101
  // Public attributes


  // Public attribute accessor methods
  //


  // Public attribute accessor methods
  //

protected:
  // Static Protected attributes
  //

  // Protected attributes
  //


  // Protected attribute accessor methods
  //


  // Protected attribute accessor methods
  //

private:
  // Static Private attributes
  //

  // Private attributes
  //

  // En [MPa]
  unsigned int Esfuerzo_de_Von_Misses;
  // En [m²]
  unsigned int Área;
  // En [m²]
```



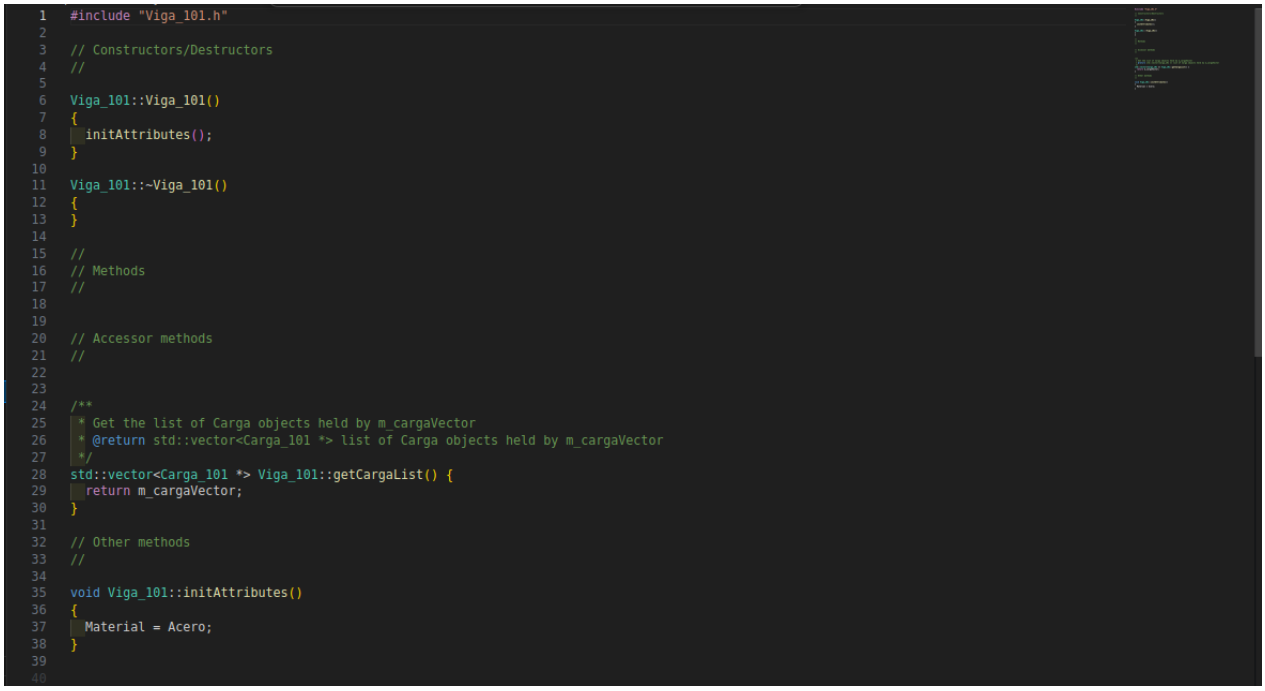
```
#ifndef PERFIL_101_H
namespace TP1Act4 {
class Perfil_101
  unsigned int Área;
  // En [m³]
  unsigned int Momento_Primer_Orden;
  // En [m⁴]
  unsigned int Momento_Segundo_Orden;
  string Id_Estandar_del_Perfil;

  // Private attribute accessor methods
  //


  // Private attribute accessor methods
  //


  /**
   * Set the value of Esfuerzo_de_Von_Misses
   * En [MPa]
   * @param value the new value of Esfuerzo_de_Von_Misses
   */
  void setEsfuerzo_de_Von_Misses(unsigned int value)
  {
    Esfuerzo_de_Von_Misses = value;
  }

  /**
   * Get the value of Esfuerzo_de_Von_Misses
   * En [MPa]
   * @return the value of Esfuerzo_de_Von_Misses
   */
  unsigned int getEsfuerzo_de_Von_Misses()
  {
    return Esfuerzo_de_Von_Misses;
  }

  /**
```

```
#ifndef PERFIL_101_H
namespace TP1Act4 {
class Perfil_101

  /**
   * Set the value of Área
   * En [m²]
   * @param value the new value of Área
   */
  void setÁrea(unsigned int value)
  {
    Área = value;
  }

  /**
   * Get the value of Área
   * En [m²]
   * @return the value of Área
   */
  unsigned int getÁrea()
  {
    return Área;
  }

  /**
   * Set the value of Momento_Primer_Orden
   * En [m³]
   * @param value the new value of Momento_Primer_Orden
   */
  void setMomento_Primer_Orden(unsigned int value)
  {
    Momento_Primer_Orden = value;
  }

  /**
   * Get the value of Momento_Primer_Orden
   * En [m³]
   * @return the value of Momento_Primer_Orden
   */
  unsigned int getMomento_Primer_Orden()
```

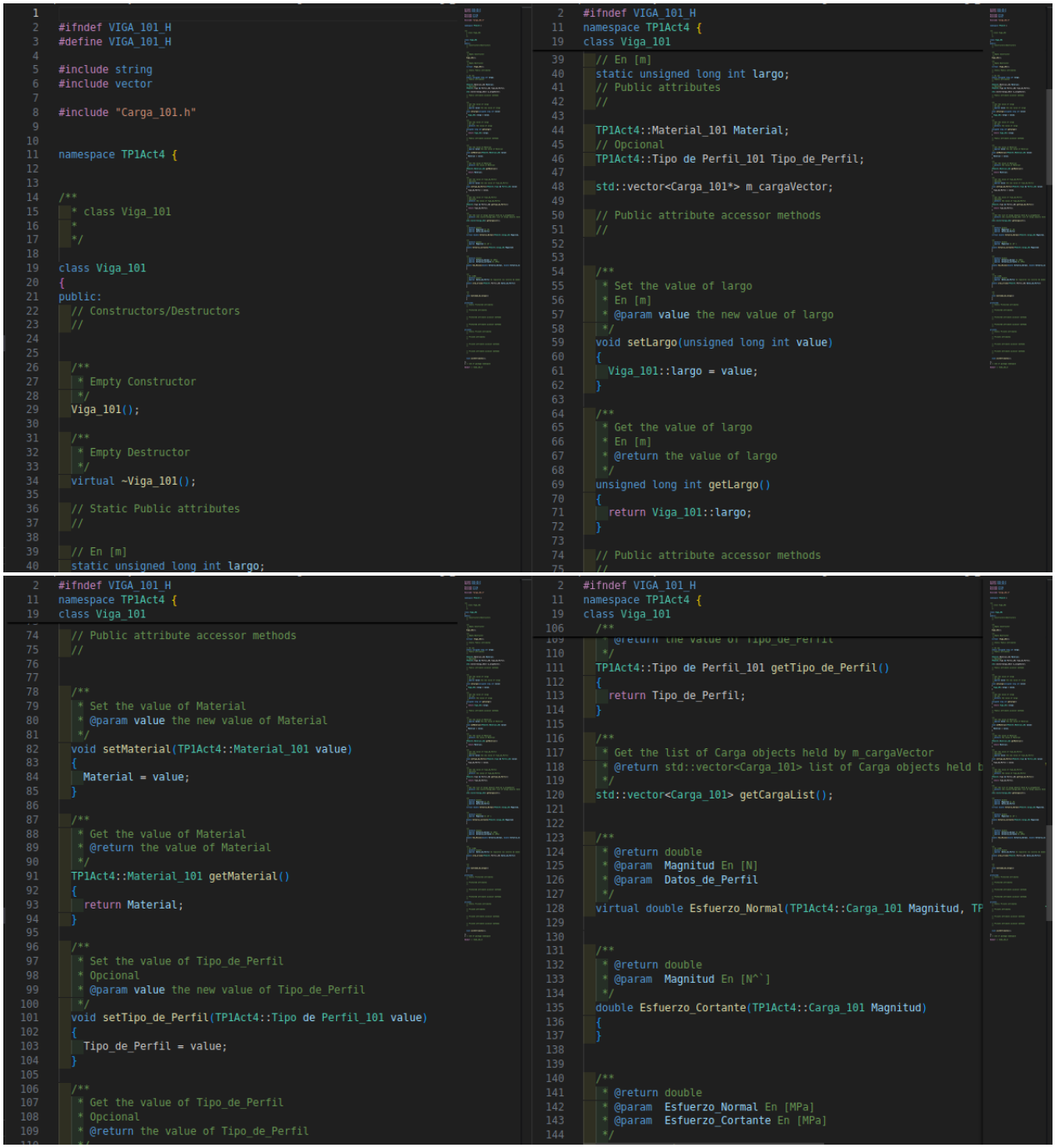
```
#ifndef PERFIL_101_H
namespace TP1Act4 {
class Perfil_101
  /**
  unsigned int getMomento_Primer_Orden()
  {
    return Momento_Primer_Orden;
  }

  /**
   * Set the value of Momento_Segundo_Orden
   * En [m⁴]
   * @param value the new value of Momento_Segundo_Orden
   */
  void setMomento_Segundo_Orden(unsigned int value)
  {
    Momento_Segundo_Orden = value;
  }

  /**
   * Get the value of Momento_Segundo_Orden
   * En [m⁴]
   * @return the value of Momento_Segundo_Orden
   */
  unsigned int getMomento_Segundo_Orden()
  {
    return Momento_Segundo_Orden;
  }

  /**
   * Set the value of Id_Estandar_del_Perfil
   * @param value the new value of Id_Estandar_del_Perfil
   */
  void setId_Estandar_del_Perfil(string value)
  {
    Id_Estandar_del_Perfil = value;
  }

  /**
   * Get the value of Id_Estandar_del_Perfil
```

```
#ifndef PERFIL_101_H
namespace TP1Act4 {
class Perfil_101

  /**
   * Set the value of Id_Estandar_del_Perfil
   * @param value the new value of Id_Estandar_del_Perfil
   */
  void setId_Estandar_del_Perfil(string value)
  {
    Id_Estandar_del_Perfil = value;
  }

  /**
   * Get the value of Id_Estandar_del_Perfil
   * @return the value of Id_Estandar_del_Perfil
   */
  string getId_Estandar_del_Perfil()
  {
    return Id_Estandar_del_Perfil;
  }

  void initAttributes();

};
} // end of package namespace

#endif // PERFIL_101_H
```

Viga



```cpp
#include "Viga_101.h"

// Constructors/Destructors
//

Viga_101::Viga_101()
{
  initAttributes();
}

Viga_101::~Viga_101()
{
}

//
// Methods
//


// Accessor methods
//


/**
 * Get the list of Carga objects held by m_cargaVector
 * @return std::vector<Carga_101 *> list of Carga objects held by m_cargaVector
 */
std::vector<Carga_101 *> Viga_101::getCargaList() {
  return m_cargaVector;
}

// Other methods
//

void Viga_101::initAttributes()
{
  Material = Acero;
}

```



```
#ifndef VIGA_101_H
#define VIGA_101_H

#include string
#include vector

#include "Carga_101.h"


namespace TP1Act4 {


/**
 * class Viga_101
 *
 */

class Viga_101
{
public:
  // Constructors/Destructors
  //


  /**
   * Empty Constructor
   */
  Viga_101();

  /**
   * Empty Destructor
   */
  virtual ~Viga_101();

  // Static Public attributes
  //

  // En [m]
  static unsigned long int largo;
```

```
#ifndef VIGA_101_H
namespace TP1Act4 {
class Viga_101
  // En [m]
  static unsigned long int largo;
  // Public attributes
  //

  TP1Act4::Material_101 Material;
  // Opcional
  TP1Act4::Tipo de Perfil_101 Tipo_de_Perfil;

  std::vector<Carga_101*> m_cargaVector;

  // Public attribute accessor methods
  //


  /**
   * Set the value of largo
   * En [m]
   * @param value the new value of largo
   */
  void setLargo(unsigned long int value)
  {
    Viga_101::largo = value;
  }

  /**
   * Get the value of largo
   * En [m]
   * @return the value of largo
   */
  unsigned long int getLargo()
  {
    return Viga_101::largo;
  }

  // Public attribute accessor methods
```

```
#ifndef VIGA_101_H
namespace TP1Act4 {
class Viga_101
  // Public attribute accessor methods
  //


  /**
   * Set the value of Material
   * @param value the new value of Material
   */
  void setMaterial(TP1Act4::Material_101 value)
  {
    Material = value;
  }

  /**
   * Get the value of Material
   * @return the value of Material
   */
  TP1Act4::Material_101 getMaterial()
  {
    return Material;
  }

  /**
   * Set the value of Tipo_de_Perfil
   * Opcional
   * @param value the new value of Tipo_de_Perfil
   */
  void setTipo_de_Perfil(TP1Act4::Tipo de Perfil_101 value)
  {
    Tipo_de_Perfil = value;
  }

  /**
   * Get the value of Tipo_de_Perfil
   * Opcional
   * @return the value of Tipo_de_Perfil
```

```
#ifndef VIGA_101_H
namespace TP1Act4 {
class Viga_101
  /**
   * @return the value of Tipo_de_Perfil
   */
  TP1Act4::Tipo de Perfil_101 getTipo_de_Perfil()
  {
    return Tipo_de_Perfil;
  }

  /**
   * Get the list of Carga objects held by m_cargaVector
   * @return std::vector<Carga_101> list of Carga objects held b
   */
  std::vector<Carga_101> getCargaList();


  /**
   * @return double
   * @param  Magnitud En [N]
   * @param  Datos_de_Perfil
   */
  virtual double Esfuerzo_Normal(TP1Act4::Carga_101 Magnitud, TP


  /**
   * @return double
   * @param  Magnitud En [N^`]
   */
  double Esfuerzo_Cortante(TP1Act4::Carga_101 Magnitud)
  {
  }


  /**
   * @return double
   * @param  Esfuerzo_Normal En [MPa]
   * @param  Esfuerzo_Cortante En [MPa]
   */
```