```
┌─────────────────────────────┐
│      std::enable_shared       │
│      _from_this< GUII >        │
├─────────────────────────────┤
│                               │
├─────────────────────────────┤
│                               │
└─────────────────────────────┘
                △
                │
┌─────────────────────────────┐
│             GUII              │
├─────────────────────────────┤
│                               │
├─────────────────────────────┤
│ + virtual std::unique         │
│   _ptr< CommandI > Update      │
│   (std::shared_ptr< CommandFactory │
│   I > command_factory)=0       │
│ + virtual bool ShouldClose()=0 │
│ + virtual void OpenChoiceDialog │
│   (DialogMessage dialogMessage, │
│    std::vector< std::pair< std  │
│   ::string, bool > > &choices)=0│
│ + virtual void SetPreventInput  │
│   Flag(bool set)=0             │
│ + virtual std::unique          │
│   _lock< std::mutex >           │
│   LockMutex()=0                │
│ # virtual void Initialize       │
│   (const std::string &windowName, │
│    std::shared_ptr< SDFormatParserI │
│    > sdformat_parser, bool &success)=0 │
└─────────────────────────────┘
                △
                │
┌─────────────────────────────┐
│             GUI               │
├─────────────────────────────┤
│ - std::shared_ptr< Command     │
│   Factory > command_factory    │
│ - std::atomic< bool >          │
│   prevent_input_flag           │
│ - bool model_viewer_running    │
│ - bool render_collisions       │
│   _in_model_viewer             │
│ - GLFWwindow * window          │
│ - ImGuiIO * io                 │
│ - std::shared_ptr< SDFormat    │
│   ParserI > sdformat_parser    │
│ - sdf::ElementPtr element       │
│   _to_append_to                │
│ - ImVec4 background_colour     │
│ - std::mutex gui_mutex         │
│ - bool use_dropdown_for        │
│   _editing_attribute           │
│ - sdf::ParamPtr attribute       │
│   _to_edit                     │
│ - sdf::ElementPtr element       │
│   _to_edit                     │
│ - bool use_dropdown_for        │
│   _editing_element             │
├─────────────────────────────┤
│ + GUI(const std::string        │
│   &window_name, std::shared    │
│   _ptr< SDFormatParserI > sdformat │
│   _parser, bool &success)      │
│ + ~GUI()                       │
│ - void Initialize(const        │
│   std::string &window          │
│   _name, std::shared_ptr        │
│   < SDFormatParserI > sdformat │
│   _parser, bool &success) override │
│ - bool ShouldClose()           │
│   override                     │
│ - std::unique_ptr< Command     │
│   I > Update(std::shared        │
│   _ptr< CommandFactoryI >       │
│   command_factory) override    │
│ - void SetPreventInputFlag      │
│   (bool set) override          │
│ - void OpenChoiceDialog         │
│   (DialogMessage dialogMessage, │
│    std::vector< std::pair< std  │
│   ::string, bool > > &choices)  │
│   override                     │
│ - void DisplaySDFRootElement    │
│   (std::unique_ptr< CommandI    │
│    > &command, std::shared_ptr  │
│   < SDFormatParserI > sdformat │
│   _parser, std::shared_ptr< CommandFactory │
│   I > command_factory)         │
│ - bool SetupNewFrame()          │
│ - void DrawCoreFrame            │
│   (std::unique_ptr< CommandI    │
│    > &command, std::shared      │
│   _ptr< CommandFactoryI > command │
│   _factory)                    │
│ - void CreateModifyAttribute    │
│   Dropdown(sdf::ParamPtr        │
│    attribute, std::unique       │
│   _ptr< CommandI > &command,   │
│    std::shared_ptr< CommandFactoryI │
│    > command_factory, int &unique_id) │
│ - void CreateModifyElement      │
│   Dropdown(sdf::ElementPtr      │
│    element, std::unique_ptr     │
│   < CommandI > &command, std    │
│   ::shared_ptr< CommandFactoryI │
│    > command_factory, int &unique_id) │
│ - void CreateAppendElement      │
│   Dropdown(sdf::ElementPtr      │
│    element, std::unique_ptr     │
│   < CommandI > &command, std    │
│   ::shared_ptr< CommandFactoryI │
│    > command_factory, int &unique_id) │
│ - void CreateDropdown           │
│   (const std::vector<           │
│    std::string > &items,        │
│    const std::vector< std       │
│   ::string > &item_descriptions, │
│    int &selected_item, int &unique_id) │
│ - std::unique_lock< std         │
│   ::mutex > LockMutex()         │
│   override                     │
│ - static void GLFWErrorCallback │
│   (int error, const char *description) │
└─────────────────────────────┘
```