Department of Computer Science
North Carolina A&T State University

System Test Plan

COMP 496: Senior Design II

Fall 2020

Group 3
Annotext

Tiara Bell
Myles Neloms
Patrick Crump, Jr.

Revision History

Revision	Date	Author(s)	Description
0.1	10.01.2020	MN, TB, PCJ	Document Creation and DB Save
0.2	10.05.2020	MN, TB, PCJ	complete draft
0.3	10.12.2020	MN, TB, PCJ	release candidate 1
1.0	10.20.2015	MN, TB	Display Functionality
1.1	10.31.2020	MN	Button Functionality

Contents

List of Figures

1 Introduction

1.1 Product Concept

Annotext is a web based application that allows users to access PDF files through our system and annotate. These annotations will consist of highlights, footnotes, and summaries. Annotext uses a multi-threaded server connection process to allow for many users to constantly connect to the website and use the features that we offer. Each group and/or individual will be given a specific port on our domain to work from while they are annotating the PDF files they choose. When a user connects they will be given a "pane" to work from. In this pane they will be able to annotate the document. Other users will not be able to edit another users pane, unless given access by an administrator. Users are permitted to see the annotation of another user. We propose the following requirements, which encapsulate our product. [You must have at least 7 requirements, of which 5 are functional requirements and 2 are non-functional requirements. You must make the case that one of your functional requirements is a key innovation not currently solvable.] Functional Requirements 1. More than one student can add annotations at a time 2. User can rate different annotations (good or bad) based on their quality and usefulness 3. User can add comments/annotations/notes (text) 4. Maybe: Connect via a server 5. User should be able to create an account for access 6. User should be able to turn annotations on and off from view Non-Functional Requirements 1. Availabilty, Needs to be functional 24/7/365 2. Scalability, needs to allow room for growth in users

1.2 Product Scope

The purpose of Annotext is for students to create secure accounts on our application that will allow them to annotate PDFs of different subjects as they relate to the students course. Fro a step to step explanation, the student will first go and create an account on our website. From there they will be allowed to see the list of PDF files within our library. From this library they will be able to pick and choose what file they would like to annotate. This they takes the user to an annotate page that lets them view the whole document, write annotations and view listed annotations on the side.

1.3 Testing Scope

1.3.1 Test Inputs

We plan for the test to consist of multiple people accessing the Annotext website and creating multiple accounts. We also need tests that will allow us to have multiple users create annotations on different documents concurrently. They will also be required to up-vote and down-vote annotations to assist us in giving the better annotations priority over the worse annotations.

1.3.2 Test Environment

A web application that is hosted on a NCAT virtual machine hosted server. We are allowing access to all web browsers including Google Chrome, Mozilla Firefox, Microsoft Edge and more.

2 References

This section provides information about our system requirement specification, and the architecture design specification along with the design diagram. This section will cover in detail what systems our project will require to function properly. The architectural design specification displays how our layers are implemented.

2.1 System Requirement Specification

The SRS is important because it provides feedback to the customer and acts as the customerâs assurance that the development organization understands the issues or problems to be solved and the software behavior necessary to address those problems. It also breaks down the problem into smaller parts, solidifies ideas, and serves as the parent document for testing and validation strategies that will be applied to the requirements for verification.

Table 2: Customer Requirements

SRS#	Requirement	Description	Priority
#01	Account Registration	create an account to login to access the web app	X - high
#02	Add and Save Annotations to PDFs	Add and save notes on a pdf and link them to the pdf	X - high
#03	Rate Annotations	Rate annotations on their usefulness	X - medium
#04	Toggle on/off annotations	Ability to hide annotations from view	X - high
#05	Server Connection	Annotations are saved in a database located on our own server	X - high
#06	Multiple Accounts have access simultaneously	More than one account can add annotations at the same time	X - high
#07	Reply to other Annotations	Comment style replies to previous annotations	X - medium

2.2 Architecture Design Specification

2.2.1 Architecture Design Diagram

2.2.2 Data Flow Definitions

Definitions:

- Server: Connects user with application
- Router: Transports userâs request to the Controller
- Controller: Handles HTTP requests
- Model: Application data and and behavior in terms of its problem domain
- Database: Organized collection of data
- View: HTML markup that is displayed to the user

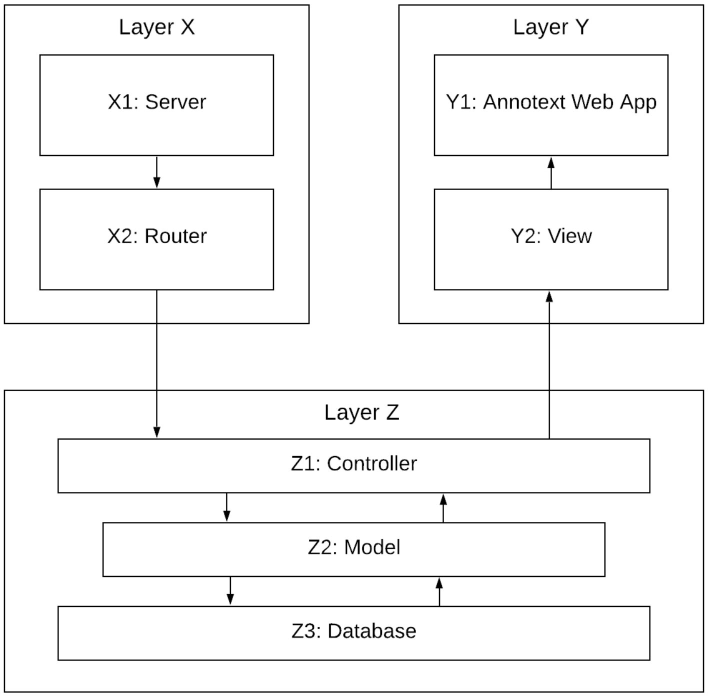

Figure 1: Architecture Design Diagram

- Annotext Web App: Our web application

2.2.3 X Layer

This layer is about setting up a connection with the server and sending the request to the appropriate party. Once the HTTP request is sent from the clientâs system, a connection is made with our server to accept the request. The request is then transported to itâs intended location via the Router. The information required to send the data request to its destination is embedded in the HTTP request.

2.2.4 Y Layer

This layer is about viewing the data the client requested. Once the request has been processed and returned, the View class then formats the information in a format conducive for user interaction and use. The View is displayed on our web application, Annotext and the data cycle will begin again when the user makes another request.

2.2.5 Z Layer

This layer is about retrieving the data and transferring it back to the client in a user friendly format. The controller handles the HTTP request of the user. The specific Controller is based upon the HTTP request. Once the Controller is chosen it then utilizes the Model layer to search and pull the data from the database. Once the data has been retrieved, the Model returns this information to the Controller and the Controller routes it to the appropriate View

2.3 Detailed Design Specification

2.3.1 Module Descriptions

Module X:

This module is uses to show data connections and data requests from our layer X. These requests then sent to our layer Z which uses controllers and models to navigate the user and allow for our core functionalities to work properly. After the processing through the Z layer, the Y layer takes precedence at this point. The layer is used then for displaying what is within our databases and documents that the user wishes to view.

2.3.2 Module Decomposition

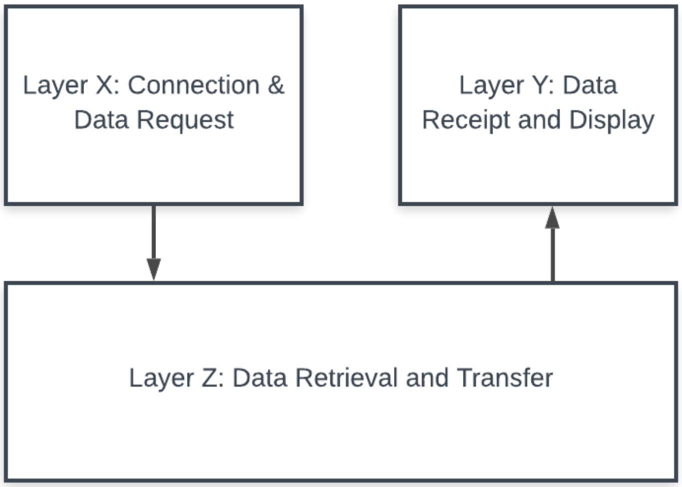
Layer X: Connection & Data Request
Layer Y: Data Receipt and Display
Layer Z: Data Retrieval and Transfer

Figure 2: Data Flow Diagrams

2.3.3 Data Flows

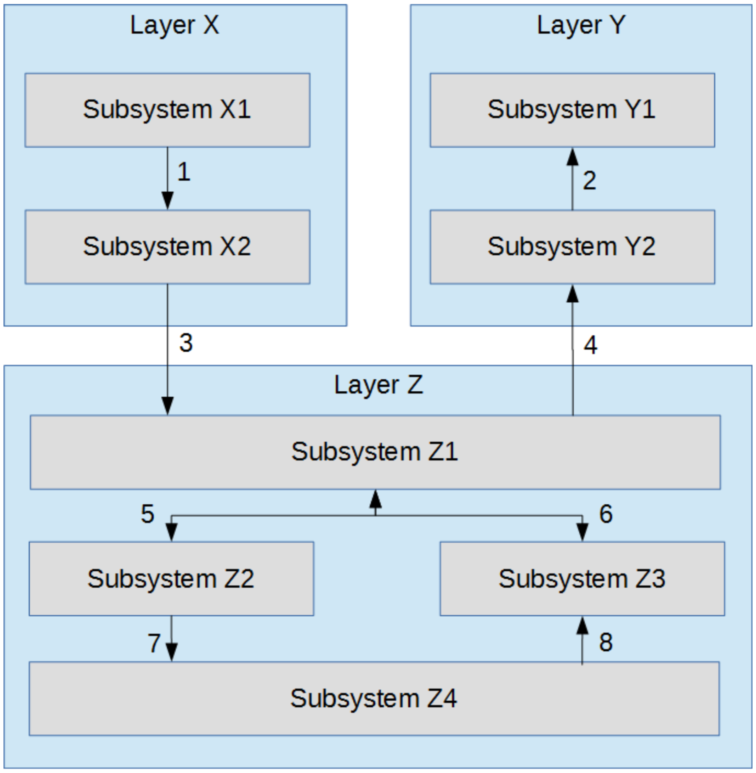

Figure 3: Data Flow Diagrams

3 Test Items

3.1 Hardware Tests

There are not hardware tests associated with the product.

3.2 Unit Tests

3.2.1 Connection / Request Subsystem

Table 3: Connection / Request Unit Test Table

Test ID	Module	Input	Output	Test	Priority
#x1	Server	User Request	Redirection	Connection to server	high

3.2.2 Data Retrieval and Transfer Subsystem

Table 4: Data Retrieval and Transfer Unit Test Table

Test ID	Module	Input	Output	Test	Priority
#x2	Database	Annotation and user data	view of requested material	Information stored in database	high

3.3 Component Tests

3.3.1 Data Retrieval and Transfer

Table 5: Data Retrieval and Transfer Test Table

Test ID	Subsystem	Input	Output	Test	Priority
#x3	Data Retrieval and Transfer	Annotations and user requests	annotations, pdfs, redirection	Annotation and pdf viewing, account registration and overall site navigation is tested here.	high

3.4 Integration Tests

Table 6: Integration Test Table

Test ID	Layer	Input	Output	Test	Priority
#x4	Data Retrieval and Transfer	User data and requests	Retrieved data and responses	The database saves user inputs and displays their requested information back to them.	high

3.5 System Validation

Table 7: System Validation Test Table

Test ID	Validation	Input	Output	Test	Priority
#x5	Server	Users access and interact with web app	Actions receive accurate responses	Users should be able to interact with the web app as systematically designed.	high

4 Risks

We are testing the functionalities that we created in Visual Studios MVC and Entity Framework

B. Having multiple users access the annotations at the same time
C. The ability to create an account
D. The ability to access the annotations

There are some inherent software risks such as complexity; these need to be identified.

A. Account Security
B. Multiple interfaces
C. Impacts on Client
D. Government regulations and rules

Complete the following table and remove all instructional text up to this point.

4.1 Risk Table

Table 8: Risks

Test ID	Risk	Impact	Severity	Mitigation Strategy
#01	Information Breach	Customer trust and data authenticity	High	We use secure libraries within our web application that can prevent an information breach
#02	Change of Annotation	Cause confusion to students	Medium	We have created our annotation database to only create the annotation and be able to edit it by the user.

5 Features to be Tested

This section describes all the requirements and features for your product that will be tested. For each requirement, briefly describe the feature and give a test approach.

5.1 Customer Requirements

5.1.1 Requirement ID: Annotation submission and page View Continuity

Description: We want Users to be able to submit annotations to each pdf while viewing the referenced pdf. We want the set fields of "VoteVal","Author",and "filename" in each pdf to be set without user input. And after the submission of an Annotation we want the PDFs viewability to be continuous.
Test Approach: Submit several Annotations for each PDF and check the database to assure that the value of "VoteVal" is set to 0, and the filename and Author are saved appropriate to the Pdf being viewed and the User logged into Annotext.

5.1.2 Requirement ID: List view of annotations

Description: List view of annotations are available for each PDF on the appropriate page. List updates as annotations are submitted and is sorted by vote val.
Test Approach: Log in as a User and submit and annotation. Check if List is updated. Vote up and down different annotations and see if list is updated.

5.1.3 Requirement ID: Voting functionality

Description: Users are able to vote for annotations and influence the sorting of said annotation.
Test Approach: Log in as user with dummy annotations(both with voteVal = 0) and vote for one, check if the vote was taken into account in the sorting of the list.

5.2 Performance Requirements

5.2.1 Requirement ID: Multiple Users can use website

Description: Multiple Users can use website at the same time to view and annotate separate or the same PDFs. Without unreasonable
Test Approach: Log in as a group using our Servers URL "annotext.ncat.edu". Annotate on the same and different PDFs to check the websites performance with multiple users making entries to the database.

5.3 Maintenance and Support Requirements

5.3.1 Requirement ID: Adding PDFs to database

Description: Force Admin Authentication to add PDFs to database
Test Approach: Log in as a normal user and try to access the path "PDFs/Create". This should not be possible.

6 Features not to be Tested

6.1 Security And Safety Requirements

6.1.1 Requirement ID: Login Security

Description: User passwords and login credentials are saved Safely and appropriately to the database.
Why Not: "Asp.Net MVC core Identity" allows for the scaffolding of a secure Login page including secure and Credential storage along with user role identification and Authorization checks for selected methods.

7 Approach

7.1 Strategy

Our team submitted several annotations for each PDF and check the database to assure that the value of "VoteVal" is set to 0, and the filename and Author are saved appropriate to the PDF being viewed and the User logged into Annotext. Our team is also logging in as a User and submit and annotation. Check if List is updated. Vote up and down different annotations and see if list is updated.

7.2 Tools

- Entity Framework
- MVC
- Visual Studio

7.3 Core Functionality

- Display PDF documents on screen
- Add Annotations to the selected PDF document
- View Submitted Annotation to refer to while reading

8 Item Pass/Fail Criteria

8.1 Unit Tests

Table 9: Module pass/fail criteria

Test ID	Module	Input	Output
#x1	Database: Are annotations saved correctly? Are users accessing the requested information accurately?	User Input or Interaction	Requested information

8.2 Component Tests

Table 10: Subsystem pass/fail criteria

Test ID	Component	Input	Output
#x1	Account Creation	Successful Registration	Successful Login

8.3 Integration Tests

Table 11: Integration pass/fail criteria

Test ID	Integration Layer	Input	Output
#x1	Annotation Interaction	User View and input (add annotation or rate annotation or comment on another annotation)	Displayed information to the user

8.4 System Validation

Table 12: Validation pass/fail criteria

Test ID	Validation	Input	Output
#x1	Server	Users access and interact with web app	Actions receive accurate responses

9 TEST DELIVERABLES

- Group Meeting Testing Log - Team meeting Minutes on decisions made when searching for solutions.
- User Testing Logs - Notes from Users outside of our groups logging their experience on "annotext.ncat.edu".
- Database Migration History Log - Database changes and logs of Schema used to explore possible solutions.
- Git Hub Push History - Push history of code after each iteration of solution exploration.

10 Test Schedule

Table 13: Connection / Request Unit Test Table

Task	Start Date	End Date	"Predicted" hours
Test Annotation	11/28/20	12/06/20	15
View Annotations	11/28/20	12/06/20	15

REFERENCES