|  |
| --- |
|  |

**Unbabel**

Test Planning

Finish Annotation

Author: (Herbert.rlopes@gmail.com)

|  |  |
| --- | --- |
| Author : Herbert Lopes | Signature :  Date : |
| Approved by : Emanuel Velho  Institute : | Signature :  Date : |
| Released by :  Institute : | Signature :  Date : |
|  | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Description of Change** | **Author** | **Date** |
| 1.0 | Closing the scope with definition of all requirements to be implemented | Herbert Lopes | 07/05/2018 |
| 1.0 | General Document Review | Herbert Lopes | 08/05/2018 |
|  |  |  |  |
|  |  |  |  |

**CONTENTS**

1 INTRODUCTION 4

1.1 Purpose 4

1.2 Scope 4

1.3 Background 4

1.4 References 4

2 METHODOLOGY 5

3 FUNCTIONAL REQUIREMENTS 5

4.1 Context 5

4.2 User Requirements 5

4.3 Data Flow Diagrams 6

4.4 Logical Data Model/Data Dictionary 6

4.5 Functional Requirements 6

5 OTHER REQUIREMENTS 6

5.1 Interface Requirements 6

5.2 Data Conversion Requirements 7

5.3 Hardware/Software Requirements 7

5.4 Operational Requirements 7

11

1. **INTRODUCTION**

This document specifies the requirements of the Annotation Tools providing developers and testers with the necessary information for the design and implementation to perform system testing and approval.

## **Purpose**

Presents an overview of the system, characterizing its scope and describing its users.

## **Scope**

The test that will be covered by this document will be: Finish annotation process.

## **Background**

Unbabel is a translation service supported by technology. It is a platform that combines a Machine Translation Engine with a crowd of Human Editors, empowering communication in many different languages. They developed this document to validate all the specification to perform annotation in Annotation Tools.

## **References**

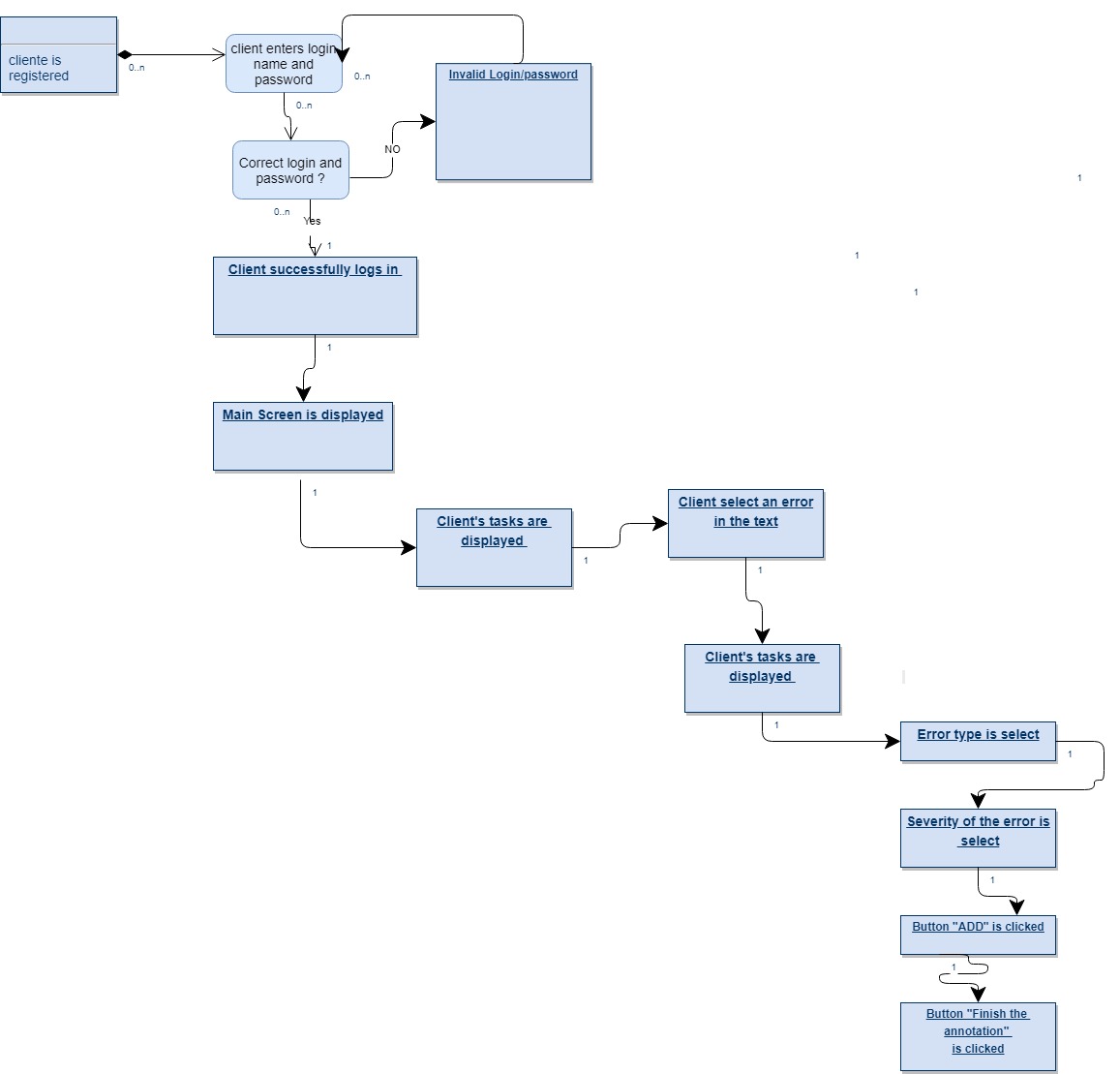
This Functional Analysis specification was created without external references.

**2 FUNCTIONAL REQUIREMENTS**

## 2.1 **Context**

This diagram represents the way that the user will use to login into Annotation tools. In this document is possible to identify possible problems and what the return that the system should give to the user.

Finish annotation **Context Diagram**



## 3. **User Requirements**

The main screen appears with at least one task to user perform the review.  
 User can view the name of the task.   
 User can view the date that the task was created.  
 User can view the progress bar of the task to know if the review is finished or not.

User can open the task to perform an annotation.

## 3.1 Identification of requirements

By convention, referrals are made by the name of the subsection where they are described, followed by the requirement identifier, according to the following specification:

[name of subsection. identifier of requirement]

Requirements must be identified with a unique identifier. The numbering begins with the identifier [FR001] for functional requirement or [NFR001] for non-functional requirement and continues to be incremented as new requirements arise.

## 4.0 Functional Requirements

4.1 Description

Allows registered users to access other system features that require authentication.

4.2 Actor

Pre-registered users in Annotation Tools.

4.3 Pre-condition

The user must have a registration in the system.

4.4 Post condition

The user will be logged into Annotation Tools and will have access to the other features.

## 5.0 Classification of Functional Requirements

## 5.1 Essential: Without this part the system does not go into operation. Essential requirements are essential requirements, which must be implemented at all times.

## 5.2 Important: Without this part the system goes into operation, but in an unsatisfactory way. Important requirements need to be implemented, but if they are not, the system can be deployed and used anyway.

## 5.3 Desirable: It’s the requirement that it does not compromise the basic functionalities of the system, that is, the system can function in a non-conforming way without it. Desirable requirements can be left for later versions of the system if there is not enough time to implement them in the version being specified.

## 

**6.0 Functional Requirements**

6.1 Perform Annotation

|  |
| --- |
| [FR001] Button “Finish annotation” |

Description of the use case:  
 After user has select an available task on main screen. The user will select a part of the text that appears with an error. The select text will appear in blue color. User will provide the error type in tab “Error Type”. Use will choose a severity. User click in button to finish the task.  
  
 Actors:   
 Unbabel’s Translators.   
 Priority:   
 Essential ( X ) Important () Desirable ()  
  
Preconditions:  
 Valid registered user logged in Annotation Tools  
 At least one available task.  
 An Error Type select

An available severity is chosen.  
  
 Post-condition:   
 ´The Annotation process is finished correctly.  
Data:  
 Finish of annotation

### 7.0 **Hardware Interfaces**

This application will in Browser Platform. It’s necessary to have updated Browser installed.

### 7.1 **Software Interfaces**

Google Chrome 65.0.3325.181 installed.

## 7.2 **Hardware/Software Requirements**

Windows Seven or above with Google Chrome installed

### 8.0 **Security and Privacy**

Annotation tools will security all theses possibly problems and privacy.

8.1

1. Loss or corruption of data
2. Disclosure of secrets or sensitive information
3. Disclosure of privileged/privacy information about individuals
4. Corruption of software or introduction of malware, such as viruses

8.2

1. Physical security.
2. Access by user role or types.

### 9.0 **Recoverability**

9.1

If the processing site (hardware, data, and onsite backup) is destroyed, how soon must the application be able to be restored?

The application must be restored as soon as possible using backups

### 10. **System Availability**

The application will be available twenty four hours a day, seven days a week and uninterrupted.

### 11. **Data Retention**

The data must be saved until the user deletes his own account from the Annotation Application