DocKloud

Software Requirements Specification (SRS)

Revision History

Date	Revision	Description	Author
09/25/2012	1.1	Initial Version	Amit Nain Bharat Gowda Kunal Chilka Kushal Bandi Prasad Korhale
09/28/2012	1.2	Final Version	Amit Nain Bharat Gowda Kunal Chilka Kushal Bandi Prasad Korhale

Table of Contents

1. Purpose

The purpose of this document is to exhibit complete description of web based document management system. It captures the features and functionalities of the system, detailed design of the system, requirements of the system and constraints of the system. This document is intended for developers, testers and customers.

1.1. Scope

This document will cover the expectations of the customer, project design, functional and non-functional requirements, security features and traceability matrix for validation.

1.2. Definitions, Acronyms, Abbreviations

1.2.1. Definitions

- Authentication: Process by which you verify that someone is who they claim they are.
- Authorization: Function of specifying access rights to resources, which is related to information security and computer security in general and to access control in particular
- Bandwidth: Rate of data transfer, bit rate or throughput
- Database: Organized collection of data
- Decryption: Activity of making clear or converting from code into plain text
- Encryption: Activity of converting data or information into code
- Digital Certificates: A method of providing other systems (or users) a level of trust that the public key claimed to belong to a user (or organization) does indeed belong to that user
- Metadata: A set of data that describes and gives information about documents/files.

1.2.2. Acronyms

- GUI: Graphical User Interface
- HTML: Hyper Text Markup Language
- HTTP: Hyper Text Transfer Protocol
- HTTPS: Hyper Text Transfer Protocol Secure
- OS: Operating System
- RAM: Random Access Memory
- SRS: Software Requirements Specification
- SQL: structured query language

1.3. References

Following are the list of documents referred

- Project description document
- User guide
- Test plane document

1.4. Overview

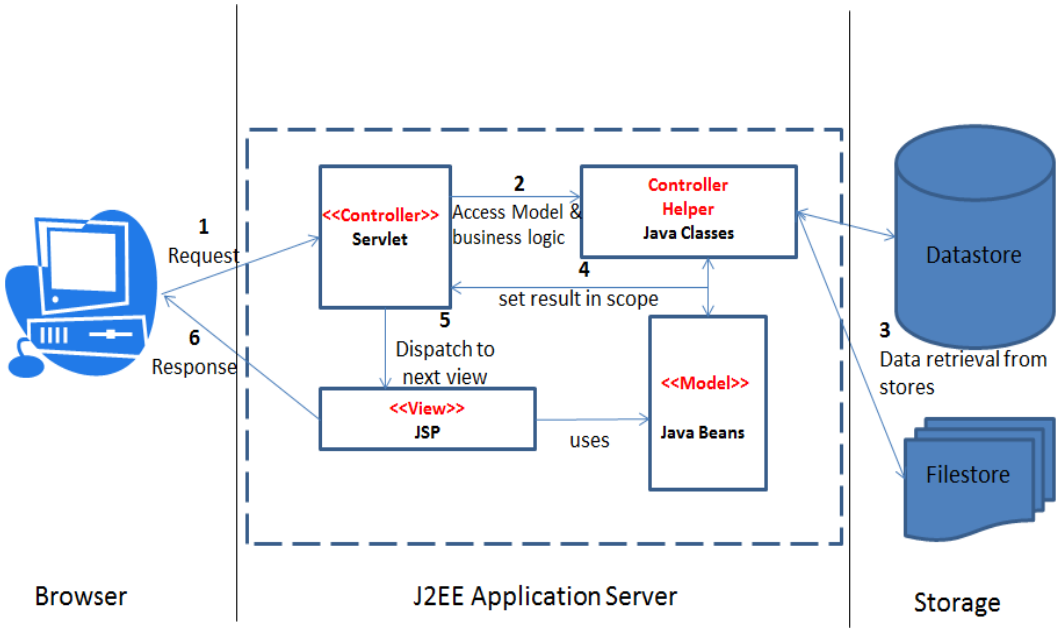
This software system is designed for web based document management system for employees of the company. This system facilitates user to download, read, uploads, modify, check out and check in the document online using graphic user interface. It allows user to share document along them and access to the document from any place. The system is based on relational database of employees and documents. The database will be supporting employee's privileges for documents based on its department and role. This system is designed to handle the various security attacks as well, as it allows user to share/upload the confidential documents.

2. Overall Description

2.1. Product Perspective

An organization is a collection of departments structured in hierarchy. Employees of each department communicate and share resource with each other in same department or inter-department through documents or files. In large organizations it becomes a need of quick sharing via electronic mode. These documents are categorized in to different level based on the confidentiality which needs to be controlled depending upon role and level of the employee. "DocKloud" is a web based document management system which helps employees to share, read, update and upload in a collaborative way. Being web based DocKloud provides employee an easy to access the system from any location at any time. This system will enable automation of document management features like centralized storage, limited access and change tracking replacing legacy system.

2.2. Product Architecture



The product is a MVC (Model View Controller) architecture based product.

- **Model**: Are data models which maps real word entity and relational schema. They are inherently called as "beans" consisting of all attributes as that in relational data model.
- **View**: Controller selects or dispatch to the appropriate view for the user response. The models are manipulated in business logic and view uses them to render the user content.

- **Controller**: Triggers action which manipulates the models. They control the flow of the application between view and models. Business logic is implemented in the controller

Advantage of MVC architecture

- **Modularity**: Each component is loosely coupled which allows component to be replaced with other component performing same functionality. Development can be done in parallel by each owner of the component and changes in one component wouldn't affect the other.
- **Scalable**: MVC architecture is scalable allowing new function to be implemented at any point of time.
- **Multi views**: Multiple views can be used at for the same data as view is loosely coupled with other components.
- Multiple external interfaces can be plugged into the application with controller enacting as the bridging component.
- Multiple MVC model of the same application can be replicated in order to scale across larger geographic area creating inter-MVC communication.

2.3. Product Functionality/Features

- User login
    - The product allow user to login to system with which he or she can manage its own document workspace
    - Product allows new user to register to the system
    - User can retrieve password if lost or forgot
- Document management
  Base on the privileges given assigned to user operation can be performed on documents:
    - User can view the list of document which he/she created
    - View the list of documents which has been shared
    - User can view/read the document
    - User is given option to encrypt the document in order to keep the confidentiality
    - User can upload/update the documents
    - User can delete a document
    - User is given option to create folder in order to maintain systematic workspace
    - User can check in/out a document in order to get a lock
    - Share a document by assigning necessary access rights
    - User can get information about the document metadata as well as can edit them.
    - Overwrite protection: As per definition uploading a file is first time creation of the file to server and updating of file is modifying the uploaded file and updating the server.
        - System maintains metadata file for each document.

- This file will have the information of the last update and all the user's entry who clicked the file after this update.
- This way we can make sure when the user tries to update a file we can check if this user had clicked the file after the previous update
- If so, that means he has written the valid file and trying to update it.
- If he is not in the user list that means he is trying to overwrite file illegally. By this way we do overwrite protection.

- System security
  - All the communication between user and the system will be carried out on a secure channel i.e. HTTPS
  - Session key will encrypted with standard cryptographic protocol.
  - Document will be stored encrypted at the back-end
  - Cross site scripting and SQL injection will be taken care.
  - User login will be based on 1 factor authentication (1 FA)
  - Security against session hijacking will be provided.
- Sharing feature
  - User who owns the document can share among other user and can give necessary access rights.
  - Only one level of sharing is permitted i.e. sharing cannot be forwarded to third user
  - Sharing feature is made sure that it is synchronize
- System
  - System is tuned to handle heavy traffic and concurrent access.
  - System is portable i.e. it can be accessed from other location as well
  - It is designed to store files of different format

2.4. User Characteristics

Employee in an organization has multiple roles. Each role has different access privileges associated with it.

Following are different categories based on which users are classified

1. Corporate-level management officials:
   a. This includes the CEO, president, vice presidents, responsible for the operations of multiple departments.
   b. They can access one or more department document and hence the product considers them as root user.
   c. They can upload, read, delete, update or check in/out all the documents in the departments he/she is responsible for

2. Department manager:
    a. A department manager have entire control of one department to which he is assigned
    b. He has full permission to access (upload, read, delete, update or check in/out) all documents for that department except corporate-level document unless shared by the corporate-level officials
    c. He can access other department's document if shared by users of other department
3. Regular employee:
    a. These set of users can upload, read, delete, update or check in/out the documents he/she created in his/her department
    b. They cannot access documents owned by department manager or corporate-level manager unless a department manager specifies to share the document with him/her
4. System administrator:
    a. Can add, delete, modify user's account
    b. Can assign roles to a user
    c. Administrator verifies new user and approves the request
    d. System logs are generated by admin user
    e. Cannot access the documents uploaded by users.
5. Guest user:
    a. These are non-functional user of the organization
    b. They cannot upload or delete any document
    c. They can read, update or check in/out the documents that have been shared with him/her
6. Temporary user:
    a. These set of users are first time user who have registered themselves through registration page
    b. They can only send request to system administrator in order to get access to the system
    c. Until approved by admin he/she will only be able to see request approval pending page

2.5. Constraints

1. Product does not provide an editor to edit a file online. This means in order to edit a file user needs to download a file and perform edit operation
2. The file formats supported by the system are .doc, .docx, .ppt, .pptx, .pdf, .jpeg, .jpg .png, .bmp, .xls

3. File size of maximum 5MB is permitted to upload
4. No new role or department can be added by the system online
5. User will be able to see only the latest version of the file. Earlier versions of file still exist at back end for archival.

2.6. Assumptions and Dependencies

1. No nested sharing of document is allowed as it may violates sharing policy
2. A default administrator will be created at the time of deployment of the product
3. When a user locks a folder it will lock all the files in that folder. Folder lock does not imply locking of the folder only. This will give user to unlock selective file.
4. This product is not designed to work in offline mode as it will increase security risk.
5. Users rely on network connection in order to access the system.

3. Specific Requirements

3.1. Functional Requirements

The functional requirements describe operations that will help application developers for planning, design and developing the application. It will also help individuals who are operating the application to understand what operations users can perform on the website depending on the permissions which they have. Purpose of this functional requirement is to provide a detailed procedure to operate the Web-based document management system.

3.1.1. Authentication

System shall provide a login page for users. It should prompt user to enter username and password.

a. If user enters a valid username and matching password, system should redirect to home page.
b. If username do not exist or password was entered wrong, system shall prompt to click on forgot password link or register new user link.

3.1.2. User Registration

System shall provide new user registration page. It should prompt user to enter name, username, password, re-type password, email id, role required, one or more department to which access is required.

a. If a user name is already allotted system shall prompt to provide a different user name
b. If password's entered do not match system shall provide error message
c. Email id is verified if it is a valid email id or not

3.1.3. Website Administration

System shall provide admin page for administrator of the website, which shall contain functionality to verify new user's registration requests, manage existing users, archive data and should have access to application system log.

a. If the user request is valid, that is user's requested role and departments are legitimate administrator shall approve the request
b. If the request is not valid, if there is a conflict in either the role or department requested administrator shall either decline request or change the department or role which was requested and approve the request.
c. System shall allow administrator to delete the existing users
d. System shall allow administrator to access and download the system log
e. System shall provide option for administrator to create a backup of the data

3.1.4. Document Management Operations

After a user logins to the system, user shall navigate to the document management page. Based on the users role and department's to which he/she has the permission (next section describes the permission user has based on the role) user shall be able to do different operations. Below are the operations on the document the system shall provide

a. System shall list the departments and user can navigate to a specific department. After selecting a department user can see a list of all the documents for which he/she has access and also access the met-data of the document
b. System shall provide an upload option to add a new document to a particular department.
c. System shall allow users to read a document by downloading it to the local system
d. System shall allow user to edit the document and then update it to the server provide the document is checked out by the user.
e. While uploading or updating a file system shall
f. Every document or the entire folder can be checked/out which means it is now locked to the user and no other user can update this file until lock is released
g. Similarly once after checking out user shall be provided with the option of check-in this will release the lock on the checked out file.
h. System shall provide option to share a document with other users by providing the user id with whom the document is to be shared
i. System shall allow user to specify if encryption needs to be done on a particular file. If specified so, the file should be encrypted while storing in the file system and decrypt the file while retrieving it.

3.1.5. User Roles and Permissions

Various roles shall be predefined in the system and each user can be assigned to one of the roles. Permissions for using the system is based on the roles. System shall allow user to register to one of the below roles.

a. Corporate-level management official: User with this role while registering can request permissions for one or more department. And this user shall be allowed to perform all the operations discussed in the previous section on the documents of departments to which he/she has access.
b. Department manager: User with this role shall be allowed to perform all the operations on the documents of department to which he/she has access. But cannot access the documents uploaded by corporate management official unless the document is shared to him/her
c. Regular employee: User with this role shall be allowed to perform all operations on the documents uploaded by him. But cannot access the documents uploaded by department manager and corporate management officials unless the document is shared to him/her

d. Guest user: User with this role shall not be able to upload or delete any document. But he can do other operations on the documents shared to him
e. Temporary user: A new user after he registers and until administrator approves his request is considered as temporary user. System shall not allow temporary user to perform any document management operation.

3.2. External Interface Requirements

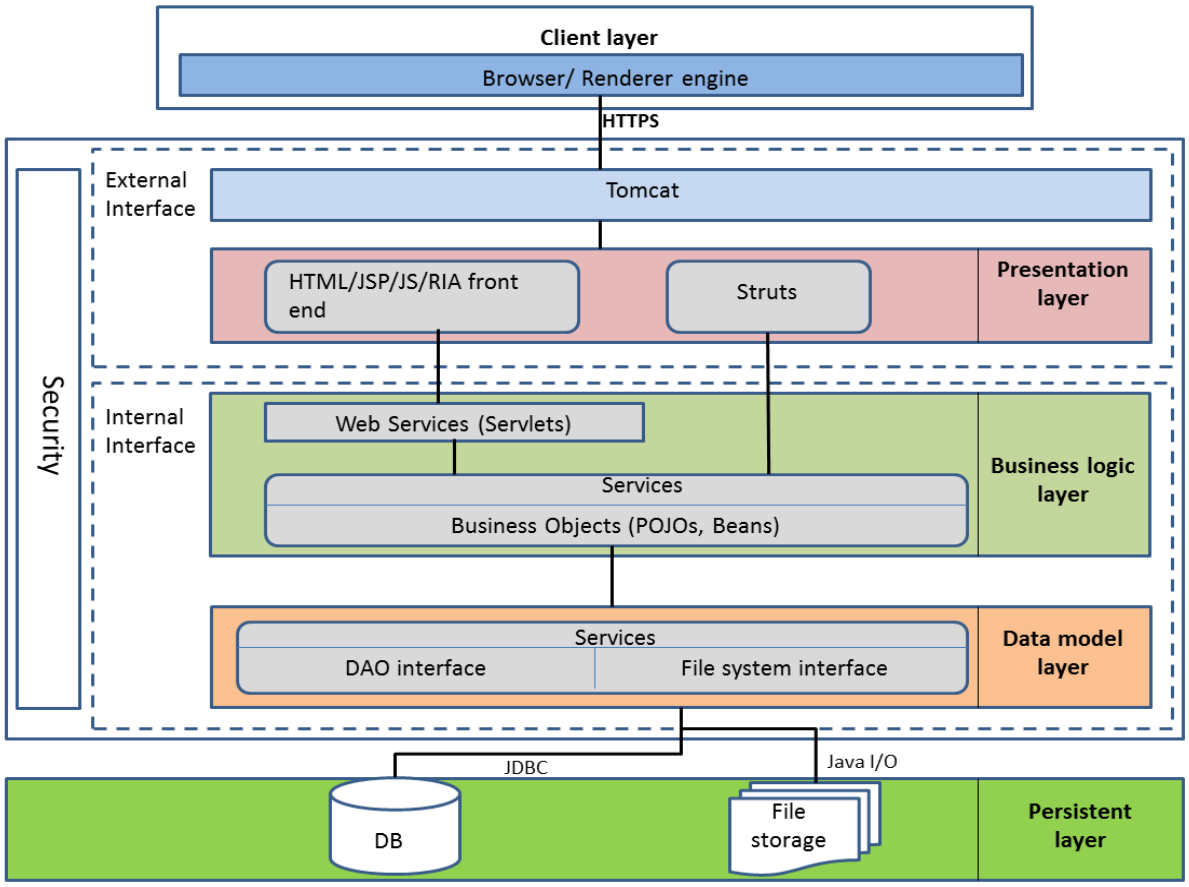
3.2.1. User Interface:

a. System shall provide login page for user to sign in to the system
b. System shall provide new user register page
c. System shall provide document management operations page through which user can access the documents to which he has access and perform various operations like upload, read, update, delete, check-in and check-out based on user permissions

3.2.2. Admin Interface

a. System shall provide admin page through which system administrator can approve new user requests
b. Also admin options to download system log and create back up

3.3. Internal Interface Requirements



3.4. Internal Data Requirements

As depicted in the above figure from the previous section the data flow to and from data base or file system happens through data base interface and file system interface layers. By doing system shall do data abstraction at different layers.

3.5. Design and Implementation Constraints

a. All the user data, document relative path and document status shall be stored in relational database; hence data base design shall adhere to relational data base.
b. Browsers on the client machine should work with HTML standards.

3.6. Other Requirements

Source code for the application shall be maintained in Git hub version control system.

4. Non-Functional Requirements Prasad

4.1. Security and Privacy Requirements

4.1.1. Authentication

First and foremost, authentication should be needed to log into the document management system. Access to any page in the website should need a valid session.

4.1.2. Incorrect Login Attempts

There should be a limit for successive incorrect login attempts as this is a possible attack to break the authentication. After three times authentication fails, system shall show a CAPTCHA challenge. The system should proceed only after the user enters the proper characters in the CAPTCHA.

4.1.3. Secure Communication Channel

The communication channel between the client and server should be secure. This can be accomplished by using HTTPS (HTTP + SSL/TLS) protocol for communication. This ensures an encrypted channel of communication and will avoid any eavesdropping and man in middle attacks.

4.1.4. Role based authentication and authorization

Privilege level should be defined according to the roles of users. Elevation of privileges should be avoided and mechanism should be provided to detect any such behavior. Unneeded privilege should not be given to any user.

4.1.5. Session Time-outs

If a particular session remains inactive for a predefined duration, it should be timed out. This will avoid exploits when a user forgets to log-out from his session on a machine and an attacker gets hold of his account.

4.1.6. Defense against DOS attacks

System should have a defense mechanism against Denial of Service attacks. Whenever a particular client tries to access a resource for more than a predefined threshold, block further requests from that client for a particular time. Black list and White list can be maintained either in web server or application server.

4.1.7. Avoid SQL injection

System shall avoid SQL injection attacks by executing SQL queries using prepared SQL statements. User inputs can be validated before querying the database.

4.1.8. XSS scripting attacks

Cross side scripting attacks should be avoided by input validation and tagging the input with HTML tags.

4.1.9. Document Encryption

User should have a functionality to generate passcode/password for encrypting their documents on request. These documents shall be stored in an encrypted format on the server to avoid any leakage of information in case the data-store is exposed. This also serves as a defense against a session hijack.

4.1.10. Hashed Passwords

The passwords should be stored in encrypted format. This solves two security problems. First, even if database falls into wrong hands, the authentication cannot be broken. Second, the trust on the system administrator does not have to worry about.

4.2. Other Requirements

1. Multiple users should be able to log in to the system at same time.
2. System should maintain log files.

4.3. Environmental Requirements

As this is a web application, it should run as far as the client machine is connected to the internet. Client machine needs the Browser environment with JavaScript enabled for doing various validations.

4.4. Computer Resource Requirements

4.4.1. Computer Hardware Requirements

Hardware requirements on the client side are as follows:

1. RAM: Greater than 512MB
2. Processor: At least 1GHZ
3. Free space: 200MB
4. Bandwidth: 512KBPS

Hardware requirements on the server side are as follows:

1. RAM: Greater than 2GB
2. Processor: At least 2GHZ
3. Free space: Minimum 50GB
4. Bandwidth: 2MBPS

4.4.2. Computer Software Requirements

Software Requirements for the client side are as follows:

1. OS: Windows/Mac/Linux

2. Browser: Mozilla/IE 8 and above/Chrome
3. Applications to open Word, Excel, PDF,etc.

Software Requirements for the server side are as follows:

1. OS: Ubuntu
2. Database: MySQL
3. Web Server: Tomcat 6.0
4. J2EE framework

4.4.3. Computer Communication Requirements

1. Digital certificates are required to establish a secure HTTPS connection during session establishment and all the operations when a user is logged in.
2. A minimum bandwidth of 512KBPS is needed between client and server to ensure smooth functioning.

4.5. Software Quality Factors

The software would be considered of high quality if it has all of the following attributes.

1. System should be scalable and serve high traffic without any hit on the performance.
2. System should be compliant to all the requirements specified in the requirement document and elaborated in the functional/non-functional requirements.
3. All the security aspects should be taken care of so as to avoid any theft of confidential user data.
4. System should be maintainable and design should be fully documented comprehensively.
5. In case of any errors system fails gracefully and notifies the user with a comprehensive message to alert him.
6. The system should provide a user friendly interface.

4.6. Packaging Requirements

The system should be easy to migrate between servers. Hence, it should not have any dependency of libraries on the development machine. System shall be packaged with software and documents like user guide to build a deployable WAR file.

4.7. Precedence and Criticality of Requirements

Precedence number starts from 1. Requirement with precedence 1 is to be developed first. Criticality weightage has been defined as follows,

1 - Trivial

5 – Very Critical

The requirements can be segregated as follows:

Requirement	Precedence	Criticality
Authentication and Authorization based on privileges	1	5
Admin Functionalities	1	4
Document Management Requirements	2	4
Security Features	3	4
Software Requirements	1	3
Hardware Requirements	2	3
Software Quality Factors	3	2
Environment Requirements	1	2

5. Qualification Provisions

5.1. Qualification Methods:

Qualification Method	Description
Analysis	A qualification method that is carried out by visual examination, physical manipulation, or measurement to verify that the requirements have been satisfied.
Inspection	The visual examination of software item code, documentation, etc
Test	A qualification method that is carried out by operation of the item/component/I/F (or some part of the computer S/W configuration item, etc.) and that relies on the collection and subsequent examination of data
Demonstration	A qualification method that is carried out by operation of the item/component/I/F (or some part of the computer S/W configuration item, etc.), and that relies on observable functional operation not requiring the use of elaborate instrumentation or special test equipment
Special	Any special qualification methods for the software item, such as special tools, techniques, procedures, facilities, and acceptance limits

5.2. Qualification Matrix:

Requirement No.	Requirement	Verification Method	Comments
3.1.1	System shall provide a login page for users. It should prompt user to enter username and password	I	Check that the webpage gives option for login.
3.1.2	System shall provide new user registration page.	D	Go to Login page and click for sign up and verify that registration page contains all the options.

3.1.3	System shall provide admin page for administrator of the website	D	Open the website, login as an admin and observe that administrator page opens up.
3.1.4	User shall navigate to the document management page. Based on the privilege of user shall be able to do different operations such as upload, download, modify etc.	T	Login as a user and perform operations such as download, upload and modify document. Share the document with other users.
3.1.5	Various roles shall be predefined in the system and each user can be assigned to one of the roles. Permissions for using the system are based on the roles.	D	Login as different users based on the roles. Observe that permissions for the documents are based on the predefined roles of the users.
3.2.1	User Interface	I	Open the website and inspect different navigation possibilities for user to perform different operations.
3.2.2	Admin Interface	I	Open webpage and login as admin. Inspect that admin have all the options at the page, have access to all the pages to perform its operations.
3.3.	Internal Interface Requirements	A	Analyze the design of the software system to confirm that internal interface requirements are met.

3.4	Internal Data requirements	A	By code analysis, verify that data flow to or from database using different layers and interfaces is as per requirements.
3.5	Browsers on client machine should work with HTML standards	I	Observe code of webpage.
3.6	Source code shall be maintained on the Github	I	Observe the source code directory and verify that github is used to maintain the code.

6. Requirements Traceability

Sr. No.	Requirement Reference	Requirement Description	Test Case Reference
1	3.1.1	System shall provide a login page for users. It should prompt user to enter username and password	1.1
2	3.1.2	System shall provide new user registration page.	2.1
3	3.1.3	System shall provide admin page for administrator of the website	5.10
4	3.1.4	User shall navigate to the document management page. Based on the privilege of user shall be able to do different operations such as upload, download, modify etc.	5.8
6	4.1.1	Authentication should be needed to log into the document management system.	1.2
7	4.1.2	There should be a limit for successive incorrect login attempts.	1.8
8	4.1.3	The communication channel between the client and server should be secure.	6.1
9	4.1.4	Privilege level should be defined according to the roles of users.	5.4
10	4.1.5	If a particular session remains inactive for a predefined duration, it should be timed out.	4.15
11	4.1.6	System should have a defense mechanism against Denial of Service attacks.	6.2
12	4.1.7	System shall prevent SQL injection.	6.3
13	4.1.8	Cross side scripting attacks should be prevented.	6.4

14	4.1.9	User should have a functionality to generate password for encrypting their documents on request	4.17
15	4.1.10	The passwords should be stored in encrypted format	6.6