

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CODEDU’s Students’ Training Course** | | | | | | |
| **Section 6:**  Presenting and Documenting Projects, Assessing Learning and Progress | | | | | | |
| **Subject: Arduino in STEM** | **Duration (in hours): 15 lesson hours** | | | |  | |
| **Target audience:**  **Upper-primary and Secondary School Students** | | | | | | |
| **Training methodology:**  **Online and Flipped Learning,**  **Demonstration** | | | | | | |
| **Level (and cycle, if applicable) of the learning experience: Upper-primary and Secondary School Level (EQF3)** | | | | | | |
| **Assessment method: Multiple Questions** | | **Form of participation in the learning activity:** | | | | |
| **Expected Learning outcomes:**   * Create engaging project presentations using multimedia effectively. * Structure clear and professional project reports or portfolios. * Apply self-assessment and peer feedback for improvement. * Use documentation and reflection to track progress. * Enhance learning through iterative feedback and revision. | | * Active methodologies and teacher explanation | | | | |
| **Prerequisites needed to enrol in the learning activities (if needed):** | | | | | | |
| **Supervision and identity verification during an assessment:** | | | |  | |  |
| • Unsupervised with no identity verification. | | |  |  | |  |
| • Supervised with no identity verification. | | |  |  | |  |
| • Supervised online or onsite with identity verification. | | | X |  | |  |
| **Further information:**  **1. Arduino Project Hub** – <https://create.arduino.cc/projecthub>  **2. Arduino Style Guide** – <https://www.arduino.cc/reference/en/> | | | | | | |

|  |
| --- |
| **Module 6** |
| **Arduino in STEM Education** |
| **Introduction:**  In this topic, you will learn how to effectively present and document your Arduino projects while also developing critical assessment skills. Communicating your work clearly is essential, whether you are sharing your project with classmates, teachers, or a wider audience. A well-structured presentation helps highlight your creativity and problem-solving skills, while proper documentation ensures that others can understand and even replicate your work. Additionally, creating a project report or portfolio will allow you to keep track of your learning process and showcase your progress over time.  Beyond presentation skills, this topic also focuses on self and peer assessment. Evaluating your own work and providing constructive feedback to others are key aspects of continuous improvement. You will explore different techniques for assessing projects, including reflection on challenges, solutions, and outcomes. Learning how to give and receive feedback in a productive way will help you refine your projects and develop a growth mindset. By the end of this module, you will have the tools to not only present your projects effectively but also critically assess your learning journey.  **6.1. How to Present and Document Your Project** ****6.1.1. Why is Project Presentation Important?**** When working on an Arduino project, the way you present it determines how well your audience understands and appreciates your work. A strong presentation showcases your problem-solving process, creativity, and technical skills. Whether you are sharing your project with classmates, teachers, or even a wider audience, an effective presentation helps communicate your ideas clearly and makes your work more impactful. ****6.1.2. Tips for Creating a Compelling Project Presentation**** To create an engaging presentation, consider the following elements:   * **Clear Structure:** Organize your presentation into sections, such as introduction, objectives, process, results, and conclusion. * **Visuals and Demonstrations:** Use images, diagrams, and live demonstrations to make your explanation more dynamic. * **Concise and Engaging Speech:** Avoid overly technical explanations; instead, explain your project in a way that is accessible to your audience. * **Confidence and Preparation:** Rehearse your presentation to ensure clarity and fluency.  ****6.1.3. Using Multimedia and Demonstrations**** Multimedia elements such as slides, videos, and animations can enhance your presentation. A **live demonstration** of your Arduino project is also a powerful way to engage your audience. If a live demo is not possible, consider recording a video of your project in action. ****6.1.4. Writing a Project Report or Creating a Project Portfolio**** Proper documentation is crucial for sharing your work with others and keeping a record of your progress. A **project report** should include:   * **Title and Objective:** A brief explanation of what your project aims to achieve. * **Materials and Components:** A list of all necessary components and tools used. * **Development Process:** Step-by-step explanation of how you built your project. * **Challenges and Solutions:** Describe any issues faced and how you resolved them. * **Results and Conclusions:** Summarize what you learned and possible improvements.   Alternatively, a **project portfolio** compiles multiple projects, highlighting key aspects and demonstrating growth over time. Portfolios are useful for showcasing skills in future academic or professional opportunities. **6.2 Self and Peer Assessment******6.2.1. The Importance of Documentation and Reflection**** Beyond presenting a project, it is essential to reflect on the work done. Keeping a record of your progress helps you recognize what you have learned and areas where you can improve. **Reflection** is a key part of learning, as it allows you to analyze successes and mistakes, preparing you for future projects. ****6.2.2. Techniques for Evaluating Your Own and Your Peers' Work**** Self and peer assessment are valuable tools for improving skills and achieving better results. Some common techniques include:   * **Checklists and Rubrics:** Use predefined criteria to evaluate project quality. * **Strengths and Areas for Improvement:** Identify what works well and what can be improved. * **Comparing Against Objectives:** Assess whether the project meets its original goals.  ****6.2.3. Constructive Feedback and Continuous Improvement**** Giving and receiving feedback is a skill that helps in personal and academic growth. **Constructive feedback** should be specific, positive, and focused on improvement. Follow these guidelines:   * **Be Specific:** Instead of saying "It looks good," say "The wiring is well-organized, but the code could use more comments for clarity." * **Focus on Solutions:** If you identify an issue, suggest ways to improve it. * **Balance Positive and Negative Feedback:** Recognize achievements while also pointing out areas for growth.  ****6.2.4. Reflecting on the Learning Process and Outcomes**** Reflection helps you understand the broader learning experience beyond the technical aspects of a project. Ask yourself:   * What challenges did I face, and how did I overcome them? * What skills have I improved through this project? * What would I do differently in my next project?  ****6.2.5. Iterating on Projects Based on Feedback and Reflection**** The best projects are developed through multiple iterations. After receiving feedback and reflecting on the process, you can make improvements and test new ideas. This iterative process is essential in coding and engineering, where refining a project leads to better final results. |
|  |

|  |  |
| --- | --- |
| **Activity** | |
| **Activity 1** | **Hands On Application-1**  **Project Name: Crafting an Engaging Project Presentation** |
| **Learning outcomes:**  After this module you will be able to:   * Develop skills in structuring and delivering a project presentation. * Learn how to use multimedia effectively to communicate technical content. * Gain confidence in explaining technical concepts clearly.   **Description:** In this activity, students will prepare and deliver a short presentation (5-7 minutes) on their Arduino project. They will focus on structuring their presentation, integrating multimedia elements (slides, images, or videos), and demonstrating their project if possible. They will also practice public speaking skills and clarity in technical explanations.  **Level** : Intermediate – Suitable for students with basic experience in Arduino projects and presentations.  **Duration :** 90 minutes (30 minutes for preparation, 45 minutes for presentations, 15 minutes for feedback).  **Learning Subject:**   * Computer Science. * Engineering. * Communication Skills.   **Basic Skills:**   * Public speaking and communication. * Digital literacy (using presentation tools such as PowerPoint, Google Slides, or Canva). * Technical explanation and demonstration.   **Preparation & Research:**   * Review examples of effective technical presentations. * Gather multimedia materials such as images, circuit diagrams, and videos of the project in action. * Prepare a concise script outlining key points.   **Design:**  This activity is designed to guide students through the process of preparing and delivering an effective project presentation. The goal is to ensure that students can clearly communicate their Arduino project, making use of multimedia elements and, if possible, a live demonstration. The design of the activity follows a structured approach that helps students organize their thoughts, develop public speaking skills, and present technical information in an engaging way.  **Step 1: Structuring the Presentation (15 minutes)**  Before starting the presentation, students need to define a clear structure to organize their ideas. The recommended format is:   1. Introduction (1-2 minutes)  * Begin by introducing the project with a clear and engaging statement. * Explain why the project was created and what problem it aims to solve. * Mention the main objective and provide a brief overview of what the audience will learn.  1. Development Process (2-3 minutes)  * Explain the steps taken to build the project. * Use visual aids (diagrams, images, or slides) to support explanations. * Highlight any challenges encountered and how they were solved. * Provide a brief explanation of the code, if relevant, ensuring it is understandable to the audience.  1. Demonstration of the Project (1-2 minutes)  * If possible, conduct a live demonstration of the project in action. * If a live demo is not feasible, show a short video recording of the project working. * Explain what is happening during the demonstration to help the audience understand the system.  1. Conclusion (1 minute)  * Summarize the key takeaways from the project. * Reflect on what was learned, any difficulties faced, and how the experience can help in future projects. * Mention possible future improvements or next steps for the project   **Conclusion:** Students will receive feedback from their peers and instructor based on clarity, engagement, and technical explanation. The objective is to improve communication skills and ensure they can effectively present technical projects. |
| **Sources:**   TED Talks on effective presentations.   Example Arduino project presentations from YouTube or educational platforms. |
| **Extra contents:** |

|  |  |
| --- | --- |
| **Activity** | |
| **Activity 2** | **Hands On Application-2**  **Project Name:** Creating a Project Portfolio |
| **Learning outcomes:**   * Understand the importance of documenting technical work. * Learn how to structure a project report or portfolio. * Develop digital literacy skills in portfolio creation.   **Description**: Students will create a project portfolio that documents their Arduino project from start to finish. They will organize it into sections, including project objectives, materials, process, challenges, solutions, and final outcomes. The portfolio can be in digital format (PDF, website, blog, or slideshow) or physical format (printed document with photos and descriptions).  **Level :** Intermediate – Requires basic knowledge of Arduino projects and digital documentation tools.  **Duration :** 2 hours (90 minutes for creation, 30 minutes for peer review and discussion).  **Learning Subject:**   * Computer Science. * Engineering. * Communication Skills.   **Basic Skills:**   * Digital literacy (using Google Docs, Word, or portfolio platforms like Notion or GitHub). * Writing and structuring technical reports. * Organizing multimedia evidence (photos, code snippets, diagrams).   **Preparation & Research:**   * Analyze examples of well-structured project portfolios. * Gather all documentation and media related to the Arduino project. * Plan the structure of the portfolio before writing.   **Knowledge & Science:** This activity emphasizes the importance of documentation in engineering and programming. In professional and academic settings, keeping well-structured records of a project is essential for understanding progress, replicating results, troubleshooting errors, and showcasing skills. A project portfolio serves as a detailed archive of the entire development process, allowing both the creator and others to revisit the steps taken, analyze challenges, and improve upon previous work.  **Design:**  In this activity, students will create a structured project portfolio documenting their Arduino project from start to finish. The goal is to ensure that their work is well-documented, easy to understand, and professionally presented. The activity is divided into clear, progressive steps to help students effectively compile, organize, and present their project data.  **Step 1: Understanding the Structure of a Project Portfolio (15 minutes)**  Before starting, students will learn about the key sections of a portfolio and how each contributes to documenting a project effectively. The portfolio should include:   1. Title Page & Introduction: A brief overview of the project, including the title, author(s), date, and a short introduction explaining the project’s purpose. 2. Objectives & Goals: A clear statement of what the project aims to achieve. 3. Materials & Components List: A complete list of hardware and software used in the project. 4. Development Process: A step-by-step breakdown of how the project was built, including wiring diagrams, screenshots of the code, and progress images. 5. Challenges & Solutions: A section reflecting on problems encountered during development and how they were solved. 6. Final Results & Outcomes: A summary of whether the project met its objectives, including images of the final product in action. 7. Future Improvements: Suggestions for how the project could be improved in future iterations.   **Step 2: Gathering Documentation (20 minutes)**  Students will collect all relevant project data they have generated during development. This includes:   * Photographs and videos of the project at different stages. * Circuit diagrams and schematics (hand-drawn or software-generated). * Code snippets, with clear explanations of key functionalities. * Testing results and observations. * Personal notes or journal entries on difficulties faced and lessons learned.   If students haven’t documented their progress adequately so far, they will recreate missing steps, such as taking pictures of their final project or describing their coding process.  **Step 3: Organizing and Structuring the Portfolio (30 minutes)**  Students will assemble their information into a structured and visually appealing document. They can choose one of the following formats:   * A digital portfolio (PDF or website/blog format) * A physical printed portfolio   Digital Portfolios:  Students will use tools such as Google Docs, Canva, PowerPoint, Notion, or GitHub Pages to format their portfolios neatly. They will:   * Use headings and subheadings for clarity. * Insert images and diagrams to illustrate steps. * Keep text concise and professional.   Physical Portfolios:  Students who prefer a printed version will organize their work in a clear binder, ensuring:   * Pages are well-arranged and labeled. * Images and diagrams are neatly pasted with captions. * Handwritten notes are legible, or printed if possible.   **Step 4: Writing the Reflection Section (15 minutes)**  Students will reflect on their learning experience, answering:   * What was the biggest challenge I faced in this project? * How did I solve any major problems? * What skills did I improve through this project? * How would I improve the project in a second iteration?   This reflection is key to demonstrating the learning process and showing how students have grown technically and creatively.  **Step 5: Peer Review & Feedback (30 minutes)**  Each student will exchange portfolios with a peer and provide constructive feedback on:   * Clarity & Organization – Is the portfolio easy to follow? * Completeness – Are all key sections included? * Visuals & Presentation – Are images, diagrams, and formatting effective? * Reflection Quality – Does the reflection provide meaningful insights?   After receiving feedback, students will make final improvements before submitting their portfolio.  **Conclusion:** By the end of this activity, students will have created a comprehensive, structured portfolio that documents their project in a professional and engaging way. They will have improved their skills in technical writing, organization, and self-reflection, all of which are valuable in both academic and professional settings. |
| **Sources**: |
| **Extra contents:** |

|  |
| --- |
| **Activity 3 (Assessment of Module 6.1)** |
| **Type**: |
| ****Section 1: Multiple Choice Questions (1 point each)**** **1. Why is it important to present an Arduino project effectively?**   1. To impress classmates and teachers 2. To communicate the project’s purpose and process clearly 3. To increase the complexity of the project 4. To make the presentation longer   **2. What is a key element of an engaging project presentation?**   1. Using highly technical language that only experts understand 2. Avoiding any visuals or multimedia elements 3. Organizing content into a clear and structured format 4. Reading directly from slides without making eye contact   **3. When demonstrating an Arduino project during a presentation, what is the best approach?**   1. Show a video or live demonstration and explain what is happening 2. Avoid demonstrating because technical errors may occur 3. Only show images of the final project without explaining the process 4. Speak quickly to avoid questions from the audience   **4. According to Ohm's Law, what is the relationship between voltage (V) and current (I) in a circuit?**   1. To keep a detailed record of the project for future reference 2. To only serve as a requirement for the course 3. To replace the need for a live presentation 4. To limit how much information is shared about the project   **5. Which of the following should NOT be included in a project report?**   1. A list of components used in the project 2. An explanation of challenges and solutions 3. Personal opinions unrelated to the project 4. A summary of the final outcomes   **Section 2: True or False (1 point each)**  **6.** A project presentation should be detailed, but it is not necessary to structure it in sections. (True / False)  **7.** Multimedia elements like images, videos, and slides help make a presentation more engaging and clear. (True / False)  **8.** Writing a project report is useful only if the project has problems that need troubleshooting. (True / False)  **9.** When giving a presentation, you should make eye contact with your audience and speak clearly. (True / False)  **10.** A good project portfolio includes information about the project's objectives, components, process, and future improvements. (True / False)  **Section 3: Short Answer Questions (2 points each)**  **11.** What are two key tips for making a compelling project presentation?  **12.** Name two different ways to document an Arduino project effectively.  **13.** Why is it important to include challenges and solutions in a project portfolio?  **14.** What are the advantages of using a project portfolio for future learning or career opportunities?  **15.** Imagine you are presenting your Arduino project to a group of beginners. How would you explain it in a simple and engaging way? |
| **Total Score: /20 Points**  Grading Scale:  18-20 points: Excellent understanding of project presentation and documentation.  15-17 points: Good understanding with minor gaps.  12-14 points: Basic understanding, but needs improvement.  Below 12 points: Needs further review of module 6.1.  **Answers**:  Section 1: **B, C, A, A, C**  Section 2: **F, T, F, T, T** |

|  |
| --- |
| **Module 6.2** |
| **Cross-Disciplinary Applications** |
| **Introduction:**  **The Importance of Documentation and Reflection**  Beyond presenting a project, it is essential to reflect on the work done. Keeping a record of your progress helps you recognize what you have learned and areas where you can improve. Reflection is a key part of learning, as it allows you to analyze successes and mistakes, preparing you for future projects.  **Techniques for Evaluating Your Own and Your Peers' Work**  Self and peer assessment are valuable tools for improving skills and achieving better results. Some common techniques include:   * Checklists and Rubrics: Use predefined criteria to evaluate project quality. * Strengths and Areas for Improvement: Identify what works well and what can be improved. * Comparing Against Objectives: Assess whether the project meets its original goals.   **Constructive Feedback and Continuous Improvement**  Giving and receiving feedback is a skill that helps in personal and academic growth. Constructive feedback should be specific, positive, and focused on improvement. Follow these guidelines:   * Be Specific: Instead of saying "It looks good," say "The wiring is well-organized, but the code could use more comments for clarity." * Focus on Solutions: If you identify an issue, suggest ways to improve it. * Balance Positive and Negative Feedback: Recognize achievements while also pointing out areas for growth.   **Reflecting on the Learning Process and Outcomes**  Reflection helps you understand the broader learning experience beyond the technical aspects of a project. Ask yourself:   * What challenges did I face, and how did I overcome them? * What skills have I improved through this project? * What would I do differently in my next project?   **Iterating on Projects Based on Feedback and Reflection**  The best projects are developed through multiple iterations. After receiving feedback and reflecting on the process, you can make improvements and test new ideas. This iterative process is essential in coding and engineering, where refining a project leads to better final results. |

|  |  |
| --- | --- |
| **Activity** | |
| **Activity 4** | **Project Name:** Peer and Self-Assessment of Arduino Projects |
| **Learning outcomes:**   * Develop skills in constructive feedback and self-reflection. * Learn how to evaluate the quality, functionality, and creativity of an Arduino project. * Understand the importance of continuous improvement through iteration based on feedback.   **Level :** Intermediate – Suitable for students who have completed at least one Arduino project.  **Duration :** 90 minutes (30 min self-assessment, 30 min peer review, 30 min reflection and iteration).  **Learning Subject:** Application Development with Arduino   * Computer Science * Engineering * Problem-Solving * Critical Thinking   **Basic Skills:**   * Evaluation and analytical skills – Assessing project effectiveness and identifying areas for improvement. * Constructive feedback skills – Learning how to give and receive feedback in a positive way. * Reflection and iteration – Understanding how to improve based on feedback.   **Preparation & Research:**   * Review examples of project evaluation rubrics. * Discuss the importance of constructive feedback and how to phrase it positively. * Learn how to iterate on a project by making improvements after an assessment.   **Knowledge & Science:**  This activity is based on the principles of self-reflection, peer learning, and iterative development—all essential in engineering, programming, and creative problem-solving.   1. Self-Assessment: Reflecting on one’s work is key to recognizing what went well and what needs improvement. Engineers and developers constantly review their projects to optimize efficiency, performance, and user experience. 2. Peer Review: In professional settings, code and projects are frequently reviewed by colleagues. Peer feedback helps uncover issues, sparks new ideas, and enhances overall quality. 3. Iteration and Continuous Improvement: In coding, robotics, and engineering, the first version of a project is rarely the final one. Professionals use feedback loops to refine and enhance their work over multiple iterations.   By learning these techniques, students develop a growth mindset and improve their ability to analyze and refine their work based on structured feedback.  **Design:**  This activity is structured into three main steps, ensuring that students first assess their own work, then receive feedback from peers, and finally reflect on improvements.  **Step 1: Self-Assessment (30 minutes)**  Each student will evaluate their own Arduino project by answering the following questions:   * Project Functionality: Does my project work as expected? Are there any bugs or issues? * Code Quality: Is my code well-structured and commented? Could it be optimized? * Design & Documentation: Is my wiring organized? Have I documented my process properly? * Challenges Faced: What difficulties did I encounter? How did I overcome them? * Future Improvements: If I had more time, what would I improve or add to the project?   They will write their responses in a self-assessment report (1-2 pages) to organize their thoughts before the peer review.  **Step 2: Peer Review (30 minutes)**  Students will exchange projects with a classmate and evaluate each other’s work using a structured rubric.  Peer Review Criteria (Scored 1-5 for Each Section):   * Functionality: Does the project work correctly? * Clarity of Presentation: Is the project well-documented and explained? * Creativity & Innovation: Does the project show originality or problem-solving? * Code Structure: Is the code readable and well-commented? * Wiring & Hardware Setup: Is the wiring clear, organized, and safe?   Providing Constructive Feedback:   * Each student must write at least three positive points about the project. * Each student must provide at least two areas for improvement, phrased positively. * Example: Instead of saying, "Your wiring is messy," say, "Your project works well, but organizing the wiring could make it easier to troubleshoot issues."   **Step 3: Reflection and Iteration (30 minutes)**  After reviewing the peer feedback, students will:   * Read their feedback carefully and highlight key suggestions. * Reflect on how they can improve their project in the future. * Write a short reflection (5-7 sentences) summarizing:   + What they learned from the feedback.   + One specific improvement they plan to make in future projects.   + How self and peer assessment helped them understand their work better.   If time allows, students can start making small adjustments to their projects based on feedback.  **Conclusion :** Through this activity, students recognize the value of structured self and peer assessment in improving their Arduino projects. Self-evaluation enhances personal growth by fostering self-awareness, helping students identify strengths and areas for improvement in their work. Peer review cultivates collaboration, as constructive feedback from classmates offers fresh perspectives and practical suggestions. This process mirrors real-world engineering and coding practices, where continuous iteration and refinement are essential. By reflecting on received feedback, students develop a growth mindset, understanding that improvement is an ongoing process. Ultimately, this experience equips them with critical thinking, problem-solving, and communication skills that are essential for both academic and professional success. |
| **Sources:**   * Sample project assessment rubrics (can be provided by the instructor). * Example of constructive feedback techniques. * A checklist for improving Arduino projects based on common feedback. |
| **Extra contents:** |

|  |
| --- |
| **Activity 5 (Assessment of Module 6.2)** |
| **Section 1: Multiple Choice Questions (1 point each)**  **1. Why is self-assessment important in project development?**  a) It helps identify strengths and areas for improvement.  b) It is only necessary if the project fails.  c) It is used to criticize one’s own work negatively.  d) It has no impact on learning or progress.  **2. What is the primary goal of peer assessment?**  a) To point out mistakes and judge others' work.  b) To provide constructive feedback for improvement.  c) To compare projects and decide whose is better.  d) To replace self-assessment in the evaluation process.  **3. What should effective peer feedback include?**  a) Only positive comments to encourage the student.  b) A balance of strengths and areas for improvement.  c) Harsh criticism to push improvement.  d) General statements without details.  **4. Which of the following is an example of constructive feedback?**  a) “Your project is not good at all.”  b) “Your wiring is messy, and you should redo it.”  c) “Your project is great, no changes needed.”  d) “Your project is well-structured, but organizing the wiring could make troubleshooting easier.”  **5. Why is it important to iterate and improve a project based on feedback?**  a) Because every project must be perfect on the first try.  b) To apply suggestions and refine ideas for better results.  c) To satisfy the reviewer, even if the project works fine.  d) To restart the project completely from scratch.  **Section 2: True or False (1 point each)**  6. Peer assessment is meant to replace teacher evaluations. (True / False)  7. Feedback should always include specific suggestions for improvement. (True / False)  8. Constructive feedback should focus only on negative aspects. (True / False)  9. Reflecting on challenges and solutions helps improve future projects. (True / False)  10. Iteration means making modifications to a project based on feedback to enhance its functionality and design. (True / False)  **Section 3: Short Answer Questions (2 points each)**  11. Name two benefits of self-assessment in project development.  12. What are two key characteristics of constructive peer feedback?  13. Why is it important to reflect on feedback before making changes to a project?  14. How does peer assessment contribute to a project’s improvement?  15. Imagine a peer has a functional project but lacks proper documentation. How would you phrase constructive feedback to help them improve? |
| Total Score: /20 Points  Grading Scale:  18-20 points: Excellent understanding of self and peer assessment.  15-17 points: Good understanding with minor gaps.  12-14 points: Basic understanding, but improvement is needed.  Below 12 points: Needs further review of module 6.2.  **Answers:**  Section 1: **A, B, B, D, B**  Section 2: **F, T, F, T, T** |

|  |  |
| --- | --- |
| **Assessment** | |
| **Overall Assessment** | **XX/100** |