**Software Design Description**

**Aspose-Online Document Processing Service**

**Prepared By:**

**Adeel Umar Usmani**

**JUL 26th, 2016**

Contents

[1. Scope 3](#_Toc457382064)

[2.Definitions 3](#_Toc457382065)

[3. Document Purpose: 3](#_Toc457382066)

[4. Design Description Information 3](#_Toc457382067)

[4.1 Introduction 3](#_Toc457382068)

[Scope 3](#_Toc457382069)

[4.2. Architectural Representation 3](#_Toc457382070)

[4.3. Architectural Goals and Constraints 4](#_Toc457382071)

[Technical Platform 4](#_Toc457382072)

[Technologies 4](#_Toc457382073)

[4.4. Use-Case View 4](#_Toc457382074)

[4.5. Logical View 4](#_Toc457382075)

[Overview 4](#_Toc457382076)

[4.6. Deployment View 5](#_Toc457382077)

[5. Design description organization 5](#_Toc457382078)

[5.1 Introduction 5](#_Toc457382079)

[5.2 Design views 6](#_Toc457382080)

[5.2.1 Decomposition description 6](#_Toc457382081)

[5.2.2 Flowchart Description 7](#_Toc457382082)

# 1. Scope

* Document will define the high level design and technology decisions of the Aspose-Online Document Processing System (ODPS).
* Document defines and describes the use of each service feature, the architectural constraints of the system, the functional requirements with a significant impact on the architecture.

# 2.Definitions

ODPS is basically using Aspose-Java Components to facilitate its users in different ways:

* Converting a document to variety of formats like PDF, HTML, PNG, JPEG etc.
* Password based document protection using different Protect options like Allow Comments Only, Allow Only Revisions, Allow Only Form Fields , Read Only & No Protection.
* Retrieving protection info on a document

ODPS is meant for relieving the user from installation of Aspose-Java Component on local machine. User only needs to have any modern day browser to use the mentioned features.

# 3. Document Purpose:

The purpose of this document is to define and describe the use of each feature, the architectural constraints of the system and the functional requirements with a significant impact on the architecture.

# 4. Design Description Information

# Introduction

The Software Design Descriptions (SDD) provides an architectural overview of ODPS. It aims to provide the stakeholders a clear understanding of the system.

## Scope

The scope of this document is to define high level design and technology decisions of the ODPS.

# Architectural Representation

**Deployment view**

**Audience**: Deployment managers.

**Use Case view**

**Audience**: all the stakeholders of the system, including the end-users.

# Architectural Goals and Constraints

This section describes the software requirements and objectives that have some significant impact on the architecture.

## Technical Platform

ODPS application will be deployed onto a J2EE application/Web server.

## Technologies

Spring MVC framework has been used to implement the software.

# Use-Case View

This section provides a functional overview of the system by a use-case diagram.

## 

# Logical View

## Overview

ODPS application is divided into layers based on the N-tier architecture

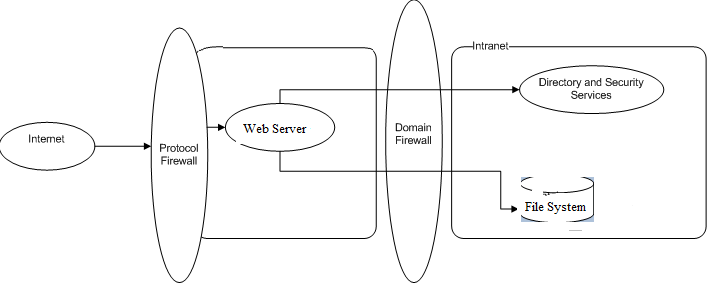


The layering approach is the mostly accepted solution for enterprise applications, which require scalability, modularity and easy maintenance.

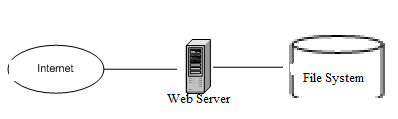
* The **web Clients Interaction Services** deals with the presentation logic, request dispatching and response rendering
* The **Application Logic** deals with the core functionalities of the system. (Document Conversion, Protection)
* The **File System** is responsible for storing & streaming files.

# Deployment View

**Logical Structure**

****

**Physical structure**

****

**Details**

* Spring MVC Rest features have been used in order to develop rest API.
* Eclipse Mars 4.5.2 & jdk1.7 has been used for development on Win7 machine.
* Apache Maven 3.3.9 has been used to build the project
* Apache TomCat 8.0.33 has been used as Web Server
* Application servers are IBM servers running Jboss Application Server
* Database server is IBM server running MySQL server.

# 5. Design description organization

# 5.1 Introduction

The Software Design Description (SDD) details the chosen software architecture and the justification for selecting that architecture. In this project the team was tasked with architecting and implementing an Online Document Processing Service(ODPS) which uses different features of Aspose Java Component. Spring MVC framework has been used to implement the software. Reason of preferring Spring MVC is that it not only provides a sophisticated template to implement Rest API but also in future we can incorporate different Spring features in our application.

# 5.2 Design views

Entity attribute information can be organized in several ways to reveal all of the essential aspects of a design. In so doing, the user is able to focus on design details from a different perspective or viewpoint. A *design view* is a subset of design entity attribute information that is specifically suited to the needs of a software project activity. Each design view represents a separate concern about a software system. Together, these views provide a comprehensive description of the design in a concise and usable form that simplifies information access and assimilation.

# 5.2.1 Decomposition description

**5.2.1.1 Scope**

We first must understand our organization's business functions before beginning developing information systems. The decomposition descriptions are done in order to plan business functions, processes, and sub processes within ODPS Project.

**5.2.1.2 Use**

Our Project will perform a variety of different functions. Before we plan what systems to build for the organization, it is helpful to first understand the functions of ODPS Projects needs to perform. Then it is much easier to identify processes that occur within the business functions, and ultimately the systems that will support those processes.

The process of starting at a high level and moving into smaller and smaller subsystems is called decomposition. The functional decomposition diagram (FDD) is a planning tool for identifying business functions and the processes that comprise them.

**5.2.1.3 Representation**

Gets Word Document Name to determine Protection type

Get Protection Type

Gets Word Document, Password, Protection Type, & Protected Document Name

Protect Document

Document Processing System

Gets Word Document & Destination Document Format

Convert Document

# 5.2.2 Flowchart Description

**5.2.2.1 Convert Document:**

Start

.DOC /.DOCx Type

User Uploads Word Document & Specifies target format

System prompts error of invalid format

No

System converts the document in required format

End

Yes

**5.2.2.2 Protect Document:**

Start

Start

User Uploads Word Document & Specifies Protection type, password & protected document name

System prompts error

Yes

Document already exists on server

No

System applies protection & stores document on server

End

**5.2.2.3 Get Protection Type:**

Start

Start

User inputs document name

System prompts error

No

Document exists on server

Yes

System sends back protection information

End