Mihai Iusan mihai.iusan@lut.fi
Camilo Cifuentes cgcamilo@gmail.com
Alexander Ovcharenko Alexander.Ovcharenko@lut.fi

INTRODUCTION

In this document presents the project created during the WP7 codecamp 2011. We will start showing the idea which pushed us to create the FindIT game application. We will continue by presenting the implemented features. After the features, we will present the problems that we faced during development which will be followed by our solution. The technical description will explain the application architecture using the class diagram. We will also describe the technologies that we used to achieve the application goals. In the final part we finalize with our conclusions.

You can find the project at: http://codecamp.fi/doku.php/wp7_xna2011/lut3/start

IDEA

The idea is to create a game that challenges the user visual ability and his fast reactions, all that together in a fun way.

MOTIVATION

The reasons that made us to create:

- While waiting for something, like been in a queue, people like to spend their time
- People like to compete against others

SOLUTION

A Game:

- Based on time
- With high score system.
- That includes different types of difficulty
- That have media elements such as sounds

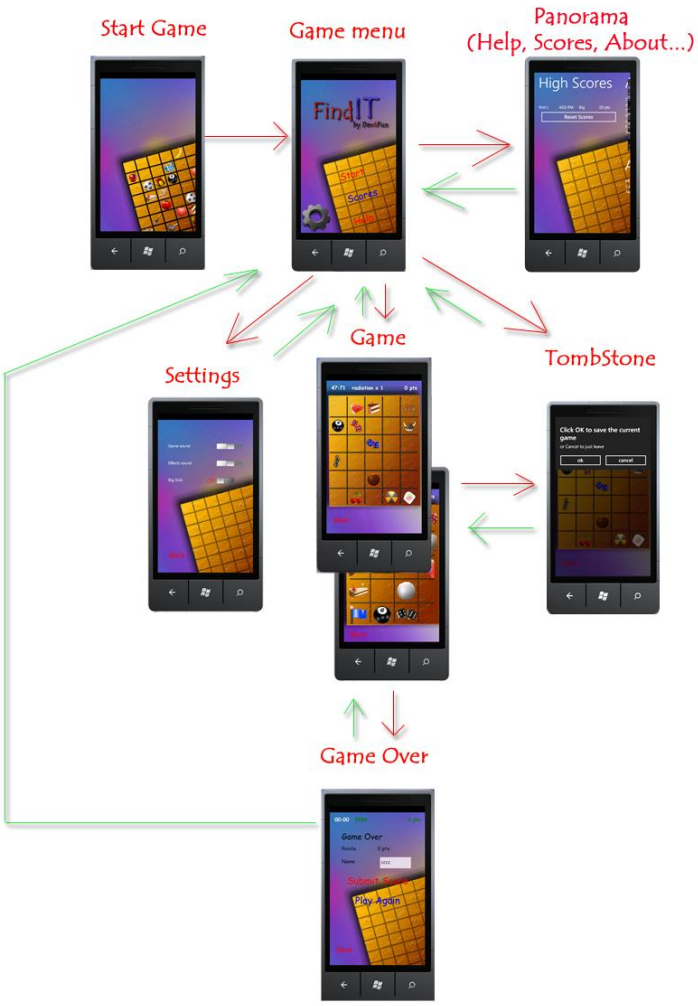
Mihai Iusan mihai.iusan@lut.fi
Camilo Cifuentes cgcamilo@gmail.com
Alexander Ovcharenko Alexander.Ovcharenko@lut.fi

THE GAME: FINDIT

FindIT is a game that shows a random number of icons into a grid. Each icon contains an image. The player has to find the icons that match with a given name in order to get points within a predefined time. If the player makes a mistake 5 seconds will be removed, otherwise 5 points will be added to his score, this means that, the faster the user is matching the images and names the more points he will get. When the time is over the application will show the number of points and the option to submit his score and/or play again.

USER INTERFACE

In games applications the user interface is one of the most important elements, this is why we focus to have an intuitive, minimalist and easy-to-navigate interface.



Mihai Iusan mihai.iusan@lut.fi
Camilo Cifuentes cgcamilo@gmail.com
Alexander Ovcharenko Alexander.Ovcharenko@lut.fi

FEATURES

The following features were implemented:

- Settings for
  - Background sounds
  - Effect sounds
  - Size of grid
- Counter that ends the game with it reaches 0
- Random system that add icons in the game grid
- Random system that rotate the icons
- Panorama pages that
  - Displays the high scores (sorted)
  - Shows Instructions
  - Shows the “About us” section
- Tombstone, application will maintain the current state of the game in case of deactivating the application.1

TECHNICAL DESCRIPTION

For this project we focus on divided the application in “Pages” (view) and “Model”, the pages are basically the User-Controls and different UI elements, in the Model we have classes that handle different aspects of the application.

Basically our application has 2 packages:

- Model package
- Pages package

For the Model It is worth mentioning the following classes:

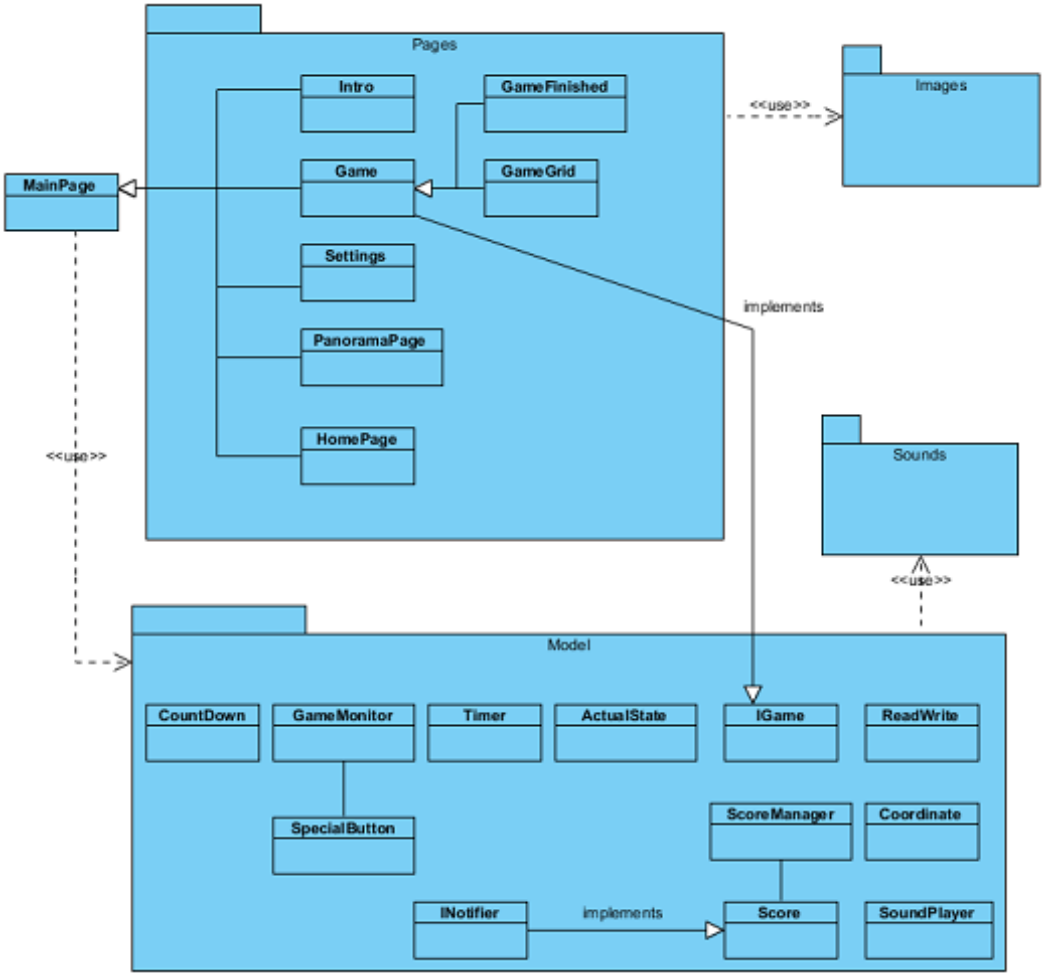
- GameMonitor: is a class that contains all the game logic and it reports all events to the view (Game)
- SoundPlayer: allow us to play background sounds inside a media container.
- SoundEffectAction: Enable to play multiple sound effect using the XNA framework
- Actual state: it maintain the current state (time and points) of the game in progress
- ReadWrite: handles the isolated storage

For the Pages:

- Game: handle all game view of the application, this is the game grid, icons, counter time and also the game finished frame.
- PanoramaPages: it shows the game instructions, high scores and About us section.

Mihai Iusan mihai.iusan@lut.fi
Camilo Cifuentes cgcamilo@gmail.com
Alexander Ovcharenko Alexander.Ovcharenko@lut.fi

We based our development in the following class diagram:



NOTE: for clarity, not all relationships are shown.

Mihai Iusan mihai.iusan@lut.fi
Camilo Cifuentes cgcamilo@gmail.com
Alexander Ovcharenko Alexander.Ovcharenko@lut.fi

INVOLVED TECHNOLOGIES AND TOOLS

For the project we made use of the following technologies and Tools:

- Visual Studio
- Expression Blend
- Silverlight
- XNA

CONCLUSIONS

Games are good candidate applications to motivate developers to explore new technologies.

Silverlight is an easy to use and easy to learn technology. We find it easy to develop the different user interfaces and in the same way to connect all together with other .net technologies.

One of the few problems that we faced was related with the use of threads, even if is easy to define them, it is hard to control and link their functionality with the current process, for example, it is hard to modify interface elements within threads.

We believe that Tombstoning is a good idea to catch events related to the deactivation/activation of the application. It is not easy to implementing but thanks to the multiple examples provided by developer community it is easy to understand it.

Talking about the Emulator, we found it very reliable because when moving the application to the mobile device it our application behaved exactly the same.

Overall our experience in developing the application was very good, without major problems, giving us the opportunity to create a complete application.