|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Description | Updated by |
| 23 August 2015 | 1.0 | Created the document | Collen |
| 23 August 2015 | 1.0 | Functional requirements | Collen |
| 23 August 2015 | 1.0 | Nonfunctional requirements | Collen |
| 23 August 2015 | 1.0 | Architecture | Collen |
| 23 August 2015 | 1.0 | Domain model | Collen |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**E-Cheat Sheet site**

1. **Functional and non-functional specification**

E-Cheat sheet site is a system that was initiated by Banchi Tech with the aim of sharing course material (exam, semester tests, etc.) of students among students. This site shall be available across all institutions of higher learning, ensuring that relevant content that students gets distributed among students. Students shall upload documents, files and images of the study materials that in turn gets downloaded by other student for a specific amount of fee.

**Functional requirements**

Features:

Upload of files

Download of files

Comment and chat feature

Main system

Priorities

Low : not vital to the functioning of the system

Medium : Standard system feature

High : critical feature

|  |  |
| --- | --- |
| Requirement name [#1] | User Account login |
| Priority | high |
| Description | Valid user account gives user access to system functionality. The users need to login first in order to get access to functions of the system. |
| Motivation |  |
| Dependencies | #2 |

|  |  |
| --- | --- |
| Requirement name [#2] | User registration |
| Priority | high |
| Description | User needs to register with the system in order to have valid accounts that will be used to access the system. User must have a profile page that they can update their details. |
| Motivation |  |
| Dependencies |  |

|  |  |
| --- | --- |
| Requirement name [#3] | Buy subscription/ downloads |
| Priority | high |
| Description | Users need to buy number of downloads for user to be able to download files on the system. The downloads are bought in the form of points. When user logs into the system, user need to see subscription balance, and view calculated range of course material that can be downloaded |
| Motivation |  |
| Dependencies | #1 |

|  |  |
| --- | --- |
| Requirement name [#4] | Upload content |
| Priority | high |
| Description | This the process of adding they actual system content. The source of system content is registered users of the system who plays a roles of uploaders. Content such as pdf, doc, docx, png, jpg and txt files are uploaded with the actual content of the system. |
| Motivation |  |
| Dependencies | #1 |

|  |  |
| --- | --- |
| Requirement name [#5] | Download content |
| Priority | high |
| Description | This the process where users actually gets access to the relevant study material they need. Download is done as a form of sending an email a copy to the downloader. |
| Motivation |  |
| Dependencies | #1,#2 |

|  |  |
| --- | --- |
| Requirement name [#6] | Rate uploads |
| Priority | high |
| Description | For every uploaded material the quality and relevancy gets to be judged by users through rating of the document. |
| Motivation |  |
| Dependencies | #4,#5 |

|  |  |
| --- | --- |
| Requirement name [#7] | Score download points |
| Priority | medium |
| Description | The download points can either be bought with money or earned through the number of uploads the user made on the system. |
| Motivation |  |
| Dependencies | #4 |

|  |  |
| --- | --- |
| Requirement name [#8] | Add comment |
| Priority | medium |
| Description | Users should be able to give comment on the uploaded content material. |
| Motivation |  |
| Dependencies | #4,#5 |

|  |  |
| --- | --- |
| Requirement name [#9] | Delete comment |
| Priority | low |
| Description | The comments user makes on the uploaded content gets removed. |
| Motivation |  |
| Dependencies | #8 |

|  |  |
| --- | --- |
| Requirement name [#10] | Update comment |
| Priority | low |
| Description | The comments user makes on the uploaded content gets updated. |
| Motivation |  |
| Dependencies | #8 |

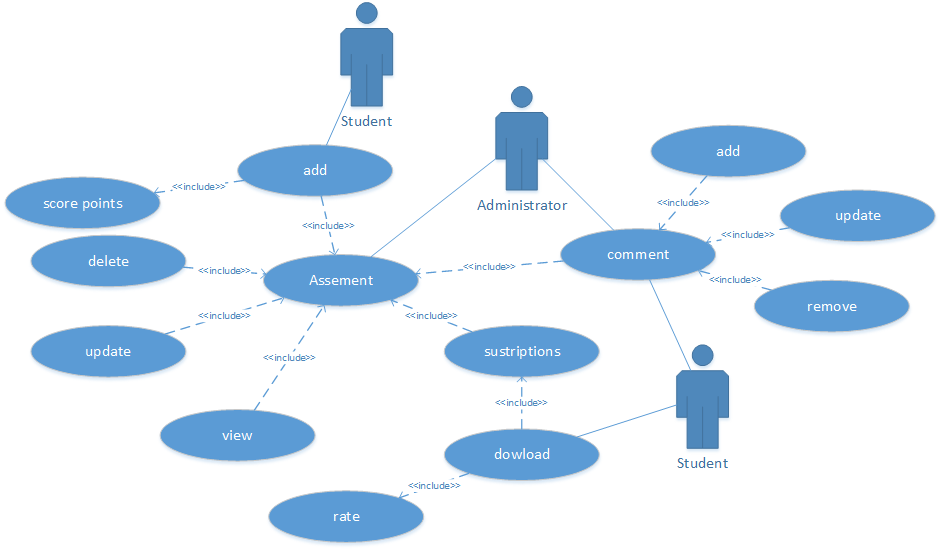
|  |  |
| --- | --- |
| Requirement name [#11] | View comment |
| Priority | low |
| Description | The comments user makes on the uploaded content gets viewed. |
| Motivation |  |
| Dependencies | #8 |

|  |  |
| --- | --- |
| Requirement name [#12] | Add disclaimer |
| Priority | high |
| Description | This functionality gives more of policy and details of the system on how to use the uploaded content. User gets to understand more info about the system. |
| Motivation |  |
| Dependencies |  |

|  |  |
| --- | --- |
| Requirement name [#13] | Audit log |
| Priority | high |
| Description | Keeps track of all interaction on the system. |
| Motivation |  |
| Dependencies |  |

1. **Use cases**

Main system



1. **Domain model**

|  |
| --- |
| Assessment |
| Add  Delete  Update  View  getAllAssesment  getAssesmentByKey |

|  |
| --- |
| uploads |
| Add  Update  delete  trackUpload  count uploads |

|  |
| --- |
| Audit log |
| Note system access  Note uploads  Note downloads  Note subscriptions  Note assessments |

|  |
| --- |
| users |
| Add  Delete  Update  Block |

|  |
| --- |
| Subscriptions |
| Buy  Track  Allocate downloads  Deallocate downloads  Enable downloads  Disable downloads |

|  |
| --- |
| downloads |
| Add  Update  Track  Remove  Count |

1. **Nonfunctional requirements**

Security

Performance

Scalability

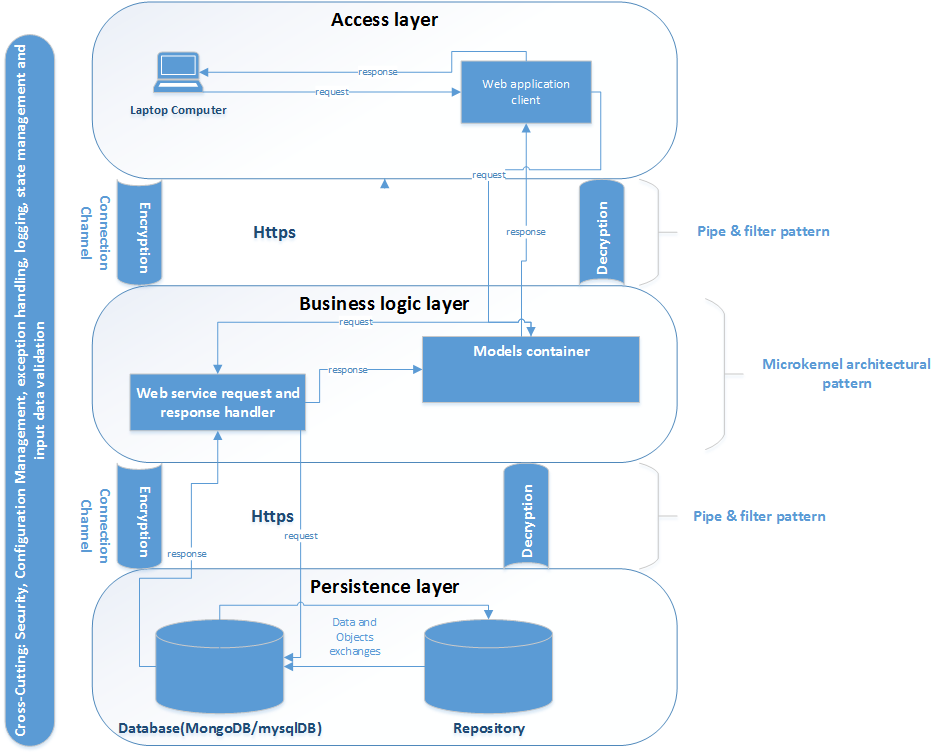
Availability

Usability

Maintainability

1. **Architecture**

Layered pattern



Client Server Pattern

