Meeting	Date	Content
1	29-04-2020	This was an introductory meeting during which the group met with the tutor, Wouter Houtman. We briefly discussed with him what he expected of us and how we were planning to make progression. Also, we discussed the contents of the design document. We had a general discussion about the expectations of the course and the two challenges. As a group, we agreed to have at least two meetings per week, one with our tutor to discuss progression and to ask questions and one without him. We agreed that every group member should watch the lectures and make the tutorials. If one does not do so, he will fall behind and will not be able to help the group progress. Moreover, since none of the group members has experience with C++, every group member is advised to practice in the programming language. For next meeting, every group member should, apart from watching the lectures and finishing the tutorials, get some inspiration for the design document. Sources of inspiration are the wiki pages of groups of previous years and the general wiki page with the description of the challenges.
2	01-05-2020	To remember what is agreed upon during the meeting, we assigned someone to take minutes. Next time, he will lead the meeting to improve the structure and someone else will be taking the minutes. A role division scheme will be made. During this meeting, we discussed the five items that should be included in the design document. First, we agreed that we should first focus on the requirements and specification. By further specifying the requirements and specifications, functions will follow. Once the functions are known, they can be divided among the components of PICO. The components will on its turn be included in the software architecture with the corresponding interfaces. We decided to roughly follow the scheme of lecture 2.1 for the requirements and specification. Next, we divided the tasks. Two people will work on the requirements and specifications, two on the functions and two on the interfaces and components. Since the deadline for the design document is 04-05-2020, we set a deadline on 03-05-2020 for a first draft of the different tasks. This gives us sufficient time to process feedback from others and finalize the design document.
3	03-05-2020	This meeting was planned as a follow up on the tasks assigned in the previous meeting. First, some general agreements were made. It was decided to create an OneDrive folder so that it would become easier to share and store documents. Also, the minute taker was made responsible to update this meeting log on our Wiki page. Afterwards, each subgroup shortly explained their work. After this brief explanation, some possible improvements were discussed and noted in the minutes, so that they can be implemented. The main points of improvement were focused on small adjustments for each part to be coherent with the other parts. At last, we looked at how to continue towards the next meeting planned on Thursday. The priority would be to finish the Design Document. It was decided to finish it today so that it can be checked and delivered on Monday before 5 pm. After that deadline, all members would further investigate Tutorial 12 since this would help to create an understanding of simple software implementations. Besides that, the same subgroups were appointed to work out

		some of the capabilities as formed in the Design Document. The expected outcome of this task would be a general approach (or possible libraries/algorithms to use) and possibly a piece of code.
4	07-05-2020	This meeting was used to receive feedback from our tutor on the design document, and to discuss our approach for the escape room challenge. One of the feedback points that was received was that the design document could have been more explicit. We got the recommendation to continue to update the design document since this will make the programming part easier. After this we discussed the algorithms we are working on, and what the advantages and disadvantages of each algorithm are. In the end, we divided the work for next meeting (08-05) in three sub assignments. The first duo would work on the project structure and corner detection, while another duo would work on the potential field algorithm. The last duo is going to investigate and program the corridor exit procedure. The goal for the next meeting is that the structure is finished and that the individual parts can be implemented into one system since integral testing will probably uncover quite a few problems between the components.
5	08-05-2020	During this meeting, the progress of the implementation of the potential field algorithm, an algorithm able to identify the target at the exit of the room, and the investigated approaches of driving straight through the exit corridor are discussed. The rejective field gradient has been implemented successfully and movement in (x, y, theta) can be initiated according to this gradient vector. The potential field method is likely to be able to steer the robot through a corridor if the execution rate is sufficiently high. A split & merge approach for detecting wall features is investigated mathematically using Matlab. Recursive fitting is used to fit lines to measurement points from the laser range finder corresponding to a single feature. Finally, the strategy for the room challenge and hospital challenge has been further elaborated and the structure corresponding to the approach for the room challenge has been implemented. The goal for the next meeting is to compare the split & merge approach with the corner detection algorithm, to see which one is more efficient. Furthermore, the potential field algorithm will be further implemented to include an attractive field. Finally, the implementation in C++ for the room challenge will be continued and the possibility of visualizing gradient vectors and identified corners will be investigated.
6	11-05-2020	In this meeting, we met with our new tutor, Jordy Senden. We got some more feedback on our design document: i) the graph of stakeholders and different types of requirements is good, but the connections can be worked out further and some requirements are still missing a specified value (e.g. min. speed), and ii) it is questionable whether the planning state of the finite state machine is needed (or is planning simultaneously with other tasks). Moreover, we asked the tutor some questions regarding visualization, our strategy for the escape room challenge and programming in C++. After that, we shortly introduced our current software to the tutor: part 1 that (mainly) applies the potential field algorithm and part 2 that (mainly) applies (a sort of) split and merge segmentation. Finally, we further discussed what we were going to work on up until the escape room challenge which is in two days. Regarding part 1 (potential field algorithm), we decided to start using odometry data as a means of updating the target location (i.e. the corridor) for cases where part 2 fails to produce a new target location. A lot of debugging still needs to be done to get part 2 working as expected, therefore we decided to compare MATLAB and C++

		results to see where it goes wrong. Moreover, part 2 will be tested more thoroughly in MATLAB, i.e. using more data obtained from the PICO simulator, and we plan on adding a more thorough preprocessing on the LRF data at the beginning of the part 2 software to increase robustness.
7	12-05-2020	During this meeting, we discussed our results in order to define the last tasks for the Escape Room Challenge. Our potential field algorithm ensures that PICO keeps track of the odometry data in order to identify its current position relative to the target. The split and merge algorithm is now able to locate the target at the exit of the room and at the end of the corridor. However, this algorithm has some robustness issues which need to be fixed. Next to this, the visualization of PICO's perception of the surrounding is implemented. Now, both algorithms need to be combined into a single working program, which then can be tested and finally used for the Escape Room Challenge.
8	15-05-2020	This was the first meeting after the successful Escape Room Competition, which we finished in first place! During this meeting, we discussed how we plan to continue. First, we agreed that we want to update our Wiki. In order to do so, we are going to improve the design document. We have already copied the content of the design document onto our Wiki page, but we are going to implement previously received feedback. Next, we want to elaborate upon the software used for the Escape Room Competition. We are going to explain the split & merge detection method and the potential field algorithm in more detail. While working towards our final version of the software, visualization has been used to check whether the algorithms were implemented correctly and for debugging. However, we intentionally left this out during the Escape Room Competition. Since we do want to show how this has been used, two videos will be added to the Wiki page: one of how PICO behaved during the Escape Room Competition and one of the corresponding visualizations of our software. Moreover, we will prepare for the next meeting by exploring three different topics relevant for the Hospital competition. We will investigate the possibilities for (1) localization and mapping, (2) motion control and obstacle avoidance and (3) path planning.
9	18-05-2020	After the last meeting, it was decided to shortly discuss the findings of the different subgroups before the tutor joined in. During this discussion, the different solutions for (1) localization and mapping, (2) motion control and obstacle avoidance and (3) path planning were related to each other, to understand how the interaction between these components must go. When the group was somewhat in line about the general approach to the challenge, this was discussed with the tutor. From this, it became apparent that there could possibly be some issues with local minima and scalability of our proposed solution. This will then be a point of attention while trying to implement some of the algorithms and solutions. For today's meeting, we had set some goals about roughly finishing the parts of the Wiki concerning the Escape Room Challenge, and the deadline for that has now been set on Thursday evening. Jordy will then be able to read through the parts and give constructive feedback during next the next meeting he will attend. Besides continuing the Wiki-related tasks, three subgroups were formed. One of them is appointed to researching a possible localisation algorithm using a Kalman filter as explained in one of the tutorial lectures. The second group were assigned to path planning, where grid-based methods were now the focus. Lastly, the last duo was assigned to work

		on a variant of the Finite State Machine and possibly start with the Mapping part. The outcomes of this research will be discussed on Friday so that they can be implemented in a software structure.
10	22-05-2020	In this meeting without the tutor, the outcomes from the research on path-planning, mapping and localization were discussed. We looked extensively in the risks an A* algorithm in combination with a potential field algorithm for motion control could pose (local minima), but the group is confident these problems can be avoided/solved. It is therefore decided to start working on the implementation in C++ of this part. Next we discussed the progress with respect to Mapping, and a first Matlab script was made. The group that looked into localization using a Kalman filter studied the literature behind this and came up with an approach for implementation. At the end of the meeting it was decided that the same groups will continue with the implementation of the subjects that were researched until now. The progress of the implementation, and possible problems that were encountered, will be discussed on Monday.
11	25-05-2020	In this meeting with the tutor, the progress of implementing the localization approach, the path planning approach, and the mapping approach were discussed. Regarding the localization method, the feature extraction has almost completely been implemented. Next steps are to implement the EKF to determine position of PICO within world frame during motion, this will be continued by Lars & Jeroen. Regarding mapping it is now possible to obtain a grid from the .json map. Also grid position can be linked to distance measurement from LRF. Next step is to export this grid to a matrix for path planning and update the map based on LRF data, which will be continued by Bas & Aron. Regarding A* path planning, it is possible to include a weight map which has a larger weight close to obstacles, such that optimal path is planned further away from obstacles. Implementation in C++ is started and will be continued by Jelle & Andreas.
12	29-05-2020	In this meeting, the progress of implementing the code for the mapping, localization and path planning functionality was discussed. The mapping software has almost been implemented as far as possible. Since it depends largely on localization, further implementation can be done as soon as a first version of the localization software is available. The progress on the localization software halted a bit, because the implementation of the motion model update part of the EKF was unclear. At this point, the filter itself is being implemented and the main focus now is on the correct implementation in c++ code of the matrix multiplications/inversion etc. The path planning is also largely finished; some final implementations need to be finished. It was decided to finish off the current parts of software and work in the source files of the corresponding classes as much as possible, to combine the visualization software (s.t. everything can be effectively debugged and tested when merging the different functionality), and to have finished the (first working versions) of the individual functionalities by next wednesday. Next week, the different parts can then be merged and the focus can be shifted to implementing the finite state machine, initialization procedures and testing. Moreover, the structure of the presentation was discussed and it was decided that everyone will prepare the slides corresponding to the functionality they worked on (in a shared presentation file).
13	01-06-2020	During this meeting, the progress with respect to mapping, path planning and localization are discussed. Moreover, we also discussed the presentation of coming Wednesday. The tutor was not present during this meeting, we can send questions to him by mail if we have any. By means of this meeting summary, he will be informed what has been discussed during the meeting.

		Mapping has almost finished. To check if it works, an output window is produced while running the code. In this output window, the complete hospital map is shown. This map does not show static objects yet, these need to be recognized by PICO first. For now, PICO is randomly positioned in this map. The static objects that are discovered by PICO during operation are then shown in the output window. Next, it will be checked what happens if a dynamic object is encountered. A dynamic object should be recognized as well and should not be plotted in the output window. Free and occupied grid points in the hospital map are provided to the path planning algorithm. Based on this, a path is produced. This works already quite well, but still some debugging is required. The produced path can still be shorter, and it is wanted that PICO enters a room in the middle of the door opening instead of very close to the wall as it is right now. It is found that path planning can become quite computationally expensive when using a small grid. Some solutions are to compress the given map, use a low refresh rate or to make use of a weighted map that isn't updated every iteration. Tuning of the grid size will mainly depend on what we find when testing the complete model. The potential field algorithm used for the Escape Room Challenge will be reused for the motion of PICO. A change will be made with regards to the rejective field. Previously, PICO was pushed away from what it sees. Now, a rejective field will be placed around PICO. The closer an object gets to PICO, the larger the rejective field will become. It is expected that this is useful when encountering dynamic objects. The extended Kalman filter for localization is almost implemented, but then it must be figured out how the inputs are provided in a clever way. The aim is to finish the filter by the end of today. Initial localization still needs to be implemented. Once this is finished, we can combine the different parts and testing and tuning can start. We discussed that we might want to use different frequencies for different interfaces. For instance, we might want to update mapping only when big changes are found. Also, it might not be necessary to use the complete extended Kalman filter each iteration, but maybe only partially for an update of the odometry. Also, it is not necessary to update the produced path every iteration. In case the map is updated, for instance when a static object is added which interferes with the produced path or when PICO significantly deviates from its path, a new path should be planned. Also, we discussed how to keep the software organized. We agreed to finish the presentation slides by the end of today. A general setup is made for the presentation. Each group will produce a couple of slides for their own topic (mapping, path planning, localization). We have set up a meeting for tomorrow in which we will discuss the presentation, add what is still missing and discuss who will give the presentation.
14	05-06-2020	During this meeting the results up till now were discussed. First the localization algorithms. This seems to be the bottleneck at this point since the implementation took a while. However, we managed to give an output with the estimated position of PICO (x,y,theta) with respect to the world frame. This works perfect without drift, however when drift is enabled we see a little error appearing which seems to be very small. It needs to be tested to see if this error causes complications when driving around in the simulation map. The localization is based on the indentification of landmaks placed at corners of the map. To make the algorithm more robust, only the landmarks of the room which pico is in should be considered.

		We are still uncertain if the motion model that we arrived is correct since there is still a term that contains drift in it. We stated a discussion on canvas, to see if we could get some help to figure this out. If this won't work in the end we could also look at a velocity based motion model. Considering planning and mapping, the algorithms work well together. Currently a gross path is planned to the final destination based on a compromised version of the map. This path is then divided in shorter paths which are sequentially executed by PICO. To execute this short path, a more detailed path is planned to ensure that PICO moves smoothly. Things that still need to be done are: building an emergency stop when this is needed; making PICO align with the cabinets using the attractive field algorithm; implementing the FSM structure and making snapshots when pico is aligned to a cabinet.