**Back-End Framework Choice**

# Introduction

The aim of this document is to settle which framework would be suited best for the Atos Group Project [re.shape] is conducting during Semester 6 at Fontys ICT.

The MVP of the project is to build a working prototype of the Business Card Game Atos has internally developed, so that the process of playing the game becomes seamless and fully available online.

Contents

[Introduction 1](#_Toc96602945)

[1. Description 2](#_Toc96602946)

[2. Tools of choice 2](#_Toc96602947)

[3. Tools comparison 3](#_Toc96602948)

[4. Framework Selection 4](#_Toc96602949)

[5. Conclusion 4](#_Toc96602950)

# Description

This document focuses on finding a number of back-end suited tools/ frameworks that would best satisfy the needs to complete the Card Business Game project. Throughout the chapters, the main and most popular available tools will be mentioned in the document. An analysis/ comparison between tools will be done to see the pros and cons of each tool and come to a conclusion on the best tool that suits this project. The focus will be mainly on the technical part, as the tools/ frameworks are targeted to be free, therefore no costs are involved.

# Tools of choice

There are a number of tools and frameworks on the internet to be used in the creation of a project. They are meant to satisfy different kind of needs for different types of projects. These tools, of course, require experience and knowledge from the programmers to be implemented correctly, as they are working based on different programming languages. Some of the tools or frameworks that could be suited for this project are:

* **Node.JS** – is an asynchronous event-driven JavaScript runtime, designed to build scalable network applications.
* **PHP** - is a general-purpose scripting language geared towards web development. PHP code is usually processed on a web server by a PHP interpreter implemented as a module, a daemon or as a Common Gateway Interface (CGI) executable.
* **Django** - is a high-level Python web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel. It’s free and open source.
* **Laravel –** is a PHP based framework meant for web development projects with built-in support. It is to mention that Laravel has a large community, making it easier to solve common problems.
* **Spring** **Boot -** is an open-source Java-based framework used to create micro-Services. It is used to build stand-alone and production ready spring applications.

It is of course to mention that there are a high number of tools and frameworks available on the internet that could be suited for this project, but they might imply costs, less documentation, a smaller community. These factors can slow down the process of development, specially for a junior team.

# Tools comparison

The main focus of the comparison will not be only on the technical advantages of the tools and frameworks, but also on the technical capabilities and experience of the team members. Upon a short voting session, it was decided that the majority preferred Node.JS, but is that actually the better choice? During this chapter of the document a comparison of the advantages and disadvantages between the previously selected tools and frameworks will be done, looking at different properties that each tool should have in order to be relevant for this project.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Property | PHP | Django | Laravel | Spring Boot | Node.JS |
| Open-source server-sided | √ | √ | √ | √ | √ |
| Asynchronous | - | √ | √ | √ | √ |
| Suited for both Frond-end and Back-end | - | - | - | - | √ |
| Has Great Security Implemented | - | √ | √ | √ | - |
| Ideal for NoSQL DBs | - | - | - | √ | √ |
| Developed/ Active Community | - | ~ | √ | √ | √ |
| Good I/O Model | √ | - | - | √ | √ |
| Multi-threading | √ | √ | √ | √ | - |
| Low Memory Usage | √ | √ | - | - | √ |
| Complexity | medium | medium | medium | high | Low |
| Popularity | high | high | medium | high | Medium (increasing) |

# Framework Selection

The framework that we are going to be selecting for this project will be based on Node.JS, as mentioned initially in the document, the team has the most experience with it and as shown in the table at point 3, it achieves most of the features that we are looking for.

The framework that was selected to be used during the project is [NestJS](https://nestjs.com/) as it is an extensible, versatile and progressive framework. Aside from the already mentioned advantages of Node.JS, this framework is the fastest growing of all of the same type, it has a powerful CLI for productivity-boosting and facilitate development with ease, it has support for dozens of nest-specific modules that facilitate easy integration with common technologies and concepts such as GraphQL, TypeORM, Mongoose, Logging, Caching, Validation or others, it provides easy unit-testing and many other advantages.

# Conclusion

After analyzing the advantages and disadvantages of each tool in the table at the previous point we can clearly see that the best suited tool for this project is Node.JS. Our final decision will therefore be based on Node.JS, adding the NestJS framework to be used during the development as it further makes the process faster and easier.