# Projeto Final em Engenharia de Computação
# Especificação dos Requisitos do Sistema

## 1. Introdução

Este documento apresenta uma descrição mais detalhada dos requisitos do sistema de software a fim de identificar os requisitos funcionais do sistema e determinar a especificação em relação a algumas características do sistema de software do projeto final. Além de identificar as qualidades do sistema. Dessa forma, atendendo a essa especificação possa ser evitado alguns problemas que podem surgir na engenharia de software.

## 2. Requisitos Funcionais do Sistema

Os requisitos funcionais do sistema indicam reações que o sistema deve ter tanto em um aspecto geral como em um aspecto específico. No sistema de software deste projeto como requisitos funcionais tem-se como principais que:

## 3. Qualidades do Sistema

### 3.1 Usabilidade

Essa propriedade reflete quão fácil é usar o sistema. Depende dos componentes técnicos de sistema, seus operadores e seu ambiente operacional.

### 3.2 Confiabilidade

A confiabilidade de sistema depende da confiabilidade de componentes, mas interações inesperadas podem causar novos tipos de falhas e, portanto, afetar a confiabilidade do sistema.

### 3.3 Suportabilidade

A escalabilidade do sistema de software pode ser avaliada visto que um sistema reflete sua capacidade de oferecer um serviço de alta qualidade, uma vez que aumenta a demanda de sistema. Segundo, Neuman (1994) são identificas três dimensões da escalabilidade:

1. Tamanho: Deve ser possível adicionar mais recursos a um sistema para tratar com aumento de usuários utilizando o sistema de software.
2. Distribuição: Deve ser possível dispersar geograficamente os componentes de um sistema sem comprometer seu desempenho.
3. Capacidade de gerenciamento: É possível gerenciar um sistema à medida que ele aumenta de tamanho, mesmo que partes dele estejam localizadas em organizações independentes do sistema.

Diante dessas necessidades, tem-se que nesse projeto será analisada a capacidade de gerenciamento alinhado com o seu tamanho. Será desenvolvido um sistema que garanta uma escalabilidade devido ao aumento de usuários utilizando o sistema de software.

# 4. Interfaces do Sistema

*[Interface Requirements are part of the + in the FURPS+ classification of supporting requirements. Define the interfaces that must be supported by the application. It should contain adequate specificity, protocols, ports and logical addresses, and so forth, so that the software can be developed and verified against the interface requirements.]*

## 4.1 Interfaces do Usuário

*[Describe the user interfaces that are to be implemented by the software. The intention of this section is to state requirements relating to the interface. Interface design may overlap the requirements gathering process.]*

### *4.1.1 Look & Feel*

*[Provide a description of the spirit of the interface. Your client may have given you particular demands such as style, colors to be used, and degree of interaction and so on. This section captures the requirements for the interface rather than the design for the interface.]*

### *4.1.2 Layout and Navigation Requirements*

*[Capture requirements on major screen areas and how they should be grouped together.]*

### *4.1.3 Consistency*

*[Consistency in the user interface enables users to predict what will happen. This section states requirements on the use of mechanisms to be employed in the user interface. This applies both within the system and with other systems and can be applied at different levels: navigation controls, screen areas sizes and shapes, placements for entering / presenting data, terminology.]*

### *4.1.4 User Personalization & Customization Requirements*

*[Requirements on content that should automatically displayed to users or available based on user attributes. Sometimes users allowed to customize the content displayed or to personalize displayed content.]*

## 4.2 Interfaces to External Systems or Devices

*[Are there any external systems with which this system must interface? Are there any constraints on the nature of the interface between this system and any external system, such as the format of data passed between these systems, and any particular protocol used? Consider both provided and required interfaces.]*

### *4.2.1 Software Interfaces*

*[This section describes software interfaces to other components of the software system. These may be purchased components, components reused from another application or components being developed for subsystems outside of the scope of this SRS, but with which this software application must interact.]*

### *4.2.2 Hardware Interfaces*

*[This section defines any hardware interfaces that are to be supported by the software, including logical structure, physical addresses, expected behavior, and so on.]*

### *4.2.3 Communications Interfaces*

*[Describe any communications interfaces to other systems or devices such as local area networks, remote serial devices, and so on.]*

# 5. Regras de Negócio

*[Business rules are statements that define or constrain some aspect of the business. Business rules are often represented as production rules when they are meant to be directly executed by an IT System: a production rule is an independent statement of programming logic that specifies the execution of one or more actions in the case that its conditions are satisfied. Production Rules define the operation semantic for the system in a technologic independent way. They constrain the behavior expressed in system use cases.*

*Organize this document on rule classes, a high level grouping of candidate or actual rules about one **business concept** with a specific kind of **logic processing**, example: Driver Risk Assessment Rules or Customer Validation Rules.]*

## 5.1 <Rule class name>

### *5.1.1 <Rule name and ID>*

*[The description defines the rule. It can be made in natural language typically following a decision table or a pattern like: if [condition-list] then [action-list], example:*

*If there are at least 3 items of the same type in the customer shopping cart and each item's value is greater than $30 then give to the customer a voucher whose value is 10% of the cheapest item.]*

# 6. System Constraints

*[Constraints are part of the + in the FURPS+ classification of supporting requirements. Describe any design; implementation or deployment constraints on the system being built that have been mandated and must be adhered to. Examples include software implementation languages, prescribed use of developmental tools, third-party components or class libraries, platform support, resource limits and requirements on the shape, size or weight of the resulting hardware housing the system.]*

# 7. System Compliance

## 7.1 Licensing Requirements

*[Define any licensing enforcement requirements or other usage restriction requirements that are to be exhibited by the software.]*

## 7.2 Legal, Copyright, and Other Notices

*[This section describes any necessary legal disclaimers, warranties, copyright notices, patent notice, wordmark, trademark, or logo compliance issues for the software.]*

## 7.3 Applicable Standards

*[This section describes by reference any applicable standards and the specific sections of any such standards that apply to the system being described. For example, this could include legal, quality and regulatory standards, industry standards for usability, interoperability, internationalization, operating system compliance, and so forth.]*

# 8. System Documentation

*[Describes the requirements, for on-line user documentation, help systems, help about notices, and so on. Set expectations for the documentation and to identify who will be responsible for creating it.]*