**Project Summary**

The project represents a web application for a food center company that is populated with all sorts of different foods that can be purchased by a client. The client is allowed to select a number of products that he wants to purchase.

The purpose of the project was to create a User Friendly Interface for the user for a simple task. He can do it online instead of having to go in downtown to eat. The user can just simply order it online and have it delivered.

Regarding the technologies I’ve used to develop tor this application is mainly from a front-end point of view. Some of the languages that I’ve used to achieve this are Typescript, HTML, CSS, Bootstrap using the platform Angular.

**Note:** To run this application the instruction from the file “README.md” needs to be followed thoroughly.

1. **Introduction**
   1. **Purpose**

The purpose of this project is to provide to the user with an easy to use interface for doing a simple task: to order online. The user is allowed to order whatever the food center can provide and the amount he wants.

* 1. **Motivation**

The motivation to make such an application started from the though of a daily basis needs of an individual. It is aimed for every user that wants to order online and receive their order without having the necessity to cook it or going out.

* 1. **Tasks**

The tasks that the application is suppose to do is to receive details about the food made in code. Then the user can either select the food he likes or he can filter by one of the tags, or search it by name. Once he selected his food he can then add it to the cart and purchase it.

1. **Technologies and tools used**

Most of my technologies and tools used in this project were mainly learned in college, but I’ve had to study more about in detail to develop the application in a simpler way.

**Technologies/Tools** used are:

* 1. **TypeScript Language**

TypeScript is a [free and open source](https://en.wikipedia.org/wiki/Free_and_open_source) [programming language](https://en.wikipedia.org/wiki/Programming_language) developed and maintained by [Microsoft](https://en.wikipedia.org/wiki/Microsoft). It is a strict syntactical [superset](https://en.wikipedia.org/wiki/Superset) of [JavaScript](https://en.wikipedia.org/wiki/JavaScript) and adds optional [static typing](https://en.wikipedia.org/wiki/Static_typing) to the language. It is designed for the development of large applications and [transpiles](https://en.wikipedia.org/wiki/Source-to-source_compiler" \o "Source-to-source compiler) to JavaScript. As it is a superset of JavaScript, existing JavaScript programs are also valid TypeScript programs.

TypeScript may be used to develop JavaScript applications for both [client-side](https://en.wikipedia.org/wiki/Client-side) and [server-side](https://en.wikipedia.org/wiki/Server-side) execution.

Strictly for my project I’ve had to study a lot about this language and develop in it, so it was the most used technologies. Later on this document I will describe more about used this language, but as short overview I’ve used it mainly for function purposes and routing.

* 1. **HTML(HyperText Markup Language)**

The HyperText Markup Language or HTML is the standard [markup language](https://en.wikipedia.org/wiki/Markup_language) for documents designed to be displayed in a [web browser](https://en.wikipedia.org/wiki/Web_browser). It can be assisted by technologies such as [Cascading Style Sheets](https://en.wikipedia.org/wiki/Cascading_Style_Sheets) (CSS) and [scripting languages](https://en.wikipedia.org/wiki/Scripting_language) such as [JavaScript](https://en.wikipedia.org/wiki/JavaScript). [Web browsers](https://en.wikipedia.org/wiki/Web_browser) receive HTML documents from a [web server](https://en.wikipedia.org/wiki/Web_server) or from local storage and [render](https://en.wikipedia.org/wiki/Browser_engine) the documents into multimedia web pages. HTML describes the structure of a [web page](https://en.wikipedia.org/wiki/Web_page) [semantically](https://en.wikipedia.org/wiki/Semantic_Web) and originally included cues for the appearance of the document.

The usage of this language on my project was to create all sorts of elements or add to an existing element a script(function/functionality) that was developed using the TypeScript language and then later styled using CSS.

* 1. **CSS(Cascading Style Sheets)**

Cascading Style Sheets (CSS) is a [style sheet language](https://en.wikipedia.org/wiki/Style_sheet_language) used for describing the [presentation](https://en.wikipedia.org/wiki/Presentation_semantics) of a document written in a [markup language](https://en.wikipedia.org/wiki/Markup_language) such as [HTML](https://en.wikipedia.org/wiki/HTML) or [XML](https://en.wikipedia.org/wiki/XML). CSS is a cornerstone technology of the [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web), alongside HTML and [JavaScript](https://en.wikipedia.org/wiki/JavaScript).

CSS is designed to enable the separation of presentation and content, including [layout](https://en.wikipedia.org/wiki/Page_layout), [colors](https://en.wikipedia.org/wiki/Color), and [fonts](https://en.wikipedia.org/wiki/Typeface).[[3]](https://en.wikipedia.org/wiki/CSS#cite_note-3) This separation can improve content [accessibility](https://en.wikipedia.org/wiki/Accessibility); provide more flexibility and control in the specification of presentation characteristics; enable multiple [web pages](https://en.wikipedia.org/wiki/Web_page) to share formatting by specifying the relevant CSS in a separate .css file, which reduces complexity and repetition in the structural content.

In this project the usage of CSS was to create a style for every element I’ve made using the HTML. These styles can represent the shape, color, background image, font, margin, how the elements are displayed and so on. Essentiality anything that is different(a text for example) from a simple font starting from top-left corner in a straight line is modified using the CSS.

* 1. **GIT**

Git is [free and open source software](https://en.wikipedia.org/wiki/Free_and_open_source_software) for [distributed version control](https://en.wikipedia.org/wiki/Distributed_version_control): tracking changes in any set of [files](https://en.wikipedia.org/wiki/Computer_file), usually used for coordinating work among [programmers](https://en.wikipedia.org/wiki/Programmer) collaboratively developing [source code](https://en.wikipedia.org/wiki/Source_code) during [software development](https://en.wikipedia.org/wiki/Software_development). Its goals include speed, [data integrity](https://en.wikipedia.org/wiki/Data_integrity), and support for distributed, non-linear workflows.

For my project GIT was very helpful and was also a technology that I had to learn more about since I wasn’t familiarized to work with it. But once I worked more with it was so much easier because it helps me store a version of my code and if I had a problem with a code, I can go to a previous version it worked the way I wanted and continue from there with a new implementation.

* 1. **Angular**

Angular is a [TypeScript](https://en.wikipedia.org/wiki/TypeScript)-based [free and open-source](https://en.wikipedia.org/wiki/Free_and_open-source_software) [web application framework](https://en.wikipedia.org/wiki/Web_framework) led by the Angular Team at [Google](https://en.wikipedia.org/wiki/Google) and by a community of individuals and corporations. Angular is a complete rewrite from the same team that built [AngularJS](https://en.wikipedia.org/wiki/AngularJS).

It aimed to simplify both the development and the [testing](https://en.wikipedia.org/wiki/Software_testing) of such applications by providing a framework for client-side [model–view–controller](https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller) (MVC) and [model–view–viewmodel](https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93viewmodel) (MVVM) architectures, along with components commonly used in [web applications](https://en.wikipedia.org/wiki/Web_application) and [progressive web applications](https://en.wikipedia.org/wiki/Progressive_web_application).

The AngularJS framework worked by first reading the [Hypertext Markup Language](https://en.wikipedia.org/wiki/Hypertext_Markup_Language) page, which had additional custom [HTML attributes](https://en.wikipedia.org/wiki/HTML_attribute) embedded into it. Angular interpreted those attributes as [directives](https://en.wikipedia.org/wiki/Directive_(programming)) to bind input or output parts of the page to a model that is represented by standard JavaScript [variables](https://en.wikipedia.org/wiki/Variable_(computer_science)). The values of those JavaScript variables could be manually set within the code or retrieved from static or dynamic [JSON](https://en.wikipedia.org/wiki/JSON) resources.

In my project the Angular was the main resource to produce all the components and files that are seen in the project. Those files were first just a skeleton-code and then they had to be develop further with proper functions and adding elements. These components had to be created using a windows terminal or the Visual Studii Code terminal. My choice was to use the Windows CMD(Command Prompt).

* 1. **Visual Studio Code**

Visual Studio Code, also commonly referred to as VS Code, is a [source-code editor](https://en.wikipedia.org/wiki/Source-code_editor) made by [Microsoft](https://en.wikipedia.org/wiki/Microsoft) for [Windows](https://en.wikipedia.org/wiki/Windows), [Linux](https://en.wikipedia.org/wiki/Linux) and [macOS](https://en.wikipedia.org/wiki/MacOS). Features include support for [debugging](https://en.wikipedia.org/wiki/Debugging), [syntax highlighting](https://en.wikipedia.org/wiki/Syntax_highlighting), [intelligent code completion](https://en.wikipedia.org/wiki/Intelligent_code_completion), [snippets](https://en.wikipedia.org/wiki/Snippet_(programming)), [code refactoring](https://en.wikipedia.org/wiki/Code_refactoring), and embedded [Git](https://en.wikipedia.org/wiki/Git). Users can change the [theme](https://en.wikipedia.org/wiki/Theme_(computing)), [keyboard shortcuts](https://en.wikipedia.org/wiki/Keyboard_shortcut), preferences, and install [extensions](https://en.wikipedia.org/wiki/Plug-in_(computing)) that add additional functionality.

Visual Studio Code is a source-code editor that can be used with a variety of programming languages, including [Java](https://en.wikipedia.org/wiki/Java_(programming_language)), [JavaScript](https://en.wikipedia.org/wiki/JavaScript), [Go](https://en.wikipedia.org/wiki/Go_(programming_language)), [Node.js](https://en.wikipedia.org/wiki/Node.js), [Python](https://en.wikipedia.org/wiki/Python_(programming_language)), [C++](https://en.wikipedia.org/wiki/C%2B%2B), [C](https://en.wikipedia.org/wiki/C_(programming_language)).

Visual Studio Code includes basic support for most common programming languages. This basic support includes [syntax highlighting](https://en.wikipedia.org/wiki/Syntax_highlighting), [bracket matching](https://en.wikipedia.org/wiki/Bracket_matching), [code folding](https://en.wikipedia.org/wiki/Code_folding), and configurable snippets. Visual Studio Code also ships with [IntelliSense](https://en.wikipedia.org/wiki/Intelligent_code_completion) for JavaScript, TypeScript, [JSON](https://en.wikipedia.org/wiki/JSON), [CSS](https://en.wikipedia.org/wiki/CSS), and [HTML](https://en.wikipedia.org/wiki/HTML), as well as debugging support for Node.js. Support for additional languages can be provided by freely available extensions on the VS Code Marketplace.

Using Visual Studio Code was a personal choice that I preferred and I would recommend it because it is an User Friendly Environment where you can edit the code or can even synchronize it fast with GIT. The highlights(visualization of the color) alone on important keywords are a very important feature because you can develop easier and see things clearer.

* 1. **Node.js**

Node.js is an [open-source](https://en.wikipedia.org/wiki/Open-source_software), [cross-platform](https://en.wikipedia.org/wiki/Cross-platform), [back-end](https://en.wikipedia.org/wiki/Front_end_and_back_end) [JavaScript](https://en.wikipedia.org/wiki/JavaScript) [runtime environment](https://en.wikipedia.org/wiki/Runtime_system) that runs on a [JavaScript Engine](https://en.wikipedia.org/wiki/JavaScript_Engine) and executes JavaScript code outside a [web browser](https://en.wikipedia.org/wiki/Web_browser), which was designed to build scalable network applications.

Node.js lets developers use JavaScript to write command line tools and for [server-side scripting](https://en.wikipedia.org/wiki/Server-side_scripting) - running scripts server-side to produce [dynamic web page](https://en.wikipedia.org/wiki/Dynamic_web_page) content before the page is sent to the user's web browser. Consequently, Node.js represents a "JavaScript everywhere" paradigm, unifying [web-application](https://en.wikipedia.org/wiki/Web_application) development around a single programming language, rather than different languages for server-side and client-side scripts.

* 1. **Bootstrap**

Bootstrap is a [free and open-source](https://en.wikipedia.org/wiki/Free_and_open-source) [CSS framework](https://en.wikipedia.org/wiki/CSS_framework) directed at responsive, [mobile-first](https://en.wikipedia.org/wiki/Responsive_web_design#Mobile_first,_unobtrusive_JavaScript,_and_progressive_enhancement) [front-end web development](https://en.wikipedia.org/wiki/Front-end_web_development). It contains [HTML](https://en.wikipedia.org/wiki/HTML), [CSS](https://en.wikipedia.org/wiki/CSS) and [JavaScript](https://en.wikipedia.org/wiki/JavaScript)-based design templates for [typography](https://en.wikipedia.org/wiki/Web_design#Typography), [forms](https://en.wikipedia.org/wiki/Form_(HTML)), [buttons](https://en.wikipedia.org/wiki/Button_(computing)#HTML), [navigation](https://en.wikipedia.org/wiki/Web_navigation#Local_website_navigation), and other interface components.

Bootstrap is an HTML, CSS & JS Library that focuses on simplifying the development of informative web pages. The primary purpose of adding it to a web project is to apply Bootstrap's choices of color, size, font and layout to that project. As such, the primary factor is whether the developers in charge find those choices to their liking. Once added to a project, Bootstrap provides basic style definitions for all [HTML elements](https://en.wikipedia.org/wiki/HTML_element).

* 1. **JavaScript**

JavaScript  (JS) is a [programming language](https://en.wikipedia.org/wiki/Programming_language) that is one of the core technologies of the [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web), alongside [HTML](https://en.wikipedia.org/wiki/HTML) and [CSS](https://en.wikipedia.org/wiki/CSS). All major [web browsers](https://en.wikipedia.org/wiki/Web_browser) have a dedicated [JavaScript engine](https://en.wikipedia.org/wiki/JavaScript_engine) to execute the [code](https://en.wikipedia.org/wiki/Source_code) on [users](https://en.wikipedia.org/wiki/User_(computing))' devices.

JavaScript is a [high-level](https://en.wikipedia.org/wiki/High-level_programming_language), often [just-in-time compiled](https://en.wikipedia.org/wiki/Just-in-time_compilation) language. It has [dynamic typing](https://en.wikipedia.org/wiki/Dynamic_typing), [prototype-based](https://en.wikipedia.org/wiki/Prototype-based_programming) [object-orientation](https://en.wikipedia.org/wiki/Object-oriented_programming), and [first-class functions](https://en.wikipedia.org/wiki/First-class_function). It is [multi-paradigm](https://en.wikipedia.org/wiki/Programming_paradigm), supporting [event-driven](https://en.wikipedia.org/wiki/Event-driven_programming), [functional](https://en.wikipedia.org/wiki/Functional_programming), and [imperative](https://en.wikipedia.org/wiki/Imperative_programming) [programming styles](https://en.wikipedia.org/wiki/Programming_paradigm). It has [application programming interfaces](https://en.wikipedia.org/wiki/Application_programming_interface) for working with text, dates, [regular expressions](https://en.wikipedia.org/wiki/Regular_expression), standard [data structures](https://en.wikipedia.org/wiki/Data_structure), and the [Document Object Model](https://en.wikipedia.org/wiki/Document_Object_Model).

JavaScript engines were originally used only in web browsers, but are now core components of some [servers](https://en.wikipedia.org/wiki/Server_(computing)) and a variety of [applications](https://en.wikipedia.org/wiki/Application_software).