UNIVERSITAS
FRIBURGENSIS

UNIVERSITÉ DE FRIBOURG SUISSE
UNIVERSITÄT FREIBURG SCHWEIZ

# Challenger 1

## Multimodal Interfaces - Project

Fitim Ismaili | Ruzhitsa Velkova | Resul Collaku

Supervised by:
Denis Lalanne, Elena Mugellini, Jacques Bapst, Omar Abou Khaled
May 29, 2011

# 1 Introduction

Besides many other important objectives of “Multimodal Interfaces” course, designing and implementing a multimodal game was a very interesting and fun. Multimodal application concerning this scope means the game uses some interactions that can only be achieved using multiple modalities at the same time. After brainstorming for different ideas and searching the internet for a while we decided to develop an arcade game called “Space Invaders”. Furthermore, this project report includes the steps of development and also the important decisions we took during this project.

## 1.1 Game Idea

The initial project idea was to take an old famous game and upgrade it with the latest technology available. We selected a very famous game called “Space Invaders”. The view of the game is saved as the original but the possibilities to play this game include one of the most advanced ones. The idea of the game is to fight against the alien invaders ships. There are 5 rows of invaders moving left and right along the screen. Moreover, the mother ship shows at the top of the screen once in a while.  The movements of the airplane are handled by different modalities such as voice, gesture and touch. Touch modality is tested only with the keyboard.  By shooting the invaders the players gets points. The higher in the hierarchy is the airplane the more point the player gets. These points are calculated and at the end the highest score is saved. The main source code is taken from Mike Gold implementation. This code was free for download and had no author right for using it [2].

## 1.2 History of the Game

According to an article in Wikipedia **Space Invader** is an arcade video game designed by Tomohiro Nishikado and released in 1978 [1]. It was originally manufactured and sold in Japan and later licensed in other countries. It is one of the earliest shooting games and aims to defeat waves of aliens with laser cannon in order to earn as much points as

Figure 1 – Space Invaders first game console

possible [1]. As the article in Wikipedia states *"It was one of the forerunners of modern video gaming and helped expand the video game industry from a novelty to a global industry*" [1]. When first released, *Space Invaders* was very successful. Following its release, the game caused a temporary shortage of 100-yen coins in Japan and grossed US $2 billion worldwide by 1982 [1].

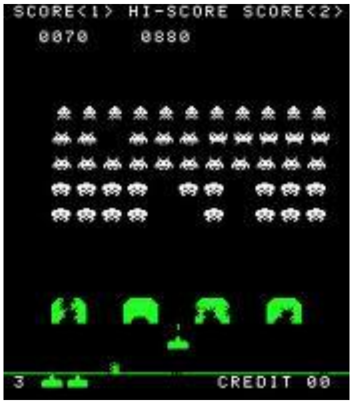


Figure 3 - Game Screen

This game has been a big inspiration for other implementations in different consoles and platforms. Moreover, since this game has been one of the first games in arcade mode, it was also an inspiration for more games of this type to come in the future.

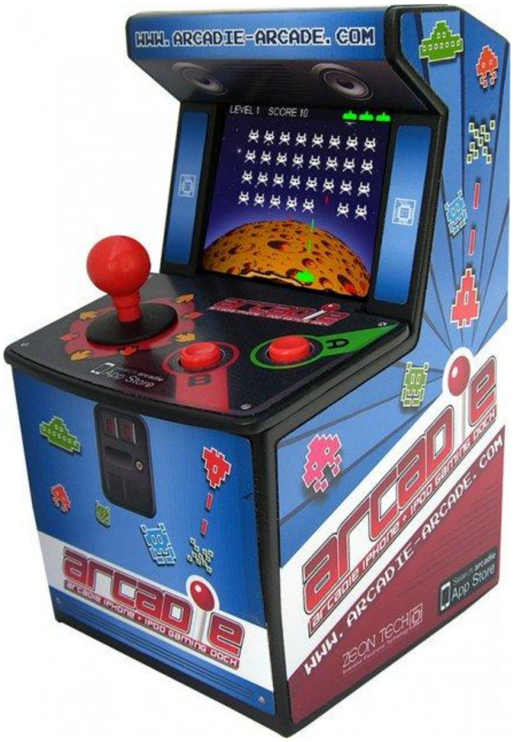


Figure 2 - 2012 console version of the game

# 2 Architecture

## 2.1 Hardware

A personal computer running windows 7 and a Kinect Device optimized for PC is required in order to run this game and have fun. Moreover, the same hardware was used to develop the game.

## 2.2 Software

### 2.2.1 Visual Studio 2010

Microsoft Visual Studio Express is a set of freeware integrated development environments (IDE) developed by Microsoft that are lightweight versions of Microsoft Visual Studio product line. The program has a set of tools to develop custom applications that run on .NET framework. Visual studio 2010 express version can be downloaded for free from Microsoft DreamSpark.

### 2.2.2 Kinect SDK

According to Microsoft, Kinect for Windows SDK can help you develop innovative applications that harness speech and human tracking [3]. The latest version of the SDK, is released in May 2012, offers improved skeletal tracking, face tracking, seated, mode, and new resources, including a developer toolkit with components, tools, and samples [3]. The Kinect SDK can be downloaded for free from Microsoft web page.

### 2.2.3 Speech SDK

Speech SDK is an API developed form Microsoft that allows speech recognition and also text to speech features in applications developed with any visual studio version available in the marker. Speech SDK is supported by most of .NET languages. Furthermore, windows seven incorporates speech recognition feature. This framework uses some of the most advanced algorithms such as Hidden Markov Model and Dynamic Time Warping. The current version of the framework is 5.1 which is also used in our project. This SDK can be downloaded here.

# 3 Modalities

After having found a god idea for our project (game) was raised a question on what modalities should we use? We discussed the possibilities of Kinect and the knowledge we had on dealing with it. As a result we ended up interested to implement all the possibilities of Kinect since none of us had any experience with the Kinect before. So we ended up with the idea of implementing a Touch, Voice (speech) recognition and gesture recognition.

## 3.1 Used modalities

### 3.1.1 Touch

Touch modality was already implemented when we downloaded the source code of the game so the only upgrade we did to this modality was to change the keys used for playing the game in order to make it more user friendly. The player is able to play the game by using the arrows keys left and right located on every keyboard and also the space button for shooting the invaders.

### 3.1.2 Voice Recognition

Voice Recognition was a good and interesting feature to our project. It gave us the first experience and impressions about voice recognition. However, after implementing completely and making a few tests it resulted that voice recognition commands had a few milliseconds delay during the game. This resulted as negative feature for our game since the commands resulted in delayed actions and the game lost the impression of action and fun by making us more nervous than having fun. However, we decided to leave the implementation of voice recognition and test the modality again in our user evaluation tests. With this modality the user is able to trigger all the required actions for playing the game such as: moving left and right and firing to the alien invaders. Furthermore, this modality in cooperation with gesture modality is used to trigger a special action in order to lunch a rocket which is not possible to be done with combination of other modalities.

### 3.1.3 Gesture Recognition

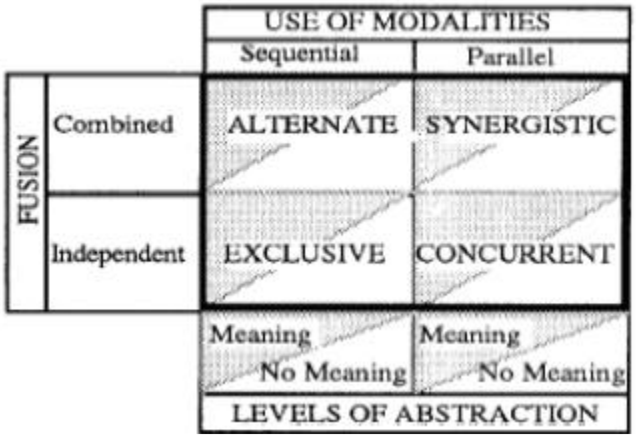
Gesture recognition is of great impact for the future technology because of the extensive applications in virtual reality, signal recognition and computer games. The recently developed depth sensor of Kinect has provided an opportunity in this aspect in terms of features it offers. We used the Kinect sensor in our project in order to recognize some basic gestures of the user. By recognizing these gestures the system makes possible to trigger some actions with impact on the game. Working with gesture recognition was one of the most enjoyable parts of the projects. Besides we had fun we were introduced to a completely new area, the area of gesture recognition. This modality is able to trigger all the actions required in the game and user is completely able to play the game by using only this modality. Moreover, this modality in cooperation with voice recognition is used to trigger a special action in order to fire a rocket.

## 3.2 Case and Care models

### 3.2.1 Case

To express the machine side aspect there are four case modalities; concurrent, alternate, synergistic and exclusive. In our project three cases can be distinguished in order to classify the communication of modalities. The user can use three modalities touch, voice and gesture in order to trigger an action. As a result we have

- **Synergistic Case** between voice and gesture where both modalities are used to trigger an action in order to fire a rocket or using touch in combination with voice or gesture to play the game.
- **Alternate Case** between gesture and voice where one modality at a time can be used to move right ant left
- **Concurrent Case** between gesture and voice where with gestures you move left and right and with voice you say fire.



### 3.2.2 Care

In order to explain the system considering the human side we need a care model. Care model has four properties of multimodal interaction, complementarity, assignment, redundancy and equivalence. In our project three out of four properties are used:

- **Complementary** we have when we use gesture and voice in order to fire a rocket. The player has to keep left hand above the head and say the word “rocket” in order a rocked to be fired.

- **Redundancy** we have between touch and gesture or touch and voice where both this combinations can be used in parallel or sequential to trigger actions for moving right, left or firing a bullet.
- **Equivalence** we have between our three modalities touch, gesture and voice where we are able with all this modalities to trigger the actions of the game.

# 4 User Evaluation

User evaluation was used during the whole development of the game in order to create a playable game. This came as a result of adjustments needed to be done so the game could be playable and fun. Some of the evaluation we did was the method of hand movement type we selected in order to make it more practical for playing. Moreover, we adjusted the speed of the enemy bullets and the airplane in order to be synchronized and deliver more action. Selecting the special action for firing a rocket required a good evaluation during the development in order to make it easily achievable. But even though we had an idea at the end presenting the game to random persons for evaluation is the best method to receive the right feedback and values. So, in order to evaluate the game we tested the game on more people. We tried to have balanced number of genders and not to have much knowledge on computer science and games. In the record we have the participant's age, gender and if he/she knows video games or not. Each participant tested the game a few times according to their will. The recommended steps were first to learn and discover the environment and the game controls with the modalities. So participants first play with all the modalities at least one game in order to get familiar with the game and with the controls. Once the mechanisms and the environment are learned the participant can start the new final games. During these games the participant must use at least once all modalities. This allows later to better see the participant preferences. During participants final games we made sure to note down the scores for every game and higher score at the end. After they finish the demo we provide the participants with a questionnaire or we directly ask the questions and fill the questionnaires in their place. We took in consideration the evaluation of 7 persons. The breakdown is as follows, we had 3 women and 4 men. Their ages varied form 20-29. Most of participants had no god knowledge of computer games except two of them who play regularly computer games. In order to make the question as simple as possible we made a multiple choice questionnaire. Every question had 4 degrees of appreciation. The participant can judge the game and the quality of modalities by answering if he/she likes a lot, if he/she likes, likes a bit, and does not like. Moreover, the participants were free to give any comment according to their wish.

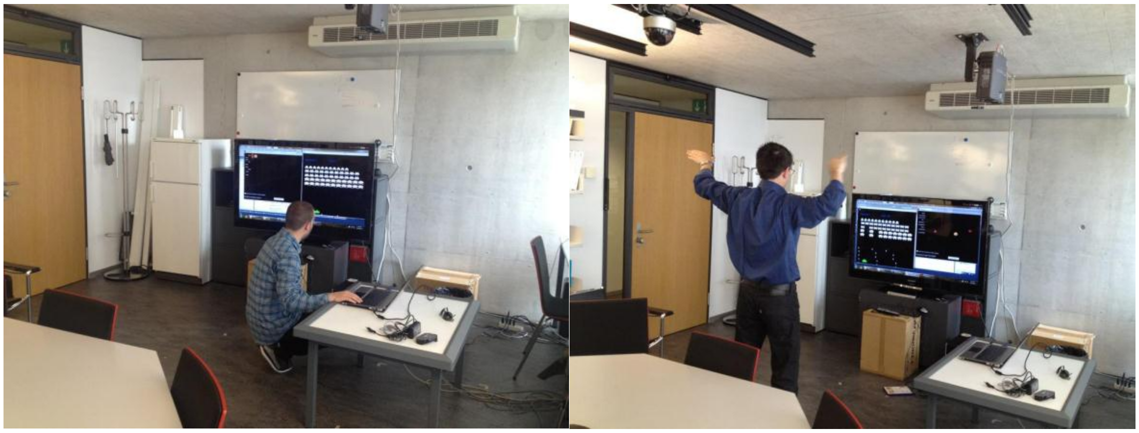


Figure 4 - User Evaluation In Action

# 5 Conclusion

Multimodality in computer science and in everyday technologies lastly has been developing very fast. Today we see this technology being supported from smart phones, computers, TVs and other everyday used devices. As a result, we can conclude that multimodality will be present more and more every day in our everyday life in the near future. This course was a great experience for us. We all came closer to the multimodal technology and the possibilities it offers. However, taking in consideration that the tree of us didn't had any experience in game programing and moreover in voice and gesture recognition this project was very challenging and had a great impact. It was a great learning experience in a few new unexplored areas. We were very concerned about the ongoing of the project at the beginning, but, on the other hand for sure we should had taken it more relaxed since implementing these modalities resulted being more fun and interesting than not passable barriers. During the user evaluation we noticed that we were not the only one who had fun but also friend of us and other users had fun.

# 6 Acknowledgments

We would like to thank Mike Gold for making available the source code to us. Also a special thanks goes to persons who contributed for our user evaluation. Moreover, we are very grateful to the university for making us available all the technology required in this project.

Least but not least we want to acknowledge the helpful comments to Denis Lalanne when we most needed them.

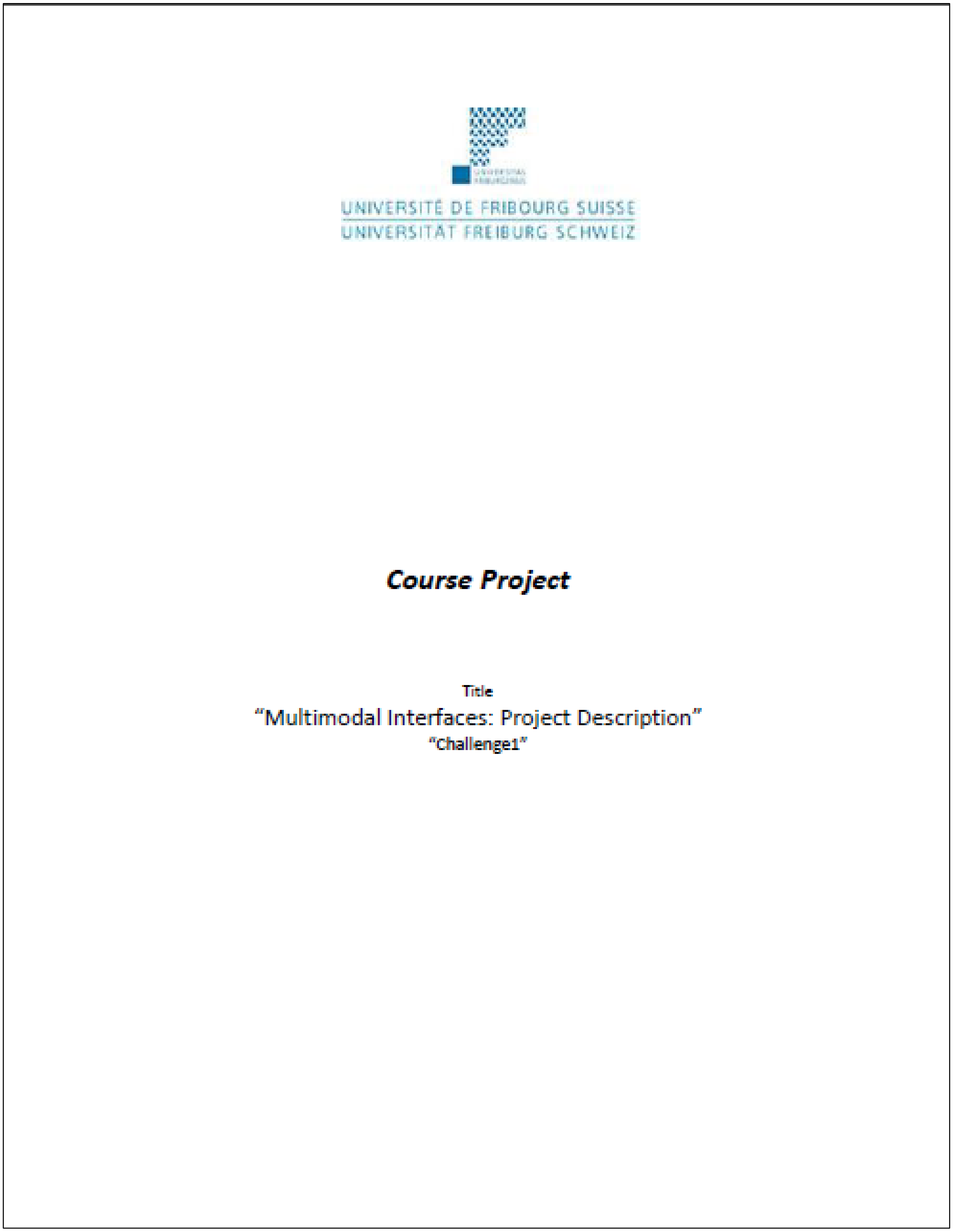
# 7 Resources

[1] Space Invader. 20 May, 2012. http://en.wikipedia.org/wiki/Space_Invaders

[2] Mike Gold. 2007. Space Invaders Source Code http://www.gamespp.com/c/spaceInvadersSourceCode.html

[3] Microsoft. 2012. http://www.microsoft.com/en-us/kinectforwindows/develop/

## A. Project proposal and Description



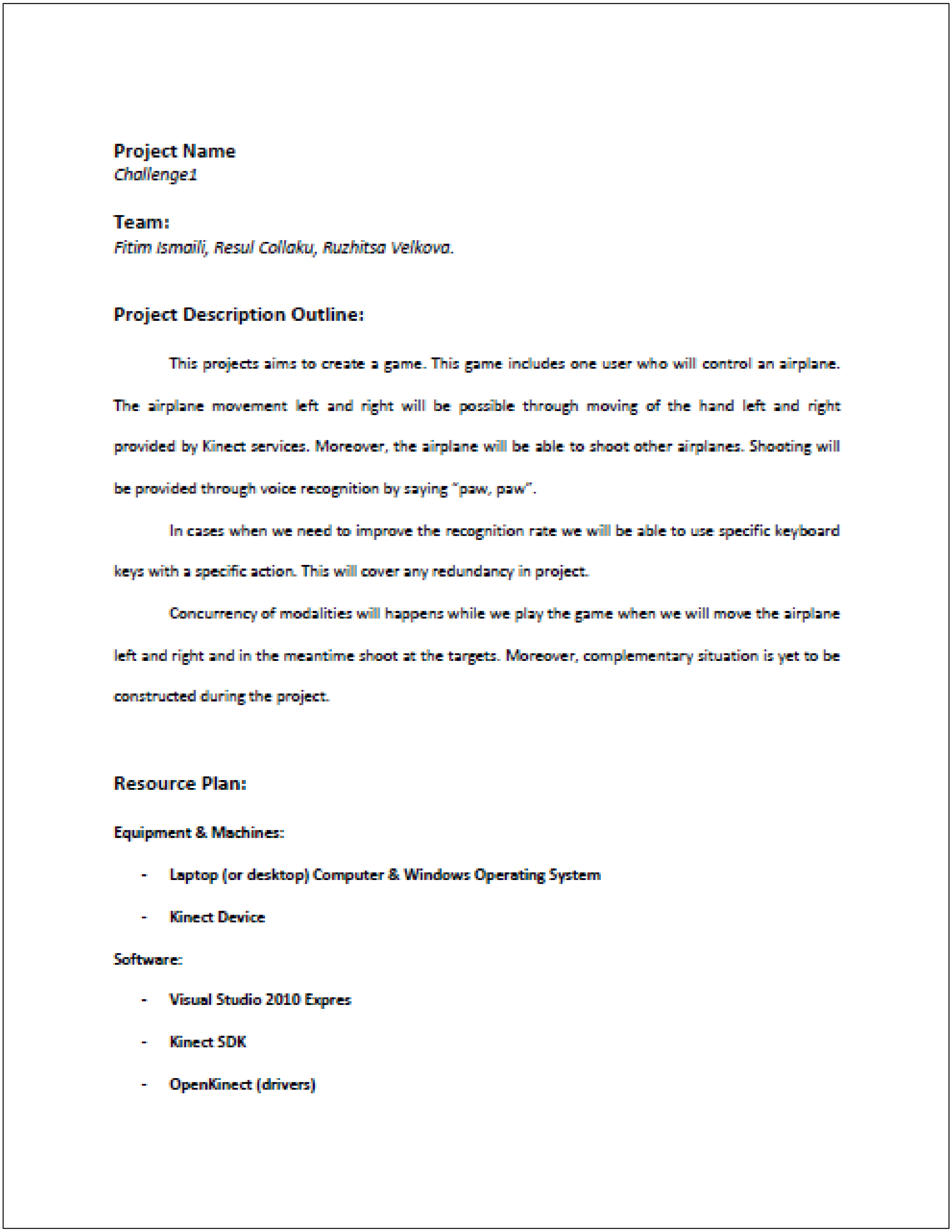
UNIVERSITÉ DE FRIBOURG SUISSE
UNIVERSITÄT FREIBURG SCHWEIZ

***Course Project***

Title

“Multimodal Interfaces: Project Description”

“Challenge1”



## Project Name

*Challenge1*

## Team:

*Fitim Ismaili, Resul Collaku, Ruzhitsa Velkova.*

## Project Description Outline:

This projects aims to create a game. This game includes one user who will control an airplane. The airplane movement left and right will be possible through moving of the hand left and right provided by Kinect services. Moreover, the airplane will be able to shoot other airplanes. Shooting will be provided through voice recognition by saying "paw, paw".

In cases when we need to improve the recognition rate we will be able to use specific keyboard keys with a specific action. This will cover any redundancy in project.

Concurrency of modalities will happens while we play the game when we will move the airplane left and right and in the meantime shoot at the targets. Moreover, complementary situation is yet to be constructed during the project.

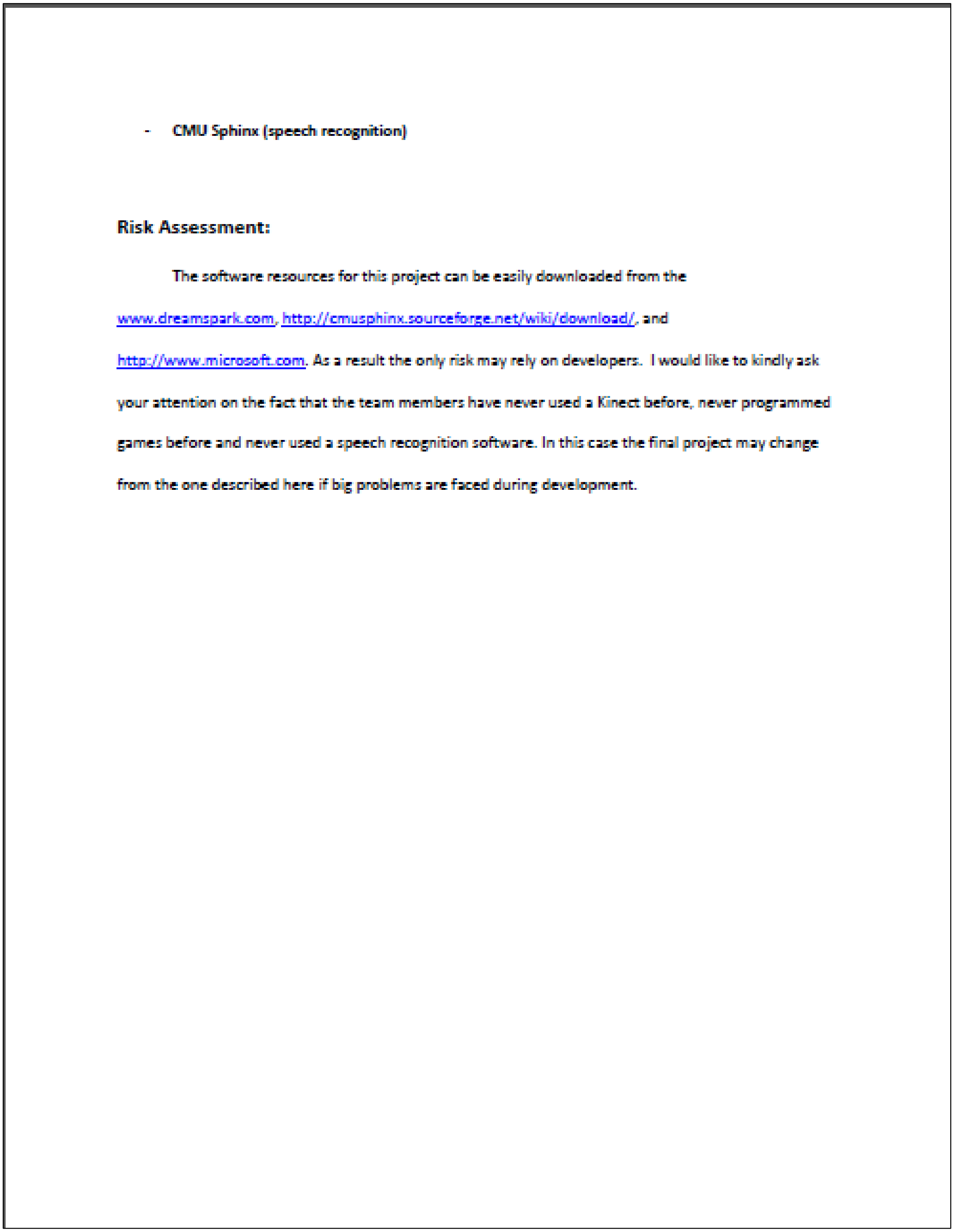
## Resource Plan:

**Equipment & Machines:**

- **Laptop (or desktop) Computer & Windows Operating System**
- **Kinect Device**

**Software:**

- **Visual Studio 2010 Expres**
- **Kinect SDK**
- **OpenKinect (drivers)**



- **CMU Sphinx (speech recognition)**

### Risk Assessment:

The software resources for this project can be easily downloaded from the www.dreamspark.com, http://cmusphinx.sourceforge.net/wiki/download/, and http://www.microsoft.com. As a result the only risk may rely on developers. I would like to kindly ask your attention on the fact that the team members have never used a Kinect before, never programmed games before and never used a speech recognition software. In this case the final project may change from the one described here if big problems are faced during development.

## B. Disc Content

The CD-ROM attached to the report contains the following resources:

- Source Code of the game.
- Project Report
- Intermediary and Final presentation
- Video Demo