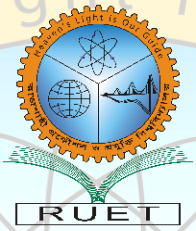
Heaven's light is our guide.

Rajshahi University of Engineering and Technology, Rajshahi.



Department of Electrical and Computer Engineering.

Course Code:

ECE 3118

Course Title:

Software Engineering & Information System Design Sessional

Submitted To:	Submitted By:
Oishi Jyoti	Md. Towhidul Alam
Assistant Professor	Roll: 2010016
Department of ECE	Department of Electrical & Computer Engineering
RUET	

Experiment 1

Experiment Name: Basic Documentation Using Markdown Language

Objectives

The primary objectives of this experiment are:

- To learn the fundamentals of **Markdown**, a lightweight markup language used for formatting text.
- To understand how Markdown syntax is applied to structure documents efficiently.
- To explore the utility of Markdown for creating documentation in a clear, readable, and organized format.
- To practice writing basic documentation for a project using various Markdown elements, such as headings, lists, code blocks, and links.

Tools

- A text editor (e.g., Visual Studio Code, Sublime Text, or any plain text editor).
- Access to a platform that renders Markdown (e.g., GitHub, GitLab, Markdown Preview in VS Code).

Methods

1. **Setup**: Open a new file in a text editor and save it with a .md extension to ensure it's recognized as a Markdown file.
2. **Documentation Creation**:
    - Start with headers to structure the document.
    - Create an ordered list for sections like objectives, methods, and results.
    - Insert code blocks and inline code where necessary to demonstrate examples.
    - Use bold and italic text to highlight key points and terms.
    - Add links to relevant resources or sections within the document for easy navigation.
3. **Rendering**: View the rendered Markdown in a platform or editor that supports live preview to assess the structure and formatting.

5. Source Code and Output

Source Code: Markdown Example

````
# Project Title
A brief description of the project or experiment.

## Table of Contents
1. [Introduction](#introduction)
2. [Installation Instructions](#installation-instructions)
3. [Usage](#usage)
4. [Features](#features)
5. [License](#license)

## Introduction
This project aims to explore basic Markdown documentation. It
introduces the syntax for writing readable and well-structured
documentation files using plain text.

## Installation Instructions
To get started, clone the repository and install the necessary
dependencies:
1. Clone the repository using `git clone <repo_url>`.
2. Install dependencies using `npm install`.
3. Run the project using `npm start`.

## Usage
To use the project:
```bash
npm start
```
<details>
  <summary>How do I install the project?</summary>

  - Clone the repository:
    ```bash
    git clone <repository_url>
    ```
  - Navigate into the project directory:
    ```bash
    cd project-directory
    ```
  - Install dependencies:
    ```bash
````
````

````
    npm install
    ```


</details>

<details>
  <summary>How do I contribute to the project?</summary>

  1. Fork the repository.
  2. Create a new branch: `git checkout -b my-feature-branch`.
  3. Commit your changes: `git commit -m "Add new feature"`.
  4. Push to the branch: `git push origin my-feature-branch`.
  5. Create a pull request.

</details>


[More Practices](#next-section)
````

Preview:

Project Title

A brief description of the project or experiment.

Table of Contents

Introduction

This project aims to explore basic Markdown documentation. It introduces the syntax for writing readable and well-structured documentation files using plain text.

Installation Instructions

To get started, clone the repository and install the necessary dependencies:

1. Clone the repository using `git clone <repo_url>`.
2. Install dependencies using `npm install`.
3. Run the project using `npm start`.

Usage

To use the project:

```
npm start
```

▶ How do I install the project?
▶ How do I contribute to the project?

More Practices
```

More:

```
# Next Section
## Advanced Table Formatting
Tables are useful for presenting structured data in Markdown. This
example includes formatting for alignment, headers, and cell styling.
| Feature           | Description                                 |
Supported Platforms         |
|-------------------|---------------------------------------------|-----
------------------------|
| **Simple Syntax** | Uses minimal, easy-to-read symbols          |
All                         |
| **Portability**   | Works across different platforms            |
Windows, Mac, Linux, Web    |
| **Readability**   | Can be read in plain text or rendered       |
All


## Code Blocks with Syntax Highlighting

```javascript
// JavaScript example
function greet(name) {
  console.log(`Hello, ${name}!`);
}
greet('Alice');

```


Project Tasks
- [x] Set up project repository
- [x] Initialize README file
- [ ] Design initial layout
- [ ] Implement core functionality
  - [ ] Add user login
  - [ ] Create user dashboard
- [ ] Conduct testing
- [ ] Prepare project documentation

Image Embedding with Custom Size

![Small logo](https://example.com/logo.png)
```

```
<img src="https://example.com/logo.png" alt="Custom logo"
width="200">


## Adding Links with Tooltip Descriptions

[OpenAI](https://openai.com "Visit the OpenAI homepage") is an AI
research and deployment company.



This is an example of a statement with a footnote.[^1]
```
[^1]: This is the footnote text providing more context or citation.

Preview:

More Practices

Next Section

Advanced Table Formatting

> Tables are useful for presenting structured data in Markdown. This example includes formatting for alignment, headers, and cell styling.

markdown Copy code

Feature	Description	Supported Platforms
Simple Syntax	Uses minimal, easy-to-read symbols	All
Portability	Works across different platforms	Windows, Mac, Linux, Web
Readability	Can be read in plain text or rendered	All

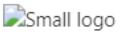
Code Blocks with Syntax Highlighting

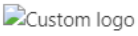
```
// JavaScript example
function greet(name) {
  console.log(`Hello, ${name}!`);
}
greet('Alice');
```

Project Tasks

- [x] Set up project repository
- [x] Initialize README file
- [ ] Design initial layout
- [ ] Implement core functionality
    - [ ] Add user login
    - [ ] Create user dashboard
- [ ] Conduct testing
- [ ] Prepare project documentation

Image Embedding with Custom Size





Adding Links with Tooltip Descriptions

OpenAI is an AI research and deployment company.

This is an example of a statement with a footnote.[^1]

```
[^1]: This is the footnote text providing more context or citation.
```

Features

- **Simple syntax**: Markdown is easy to learn and use.
- **Portable**: Works across all platforms and can be rendered by various tools.
- **Readable**: Can be viewed in plain text and rendered for visual clarity.

6. Results

The experiment was successful in creating a structured document using Markdown. All essential elements (such as headers, lists, code blocks, and links) were successfully incorporated, demonstrating how Markdown syntax can be used to format a document efficiently. The rendered output is clear, organized, and visually accessible.

7. Discussion

Markdown is an essential tool for developers and technical writers, as it allows for creating clean, readable, and well-structured documentation without requiring complex software or

formatting tools. Its simplicity and versatility make it ideal for use in version control systems, such as GitHub, where README files are often written in Markdown.

The features demonstrated in this experiment highlight the key advantages of Markdown:

- **Simplicity**: Markdown uses plain text and minimal syntax to produce a well-formatted document, making it easy to learn and implement.
- **Portability**: Markdown files can be opened and rendered on any platform that supports the format, making them ideal for cross-platform documentation.
- **Flexibility**: Markdown supports various elements (such as headers, lists, images, and tables) while maintaining its readability in plain text form.

However, despite its many advantages, Markdown has some limitations. It does not provide advanced formatting capabilities found in other types of markup languages, such as HTML or LaTeX. Additionally, rendering may vary slightly across different platforms, making it important to test the document in different environments to ensure consistent output.

8. Conclusion

The experiment successfully demonstrated the use of Markdown for creating structured and readable documentation. Markdown's simplicity and portability make it an excellent choice for technical documentation, and it is widely adopted in software development, especially for open-source projects. As a next step, more complex elements of Markdown, such as tables and embedded media, can be explored to enhance documentation further.

9. References

- Markdown Guide. (2024). *Markdown Basics*. Retrieved from https://www.markdownguide.org
- GitHub. (2024). *Mastering Markdown*. Retrieved from https://guides.github.com/features/mastering-markdown/