

Educational Technology Lab

National and Kapodistrian University of Athens

School of Philosophy

Faculty of Philosophy, Pedagogy and Philosophy (P.P.P.), Department of
Pedagogy

Director: Prof. C. Kynigos

ChoiCo Manual



The screenshot displays the ChoiCo game interface. On the left is a 3D city map with various buildings and landmarks. On the right is a sidebar with two tables and a Game Log icon.

Point Information

Description	Restaurant
Energy	-5
Money	-60
Hygiene	1
Fun	45
Social	10
Health	1

Overall Values

Energy	40
Money	410
Hygiene	6
Fun	95
Social	60
Health	6
Points Visited	2

Game Log

Contents

What is ChoiCo	3
ChoiCo Game Rationale	4
Play a Game	4
Design a Game with ChoiCo	5
Game Interface (1st tab)	5
Change the background	6
Add/Delete/Move a point	6
Data Table	6
Adding / Deleting new attribute (column)	7
Change the type of an attribute	7
Set the attributes' values of a point	8
Initial settings (2nd tab)	8
Gameplay rules (3rd tab)	9
End rules (4th tab)	9
Programming with blocks	10
The programming Environment	10
Block List	11
Category: Initialize	11
Category: Conditions	11
Category: variables	14
Category: Maths	14
Category: Actions	15
Category: Game Flow	16

What is ChoiCo

ChoiCo software was built to support students and teachers to create their own digital games. It's a "generator" for games that belong to the category "choice-driven" games. In these games the player has to choose between some given options and every choice has specific consequences to the game score. ChoiCo differentiates from the existing games in that here the mechanism that determines the choices, their consequences and the rules of the game is not closed and inaccessible to the user, but becomes an object of thought, discussion, negotiation and modification. Thus, the ChoiCo software offers two different modes: The Play Mode, where one can play any predefined game and the Design Mode, where one can design a new game or modify an existing one by using ChoiCo's special tools.

ChoiCo Game Rationale

The games that are designed by ChoiCo are choice-driven games. This means that the flow of the game is based on player's decisions with respect to the available choices. More precisely, every game consists of two basic elements: the "Scene" and the "Game Status" board (Image 1).



Image1: ChoiCo Play Mode

The "Scene" contains all the available choices for the player. Depending on the game scenario, these choices may represent various things, i.e. places to visit on a map, foods to eat, persons to talk or objects to use. The "Game Status" board contains a number of arithmetic attributes, the values of which are changing during the game. The "Game Status" is related to the game's scenario; for example it may contain the attributes "Health", "Money", "Friends", "Time", "Fun" etc. Every choice of the Scene has a list of characteristics (shown in "Point Information" board) which affect some or all the Game Status attributes by adding or subtracting a specific value to them. For example the choice "Go to work" adds 1000 points to "Money" attribute, 1 point to "Health" attribute and subtract 80 points from "Energy" attribute. Thus, when the player selects one option from the scene the Game Status is affected accordingly. The player keeps making choices until he/she wins or loses the game. On every turn the player has to examine the available options and their consequences on the Game Status and then select one of them according to the game's goal. Some examples of game goals are:

- To be healthy: Make your Weight below 100 in 10 turns (The player loses if on his/hers 10th choice the Weight indicator is over 100)
- Saving money: Make 15 choices without spending all your Money (The player loses if the Money attribute becomes 0 before he/she makes all the 15 choices)
- Socializing: Play until you make 10 new friends (The player doesn't lose. However he wins only when he/she will have increased the Friends attribute to 10)

Play a Game

You can play a game by clicking the "Play Game" option at the home screen which will transfer you to a new screen. There you can either open a ChoiCo game that it is saved locally on PC, or select to play one of the available online games. When a game loads, the scene appears on the left side and the Game Status board on the right side. If you click on a point of the scene, the "Point Information" table inform you about the characteristics of this point. That table contains information about the effects this point will have to the game status and probably some extra information i.e. a description. To select a point you have to click on the button "Select Point". After that the Game Status will be updated according to

the point you selected and you will probably receive some feedback from the system. When the game ends a red pop up window will show up.

Other available buttons in play mode are:



Shows/Hide the **game instructions**.



Shows/Hide the **game log**. The game log contains all the actions you have done in the current game (the points you selected and the game status at the end of every turn) Moreover it may contain printed messages that have been programmed through the game rules.



Restarts the game. You can restart a game at any time you wish.



This button transfers you to **Design Mode** where you can edit the game you play.

Design a Game with ChoiCo

To design a new game select the “Design Game” option at the home screen which will transfer you to the Design Mode of ChoiCo environment.

The Design Mode offers four separate tabs representing the different elements of the game that you will design. These tabs are:

a) Game interface

Here you design the interface of your game (how your game will look like). You can change the scene’s background, add and modify points on the scene and set up the attributes of each point.

b) Initial settings

This is where you set up the initial status of the game (how the game will start).

c) Gameplay rules

In this tab you set up the rules that will take place in every turn of the player (how the game will progress).

d) End rules

Here you set up the rules that will end the game (how the player wins or loses).

Below there is a detailed description of each tab and how to use its functionalities


Game Interface (1st tab)

In this tab you can find the scene of the game and a database, as seen in Image 2. The scene contains a) a background (by default a city map) and b) a number of points on it (by default is empty). The scene is a very important game element, as through it the player will actually interact with the game by clicking on the available points.





Image2: ChoiCo Design Mode - "Game Interface"Tab

Change the background

To change the background of the scene, click on the icon  on the toolbar left of the scene. Then you can select any image from your computer and click OK. The image will automatically be added and adjusted as background of the scene.

Add/Delete/Move a point

To **add** new points on the scene, click on the icon  on the toolbar left of the scene. By clicking this icon you activate the "insertion of points" functionality. While this functionality is activated, you can click anywhere on the scene with the cursor to add a point. After you have finished with the points' insertion you have to click the  button again to return to the normal mode of the map.

You can **move** a point by clicking on it and drag it to a new position on the scene (drag-and drop).

To **delete** a point you should right click on it and select "Delete Point"

Data Table

When you add a new point on the scene, the system automatically inserts a row at the table on the right. This **line** represents the specific point (like a record in a database) and gives you the opportunity to change the point's attributes. All points have the same attributes which are depicted by the **columns** of the table (Image 3). By default there are 4 attributes: ID, Description, FieldA, FieldB. The ID is a unique number of each point and cannot be deleted or modified. The Description attribute is a small text for a point that will appear next to it on the scene.


Scene Points

ID	Description	Money	Energy
26	Home	100	200
35	Game Store	-50	-40
38	Pizzeria	-20	100
41	School	0	50
44	Football	0	-100

Attributes

Image 3: Points' Database Table


Adding / Deleting new attribute (column)

To add a new attribute (column) at the table you should click on the icon  of the table. An attribute will be added as a new column of the table. To set the name of the attribute click on the orange box.

ID	Description	Money	Energy	
26	Home	100	200	
35	Game Store	-50	-40	
38	Pizzeria	-20	100	
41	School	0	50	
44	Football	0	-100	

Click here to insert a new attribute

Image 4: Add new attribute

To delete an attribute, click on the little box next to its name. Then click on the  icon on the top of the table. The "Description" and "ID" attributes cannot be deleted.


ID	Description	Money	Energy	Knowledge
26	Home	100	200	50
35	Game Store	-50	-40	0
38	Pizzeria	-20	100	0
41	School	0	50	100
44	Football	0	-100	0

2.Delete the attribute

1.Select the attribute

Image 5: Delete Field

Change the type of an attribute

Every attribute is characterized by its name and its type. Available types are: Number, Text, Image, Date, Url. The default type of every attribute is "Number". To change the type click on the checkbox next to the attribute's name and then click on the icon  on the top of the table. At the pop up window you can select one of the above types.

The “Description” attribute is by default set to “Text” type and cannot be changed.

Set the attributes’ values of a point

You can set the attribute values for a point by clicking to the respective white boxe of the point’s row.



ID	Description	FieldA	FieldB	photo
28	Home	2	-10	 Browse...
32	School	4	40	 Browse...

Image 6: Edit the values of a specific point

Initial settings (2nd tab)

After you have set the attributes and their properties at the second tab, you can then define the initial values of the numerical attributes at the second tab “Initial settings”. There you will find a block-based programming area where you can make your definitions. At the left side there is a sidebar with the available blocks, which you can drag and drop them in the white workspace to the right in order to program the initial values of the game (image 7)



Image 7:Sidebar and workspace of tab 1

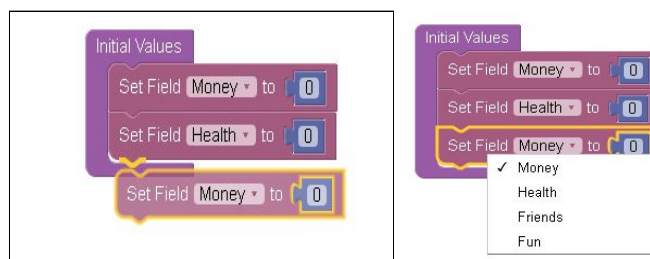


Image 8: a) Drag and drop a block to the workspace. b) Select attribute to define

When you visit this tab for the first time, it has only two attributes set **as an example**. You have to select from the sidebar the block “set Field ... to ...” and drag it under the other “set Field” blocks (Image 8 a). Click on the field name to select from a dropdown menu the field you want to define (Image 8 b). As you can see in figure 2 the default value for all attributes is set to 0. You can change this number by clicking in the blue box that contains the number.

PAY ATTENTION

- All the Set Field blocks must be under the Initial Values purple function as in the example above
- You must set the initial value for every numeric attribute you have in the game.

More information about the available blocks and how to use the programming environment you can find in the section “Programming with blocks”.

Gameplay rules (3rd tab)

At the 3rd tab you can program rules for the game that will occur **every time the player selects a point**. The code that will be programmed here, will run every time the player selects a point and the result will occur automatically. Similar with the 2nd tab, this tab contains a working space for block-based programming and a side bar with the available blocks. The available blocks here are divided in four different categories according to their functionality which are: conditions, variables, maths and actions.

In Figure ... there is an example of 2 rules programmed in tab 3. In this example after the player makes one choice, the game will check if his money are below 50, and if they are a pop up window will appear with the message “Be careful”. After that it will check if his Health is over 5 and if it is the message “Your health is very good right now” will be printed at the game log area.



Image 9: Example of Gameplay rules

More information about the available blocks and how to use the programming environment you can find in the section “Programming with blocks”.

End rules (4th tab)

At the 4th tab you can program the ending rules of the game. The code that will be programmed here, will run every time the player selects a point, after the code of check conditions, and the result will occur automatically. Similar with the tabs 2 and 3, this tab contains a working space for block-based programming and a side bar with the available blocks. The available blocks here are divided in five different categories according to their functionality which are: conditions, variables, maths, actions and game flow.

In Figure ... there is an example of 2 ending rules programmed in 4th tab. Every time the player makes a choice the code checks first if his Money became lower than 0 and If they are a pop up window appears with the message “You spent all your money” and the game finish. Secondly it checks if the player’s Friends became bellow 5 AND the Fun became bellow 10 and if they are the game finish.

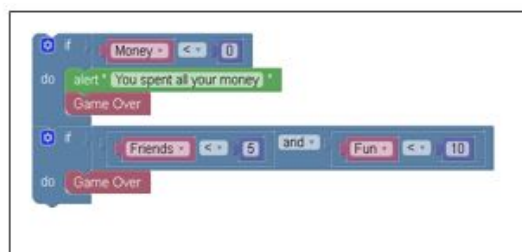


Image 10: Example of end rules

Pay attention

It is necessary to use the "Game Over" block to finish the game.

More information about the available blocks and how to use the programming environment you can find in the section "Programming with blocks".

Programming with blocks

In this section we describe the programming environment of ChoiCo and we provide an extent list with all the available blocks.

The programming Environment

The programming environment consists of two parts: the sidebar on the left and the workspace on the right. The sidebar contains all the available blocks that can be used at the workspace. While ChoiCo has 3 different workspaces (initial settings, gameplay rules, end rules), there are some differences to the available blocks according to the functionality of each workspace.

Insert new block

To insert a block in the workspace and make it part of the code you select it from the sidebar and drag and drop it in the workspace. Different blocks can be connect together if they have edges that fit with each other, like pieces of a puzzle. Figure ... shows an example of two blocks that can be connected and two that cannot.

Delete block(s)

To delete a block or a group of connected blocks you can do one of the following:

- a) Right click on it and select "Delete Block"
- b) Drag and drop it on the bin at the right down corner
- c) Drag and drop it on the side bar

Duplicate block(s)

To duplicate a block or a group of blocks you should right click on it and select "Duplicate"

Disable block(s)

To disable a block or a group of connected blocks you should right click on it and select "Disable Block". The block(s) will appear as disabled in the workspace and will not be executed. You can enable it again by right clicking and selecting "Enable Block".

Collapse block(s)

To collapse a block or a group of connected blocks you should right click on it and select "Collapse Block". The block(s) will be collapsed and they will be represented by a single block. You can expand the block(s) again by right clicking on the single block and select "Expand Block". This is a very useful functionality if you want to save some space on your workspace.

Undo

You can undo your latest action by pressing ctrl+z on your keyboard.

Block List

Bellow there is a list with all the available blocks in ChoiCo and their description

Category: Initialize



Block:

Description: Sets the initial value of a Field (FieldA) to a number (0)

Modifications: Click on the name of the field to appear a dropdown menu with the available fields to select from. Click on the number box to change the number

Available at workspace: 1



Block:

Description: All the initial settings of the game **must** be placed under this block

Modifications: -

Available at workspace: 1

Category: Conditions



Block:

Description: A conditional statement block. It needs a statement to be connected at the top and some blocks to be included inside. If the statement is true, then the blocks it includes will be executed

Modifications: -

Available at workspace: 2,3

Examples:

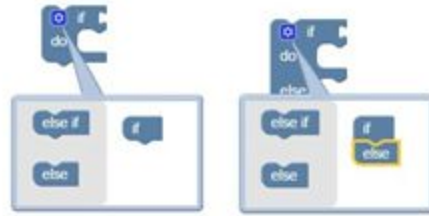


If FieldA is smaller than 50 then alert "Be careful!!"



Block:

Attention ! To use this block you have to click on the settings icon of an if block and create the if else block like below!



Description: A conditional statement block. It needs a statement to be connected at the top and some blocks to be included inside. If the statement is true, then the first group of blocks will be executed else the second group of blocks will be executed.

Available at workspace: 2,3

Example:

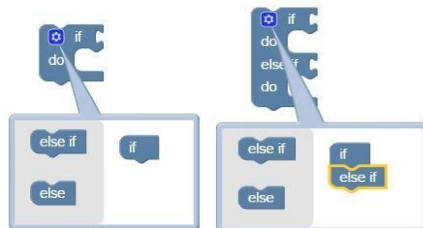


If FieldA is smaller than 50 then alert "Be careful!!" else alert "Very Good!"

Block:



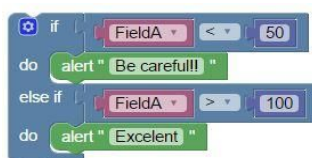
Attention ! To create this block you have to click on the settings icon of an if block and create the if else block like below!



Description: A conditional statement block. It needs a statement to be connected at the top and two group of blocks to be included inside. If the statement is true, then the first group of blocks will be executed, else if the second statement is true then the second group of blocks will be executed.

Available at workspace: 2,3

Example:



If FieldA is smaller than 50 then alert "Be careful!!" else if FieldA is greater than 100 alert "Very Good!"

Block:



This is an input block This block must be connected as input to other blocks in order to work

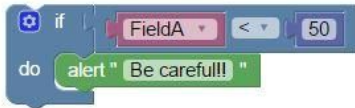
Description: A mathematical condition block that can be used as input for the if blocks. The block describes a mathematical condition (inequality) and it returns true if the condition is true or false if it is false. It is a complex block that consists of three parts: the pink block which is the first part of the condition (FieldA), the operator (=,<,> etc) and the second part of the condition which is by default a number. The 2 parts of the condition can be removed and be replaced by other blocks.

Modifications: Click on the name of the field to select from a dropdown menu one of the available attributes. Click on the operator to select one of the available operators. Click on the number box to change the value of the number. Drag the Field block or the number block and drop it out of box in order to remove it from the condition

Available at workspace: 2,3

Attention: This block represents a mathematical condition not a setter. It DOES NOT set the value of FieldA to 0 !

Example:



The block is used as input to the if conditional block. It will return true when FieldA is smaller than 50 and false when FieldA is larger or equal to 50.

Block:



This is an input block This block must be connected as input to other blocks in order to work

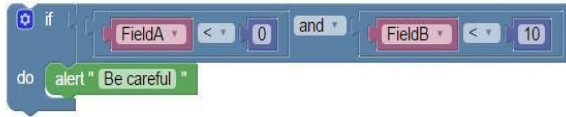
Description: A logical operation block that can be used as input for the if blocks. The block contains 2 mathematical condition blocks and a logical operator (and, or). The **and** block will return true only if **both** of its two inputs are also true. The **or** block will return true if **either** of its two inputs are true.

Modifications: Click on the name of the field to select from a dropdown menu one of the available attributes. Click on the operator to select one of the available operators. Click on the number box to change the value of the number. Drag the mathematical condition block and drop it out of box in order to remove it from the logical operation block

Available at workspace: 2,3

Attention: This block represents a logical operation not a setter. It DOES NOT set the value of any Field to a number!

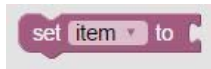
Example:



The block is used as input to the if conditional block. It will return true when FieldA is smaller than 0 **and** FieldB is smaller than 10.

Category: variables

Block:



Description: A setter block. This block sets the value of a variable to a specific value given as input.

Modifications: Click on the name of the variable to select a variable from the list of the available variables or to create a new one.

Available at workspace: 2,3

Attention: This block needs to be connected with an input block in order to work correctly.

Examples: a) b)



a) Sets the value of variable a to 0. (A number block is used as input) b) Sets the value of variable a to "George" (A text block is used as input)

Block:



This is an input block. This block must be connected as input to other blocks in order to work.

Description: A getter block. This block returns the current value of a game's attribute.

Modifications: Click on the name of the attribute to select an attribute from the list of the game attributes.

Available at workspace: 2,3

Example: Look the conditional blocks

Category: Maths

Block:



This is an input block. This block must be connected as input to other blocks in order to work.

Description: A number block. A block with a number in it.

Modifications: Click inside the box to change the value of the number

Available at workspace: 2,3



This is an input block. This block must be connected as input to other blocks in order to work.

Description: A mathematical operation block. This block makes an operation between two parts and returns the result as output. The available operators are + (sum), - (subtraction), ÷ (division), x (multiplication), ^ (power).

Modifications: Click on the operator symbol to select one of the available operators. Drag and drop the desired blocks in the empty spaces to create the operation (usually number blocks or attributes)

Available at workspace: 2,3

Examples:



a) This block **returns the result** of the division $FieldA/2$ b) This block is used as input to the first part of a conditional statement.: If the result of the division is greater to 100 then the statement is true.

Category: Actions



Description: This block prints its input to the log window of the game. It can print any input block in the form of plain text (the value of a variable, the result of a mathematical operation or a text)

Modifications: -

Available at workspace: 2,3



a) Prints the **text** "Very good" b) Prints the current **value of variable** Money c) Prints **the result** of the operation "Money + 5"



This is an input block. This block must be connected as input to other blocks in order to work.

Description: This block returns the text it has written in it. You can type a text in the green box

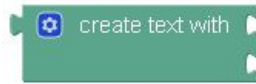
Modifications: -

Available at workspace: 2,3



a) Returns the text "Very good" to the print block

Block:



This is an input block. This block must be connected as input to other blocks in order to work.

Description: This block merges two different inputs into one text output.

Modifications: -

Available at workspace: 2,3

Examples:



Returns a text that contains the text "Your Health is" followed by the current value of the variable Money. For example if the current value of the variable Money is 100 the above block will return "Your Money is 100" and then the print block will print this text to the game log

Block:

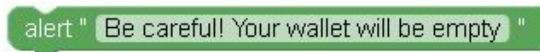


Description: This block shows a pop up window that contains a text message.

Modifications: -

Available at workspace: 2,3

Examples:



Appears a pop up box with the message "Be careful! Your wallet will be empty"

Category: Game Flow

Block:



Description: This block end the game.

Modifications: -

Available at workspace: 3

Examples:

Appears a pop up box with the message "Be careful! Your wallet will be empty"