

Jari Köngäs

Prabhjot Singh

Oulu University of Applied Sciences

contents

[1. Introduction 3](#_Toc532755560)

[2. THE WORK ENVIRONMENT 3](#_Toc532755561)

[3. Definition 4](#_Toc532755562)

[4. Implementation 4](#_Toc532755563)

[5. Testing 5](#_Toc532755564)

[6. POSSIBILITIES OF FURTHER DEVELOPMENT 5](#_Toc532755565)

[7. conclusion 6](#_Toc532755566)

[REFERENCES 7](#_Toc532755567)

[APPENDICES 7](#_Toc532755568)

# 1. Introduction

We were assigned to create a web-based mathematics application with six functionalities: The first one allows the end-user to convert numbers between binary, octa decimal, decimal and hexadecimal number systems, the second tool prints a conversion table showing numbers 0 to 50 in the aforementioned number bases, the third tool calculates combinations and permutations, fourth page generates truth tables, fifth tool tests random number distributions and the sixth tool was an optional task . We decided to go with animation so our sixth tool draws animations on the HTML Canvas element based on hardcoded polynomial equations[1].

# 2. THE WORK ENVIRONMENT

We worked on the project both in the school premises and from our homes, where we gathered together to work on the project.

We used GitHub for version control and Visual studio code as an integrated development environment (IDE) . In addition, we used the browsers called Google Chrome and Mozilla Firefox to see how our website works and looks in an actual browser. Visual Studio Code is a free code editor software with support for debugging, task running and version control tools. It also has possibilities for a variety of other kinds of extensions. We used a third party extension for auto saving of the files we were working on. For the sixth tool we also used an HTML element called “Canvas”. Canvas[2] is a HTML element which draws graphics to a web page ,with the use of JavaScript, along with its other functionalities .

# 3. Definition

**Convocal** lets the user to do a variety of mathematical tasks, including number base conversion, truth table calculation, random number generation and counting combinations and permutations.

The number base conversion takes a user input in one number system and converts it into the selected number base automatically.

The truth table also takes user input as the statement for which the truth table is then computed. The result is then printed on the standard output in form of a table.

The sixth tool uses Canvas element for HTML.

Our solution does not use any external dependency whatsoever.

# 4. Implementation

The whole project was programmed only with HyperText Markup Language (HTML), Cascading Style Sheets (CSS) and JavaScript (JS). HTML was used to define the layout of the webpages, CSS was used to style those pages and JS was used to give functionality to the said pages.

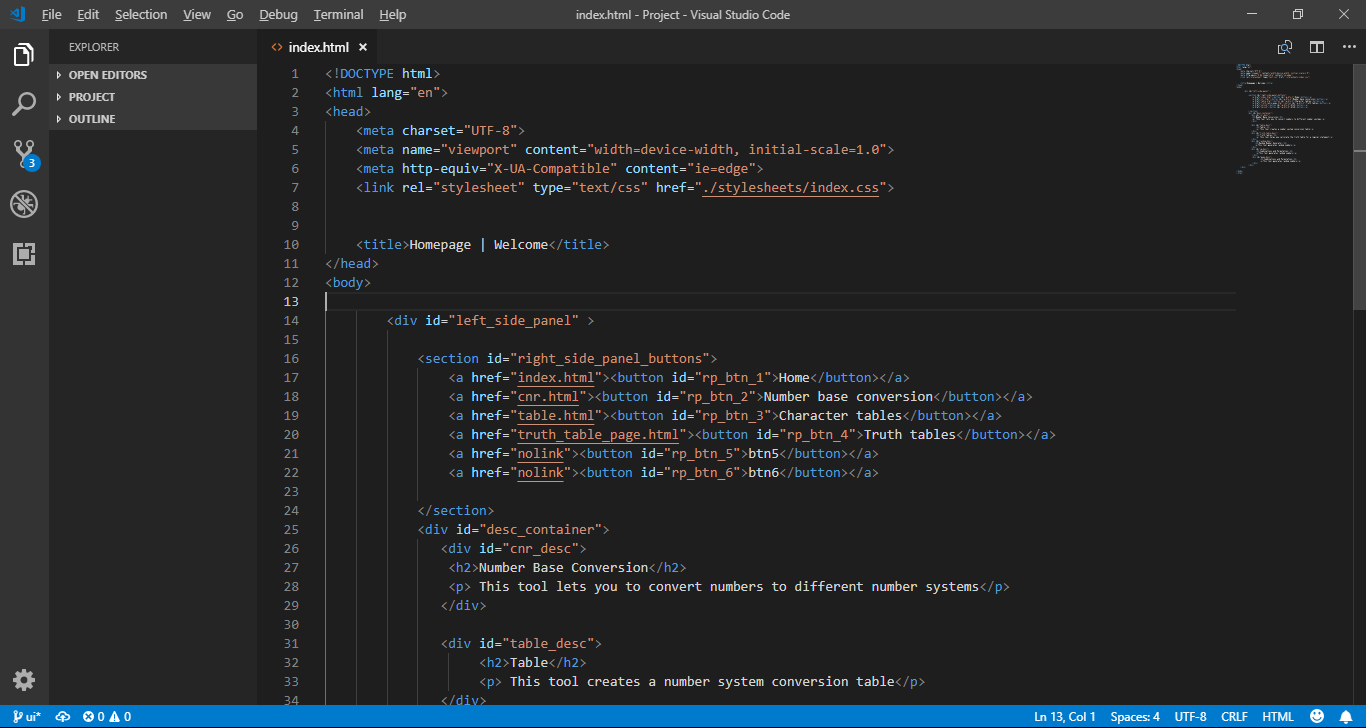
Each tool has it’s own html file which is openable with any kind of internet browser, and each html file has linked css and javascript files which contain the stylesheets and scripts for the page in question. The HTML, JS and CSS files were written using the Visual Studio Code integrated development environment. 

Figure 1: Visual Studio Code

The art used in the project, logos, icons etc were created with Adobe Illustrator.

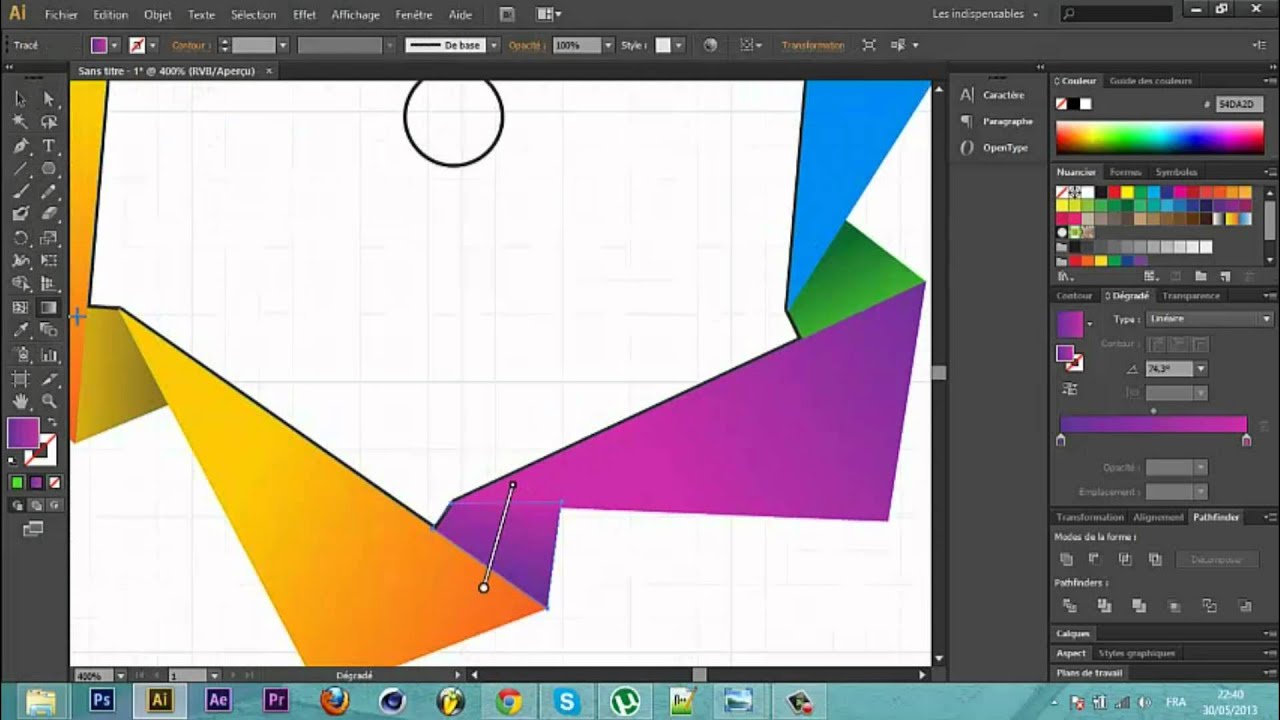


Figure 2: Adobe Illustrator

# 5. Testing

The project was tested by the two of us developers with the use of both Mozilla Firefox and Google Chrome browsers. We used the browsers’ inspect element tools to also see how the tools would function and how the pages would show when viewed on a smaller screen, like tablets and smartphones. We also used a tool called “console” to debug the scripts whenever we ran into any problems.

During the testing we found out quite a few problems in how we initially had started working on the project, and we had to redo some of the features and redesign most of the site, so that we could make the whole website work and show optimally on a larger variety of devices including desktops, laptops, tablets, TVs and of course mobile phones.

# 6. POSSIBILITIES OF FURTHER DEVELOPMENT

The Convocal project could be, in the future, developed to add more functions into it, since right now it has only six mathematical tools included. We made sure to keep that is mind when developing the scripts. We’ve made is comparatively easy to add new functions to the application.

Possibilities for future additions include –

* A statistics tool which would make various graphs out of statistics or fractions.
* A tool to automatically solve algebraic equations.
* A scientific calculator.
* A feature that saves all your work so next time you login you could continue where you left.

The project could be expanded into other kinds of mathematical disciplines as well, like series, or exponents.

# 7. conclusion

The biggest inconveniences and challenges we faced with the project were with making the website responsive. At one point during development, we had to redesign the project extensively and we lost some progress. From this we learned to make the product viewable on mobile and tablet devices from the very start, as trying to make a website responsive for mobile devices afterwards was significantly harder.

The overall UI was also not easy as other applications make it seem. We needed something that could be consistent throughout the website and would be easy on eyes and yet captivating and inspiring enough for people to actually try it.

We also noticed that styling is, in actuality, significantly harder than it looks. Choosing colours, for instance, was a surprisingly hard task, and in the end, we opted to use an example palette from the internet for our project.

We learned using various tools like Git in practical situations. We definitely sharpened our skills.

With the relatively independent nature of the project, we were able to gain valuable experience with planning a project, even though many of our initial plans had to be later reviewed and changed. We also learned about teamwork and how it is important to understand what the other people in the group are working with and why in order to make the finished product coherent and unified.

In the end, while we did succeed in the project goals, we are still not quite satisfied with the website, and would like to improve its appearance, design and functionality.

# REFERENCES

1. Polynomial, <https://en.wikipedia.org/wiki/Polynomial>, Wikipedia, read 16.12
2. HTML5 Canvas, <https://www.w3schools.com/HTML/html5_canvas.asp>, W3Schools, read 15.12
3. Express, <http://expressjs.com/>, Node.js Foundation, read 16.12

Figure 1: Visual Studio Code, by Microsoft. Image by Jari Köngäs.

Figure 2: Adobe Illustrator, by Adobe Systems, image from: https://www.windows10download.com/adobe-illustrator-cs6/

# APPENDICES

We tried to use proper indentation while writing code which made it easier to read later and modify whenever needed .We used camel casing to name our variables and functions. We custom designed all the icons used in the application . They are presented below-

