Engineering Design 4WBB0 2016-2017

Group nr: 78

Name	Student ID
Dorenbos, H.M.R.	0959563
Funnekotter, J.O.	0949863
Kruithof, C. C.	0968453
Oosterbaan, J.V.	0973875
Roberts J.	0969427

1. Group Effectivity

Technical role: mostly Jeroen and Haico
Managerial/social role: mostly Carlotta, Job and Josefine

Start and at Preliminary Design

Jeroen
Knows a lot about the technical subjects, says a lot of useful things and is nice to work with. Showed more initiative at the time of the preliminary design, than in the beginning. Especially when it is something he knows a lot about, he is speaking more and it always has value.

Haico
Appeared to be a bit quiet in the first meeting, but soon it turned out that he is very active and could contribute well in the programming part. Sometimes, he can focus a bit too much on his own ideas and not on the ideas of the group as a whole. Though, this determined mind set helps in guiding meetings and keeps them efficient.

Job
Is a really enthusiastic member of the group and has a lot to contribute. He listens to what others say and has quite some knowledge about various topics. Job also talks a lot, and sometimes a bit too much, by which others do not always get the chance to say as much. In general, he is a good 'bridge' between social, technical and design aspect.

Carlotta
Is a great value to the group and has a lot of ideas. Also good as a leader or manager; she likes to take the lead a bit, but not in a bad way. She has taken the initiative to do things, from the beginning. In addition, she knows more about materials and their costs. Carlotta is a nice person to work with, as well.

Josefine
Participates in the group enthusiastically and is a talker. She wants to learn a lot and wants to take the opportunity to do that, within this subject. Josefine tends to take the managerial role and could sometimes give others more space to talk.

End of the project

What can be absolutely said here, is that all group members have grown. Every member has learned to deal with weaknesses of others, and with strengths of others. What everybody thought was interesting to see, was that everyone had a different background, different experiences with working in groups and had had different roles in those groups.

In the beginning/middle of the project, there were some difficulties with the communication and the path of thoughts of different members. Haico was for instance a very organized member, who wanted to plan all steps in the process. Other members had difficulties with catching up on these expectations, but this was not Haico's fault. It was a fault made by all the members at the moment the planning was made, because Haico had made one and we blindly agreed on it.

After some discussions and talks, we found a way of planning that worked better. For Job, Josefine and Carlotta, this was roughly planning when the chassis had to be finished. Haico and Jeroen preferred to specifically plan when and where to build on the electronics/the program.

This kind of planning did work well for the group. The members that liked planning things in a rougher way, were willing to take the consequences of that (like working unexpectedly longer on an element). In the end, everyone planned things on his/her own way, but also with some core points/deadlines that everyone had to meet.

We have learned to respect the kind of planning of one each other, to accept with it and deal with it. We've all learned that everybody is different.

It was also a good idea to plan more meetings with the whole group together during the end phase of the project. In the beginning, the technical side (Jeroen and Haico) worked together, and the chassis/building side worked together (Job/Josefine/Carlotta). But to prevent miscommunication in the end of the process, we decided to plan meetings all together and to talk about expectations of each other, the meetings became more effective because we could quickly ask each other questions and everyone was better informed about how far everyone was in the process.

The personal end evaluations can be found in the appendix (Chapter 10).

2. Requirements, Preferences and Constraints

Requirements	
The robot has to be able to move.	It has to be able to move the whole distance during the time of the presentation of the robots.
It may not kill or harm any living creature.	There should not be any sharp edges or crushing the casualties
It needs to be able to move on a harsh/bumpy/wet underground.	The wheels need to be firm and big enough not to let the robot crash under the circumstances of the presentation.
The robot is waterproof.	There should be no water inside the robot, during the transportation.

Preferences	
It must rescue as most human beings as possible within a certain time.	When it can transport more than one person at a time, this goal is achieved.
The travel with our vehicle must be as comfortable as possible.	The casualties should not experience a lot of movement in the vehicle.
The life duration of our device must be as long as possible.	The robot is still able to be used after the presentation.
It should be as cheap as possible.	The bigger the gap between the expenses and €70, the better.
It is easy to recognize/see/notice.	There should be bright colors and reflective stickers.

Constraints	
The robot cannot cost more than €70,-.	All the used parts together, are less than €70,-.
The parts being used cannot have a complete function on itself.	Only parts with an incomplete function will be ordered.
It cannot be too big to fit through a 30x30 cm hole/gate.	The vehicle will be less than 30 cm wide and high, but a bit longer, to optimize the space.
It needs to be remotely controlled by Wi-Fi.	The equipment to control it by Wi-Fi will be used, and no other remote control technology.

3. Functions

The robot needs to be able to move from A to B. If the robot cannot move, it will not be able to search people who need to be rescued. It cannot bring them to a safer place or hospital either, so the task of the robot cannot be fulfilled.	• Rolling • Jumping • Driving • Swimming • Floating • Flying • Skiing • Underwater swimming, like a submarine • Skydiving (from a higher place like a mountain to a lower situated area) • Crawling • Hovering • Walking • Hopping • Slithering

The robot needs to be able to pick up and deliver casualties. In order to help people who cannot move themselves anymore, it is important that the robot has the ability to pick them up and deliver them at the hospital or a safe area.	• Treadmill • Tailgate • Lever • Boat • Arms • Crane • Scoop • Balloon • Net • Suction • Hook

The robot needs to be able to transport casualties. To rescue people, it is important that the transport is as safe and comfortable as possible. Especially when the circumstances outside are far from ideal, the robot has to be built in a way so it can transport people, without bringing them in greater danger.	• Container • Trunk • In the arms of the robot • Tow rope • Unsheltered on vehicle • Sheltered on vehicle • Live-support unit • Balloon (vehicle stays) • 'Escape pod' (Like the ones they use for space landings) • Alternate small remote controlled vehicle for fast delivery • Basket • One big bed

The robot needs to be able to recognize casualties. A robot is not a human being, so it needs to have a function to enable it to recognise the kind of 'things' it	• Dash cam • Body heat detection/IR camera • Movement recognition • Night vision camera

has to pick up. These are people, who need to be rescued. Often, when there is a humanitarian disaster, people are hidden underneath collapsed buildings, it is dark, or there is fire/smoke/dust in the air. In these situations, it is important that the robot can still recognise people.	• Facial recognition • Echography • Cellular signal detection • Voice recognition • Touch recognition of human hands/fingerprints • Breath recognition (reaction to the breath of people, which contains high amounts of CO_2 in relation to the environment.)
The robot needs to be able to stand out, so it can easily be found and recognized.	• Bright, reflective colors (red/yellow/orange/green) • Warning lamps • Reflection stickers (like the ones on safety jackets) • Music/alarm sounds • Leaving traces • Glow-in-the-dark paint • Searchlights • Light beacons • Audible messages ("Stay calm" etc.) • Vibrations through water/earth (can help deaf/sheltered people)

4. Concepts

Our choice for specialization, was the transport function. We chose this function, because we all had the most ideas and saw the most opportunities for it. After deciding to specialize in transportation, we all made a 'shitty' (low-fidelity) prototype, with which we could show what we came up with.

Prototype 1: Josefine

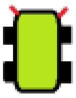
Circular transportation basket, with movable wall/windows on the side and a removable roof. People can be brought in from every side, it will stay dry while it rains, many options for openness, flexible shape.

Prototype 2: Carlotta

The idea is that the small white ball (with the person drawn on it) is inside the bigger, blue painted ball. This blue painted ball is made of quite hard plastic, that protects the person in it from falling objects. The person also wears a belt to prevent it from tumbling around inside the ball. The ball could be rolled towards other helping humans at the disaster site.

Prototype 3: Jeroen

To prevent the robot from running over wounded people my idea for a "shitty prototype" is using IR sensors. These sensors could sense nearby objects (up to 10 cm) and therefore help prevent collision with obstacles, walls and wounded humans. Some soft material could also be implemented on the outside of the robot so that if these sensors malfunction the severity collision will be diminished. The sensor would be able to detect objects at a close range of up to roughly 10 centimetres. The sensors would be mounted very low on the robot to be able to spot people lying on the ground.

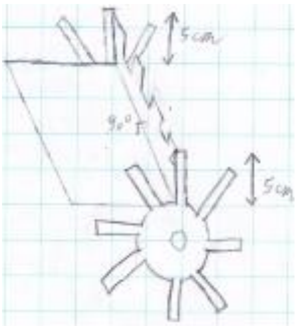


Prototype 4: Job

I created a prototype for two ideas for stable transport. One was a double hull as used in cruise ships, the other was a hammock system for transport of multiple casualties. The idea here being to minimize the effect of bumping into and traversing rough terrain. The double hull idea will be hard to implement with wheels or other standard forms of transport.

Prototype 5: Haico

I thought about fences around the passenger place. There are 4 fences (5cm) placed around the platform where the victims are located. One of the fences at the large side of the platform is replaced by a controllable fence as shown below. Gears at both side of the fence will be able to rotate this from 90 degrees to 270 degrees. The gears are connected with an iron pipe to a motor which can put a motion on the gears. The motor will be connected to the Arduino processor. Such that the motor will be controllable by two buttons at our panel, one button will send a signal to the Arduino processor to make the fence rotate to the left (closing the fence) and the other button will send a signal to the Arduino processor to make the fence rotate to the right (opening the fence). You must continuously press the button to make the motor move, when you release it, the motor will stop moving.



5. Preliminary Design

Introduction Preliminary Design

The preliminary design is split up in four parts. The description of the standard design (building of the vehicle, Wi-Fi communication, etc.), the description of the expanded design (when we have time and money left we will do this), the argumentation design (why we did choose this design) and satisfaction of the functions and RPC's (how the design satisfies the RPC's and why it satisfies the functions or why it does not satisfy the functions).

Description Standard Design

Visible Design

Our design will be a rectangular platform, with four wheels under it. Furthermore, the rectangular platform is surrounded with small walls at all sides of the platform (passengers can still look to the outside of the vehicle). Moreover, the platform will have a roof which is supported by pillars under it and the platform itself will be filled with soft material. Also at the smaller side of the platform we will have a remote controllable slide with at the end a curved part. And our vehicle will have a bright lamp at the roof and is entirely painted with bright colors. Last of all reflective stickers will be put at the outside of the walls of our vehicle.

Invisible Design

Motors will be located under the platform where one motor will control the slide, one motor will control the rotation of the wheels in the front and one motor which will control the wheels in the back. The motors will be connected to the Arduino processor which will also be located below the platform. Moreover, communication will be done encrypted using symmetric encryption, furthermore no extra error correction is implemented and no data is send back to the controller. Last of all the frequency of the communication will be fast.

Description Expanded Design

- Conveyor belt at the platform: A conveyor belt will be built at the platform which is filled with soft material. This conveyor belt is controllable by an extra motor.
- Infrared sensors at the walls: These sensors will detect walls and will stop the vehicle from moving when walls are in front of the vehicle.
- Rotating lamp: The lamp which is at top of the vehicle will be replaced by a rotating lamp (likewise a lighthouse do) where the lamp is put in a bulb with mirrors.
- Microphone: Also a microphone is placed at the top of the vehicle with a recorded voice message.

Argumentation Design

- Rectangular platform: We could also have chosen an oval platform, but a rectangular platform is far more usable because the space is used more efficiently and the vehicle can move more easily through small passages.
- Small walls: This is to prevent victims from falling of the vehicle, this will make the transport more comfortable for the victims and will therefore support our specialized function the transport function. Furthermore no large walls were chosen because then it would not be possible to look outside the vehicle for victims which is unpleasant for claustrophobic victims, also this will make it possible for victims to get fresh air and will make it possible for the victims to flee when the vehicle get stuck.
- Roof: This is done to protect the passengers from weather condition such as rain and hail, this will make the transport more comfortable and

thus will support our transport function.

- Soft floor: This is also done to make the travel as comfortable as possible for the victims. When lying at the platform the victims must not lay on a hard floor. But this is also done for the victims when they fall of the slide they will fall on a soft floor so their fall will not be very painful and therefore it will increase the comfort of the victims.
- Controllable slide: This will be easily to implement and will easily solve the problem to board victims because only one motor is needed to control the controllable slide.
- Curved part of the slide: The curved part is needed to easily pick up victims of the ground such that they will not fell of the slide when they are picked up.
- Bright lamp, bright colors and reflective stickers: A bright lamp at the top, bright colored vehicle and reflective stickers will make the vehicle more noticeable.
- Communication: We chose a symmetric encryption because it is safe, fast and easily to implement. Asymmetric encryption was more safe, but a lot slower, very hard to implement and took a lot of storage therefore symmetric encryption is the best option. Error correction is possible not needed because Wi-Fi will not be an unreliable channel. And the frequency must be fast because otherwise there will be a lot delay in sending commands to the vehicle.
- Conveyor belt at the platform: This will make it possible to comfortably transport multiple victims per ride.
- Infrared sensors at the walls: Through this system the vehicle will not bump and therefore the ride will be more comfortable for the victims, which will also support our transport function.
- Rotating lamp, microphone: A rotating lamp and a microphone will make the vehicle more noticeable.

Satisfaction of the RPC's and the Functions

Functions:

- Movement and moving from A to B is achieved by the wheels and the motors which will rotate the wheels.
- Causality boarding is achieved by the controllable slide.
- Causality delivery is not achieved by our design however by having an open platform we can take the victims out of the vehicle with our hands at the rescue post.
- Causality recognition is not achieved by our design because it is very hard to implement, however with our eyes we can see the victims and so we can move the vehicle towards the victims.
- Causality transport is achieved by the soft platform, small walls and roof.
- Signal function is achieved by the (rotating) lamp, microphone, bright colored vehicle and reflective stickers.

Requirements:

- The robot has to be able to move. This requirement is achieved by having wheels which will rotate when the motor starts running.
- It may not kill or harm any living creature. This requirement is met by the fact that everything is controllable by us and by the infrared sensors which prevent bumping.
- It needs to be remotely controlled by Wi-Fi. This requirement achieved because we have an implemented communication function.
- It needs to be able to move on a harsh/bumpy/wet underground. This requirement is not yet achieved however it is not excluded from being achieved. It will be clear in the detailed design phase if this requirement is achieved or not.
- The robot is waterproof. This requirement is achieved partially by having a roof.

- It needs to be able to move on a harsh/bumpy/wet underground. This requirement is not yet met but also not excluded from being met, it will be clear if these requirements are met at the detailed design.

Preferences:

- It must rescue as most humans as possible within a certain time. A driving vehicle is easily controllable therefore it can drive very fast to victims, furthermore by using a conveyor belt at our platform we can transport more victims at the same time which make it possible to rescue many humans within a certain time.
- The travel with our vehicle must be as comfortable as possible. The travel is definitely very comfortable due to the soft floor, the roof, the small walls and the bump sensors.
- The life duration of our device must be as long as possible. Because our device is not very complex and has not many parts, there will not be much parts which can crash. Therefore, the life duration will be longer.
- It should be as cheap as possible. We have divided our design in two parts, one standard and one expanded and when we see that the standard one already is very expensive we will not continue with the expanded design. Therefore, we can reduce the cost of our machine, furthermore our machine does not need very expensive components, thus our design will be cheap.
- It is easy to recognize/see/notice. The vehicle is easily recognizable due to the lamp, microphone, brightly colored vehicle and the reflective stickers.

Constraints:

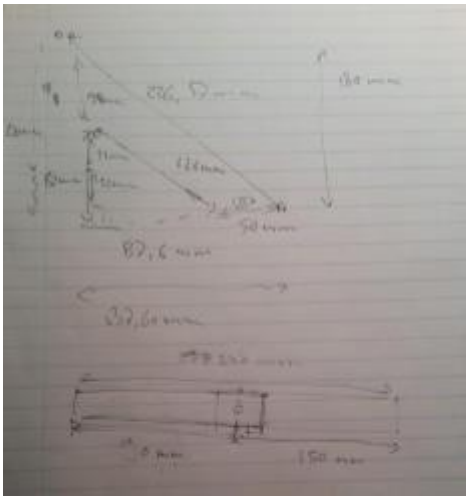
- The robot cannot cost more than €70,-. For this hold the same argument as for the preference as cheap as possible. With our standard design our cost will definitely stay below €70,-.
- The parts being used cannot have a complete function on itself. This is not yet violated because we do not use other machines than the Arduino processor in our design.
- It cannot be too big to fit through a 30x30 cm hole/gate. This is not violated because our vehicle only consists of a platform and a roof which can fit in a 30 by 30 cm hole/gate.

6a. Detailing

Body

A small, but critical detail in our design is the small block attached to the smaller arm of the tailgate-system. It prevents the bigger arm to fall over and to scratch along the floor. This is critical, because the robot could meet more obstacles when the arm just lays on the floor rather than when it is stopped by that little block and stays, due to that, in the air. When the servo rotates, the longer arm touches the floor only when needed, namely in the case of pulling a victim into the mobile. The block is placed directly after the hinge-point of the 2 arms, the first possibility where the longer arm can be stopped.
Above, the arms of the tailgate system are shown. The little block can be seen at the hinge point. The little wooden ball is for aesthetic and functional reasons, it is pleasing that no sharp metal ends are on the outside of the robot, and it looks nice. The block is 3 mm thick, just as thick as the arms themselves, to be able to effectively stop the long arm.

The length of the arms is calculated with the Pythagoras formula, and are placed near the entrance of the robot because it would cost more material and more force to lift up the other elements of the system if they would be placed further away.



The calculations

The second detail would be the placement of the VIVAK into the window frames of the robot. It is not just placed to the frame on the outside or inside, but inside the cut-out, in the middle. This is nicer to look at, it looks like there is taken care of the placement of this element. The sizes of the rectangular window frame were measured (260*32MM) and the VIVAK (1MM thick = thick enough for this design) was cut with a saw. The size of the VIVAK was exactly the same size of the window frame, but on 1 side with a cut-out for the servo wheel. Because the window frame was laser-cut, it had no bumps and was very straight, the VIVAK fit perfectly because of that and looked good.

The window frame cut-out, directly above the black servo wheel.

In the end, the costs of our other parts were too high to include VIVAK in the design. That is why the robot does not have VIVAK windows.

Another detail is the attachment of one of the two arms to the robot. One arm is attached to a servo, so it will be motorized. The other arm is connected to the robot with an easy-rotating hinge with as less as friction as possible. If one of the two arms is rotated by the servo, the other arm, connected to the first arm via the scraper at the end of the two arms, will rotate. The location of the hinge is on the opposite side of the robot, without the servo, but on the same height as where the servo is on the other side. The servo and the hinge are thus on the same height, but at opposite places. Thanks to the hinge being quite loose, the arm will move easily with the motorized arm.

When the chassis of the robot arrived from the laser-cutter, further changes could be done to it with the help of skills in combination with drilling and sawing machines. With the help of a 4MM drill head, we made small holes for the axes of the conveyor belt and we also made a plateau for the servo of the tailgate, so that it could reach the small gear at the tailgate itself, so that both gears could grip into each other perfectly. we drilled out the holes inside the gears very carefully, also with a drill of 4MM. The gear for the tailgate servo had to be drilled out with the drills 4.6MM, 4.8MM,5.2MM, 5.4MM and 5.6MM (the hole in the middle “grew” bigger in this way) to prevent the delicate gear from breaking by directly using the 5.6MM drill. The chassis has been painted bright pink, to make it outstanding in the disaster area during the end-event. Our philosophy about this was that a rescue robot should not disappear between its surroundings, it should stand out and be visible for other operating areas in the field, for human units as well as for technical robotic units. If something goes wrong during the operation, these units should be able to directly spot the robot.

Another problem that occurred during the process, was that the servo of the tailgate system (the one on the plateau) had a lot of power and would be dragged towards the tailgate if it was active. This was not a desired effect, therefore small wooden blocks were sawed with exactly the sizes of the servo, to glue them directly around the servo. The servo could release its powers into these blocks, which could transport the powers into the chassis. The servo was not dragged into the tailgate anymore because of the smart placement of these blocks.

The motors of the wheels had to be connected to the bottom plate of the chassis. Different materials were tried out like tape, duct tape, wood glue and hot glue, but only hot glue was strong enough to connect the motors to the chassis. Tape was too weak to hold the motors in their place and did also took away the desired current through the motors of the wheels, the tape did conduct the current. Wood glue was also not strong enough and the duct tape was strong, but did also take away the current from its desired path.

The small 5V pin-wires that conducted energy from the Arduino or the power bank to the wheels, had therefore also be connected to the motors without tape. But also without glue, because this could harm the wires. Therefore, we used rubber elastic bands to connect the wires to the motors, because of their strong and non-conducting properties.

Programming

Easy readable/adaptable code
In the Arduino code a lot of comments and constants are used. The comments were written to make the code readable for myself (the programmer) and the other members of the group. It is important that the code is readable for yourself and others, because you can remember more easily what you did exactly. Others then know what decisions are made in the code and can say if they agree with the code or not, without understanding the code itself.

Also in the Arduino code a lot of constants are used, these constants are used to make the code more adaptable. When, for example, the keys must be redefined it can be very easily changed by changing some constants and then in the entire code another key is used. Furthermore, angles of servos and speed of servos can be easily changed by changing some servo constants, which will make it possible to test the servos more easily without changing a lot of code. And last of all pin integers are also constants, this will make it much easier to change the pins when it is hard to do that electronical.

Delay instead of a timer
We use a delay instead of a timer. A delay is not a very decent solution, because during the delay the code does not run, which means that the CPU of our Arduino is not optimal used. However, it not easy to run our code without a delay. Because the "TimerOne.h" which can use the TimerOne on the Arduino conflict with the <Servo.h> library. Due to that both the "TimerOne.h" and <Servo.h> library use both the TimerOne. So we could solve this with:

- *Don not use the <Servo.h> library*, then we have to write the servo angles with analogWrite(), or digitalWrite() command. But these commands do behave unpredictable, therefore it was not possible to remove the <Servo.h> library out the code.
- *Use another Timer,* there were two other timers which could be used: TimerZero or TimerTwo. But TimerZero is used for delays and therefore behave in the same way as a delay, hence does not count down when other lines of the code are runned. Also TimerTwo is very hard to implement and we doubt if it is really to implement that timer on our Arduino.
- *Multithreading,* with this method we could have runned a separate thread which is delayed at the end to simulate a timer. When we use this the CPU is used optimal, because there will be always a running thread. However, it is not possible to implement this, cause our Arduino only has one core.

- *Don not use a timer or delay at all,* but then the code runs very fast and will call every time the checkKeyboardInput() void, which will return most of the time that no keys are pressed because if a key is continuously pressed, only every 80 milliseconds a character is sent to the Arduino.

Thus we used a delay() at the end of the script, because the other solutions didn't solve our problem.

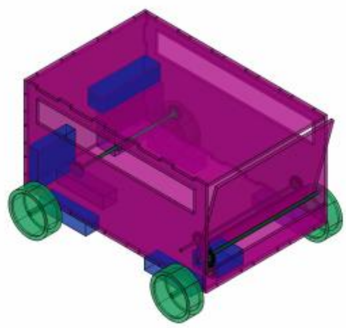
Starting Password

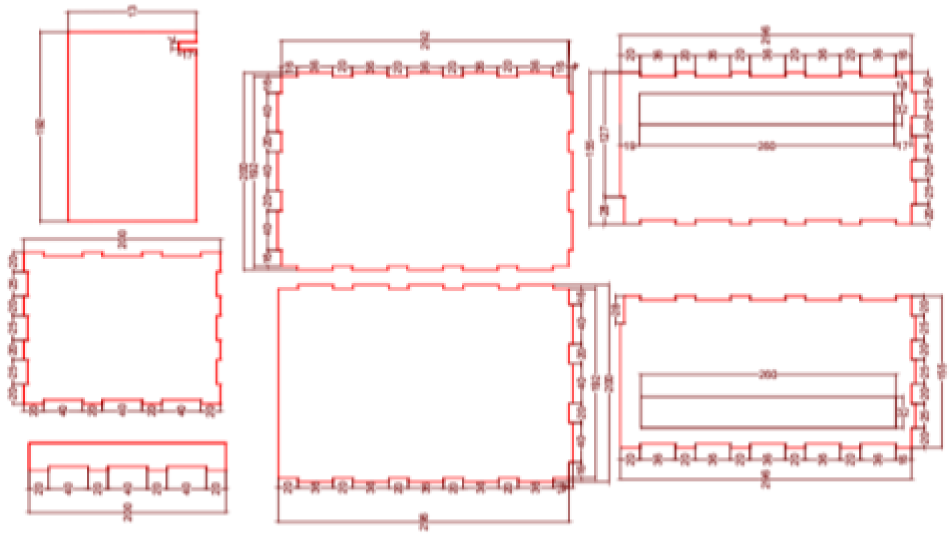
When connecting with the Arduino using "4WBB0-control.exe", you have to send the password characters first through "4WBB0-control.exe" to make it possible to control machine. This is done because otherwise:

- It would be possible for a hacker to drive our vehicle against some wall, which will damage our vehicle unnecessary.
- Our vehicle moves randomly when starting the code. B we suspect our code to receive random characters by Wi-Fi when not ready connected through the "4WBB0-control.exe" program, we will use a password before running our machine. Therefore, it is not possible to randomly move by sending characters, because first the password must be sent with random characters, which is unlikely to happen.

6b. Assembly

The fittest and detailed design:

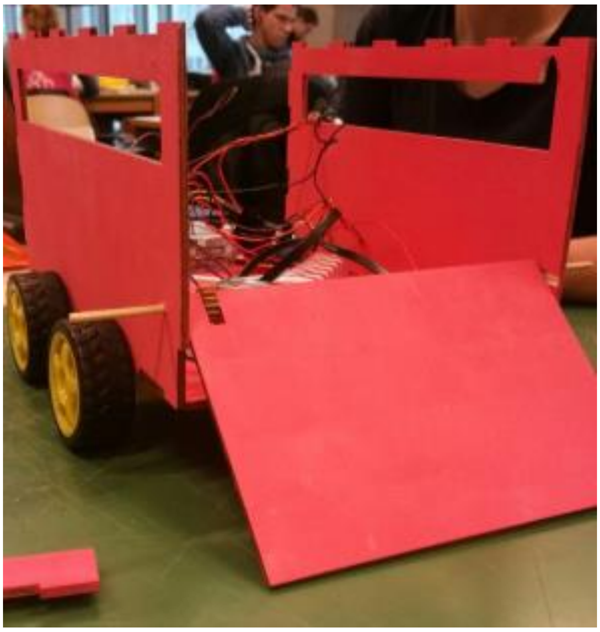




The fittest was done in AutoCAD. The blue elements in the purple vehicle show where the diverse electronics are attached. The power bank, Arduino and the broad board can be placed to the back wall or beneath the conveyor belt. The motors of the wheels are placed to the sides of the walls. There is room for the servo, beneath the tailgate. In this way, the “blades” of the servo can be attached to the tailgate and in this way, motorize the tailgate. Exact measurements of each piece are imaged below the animation with the purple vehicle.

The detailed design can also be read from these fittest drawings. The used materials are plywood for the “walls”, VIVAK for the windows, wood glue to connect the plywood, MDF for the tailgate system, and screws to function as hinges in the tailgate system. The wheels are basic rubber wheels. A servo is placed beneath 20 MM high plywood for the tailgate itself, the servo for the arms is placed on the right side of the robot (top view). The Arduino/power bank/all bigger parts are placed in the back, to the wall or to the ceiling of the robot. The motors of the wheels are attached to the floor of the robot (beneath it), fixated with glue/tape. Pictures of the detailed design are presented on the next page.

The pictures show the setup of the robot (detailed design), the cuts are really clear and every small part fits into each other. The arms of the tailgate systems are laying in front of the detailed design in the first pictures.

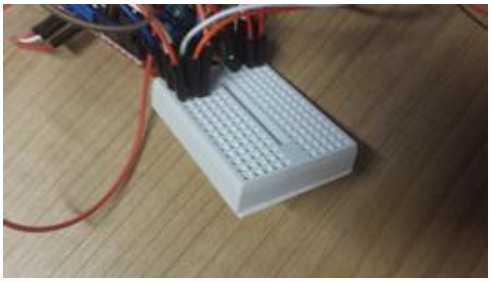


This is what the detailed/final design looks like after painting and attaching wheels. The electronics were not applied yet at the moment the photo was taken, but were just put onto the bottom plate of the robot to see if everything would work as expected. The pink color of the robot was not just for aesthetic reasons, but also for eye catching reasons. The robot needs to stand out in the area where it is operating, so that it can be easily spotted by humans or other operating units. In combination with the bright yellow wheels, a remarkable color combination is ensured.

Pink is also a color that seems to have a soothing effect. This is an extra positive point for the victims, they might feel calmer when being rescued.

Breadboard Test and Test Parts

Breadboard
The breadboard has been successfully tested. It works the following way: every column of four slots on the breadboard can be used to put a wire in and then all other three slots will get the same electricity as the one wire is sending to the breadboard. So putting two wires in two slots of the same column will connect the two wires. It is tested by connecting the wire of a component which needs electricity with the breadboard and by connecting a wire from the breadboard to the power source, which was the 6 voltage output of the Arduino.



Big wheels with motors
Also all the big wheels with motors have been successfully tested. It works pretty simple, the motor (that yellow thing) has two copper parts at its backside. You must connect it by putting a wire through one copper part and connect the other part of the wire with the 6 voltage output from the Arduino (the wire will not release automatically when you put the end of the wire deeper through the copper part). Moreover, you have to connect the other copper part with a wire which is also connected to the ground input of the Arduino. If you do this the motor will start spinning. And if you switch the wires from copper part it will spin in reversed order. Thus switching the current will make the motor spinning in reversed order. When testing the motor, it is important to keep the wheel away from the ground otherwise it starts driving.

The purchased servo
The servo which we bought also has been successfully tested. It must be connected with a ground pin on the Arduino, the 6-volt output pin on the Arduino and some analog pin on the Arduino. To control the servo, you must use the Arduino library <Servo.h>. It is not clever to do it with analog write because its angle in degree’s is not the same as the analog write signal, so you have to make more calculations when you do not use the <Servo.h> library. Therefore, you must use the <Servo.h> library which does not have to be downloaded. To control the servo, you have to:

1. Declare a global servo variable: Servo s;
2. Attach the analog pin to it in the setup(): s.attach(6);
3. Use s.write(angle) to change the angle of the servo.

When changing the angle with s.write(angle) keep in mind that the position is changed to that absolute value of the angle value. So s.write(angle) does not change its position in a relative way. Also keep in mind that it take some time to execute s.write(angle).

The included servos

The included servos also have been successfully tested. It works the same way as the bought servo. However, it can be controlled in two ways:

1. It has one servo that make it possible to rotate upwards towards the roof or rotate downwards towards the floor.
2. It has another servo that make it possible to rotate from left to right.

When using this included servos it is important that both servos are declared with a separate variable, for instance:

Servo s;-

Also s.attach(pin) and t.attach(pin2) must use a different pin.

The camera

The camera has been successfully tested. You can make it work by putting the USB-cable from camera into the Arduino USB port. When doing this you must be sure that the USB-cable from the computer to the Arduino is disconnected if it can send data (if it cannot send data then it does not matter). Moreover, in the code you must activate the Wi-Fi connection, you can do this with: Serial1.begin(115200); And then it is possible without adding further code to look at the camera screen with Wi-Fi by using "4WBB0-control.exe" and then looking at the stream screen.

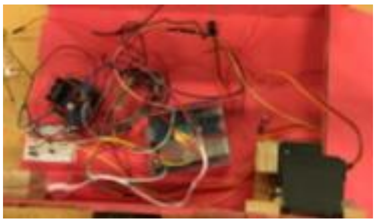
7. Realization

Bill of Materials

Part	Amount	Total price [€]
Pulse power bank (5V, 1A)	2	7.98
Wood (laser cut)	1	0.5
Micro-USB cables	1	2.28
Small wooden gear	6	5.53
Chain belt	1	6.99
Small DC motor 3-6V	1	1.25
Rechargeable batteries (18650, 5000mAh, 3.7V)	2	8.52
Battery case (18650)	1	1.38
DC-DC Buck converter LM2596 3A	1	3
MG996R Towerpro servo	1	8.28
Actuator kit (4x DC motor with plastic tire + MG996R Towerpro servo)	1	15.48
NPN transistor	1	0.25
Battery charger (18640)	1	4.24
Electrical wires	20	1
Glue		0
Power bank (5V, 1A)	1	0
L298N DC motor controller	1	0
Camera holder with servos	1	0
USB camera	1	0
Arduino Yun	1	0
		66.68

Table of manufacturing techniques

Part of the vehicle	Manufacturing technique
Outer parts of the vehicle (sides, bottom, roof, tailgate)  	For the outside, we used laser cutting. The parts are glued together with wood glue and painted with pink paint from Vertigo.
Slider 	The wooden parts of the slider are sawn out of a wooden plane, glued with wood glue and attached to the vehicle at the upper servo; on the other side with a nail. In order to keep this in place, we put a little wooden ball on the inside, around the nail. At the turning point, nails are used as well, and a small wooden ball is glued on, in order to prevent sharp parts sticking out.

Motors with wheels 	These are glued on the bottom of the vehicle with hot glue; the wheels can be clicked onto the motors. In order to keep the wires in place, elastic bands from Vertigo are used.
Tailgate 	The tailgate is attached to a thin, wooden stick (with wood glue), what sticks through the sides of the robot. On the stick, we glued a wooden gear, as well. The gear falls into another gear, which is attached to the lower servo.
Servos 	The upper servo is glued to the inner side of the wall, with hot glue. Both of the servos are sandwiched between small wooden blocks, to keep them in place. These are attached to the robot with wood glue.
Conveyor belt 	In the back, at the right side, there is a small wooden block, in which the motor for the belt is placed. This motor is glued to one of the two wooden sticks. On each of these sticks, two wooden gears are attached with wood glue. Two chains are wrapped around these four gears. A bit of fabric from an old T-Shirt is stitched on the chain and on the bottom, in the middle, a small wooden stick is glued.
Camera 	We drilled four small holes, in order to put screws through the holes in the camera stand. They turned out to be too big for screws, so in the end, we attached it with wires and tired. These are tightly wrapped around the stand and through the drilled holes. An elastic band is used to keep the camera in its stand.
Electronics 	The wires, bread boards, Arduino, batteries, power banks and motor controller are all situated underneath the conveyor belt, on the floor of the robot.

8. Test Plan

Describe and explain a set of experiments to test several critical functions of your design.
Present the results of the tests and compare the results with the original RPC's.

Not tested RPC's

There are some RPC's which will not be tested. All the constraints for example will not be tested, because these are constraints that were not violated during the project. Also the following requirements and preferences will not be tested:

- *It may not kill or harm any living creature*, do not confuse this with testing whether our machine is very comfortable, because this will be tested. However, the requirement "*It may not kill or harm any living creature*" will not be tested because we assume that driving over a victim is the only way to kill living creatures. We assume this because our machine hasn't implemented a gun or anything else that can kill a living creature. Also we succeed in this assumption by controlling the vehicle ourselves. Therefore, we have met this requirement. Moreover, we do not test this because it is not very ethical to test this, cause trying to harm or kill any living creature with your machine should be the best way to test this requirement, however this isn't very ethical to do.
- *The life duration of our device must be as long as possible,* we will not test this because this cannot be tested but only reasoned, this is due to the fact that we only use our vehicle during this quartile. We do not use our vehicle over a period of 5 years or longer. Therefore, this can only be reasoned. This is simply done by looking at the construction of our machine and the materials used for our machine. With this information and statistical data, you can conclude if the life duration of the vehicle is very long or not.
- *It should be as cheap as possible,* this will also not be tested because simply we already know how much our machine cost, due to the fact that we have a list of all things we have bought. Therefore, this will not be tested.

Passed or not tests

These test will be called passed or not tests because the test result will be a boolean variable (true or false), because the test is either passed or not. The following things will be tested (first the test name and then which requirements and preferences it will test):

- Moving Shortly Forward - *The robot has to be able to move, it must rescue as most human beings as possible within a certain time.*
  We do this by pressing the "8" key until the machine has move forward over a distance of approximately 1 decimetre forward. This test has passed if the machine does not need any help from us to move over a distance over approximately 1 decimetre forward. This test will be done to check if the vehicle is able to move during the entire presentation, because if it cannot move 1 decimetre forward it will not be possible to move during the entire presentation. Also if it cannot move 1 decimetre forward for the vehicle it is not possible rescue many human beings within a certain time.

- Rotating Shortly to the Left - *The robot has to be able to move, it*

must rescue as most human beings as possible within a certain time.
We do this by pressing the "4" key until the machine has rotated left over an angle of approximately 90°. This test has passed if the machine does not need any help from us to rotate over an angle of approximately 90°. This test will be done to check if the vehicle is able to move during the entire presentation, because if it cannot rotate over an angle of 90° it will not be possible to move during the entire presentation. Also if it cannot rotate over an angle of 90° it is not possible rescue many human beings within a certain time.

- Moving Shortly Backward - *The robot has to be able to move, It must rescue as most human beings as possible within a certain time.*
  This will be tested the same way as moving shortly forward, but instead of pressing the "8" key the "2" key is pressed until the machine has moved over a distance of approximately 1 decimetre backward.
- Rotating Shortly to the Right - *The robot has to be able to move, it must rescue as most human beings as possible within a certain time.*
  This will be tested the same way as rotating shortly to the left, but instead of pressing the "4" key the "2" key is pressed until the machine has rotated right over an angle of approximately 90°.
- Lift a casualty in the Vehicle - *It must rescue as most human beings as possible within a certain time.*
  Extra requirements:
    - A casualty

  We put a casualty in front of the vehicle at a distance of approximately 20 cm and then drive the vehicle a bit closer to the casualty and then try to lift the casualty inside the vehicle. If the casualty ends in the vehicle without touching the casualty or the vehicle ourselves the test will pass. This test is done because if the vehicle cannot lift a casualty in the vehicle it will not be possible to rescue as most human beings as possible.
- Camera Test – *It must rescue as most human beings as possible within a certain time.*
  The robot can use the camera to look in left, right, forward and down directions. The camera sends a video stream to the computer so casualties can be seen. This test is needed because if the camera does not move or does not send a video stream we will not be able to see the victims and therefore are not able to rescue as most human beings as possible within a certain time.

Measurements tests

These test will be called measurements test because the result of these test will be a single number or a set of numbers. The following things will be tested (first the test name and then which requirements and preferences it will test):

- Outside driving - *It needs to be able to move on a harsh/bumpy/wet underground*
  Extra requirements:
    - Stop Watch

  This test will be done outside on the grass. One of us will control the vehicle and try to move (forward, backward, left and right) the vehicle as long as possible on the grass. The other will have the stop watch and start the stop watch when the vehicle moves and stops the time when the vehicle get stuck, which is considered when the vehicle does not move for 10 seconds. Also it will stop the stop watch if the vehicle succeeds in moving for 5 minutes. This test will be executed a few times and the time until the vehicle get stuck will be written in a table. This test will be done to check if it is possible to move on harsh/bumpy/wet underground, because a terrain with grass can be seen as harsh/bumpy/wet ground.

- Maximal speed - *It must rescue as most human beings as possible within a certain time.*
  Extra requirements:
    - Signs (tape or any other visible thing to put on the ground)
    - Stop Watch
    - Tape-measure ("Rolmaat" in dutch)

  This test starts by putting the vehicle at some start place (which will be made visible with a sign). Then one of us continuously press the "8" key and the other will watch at which distance the vehicle get at maximal speed and will put at a safe distance from that point a second sign (where the vehicle is definitely at maximal speed). Moreover, the third sign is placed with a distance to the second sign which will be written in a table. Then one of us will put the vehicle at the start position and will continuously press the "8" key. Another one will have a stop watch and start the stop watch when the vehicle has passed the second sign and will stop the stop watch when the vehicle has passed the third sign. When the third sign has been passed the one that presses the "8" key may release the "8" key. Also the measured time is written in the table. This test will be done to check what the maximal speed of the vehicle is, which is of course important to know because a vehicle that can move faster can save more human beings as possible within a certain time.
- Uncomfortable brake acceleration - *The travel with our vehicle must be as comfortable as possible.*
  Extra requirements:
    - A program which can simulate key presses
    - 2 Casualties

  The program will be written first which presses the "8" key until a certain time which we call "A". Then with a frequency of "B" milliseconds the "2" key is pressed. "A" will be based on the time when the vehicle reaches its maximal speed and "B" will be written in a table. During this test 2 casualties will be put in the vehicle, one in the front of the vehicle and the next one at the distance which we will use for our vehicle. When the casualties collide or some casualty fell of the vehicle or something else uncomfortable happens we will write false next to the "B" value. When nothing uncomfortable happens we will write true next to the "B" value. This test is needed because we must know what the lowest frequency of pressing the "2" key (brake key) is to make the trip still comfortable for the victims in our vehicle.
- Maximal lift power - *It must rescue as most human beings as possible within a certain time.*
  Extra requirements:
    - A set with weights where the weights have a variable mass amount.

  This test can be skipped if we do not have the extra requirements. During this test we put weights until we have a mass of "A" at our slide which must be lifted by our vehicle. If the vehicle manages to lift that amount of mass, then we write next to "A" in the table true. If it fails to lift that amount of mass, then we write false next to "A" in the table. This test is needed because we must be sure that even the heaviest human can be lifted by our vehicle otherwise it isn't possible to rescue as many human beings within a certain time.

Subjective tests

These test will be called subjective tests because a criterion whether the test has been passed or not cannot be easily determined. Instead of determine this criterion we will look what happens during the test and subjectively determine if the test has passed or not. The following things will be tested (first the test name and then which requirements and preferences it will test):

- Vehicle visible in the dark - *It is easy to recognize/see/notice.*
  Extra requirements:
    - A flashlight
    - A dark room (optional)

  This test will be executed in a dark room or during the night. The vehicle will be placed on the ground and we check with and without using the flashlight if the vehicle is easy to recognize or not. This is tested because the vehicle must be easy recognizable.
- Comfortable trip for a casualty - *The travel with our vehicle must be as comfortable as possible.*
  Extra requirements:
    - Multiple casualties

  During this test we feel the floor of our vehicle and try to lift multiple casualties in our vehicle which are placed at a distance of approximately 50 cm from the vehicle. We check if the floor of our vehicle is comfortable and if the lifting and driving of the casualties is done in a comfortable way. This is tested because the travel with our vehicle must be as comfortable as possible.

Test Results

- Moving Shortly Forward: Passed, no problems happened during this test.
- Rotating Shortly to the Left: Passed, no problems happened during this test.
- Moving Shortly Backward: Passed, however the vehicle moves slightly to the right when driving backwards.
- Rotating Shortly to the Right: Passed, no problems happened during this test.
- Lift a casualty in the Vehicle: Passed, however instead of a casualty a thick marker is used.
- Camera Test: Passed, no problems happened during this test.
- Outside driving: Failed to do this test, because with an Ethernet cable this test is not possible and we had only the rescue network Wi-Fi in Gemini and Auditorium, so it wasn't possible to control the vehicle outside in the grass.
- Maximal speed: We found the following results:

Test number	Distance (cm)	Time (s)
1	80	1,1
2	180	2,3
3	180	2,45

The distance in this test is measured with a ruler instead of a tape-measure. The time is measured by a stop watch. We took test 3

to determine the maximal speed, because the time in test 3 is determined by 2 stopwatches. To calculate the average speed, we know that if we calculate the average speed (which was the maximal speed) of test 3 we will get: m/s. Which is 2,6 km/h. • Uncomfortable brake acceleration: Failed to do this test, because the conveyor belt wasn't implemented in our vehicle at that moment. • Maximal lift power: This test is skipped, because we didn't have the extra requirements. • Vehicle visible in the dark: Failed, this test is executed but the vehicle wasn't visible in the dark and also hard to see when using a flashlight. • Comfortable trip for a casualty: Failed to do this test, because the conveyor belt wasn't implemented in our vehicle at that moment.

9. Design evaluation

The original assignment for this subject, was to create a rescue robot with a specialization function, that would be able to be remotely controlled through a Wi-Fi connection. Our design meets the requirements that were set; we did not pay more than 70 euros for our robot, it fits in the context of a search-and-rescue-mission, it is able to move, et cetera.

Our most critical part in the design procedure is probably that at some points, we were constantly discussing the collaboration instead of the design. There was a period of time when it was just about talking and less building/doing, which did not work that well for us. We had a good meeting about the collaboration, after which the it improved. Had we expressed our expectations and set clearer goals earlier, we would probably have achieved our goals sooner and created a more advanced design. This is something we can take with us in future (group) projects.

A thing that we could have done better, is starting to create and test our conveyor belt earlier. We made the transportation our specialization, which means it is one of the most important aspects of our design. It would have been smart to test out if this idea would work in earlier stages than during the building process, because it turned out to be more difficult than expected. We could have started testing it out after making the shitty prototype, so we would have had more time to improve it and make it work.

The second aspect of our design that should have been thought of more, is the tailgate we made. When we started assembling our robot, it turned out that it was too steep and we could not pick up casualties with the tailgate alone. It would have been better if we would have thought of this in the conceptualizing phase already, like with the conveyor belt. Luckily, we found a solution to our problem; the slider. This definitely improved the design and made it easier to pick up casualties. It can even be said that picking up casualties eventually was more our specialization than the conveyor belt.

Something that has been a problem throughout the whole assembling/building the electronics phase, was that connections broke a lot. It cost a lot of time to find out where it went wrong and connect all the wires again. We could have thought about a better division of the electronics in the robot, instead of putting everything on the same level and without a clear structure. This is something for both the conceptualizing and assembling phase.

10. Appendix

Personal end evaluations (1. Group effectivity):

Jeroen

I improved on communication and taking initiative. I tried to this by being more active in the meetings. I took initiative when working on the electronic parts. However, this was easier for then taking initiative in communication or planning meetings. However, this was not very necessary in the group because other people showed a lot of initiative in communication and planning. I tried to communicate as much as I thought was necessary to the other member of the group not working on the electronics about how they worked.

Josefine

In the beginning of this project, it was mostly a discussion about our design. As I am pretty good at managing a team and like to take the lead a bit, this went well for me. Though, I missed feeling productive and useful. The other team members had a more specific expertise area than I did. As an Industrial Designer, I am an all-rounder. I know a bit about everything, but there is not a specific thing I am excellent at. In the past two or three weeks, we finally started building the robot and creating. Finally, I could learn a bit about programming and I was informed about how the code worked. I even helped programming a bit, and this really brought it all together for me. I am glad that we had a couple of meetings all together, or at least with a smaller group, to work on it and use all of our capacities combined. I could saw, sand, paint; all those practical things. At the same time, I was able to stay updated about the electronics.

Another thing I could contribute to the group, was opening up about our expectations. In my opinion, it is very important to other people what you expect from them, and let them tell what they expect from you. Communication is number one priority when collaboration. In our group, there were a few small things that evoked frustration and confusion. I opened up the conversation about this, and I think it really helped us as a group.

I contributed to the report as well; I wrote a big part of the evaluations and checked and assembled everything in the end.

Haico

My opinion about the other members of the group cannot easily be determined because we have often worked in two separate teams. But I can say that Jeroen also did a lot during this project, because buying and constructing the electronics wasn’t very easy, but he managed to do it. Therefore, I do not think it is fair if he gets a lower grade than we get. For Job, Carlotta and Josefine it is harder for me to determine an opinion, because they had separate meetings for the physical design, however I heard that Job spend the most time on the physical design and I also get the idea that Carlotta spent much time on this project, because I saw her more often than Josefine and Job when we had to discuss something. Last but not least it was sadly that the other members of my team did not want to have deadlines. This is sadly because deadlines are useful and needed for the group to check if you are lacking behind or not.

When I look back to my midterm reflection I will see that they match a lot, however during the midterm reflection I hadn’t expected that I would have helped Jeroen a lot with the testing of the electronics.

Carlotta
The interesting thing about the process was, in my opinion, the learning to work together with totally different people. Although there were struggles, I knew that these struggles could be overcome, because I have got lots of group projects in my own faculty and there is always some point where a big project seems to go totally wrong. (Because people do not know what to do anymore, lack of inspiration, busy with other things, etc.). In the end, a project will turn out fine, even when not everything works.

I have learned basic skills in electronics and programming, and was interested in how a practical side (building a chassis) and making it come "alive" with electronics came together. I have learned how the motors of wheels were connected to the Arduino via 5Vpin-cables, how they were connected to power sources, how many degrees a servo could turn around, how much weight a servo could carry (1cm away from the centre point=10kg, 2cm=9kg…. 10cm=1kg, so the ratio between arm and force), and what the different colors of cables did stand for coming from the servo and where they were connected to.

I was also happy that I could bring my (expected) input into the design. Manufacturing and assembling the chassis was the biggest task among the things that I had to do. And it was nice that I could write a considerably piece of the report. (I wrote the end part of the evaluation for instance)

Job
My major input ended up being primarily in development of the physical design. Both in the pre-production as in the production phase. I believe my broad background was what lead to me to fulfilling this role.
When reflecting back on my answers to the expectation questions at the start of this course this seems like a good fit. Although having an affinity for programming as well Haico felt most confident in taking this on and so he has. The weakness I mentioned was subpar electronics knowledge. During this past week while integrating some of the electronics in the structural design I got a general idea of how this worked. Although it proved harder for me to decide on what parts we need, the connecting and using of the electronics seemed easily understandable.
Reflecting at the mid-term evaluation my role in the project hasn't shifted much, it just became more practical and technical, by means of learning AutoCad from scratch and applying my understanding of design and wood-/metalworking.
Doing the same reflecting for the entire group it is clear that everyone's strengths were utilized. Furthermore, everyone has got a glimpse of other people's strength, really developing skills inside ones' weakness does not seem to have truly happened however.
Reflecting on the full process this was a good balance however. Finishing the project just before the deadlines makes it clear we used our strengths in a manner that was necessary.