



**DigiEduHack Solution**  
**Aalto Main Event - Making ICT**  
**education accessible for all**  
**Challenge: Help us find tools for**  
**making ICT education accessible for**  
**adult learners**

## **RainbowIDE**

### **Thematic color for code**

Helping dyslexics and visual learners see code structure and not get lost in the details by thematically coloring code, minimizing common elements and bringing out what makes this particular piece of code unique.

### **Team: RainbowIDE**

#### **Contact details**

minna.kivihalme@aalto.fi

### **Solution Details**

#### **Solution description**

By coloring key parts of code in themes, it is easier for visual learners and dyslexics to not get lost in the text and to see the big picture (the code architecture, which really matters) instead.

#### **Solution context**

Code is text-dense and hard for certain groups to parse.

#### **Solution target group**

Dyslexics, visual learners, people who lack a freakish knack for manipulating abstract monochrome symbols

## **Solution impact**

This solution would bring in groups that are normally hesitant, overwhelmed by code as a wall of text, and therefore feel like it isn't for them.

The impact could be measured by testing how many self-identified visual learners and dyslexics no longer drop off the pipeline of code learning.

## **Solution tweet text**

Thematic color highlighting is the new syntax highlighting. Don't just read code, understand it. Follow the big picture instead of getting lost in the weeds. And yes, this is for you too, art school dropout.

## **Solution innovativeness**

Syntax highlighting exists. Thematic highlighting currently does not. The closest thing to it is VSCode's ability to highlight a single function throughout the code at a time.

## **Solution transferability**

Thematic highlighting can also be used with regular text to keep certain details straight or help dyslexic kids not fear the words coming up in a long document (by coloring easy words green, for example). This code-based prototype was built on a text-based tool that I also developed that does this. ([www.rainbowedit.com](http://www.rainbowedit.com))

## **Solution sustainability**

A proof-of-concept prototype exists. The next step is to make a version 1.0 with core functionality, custom built for this purpose, not built on something designed for another purpose. In the mid term, this is a code analysis tool for advanced beginners. Long term it could also become an IDE for advanced beginners.

## **Solution team work**

well and yes