

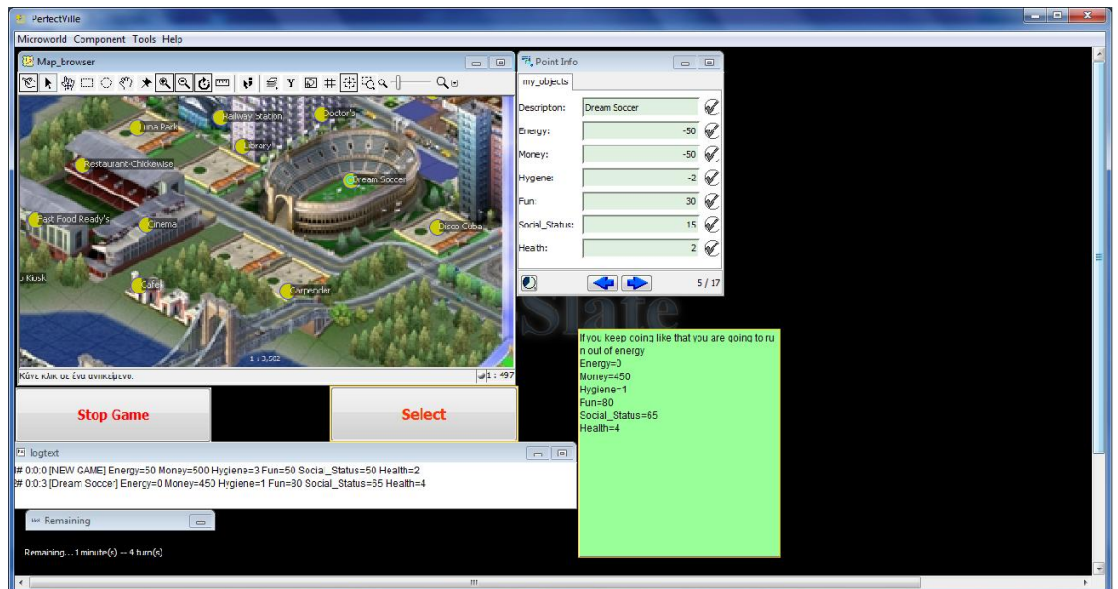


Educational Technology Lab

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



Sus-X

Game construction manual

Ver. 2.2

Table of contents

Introduction	3
About Sus-X	Error! Bookmark not defined.
The purpose of the manual	3
Sus-X can be used in two ways.....	3
Game design	4
Insert new map.....	4
map settings	5
Setting initial game properties	Error! Bookmark not defined.
Establishing check conditions.....	7
Establishing ending criteria.....	9
Setting fame restrictions	Error! Bookmark not defined.
Creating a game.....	Error! Bookmark not defined.
The game.....	Error! Bookmark not defined.
The aim of the game	Error! Bookmark not defined.
Starting the game	11
Game in progress.....	Error! Bookmark not defined.
Strategies	Error! Bookmark not defined.
Ending the game	Error! Bookmark not defined.
Extensions-adds	Error! Bookmark not defined.

Introduction

About Sus-X

Sus-X software was built to support students and teachers to create their own digital games. It's a specific category game "generator", which is related to the resources management of a system (along the lines of simcity) the objective of which is to manage the system without disturbing its balance. The difference from the existing games (like simcity) is that here the mechanism that determines the balance of the system is not closed and inaccessible to the user, but becomes an object of thought, discussion, negotiation and modification.

Sus-X is constructed by the authoring system "E-slate" and leverages prefabricated functional "entities" which called "components". Sus-X needs E-Slate platform to run, so E-slate platform needs to be installed.

The purpose of the manual

The purpose of this manual is to provide the necessary information to the prospective game designer of Sus-X in order to utilize it in the best possible way. The designer may be the teacher and the student himself who designs the games to be played either by one's own group or by other groups of students. Using this manual, does not require specialized knowledge. However, the basic knowledge of "E-Slate" authoring tool is necessary and must be taken for granted.

Although efforts were made so as the manual to become more and more explanatory and complete, however due to the nature of the software and to the flexibility of "E-Slate", all methods of using it (Sus-X) are not possible to be listed.

Sus-X can be used in two ways

Sus-X can be used as an educational activity providing the user (teacher or student) with two available roles:

- A) The role of the designer of a new game
- B) The role of the final player of the constructed game

The student can switch the roles to recognize the rules and the game elements, to modify, improve or alter the objectives, the tools and the

limitations, so as to understand the balance mechanism of a city in various fields.

Making better use of Sus-X as an educational tool, the game can be designed to be completed in 4 different ways:

- 1) Based on time
- 2) Based on the number of the selected items from the city map
- 3) Based on the selected criteria for the players' score
- 4) A combination of the above



Game Design

For designing a new game the following actions are needed to be performed:

- 1) Optionally inserting a new map image
- 2) Determining data- map points and site properties
- 3) Set Check conditions
- 4) Establishing termination criteria- End conditions
- 5) Settings related to the degree of difficulty

Insert new map

To change the background of the map we are doing the following:

- We display (if is not displayed) the map component named «mymap» (select menu Component> Components> mymap)
- At the «mymap» component we select [File> Edit] and then we return to «myarea» tab.
- We press the "Add" button, select the desirable image and then we give a name to it.
- On the "background images" list, the new image will now appear. For activating it we choose 'set as default'
- We close the map creator either by selecting the "Check the integrity of the map" button  or "X" button .

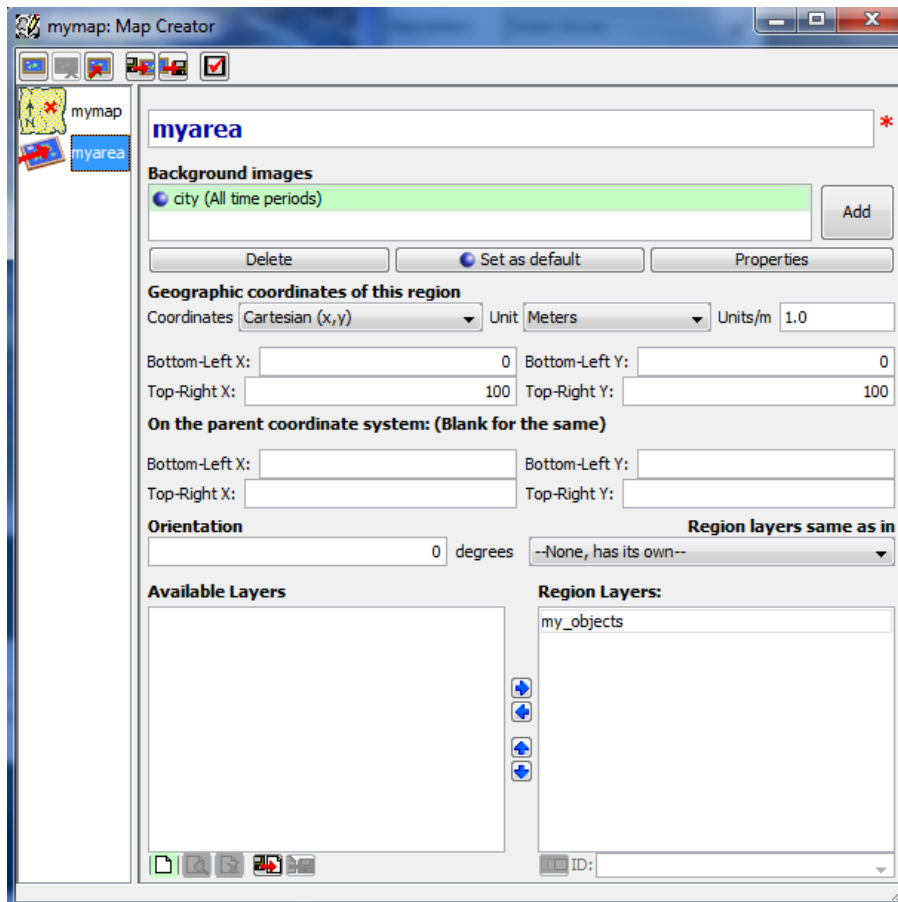



Figure 1: insert map background

As background, the map can accept gif or jpg images (image resolution 620 X 380 or higher). Files larger than 620X380 may not be displayed as a whole or the components' appearance "map_browser" and may need to be resized so as to display them properly.

Map Settings

The design of the game starts with the insertion of the points on the map. These points will be available for selection while playing the game.

- We select the button with the pin ("Insert Object" ) from the toolbar of the map and then we click and highlight the points on the map. If a question about the map layers is appeared we simply confirm the points' entry at "my_objects" layer.
- Each point added to the map automatically creates a new record (row) on the data base of the map (map_data). For every new record, we complete the Description column which corresponds to the desired point name.
- After adding the points, we add fields (columns) in "my_objects" table beside the completed *Description* column. The fields are the "properties" that all the map points will have. For example, point properties can be: energy, money, entertainment, health, etc. For adding them, we can either right clicking on the table and choosing "Add Field" or adding them from the menu [Field> New field]. We give a name to the new field and then we select "Accept".

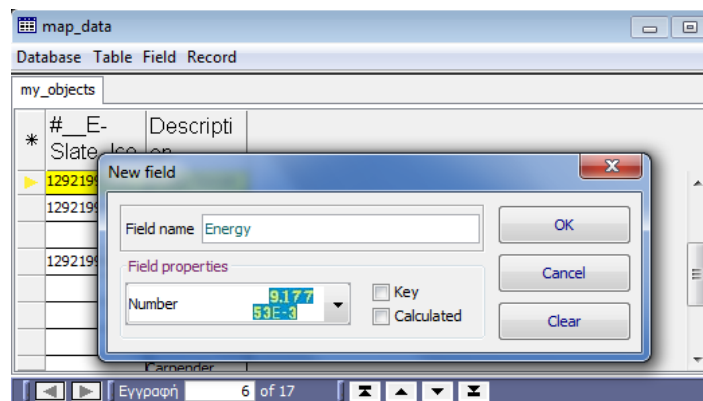


Figure 2 – Adding a new field

- Finally, we complete all the columns -fields which created. We give positive or negative numerical values while maintaining a "scale" of our choice. For example we consider from the outset that all properties will have values from -10 to 10 in steps of 1 or from -1000 to 1000 in steps of 100. If the value is negative while playing the game, subtracts from the units of the user, while if the value is positive, adds to the units of the player. So for a point named "Restaurant" the property of "Energy" may have value 10 but the property of 'Money' may have value -5.
- To delete a point from the map:
 - We select the "Determining Object" tool from the toolbar of the map.
 - We approach the mouse pointer (pin) to the point we wish to delete. A "trash can" appears on the top left on the map.
 - We select and drag the point to the trash can (drag and drop) where we leave it.

Remarks:

- The number of fields can influence the degree of difficulty of the game (more fields → more parameters to be taken into consideration by the player).
- There is no limit to the number of points that will be listed on the map.
- The field names can not contain spaces. Instead we can use the blank character '_'

Setting initial game properties

The initial properties are the initial "capital" with which the game starts. The properties of each point of the map chosen by the player are added or subtracted in this "capital".

- The first column of the table «table1» automatically displays all registered fields – and the properties, exactly as given on the previous steps on the map data base “map_data”.
- On the second column we insert the initial values for each of the fields. If, we insert for example "Energy" "100" the game will start with this value.

* Field_name	Initial_value
Social life	200
Energy	100
Money	200

Figure 3 – Setting initial game properties

Remarks:

- The initial values are important for the difficulty degree of the game.

Establishing Check conditions

A series of predefined checks can be made during the acceptance- selection of the points in the game. The purpose of these checks is the display of warning - informational messages depending on the game situation. The designer of the game can determine both the check conditions and the message which appears. A check like this, can be described by the following sentence:

"If the energy is less than 5 then display the message "Warning! You run out of energy"

There is the possibility of combining up to 4 different conditions for displaying each message. Thus, the corresponding message will be displayed only when the four different conditions are valid in the same time (logic procedure AND). For example:

"If the energy is less than 2 and health is less than 5 then display the message" You will probably get sick if you keep it up! "

To insert a check condition:

- In «Check_conditions» data base we add a new record-line. We can either right click on the table and select "Add Record" or select from the menu [Record> New record]
- In each of the columns «Condition1», «Condition2», «Condition3», «Condition4», we insert the condition in the form of <Field> <operator> <number>. Valid conditions are:
"Energy> 0", "Energy <100", "Energy = 10"
- In the column («Con_message») we type the message which will appear while checking condition is valid.
- In the last column («Con_enabled») we select whether the condition is active (true) or not (false)- ignored during the check.

* Condition1	Condition 2	condition 3	Condition 4	Con_Message	Con_Enabled
Social life <1				is true that you don't have any friends?	true
▶ Money <20	Social life >=400			are you having fun all the time?	true

Figure 4 - Check conditions

Remarks

- It is not necessary to complete all 4 conditions per line
- There is no limit restriction for the records – check lines. They may typically be as many as the point properties.
- The «Con_enabled» column gives us the opportunity to have registered many checks but also to enable or disable some of them.
- A check condition with an operator "=" in some cases, cannot display the posted message because the declared value may be overtaken by the choices of the player. Eg If the player had "100" "money" and after selecting a new point it became "120" then all the intermediate values cannot be checked with equal ("="). In that case a message is displayed eg when "Money = 110 "

Establishing ending criteria

The ending criteria determine when the game is ending. Ending criteria can be defined for all the properties of map points or only for some of them ignoring the development of the others.

The game ends when one of the listed criteria is verified or when a combination of up to four different criteria - conditions is valid. For the first case it is sufficient that one of the criteria - conditions are listed in separate lines of the database, while, in order to combine some criteria, they must be registered in the same row of the database.

The ending criteria are inserted in the "End_conditions" database in the exact way the check conditions are inserted:

- In the "End_conditions" database we add a new record-line either by right clicking on the table and by selecting "Add Record" or by selecting [Record> New record].
- In each of the columns «End_Condition1», «End_Condition2», «End_Condition3», «End_Condition4», we insert the condition in the form of <Field> <operator> <number>. Valid conditions are: "Energy> 0", "Energy <100 ',' Energy = 10 "
- In the column («End_message») we type the message that will appear when the game is ending due to the corresponding condition.
- In the last column («End_enabled») we select the criterion-condition to be active (true) or inactive (false)

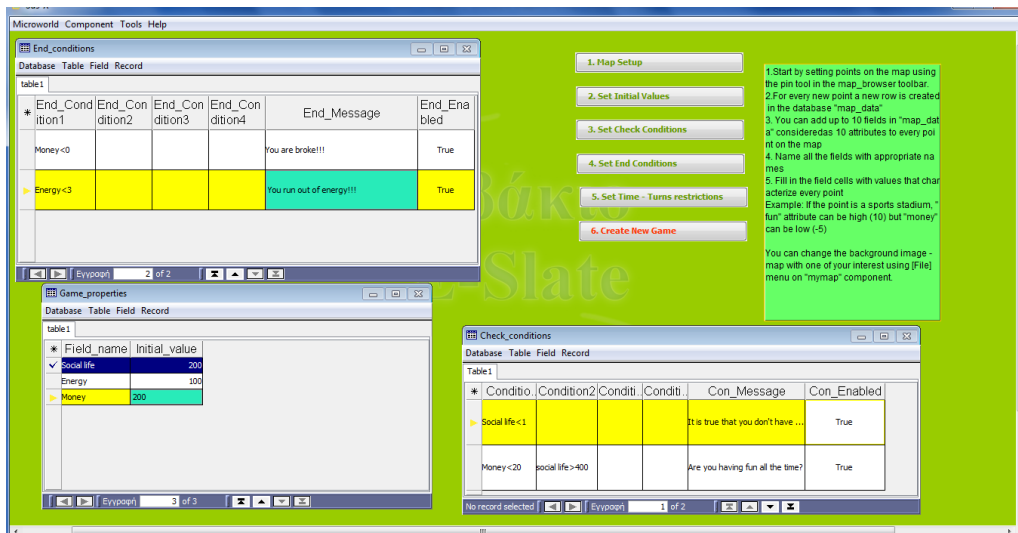


figure 5 – Ending criteria

Remarks

- The “End_enabled” column gives us the opportunity to have registered many criteria but also to enable or disable some of them.
- The check condition with the operator “=” in some cases cannot terminate the game because the declared value may be overtaken by the choices of the player. Eg If the player had “50” “money” and after selecting a new point it became “-50” then all intermediate values cannot be checked with equal (“=”) in order to terminate the game eg. when “Money = 0”

Setting game restrictions

The last step before creating a new game is the regulation of the two game constraints: the duration and the number of the players’ selected points.

Duration

The unit for measuring the duration (time limit) is a minute.

When the selected time is completed, the game ends regardless of any other defined criteria and a corresponding message which gives information about the state of the game appears.

Number of selected points

The game ends, when the number of selected points is reached, regardless of any other ending criteria. When a game is ended a corresponding message which gives information about the state of the game appears.

Remarks

- If the designer wishes a game with no time limits, he/she can set a long period of time (in minutes).
- Similarly, if the designer does not want to set a maximum number of selected points, he/she sets a large number of points.

Creating a game

By clicking on the button «Create New Game» after above settings, the game is ready and available to players. Before the final acceptance and creation of the game a design storage is suggested (menu [File> Save As> Save local microworld]) so as to be available for future changes.

The game

The aim of the game

In the basic Sus-X design, the aim of the game is that players can "survive" in the city for a specific period of playing. Players "wander" in the city and "visit" selected points in order to earn "points" or better in order to maintain a "balanced life." Each point of the city can give or take points from the player in different sectors-properties. Thus e.g. in a "Restaurant" the player loses "money" but gains 'Energy'.

Starting the game

The game starts by pressing the 'Start' button. At the bottom left of the screen the remaining time and the number of remaining points on the map appears.

Game in progress

The players navigate to points on the map by selecting the "Specifying Active Objects" from the toolbar of the map and then click on any point. On the "Point Info" component, they observe the properties of the active site and either accept-select the point by clicking on the "Selecting" button or continue to navigate through the map by clicking on the next point. If the map is big enough, they can move the visible part of the map by clicking on the "Scrolling map" button on the map toolbar.

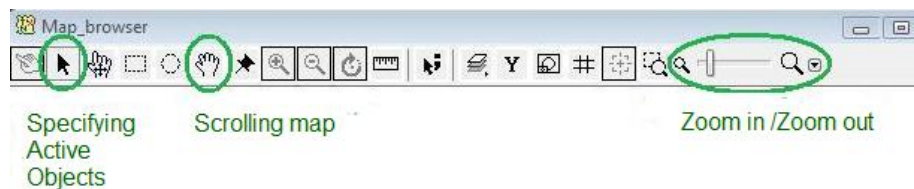


Figure 6 – Map toolbar

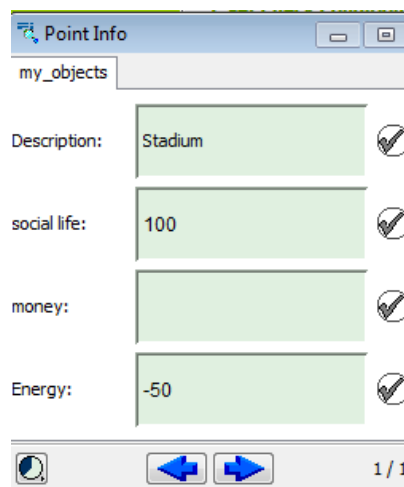


Figure 7 – Information for each point on the map

Strategies

The players may follow different strategies to complete the game, depending on the ending criteria, the duration and the available points on the map.

In every case and depending on the degree of difficulty, the game design encourages and almost “defines” the strategies, whether they come spontaneously while playing, or defined collaboratively at the start of the game.

Examples of the following strategies can be displayed individually or in combination:

- The points initially selected at random and based on warning messages.
- Rough calculations are made based on the point properties. The point is selected only if it serves the purpose.
- Calculations are made for at least 2-3 subsequent points. The points are selected in turn.
- The points are selected based on the interests and the desires of the players.

- Some 'good' points (with respect to the target) are identified and are used constantly.

Ending the game

At the end of the game information is displayed for both the ending reason, and the game situation. This enables players to improve (if necessary) their strategy and designers to renegotiate and modify the parameters of the game.

Besides the default ending mode, the game can be ended (button "Stop") at any time, by the user. In that case information is displayed about the latest state of the game.

Extensions- adds

If the game design and thus the educational scenario requires additional information for each of the city points, fields can be inserted into the map database («map_data») that are not numerical. Eg with additional information about each point.

These fields must begin with the characters "#_" (hash+ underscore) so as not to be considered as computational fields which can influence the development of the game.

When playing the game (if necessary) the image ("frame") and text component can appear separately.

The game "View" needs to be updated in order to display properly the new imported components in the microworld. The "View" is a pre-registered microworld desktop which contains visible and hidden components with selected sizes and locations on the screen. By using the "Views" the same components can be displayed in many different predetermined ways on the screen.

In Sus-X separate "views" for each design stage are created, and also for the final game. Views can be updated by selecting from the menu [Tools]> [View Editor]. There is also the ability to instantly view "a view" from the menu [Microworld]> [Views].