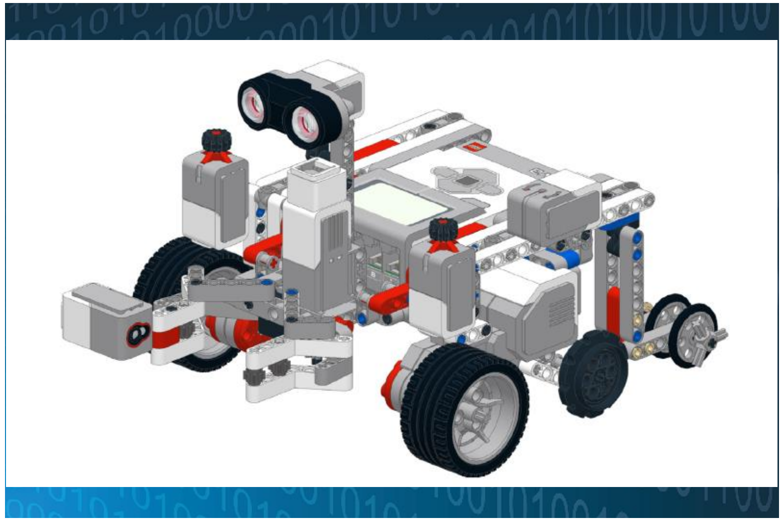
TEC
Technik

Hochschule Reutlingen
Reutlingen University



Phase 3 –
Challenge

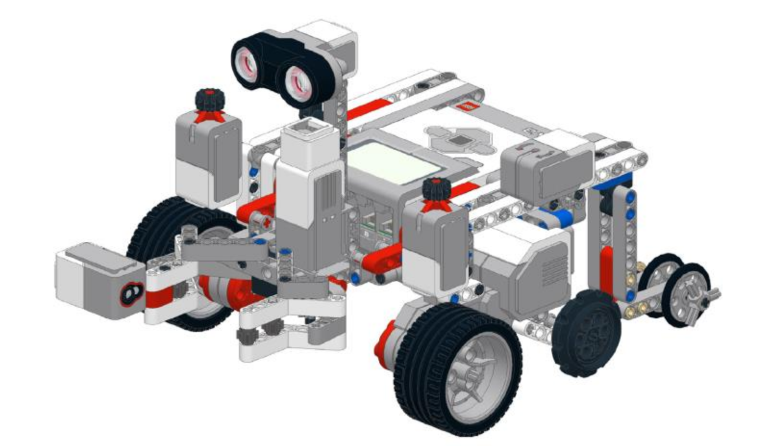
Harvester

Computer Science for Engineers

Michael Danner
Markus Wachter

International Project Engineering

Reutlingen University

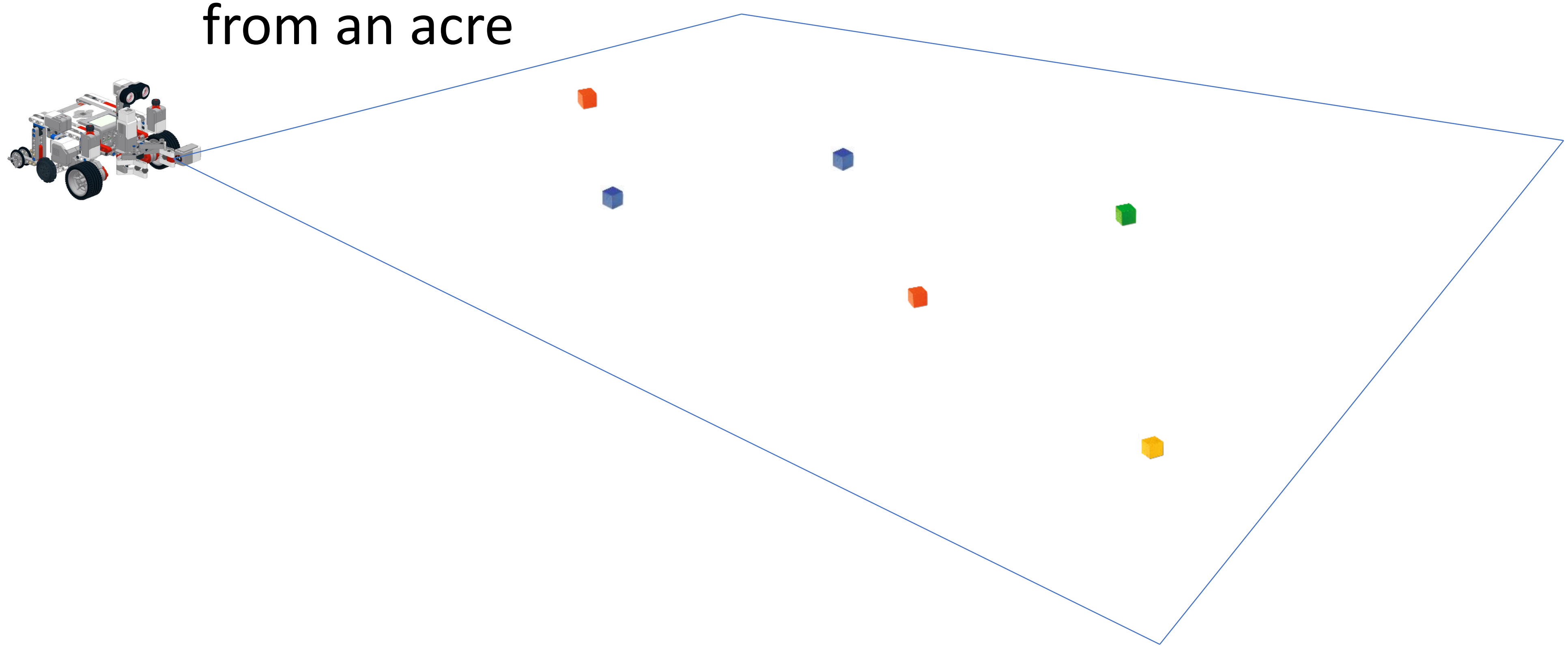


Contents

1. Challenge: Harvester
2. Time planning and lecture:
   Retrospective and reviewThe EV3 sensors
3. Grading:
   Presentations, documentation, challenge

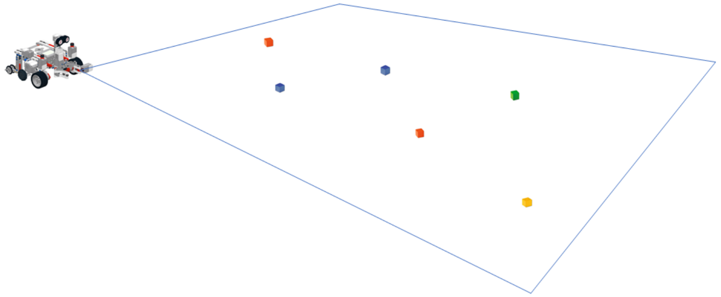
The Task

- The scope of your project is to develop a robot that harvests fruit from an acre



https://cdn.pixabay.com/photo/2014/04/03/00/40/man-309030_960_720.png
https://www.wiemann-lehrmittel.de/shop/naturwissenschaften/mathematik/krper/geometrische-krper/bunte-holzwrfel

Boundary Conditions



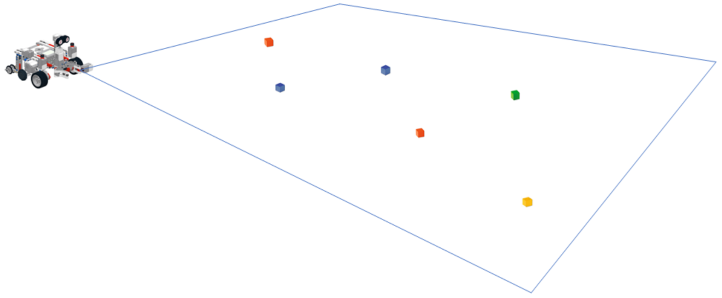
Material:

- Receive:
  The EV3 Mindstorm Educational Set
- Create:
  A robot which can solve the challenge. For your robot you can use all parts provided in the EV3 set which you received, but no additional parts.

Goal:

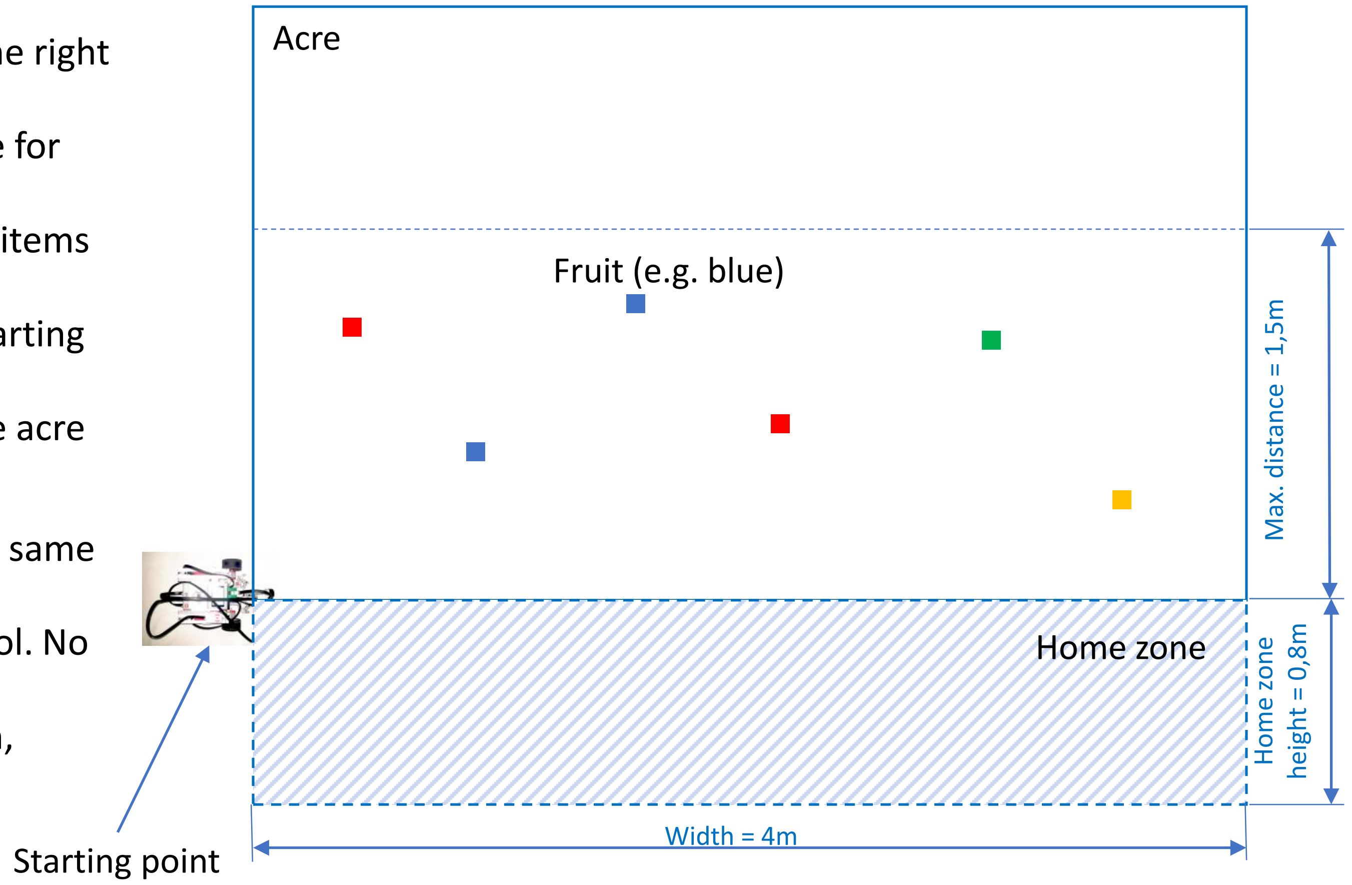
- Your robot searches for the right type of fruit on an acre and harvests it.
- You have to structure your project according to the Scrum approach.

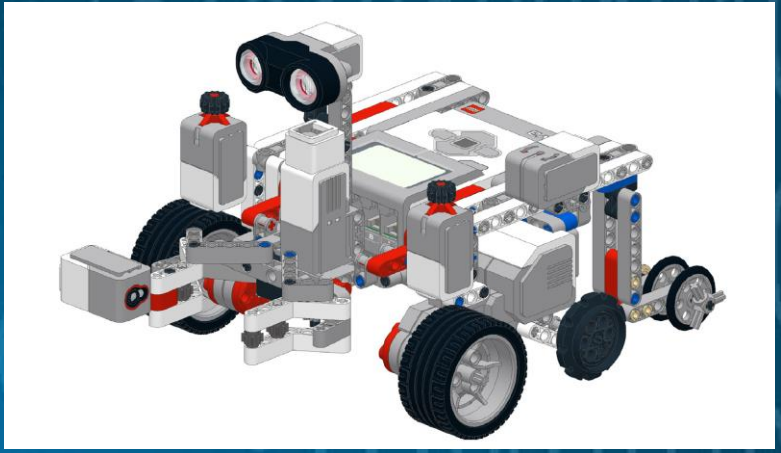
- Mind that the customer may have implicit requirements or may change his mind after the review of increments.
- You can request requirement changes throughout the project, which will be reviewed and decided by the customer.
- If requirements are unclear during the project, you need to address these during the sprint presentations.



Requirements V1.0 (24.10.2022)

- R1: After the program is started, the robot shall ask for the color of the fruit to be harvested („right fruit“)
- R2: The user shall be able to „tell“ the robot the color of the right fruit in an intuitive way
- R3: After the robot knows the color, it shall search the acre for the right fruits and harvest them within 5 minutes
- R4: Harvesting means, that the robot places the right fruit items into a home zone
- R5: After the 5 minutes, the robot shall drive back to its starting point
- R6: The starting point will be in the lower left corner of the acre
- R7: See additional details in the image
- R8: Fruits will be placed randomly, but for all groups at the same place, especially the right fruits
- R9: Robot must be autonomous without any remote control. No reprogramming after the challenge is started
- R10: The fruits are wooden cubes, colored blue, red, green, yellow, edge dimension roughly 2,5cm
- R11: There will be 2 right fruits amongst all fruits





Contents

1. Challenge: Line follower
2. Time planning and lecture:
   Retrospective and reviewThe EV3 sensors
3. Grading:
   Presentations, documentation, challenge

Summary

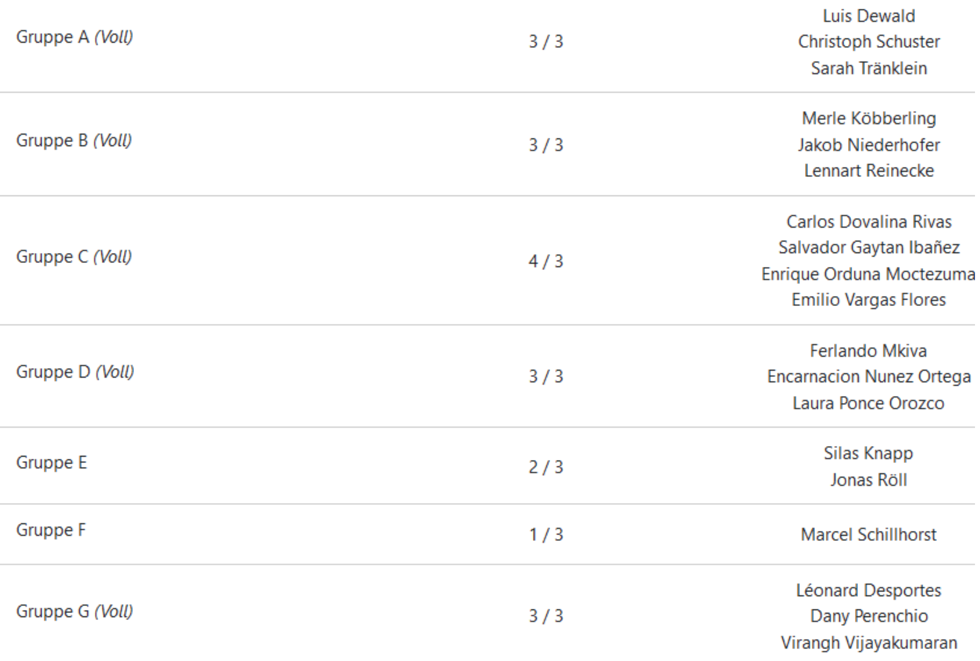
	Content	Duration	Examination	% of grade	
1	• Process • Python • SCRUM • Hardware	5 weeks	Test (CW TBD)	50%	CW 40, 41, 42, 43, 44
2	• Robot exercises • Challenge announcement	1 week			CW 45
3	• Project	4 weeks	Presentations Documentation	20% 20%	CW 46, 47, 48, 49
4	• Challenge	1 week	Challenge code Challenge	10% Bonus	CW 50

Time planning for the project: Slots & Groups

4 Slots for Sprint presentations

Slot	Time
1	13:45 – 14:15
2	14:20 – 14:50
Q&A	14:50 – 15:00
3	15:10 – 15:40
4	15:45 – 16:15
Q&A	16:15 – 16:25
Buffer	16:25 – 17:00

Groups stay the same as documented in Relax



Gruppe A *(Voll)*	3 / 3	Luis Dewald Christoph Schuster Sarah Tränklein
Gruppe B *(Voll)*	3 / 3	Merle Köbberling Jakob Niederhofer Lennart Reinecke
Gruppe C *(Voll)*	4 / 3	Carlos Dovalina Rivas Salvador Gaytan Ibañez Enrique Orduna Moctezuma Emilio Vargas Flores
Gruppe D *(Voll)*	3 / 3	Ferlando Mkiva Encarnacion Nunez Ortega Laura Ponce Orozco
Gruppe E	2 / 3	Silas Knapp Jonas Röll
Gruppe F	1 / 3	Marcel Schillhorst
Gruppe G *(Voll)*	3 / 3	Léonard Desportes Dany Perenchio Virangh Vijayakumaran

Time planning for the project: details

Week	Topic	Slots and Groups	Project Work (during next week)
45	- Challenge rules - Presentation requirements - Documentation requirements - Retrospective	All	Project setup
46	Project setup - Presentation - Q&A	**Slot 1:** Team A, B, C ; **Slot 2:** Team D, E; F **Slot 3:** Team G	- Sprint 1 - Retrospective
47	Sprint 1 - Review presentations - Q&A	**Slot 1:** Team D, E, F ; **Slot 2:** Team G, A, B **Slot 3:** Team C	- Sprint 2 - Retrospective
48	Sprint 2 - Review presentations - Q&A	**Slot 1:** Team G, A, B ; **Slot 2:** Team C, D, E **Slot 3:** Team F	- Sprint 3 - Retrospective
49	Sprint 3 - Review presentations - Q&A	**Slot 1:** Team A, B, C ; **Slot 2:** Team D, E; F **Slot 3:** Team G	- Prepare for challenge - Prepare for test
50	**Challenge on 15.12.2022**	All together	**Upload documentation until 22.01.2023**

Plan workload for this course 6 ECTS = 180 h

- Lectures: physical presence (27 h)
  - Python / Robot / SCRUM 6 * 3h = 18 h
  - Sprint Presentations 4 * 1.5 h = 6 h
  - Challenge 1 * 3 h = 3 h
- Python Phase 1 / Robot Phase 2: own preparation (45 h)
  - 7.5 h per week 6 * 7.5 h ˜ 45 h
- Project work incl. documentation (96 h)
- Test preparation & Test (12 h)

Test content

- Test preparation & Test (12 h)
  - Manage your time well to work on “2 projects”:
    - The challenge
    - The test preparation and the test itself
- Possible test content:
  as already announced on Relax
  - waterfall, v-modell and scrum methods (theoretical)
  - activity and use case diagram
  - code a python program: input, print, if...else, for/while, functions
    data types and list/dictionary/set
    write code from scratch / fill the missing parts of code / correct some errors
- Only pen is allowed, duration 1 hour

Lecture during the challenge

- **During the lecture**
  - Present the current **increment** to the customer (Danner/Wachter)
  - Ask the customer **questions regarding the requirements**, if any

- **Outside lecture**
  - **Plan** the new sprint
  - Carry out the **sprint**
  - Reflect on **team processes** (Retrospective)
  - Prepare **Sprint review**
  - Option to clarify **questions regarding programming** in Mr. Wachters consultation hour

Retrospective

This is NOT about the sprint goal / increment!

Improve the development process and practices; make it more effective and enjoyable; increase product quality by improving work processes.

Six typical steps

1. Warm-up
2. **Review of the goal from the last retro (not sprint)**
   Was it implemented? Why (not)?
3. **Collect data**
   What went well, what challenges are there?
   Collect at least one highlight and one lowlight.
4. **Evaluate data**
   What is the root cause for our lowlight?
   i.e. 5Why.
5. **Goal**
   What can we do to improve the process and quality of your software development?
   Note down a SMART goal for next sprint.
6. Cool-down

Outcomes to be presented in review meeting

1. **Last retro goal met?**
   - If yes, did it improve the process (keep)?
   - If not, why not (drop or adapt)?
2. **Present data**
   - One new highlight
   - One new lowlight
   - Describe reason for new lowlight
3. **New retro goal**
   - Spell out your SMART goal
   - Clearly mark the elements (**specific, measurable, achievable**)

Presentation of project setup (Petcha Kutcha)

10 slides* in power point format that will auto forward after 20 seconds

Intro

- Present team / team name / product owner / robot name
- Pitch your project vision

Setup

- Present product backlog
- Explain prioritization of backlog items
- Estimate the effort (not hours – use a virtual currency) for each backlog item and show your estimated velocity
- Present your general DoDs
- Show use case diagram

*ALL slides count (also title)

Contents of Sprint Review (Petcha Kutcha)

12 slides* in power point format that will auto-forward after 20 seconds

Status

- Present the goal of the sprint and explain the current status of the increments
- Show burn down chart and product backlog
- Highlight changes in of backlog, vision, if any etc.
- Discuss actual effort vs. planned effort and show the team‘s velocity
- 1 slide: Video of your robot

Details

- Example of a detailed user story (ask and answer the w-questions)
- Example of software design for implemented items (activity diagram)
- Discussion of tests done for this user story (setup, expected & actual results, conclusion)

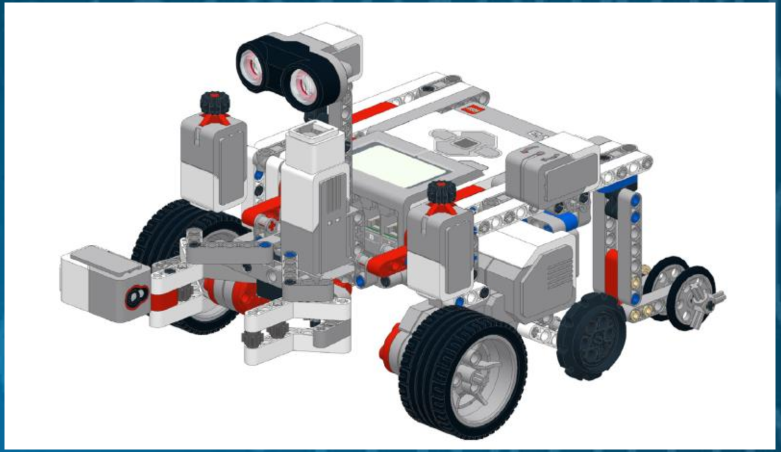
Reflection

- Have you reached the retro goal formulated in the last retrospective?
- Name one highlight and one lowlight in team processes in last sprint
- Name SMART goal how your team will address the challenge

*ALL slides count (also title)

Organisation of the presentations

- You will be assigned to a slot of 3 groups, each slot having its own timeslot for presentations
- You only have to be present during the presentations of your block. For instance Slot 1, 2, Q&A is a block
- You will get feedback from the customers (Danner / Wachter)
- Don‘t take that as personal criticism, but input for improvement
- Q&A means at the end of each block every team could ask questions about presentation and challenge



Contents

1. Challenge: Line follower
2. Time planning and lecture:
   Retrospective and reviewThe EV3 sensors
3. Grading:
   Presentations, documentation, challenge

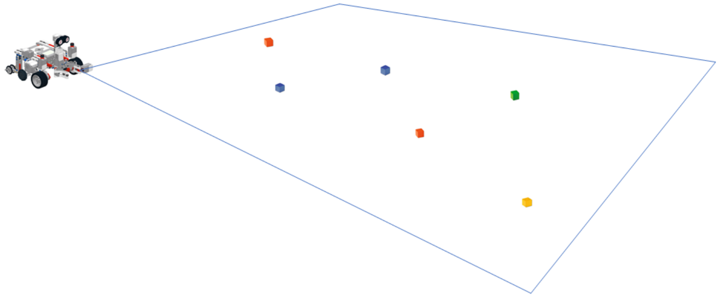
Grading

- **50% Test** (Individual, 50 points max)
  Possible contents: See other slide in this presentation
- **20% Sprint review presentation** (Team, 20 points max)
- **20% Documentation** (Team & individual*, 20 points max)
- **10% Challenge code** (Team & individual*, 10 points max)
- **Bonus Challenge** (Team & individual*, 10 extra points max)

*For students not present or talking during any presentation:
No points for presentation. 50% points deduction for documentation and bonus points in challenge.

It is not possible get full points for the project, if you are not active during the project!

Bonus points in challenge



Points	Condition
1	Robot asks person for intuitively telling the color of the right fruit
1	Telling the right fruit was successful (e.g. robot told the identified color)
1	Robot detects at least **one fruit** (e.g. robot obviously approaches the fruit)
1	Robot identifies at least one **right fruit** (e.g. robot tells it or grabber is closed)
2	Robot successfully harvests a right fruit to the home zone (each right fruit counts)
2	After the 5 minutes, the robot returned to its starting point (+/- 20cm all directions)

You can get maximum 10 points if all conditions met.
You will get the highest number of points out of 3 trials.
Whenever the robot ends up in confusion (not able to search any longer, disoriented), the trial is over.

Documentation: Structure

Suggestion:

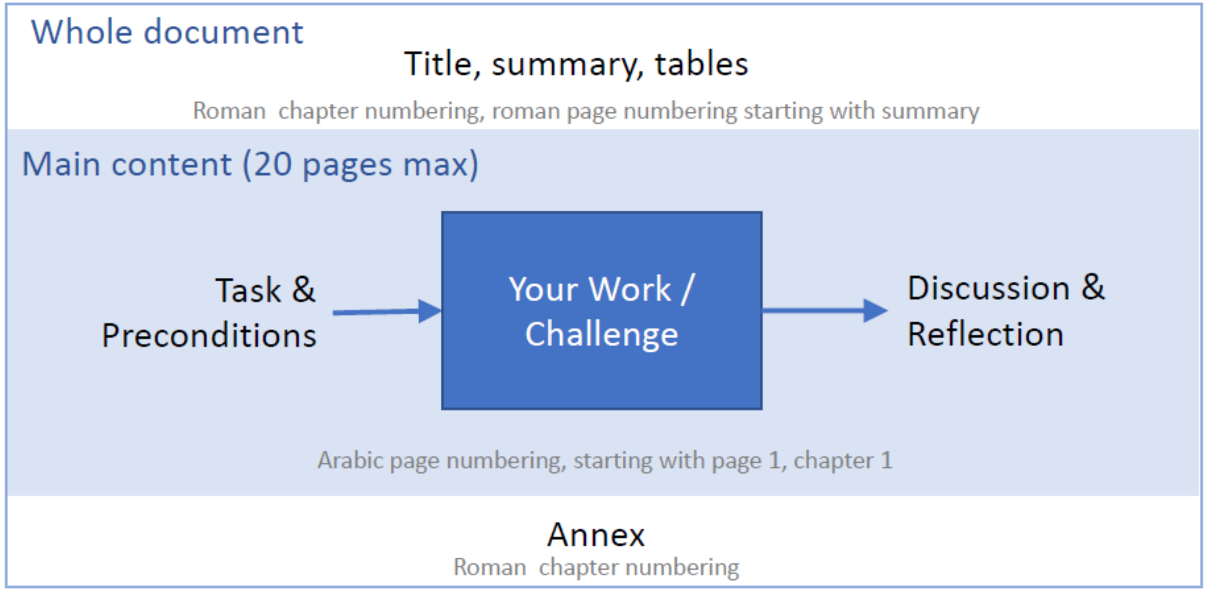
- Title page
- Table of contents, tables, graphics
- Introduction, given task (explain in own words).
- Technical and other preconditions
- Analysis: Plan how to tackle the problem
- Design and implementation with
  - overview on increments and status per sprint, detailed user stories, burndown…
  - description of one test of a specific user story per sprint including: Test environment & specification; expected & actual results; bug fixing in case of failed test
- Discussion and Reflection
  - Results of the sprint retrospectives,
  - Discussion of your final product; the suitability of Scrum for this task, the project as didactical method)
- Annex: Complete source code & others

Documentation as a scientific document

The documentation is a scientific document and must adhere to the according criteria:

- Table of contents, tables and graphics and a list of references exist.
- Tables and graphics are made by yourself or the source is referenced, they have a useful heading, are numbered and always referenced and briefly described in the text without adding redundancy.
- Chapters have useful headings and are numbered, subchapters are existing where necessary.
- Ideas that stem not from yourself are properly cited according to citation rules.

Documentation: Formatting


Whole document
Title, summary, tables
Roman chapter numbering, roman page numbering starting with summary
Main content (20 pages max)
Task & Preconditions
Your Work / Challenge
Discussion & Reflection
Arabic page numbering, starting with page 1, chapter 1
Annex
Roman chapter numbering

1. The main content may contain a maximum of 5000 words
2. Use a serif font of 11 point font size
3. All page margins: 2.7cm
4. Line spacing 1.5

Grading criteria for documentation

The documentation proves the acquired competencies on process, conceptual and factual level:

1. Process know how
2. Conceptual know how
3. Factual now how
4. Reflection
5. Formal criteria

Details see next slide.

Details on grading of documentation

Documentation	0 - 2 P	3 - 4 P	5 - 6 P	7 - 8 P	9 - 10 P.	Create and extend
Processes know how (SW Development Process Scrum)	The SW development principles where not sufficiently applied or were incorrectly applied in most steps.	The SW development process and related terms are applied in principle.	The agile SW development process is applied correctly in most cases, with only minor issues in certain steps or methods.	The agile SW development process is always applied correctly. Methods are correctly implemented described transparently and used in a goal-oriented way.	The results of the retrospective lead a successively to process improvements, process and methods always applied correctly.	Students research and applied additional methods around the SW development process to improve their results
Conceptual know how (Analysis and Design)	Methods to to analyze and design SW, such as a backlog with prioritized user stories, DoDs activity and use case diagram, acceptance criteria, tests … are rarly used to reach the project goal.	Some of the methods to to analyze and design SW, such as a backlog with prioritized user stories, DoDs activity and use case diagram, acceptance criteria, tests … are used to reach the project goal.	Most of the methods to analyze, design and test SW, such as a backlog with prioritized user stories, DoDs activity and use case diagram, … are used to reach the project goal.	Most methods to analyze and design SW, such as a backlog with prioritized user stories, DoDs activity and use case diagrams, … are used effectively and described transparently to reach the project goal.	All methods to analyze, design and improve SW, such as a backlog with prioritized user stories, detailled user stories, DoDs activity and use case diagrams, tests … are used effectively, correctly and are always described transparently to reach the project goal.	Additional tools to improve the analysis and design of software are researched, implemented, discussed and lead to an improvement to reach the project goal.
Factual know how (Implementation & Test)	The student is not able to read and interpret structured Python code. The student cannot detect and correct easy syntax errors.	The student can read and interpret structured Python code. The student can detect and correct easy syntax errors.	The student can read and implement structured Python code according to a given SW design. The student can detect and correct easy syntax and semantic errors.	The student can read and implement more complex structured Python code based on an own design. The student can detect and correct all syntax, logical and semantic errors.	The student can read and implement complex structured Python code based on an own design. The student can detect and correct all syntax, logical and semantic errors and implements strategies to improve the testing.	The student researches adapts and implements successfully new methods or functions (Example: Controllers, Test Driven Development, Model based testing).
Reflection	No reflection or very poor reflection	The documentation contains a rudimentary reflection.	The documentation contains a clearly described reflection for some of the areas.	The documentation contains a clearly described reflection for most of the areas.	The documentation contains a reflection for all the areas and is transparently and logically described.	Additional methods, tools or strategies are critically discussed and their value and transferability to further scenarios are described.
Formal criteria	The documentation setup according to a basic scientific structure is poor.	The documentation is setup according to a basic scientific structure but contains many formal errors.	The documentation is setup mostly according to the scientific structure but contains some formal errors.	The documentation is setup according to a the scientific structure and contains only minor formal errors.	The documentation is always setup according to the scientific structure, contains no formal errors.	The documentation formally exceeds the expectations due to an immaculate formal structure and visuals.

Grading criteria for presentations

Graded are:

1. Structure of presentation
2. Presentation visuals
3. Presentation style and personality

Details see next slide.

Details on grading of presentation

Presentation	0 - 2 P	3 - 4 P	5 - 6 P	7 - 8 P	9 - 10 P.
Structure according to given standards	The presentation was not structured according to given requirements, illogically structured and hence incomprehensible.	The presentation was in large parts not structured according to given requirements and hence barley comprehensible.	The presentation was in partly structured according to given requirements and hence partly logical and comprehensible.	The presentation was in most parts structured according to given requirements and lead in most parts logically from one argument to the next, in order to motivate and solve the problem.	The presentation was in all parts structured according to given requirements and always lead logically from one argument to the next, in order to motivate and solve the problem.
Presentation visuals	The presentation visuals were unclear in all parts. Slides were not helpful for the understanding or did not auto-forward.	The presentation visuals were in large parts unclear. Slides were often not helpful in supporting the presentation (i.e. just text, overloaded, etc.).	The presentation visuals were acceptable. Slides were used, they are sometimes not helpful in supporting the presentation (i.e. just text, overloaded, etc.).	The presentation visuals were predominantly clear. Slides were in most cases supporting the presentation (i.e. many graphics, and tables, not overloaded, etc.).	The presentation visuals were perfect and slides always supported the presentation. The proportions were utilized to keep the audience interested. Speaker referred to the content. (i.e. many graphics, and tables, not overloaded, etc.).
Presentation style and personality	The personal presentation style was poor, due to incomprehensible speech and no contact to the audience at all.	The personal presentation style was rather boring. The speech was often incomprehensible, little contact to the audience was made.	The personal presentation style was in some parts lively, but the speech sometime incomprehensible, or sometimes the contact to the audience lost.	The personal presentation style was in large parts lively, with few incomprehensible parts, or few lost contacts to the audience.	The personal presentation style was lively and animated, facing and including the audience, with clearly comprehensible speech.