Document Engineering Custom EBooks

Software Requirement Specification

Team Members

Eashwar V	MT2011041
Nishanth TS	MT2011092
V Anil Kumar Gontla	MT2011165
Vyshak GV	MT2011174

Under the guidance of,

Prof. R Chandrashekar

International Institute of Information Technology,

Bangalore

Table of Contents

1. Introduction

Document engineering refers to the design of a document to meet very specific requirements for Clarity. By clarity we mean the documents make complete sense to the users and the applications that need them.

1.1 Purpose

Document Engineering custom EBooks project aims in creating an authoring tool and a reader tool which provides user to create a document based on their required features and customize it, which can be read in the reader tool.

1.2 Scope

This project is aimed to help masses in technical field so that it can provide variety of features to users in a single document like text, Rich text, Images, Videos, Flash applications. The software is developed in C# language.

There a two main users.

a. <u>Document Creator</u>: Creates a document using the authoring tool and has privileges to select the template in which he wants to create the document.

b. <u>Document Reader</u>: Reads the document created by the Document creator using the reader tool.

2. Overall description

2.1 Product Perspective

This product is been created for the user to customize their document based on the features they require. i.e., unlike traditional documents which support only text and images, it'll support many other features.

2.2 Product Features

The major features of the product are:

- Selection of the desired template in the authoring tool
- Based on the template user can select the inputs to the document.
- The features provided are Text, Rich Text, Images, Flash files, Videos, Audio, Q&A.
- The document created by the authoring tool can only be read in the reader tool.
- The reader tool even provides the zooming option to enlarge the document.
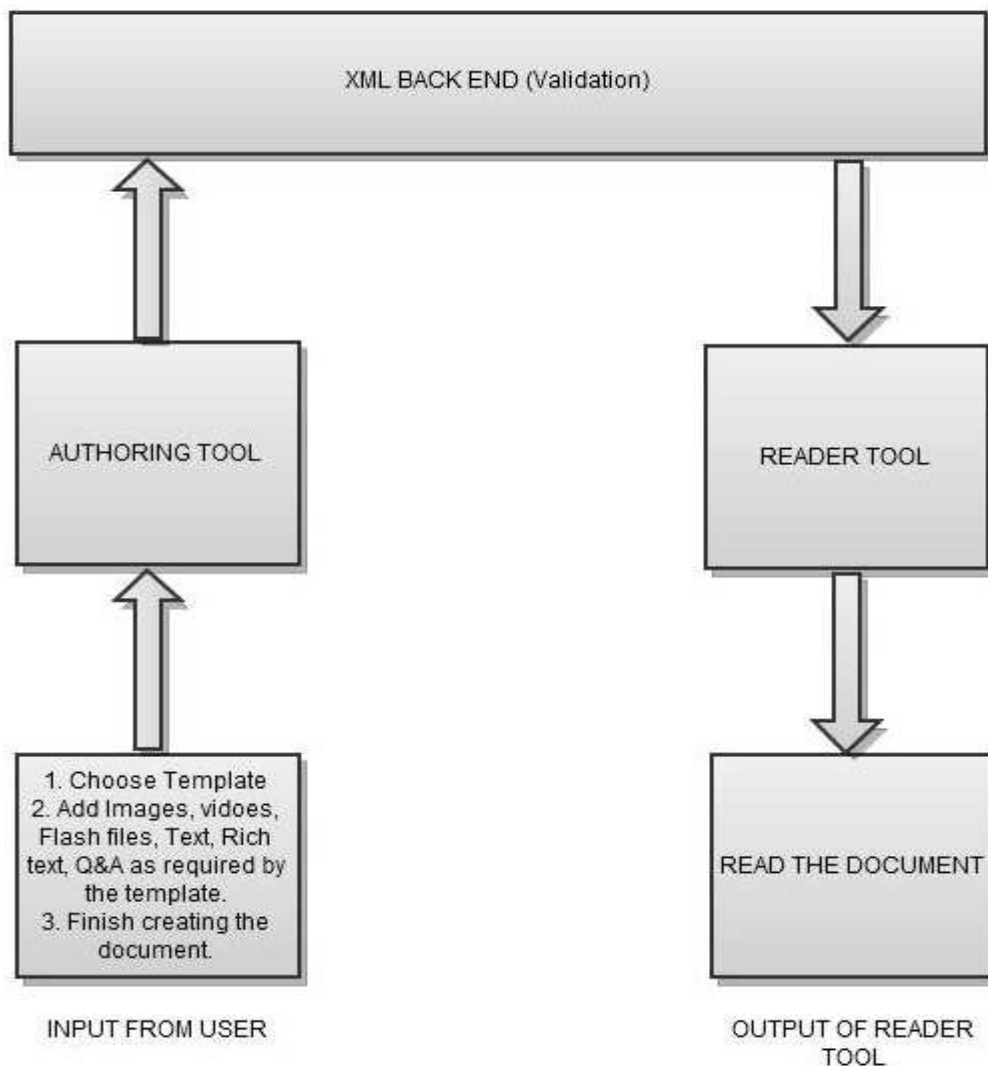- Backend XML validation will be done in the reader tool

Figure 1: Flow diagram of the project

2.3 Operating Environment

The software is created to work on Windows platform which can be installed by installing the executable of authoring and reader tool.

2.4 Design and Implementation Constraints

The constraint of our project is the user should to select the required template and choose the features that are present in that particular template. And while adding the question and answer as input they are restricted to enter only four options

2.5 Assumptions and Dependencies

In our case we assume that the user is running our application on windows operating system, and the user will provide the files which are compatible to our application. And another major aspect is the user should give only the project created by our authoring tool as input to the reader tool because at the backend XML validation takes place and password of the zipped file will be checked.

3 System features

3.1 Creation of Document using Authoring tool

The User will specify the project name and choose the required template. Then add the features present in the template, consequently a XML file will be generated in the backend which will act as the metadata of the features. When user is finished with the creation of document a package will be created, which should be given as input to the reader tool.

3.2 XML validation

Whenever a user creates a document a XML file will be generated, which acts as a metadata of all the features of the document and will be stored in the package that is created by the authoring tool. When this package is given as input to the reader tool it'll validate the XML instance file with the Schema present in it. If the file validates the reader displays the document.

3.3 Reading Document in Reader Tool

When a document is given as input to the reader tool, two level security checks will be done; first, while creating a package the authoring tool will encrypt the package with a password which will be decrypted by the reader tool. Second, to display the document the XML validation should be done at the reader tool, once validation is done the reader tool will figure out the template and display the document accordingly.

4 External Software Requirements

4.1 Hardware Interfaces

There are no specific hardware interfaces required by our project.

4.2 Software Interfaces

We need Microsoft Visual studio 2010 to build our project which supports C#.net platform.

Windows operating System to install the application

Active X control software to support audio and video files.

5 Other Non-Functional requirements

5.1 software quality attributes

Portability: We can use this software only on Windows operating system

Availability: The software can be made available to every common user who wants to create and access the document.

Testability: The software can be tested for the performance using the random inputs for different templates.

5.2 Safety Requirements

There are no safety requirements for our project.

5.3 Security Requirements

The security Requirement is that the package created by the authoring tool should be given as input to our reader tool.