# SCRIPTING PROJECT

Contents

**SECTION 1 ---------------------------------------------------------------------------------------------------------------- pg 1**

**SECTION 2 -----------------------------------------------------------------------------------------------------------------pg 1**

**SECTION 3 --------------------------------------------------------------------------------------------------------------- pg 1**

**SECTION 4 ---------------------------------------------------------------------------------------------------------------- pg 2**

**SECTION 5 --------------------------------------------------------------------------------------------------------------- pg 2**

**SECTION 6 --------------------------------------------------------------------------------------------------------------- pg 2**

**INTRODUCTION:**

This report document regards the project management and development of a pilot application for RMIT university. Essentially, our client requested for a utility program that accurately defines the number of networks needed to connect a given number of additional computers to an existing IT system. This program aims to output the appropriate IP addressing scheme based on the number of subnets entered by the user. The intention behind this is to assist in network design and ensuring IP addresses are assigned to each device correctly – This design process requires an understanding of how subnetting works that the output of this application aims to provide to the user. Within this report is an outline of each task that is to be undertaken during the development process and the applicable documentation for each part. To produce this application, our team first aim to develop a Software Development Life Cycle to compartmentalise the development process and give clear timeframe guidelines to work with. We then aim to compile all client requirements into a document to ensure we are implementing these throughout the entire process. An algorithm plan is then necessary to design the script and the logical order of each action or function within it. Succeeding this will be the development of the script for the application – the algorithm plan will be used as guidance through this. After the initial pilot is developed, our team will undertake testing to investigate any necessary modifications needed to meet all client requirements. These modifications will be implemented and documented afterwards. Finally, we aim to develop a user document to assist RMIT staff in operating the application.

**OUTCOME OF TASK 1:**

Document the SDLC process you will follow for creating this application.

1. Task 2 (User Requirements/System Requirements) – 22/09/2022
2. Task 3 (Analysis and Design) – 23/09/2022 – 26/09/2022
3. Task 4 (Pilot Building) – 26/09/2022 – 28/09/2022
4. Task 5 & 6 (Testing & User Guide Development for implementation and testing)– 29/09/0222

**OUTCOME OF TASK 2:**

**• What is the scope of the project?**

The deliverable of this project is to develop a tool that assists Network designers in what addressing scheme is applicable to the number of subnets required for the network. This project is to be completed by the 30/09/2022.

* **Identify 2 design specifications that you need to adhere to?**

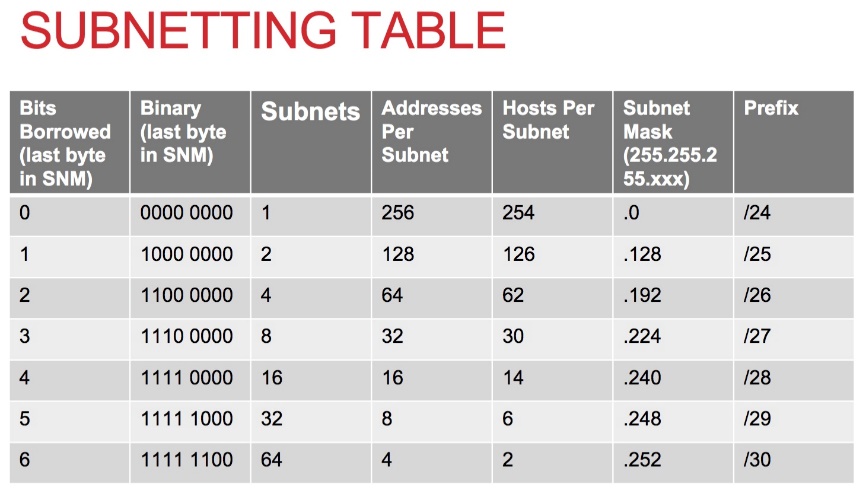
1. Script displays the correct output(s) for the number of subnets entered by the user
2. Script detects invalid inputs entered for the number of subnets by the user and displays the error message(s)

* **What scripting language would you use to write the code?**

The scripting language I will use to write the code is Bash Script

* **What are 2 characteristics of this language that you need to consider in your design?**

1. Bash script is not programming. It is a command-line interpreter utilised that can be utilised to increase productivity
2. Unless you declare otherwise, an unset variable is used as an empty string. This can cause errors.

* **You must clarify with the teacher which IP Addressing Scheme will be included in your pilot software:**

The table of content to the right showcases

The IP Addressing schemes that will be included in this pilot software

**OUTCOME OF TASK 3:**

ALGORITHM PLAN

1. Set a variable to store the users chosen amount of subnets option (this will also be used to store the exit button if selected at the end of the program)
2. Set a condition regarding the amount of subnets variable to which a loop will iterate if met
3. Check if condition is met. If not met, exit the program. If met:
4. Print a prompt asking the user to choose an amount of subnets
5. Print an alert of how to quit the program
6. Prompt user to enter amount of subnets
7. Store the input as the menu option variable
8. Run through the following conditions and execute the following if met:
9. If subnet variable equals 1, Print the appropriate scheme for 1 subnet
10. If subnet variable equals 2, Print the appropriate scheme for 2 subnets
11. If subnet variable equals 3 or 4, Print the appropriate scheme for up to 4 subnets
12. If subnet variable is less than 9 and greater than 4, Print the appropriate scheme for up to 8 subnets
13. If subnet variable is less than 17 and greater than 8, Print the appropriate scheme for up to 16 subnets
14. If subnet variable is less than 33 and greater than 16, Print the appropriate scheme for up to 32 subnets
15. If subnet variable is less than 65 and greater than 32, Print the appropriate scheme for up to 64 subnets
16. If the number of subnets entered is greater than 64, print an error message regarding the number of subnets needing to be below 64
17. If the number of subnets entered is below 1, print an error message and inform the user to select a subnet amount between 1 and 64
18. Else, print an error message regarding incorrect input and to please try again
19. Go back to step beginning of the while loop(step 3)

**OUTCOME OF TASK 4:**

**Record at least 1 error you encountered during development:**

One Error I encountered during development was the ‘unexpected end of file’ error. Through troubleshooting and research of what this error likely entails, I discovered that the script failed to correctly end the while loop. A simple ‘done’ at the end of the script was needed.

**OUTCOME OF TASK 5:**

1. Use the table provided to create 2 tests. Providing an explanation on how the tests are to be conducted. (5-15 words per test)

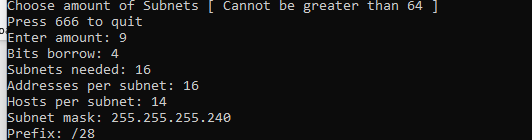
**TESTER: Trong Nguyen**

|  |  |  |
| --- | --- | --- |
| **How the tests are being conducted (5-15 words per test)** | **Test Outcome Success (10-15 words per test)** | **Modifications Required (10-15 words per test)** |
| **Test 1 (allow authorised):** Test input “9” “11” AND “64”. This will test that the if statements test the greater than and less than functions correctly | 1. Successful; Prints the scheme for 16 subnets 2. Successful; Prints the scheme for 16 subnets 3. Successful; Prints the scheme for 64 subnets | N/a |
| **Test 2 (blocking unauthorised):**  Test input “67” “68” AND “69”  All of these inputs are greater than 64, to which the software advises is not valid | 1. Successful; Prints error message for invalid input 2. Successful; Prints error message for invalid input 3. Successful; Prints error message for invalid input | N/A |

**c) After making the required modifications, run the program again to confirm the following:**

**- Specified tests are now working as expected (Take screenshots)**

Test 1.1



Test 1.2

Text

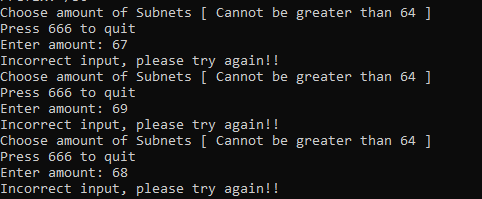
Description automatically generated with medium confidence

Test 1.3

A picture containing graphical user interface

Description automatically generated

Test 2



**- All the steps addressed in the plan (algorithm) are now covered (See Task 3)**

All steps addressed in the algorithm are covered. The loop that checks whether the Menu Option variable is the quit trigger functions. The correct prompt for input from user prints to the terminal and collects the input. This input is then utilised to form an if statement.

**SECTION 6 – OUTCOME OF TASK 6:**

***USER DOCUMENT FOR IP ADDRESSING SCHEME TOOL***

**• The operating system required to run the software application**

The operating system required to run this software is Linux/Unix. However, it is possible to execute this script on a Microsoft Windows and Mac OS utilising a Linux style command-line software called Gitbash. This is available via the following link: <https://git-scm.com/downloads>

• **The command(s) to run the software application**

To execute the script on a Linux OS, type the following command:

chmod +x scripting.sh

**• The values that can be entered in the software application**

The values that can be entered into the software application is any integer between 1 – 64. Any higher than 64 will trigger an error message.

**• The output displayed for each value entered in the software application**

If number subnets entered is 1, the script will output the following:

Bits borrow: 0

Subnets needed: 1

Addresses per subnet: 256

Hosts per subnet: 254

Subnet mask: 255.255.255.0

Prefix: /24

If number subnets entered is 2, the script will output the following:

Bits borrow: 1

Subnets needed: 2

Addresses per subnet: 128

Hosts per subnet: 126

Subnet mask: 255.255.255.128

Prefix: /25

If number subnets entered is above 2 but 4 or below, the script will output the following:

Bits borrow: 2

Subnets needed: 4

Addresses per subnet: 64

Hosts per subnet: 62

Subnet mask: 255.255.255.192

Prefix: /26

If number subnets entered is above 4 but 8 or below, the script will output the following:

Bits borrow: 3

Subnets needed: 8

Addresses per subnet: 32

Hosts per subnet: 30

Subnet mask: 255.255.255.224

Prefix: /27

If number subnets entered is above 8 but 16 or below, the script will output the following:

Bits borrow: 4

Subnets needed: 16

Addresses per subnet: 16

Hosts per subnet: 14

Subnet mask: 255.255.255.240

Prefix: /28

If number subnets entered is above 16 but 32 or below, the script will output the following:

Bits borrow: 5

Subnets needed: 32

Addresses per subnet: 8

Hosts per subnet: 6

Subnet mask: 255.255.255.248

Prefix: /29

If number of subnets entered is above 32 but 64 or below, the script will output the following:

Bits borrow: 6

Subnets needed: 64

Addresses per subnet: 4

Hosts per subnet: 2

Subnet mask: 255.255.255.252

Prefix: /30

**• Error message(s) displayed for invalid values entered in the software**

If invalid input is entered into the software the following Error message will display:

'Invalid input entered for amount of subnets. Try again!!!!'

**• The command(s) to close the software application**

Type in the ‘exit’ command to stop executing the script

**SECTION 7 -- CONCLUSION:**

The tasks outlined within this document aim to give a holistic and transparent overview of the development process. A Software Development Life Cycle is first provided to help dissect the development process into digestible tasks with clear set time frames. This is followed by a client requirement document that details the specifications our client has set with our team. An algorithm plan is provided to set a foundation for the program to be developed from. The pilot program is then referred to within this document with a link where staff can access the script from. Testing Documentation and modifications are then provided to showcase how we have further finetuned the application to meet necessary requirements and client design specifications. The report is then finalised with a user document to assist staff in operating the pilot application come its deployment. The expected date for the deployment of the pilot program is the 02/10/2022.

**APPENDIX**

1. **Introduction**
2. **SLDC**
3. **Client Requirements**

**3.1 Scope of Project**

**3.2 Design Specifications**

**3.3 Scripting Language**

**3.4 Characteristics of the Language**

**3.5 Addressing Scheme**

1. **Algorithm Plan**
2. **Script Development (Please view Subnetting.sh)**
3. **Testing** 
   1. **Test Outcomes**
   2. **Appropriate Modifications**
4. **User Document**
5. **Conclusion**