# Group Report

## Introduction, Aims, & Objectives

For our group project submission we have decided to meet the criterion set by the brief for creating a novel way of interacting with computers by developing our own software that makes use of the LEAP Motion Controller along with an interface that it can be operated within. During our original assessment the agreed aim of the project was to develop an application “that allows the user to enter numbers into a form in an intuitive manor” – whilst inherently subjective in its nature the concept of whether or not the program would be intuitive is predicated on our ability to make it as clear as possible, especially given that those who use the LEAP Motion will most likely have no prior experience of doing so. Consequently, it was necessary to create our own bespoke GUI using Windows Forms that would provide a way for the end user to effectively communicate with the computer using our chosen peripheral to meet this particular aim. Along with this particular aim we also took into account the more project specific one; the need to develop software which would allow such interaction through the GUI in the first place via our chosen programming language, C#. In short, our aims for the project as they were originally and continue to be are as follows:

* Create a working interface that can be interacted with via the LEAP Motion Controller.
* Develop software that allows the user to communicate their actions with the LEAP to the computer via an appropriate programming language (C#)

In our original documentation, these aims were quantified via a series of general objectives, which were as follows:

* Understand the needs and requirements of the cliental (produce a product to solve the issue of disability regarding computer use): The objective perhaps most fundamental to the criteria of this assessment, it has been tentatively achieved (although not yet realized) by the decision alone to use the LEAP Motion Controller. In the first assignments *Research* section we reasoned the potential use of this peripheral – by its “hands free” nature alone (in the sense that it requires no objects to be held) it proved useful for those disabled both physically and environmentally (e.g. having to wear gloves or having messy/dirty hands) and thus met this objective.
* Build on these requirements to create something unique and innovative: Whilst staunchly subjective depending upon the perception of whether or not using pre-existing hardware is innovative, it can definitely be argued that the software designed for using the LEAP along with our proprietary GUI are unique. Most form based interfaces historically make use of either the common mouse/keyboard arrangement or the use of touchscreens as opposed to our groups proposed method of interacting via the users digits without requiring a physical “touch”.
* Use iterative development to ensure usability: An objective which remains to be achieved until the very end of the project, although its continued use during our design cycle via each group member peer reviewing work and then making changes as necessary ensures that the group is well on its way to attaining this objective by the end of the project.
* Communicate our findings and work efficiently: More of an aim instead of an objective, however it was quantified by the stipulations that the group met regularly and produced minutes of these meetings to prove this, something which has been done and has been provided along with this assignment. Other manifestations of this include our meetings with our project supervisor Marc Hanheide where our progress was relayed and then used to discuss what we would do until the following meeting.

There were also more project specific objectives as well that were concerned more with the actual coding side of the task, and they are as follows:

* Create an application that allows the user to enter numbers in a contactless fashion using the Leap Motion Controller.
* Provide intuitive places to access and store the data (Forms & Fields).
* Involve the use of gestures to navigate through the aforementioned forms and fields.
* Include a way the application can detect and make use of both one and two hands for numerical entry.
* Use source code management to ensure the sharing, development, and protection of the final product.

Some of these original objectives had to be adjusted according to changing circumstances, for example:

* The groups decision to use Windows Forms for the GUI whilst practical given our knowledge of it proves to be problematic in regards to storing data – once the form has been closed, the data is lost. However, after having a group member leave at the beginning of the project we decided not to spread ourselves too thin on using software which we were uncomfortable with and instead stick with what we know to ensure that the solution is delivered on time.

Given these alterations, the final list of requirements can be summed up as follows:

* Using the LEAP Controller as a method of input, design an artefact which can read the users input and output numbers to the screen.
* Design an interface via Windows Forms that will act as a GUI to the users.
* Ensure that the interface can be interacted with by the users such that their inputs are processed by the LEAP Motion and are output to the screen.
* Develop code for both the interface and LEAP Motion Controller using one standardized language (C#).

## Group Project Use Case Scenario

Our group project program involves using only the digits on your hand to input information. We have picked a forensic scenario to demonstrate our solution as it provides an environment in which the use of the LEAP Motion controller would be appropriate for use (e.g. If forensics were wearing gloves they may not wish to enter data via a conventional manner to circumvent contaminating evidence). As a result, a solution such as the one our group has created may be considerably more appropriate due to its use of hands free data entry.

There are many different types of information that may have to be recorded at the scene of the crime, and we have to consider these for our project.

### Different Inputs

1. Whether the body is male or female:

At a scene where someone has been found dead, one of the pieces of information that would have to be recorded would be the gender of the person. This could easily be done by coding the choices of male or female to the numbers ‘1’ or ‘2’ using either 1 or 2 fingers respectively to enter the data. In short, holding one finger over the leap controller to confirm male, or holding two finger over to confirm female.

2. The age of the victim:

Again, at the scene of a murder, the age of the victim would have to be recorded. This information is just as important as the gender of the victim. Knowing the age of the person in question can help to find out what happened in the scenario and whether it was homicide or if there might have been some natural cause of death. Entering the age using fingers should be simple; if the age is in double figures, hold as many fingers up as needed on one hand for the first digit in the age and then do the same again for the second. For example, if the age of the victim was 22, you would hold two fingers up for 2 and then 2 fingers again which would complete the number. If the age only consists of one digit, then only one number needs to be scanned from the user’s hands. For example, if the victim was 9 years old, the user would hold out nine digits over the LEAP Motion and this would be scanned in as a result.

3. The height of the victim:

This would work in much the same way as entering the age of the person. There would be multiple text boxes which can filled in and in these we enter the value for height in feet and then, after a decimal point, the inches. This means the form would be composed of only two boxes in which the user enters the data. For example, if the victim was 6”1’, then the user would hold out 6 digits for the first number, and then 1 digit for the second.

4. Whether any finger prints were found in the area:

Entering this data would work exactly the same as entering the gender of the body. This works as a true or false statement where true is coded to the number ‘1’ and false is coded to the number ‘2’. By holding out either of these numbers the LEAP Motion recognizes the linked response and shows it on screen.

## Tool Appraisal

### Visual Studio 2010

Our coding for this task was carried out using the popular integrated development environment Visual Studio 2010 (VS10), a program which was our unanimous first choice for working with as opposed to other IDE’s due to our previous experience and familiarity with it over the last two years of our course. VS10 provides us with many different project environments and a selection of programming languages with which to work with, and in our case this took the form of C#. With the use of built in error checking and the helpful IntelliSense (Microsoft, 2014) the programming aspect of the project was made far more simple as opposed to merely typing code into any other code editor such as Notepad++ which would not be able to provide similar features (debugging, extensive class libraries etc.) When considering the disadvantages of this particular software it needs to be explicitly defined what it is in comparison to; in comparison to other IDE’s there are almost none, with the only possible exception being its lack of similar support on other operating systems with Windows 7/8 being the clear preferential choice. In comparison to other code editors however Visual Studio takes up a lot of space and can be inefficient when using languages with very little predefined classes (Prolog for example) however this is not the case for our project so it is very difficult to truly provide any weight behind these features as disadvantages.

### Dropbox

One of the tools we used throughout our project was the popular cloud based storage service Dropbox. This tool offered us a practical way of sharing and storing code (prior to the use of Github) but also predominantly for managing our progress on the final report, presentation, and storage of meeting minutes.

This service was particularly advantageous to us for its ability to be downloaded and utilized as a shared desktop folder, allowing members of the group to save their work as normal and have it uploaded to the shared folder without any further action. The implications of this meant that members of the group didn’t have to wait for others to go out of their way to upload a file, and that any edits that took place were instantly re-updated before another member started working on an outdated copy of any one document. Group members also found the service intuitive to use as opposed to Github with the drag and drop nature of file storage preventing a lack of understanding from being a barrier to any collaborations.

This is not to say that Dropbox isn’t without its share of drawbacks – one of the first problems encountered came about from the previously stated advantage; by having new documents saved to the shared folder immediately the folder became cluttered, and as a result it was difficult to ascertain which document other members were supposed to be peer reviewing in accordance with our use of iterative design.

### Github

GitHub is one of the best known resources around for online collaboration, as it acts as a hub sharing code and aiding software development. This resource will come in very handy when we actually begin to write code and bring our project together. There are many useful features to be had here, include each project getting its own “issues page” and the ability for a collaborative code review. This means that a discussion surrounding certain issues can commence with new topics arising as the review continues. Comments can be made on certain pieces of code for everyone to see, which makes it easier to propose changes to be put across.

Thankfully, GitHub does support many programming languages and even has an equivalent mobile app so that the project can still be supported will on the go.

One of the tools we used throughout our development process was Github, an online collaboration service that allows people to share and update code in order to aid the progress of software development. Our main justification for using this particular piece of software was the use of collaborative code commenting, making errors easy to identify and allow changes to be made swiftly between members of the group.

## Project Planning & Management