How Week 10 Connects to Your Group Project

Week 10	Visual DQN navigation from camera observations
Group project	Visual target identification + safe navigation to a designated station position
Key message	Week 10 is a building block

The group project is open-ended. The project brief does not prescribe a particular navigation, planning, vision, or control method. Week 10 gives you useful concepts and engineering patterns, but you should not assume that the project should use DQN.

The Main Connection

The strongest connection is not the DQN itself. It is the system-level pattern you are practising in Week 10:

observe → interpret → decide → act → observe again

In Week 10, the observation is the forward RGB image, the decision is one of three navigation actions, and the action is differential-drive motion.

In the group project, the same closed-loop idea appears inside a larger system. The robot should visually determine which station contains the requested target, then navigate to that station's designated observation position and stop.

Week 10	Group project	Connection
RGB camera observation	RGB camera must identify the target station	Both require reliable visual information
Three navigation actions	Any appropriate navigation/control method	Both close the perception-control loop
Goal distance / collision	Designated observation position / collision-free motion	Both need measurable task outcomes
reset / scenarios	Starts A/B/C and changing target-station arrangements	Both require repeatable testing
training vs held-out scenarios	Multiple missions and starting positions	Both require robustness, not one successful run
trajectory diagnosis	Failure cases in final report	Both require evidence-based debugging

What Week 10 Teaches You That Directly Helps

Camera data should become useful evidence

The project requires the RGB camera to determine which observation station displays the requested target. The project provides the target identity, but not the station ID or target location. Therefore, the camera should provide the evidence that connects the requested target to one particular station.

Connection to earlier workshops

Week 9 taught the same perception principle in a controlled form: raw camera data is converted into a useful representation, a detection is made, and a decision is taken only when the evidence is sufficient.

Week 10 extends that idea from object detection toward learned navigation.

Separate perception from action

A useful project architecture is to keep visual identification, navigation planning, motion control, and safety logic as separate modules. Week 10 reinforces this separation because the environment converts a high-level action into wheel motion while the policy decides which action to take.

- Perception: Which station appears to contain the requested target?
- Planning: What route should the robot take to that station?
- Control: How should the wheels be driven to follow the route?
- Safety: How do we prevent contact with obstacles and station structures?
- Evaluation: Did the robot actually reach the required observation position?

Closed-loop control is more important

Week 10 follows the loop of observing, choosing an action, moving, and observing again. The group project should also behave as a feedback system. The robot should not simply issue a long pre-recorded sequence and hope that the result is correct. The project explicitly prohibits pre-recorded routes and mission-specific timed wheel sequences.

What Is Different in the Group Project?

The project is not simply 'Week 10 with a different map'. The project gives you additional resources and imposes a specific visual identification requirement. DQN is one possible research direction, but it is not required.

Week 10	Group project	Why it matters
Policy sees forward RGB image	Camera must identify which station has target	Project vision is a required decision, not optional input
Goal used internally by environment	Station coordinates and pose are provided resources	Project can use map/GPS/pose for navigation
Three fixed high-level actions	No prescribed action/controller structure	Project teams may choose their own control design
Reward-driven learning	No requirement to use RL	A deterministic planner/controller may be more appropriate
Small scenario banks	Starts A/B/C + changing target-station arrangements	Project needs robust integration across missions
Learning-time evaluation	4-minute assessed mission	Project must complete reliably within the assessment limit

The project specifically states that the provided map, GPS and pose information may be used for localisation and navigation, but they cannot be used as a substitute for visual target identification. This is a crucial design boundary.

A Useful Way to Think About the Project Architecture

A clean high-level architecture could be understood as five connected stages:

Mission target → Visual search → Station identification

→ Navigation → Safe arrival

1. Read the required target identity from the supplied mission interface.
2. Use the RGB camera to inspect candidate visual targets and determine which station displays the required target.

3. Once the correct station is identified, obtain or use its known designated observation position.
4. Plan and execute a collision-free route using appropriate project resources such as the map, pose and sensors.
5. Stop when the robot reaches the required observation position within the project tolerance.

> Week 10 mainly concentrates on the navigation-learning loop.
>
> Your group project adds a visual search and target-identification problem before the navigation problem. That is why the project is an integration task rather than simply another navigation exercise.

Where Week 10 Ideas Can Be Reused in the Project

Observation and image processing

The Week 10 camera pipeline gives you a disciplined way to handle image data: acquire the image, convert it correctly, verify dimensions and type, and then pass it to the component that needs it. The project can use the same engineering discipline even if the actual visual matching method is completely different.

- Always inspect the raw camera image before trusting a detector or matcher.
- Verify colour representation and image dimensions.
- Keep visual preprocessing consistent between reference images and camera observations where appropriate.
- Save representative images when debugging false positives and false negatives.

Differential-drive control

Week 10 also reinforces the relationship between high-level navigation decisions and left/right wheel velocities. The project uses the e-puck as well, so this low-level control understanding is directly relevant.

Safety and stopping behaviour

Week 10 treats collision detection and stopping as part of the environment contract. The project similarly requires collision-free operation and requires the robot to stop at the designated observation position.

Reproducible evaluation

Week 10 uses fixed scenarios and held-out scenarios to distinguish memorisation from generalisation. The project similarly requires the same submitted controller to operate across different starting locations and changing target-to-station assignments. Testing only one successful run is not sufficient.

What Week 10 Can Teach You About Using DQN in the Project

If a group is considering reinforcement learning for some part of the project, Week 10 provides a useful prototype for the interface: an observation, a discrete or continuous action space, a reward, an episode boundary, and an evaluation protocol.

However, there are important practical differences. In the group project, the map, station coordinates and robot pose are already available. The project does not require the robot to learn navigation from scratch. The main unknown that should be resolved visually is which station contains the requested target.

> Do not add DQN merely because Week 10 uses DQN. Use it only if your group has a clear reason why learning-based control is useful for your chosen system and you can train, validate and reproduce it reliably within the project constraints.

A Strong Project Development Workflow Inspired by Week 10

- First make the robot move reliably with simple deterministic commands.
- Then make the camera acquisition and visualisation reliable.
- Then solve visual target identification independently of navigation.
- Then test navigation to a known station position independently of visual identification.
- Then integrate target identification with station selection.
- Then integrate station selection with navigation.
- Then add safety behaviour and verify collision avoidance.
- Finally test the complete controller across different starts and target-station arrangements.

This staged integration is important. If target identification and navigation are implemented simultaneously from the beginning, a failure becomes difficult to localise. Week 10 demonstrates why each layer should first be validated independently before relying on a learned or integrated policy.

A Practical Test Matrix for Your Group

Use a small test matrix rather than repeatedly running the complete assessment mission without recording what happened.

Test	Question	Evidence	Pass condition
Camera test	Can we reliably see station images?	Saved camera frames	Target/station region is visible
Vision test	Can we identify the requested target?	Detection/matching evidence	Correct station selected
Navigation test	Can we reach a known observation position?	Trajectory + final pose	Within required tolerance
Safety test	Can we avoid barriers/stations?	Trajectory + collision log	No collision
Integration test	Does visual identification trigger correct navigation?	Full mission log	Correct station + arrival
Robustness test	Does the same controller work across starts/assignments?	Results table	Consistent behaviour

What You Should Take Away from Week 10 for the Project

- Build the interface carefully before trying to make the robot intelligent.
- Treat camera observations as data that must be verified, not as automatically meaningful information.
- Separate perception, planning, control, safety and evaluation where possible.
- Use closed-loop feedback rather than relying on long open-loop motion sequences.
- Test individual components before integration.
- Use repeatable scenarios and record evidence.
- Evaluate task success, not just an internal score or reward.
- Diagnose failures from evidence and change one thing at a time.
- Do not assume that a method used in a workshop is a required project method.

Week 10 is teaching you how to build and validate a robot-learning system around a clean observation → decision → action loop.

Your group project uses the same engineering mindset, but adds a required visual search problem and gives you map, pose and station information to support navigation.

Suggested Group Discussion Questions

- Which parts of our project should be deterministic, and which parts, if any, should be learned?
- What exactly is the visual evidence we will use to decide that a station contains the requested target?
- How will we distinguish a target station from a distractor image on B1–B5?
- How will we know that visual identification is correct before committing to navigation?
- How will we use the provided map and pose without using them to replace visual target identification?
- What is our collision-avoidance mechanism near barriers and observation stations?
- How will we test the same controller across different starts and changing target assignments?
- What evidence will we collect when the system fails?

Final Reminder About the Project Requirements

The project requires one autonomous controller that uses the e-puck RGB camera to determine which observation station displays the specified target, then navigates safely to that station's designated observation position. The same controller should work across missions, without a hard-coded target-to-station mapping or pre-recorded mission-specific route. The robot should avoid collision and complete the mission within the four-minute limit.

The project provides an occupancy-grid map, station information, designated observation positions, robot pose access, RGB camera, proximity sensors and target reference images. These resources may be used as appropriate, but the station containing the required target must be identified from RGB camera observations.

Week 10 should make your project development more systematic.