Module « Windows Development »

-

Project « EpiBubble »

Request for proposal

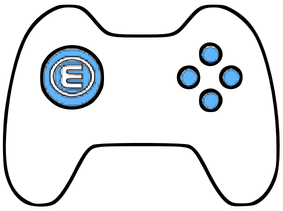


Table of contents

Introduction

The aim of this request of proposal (RFP) is to present the requirements for the EpiBubble project.

1. Company description

EpiLights is a French company which develops small video games, specialist in UWP applications.

Founded in 2015 in Lyon, the city of lights, the company has established itself as close as possible to its main client Ubisoft.

For a new project, EpiLights needs to create a platform composed of a dozen of little games and which will have to be available on Windows devices.

2. Project context

The provider will have to create one game of this platform. The game is a bubble shooter. The goal of the game is to clear the playing field by forming groups of three or more like-coloured bubbles. The game ends when the balls reach the bottom line of the screen. The more balls destroyed in one shot, the more points scored. A player wins when there are no balls remaining on the playing field.

3. Requirements

The software must implement the following requirements and are organized in four categories:

- REQ_FUNC_XXX: functional requirements,
- REQ_IHM_XXX: IHM requirements,
- REQ_DESIGN_XXX: Design requirements,
- REQ_DATA_XXX: Data requirements.

a. Functional requirements

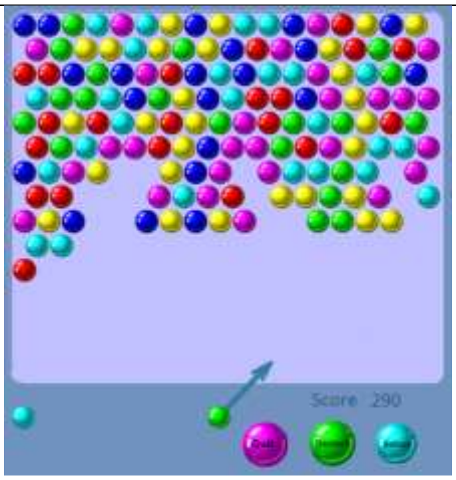
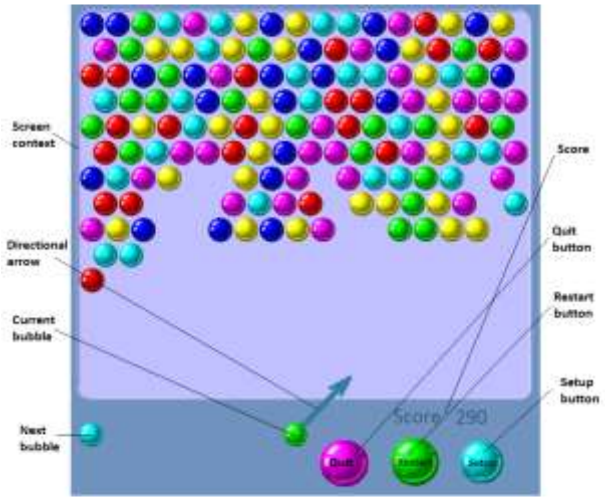
REQ_FUNC_010	There are by default 6 different bubble colours.
REQ_FUNC_020	The screen contains by default 15 rows.
REQ_FUNC_030	Each row can contain by default at most 17 bubbles.
REQ_FUNC_040	Rows are indexed from 0 to (row_count – 1) where row_count is the row count in the game.
REQ_FUNC_050	Row 0 is the row located on the top of the grid.
REQ_FUNC_060	Bubbles on rows R and R+1 are disposed in staggered rows.
REQ_FUNC_070	When user groups three same colour bubbles, or more, those bubbles are destroyed.
REQ_FUNC_075	When launch, the bubble can have attached other's
REQ_FUNC_080	If a bubble is attached to only one group and this group is destroyed, this bubble is destroyed too.
REQ_FUNC_085	When bubble cannot attached other's, the bubble is attached on row 0
REQ_FUNC_090	By default, each 6 shots, a new row is inserted in row 0.
REQ_FUNC_100	When a row is inserted, the row is fully filled by bubbles with random colour.

REQ_FUNC_110	When a row is inserted, if three bubbles or more have the same colour and are grouped, they must not be destroyed until the user shot them.
REQ_FUNC_120	For each bubble destroyed with another group, the score is increased with 15 points.
REQ_FUNC_130	Destroying a group of N same coloured bubbles increases the score of (15.(N-1)) points.
REQ_FUNC_140	The gamer can see what is the colour of its next bubble.
REQ_FUNC_150	An arrow must indicate the user the direction of his shot.
REQ_FUNC_160	Bubble can bounce on left and right sides of the screen context
REQ_FUNC_170	When the application is started, a window is displayed to allow user to start a new game or to continue its previous game if available.
REQ_FUNC_180	If a new game is selected, a level window is displayed to allow user to select the level of the game.
REQ_FUNC_190	Three levels are available for user: easy, classic, hard.
REQ_FUNC_200	On the game screen, three buttons are available: Save, Quit and Setup.
REQ_FUNC_210	When save is selected, a window is displayed to ask user name of this game.
REQ_FUNC_220	If the game is saved by the user, a file save.txt must be stored in a save directory in the game directory installation.
REQ_FUNC_230	When a game is saved, parameters of the current game must be saved.
REQ_FUNC_240	When a new game is selected, default parameters must be restored.
REQ_FUNC_250	When quit is selected, a window is displayed to ask user if he really wants to quit the game.
REQ_FUNC_260	When setup is selected, a window is displayed to modify user parameters.
REQ_FUNC_270	In the setup window, shot number between two row insertions of each levels can be modified.
REQ_FUNC_280	In the setup window, bubble and arrow colours can be modified.
REQ_FUNC_290	In the setup window, speed shot bubble can be modified to low, normal and fast.
REQ_FUNC_300	By default, speed shot bubble is: • 80 pixels/second for low speed • 150 pixels/second for normal speed • 300 pixels/second for fast speed
REQ_FUNC_310	By default, shot number between two row insertions is: • 10 for easy level • 6 for classic level • 3 for hard level
REQ_FUNC_320	The bubble colours must be selected among a selection of those colours: • Black #000000 • Silver #C0C0C0 • Grey #808080 • White #FFFFFF • Maroon #800000 • Red #FF0000 • Purple #800080 • Fuchsia #FF00FF • Green #008000 • Lime #00FF00 • Yellow #FFFF00 • Blue #0000F0 • Cyan #00FFFF

REQ_FUNC_330	The six colours which are selected by default are: • Blue • Red • Cyan • Yellow • Fuchsia • Lime
REQ_FUNC_340	The arrow colour must be selected among a selection of those colours, which can be modified in a configuration file: • Astral #327DA0 • Rose #F64A8A • Olive #808000 • Grey #808080 • Maroon #800000 • Gold #FFD700
REQ_FUNC_350	The arrow colour selected by default is Astral.
REQ_FUNC_360	On PC, the arrow can be managed with the mouse or the arrows on the keyboard.
REQ_FUNC_370	On PC, if the arrow is managed by the mouse, the bubble is shot when the user clicks.
REQ_FUNC_380	On PC, if the arrow is managed by the arrows on the keyboard, the bubble is shot when the user activates the space key on the keyboard.
REQ_FUNC_390	On Xbox, the arrow must be managed with left and right buttons on the joystick.
REQ_FUNC_400	On Xbox, the bubble is shot when user activates the A button (green button) on the joystick.
REQ_FUNC_410	On Xbox, user can access to buttons by activating the Y button (yellow button) on the joystick and navigate between them with left and right buttons.
REQ_FUNC_420	On Xbox, to get back to the game, user must activate the Y button (yellow button).
REQ_FUNC_430	On a touch-screen device, the arrow must be managed by touch: touching the arrow and positioning it to the appropriate position.
REQ_FUNC_440	On a touch-screen device, the bubble is shot when user press a “shot” button.
REQ_FUNC_450	The game can be over when no bubble remains.
REQ_FUNC_460	The game can be over when the bubble is attach over the map.

b. IHM requirements

REQ_IHM_010	Look and feel of the application is the following:

	
REQ_IHM_020	The following screenshot describes the key elements of the game:  Screen context Directional arrow Current bubble Next bubble Score Quit button Restart button Setup button Score 290 Quit Restart Setup
REQ_IHM_030	Bubbles are not selectable.
REQ_IHM_040	Bubbles cannot be moved on the screen.
REQ_IHM_050	The current bubble is the bubble which will be sent to the screen context.
REQ_IHM_060	The next bubble is the bubble which will be sent to the screen context after the current bubble.

c. Design requirements

REQ_DESIGN_010	You must use DirectX library
REQ_DESIGN_020	You must implement MVVM pattern style
REQ_DESIGN_030	You must have two platforms compatibility.

d. Data requirements

REQ_DATA_010	Default count of bubble colour must be modifiable in a configuration file.
REQ_DATA_020	Default row count must be modifiable in a configuration file.
REQ_DATA_030	Default maximum count of bubbles in a row must be modifiable in a configuration file.
REQ_DATA_040	The number of same colour bubbles required to destroy a group must be modifiable in a configuration file.

REQ_DATA_050	The number of shots required to insert a line must be modifiable in a configuration file
REQ_DATA_060	Bubble colours available must be modifiable in a configuration file.
REQ_DATA_070	Arrow colours available must be modifiable in a configuration file.
REQ_DATA_080	Default shot number between two row insertions for easy, classic and hard levels must be modifiable in a configuration file.
REQ_DATA_090	Default speed shot bubble must be modifiable in a configuration file.

4. Delivery conditions

The delivery package must contain the following documents:

- Documentary:
  - SAS (Software Architecture Specifications). This document must follow the template “Template – Software Architecture Specifications.docx” given by the company. It must contain an explanation of the provider understanding of the project, constraints and solutions proposed, global and detailed UML diagrams of the architecture proposition, and the state machine diagrams if needed.
  - SQS (Software Qualification Specifications). This document must follow the template ‘Template – Software Qualification Specifications.docx” given by the company. It must contain tests procedures to check the software. It consists with automatic and manual tests. The document must also contain a traceability matrix which link the requirement id with the test id. The provider should write a first version of this document only from this request for proposal and before any work.
  - SQSA (Software Qualification Specifications Acceptance). This is the SQS document filled with results of the tests procedures.
- Code and binaries:
  - Code. In order to be fully compliant with our existent game platform, the software must be developed in UWP and C# language. Classes and functions must be commented with Doxygen format.
  - Repository name: T-WIN-EpiBubble_$ScolarYear
- Tests:
  - Tests code.

5. Planning

The planning of this project consists of several deadlines, T0 is the beginning of the project:

Object	**Week**
Kick-Off	T0
Bootstrap	T0
Follow up	T0 + 2 weeks
Follow up	T0 + 4 weeks
Delivery of the project	T0 + 4 weeks
Pitch	T0 + 5 weeks

6. Evolutions

This is not mandatory, but the company will appreciate if the provider you go further this project, adding functionalities which make the project more complex.

Example: implementing game bonuses, add difficulties like bubbles which will have to be shot twice to be destroyed, effects on bubble etc…