

University of Tampere
Department of Information Sciences
TIEA4 Project Work – Visualization of human activities based on motion capture

E3:
Visualization of human activities based on motion
capture

Project plan

Ossi Kärki	: 415563
Andreas Achte	: 415515
Tuomas Tammela	: 422098
Vili-Veikko Rauhala	: 421839
John Mäkelä	: 421624

Version history

Version	Date	Author	Description
0.1	29.09.2017	Team	First draft
0.2	01.10.2017	Team	Comments included
0.3	02.10.2017	Team	Iterations added and more
0.4	09.10.2017	Team	supplement
0.5	21.10.2017	Team	correct personal info

Contents

1 Introduction	5
1.1 Purpose and scope	5
1.2 Product and environment	5
1.3 Customer's current system	5
1.4 Development constraints	5
1.5 Definitions, abbreviations and acronyms	5
2 Project organisation	6
2.1 Group members	6
2.2 Customer	6
2.3 Related organisations	6
3 Project goals and ending/termination	6
3.1 Goals of the project group	6
3.2 Goals of the customer	6
3.3 Goals and deliverable of the project	7
3.4 Quitting (termination) criteria of the project	7
3.5 Ending criteria of the project	7
4 Project management	7
4.1 Methods and tools	7
4.2 Monitoring and guidance	7
4.3 Learning and study plan	7
5 Project iterations and timing	8
5.1 Iterations	8
5.2 Iteration 0	8
5.3 Iteration 1	8
5.4 Iteration 2	9
5.5 Iteration 3	9
5.6 Iteration 4	9
5.7 Iteration 5	9
6 Requirements	10
6.1 Functional requirements	10
6.2 Quality goals (Non-functional requirements)	10
6.2.1 Usability goals	10
6.2.2 Performance goals	11
6.2.3 Reliability goals	11
6.2.4 Security goals	11
6.3 User interface requirements	11

7 Test and quality assurance plan	12
7.1 General approach	12
7.2 Definition of done	12
7.3 Special testing	12
7.4 Test documentation	12
8 Risk management	12
8.1 Risk list	13
8.2 Risk monitoring	13
9 Open issues	13
10 References	13
APPENDIX A [...Z]	13

1 Introduction

1.1 Purpose and scope

The purpose of this project is to create an easy-to-use application that translates raw motion capture data into a more comprehensible visualization. The scope is limited to capturing the movements of a frisbee disk being thrown.

1.2 Product and environment

Visualization of human activities based on motion capture control.

The disk will be attached with rigid body markers. Calculations can be made about the disk's motion and compared with optimal performance to give feedback to the user.

The movement will be captured on motion cameras in UTA's SimSpace laboratory. A net will be needed in SimSpace for the person to throw at.

1.3 Customer's current system

The customer is currently using third party software to visualize their motion capture data, as they do not have software of their own.

1.4 Development constraints

The throw will be recorded in SimSpace with motion capture cameras. Technologies used include Unity3D with C#.

Specifically the plan is to capture the disk's movement, instead of the human arm.

1.5 Definitions, abbreviations and acronyms

Unity3D	Development environment used in this project.
SimSpace	Motion capture laboratory located at the university.

2 Project organisation

2.1 Group members

List for each: Person, Contact info, experience, knowledge, interests

Tuomas Tammela (Tammela.Tuomas.H@student.uta.fi)

- Experience in Unity3D using C#, and in web development using JavaScript.

Vili-Veikko Rauhala (Rauhala.Vili.V@student.uta.fi)

- Experience with Java and basics of JavaScript

Andreas Achte (Achte.Andreas.x@student.uta.fi)

Ossi Kärki (Karki.Ossi.O@student.uta.fi)

Experience with Java and JavaScript

John Mäkelä (Makela.John.A@student.uta.fi)

- Experience with 3d engines in Java

2.2 Customer

University of Tampere, Jaakko Hakulinen (Jaakko.Hakulinen@uta.fi)

2.3 Related organisations

The project relates to several ongoing research projects in Uta's CIS department.

3 Project goals and ending/termination

3.1 Goals of the project group

Our goal is to make functional and user friendly product that shows relevant information of disk movement for user in the way that is satisfying for our customer and for us. We hope to stay in schedule and to work efficiently as a team. Also our goal is to learn more of project working and use of selected applications. Every group member has their own goals for academic credits.

3.2 Goals of the customer

Customer wants application that user can use to learn to throw frisbee properly. Simulation doesn't have to be corresponding with real life throw, but it should demonstrate the throw somehow. Customer also wants to see and learn what kind of data can be captured with their equipment.

3.3 Goals and deliverable of the project

Goal is to return an application that satisfy us and customer that can be used to capture and demonstrate data from frisbee throw.

3.4 Quitting (termination) criteria of the project

If customer decides to get rid of OptiTrack system, there would be no point produce this kind of program as there is no way to use it. Therefore customer and project manager can terminate the project in agreement. Terminating the project is very unlikely.

3.5 Ending criteria of the project

Customer is satisfied with the produced running piece of software that implements a reasonable number of client requirements.

4 Project management

4.1 Methods and tools

- ScrumBut
- Trello
- GitHub
- Slack
- Email

4.2 Monitoring and guidance

4.3 Learning and study plan

- Training provided by Ilari Lehtonen to use GitHub
- Training provided by the customer to use equipment
- Learning Unity3D in group
- Telling about used terms in group
- Separate teams for UI and motion capture

5 Project iterations and timing

5.1 Iterations

Project is divided into 5 iterations (sprints) that last 2 weeks each. First actual sprint will be started on 9.10.2017. See iterations described below for more information and Table 5.1 below which lists the deadlines for the project.

Table 5.1. Deadlines

Date	Deadline. What should be ready?
4.10.2017	(Iteration 0) Project plan is reviewed
23.10.2017	(Iteration 1) An investigation of whether or not the hardware timeout is sufficient to monitor the fast throw movement is ready. First initial versions of throwing (in-flight) view and engine are ready.
06.11.2017	(Iteration 2) Second and final version of throwing view is ready. Second version of engine is ready. Test specification (test cases) ready for the engine and throwing view.
20.11.2017	(Iteration 3) First initial version of Analysis (post-throw) view is ready. At this phase, the transition between throwing and analysis views happens automatically after the throw is completed.
04.12.2017	(Iteration 4) Second version of the engine is ready. Second version of analysis view is also ready.
18.12.2017	(Iteration 5) Final version of the analysis view is ready. Test specification ready and all the tests executed. Required documentation is ready.

5.2 Iteration 0

4.10.2017 - 22.10.2017

First (draft) version of this project plan will be reviewed.

5.3 Iteration 1

23.10.2017 - 05.11.2017

At first, an investigation of whether or not the hardware timeout is sufficient to monitor the fast throw movement will be done. Whole team will be involved to this on at least some level to be familiarized to the devices and the environment. However, there will be some appointed persons to this task especially.

Development work of all the necessary parts for the final running software will be started during this iteration. Also writing of the test specification (test cases) will be started.

First initial version of the software will be ready during this iteration, which contains first versions of the throwing (in-flight) view and the engine. First version of the throwing view contains at least following features: frisbee tracking and recording the data. Obviously this means that the engine must support those as well, so the the development of the engine and UI will be done in parallel.

5.4 Iteration 2

06.11.2017 - 19.11.2017

Second version of the running software will be ready at the end of this iteration. this includes a final version of the throwing view and a second version of the engine. At this phase the UI will be more sophisticated and includes at least all the known required features for the throwing view which are following: frisbee tracking, recording the data and an indicator which shows the right direction for the next throw. Test specification (test cases) will be also ready for the engine and throwing view.

5.5 Iteration 3

20.11.2017- 03.12.2017

At this phase, the first initial version of the Analysis (post-throw) view will be ready and added to the running software. This version doesn't contain much of all the features needed for the final version. However, the transition between throwing and analysis views happens automatically in this version, when the throw is completed. Also some information will be presented about the throw in analysis view after the transition. For example how fast the throw was. Test specification will be updated to include test cases for the new features.

5.6 Iteration 4

04.12.2017 - 17.12.2017

At this point the second version of the engine will be ready. Therefore, second version of analysis view will be also done. This also means that most of the missing required features for the analysis view will be implemented during this iteration. The features include scaling the size of the graph, emphasizing either the vertical or horizontal movement, examining any point in the flight path by hovering over it, playing the entire animation in real time or slow motion, and also changing some settings. Test specification will be updated to include test cases for the new features.

5.7 Iteration 5

18.12.2017 -

Final version of the running software will be ready at the end of this iteration. Finishing of the software will be done at this stage. Test specification will be ready at this point and all the test cases will be executed. All the required documentation will be written at this point.

6 Requirements

6.1 Functional requirements

The main requirement is creating a visualization from the movements of a frisbee disk being thrown. This means tracking the position, rotation, movement speed and rotational speed of the disk as it moves through space, and visualizing that information in form a graph that the user can easily understand.

The application will also require an user interface, with which the user can examine the visualization in more detail. It would be controlled with a mouse. The functions include scaling the size of the graph, emphasizing either the vertical or horizontal movement, examining any point in the flight path by hovering over it, playing the entire animation in real time or slow motion, and maybe also changing some settings.

6.2 Quality goals (Non-functional requirements)

1.1.1 6.2.1 Usability goals

The main goal is for the user to be able to train their frisbee-throwing skills using the software. This means knowing where to throw the frisbee and interpreting the resulting visualization correctly. The user could be helped by showing a marker on the screen that indicates the intended throw direction, making it clear that the frisbee is indeed being tracked (or giving a warning if it's not) and making the post-throw analysis screen as clear and unambiguous as possible.

On top of that, the interface should also be intuitive to use, and automate as much as possible. For example, the transition between in-flight and post-throw views should happen automatically after the throw is completed, without any input from the user.

1.1.2 6.2.2 Performance goals

Efficiency is key in this project, as the application will have to process large amounts of raw motion capture data in as little time as possible. For the visualization to be useful, the user shouldn't have to wait for more than a few seconds between the throws.

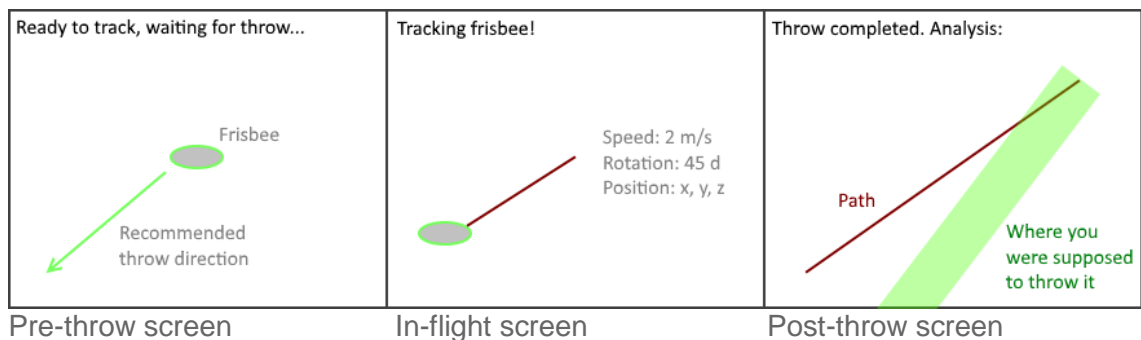
1.1.3 6.2.3 Reliability goals

The resulting visualization has to be reliable enough so that the user can improve their frisbee-throwing skills. This means finding a scale (e.g. down to the centimeters) that is reliable enough to display every throw with the same level of accuracy.

1.1.4 6.2.4 Security goals

Security is not a concern in this project, as the user will not enter any data in any systems. Even the motion capture will only track the disk, and not the thrower.

6.3 User interface requirements



The first screen appears when the application is launched. It shows the frisbee in the center and indicates that the motion capture system is ready to track it. The intended throw direction is shown with an arrow.

The second screen appears when the frisbee is detected to be flying (based on velocity). It shows some real time stats about the frisbee and draws the flight path.

The third screen appears when the throw is completed. It shows the flight path compared to the intended throw direction from the first screen. The user can then press a (invisible) button to return to the first screen.

7 Test and quality assurance plan

7.1 General approach

Developers will do the unit tests during the implementation in workstations. On system level there will be specified test cases, which will be executed during the each iteration in lab where all the equipments are. There will be some appointed testers. This time there is no need for consultants.

7.2 Definition of done

Feature is ready (done) when all the related test cases are passed and customer has approved the demonstration.

7.3 Special testing

.

7.4 Test documentation

There will be test specification which includes test cases. Test report will be used to save test results.

8 Risk management

Since this project is quite dependant on the equipment and the devices in the laboratory and generally it would not be very uncommon if there suddenly appears to be some issues with the devices in the lab. This can be considered as a quite harmful risk since it can sometimes take time to fix devices or get new ones.

Also one risk can be the access to the lab which can prevent the project to be in time within the deadlines. This risk can be prevented by giving a key to the lab for some project member.

Also sudden lack of motivation among some project members or illness can be a risk for this project to be in time. That can be quite hard to prevent.

8.1 Risk list

Issues with OptiTrack system -

- hard to predict
- Severity depends how long it takes to get system working
- If there appears problems with equipment, need to contact corresponding person

Inaccessible lab -

- Depending if we get a key or not to lab
- If we do, there shouldn't be problems as lab is not much in use
- If we don't get a key, we need to reserve lab and ask someone to open it for us for every visit

Sickness or lack of motivation -

- Hard to predict
- If someone gets sick or drops out of group, it shouldn't be big problem as we have a big group
- If this risk happens we must looking after the tasks the person had and share them in team.

8.2 Risk monitoring

Risk monitoring has been transferred to MMT.

9 Open issues

10 References

Kerzner, 2013. Project management: a systems approach to planning, scheduling, and Controlling, Wiley, 2013.

APPENDIX A [...Z]

[Include appendices as needed.](#)