Faculty of Engineering Science and Technology (IVT)
Department of Industrial Engineering

The Pre Study Report
Digital twin with Visual Components

Stud. Techn. Halldor Arnarson

UiT THE ARCTIC UNIVERSITY OF NORWAY

Title: Digital twin with Visual Components	*Date:* 05 / 02 / 2019 *Classification* Class
Author(s): Halldor Arnarson	*Number of pages:* 11 *Number of attachments:* 15
Subject Name: Master Thesis - Pre Study Report	*Subject Code:* SHO6266
Faculty: Faculty of Engineering Science and Technology	
Master Program: Industrial Engineering	
Supervisor: Gabor Sziebig	
Co-supervisor Wei Deng Solvang	
External Organization/Company: Non	
External Organization's/Company's Liaison: Non	
keywords Digital twin, Visual components, OPC UA	
Abstract (max 150 words): The industry is becoming more connected and smarter with industry 4.0 and IoT. A important part of this development is being able to create a digital twin of a physical system. This report aims to create a digital twin of a KUKA 30-3 with the use of the simulation program Visual Components. To connect Visual Components and the robot a OPC UA server is used as a tool to share information.	

Table of Contents

List of Figures

List of Tables

1 Introduction

The first part of the master thesis was to conduct a literature review on digital twin, Open Platform Communications (OPC UA) standard and how to communicate with the KUKA robot at the laboratory at UiT.

A Digital twin is a virtual copy of a physical system or sub system. Between the to systems you have a big collection of data to be able to simulate the physical system. This emerging tool can be used to improve and the manufacturing industry and production. Whoever to enable communication between the physical system and the digital model you need a way to communicate. OPC UA is a new IEC standard which allows data exchange between different manufacturing products. It defines how information should be sent and is used to collect and share data.

The secound part of the thesis is to create a digital twin at the laboratory on campus Narvik, using the information gathered in the first part. This will be done by using code and applications that is compatible with the equipment at the laboratory. A digital twin will be created of the KUKA KR 30-3 robot in the labratory. The robot is controlled by a KUKA KR C2 and to create a digital twin there has to be communication between the controller and a computer. To visualise the robot, the program Visual Components will be used to simulate the movement of the robot and also controller it.

To send data from the computer to Visual Component the comunication standard OPC UA is used. Visual Components has already built inn a conectivity tab which makes it easy to conect Visual Components to a OPC UA server.

1.1 Background

Smart house are becoming more and more popular. The concept of a smart house is to be able to control the television, lights, the coffee machine and so on, with a single controller or you're phone. This is done by connecting all the devices and lights to the internet to enable easy control over the house.

With this development of the smart house it is inevitable that the manufacturing industry becomes smart. A smart factory is able to connect the machine together and collect data that can be further analysed to improve the factory. Big part of this development is the digital twin. Having a digital twin gives you access to large amount of data, and the ability to test different programs in an offline model of the system/robot.

I have had previous experience in developing a digital twin. In the course operations research 2 we worked in a group and created a digital twin of a intersection in Narvik. First we went to the intersection and collected data on when the car arrives and the times of the lights and so on. With this data and the help of Google Maps we where able to create a digital twin of the intersection. The simulation was created in Visual Components and from the simulation we where able to find the maximal capacity of the intersection. The simulation was also used to test out other configurations of the lights, to improve the waiting time. This project shows that the digital twin tool can be used to improve a system in an offline environment.

2 Problem statement

In the EU there are strict rules when it comes to human robot collaboration. Today fences are often used to child the human from the robot or sensors which limit the human and robot collaboration. By using a real time digital twin you can get information on where the robot is and with sensors you might also be able to detect where the human is. Being able to track both the human and the robot, you can use the digital twin as a fence. This can allow for more human robot collaboration which can give a more flexible manufacturing system.

The digital twin can also be used to improve the system, with offline programming. This gives the opportunity to optimise a system and test different scenarios and preform what if analyses in offline mode.

3 Project description

The description of the project is to create a digital twin of the KUKA KR30-3 robot in the laboratory at UiT Narvik. This will be done with using Visual Component as a simulation software and OPC UA to enable communication between the the computer connected to the KUKA KR C2 controller and Visual Components.

4 Theory/hypothesis

4.1 OPC UA

OPC UA will be used as a middleware between Visual Components and the computer connected to the KUKA controller. A OPC UA server can easily be created in using ”FreeOpcUa” library. This library can be installed by writing ”pip install opcua” in the command promt on windows. In the library there are examples codes on how to set up the OPC UA server. There is a file called ”server-minimal.py” which shows how to create a minimal server. The server creates a folder called ”MyObject” that has a variable which can be edited called ”MyVariable” [1].

4.2 Visual Components

Visual Components is used to visulse the digital twin in this porject and it needs to be connected to the OPC UA server. On Visual Components website [2] you can find a brief YouTube tutorial, which show you how you can easily connect the simulation to OPC UA server. This is done first adding the connectivity tap to Visual Components and then connecting to the OPC UA servere. When you are connected you can pair variables from the OPC UA server with variables in the simulation.

The architecture of Visual Components is open and makes it easy to customize the platform. Visual Components is built on .NET, giving developers using the program an familiar framework. There is also a Python API, to be able to customize everything from the UI to simulation behaviour [3]. In other words Visual Components was built with the ability to customise and change the simulation as needed.

4.3 Communication with the KUKA controller

There already exists a open-source communication interface for KUKA robots called JOpenShowVar. It is compatible with KUKA robot controller version 4 and KUKA robot controller version 2 and therefore should be compatible with the KUKA robot controller at UIT Narvik machine laboratory. JOpenShowVar is a Java open source cross platform communication interface, that allows reading and writing of all controlled manipulators variables [4]. There exists a translated version JOpenShowVar in python [1] and it would be beneficial to use, so that cross programming language can be avoided.

The idea is to run KUKAVARPROXY on the KUKA controller and have JOpenShowVar running on a computer as shown in figure 1.

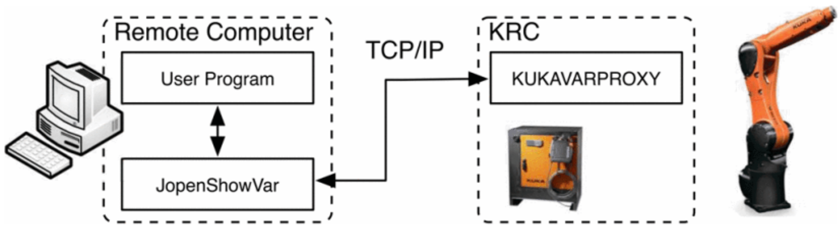

Figure 1: How proposed communication architecture for JOpenShowVar by [5].

5 Assumption

Since I am creating a digital of the KUKA robot, I assume it is possible to get position of the six rotations on the robot. I will also assume its possible to control the robot from a computer.

In [4], the KUKAVARPROXY software running on the controller uses a average of five milliseconds to send data from the robot controller to the computer. It has therefore been said that the KUKAVARPROXY can only be used as a soft real time tool. I therefore will assume that a five millisecond communication speed is sufficient for this project.

6 Risks

When a project is being executed there will always be risks. It is important to identify the risks and either accepted the risks or try to prevent them.

6.1 Risk identification

First the risk in this project have to be identified. There are risks in not being able to to communicate with the robot as well as robot collision with the equipment at the laboratory. Potential risks are listed bellow:

6.1.1 Not establishing communication:

From previous work, a digital twin was created at NTNU with a KUKA KR C4 controller. Whoever at Narvik there is a KUKA KR C2 controller and since a digital twin hasn't been created with this controller it might be a risk that it wont work.

6.1.2 Robot collision:

The first tasks when it comes to creating communication between the robot controller and the computer, will be to read the position of the robot controller and simulate it in visual component. Next task is to control the robot from Visual Component. In this part there can be risk that the robot collides with other object in the laboratory.

6.1.3 Communication being to slow and not synchronised moves:

If the communication is to slow, the digital twin cant be used as a real time digital twin.

6.1.4 Not enough time because of occupation:

There is another student which will use the KUKA for welding. This can be a problem if the uses of the robot isn't coordinated properly.

6.1.5 Not enough time to make the digital twin:

In this project i am going on a exchange to China. This limits the time i spend at the laboratory her at UiT labratory.

6.1.6 Cannot program the robot from Visual Components:

There hasn't been create a digital twin with the KUKA KR C2 robot controller. It is therefore uncertain if it is possible to create a digital twin of the robot.

6.1.7 Bugs in programs:

In this project i will use programs that have already been written like the KUKAPROXY and the python version of JOpenShowVar. There might be bugs in these programs or errors.

6.2 Risk analysis

When the risk have been identified, the consequence and probability can be given a factor between 1 to 5, where 1 is low and 5 is high. Afterwards the potential impact of the risks can be evaluated by multiplying the two number together.

Risk	Risk factor	Consequence	Probability	Impact potential
A	Not establishing communication	5	1	5
B	Robot collision	2	2	4
C	Not enough time because of occupation	5	1	5
D	Not enough time to make the digital twin	5	4	20
E	Cannot program the robot from Visual Components	4	3	12
F	Bugs in programs	5	2	10

Table 1: List of the different risks in the project

The risks can be put into a ”Risk Reporting Matrix” [6], to visualise the biggest risk in the project. If the letters are placed in the red routs, it means that both the probability and consequence are high and should be thoroughly investigated.

Consequence					
	5	10	15	20	25
	4	8	12	16	20
	3	6	9	12	15
	2	4	6	8	10
	1	2	3	4	5
	Probability				

Table 2: ”Risk Reporting Matrix”

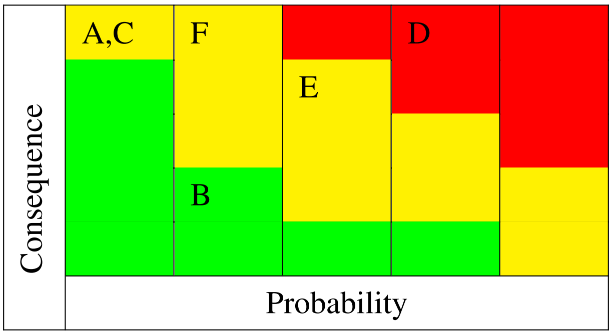

Table 3: "Risk Reporting Matrix" with the risks from table 1

Risk A - Not establishing communication

The risk with potential the biggest consequence would be that communication is not established. From previous work done in the literature review the KUKAPROXY program should be able to run on the KUKA KR C2 controller. It is therefore unlikely that communication will not be established and this risk is therefore accepted.

Risk B - Robot collision

When the robot is controlled by Visual Components there is always the risk that the robot collides with a object. To minimise this risk there will be set limitations on what values can be sent to the robot controller from the server. This will limit the movement of the robot but protect against collision.

Risk C - Not enough time because of occupation

To optimise the usage time of the robot, i will coordinate with the student that is also working with the KUKA robot.

Risk D - Not enough time to make the digital twin

I am going on exchange to China, which limits the time spent on the robot. To minimise the chance that i will not be able to finish making a digital twin i will work long hours every day to finish the digital twin.

Risk E - Cannot program the robot from Visual Components

There may be problem controlling the robot from Visual Components. This is however hard to prevent and is therefore accepted.

Risk F - Bugs in programs

If there are bugs in the program or if the program don't work, can be fixed either by changing the code or looking on the internet for people with the same problem to see if they have found a solution to the problem.

7 Activities

The activities of this project are sorted into 4 parts. In the first phase of the project a one way digital twin is create, which means that the digital twin is only controlled from the robot controller and not Visual Components. Phase to will be adding the ability to control the robot from Visual Components. The last phase will be to look at different ways to control the robot from Visual Components and improve the OPC UA server. The last part is work that is done throughout the project, like writing the report, making presentation and making the progress report.

All the activities are listed in table 4 and 5.

Phase 1			
Activity	**Name**	**Activity description**	**Results**
1.1	Pre-Study report	Plan the project with the scope of the project, risks, timetable and activity list.	A report which describes the execution of the project.
1.2	Model in Visual Components	Create a model in Visual Components of the laboratory, this includes the KUKA robot and the objects around it.	Visualised model to be able to simulate the robot movement.
1.3	Visual Components and OPC UA connection	Make a simple OPC UA server that can connect to Visual Component and control the KUKA robot in Visual Components from the OPC UA server	Being able to control the KUKA robot from the OPC UA server.
1.4	Communication with KUKA controller and computer	Being able to receive position data from the robot controller to the computer.	Receive data of the robots position.
1.5	One way digital twin simulation	Being able to simulate the robot movment in real time.	One way digital twin simulation
Phase 2			
Activity	**Name**	**Activity description**	**Results**
2.1	Robot Control	Using the computer to controller the robot.	Having the ability to control the robot from the computer.
2.2	Send position data	Send position data from Visual Components to the OPC UA server.	Can edit and control variables in the OPC UA server from Visual Components.
2.3	Two way digital twin	Putting activity 2.1 and 2.2 together to control the robot.	The ability to control the robot from Visual Components.

Table 4: Activities in phase one and two in the project.

Phase 3			
Activity	**Name**	**Activity description**	**Results**
3.1	Literature study on robot control	Looking for different ways a robot can be controlled through a simulation software like Visual Components.	Find ideas on how to control the robot from Visual Component.
3.2	Visual Components control options	Add a good control interface of the KUKA robot in Visual Components, using the information in activity 3.1.	Make it easy to control the KUKA robot from Visual Components.
3.3	Add features to server	Add features to the OPC UA server and improve up on it.	A more capable server.
Work done throughout the project			
Activity	**Name**	**Activity description**	**Results**
4.1	Write report	Continuously writing the report throughout the project.	Well done report
4.2	Presentations	Make a presentation and prepare for it.	Ready for presentation
4.3	Progress report	Fill out progress report	A written progress report.

Table 5: Activities in phase three and work that will be done throughout the project.

7.1 Milestones in the project:

To be able to measure how far the project has gotten, milestones are set up as a measurement of project maturity as shown in tabel 6.

Milestones:	Dates:	Description:
Phase 1	27.01.2019	One way digital twin.
Phase 2	24.02.2019	Two way digital twin.
Phase 3 - Visual Components control	14.04.2019	Found different ways to control the robot from Visual Components.
Phase 3 - Improved server	26.05.2019	Added features to the server.

Table 6: The millstones in the project.

8 Scope

The scope of the project is to create a digital twin of the KUKA KR30-3 robot in the laboratory at UiT Narvik. This will be done with using Visual Component as a simulation software and OPC UA to enable communication between the the computer connected to the KUKA KR C2 controller and Visual Components.

The goals of the project are listed bellow:

- Create a digital twin of the KUKA robot in the laboratory at UiT using Visual Components for simulation and OPC UA server to establish communication. OPC UA will be used as a middleware between Visual Components and the computer connected to the KUKA controller.
- See if it is a possible to create a digital twin with an older KUKA controller (KR C2), then has been done before (NTNU digital twin with KR C4).
- See if OPC UA is a good solution to connect the simulation model with the KUKA robot.
- Find ways to control the robot from Visual Components.
- Execute demonstration of the proposed solution
- Document the executed work in the form of written report.
- Prepare a PowerPoint presentation and give an oral presentation of the performed work.

References

[1] Aksel Øvern. *Industry 4.0 - Digital Twins and OPC UA*. 2018. URL: `http://hdl.handle.net/11250/2561319`.

[2] Visual Components. *Connect a Remote OPC UA Server*. URL: `http://academy.visualcomponents.com/lessons/connect-a-remote-opc-ua-server/`. (accessed: 21.11.2018).

[3] Visual Components. *VISUAL COMPONENTS 4.1*. URL: `https://www.visualcomponents.com/products/visual-components-4-0/`. (accessed: 07.11.2018).

[4] F. Sanfilippo et al. "Controlling Kuka Industrial Robots: Flexible Communication Interface JOpenShowVar". In: *IEEE Robotics Automation Magazine* 22.4 (Dec. 2015), pp. 96–109. ISSN: 1070-9932. DOI: `10.1109/MRA.2015.2482839`.

[5] F. Sanfilippo et al. "JOpenShowVar: An open-source cross-platform communication interface to Kuka robots". In: *2014 IEEE International Conference on Information and Automation (ICIA)*. July 2014, pp. 1154–1159. DOI: `10.1109/ICInfA.2014.6932823`.

[6] Henry Livingston. *COUNTERFEIT PART RISK ANALYSIS – MOVING FROM "SUBJECTIVE ASSESSMENTS" TO RISK ANALYSIS SUPPORTED BY EMPIRICAL DATA AND DEFENSIBLE ESTIMATES*. `https://counterfeitparts.wordpress.com/2013/02/28/counterfeit-part-risk-analysis-moving-from-subjective-assessments-to-risk-analysis-supported-by-empirical-data-and-defensible-estimates/`.

Appendices

A Activity description: Pre-Study report

<table>
<tr><td colspan="2">Project title:
Digital twin with Visual Components</td><td>Date:
11.01.2019</td><td>Sign:
HA</td></tr>
<tr><td>Activity no:
1.1</td><td colspan="3">Activity name:
Pre-Study report</td></tr>
<tr><td colspan="4">Responsible:
Halldor Arnarson</td></tr>
<tr><td colspan="4">Task description/intention:
A report which describes the execution of the project.</td></tr>
<tr><td colspan="4">Scope:
Plan the project with the scope of the project, risks, timetable and activity list.</td></tr>
<tr><td colspan="4">Method:
On the computer</td></tr>
<tr><td colspan="4">Dependency:
Non</td></tr>
<tr><td colspan="4">Documentation/results:
A written pre-study report</td></tr>
<tr><td colspan="2">Written by:
Halldor Arnarson</td><td colspan="2">Duration(days/weeks):
Four days</td></tr>
</table>

B Activity description: Model in Visual Components

<table>
<tr><td colspan="2">Project title:
Digital twin with Visual Components</td><td>Date:
11.01.2019</td><td>Sign:
HA</td></tr>
<tr><td>Activity no:
1.2</td><td colspan="3">Activity name:
Model in Visual Components</td></tr>
<tr><td colspan="4">Responsible:
Halldor Arnarson</td></tr>
<tr><td colspan="4">Task description/intention:
Visualised model to be able to simulate the robot movement.</td></tr>
<tr><td colspan="4">Scope:
Create a model in Visual Components.</td></tr>
<tr><td colspan="4">Method:
Measure the laboratory and the equipment and afterwards make a model with the measurements.</td></tr>
<tr><td colspan="4">Dependency:
Non</td></tr>
<tr><td colspan="4">Documentation/results:
A accurate model that can be used to simulate the robot movement.</td></tr>
<tr><td colspan="2">Written by:
Halldor Arnarson</td><td colspan="2">Duration(days/weeks):
1 week</td></tr>
</table>

C Activity description: Visual Components and OPC UA connection

Project title: Digital twin with Visual Components		*Date:* 11.01.2019	*Sign:* HA
Activity no: 1.3	*Activity name:* Visual Components and OPC UA connection		
Responsible: Halldor Arnarson			
Task description/intention: Create a OPC UA server that can connect to Visual Components.			
Scope: Enable connection between Visual Components and a OPC UA server.			
Method: Use a computer to enable communication.			
Dependency: Non			
Documentation/results: Have the ability to send data from the OPC UA server to Visual Components.			
Written by: Halldor Arnarson	*Duration(days/weeks):* 1 week		

D Activity description: Communication with KUKA controller and computer

<table>
<tr><td colspan="2">Project title:
Digital twin with Visual Components</td><td>Date:
11.01.2019</td><td>Sign:
HA</td></tr>
<tr><td>Activity no:
1.4</td><td colspan="3">Activity name:
Communication with KUKA controller and computer</td></tr>
<tr><td colspan="4">Responsible:
Halldor Arnarson</td></tr>
<tr><td colspan="4">Task description/intention:
Get position data of the robot.</td></tr>
<tr><td colspan="4">Scope:
Connecting a computer to the KUKA controller and enable communication.</td></tr>
<tr><td colspan="4">Method:
Installing KUKAVARPROXY on the controller, connect the computer to the controller and receive the rotation of the joints on the robot.</td></tr>
<tr><td colspan="4">Dependency:
Activity no: 1.3</td></tr>
<tr><td colspan="4">Documentation/results:
Enable communication with the KUKA controller</td></tr>
<tr><td colspan="2">Written by:
Halldor Arnarson</td><td colspan="2">Duration(days/weeks):
2 weeks</td></tr>
</table>

E Activity description: One way digital twin simulation

<table>
<tr><td colspan="2">Project title:
Digital twin with Visual Components</td><td>Date:
11.01.2019</td><td>Sign:
HA</td></tr>
<tr><td>Activity no:
1.5</td><td colspan="3">Activity name:
One way digital twin simulation</td></tr>
<tr><td colspan="4">Responsible:
Halldor Arnarson</td></tr>
<tr><td colspan="4">Task description/intention:
Enable one way digital twin.</td></tr>
<tr><td colspan="4">Scope:
Connect tougher activity 1.2, 1.3 and 1.4 to make the digital twin.</td></tr>
<tr><td colspan="4">Method:
Connect the OPC UA server with the robot controller to receive rotation data and send the data to Visual Components to simulate the robot movement.</td></tr>
<tr><td colspan="4">Dependency:
Activity 1.2, 1.3 and 1.4</td></tr>
<tr><td colspan="4">Documentation/results:
Simulate the robot in Visual Components.</td></tr>
<tr><td colspan="2">Written by:
Halldor Arnarson</td><td colspan="2">Duration(days/weeks):
1 week</td></tr>
</table>

F Activity description: Robot Control

<table>
<tr><td colspan="3">Project title:
Digital twin with Visual Components</td><td>Date:
11.01.2019</td><td>Sign:
HA</td></tr>
<tr><td>Activity no:
2.1</td><td colspan="4">Activity name:
Robot Control</td></tr>
<tr><td colspan="5">Responsible:
Halldor Arnarson</td></tr>
<tr><td colspan="5">Task description/intention:
Using the computer to controller the robot</td></tr>
<tr><td colspan="5">Scope:
Robot control with computer.</td></tr>
<tr><td colspan="5">Method:
Useing python on the computer to control the robot.</td></tr>
<tr><td colspan="5">Dependency:
Activity 1.5</td></tr>
<tr><td colspan="5">Documentation/results:
The ability to control the robot from the computer.</td></tr>
<tr><td colspan="2">Written by:
Halldor Arnarson</td><td colspan="3">Duration(days/weeks):
4 weeks</td></tr>
</table>

G Activity description: Send position data

<table>
<tr><td colspan="2">Project title:
Digital twin with Visual Components</td><td>Date:
11.01.2019</td><td>Sign:
HA</td></tr>
<tr><td>Activity no:
2.2</td><td colspan="3">Activity name:
Send position data</td></tr>
<tr><td colspan="4">Responsible:
Halldor Arnarson</td></tr>
<tr><td colspan="4">Task description/intention:
Send position data from Visual Components to the OPC UA server.</td></tr>
<tr><td colspan="4">Scope:
Use Visual Components to send rotation/position data</td></tr>
<tr><td colspan="4">Method:
Send data from Visual Components to the python OPC UA server on the computer.</td></tr>
<tr><td colspan="4">Dependency:
Activity 2.1</td></tr>
<tr><td colspan="4">Documentation/results:
Can edit and control variablesin the OPC UA server fromVisual Components.</td></tr>
<tr><td colspan="2">Written by:
Halldor Arnarson</td><td colspan="2">Duration(days/weeks):
1 week</td></tr>
</table>

H Activity description: Two way digital twin

Project title: Digital twin with Visual Components	*Date:* 11.01.2019	*Sign:* HA
Activity no: 2.3	*Activity name:* Two way digital twin	
Responsible: Halldor Arnarson		
Task description/intention: Controlling the robot with Visual Components.		
Scope: Enable control of the robot with Visual Components.		
Method: Putting activity 2.1 and 2.2 together to control the robot.		
Dependency: Activity 2.1 and 2.2		
Documentation/results: The ability to control the robot from Visual Components.		
Written by: Halldor Arnarson	*Duration(days/weeks):* 1 week	

I Activity description: Literature study on robot control

Project title: Digital twin with Visual Components		*Date:* 11.01.2019	*Sign:* HA
Activity no: 3.1	*Activity name:* Literature study on robot control		
Responsible: Halldor Arnarson			
Task description/intention: Look at different ways a robot can be controlled.			
Scope: Looking for different ways a robot can be controlled through a simulation software like Visual Components.			
Method: Using UIT‘s library, Youtube and Google to find articles and videos on the subject.			
Dependency: Non			
Documentation/results: Find ideas on how to control the robot from Visual Component.			
Written by: Halldor Arnarson		*Duration(days/weeks):* 2 weeks	

J Activity description: Visual Components control options

<table>
<tr><td colspan="2">Project title:
Digital twin with Visual Components</td><td>Date:
11.01.2019</td><td>Sign:
HA</td></tr>
<tr><td>Activity no:
3.2</td><td colspan="3">Activity name:
Visual Components control options</td></tr>
<tr><td colspan="4">Responsible:
Halldor Arnarson</td></tr>
<tr><td colspan="4">Task description/intention:
Make a good control interface in Visual Compoents.</td></tr>
<tr><td colspan="4">Scope:
Control interface of the KUKA robot.</td></tr>
<tr><td colspan="4">Method:
Add a good control interface of the KUKA robot in Visual Components, using the information collected in activity 3.1.</td></tr>
<tr><td colspan="4">Dependency:
Activity 3.1</td></tr>
<tr><td colspan="4">Documentation/results:
Easier to program/control the robot from Visual Components.</td></tr>
<tr><td colspan="2">Written by:
Halldor Arnarson</td><td colspan="2">Duration(days/weeks):
4 weeks</td></tr>
</table>

K Activity description: Add features to server

<table>
<tr><td colspan="2">Project title:
Digital twin with Visual Components</td><td>Date:
11.01.2019</td><td>Sign:
HA</td></tr>
<tr><td>Activity no:
3.3</td><td colspan="3">Activity name:
Add features to server</td></tr>
<tr><td colspan="4">Responsible:
Halldor Arnarson</td></tr>
<tr><td colspan="4">Task description/intention:
Add features to the OPC UA server and improve up on it.</td></tr>
<tr><td colspan="4">Scope:
Find and add features to the OPC UA server.</td></tr>
<tr><td colspan="4">Method:
Use the Internet to find features that can be added.</td></tr>
<tr><td colspan="4">Dependency:
Non</td></tr>
<tr><td colspan="4">Documentation/results:
A more capable server.</td></tr>
<tr><td colspan="2">Written by:
Halldor Arnarson</td><td colspan="2">Duration(days/weeks):
6 weeks</td></tr>
</table>

L Activity description: Write report

<table>
<tr><td colspan="2">Project title:
Digital twin with Visual Components</td><td>Date:
11.01.2019</td><td>Sign:
HA</td></tr>
<tr><td>Activity no:
4.1</td><td colspan="3">Activity name:
Write report</td></tr>
<tr><td colspan="4">Responsible:
Halldor Arnarson</td></tr>
<tr><td colspan="4">Task description/intention:
Continuously writing the report throughout the project.</td></tr>
<tr><td colspan="4">Scope:
Preper a writen report of the work.</td></tr>
<tr><td colspan="4">Method:
Write a report using a computer.</td></tr>
<tr><td colspan="4">Dependency:
Non</td></tr>
<tr><td colspan="4">Documentation/results:
A written report</td></tr>
<tr><td colspan="2">Written by:
Halldor Arnarson</td><td colspan="2">Duration(days/weeks):
21 weeks</td></tr>
</table>

M Activity description: Presentations

<table>
<tr><td colspan="3">Project title:
Digital twin with Visual Components</td><td>Date:
11.01.2019</td><td>Sign:
HA</td></tr>
<tr><td>Activity no:
4.2</td><td colspan="4">Activity name:
Presentations</td></tr>
<tr><td colspan="5">Responsible:
Halldor Arnarson</td></tr>
<tr><td colspan="5">Task description/intention:
Make a presentation and prepare for it.</td></tr>
<tr><td colspan="5">Scope:
Preparing of status meting.</td></tr>
<tr><td colspan="5">Method:
Make a PowerPoint presentation and practise it.</td></tr>
<tr><td colspan="5">Dependency:
Earlier work</td></tr>
<tr><td colspan="5">Documentation/results:
A PowerPoint presentation.</td></tr>
<tr><td colspan="2">Written by:
Halldor Arnarson</td><td colspan="3">Duration(days/weeks):
5 weeks</td></tr>
</table>

N Activity description: Progress report

<table>
<tr><td colspan="2">Project title:
Digital twin with Visual Components</td><td>Date:
11.01.2019</td><td>Sign:
HA</td></tr>
<tr><td>Activity no:
4.3</td><td colspan="3">Activity name:
Progress report</td></tr>
<tr><td colspan="4">Responsible:
Halldor Arnarson</td></tr>
<tr><td colspan="4">Task description/intention:
Fill out progress report.</td></tr>
<tr><td colspan="4">Scope:
Fill out progress report.</td></tr>
<tr><td colspan="4">Method:
Fill out progress report using a computer.</td></tr>
<tr><td colspan="4">Dependency:
Earlier work</td></tr>
<tr><td colspan="4">Documentation/results:
A written progress report.</td></tr>
<tr><td colspan="2">Written by:
Halldor Arnarson</td><td colspan="2">Duration(days/weeks):
6 weeks</td></tr>
</table>

O Gantt diagram

GANTT CHART

PROJECT TITLE	**Digital twin with Visual Components**
PROJECT MANAGER	**Halldor Arnarson**

Traveling: **Between the 19-20 february i will go to China for exhange**

China

WBS NUMBER	TASK TITLE	START DATE	DUE DATE	Time on each activity	PCT OF TASK COMPLETE	1-6 jan	7-13 jan	14-20 jan	21-27 jan	28-31 jan 1-3 feb	4-10 feb	11-17 feb	18-24 feb	25-28 feb 1-3 mar	4-10 mar	11-17 mar	18-24 mar	25-31 mar	1-7 apr	8-14 apr	15-21 apr	22-28 apr	29-30 apr 1-5 may	6-12 may	13-19 may	20-26 may	27-31 may	
						Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11	Week 12	Week 13	Week 14	Week 15	Week 16	Week 17	Week 18	Week 19	Week 20	Week 21	Week 22	PROJECT END
1	**Phase 1**																											
1.1	Pre-Study report	Week 2	Week 3	26	100%		22	4																				
1.2	Model in Visual Components	Week 1	Week 2	20	0%	12	8																					
1.3	Visual Components and OPC UA connection	Week 3	Week 3	15	0%			15																				
1.4	Communication with KUKA controller and computer	Week 3	Week 4	73	0%			40	33																			
1.5	One way digital twin simulation	Week 4	Week 4	5	0%				5																			
2	**Phase 2**																											
2.1	Robot Control	Week 4	Week 7	151	0%				11	55	55	30																
2.2	Send position data	Week 7	Week 7	15	0%							15																
2.3	Two way digital twin	Week 7	Week 8	39	0%							9	30															
3	**Phase 3**																											
3.1	Literature study on robot control	Week 10	Week 11	40	0%										20	20												
3.2	Visual Components control options	Week 12	Week 15	120	0%												30	30	30	30								
3.2.1	Add features to server	Week 16	Week 21	170	0%																30	30	30	30	30	20		
4	**Work done throughout the project**																											
4.1	Write report	Week 2	Week 22	212	0%		6	1	1	1	1	1	1	40	20	10	5	5	5	5	5	5	5	5	10	25	55	
4.2	Presentations	Week 2	Week 22	28	0%		6					3					3			3					3		10	
4.3	Progress report	Week 7	Week 20	4	0%							1					1			1					1			
					Time used each week :	12	42	60	50	56	56	59	31	40	40	30	39	35	35	39	35	35	35	35	44	45	65	
					Total time Used on the project:	918																						