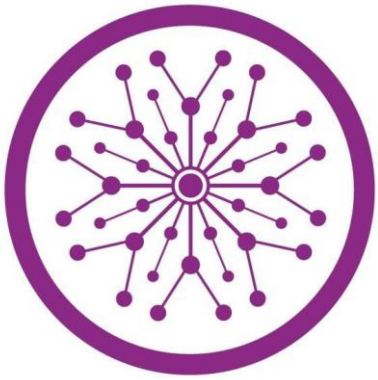
Dynamic Document Management System for MS Word

Final Year Project

Session 2019-2021

A project submitted in partial fulfillment of the degree of

Master in Information Technology



Department of Information Technology

Faculty of Computer Science & Information Technology

The Superior University Lahore

Fall 2021

Type (Nature of project)	[✓] **D**evelopment [ ] **R**esearch [ ] **R**&**D**
Area of specialization	Web Development

Project Group Members

Sr.#	Reg. #	Student Name	Email ID	*Signature
(i)	Mitm-s20-010	Saqib Iqbal	Mitm-s20-010@superior.edu.pk	
(ii)	Mitm-f19-044	Laraib Anwar	Mitm-f19-044@superior.edu.pk	

*The candidates confirm that the work submitted is their own and appropriate credit has been given where reference has been made to work of others

Plagiarism Free Certificate

This is to certify that, I **Saqib Iqbal** S/D of **Muhammad Iqbal** group leader of FYP under registration no **Mitm-s20-010** at Information Technology Department, The Superior University, Lahore. I declare that my FYP proposal is checked by my supervisor and the similarity index is _____ that is less than 20%, an acceptable limit by HEC. Report is attached herewith as Appendix D.

Date: Name of Group Leader: **Saqib Iqbal** Signature: _____

Name of Supervisor: Muhammad Javaid Iqbal

Designation: Lecturer

Signature: _____

HOD: Dr. Asad Ali Naqvi

Signature: _____

Project Report

Dynamic Document Management System for MS Word

Change Record

Author(s)	Version	Date	Notes	Supervisor's Signature
Saqib iqbal	1.0	11th June,2020	Original Draft	
Saqib Iqbal Laraib Anwar	2.0	25th June,2020	Changes Based on Feedback from Supervisor	
Saqib Iqbal Laraib Anwar	3.0	8th Jul,2020	Changes Based on Feedback from Faculty	
Saqib Iqbal Laraib Anwar	4.0	20th Jul,2020	Added Project Plan	
Saqib Iqbal Laraib Anwar	5.0	15th Aug, 2020.	Changes Based on the errors	
Saqib Iqbal Laraib Anwar	6.0	1st Oct,2020	Changes in the Final Report Based on Feedback from Supervisor	
Saqib iqbal	7.0	12th Nov,2020	Changes in the Project Based on Feedback from Supervisor	
Saqib Iqbal Laraib Anwar	8.0	17th Dec,2020	Changes in the Final Report Based on Feedback from Supervisor	

APPROVAL

PROJECT SUPERVISOR

Comments: _____

_____

Name: _____

Date: _____ Signature: _____

PROJECT MANAGER

Comments: _____

_____

Date: _____ Signature: _____

HEAD OF THE DEPARTMENT

Comments: _____

_____

Date: _____ Signature: _____

Dedication

This work is dedicated to our parents, who believed us and giving a chance to get knowledge, makes better life himself, popularizes the name of his university and his country of Pakistan also. We are also dedicating to our respected teachers and specially our supervisors, who help us to making and completing this project. We are also thanks to our chairman and all the superior team, who provides a platform to enhancing and creating our abilities and polished us for every type of situation.

Acknowledgements

Alhamdulillah, all praises are for **Allah Almighty**, who is all knowing, who always guides and leads to success and all respects for **The Holy Prophet (ﷺ)** who is exclusively the crown of all **Allah's messengers**. We are warmly thankful to our respected Teachers and specially **"Mr. Muhammad Javaid Iqbal"** for his continuous guidance, supervision and suggestion during this research work. We are also thankful of our superior management and also special thanks of our **chairman "Prof. Dr. Ch. Abdul-Rehman"**, which provide us an excellent platform that is the **"Superior University**

Executive Summary

Dynamic Document Management System for MS Word all over the world many Microsoft Word Add-ins are available and provide different type of facilities but do not provide dynamic changes. Due to which, users have to look for alternatives to make their template according to their requirements. Also, there is no option available in world to fill templates. Microsoft word doesn't allow to create dynamic templates due to which users need to fill template manually. There should be an option for creation of dynamic templates. These templates should be fillable using data stored in database. Also, they should be able to help users writing past financial values for reports. This study will be based on solving the above-described problem using MS Office Add-ins. We are actually building an add-in for Microsoft Word whose main purpose is to manage dynamic templates. So that a user can easily update it through the database. This add-in provides a way to upload data to data base and then this data can be bided to word document using placeholders. These placeholders then can be added to a document and can have variable values to provide a dynamic feature inside MS Word.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Chapter 1: Introduction

This Chapter is all about the introduction of my FYP project Dynamic Document Management System for MS Word. It defines what the Dynamic Document Management System for MS Word project is and its background, including its previous existing solutions. It also compares Dynamic Document Management System for MS Word with other systems available. As there are more than 1.2 billion people in this world which are using Microsoft products and are getting benefits.

So, as users increases, functionality demands also increase. This project is based on the user demands of dynamic template creation in MS Word. Although it is offered by other companies at different platforms but yet it is not offered in MS Word. This template creation is very useful to the users as it is of high demand in different business organizations. And due to it the documents will be more secured. By this application, the need of third party for data saving will be truncated as this app will save the data of user tables and documents.

1.1. Background

All over the world many Microsoft Word Add-ins are available and provide different type of facilities but do not provide dynamic changes. Due to which, users have to look for alternatives to make their template according to their requirements. Also, there is no option available in world to fill templates.

1.2. Motivations and Challenges

We are providing a solution of the office problems in the form of technology. That is the point which motivated us to develop this product at any cost because we are changing the difficulties of work place word through technology and overcome the communication gap of the different departments. This is the big challenge to educate and motivate the customers to moving on Dynamic Document Management System for MS Word.

1.3. Goals and Objectives

Goals and Objective of this project is very clear to control the problems of work place system and facilitate to the peoples. We are actually building an add-in for Microsoft Word whose main

purpose is to manage dynamic templates. So that a user can easily update it through the database. This add-in provides a way to upload data to database and then this data can be bind to word document using placeholders. These placeholders then can be added to a document and can have variable values to provide a dynamic feature inside MS Word.

1.4. Literature Review/Existing Solutions

The number of Microsoft Office users today, estimated to be over 1.2 billion, can easily form their own country. And it's safe to say that **Microsoft Word** users are a big part of that productive population. **Microsoft Office** and **Microsoft Excel** provide their customers static templates, due to which, offices have to look for alternative **Dynamic document management system for MS Word** to make their template according to their will and requirements. Also, there is no option available in word to fill templates. It is unique than the others.

1.5. Gap Analysis

The gap was that all other platforms which are offering same type of operations like dynamic template creation were not easy to use and were not user friendly as they are not easily available.

So as Microsoft word is widely used everywhere like in offices, homes and universities. Over 1.2 billion users use this product of Microsoft. So, it was like need of the hour to bring the dynamic template creation at big platform like this. That's why we stepped forward and took this initiative to bring more ease to the users of software.

1.6. Proposed Solution

This study will be based on solving the above-described problem using MS Office Add-ins. We are actually building an add-in for Microsoft Word whose main purpose is to manage dynamic templates. So that a user can easily update it through the database. This add-in provides a way to upload data to data base and then this data can be binded to word document using placeholders. These placeholders then can be added to a document and can have variable values to provide a dynamic feature inside MS Word.

1.7. Project Plan

In Project Plan there are two things WBS and Gant Chart. In WBS we divide the milestones into sub tasks to achieve the goals and objectives. Assign tasks to each team member according to their work experience. In Gant chart we decide which tangible things have to be deliver in which duration.

Phase One	Planning	5-7-2021
Phase Two	Designing phase	14-10-2021
Phase Three	Final Documentation	1-12-2020

Our Quality plane will be measured by the following parameters:

- User satisfaction regarding the service
- App Response Time
- User Service Time
- Security of personal Data

Table 1:Project plan

1.7.1. Work Breakdown Structure

Dynamic Document Management System for MS Word

• **Initialization**
• Inception
• Problem
• Market Analysis
• Conclusion of Problem
• Solution
• **Planning**
• Elicitation
• Analyze Problem
• **Design SRS**
• Documentation
• Analyze Documents

• Review Documents
• feasibility Report
• **Diagrams**
• Use Case
• Domain Model
• Sequence Diagram
• Activity Diagram
• State Chart Diagram
• Class Diagram
•
• **Implementation/Execution**
• Cover modules
• Review Covered Modules
• **Testing**
• Testing Tools used for the testing web Application
• **Testing Methods**
• Unit Testing
• Integrate Testing • Alpha testing • System testing
• **Deployment**

Table 2 Work background structure

Sample WBS

- **Project Management**
  - Work Breakdown Structure (WBS)
  - Roles & Responsibility Matrix
  - Change Control System
- **Reports / Documentation**
  - Final Documentation Introduction
  - Literature / Markey Survey
  - Requirements Analysis
  - System Design
  - Implementation
  - Testing & Performance Evaluation
  - Conclusion & Outlook
  - End User Documentation
  - Application Administration Documentation
  - System Administrator Documentation
- **System**
  - Development Environment
    - IDE
    - Version Control
    - Server
    - Database
  - Presentation Layer

1.7.2. Roles & Responsibility Matrix

The purpose of roles & responsibility matrix is to identify who will do what.

WBS #	WBS Deliverable	Activity #	Activity to Complete the Deliverable	Duration (# of Days)	Responsible Team Member(s) & Role(s)
1	**Initialization**	1		**14**	saqib
		2	Inception	3	Team
		3	Problem	2	Team
		4	Market Analysis	5	Team
		5	Conclusion	2	Saqib
		6	Solution	2	Saqib
2	**Planning**	7		**9**	Team
		8	Elicitation	4	Team
		9	Analyze the Problem	5	
3	**Design SRS**	10	SRS	12	Team
4	**Documentation**	11	Document	14	Team
5	**Analyze the Document**	12		2	Team
6	**Revised Document**	13	Revised Document	7	Laraib
7	**Feasibility Report**	14	Report	2	Laraib
8	**Diagrams**	15		7	Team
9	**implementation**	16	Modules	120	Team
10	**Testing**	17	System testing	35	Team
11	**Deployment**	18		17	Team

Table 3:Roles and responsibility matrix

1.8. Gantt Chart

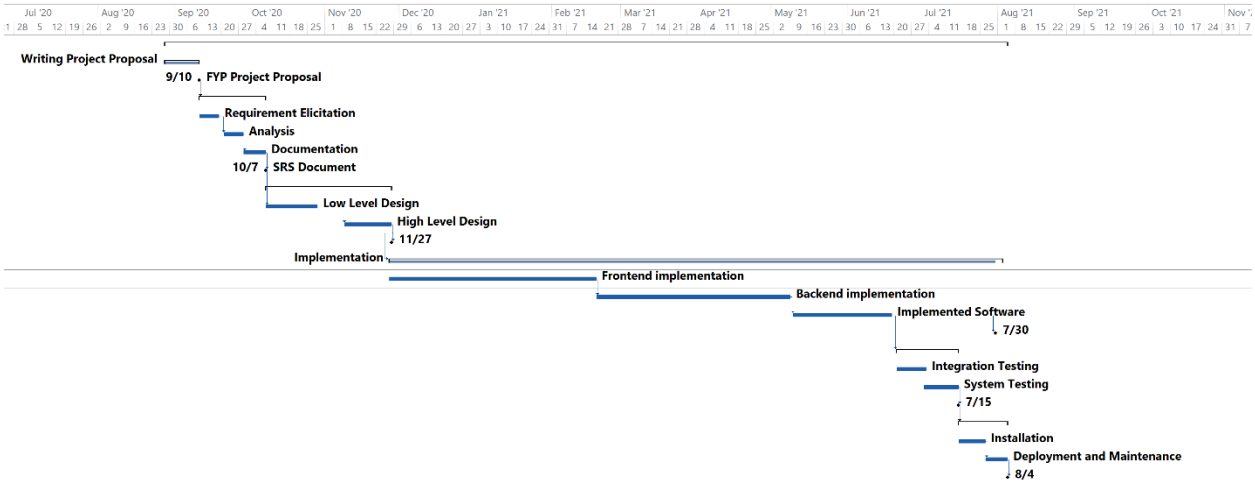


Figure 1: Gantt chart

1.9. Report Outline

Documentation is the backbone of the project. In documentation, we decide the resources for our project to complete. In this documentation have covers all possible aspects of the project that are necessary to complete a project. There are:

- Introduction, Background and Scope of the Project
- Purpose, Gap Analysis and Proposed Solution of the Project
- System Features
- Functional and non-functional of the System
- Use Case Analysis: Use Case Model, fully dressed Use Case
- System Designs
- System Implementation
- System Flow
- Version Control

Chapter 2

Software Requirement Specifications

Chapter 2: Software Requirement Specifications

2.1. Purpose

We are actually building an add-in for Microsoft Word whose main purpose is to manage dynamic templates. So that a user can easily update it through the database. This add-in provides a way to upload data to data base and then this data can be bided to word document using placeholders. These placeholders then can be added to a document and can have variable values to provide a dynamic feature inside MS Word.

2.2. Document Conventions

The conventions we used in the document are as follows:

Header = 20px

Main Heading = 16px

Sub-heading = 14px

Content = 11px

2.3. Intended Audience and Reading Suggestions

The intended audience:

- ✓ Teachers
- ✓ Manager
- ✓ Admin Staff
- ✓ Programmers
- ✓ Researchers
- ✓ Students

2.4. Product Scope

This product is actually beneficial for making documentation. The user can use it anywhere to dynamically change the content or the document. The user can also be able to make document dynamically and can use it easily in Microsoft Word. For more details, please read the documentation below.

2.5. References

- [Online Available] https://medium.com/@enisinanaj/writing-a-software-requirements-specification-document-97d622805aef Accessed by 20 Jan, 2022
- [Online.Available]https://www.slideshare.net/EmrulEmon1/software-requirement-specificationsrs Accessed by 25 Jan, 2022
- [Online Available] https://techwhirl.com/writing-software-requirements-specifications/ Accessed by 10 fab, 2022

2.6. Overall Description

2.6.1. Product Perspective

Our product is a Microsoft Word Add-in which will allow the users to make dynamic documents which are currently Word is not offering. So, it's an entire new approach towards market and user requirements. This product is actually an addition to the services of Microsoft Word. We have chosen it because it is need of today's market but isn't available on the word yet. So, basically it is a unique approach towards customer requirements.

2.6.2. User Classes and Characteristics

There are two user classes:

1-Admin:

This user can manage the add-in through admin portal like it can add or delete different functionalities in the add-in and can also update with the passage of time or with the requirement specifications. This user can access the product back-end and database.

2-User:

This user can use the functionalities of the add-in for its requirements. Like it can make a dynamic document template according to the specifications and can also make changes to it if required. This user can just use the product functionalities and cannot access the backend and database of the product.

2.6.3. Operating Environment

This will operate in Microsoft Word 2017 to onwards. There are no hardware specifications for this product. This can run anywhere where the Microsoft Word can operate. This add-in will support Word-API 1.3.

2.6.4. Design and Implementation Constraints

The only limitation for this product is version of Microsoft Word which the user is operating. If it is below than Word 2017 then user is unable to use our product. So that the user must have Word 2017 or later versions for this product.

2.6.5. Assumptions and Dependencies

Assumptions:

- The code should be free with compilation errors/syntax errors.
- The product must have an interface that is simple enough to understand.

Dependencies:

- All necessary hardware and software are available for implementing and use of the tool.
- The proposed system would be designed, developed, and implemented based on the software requirements specifications document.
- End users should have basic knowledge of computers, and we also assure that the users will be given software training documentation and reference material.

2.7. External Interface Requirements

2.7.1. User Interfaces

For Admin:

- ✓ Admin is able is make changes in functionalities
- ✓ Admin is able to manage the product.
- ✓ Admin is able to add or delete more functionalities.
- ✓ Admin is also able to manage the users.

For User:

- ✓ User can create dynamic template.
- ✓ User can change Data Runtime.
- ✓ User can also retrieve data from database.

2.7.2. Hardware Interfaces

Server End hardware Requirements:

- ✓ HDD/SSD: 512gb or greater
- ✓ RAM: 4gb or greater
- ✓ CPU: Dual Core i5 or greater
- ✓ Backup media: mouse, keyboard and monitor is needed

User Hardware Requirements:

Any computer which supports Microsoft Word 2017 or later is applicable.

Table 4Hardware Interfaces

2.7.3. Software Interfaces

Server End Software Requirements:

- o Windows/IOS Operating System
- o Visual Studio 2019
- o SQL Server 2019
- o Microsoft Word 2019

User Software Requirements:

- ✓ Microsoft Word 2017 or later.

2.7.4. Communications Interfaces

- ✓ Chat through Admin panel
- ✓ Through Email

2.8. System Features

2.8.1. Dynamic Template

2.8.1.1. Description and Priority

Through this feature user can add a dynamic template according to its requirements using data from database. This is main feature of the product.

2.8.1.2. Stimulus/Response Sequences

System response towards the dynamic templates is very quick and effective. A user can add template within no time. Like if he clicks on create then he can see the template interface within nanoseconds.

2.8.1.3. Functional Requirements

Change Table Data According to input.

Input data in template dynamically from database.

2.8.2. Change Data Runtime

2.8.2.1. Description and Priority

Through this feature user can change data runtime in a template. As the template is dynamic so that the data in it is dynamically changeable.

2.8.2.2. Stimulus/Response Sequences

System response towards the dynamic templates is very quick and effective. A user can add template within no time.

2.8.2.3. Functional Requirements

User can change data runtime.

Template will change automatically according to data.

2.8.3. Retrieve Data from Database

2.8.3.1. Description and Priority

Through this feature user can connect the template through any database and can add data from that database.

2.8.3.2. Stimulus/Response Sequences

System response towards the dynamic templates is very quick and effective. A user can add template within no time.

2.8.3.3. Functional Requirements

User can connect any database with template.

User can change template according to the requirements.

2.9. Nonfunctional Requirements

2.9.1. Performance Requirements

- ✓ The system should never crash more than 10 minutes per month.
- ✓ it should have inbuilt error testing to identify invalid username/password the system should be able to handle large amount of data

2.9.2. Safety Requirements

- ✓ The system should be closed 24 hours per week for maintenance and elaboration.
- ✓ The database may get crashed at any certain time due to virus or operating system failure. Therefore, it is required to take the database backup so that the database is not lost.
- ✓ Proper UPS/inverter facility should be there in case of power supply failure.

2.9.3. Security Requirements

- ✓ System will have different types of users and every user has access constraints.
- ✓ The system should encrypt user's details by using AES-256 encryption system.
- ✓ The users should learn how to use the system at most in 2 hours.

2.9.4. Usability Requirements

Any familiar with using Microsoft Word operation can operate the system as it has a user-friendly user interface.

2.9.5. Reliability Requirements

- This system is available based on the user needs, can work properly, and buy, sell, rent properties efficiently, including safe data management of the properties.
- For invalid and malfunctioned operations, the system will reset to prevent data loss and safe operation within 5 seconds.

2.9.6. Maintainability/Supportability Requirements

- Microsoft Word 2017 or later.

2.9.7. Portability Requirements

Portability depends on the efforts needed to transfer application from one environment to other. Our application is easy to portable because it is developed in hardware independent fiction

2.9.8. Efficiency Requirements

- This product will be efficient in handling its users' data, and the primary purpose of this system is to bridge that gap between the user requirements and the system requirements.
- The product will be easy to run and secure, and reliable so the user can have a piece of satisfaction about the data.

2.10. Domain Requirements

User should have basic knowledge of Microsoft Word and also can be able to understand English because our product’s base language is English.

Chapter 3

Use Case Analysis

Chapter 3: Use Case Analysis

This chapter will describe about the use case of the system which will further illustrate the functional requirements of the systems and how functional requirements will interact will the user or actor.

Use case analysis is a technique used to identify the requirements of a system and the information used to both define processes and classes (which are a collection of actors and processes) which will be used both in the use case diagram and the overall use case.

System:

An online web-based application of local company which name is **LA-FAMILA TOURS AND TREKS**

System Level Use Case Diagram

There are some actors and use cases which are used to making a diagram of the system;

Actors

- User
- Admin

3.1. Use Case Model

Admin:

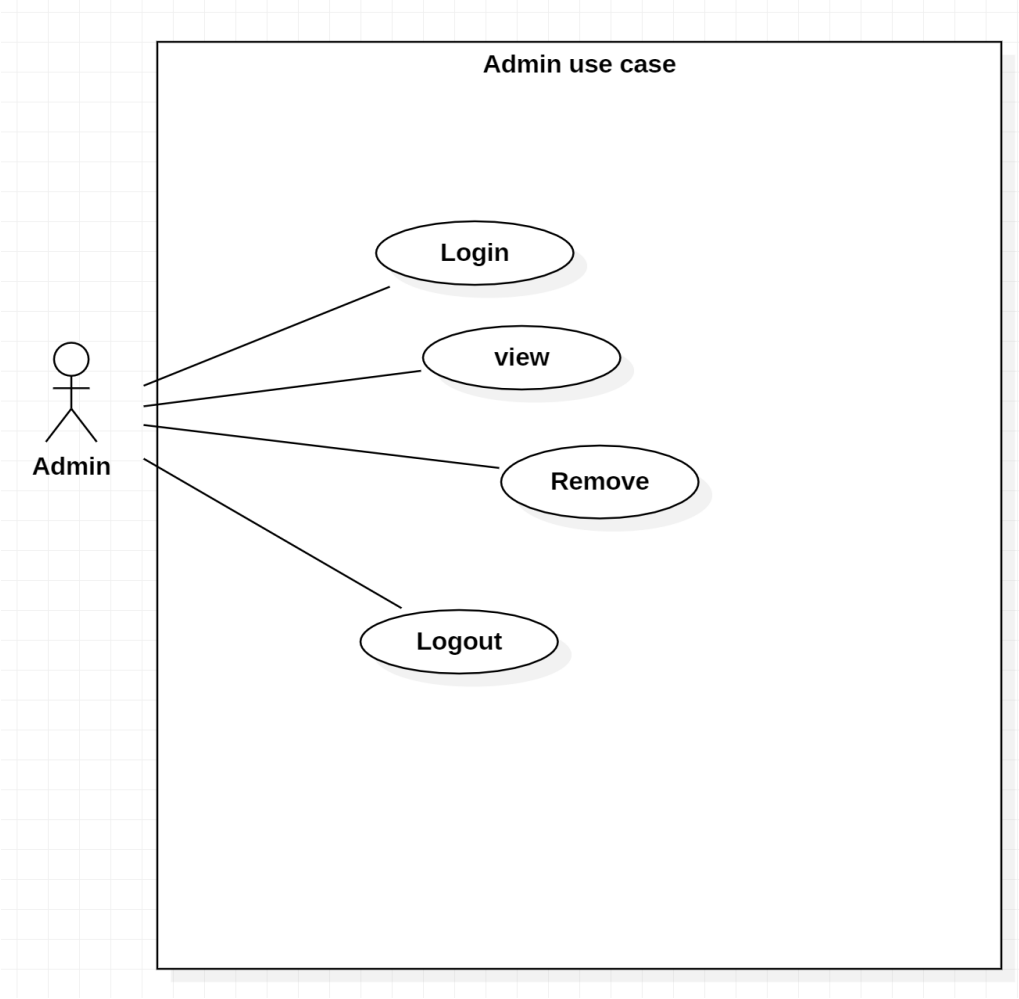


Figure 2: use case of admin

Figure 3:fully dressed use case

User:

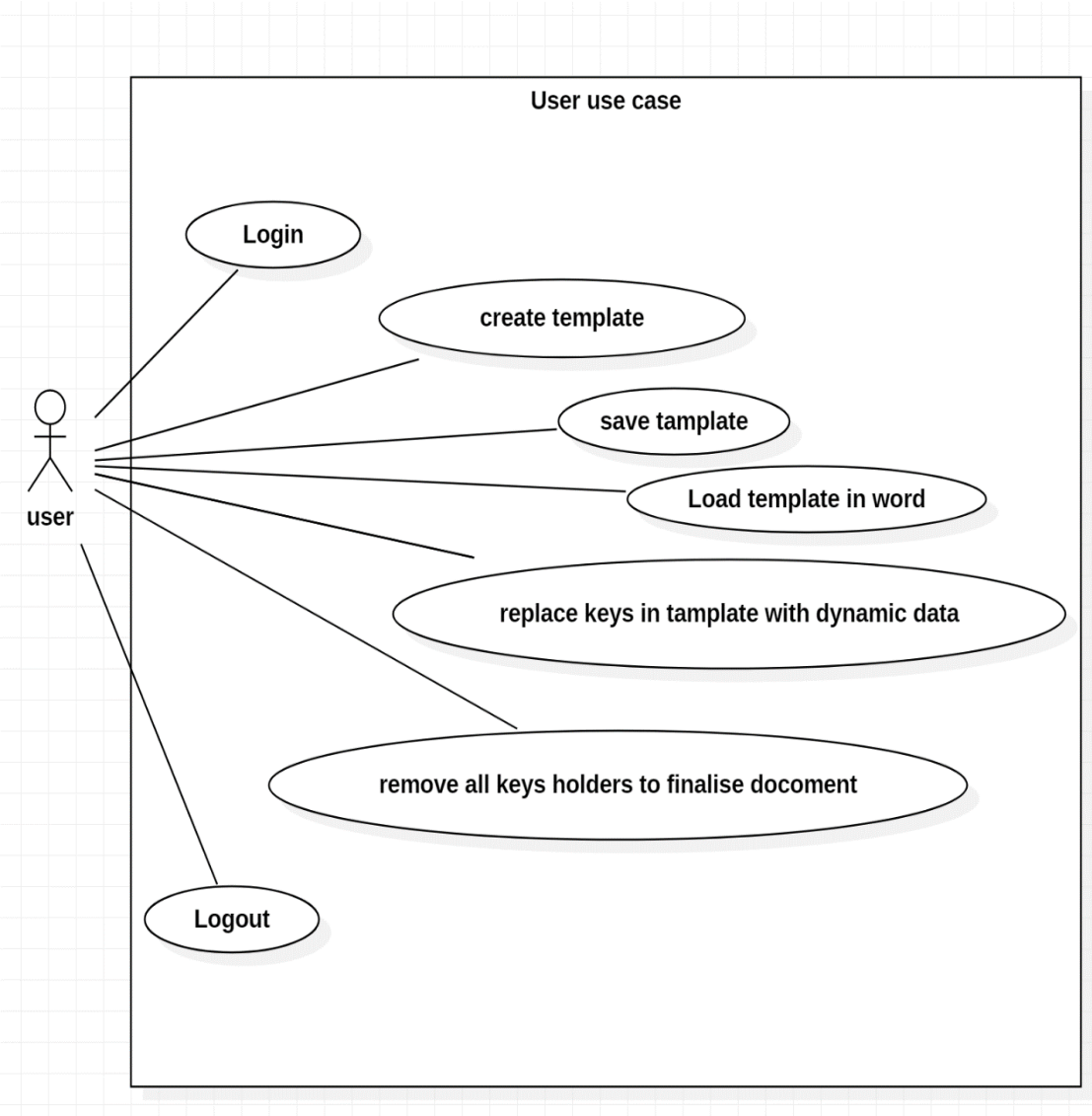


Figure 4: use case of user

3.2. Use Cases Description

3.1. Use Case: login

Actors: User and Admin

Use case description: To login an account

Pre-condition: User or Admin should have an account

Post-condition: User and Admin needs to perform their actions

Scenario:

- First splash screen appears and then login
- Enter user name and password
- Click on login.

Exception: User name or password can be wrong or forgotten

3.2. Use Case: Create template

Actors: User and Admin

Use case description: To create a new template

Pre-condition: User or Admin should have new template

Post-condition: User have already to create a template

Scenario:

- Enter user name and password
- Click on login.

Exception: User name or password can be wrong or forgotten

3.3. Use Case: Save Template

Actors: User or Admin

Use case description: User or Admin To do save a template

Pre-condition: User and Admin need to save a template

Post-condition: User or Admin have already exit template

Scenario:

- Enter user name and password
- Click on login.

Exception: User name or password can be wrong or forgotten

3.4. Use Case: Load Template In Word

1. **Actors:** User is actor of use case.

Use case description: To do load a template in MS word and start working on it.

Precondition: User should have to do load a template

Post-condition: User successfully to load a template

Scenario:

- Enter user name and password
- Click on login.

Exception: User name or password can be wrong or forgotten

3.5. Use Case: Replace keys in template with dynamic data

Actors: User or Admin actor of use case.

Pre-condition: user or admin replace and change the keys in templates by the dynamic data system.

Post-condition User or admin have already exit.

Scenario:

- Enter user name and password

- Click on login.

Exception: User name or password can be wrong or forgotten

3.6. Use Case:Remove all keys holders to finalize document

Actors: User

Precondition: User access to remove all keys holders to finalise document.

Post-condition: User have already removed.

Scenario:

- Enter user name and password
- Click on login.

Exception: User name or password can be wrong or forgotten.

3.7. Use Case: logout

Actors: User or Admin

Precondition: User or Admin should have an account.

Post-condition: User or Admin have already loged out

Scenario:

- When user or admin to login with a new account
- Then he/she logout first
- Then login with new password email

Chapter 4

System Design

Chapter 4: System Design

4.1. Architecture Diagram

An architectural diagram is a diagram of a system that is used to abstract the overall outline of the software system and the relationships, constraints, and boundaries between components. It is an important tool as it provides an overall view of the physical deployment of the software system and its evolution roadmap

.

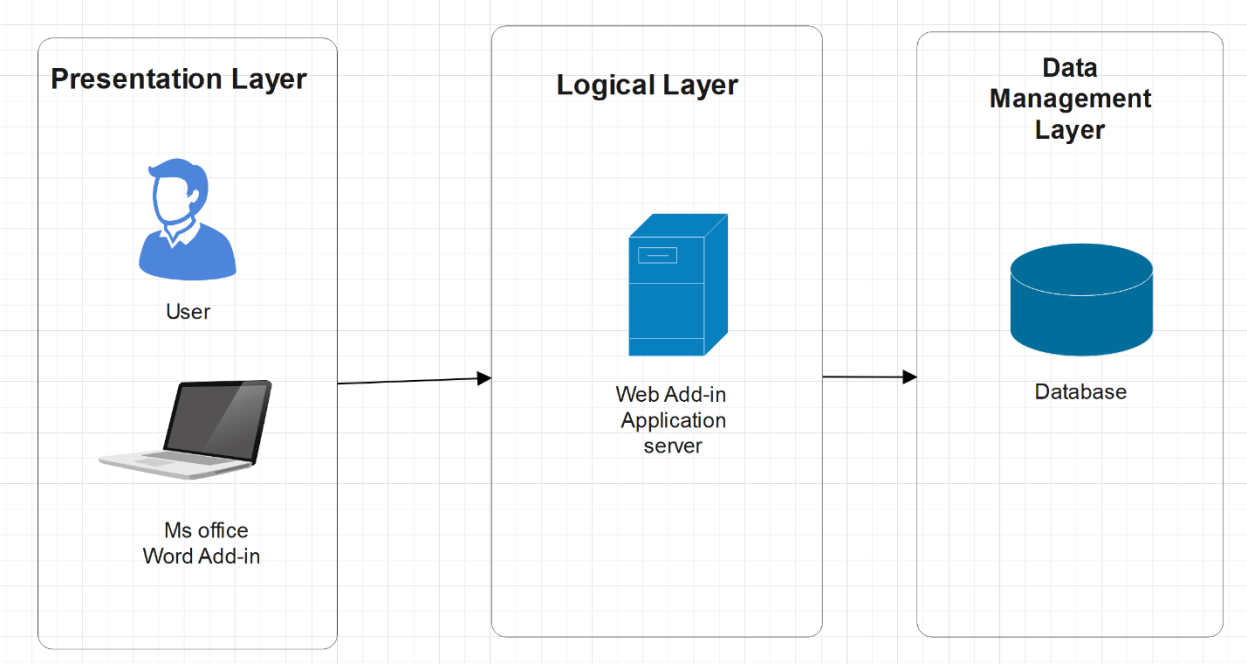


Figure 5:Architechture Diagram

4.2. Domain Model

Domain model is a conceptual model of the domain that incorporates both behavior and data. In ontology engineering, a domain model is a formal representation of a knowledge domain with concepts, roles, datatypes, individuals, and rules, typically grounded in a description logic

.

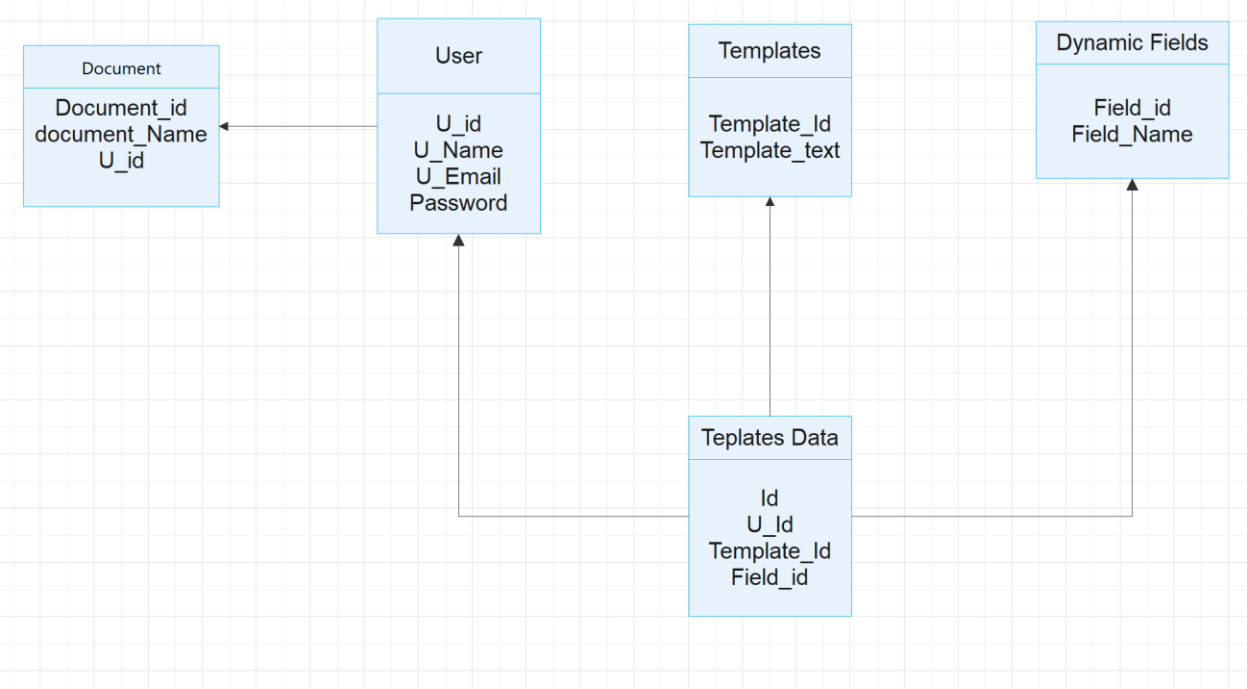


Figure 6:Domain Model

4.3. Entity Relationship Diagram with data dictionary

Entity relationship diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is an object, a component of data. An entity set is a collection of similar entities. These entities can have attributes that define its properties.By defining the entities, their attributes, and showing the relationships between them, an ER diagram illustrates the logical structure of databases.ER diagrams are used to sketch out the design of a database.

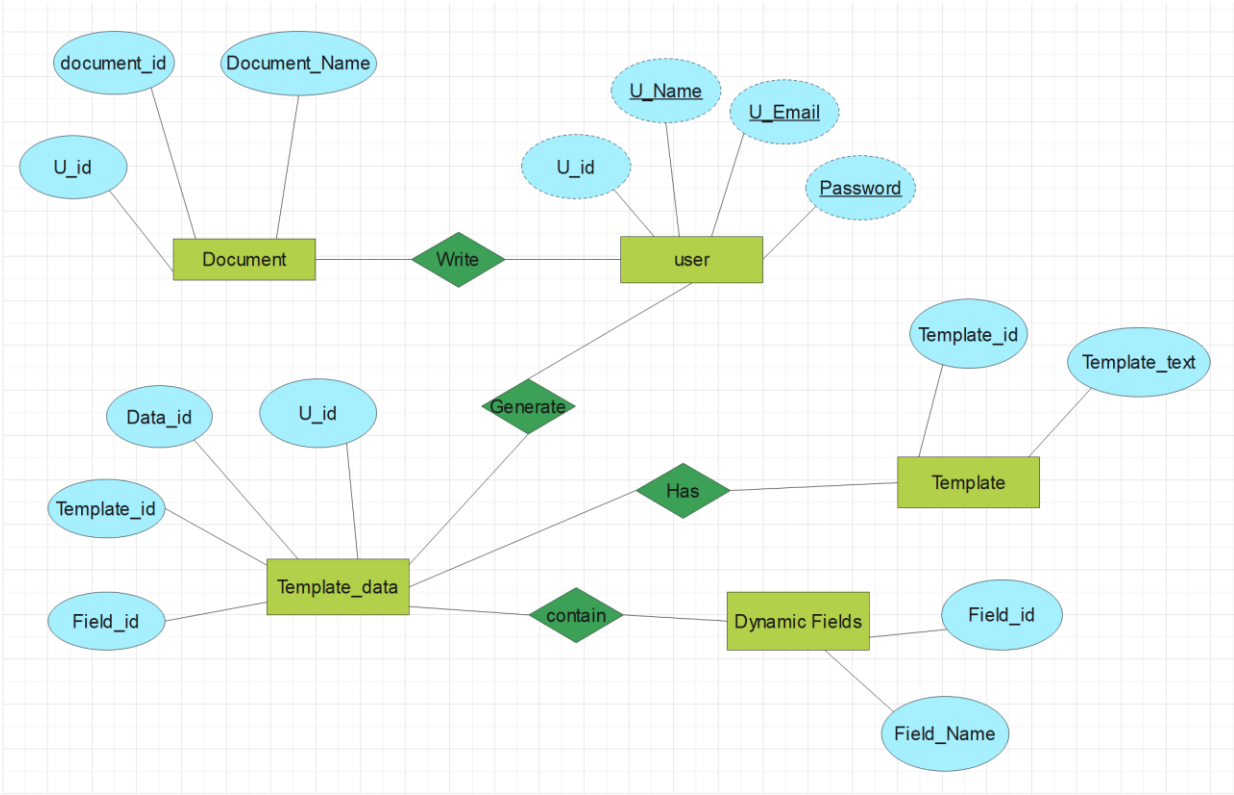


Figure 7:Entity Relationship Diagram

4.4. Class Diagram

The following Class Diagram of our project gives an overview of **Dynamic Document Management System** by displaying its classes, attributes and operations:

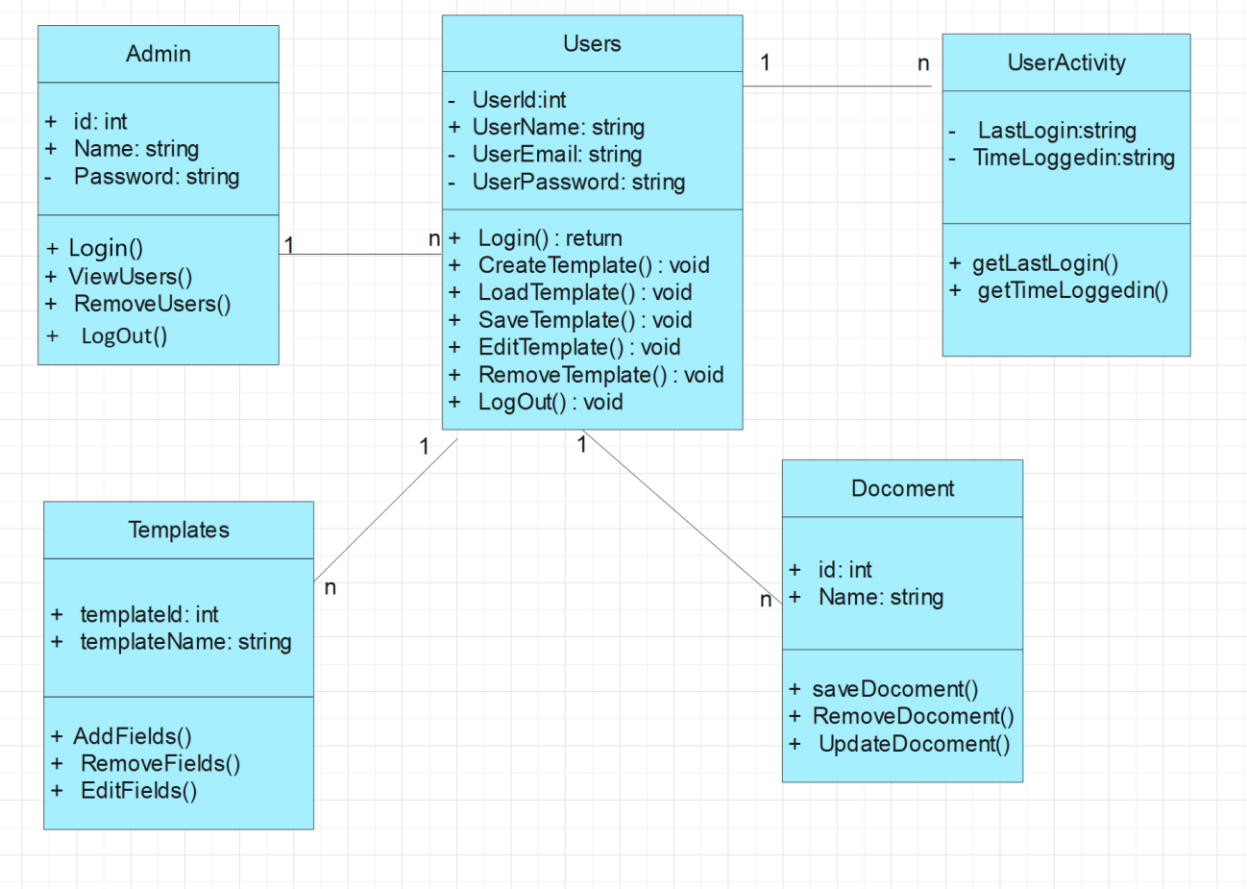


Figure 8: Class Diagram

4.5. Sequence / Collaboration Diagram

The following Sequence Diagram shows system's objects interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the system

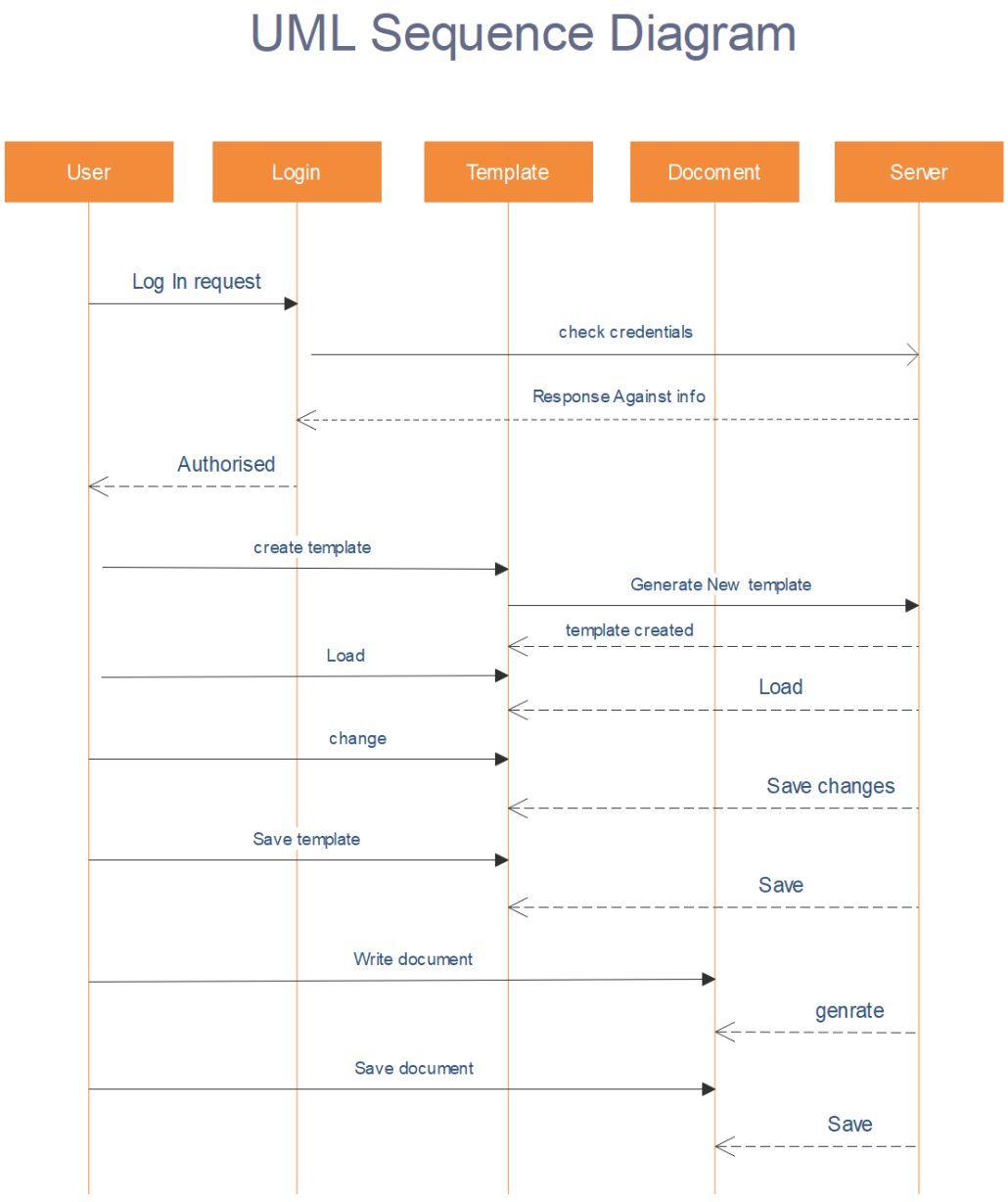


Figure 9:Sequence Diagram

4.6. Operation contracts

Use cases or system features are the main ways in the UP to describe system behavior, and are usually sufficient. Sometimes a more detailed or precise description of system behavior has value. Operation contracts use a pre- and post-condition form to describe detailed changes to objects in a domain model, as the result of a system operation.

4.7. Activity Diagram

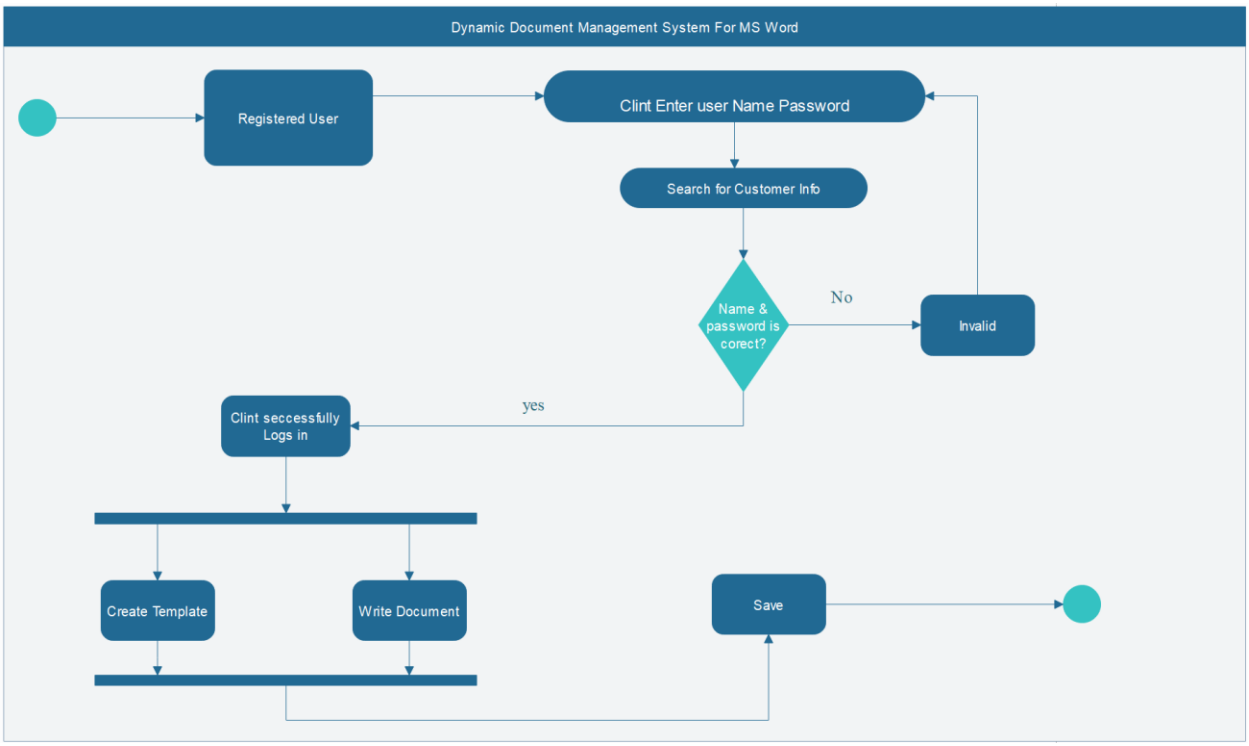


Figure 10:Activity diagram

4.8. State Transition Diagram

State-transition diagrams describe all of the states that an object can have, the events under which an object changes state (transitions), the conditions that must be fulfilled before the transition will occur (guards), and the activities undertaken during the life of an object (actions).

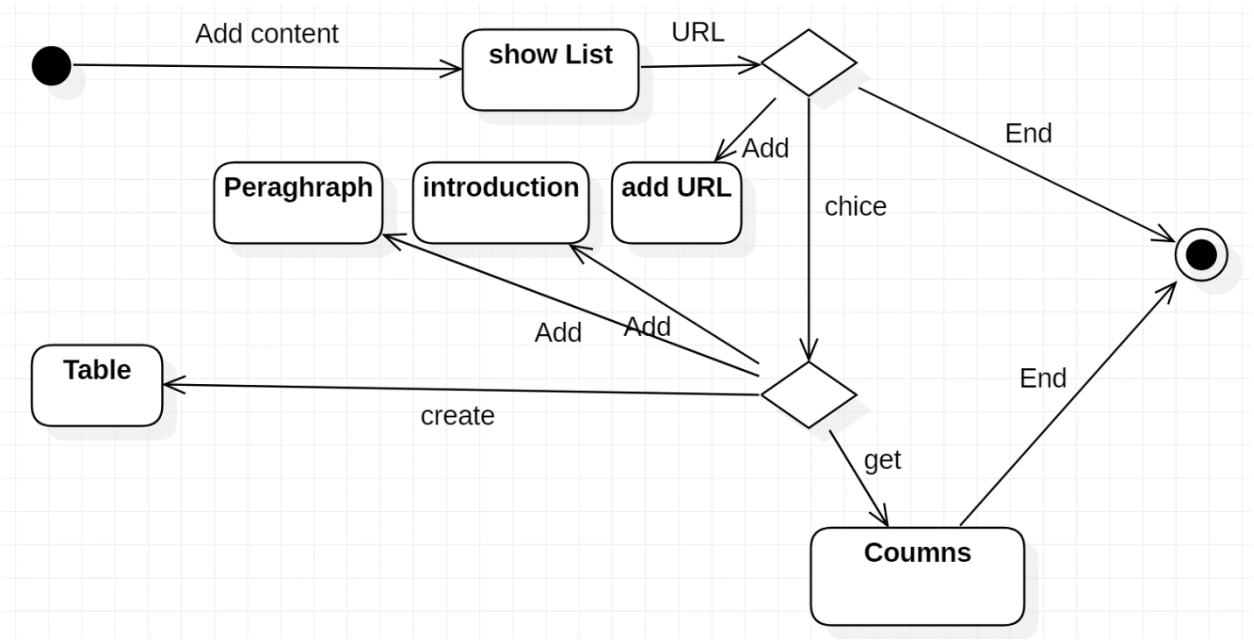


Figure 11:state transection diagram

4.9. Component Diagram

A component diagram, also known as a UML component diagram, describes the organization and wiring of the physical components in a system. Component diagrams are often drawn to help model implementation details and double-check that every aspect of the system's required functions is covered by planned development.

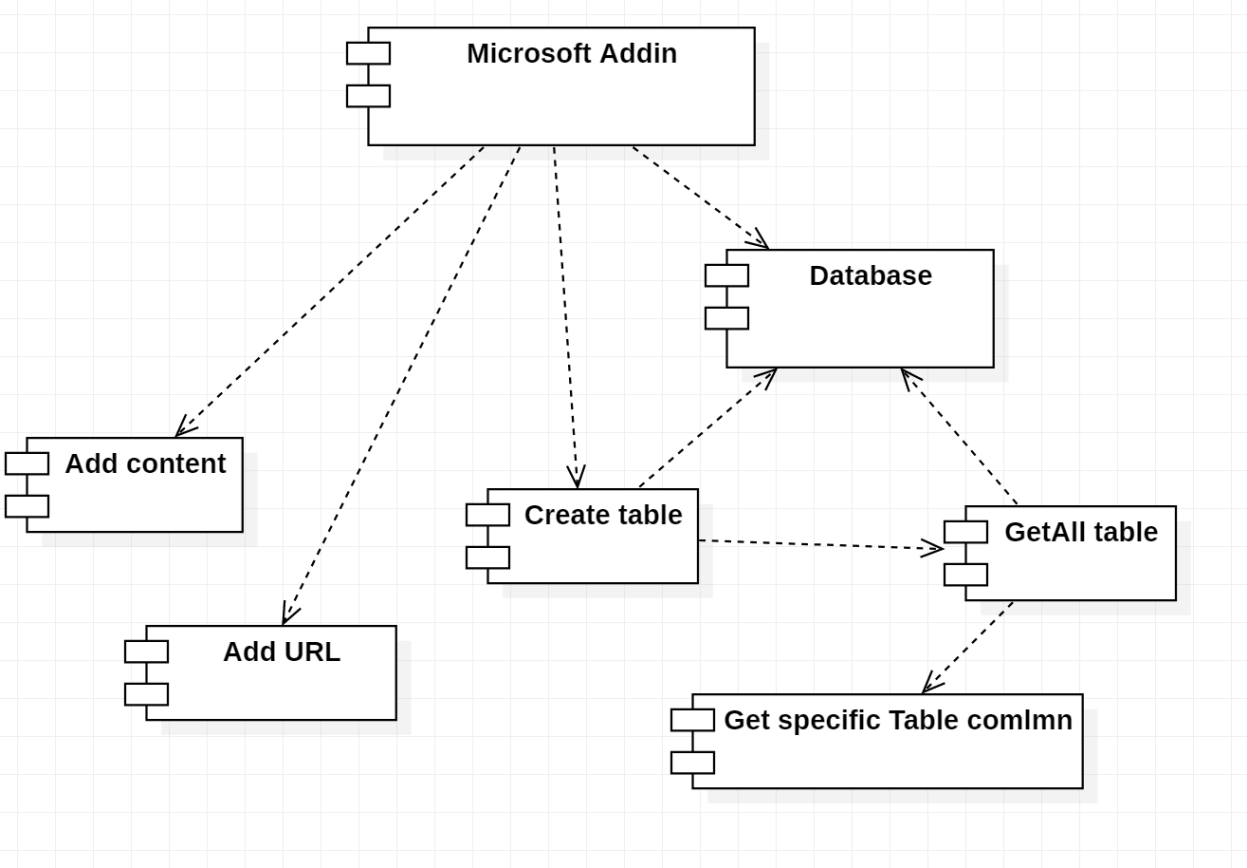


Figure 12:Component Diagram

4.10. Deployment Diagram

A deployment diagram is a diagram that shows the configuration of run time processing nodes and the components that live on them.

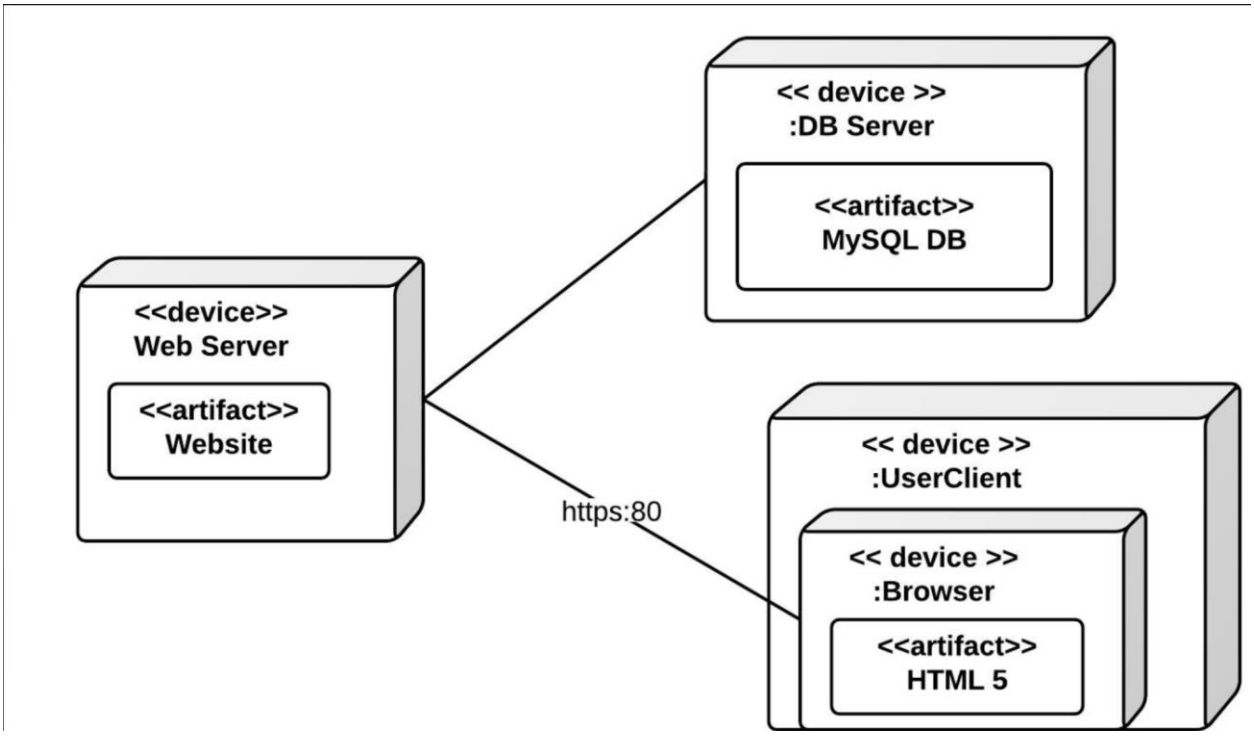


Figure 13:Deployment Diagram

4.11. Data Flow diagram

A data-flow-diagram (DFD) is a way of representing a flow of a data of a process or a system. The DFD also provides information about the outputs and inputs of each entity and the process itself.

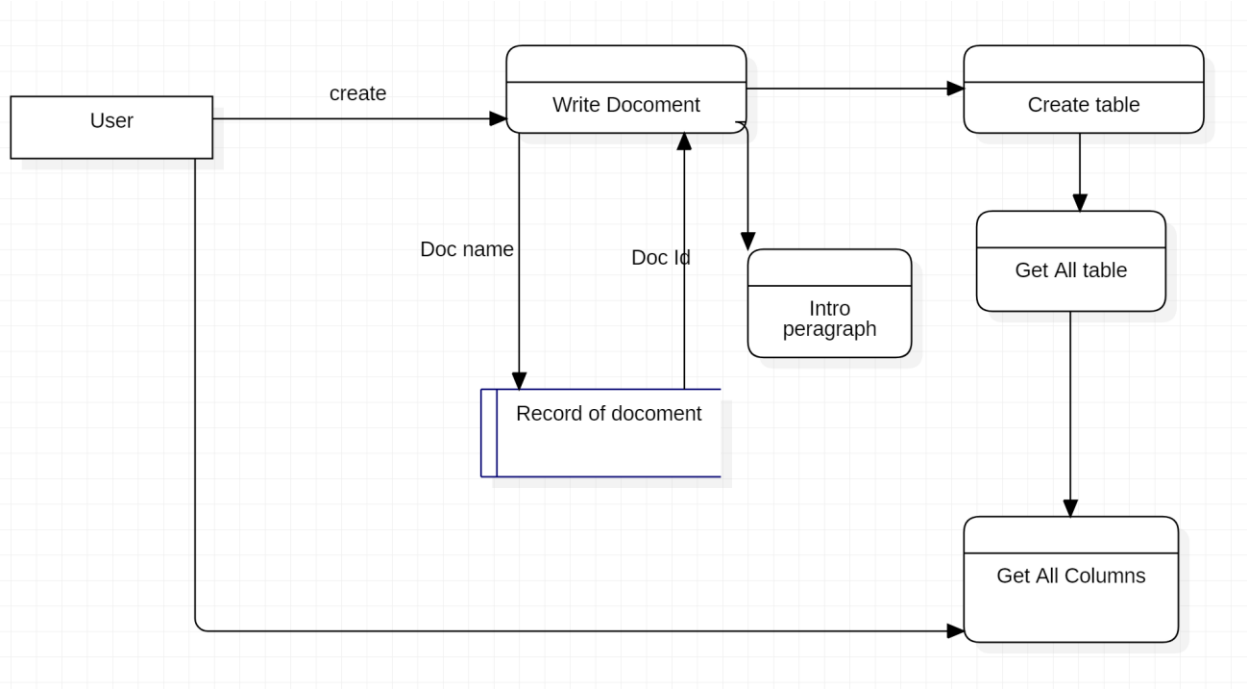


Figure 14:Data Flow Diagram

Chapter 5

Implementation

Chapter 5: Implementation

We implemented MS Word dynamic Document using ASP.NET MVC, C# and MSSQL with the help of JavaScript, jQuery and ajax requests on the frontend.

This Application creates a dynamic table and dynamic columns and dynamically create table In database. And load any Jason URL in template and insert and replace any paragraph.

This application gets the tables and table columns in task pane who help the write document in MS Word.

5.1. Important Flow Control/Pseudo codes

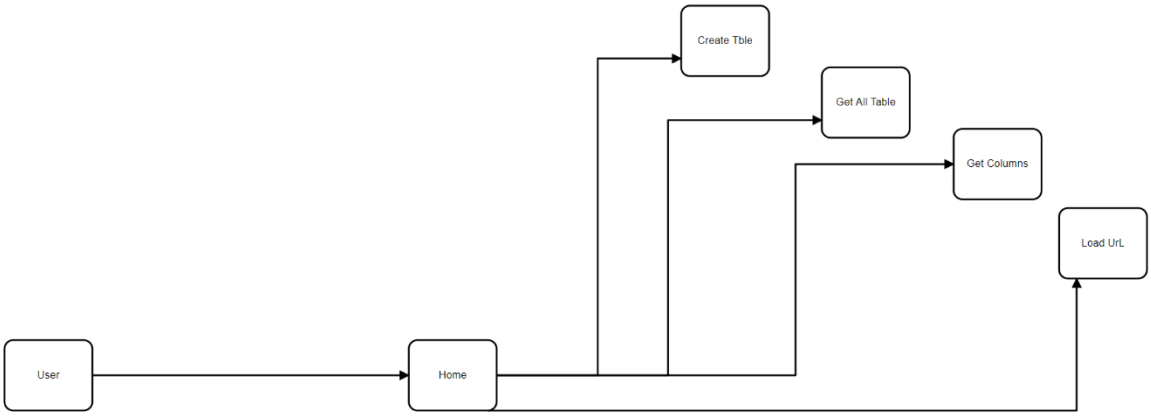


Figure 15:Important Flow

5.2. Components, Libraries, Web Services and stubs

- jQuery
- Bootstrap
- Modernizer

5.3. Deployment Environment

Deployment environment is Microsoft Word 2019.

5.4. Tools and Techniques

Tools:

- Visual studio
- Microsoft Office

Techniques:

- **C#**
- **Mvc.Net**
- **JavaScript**
- **Html**
- **CSS**

5.5. Best Practices / Coding Standards

Since using **C# language** for the coding of our Add-in in Visual Studio, in addition of Unity, following and some coding standards and practices we will follow while implementing our Add-in project:

Naming Conventions and Standards

Following Name Conventions and Standards are used while the coding phase of our game:

- Pascal casing: the first character of all words is upper case and the other characters are lower case.
- Camel casing: the first character of all words, except the first word, is upper case and other characters are lower case.

Good Programming Practices

We have avoid using too large files. If a file has 300-400 lines of code, we must consider refactoring the code into helper classes. We have avoid writing very long methods. A preferred method of our MS word Add-in has been of 1-60 lines of code. If we had a method more than 70 lines of code, we had must consider refactoring it into separate methods.

Comments

Not writing comments for every line of code and every variable declared and only wherever required. Good, readable code will require very few comments and that is what we will do.

Exception Handling

We will never do a "catch exception and do nothing." If we happen to hide an exception, we will never know if the exception happened or not. So, in the case of exceptions, we will give a friendly message to the user, but log the actual error with all possible details about the error, including the time it occurred, the method and class name, etc. We will only catch the specific exception, not generic exceptions.

5.6. Version Control

Microsoft office 2017 1nd about

Chapter 6

Testing and Evaluation

Chapter 6: Testing and Evaluation

This testing phase will use a number of testing techniques. The decision as to which technique(s) to use for any given unit of code will reside with the team leader responsible for signing-off on the Module.

6.1. Use Case Testing

- User can Load Jason file URL and Insert Paragraph and replace.
- User can create table and insert column in database.
- User can Get all table in Task pan.
- User cand Get specific Table columns.

6.1.1 Login

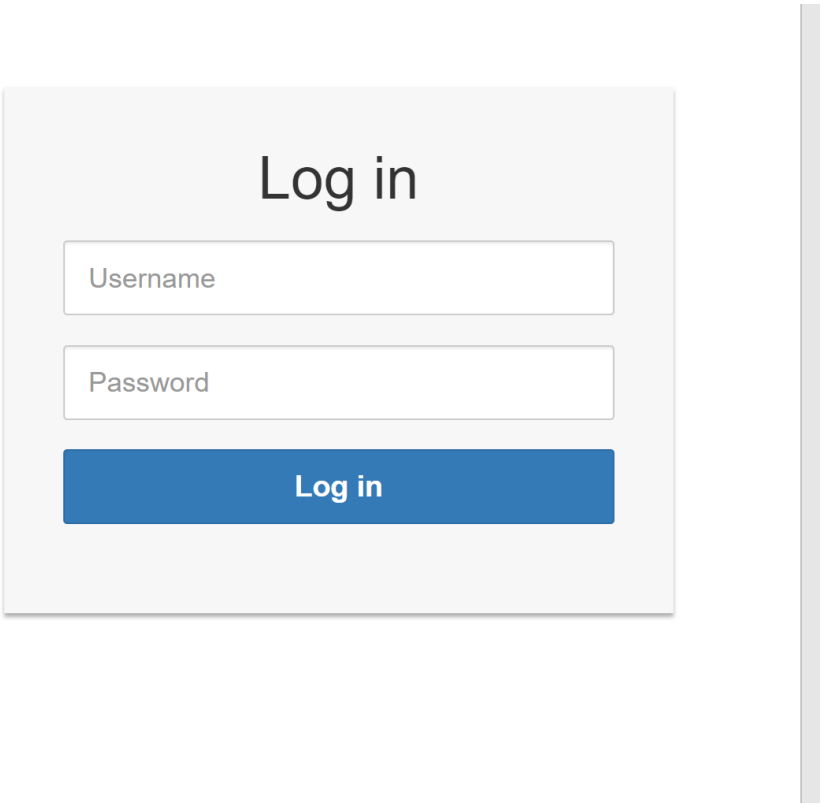


Figure 16:Login

6.1.2 Load New content and Replace

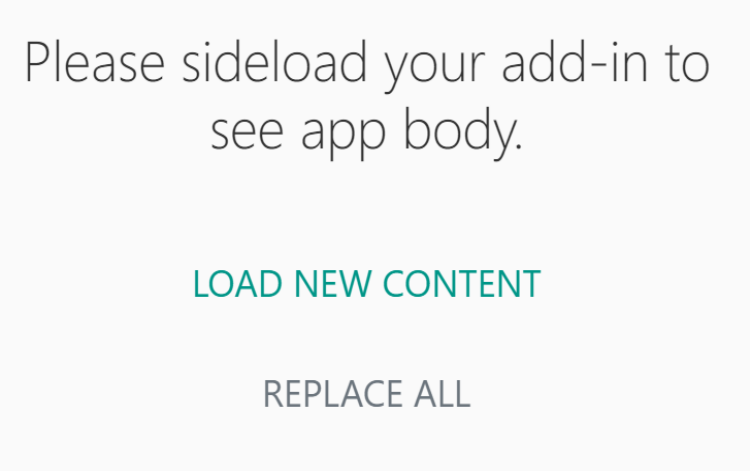


Figure 17:Load New content

6.1.3 Enter specify URL

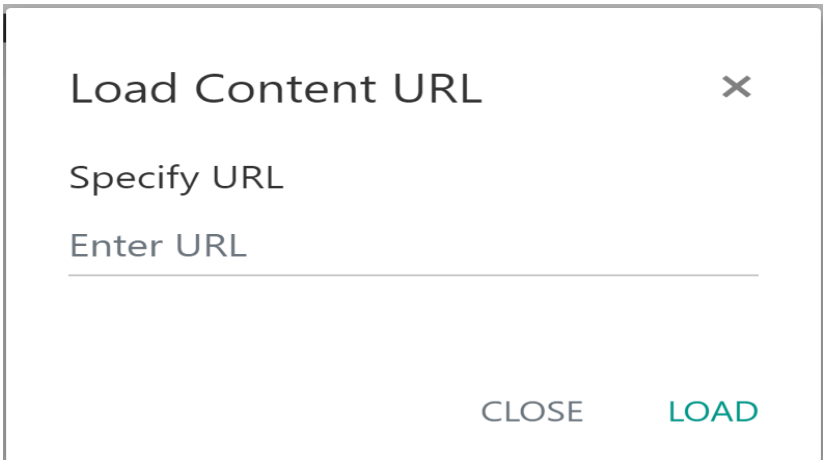


Figure 18:Enter Specify URL

6.1.4 Load Entire Paragraph

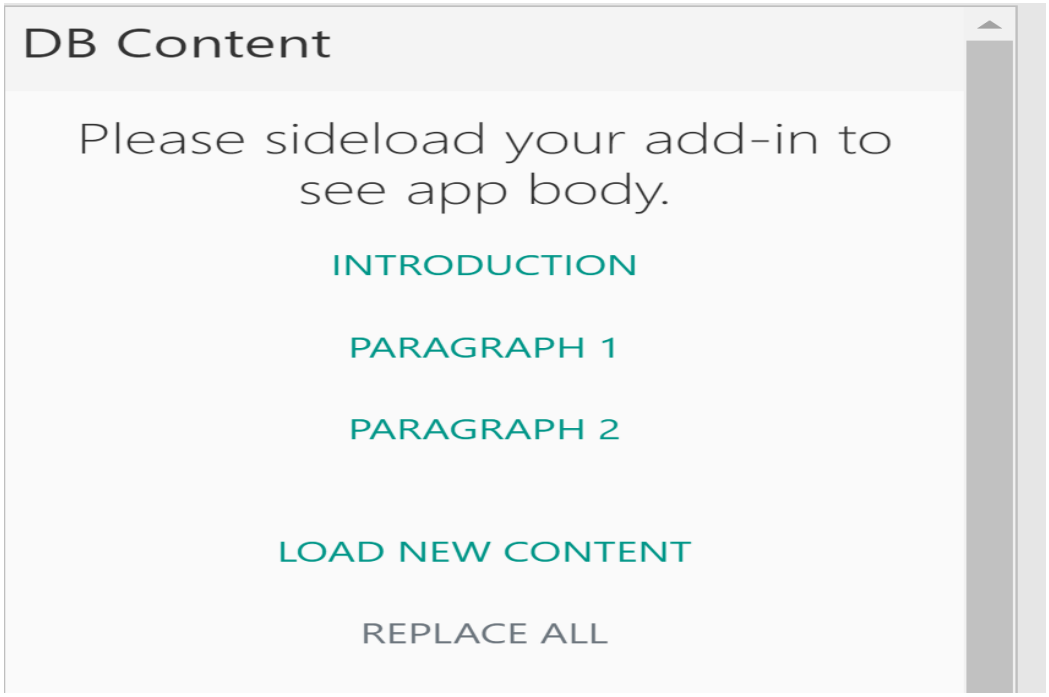


Figure 19:Load Entire paragraph

6.1.5 After Load Paragraph

This is a sample text inserted in the documentIntro new: Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Figure 20:Load paragraph

6.1.6 Create Table Dynamically

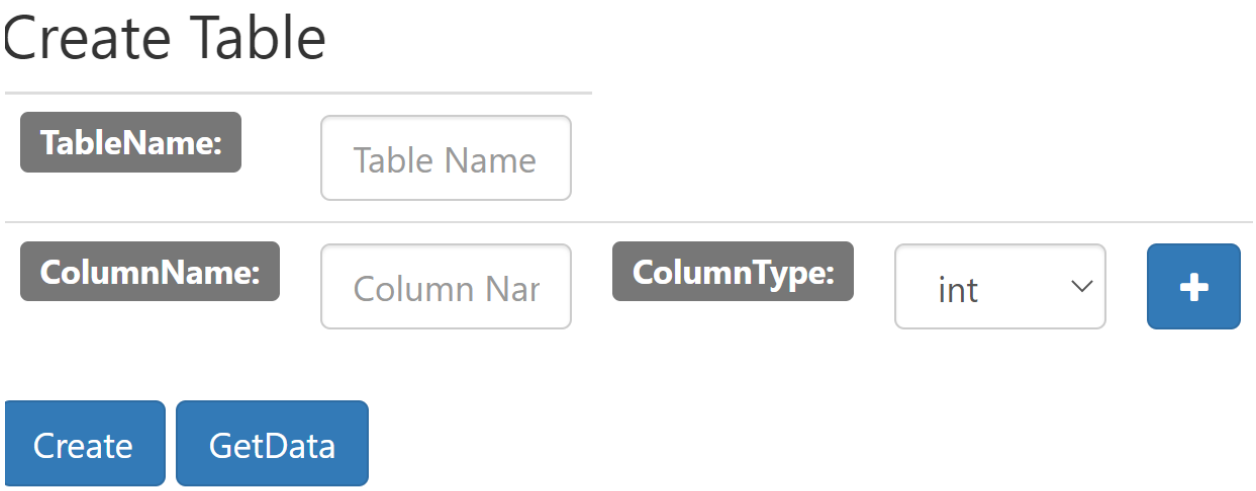


Figure 21:Create Table Dynamically

6.1.6 Add & Remove Fields Dynamically

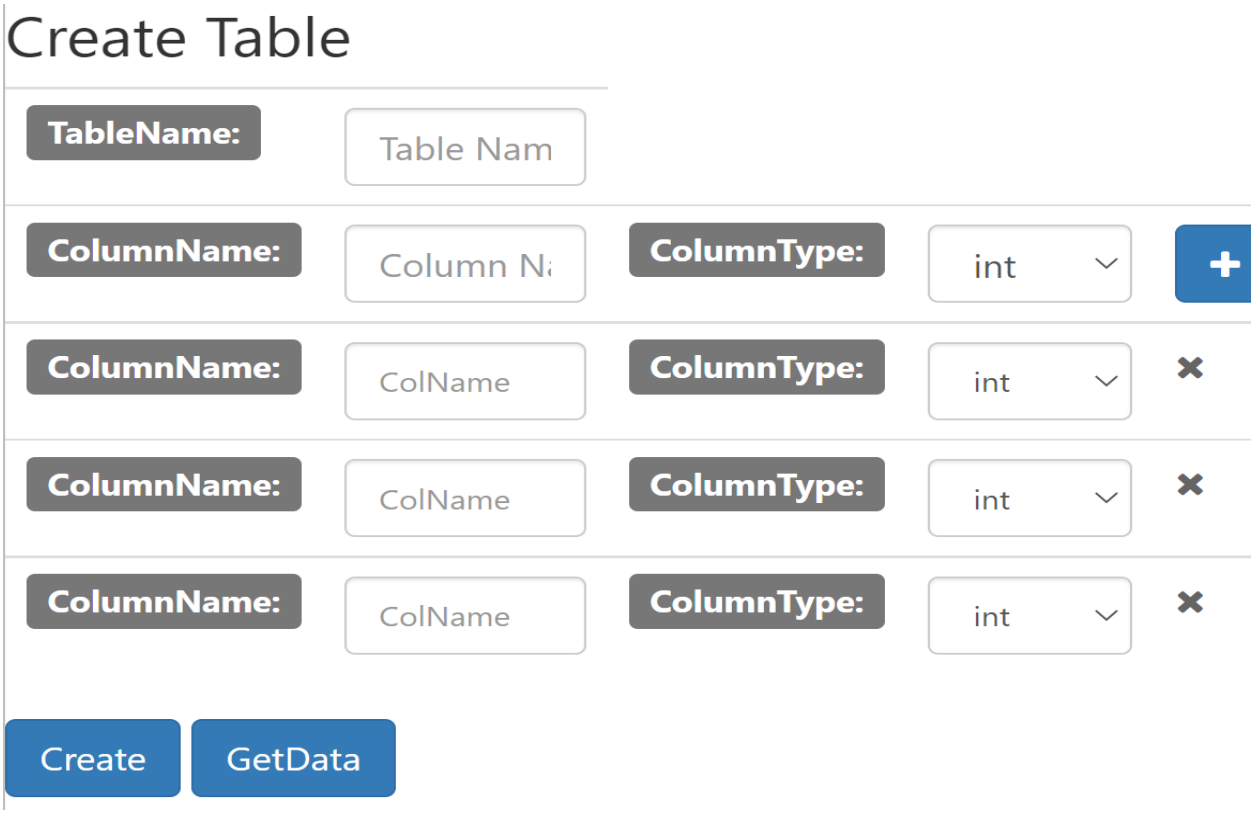


Figure 22:Add & Remove Fields

6.1.7 Dropdown of Datatype

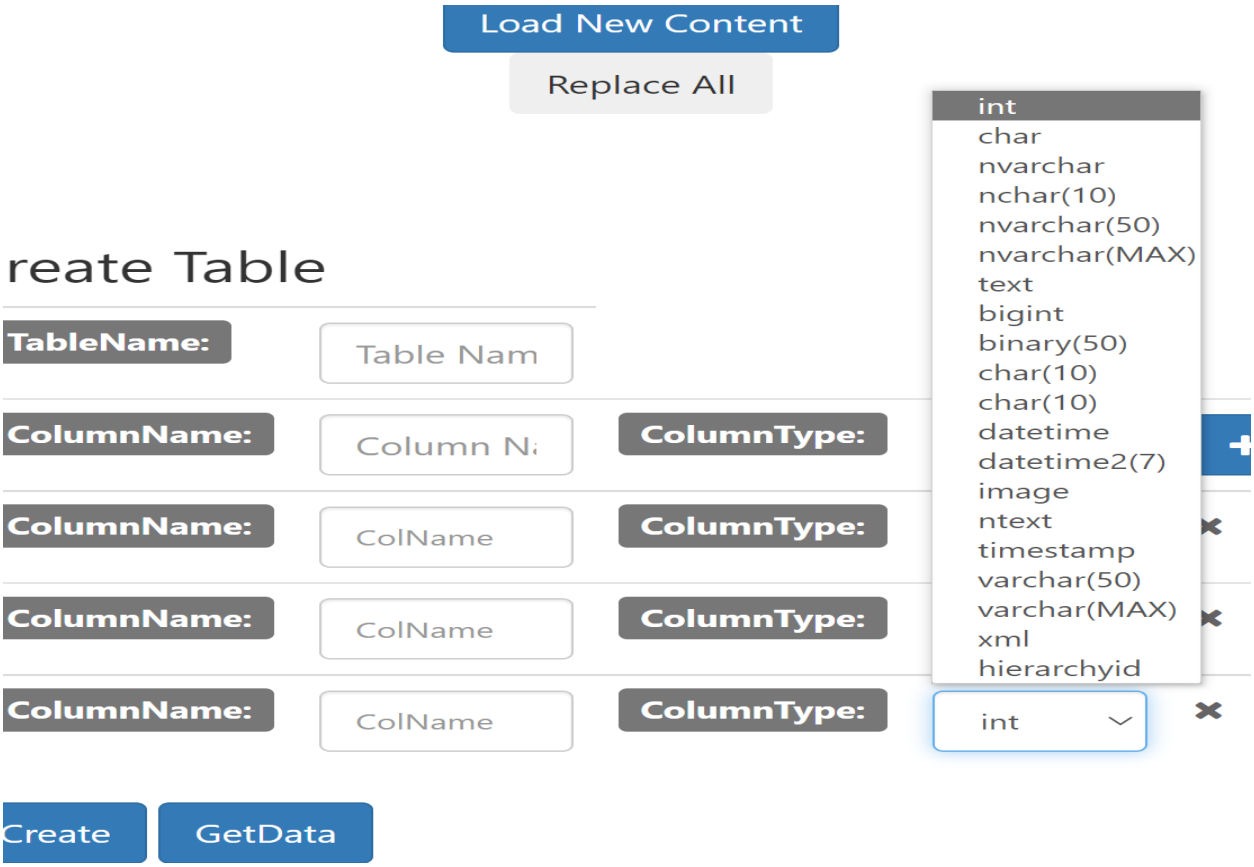
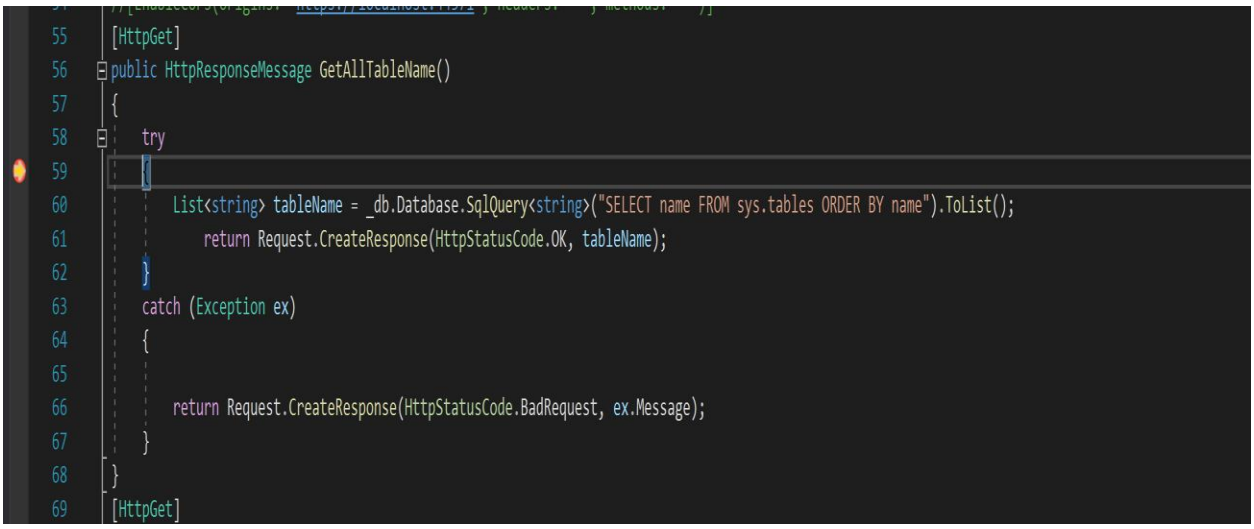


Figure 23:Data Type Dropdown

6.1.8 Get Table Name Dynamically



```
55  [HttpGet]
56  public HttpResponseMessage GetAllTableName()
57  {
58      try
59      {
60          List<string> tableName = _db.Database.SqlQuery<string>("SELECT name FROM sys.tables ORDER BY name").ToList();
61              return Request.CreateResponse(HttpStatusCode.OK, tableName);
62      }
63      catch (Exception ex)
64      {
65
66          return Request.CreateResponse(HttpStatusCode.BadRequest, ex.Message);
67      }
68  }
69  [HttpGet]
```

Figure 24:Get Table Name Dynamically

6.1.9 Dropdown Get Tables Dynamically

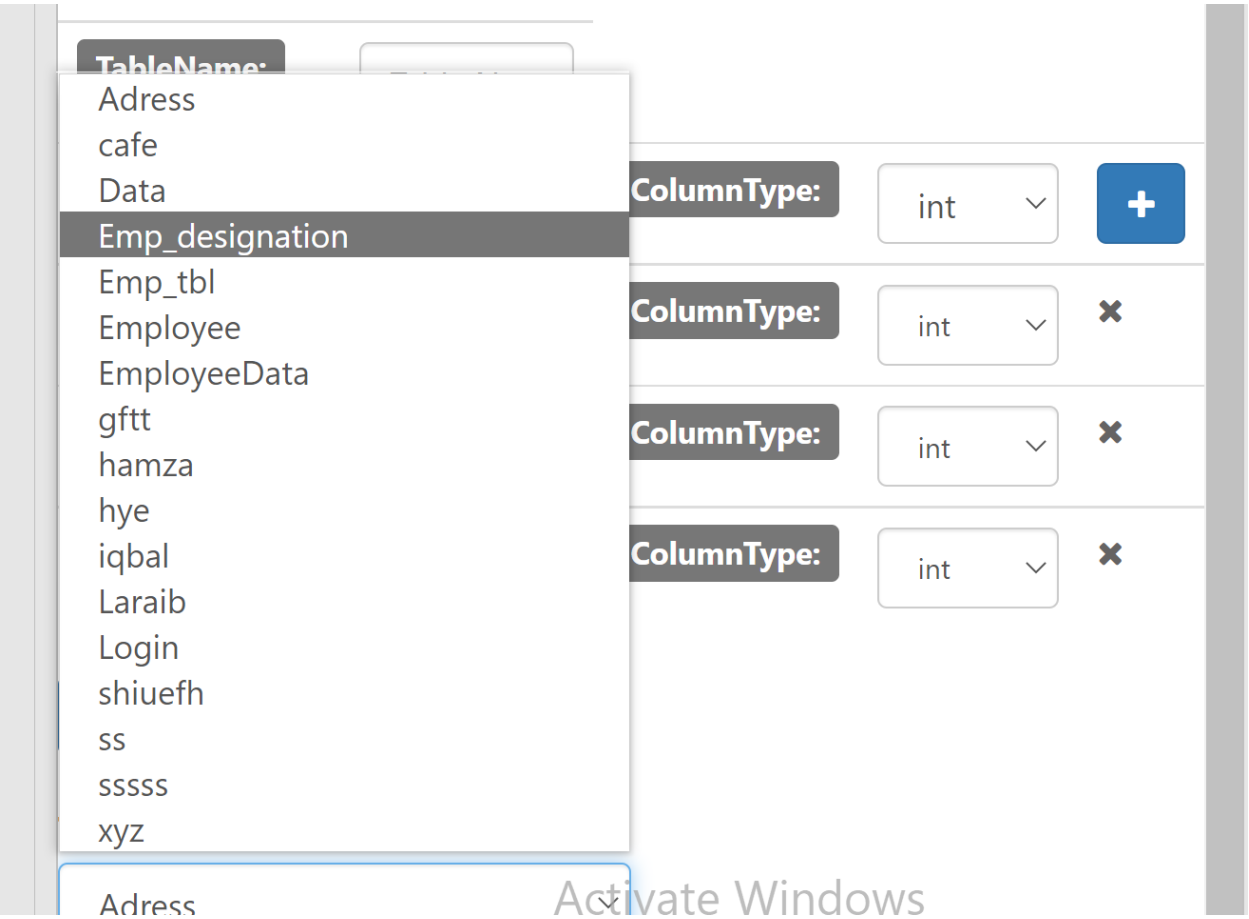


Figure 25:Drop Down Table Name Dynamically

6.1.10 Get Columns Dynamically

```
68  }
69  [HttpGet]
70  public HttpResponseMessage GetcolumnName(string tableName)
71  {  ≤ 557,132ms elapsed
72      try
73      {
74          List<string> columnName = _db.Database.SqlQuery<string>("select COLUMN_NAME FROM INFORMATION_SCHEMA.COLUMNS WHERE TABLE_NAME ='"+ tableName + "'").To
75          return Request.CreateResponse(HttpStatusCode.OK, columnName);
76      }
77      catch (Exception ex)
78      {
79
80          return Request.CreateResponse(HttpStatusCode.BadRequest, ex.Message);
81      }
82  }
83
```

Figure 26:Get Columns Dynamically

6.1.11 Dropdown Get Columns Dynamically

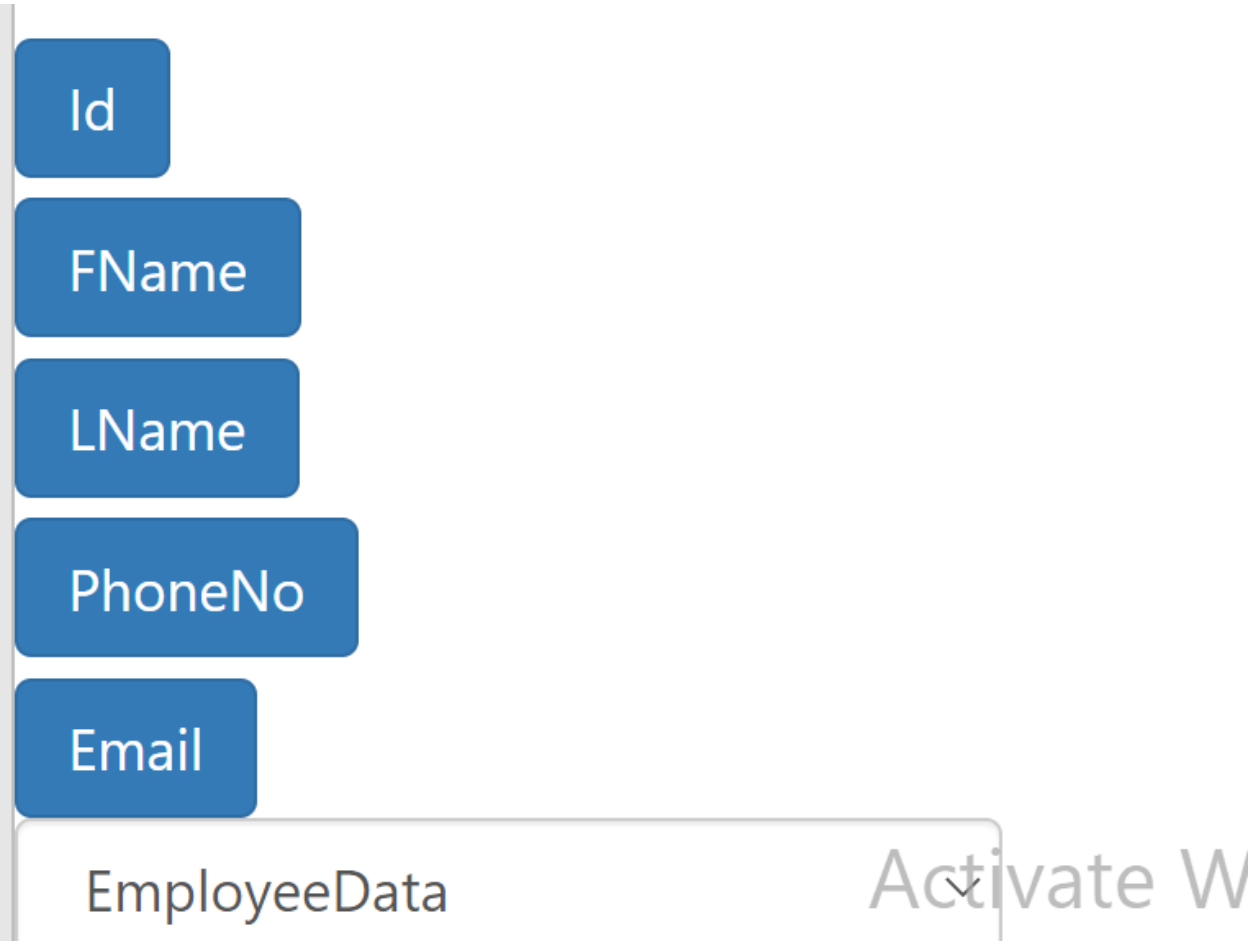


Figure 27:Get Columns Dynamically

6.1.12 Click on Any columns add placeholders

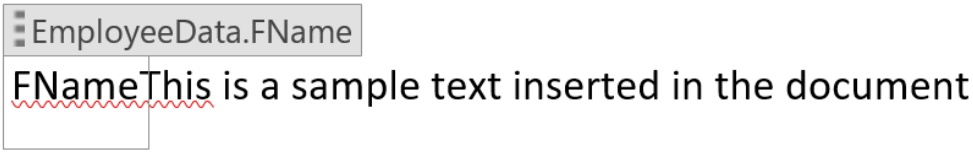


Figure 28:Click om any columns add placeholders

6.2. Equivalence partitioning

Equivalence Partitioning also called as equivalence class partitioning. It is abbreviated as ECP. It is a software testing technique that divides the input test data of the application under test into each partition at least once of equivalent data from which test cases can be derived. An advantage of this approach is it reduces the time required for performing testing of a software due to a smaller number of test cases.

6.3. Boundary value analysis

Boundary value analysis is a software testing technique in which tests are designed to include representatives of boundary values in a range. The idea comes from the boundary. Given that we have a set of test vectors to test the system, a topology can be defined on that set.

6.4. Data flow testing

Data Flow Testing is a specific strategy of software testing that focuses on data variables and their values. It makes use of the control flow graph. When it comes to categorization Data flow testing will can be considered as a type of white box testing and structural types of testing. It keeps a check at the data receiving points by the variables and its usage points. It is done to cover the path testing and branch testing gap.

6.5. Unit testing

Unit testing, a testing technique using which individual modules are tested to determine if there are any issues by the developer himself. It is concerned with functional correctness of the standalone modules. The main aim is to isolate each unit of the system to identify, analyze and fix the defects.

- Reduces Defects in the Newly developed features or reduces bugs when changing the existing functionality.
- Reduces Cost of Testing as defects are captured in very early phase
- Improves design and allows better refactoring of code.
- Unit Tests, when integrated with build gives the quality of the build as well.

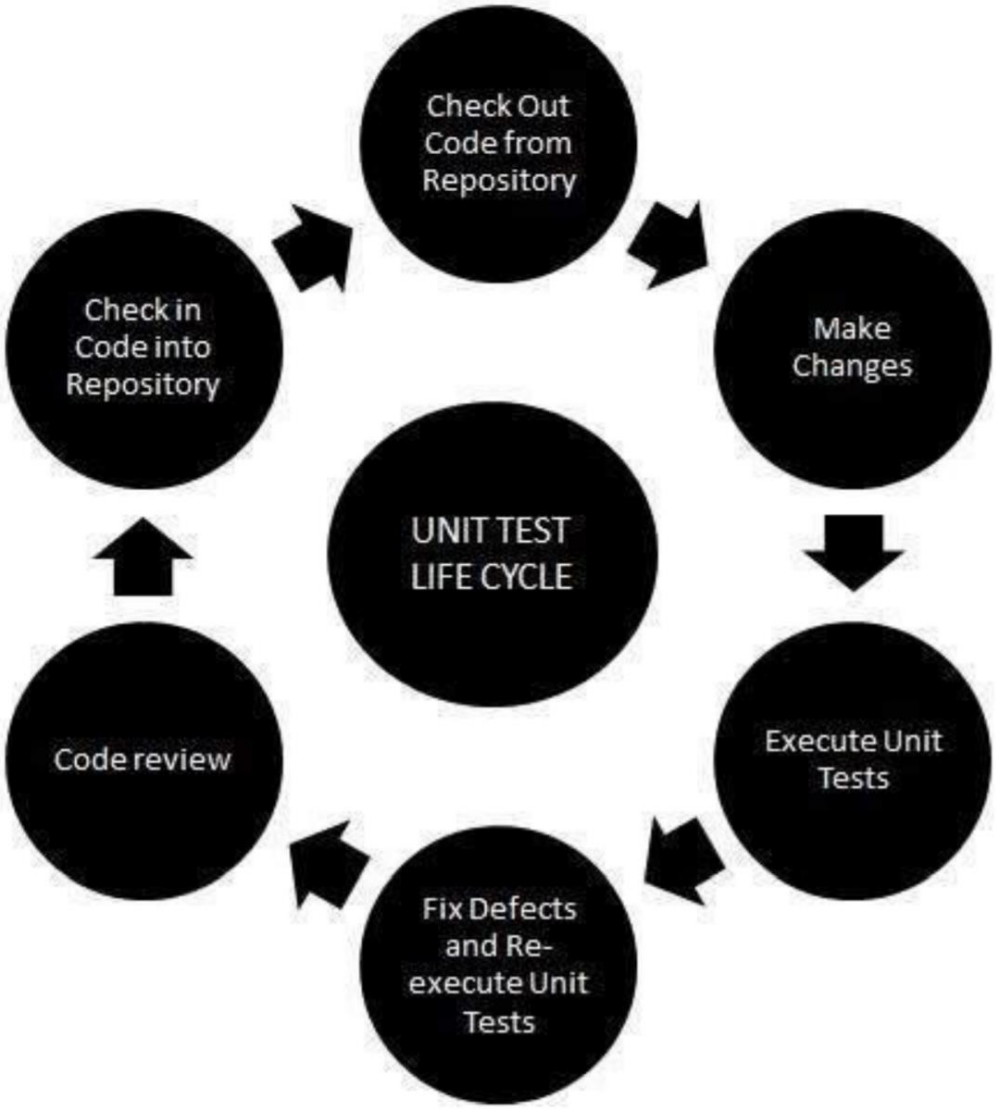


Figure 29:Unit Testing

6.6. Integration testing

Integration testing is the phase in software testing in which individual software modules are combined and tested as a group. Integration testing is conducted to evaluate the compliance of a system or component with specified functional requirements. It occurs after unit testing and before validation testing.

6.7. Performance testing

Performance testing, a non-functional testing technique performed to determine the system parameters in terms of responsiveness and stability under various workload. Performance testing measures the quality attributes of the system, such as scalability, reliability and resource usage.

- **Load testing -**It is the simplest form of testing conducted to understand the behavior of the system under a specific load. Load testing will result in measuring important business critical transactions and load on the database, application server, etc., are also monitored.
- **Stress testing -** It is performed to find the upper limit capacity of the system and also to determine how the system performs if the current load goes well above the expected maximum. **Soak testing -** Soak Testing also known as endurance testing, is performed to determine the system parameters under continuous expected load. During soak tests the parameters such as memory utilization is monitored to detect memory leaks or other performance issues. The main aim is to discover the system's performance under sustained use.
- **Spike testing** - Spike testing is performed by increasing the number of users suddenly by a very large amount and measuring the performance of the system. The main aim is to determine whether the system will be able to sustain the workload.

6.8. Stress Testing

Stress testing a Non-Functional testing technique that is performed as part of performance testing. During stress testing, the system is monitored after subjecting the system to

overload to ensure that the system can sustain the stress. The recovery of the system from such phase (after stress) is very critical as it is highly likely to happen in production environment

.

- It allows the test team to monitor system performance during failures.
- To verify if the system has saved the data before crashing or NOT.
- To verify if the system prints meaning error messages while crashing or did it print some random exceptions.
- To verify if unexpected failures do not cause security issues.

Chapter 7

Summary, Conclusion and Future Enhancements

Chapter 7: Summary, Conclusion & Future Enhancements

7.1. Project Summary

This study will be based on solving the above-described problem using MS Office Add-ins. We are actually building an add-in for Microsoft Word whose main purpose is to manage dynamic templates. So that a user can easily update it through the database. This add-in provides a way to upload data to data base and then this data can be bided to word document using placeholders. These placeholders then can be added to a document and can have variable values to provide a dynamic feature inside MS Word.

7.2. Achievements and Improvements

- Creation of Dynamic tables.
- Creation of Dynamic Fields.
- Posting data in database.
- Retrieve dynamic data from Database.
- Ajax requests to get and post data from .net MVC.

7.3. Critical Review

Although everything is good in this add-in, like crate tables, Get URL and table data. There are some other functions which can work correctly according to the standards like login functionality and becoming a member for this Add-in so that you can save your Document and see them next time you visit this add-in. There are some improvements which can be done in future like All functionalities for MS word.

7.4. Lessons Learnt

- Decision Making
- Program Management
- Project Governance
- Project Planning
- Roles & Responsibilities.
- How to Load Jason file URL and Insert Paragraph and replace.
- How to create table and insert column in database.
- How to Get all table in Task pan.
- How to cand Get specific Table columns.

7.5. Future Enhancements/Recommendations

Although this website is properly in working state according to the functions installed in it. Like Get URL, Create Tables and Get Tables. Also becoming a member of this site to save logs. But there are some enhancements which can be done in future like

- Multiple user work on same document
- And user login city and which country
- How many documents create specific user?

Appendices

Appendix A: Information / Promotional Material

This study will be based on solving the above-described problem using MS Office Add-ins. We are actually building an add-in for Microsoft Word whose main purpose is to manage dynamic templates. So that a user can easily update it through the database. This add-in provides a way to upload data to data base and then this data can be bided to word document using placeholders. These placeholders then can be added to a document and can have variable values to provide a dynamic feature inside MS Word.

A.1. Broacher

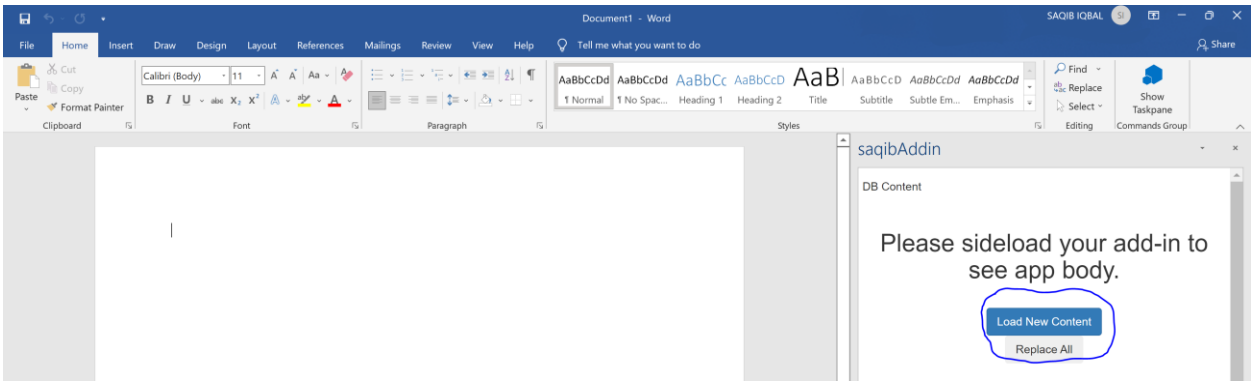


Figure 30: Broacher

A.2. Flyer

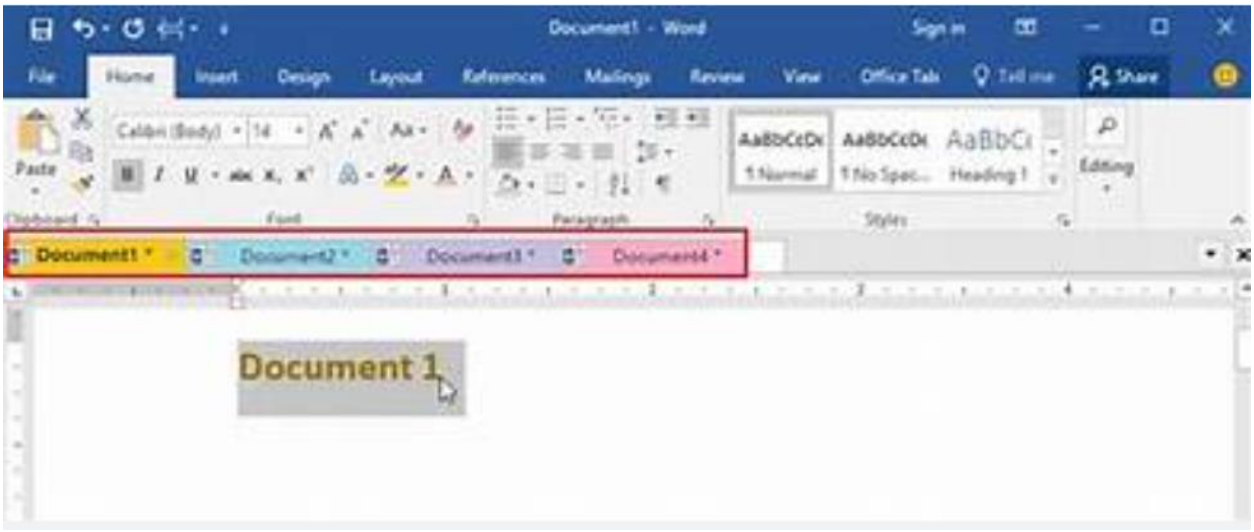


Figure 31:Flyer

A.3. Banner

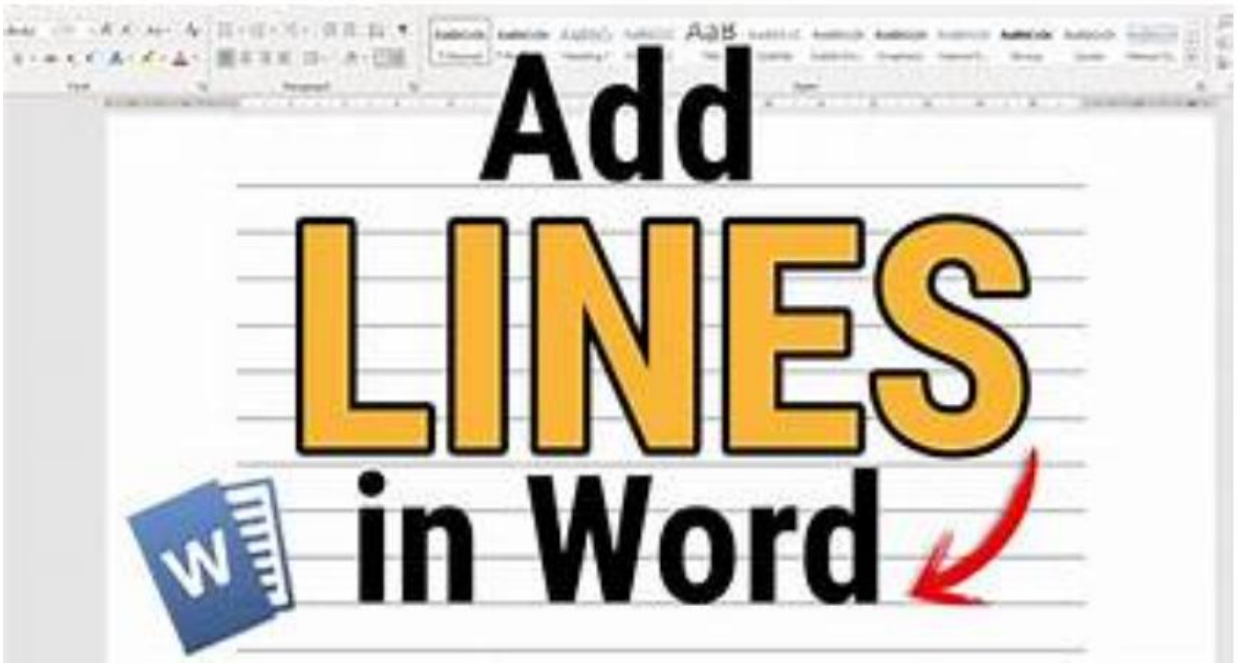


Figure 32:Benner

Reference and Bibliography

Reference and Bibliography

[1] Manually, T. (2019). *C# Coding Standards and Best Programming Practices - Code Project*. [online] Codeproject.com. Available at: https://www.codeproject.com/Articles/8971/C-Coding-Standards-and-Best-Programming-Practices [Accessed 14 Aug. 2019].

[2] Unity. (2020). *Programming in Unity: For programmers new to Unity - Unity*. [online] Available at: https://unity3d.com/programming-in-unity [Accessed 14 Aug. 2020].

[3] Msdn.microsoft.com. (2021). *Unity - Developing Your First Game with Unity and C#*. [online] Available at: https://msdn.microsoft.com/en-us/magazine/dn759441.aspx [Accessed 14 Aug. 2021].

[4] Unity. (2020). *2D Game Creation - Unity*. [online] Available at: https://unity3d.com/learn/tutorials/s/2d-game-creation [Accessed 14 Aug. 2020]

[5] Unity Forum. (2020). *2d game with 3d models*. [online] Available at: https://forum.unity.com/threads/2d-game-with-3d-models.388119/ [Accessed 14 Aug. 2018].

[6] Widman, T. (2020). *Global Games Market Revenues 2020 | Per Region & Segment | Newzoo*. [online] Newzoo. Available at: https://newzoo.com/insights/articles/global-games-market-reaches-137-9-billion-in-2018-mobile-games-take-half/ [Accessed 14 Aug. 2021].

Index

Index