
System Requirements Specification Index

For

C++ File Handling Basic Application

Version 1.0

TABLE OF CONTENTS

1	Project Abstract	3
2	Common Constraints	3
3	Template Code Structure	3
4	Execution Steps to Follow	4

C++ File Handling Basic Application

System Requirements Specification

1 PROJECT ABSTRACT

C++ file handling basic application is a pure c++ application with Evaluates the test takers on their file handling capabilities.

Test takers are required to create a menu driven program to perform following actions:

1. Create file and add data
2. Open/Create a file and append data
3. Read data from file and show on console
4. Search for some info in File

For above purpose test taker is free to take input of file name and data from user.

2 COMMON CONSTRAINTS

1. Implements common methods used in file handling operations.
2. You need to write down the definitions for multiple function in fileOperations.cpp file.

3 TEMPLATE CODE STRUCTURE

Resources

Function	Description	Status
isFilePresent	<ul style="list-style-type: none">• This function will check if file is present or not.	Partially implemented.
createFileAndWriteData	<ul style="list-style-type: none">• This function will create a file and write some data in it.	Partially implemented.
openFileAndAppendData	<ul style="list-style-type: none">• This function will open a file and append some data in it.	Partially implemented.
readData	<ul style="list-style-type: none">• This function will read the data from file.	Partially implemented.
searchText	<ul style="list-style-type: none">• This function will search some text from the file.	Partially implemented.

4 EXECUTION STEPS TO FOLLOW

1. All actions like compile, running application, running test cases will be through Terminal.
2. To open the command terminal the test takers, need to go to the folder created with user email in Desktop folder, right click in it and click on open terminal.
3. These are time bound assessments the timer would stop if you logout and while logging in back using the same credentials the timer would resume from the same time it was stopped from the previous logout.
4. You can use pre-installed editor i.e VS CODE for code implementation.
5. To compile your project use command:
`g++ -std=c++11 -I/usr/local/include main.cpp -L/usr/local/lib -lgtest -lpthread -lcurlpp -lcurl -o main`
6. To run and test your project, use the command.
`./main`
7. You must push your code after completing implementation. For pushing the files, you can use git from VS CODE. You can click on “Source Control” from the left bar and follow steps as:
 - a. First add the changed files. (by clicking on + sign)
 - b. Add a commit message. (add a message from Message field)
 - c. Push the code.(click on ... option and click “push” option)
8. After pushing the code, you must submit your assessment by clicking on “Submit Application” button.