



# Technology Integration Lesson Plan

## ET 501: Introduction to Applications in Educational Technology

Teacher(s)  
Name Nicholas A Dyer

Grade Level(s) 6-12

Content Area Computer Science

Time line 2-8 hours (depending on class, and grade level, and how it fits into over all curriculum)

**Standards** (What do you want students to know and be able to do? What knowledge, skills, and strategies do you expect students to gain? Are there connections to other curriculum areas and subject area benchmarks? )

Content Standards APCSP EU 1.1, 1.2

ISTE Standards: Innovative Designer, Computational Thinker

**Overview** (a short summary of the lesson or unit including assignment or expected or possible products)

Students will learn about software development strategies with incremental development being the focus of the lesson. Students will have a game broken down for them in steps and students will need to complete each step before moving on to the next, they will end with a complete Scratch game.

**Essential Questions** (What essential question or learning are you addressing? What would students care or want to know about the topic? What are some questions to get students thinking about the topic or generate interest about the topic? What questions can you ask students to help them focus on important aspects of the topic? What background or prior knowledge will you expect students to bring to this topic and build on?)

How are large pieces of software made and managed?  
What strategies do teams go through in order to set themselves up for success?  
Students should have some programming experience so as to appreciate and understand the approach to software design and implementation.

**Assessment** (What will students do or produce to illustrate their learning? What can students do to generate new knowledge? How will you assess how students are progressing (formative assessment)? How will you assess what they produce or do? How will you differentiate products?)

Formative assessment: Speed at which student's complete cards, group evaluations, questioning students, going over their artifact at each stage of the development process.

Summative assessments: Grading number of cards completed with equal emphasis on each card.

**Resources** (How does technology support student learning? What digital tools, and resources—online student tools, research sites, student handouts, tools, tutorials, templates, assessment rubrics, etc—help elucidate or explain the content or allow students to interact with the content? What previous technology skills should students have to complete this project?)

Students will have access to previous lecture material, previous programming artifacts, and instructor support. They will also have access to the large Scratch community and the guidance and assistance that can be gained from there.

### **Instructional Plan**

**Preparation** (What student needs, interests, and prior learning provide a foundation for this lesson? How can you find out if students have this foundation? What difficulties might students have?)

Students need to be familiar with scratch and the blocks available, or be comfortable in one or more other programming languages. This project would be done part way through a class or as a higher level class with prerequisites, though this I will be aware of students' knowledge.

This assignment is highly scaffolded so as to help students through it.

**Management** (How and where will your students work? Classroom, lab, groups, etc?)

Students will work in groups of two (a group of three if necessary) so as to engage in pair programming (another software development strategy that would be covered in a short PowerPoint before the project (slides attached). Students would need to have access to the internet and the Scratch website.

**Instruction and Activities** (What instructional strategies will you use with this lesson? How will your learning environment support these activities? What is your role? What are the students' roles in the lesson? How can the technology support your teaching? What engaged and worthwhile learning activities and tasks will your students complete? How will they build knowledge and skills? Will students be expected to collaborate with each other and others? How will you facilitate the collaboration?)

Instructional strategies include, a short presentation, catch and release (demo for the students then have them practice) after which my role become that of a resource to help guide and a manager who keeps groups on task. In order to best help build the knowledge and skills students will discuss amongst themselves first then as a whole how to break down the demoed software and will then work through how I have broken it down.

**Differentiation** (How will you differentiate content and process to accommodate various learning styles and abilities? How will you help students learn independently and with others? How will you provide extensions and opportunities for enrichment? What assistive technologies will you need to provide?)

With the structure of the lesson I will be able to give individual support through the assignment and will be able to address any shortcoming students have. Also with the nature of pair programming students will be set up in such a way that enables them to help each other extensively. I will also be able to adjust the amount of "flash cards" that each group needs to complete so as to best facilitate engagement and progress.

**Closure and Reflection** (Will there be a closing event? Will students be asked to reflect upon their work? Will students be asked to provide feedback on the assignment itself? What will be your process for answering the following questions?)

- Did students find the lesson meaningful and worth completing?
- In what ways was this lesson effective?
- What went well and why?
- What did not go well and why?
- How would you teach this lesson differently?)

Students will be asked to reflect on how difficult each card is, whether or not more or less information was needed for each, and how they can incorporate incremental development into the larger picture of software development. I will be able to answer the questions based on the feedback I get from students, as well as how well the discussion groups go and my time with each individual group.

**Template adapted from ISTE.org**

Lesson Plan in separate document labeled "ScratcSPD-2.docx"  
Pair programming power point and assignment flashcards attached