Popescu Christian

[Email address]

Abstract

[Draw your reader in with an engaging abstract. It is typically a short summary of the document.   
When you’re ready to add your content, just click here and start typing.]

LaSdeCSharpLibrary

[Document subtitle]

**History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Content** |  |
|  |  |  |  |

**Table of Contents**

Table of Contents

[1 Vision 3](#_Toc128814680)

[2 Use cases 3](#_Toc128814681)

[2.1 Business use cases 3](#_Toc128814682)

[2.1.1 BUC-01- Create resources (file structure and folders) 3](#_Toc128814683)

[2.2 Technical use cases 3](#_Toc128814684)

[2.2.1 TUC-01 – Implement a file repository. 3](#_Toc128814685)

[3 Technical constraints 3](#_Toc128814686)

[4 Analysis 3](#_Toc128814687)

[4.1 BUC-01 Analysis 3](#_Toc128814688)

[4.2 TUC-01 Analysis 3](#_Toc128814689)

1. Vision

I need tools to improve the personal Software Development Process.

The tool creation should allow using known widespread technologies and practices.

**To be replaced by a Requirement Management Tool. Azure Devops is good candidate**

1. Use cases
   1. Business use cases
      1. BUC-01- Create resources (file structure and folders)

For different reasons (software project, managing projects, or operational projects) I need to create some folder predefined structure filled up with some files created from scratch or get from some file repository.

* 1. Technical use cases
     1. TUC-01 – Implement a file repository.

As a user or as an application I need to get a file from a file repository.

As a user or as an application I need to add files to a file repository.

* + 1. TUC-02 – Implement a folder archive.

As system I need to archive a folder using a given formula to compute archive name. The archiving should log the result of this action.

1. Technical constraints

Language used for implementation: C#

Framework: Dot Net Core

Interface: WPF (final target MVVM)

To use unit tests.

1. Analysis
   1. BUC-01 Analysis

To implement the Business case, we need a **file repository**. 🡺 Technical use case created **TUC-01 Implement a file repository**.

* 1. TUC-01 Analysis

As a user I give to the system a Logical file Id and the system returns a file corresponding to the logical file id.

**FileRepositoryManager**.

Role – Add file (file\_id, input Path + file name)

* Get file (path)
* CopyFile(toPath)
* List available files

**FilesElement** – Role maps logical file with physical file location

**File Repository**

* Root
* Map between logical filename and path + file.name

**DataRepository** – Role store FilesElements