

# How does the client work?

## 1 General

The client consists of:

`Client.cs`—Startup class

**Control namespace**—Handles user- and server input, and tells the model and view how to update

`Control/Controller.cs`—Manages the application

`Control/NetConnector.cs`—Interface between controller and server

`Control/InputHandler.cs`—Handles user and server input

**Model namespace**—Status of the world, as it's known to the player

`Model/Creature.cs`—location and representation of a creature

`Model/Field.cs`—representation of a tile

`Model/LocalModel.cs`—maintains grid of known tiles and list of creatures

**View namespace**—Displays the world and gets input from the user

`View/Imagebuffer.cs`—maintains a dictionary of images with string keys

`View/TileDisplayForm.cs`—gets player input and displays model

`View/ChatMessages.cs`—form to display chat messages

## 2 Events

Its function is based on three events:

### 1. Program Start

When the program starts the display form and model are created. Everything is made ready for the connection to the server to be made, but this is not done automatically.

### 2. Text Input Event

When text ended with an enter is typed in the textbox on the display form, it is sent to the controller. If the text is “connect”, the connection to the server is made. Otherwise the command is passed on to the `NetConnect` class to be sent to the server.

### 3. Server Input Event

When the server sends updates, they are passed on to the controller. The controller updates the model, then makes the form redraw the world.