

Design your Game

Step-by-step guide and Tips

1. First, Let's play

- Open the “Covid-19” game following this url: <http://etl.ppp.uoa.gr/choico/?covidexample>
- Play the game! Your aim is to remain for as long as possible *i.e.* make the highest number in the “Number of choices” field.
- If you lose you can **replay** by clicking the  button on the top right of the screen
- Think why you lost each time.
- What was your highest **score**?

2. Modify it!



Press the button  on the top right to open the game in “**Design Mode**”

2.1 “User Interface” Tab

A. Change some consequences

- Find the choice “**Play football**” in the database (If you click the choice pinpoint and it will highlight the relevant database row)
- Change the consequences it has to the Covid Risk (make it less risky) and to the Social field
- Click the  button to see and try your changes in “Play Mode”

B. Add new choices



- Press the button  on the top right to open the game in “**Design Mode**”



- To add your new choice click the  button on the left of the game scene .
- Then click anywhere on the scene to add the new choice (pinpoint).



- Click again the  button on the left of the game scene to deactivate this function
- When you added the new choice a new line was automatically created in the database right to the scene (you may need to scroll down the database to find it)
- Give a name to your new point in the first empty cell of this row (Description column). The name will appear on the scene next to the pinpoint.
- Set the consequences values of this point in the other empty cells of that row.
- Repeat the same steps to add a second choice in **the “Bedroom” Layer**
- Click the  button to see and try your changes in “Play Mode”



You can move any pinpoint by clicking on it and drag it to a new position (drag and drop)



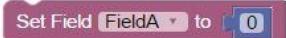
B. Add a new game field, named “Fun”

- Press the button  on the top right to open the game in “**Design Mode**”
- Click the  button on the top right of the datatable. A new column will be added to the table. This is a new game field
- Change the name of the field to “**Fun**” by clicking on its name on the top of the column
- Add the values you like in each of the cells. Each cell represents the consequence of that choice (row) to that field. When the player selects this choice the consequence value will be added to the field value. E.g. if the cell has the value 20, the field value will be increased by 20

Set the initial value of the new field



All game fields must have an initial value when the game starts!

- Open the tab “**Initial Setting**”
 - Here you program what will happen when the game starts e.g. set the initial field values
 - Click on the option “Initialize” on the menu right of the screen
-  Set Field FieldA to 0
- Drag the block  and drop it in the coding workspace, so that it is connected with the other similar blocks (it will make a “click” sound when connected)
 - From the **dropdown menu** of the block select the name of the new Field (**FUN**)
 - Set the initial value of **FUN** field in the blue puzzle

Click the  button to test the changes you made in “Play mode”

C. Program the ending rules

Press the button  on the top right to open the game in “**Design Mode**”



In this coding workspace you can program conditions for ending the game

- Open the tab “**Ending Rules**”
- See how the existing ending rules of the game have been programmed
- Change the first rule so that the game ends when **Physical is < 8**
- Add the following rule for the new game field **FUN** :
 - «If FUN is equal or lower than 0:
 - Play the sound ‘Game Over’. To play a sound use the block “**Play Sound**” that is under “**Actions**” option on the right menu
 - Show the message “You don’t have much fun”
 - End the game. To end the game use with the **gameover** block you will find under “**Game Flow**” option on the right menu

D. Program the gameplay rules

Press the button  on the top right to open the game in “**Design Mode**”

 **In this coding workspace you can program what will happen when the player selects a choice in the game e.g. show a warning message, change the background, play a sound etc**

- Open the tab “**Gameplay Rules**”
- See how the existing ending rules of the game have been programmed
- Change the **warning message** that appears if Covid_Risk is ≥ 75
- Change the second rule so that the warning message about running out of Money appears earlier (e.g. if Money ≤ 20)
- Add the following rules for the new field “**FUN**”:
 - «If Fun is below 20 then show a warning message»
 - «If Fun is greater than 80 then a sound will be played»

Add some extra, more complex rules

- If the player selects “Birthday Party” **AND** Covid_Risk is over 50 show a warning message “Do you think is a good idea to go to a birthday party?”
- If Physical is < 30 **AND** the player selects “Sleep” show the message “You should try to work out even if you stay at home”

Click the  button to test the changes you made in “Play mode”



3. Design a new game

Keep experimenting for design a simple ChoiCo game.

Some Tips for Good Game Design in ChoiCo

- 💡 The point of the game is for the "Game Player" to stay on the game as long as possible, the player staying on the longest is the 'best' player
- 💡 Every choice has consequences regarding each of a number of pre-set fields (e.g., the choice "cotton" as a raw material has consequences in the fields "global warming" and "water scarcity")
- 💡 There is no choice which has only positive or only negative values
- 💡 Each game has one or more 'red lines' respectively, regarding a field above or below which ends the game
- 💡 Think of the set of affordances intended by you, i.e., the 'Game Designer/Creator'. What are the intentions towards:
 - **the essence of game play** (i.e., what skills, knowledge, techniques, strategies, practices) you are intending for the player to adopt - improve - generate
 - **the essence of game modification or re-design or modding** (i.e., what knowledge, techniques, strategies, practices) you are intending for the 'Game Player' to adopt - improve - generate)





Keep your notes

