# Software Requirements Specification

**for**

# EasyAuthor

**Version 1.0 approved**

**Prepared by <author>**

**<organization>**

**<date created>**

# Table of Contents



# Revision History

| Name | Date | Reason For Changes | Version |
|---|---|---|---|
| | | | |
| | | | |

# 1. Introduction

## 1.1 Purpose

To generate different easy-author applications suited for different people who have different teaching methodologies. Previously, there used to be a single applications which is for a specific teaching style. This project aims in creating a framework which will enable generating the easy-author application for a specific utility sought by the instructor

## 1.2 Product Scope

This Product will develop and deliver a new web application. In this web application, the teachers will create and manage educational content for adult illiterates for different types of methodologies.

# 2. Overall Description

## 2.1 Product Perspective

The easy author system is a self-contained system that can manage all the teachers info. Various Stakeholders are involved in this easy author tool.

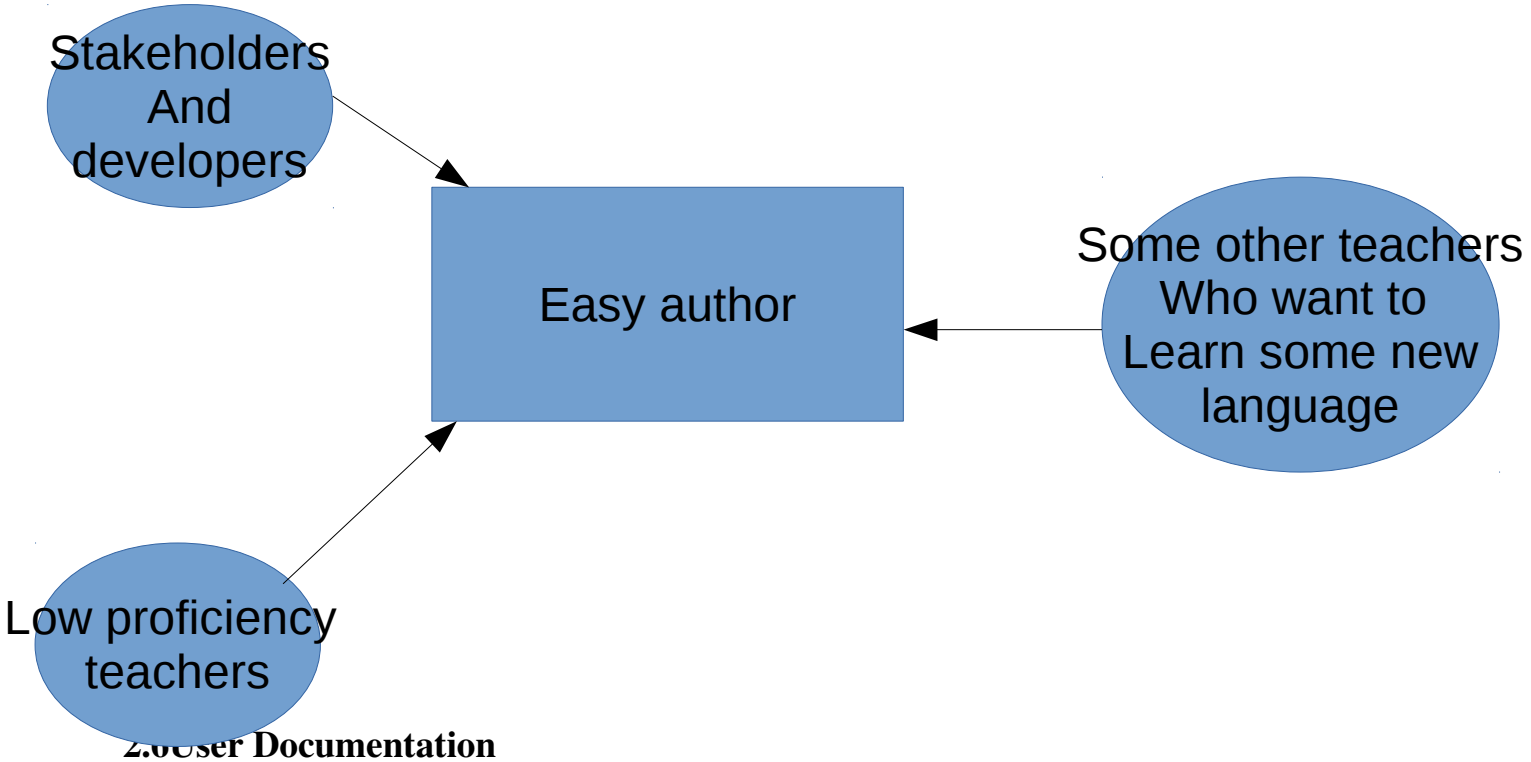
## 2.2 Product Functions

Register:When the user want to login into this easy-author tool they have to register first.
Login:Then the user can login after registration.

## 2.3 Operating Environment

In this project we use HTML,JAVA,CSS,C and XML files also required for this easy author.

### 2.4 User Classes




### 2.6User Documentation

The documentation for a product or service provided to the end users. The user documentation is designed to assist end users to use the product or service. This is often referred to as user assistance

### 2.7Assumptions and Dependencies.

*Assumption is we would be able to generate easy author tool for teachers*

*If we are not able to generate an easy author tool then we have to choose other option .*

## 3. External Interface Requirements

### 3.1 User Interfaces

*User Interface will be Login Page*

*Input:Take the input from the user to select which methodologies they want.*
*Easy author tool:They show the different types of methodologies for the user to use this app.*

## 3.2 Hardware Interfaces

*This Application will be a web application by which people can access the app without any restriction of the platform. The user just has to register for the application after which he can log-in from the application. The user can also register via different social network platforms using APIs.*

## 3.3 Software Interfaces

*This application will be made in ruby on rails which is a ruby-framework. Different gems in the framework will be used as helpers for app will be installed in the framework.*

## 3.4 Communications Interfaces

*This web-application will be using https server which will be hosting the application. Gems which will be used for file transfer will also be installed within the framework.*

# 4. System Features

## 4.1 System Feature 1

### 4.1.1 Description and Priority

*Taking the input from the user for the type of methodologies which they wants to learn.*

*And the priority is too high .*

### 4.1.2 Stimulus/Response Sequences

*After Logging into the game, the user will first select the type of methodologies they want to learn. For selecting the methodologies type, there will be different types options and the user just needs to click the required language.*

### 4.1.3 Functional Requirements

Methodology:

There will be a different methodologies for different type of teachers they can select one from among them.

There is an XML files also.

Content:

There will be a some content they can select in that options like animations etc...in which users can understand.

### 4.2 System Feature 2 (and so on)

# 5. Other Nonfunctional Requirements

## 5.1 Performance Requirements

*The user interface has to be easy and understandable. The easy author web-application may be weighted so that it does require more space and more system resources while executing that game.*

## 5.2 Safety Requirements

*User password should be very confidential.*
*They can change the password frequently.*

## 5.3 Security Requirements

The server on which the On line will have its own security to prevent unauthorized *write/delete* access. There is no restriction on *read* access. The use of email by an Author or Reviewer is on the client systems and thus is external to the system. The PC on which the Designer resides will have its own security. There is no special protection built into this system other than to provide the On line Journal to publish an article.

## 5.4 Software Quality Attributes

## 5.5 Business Rules