Software Requirements Specification

For Virtual Letter of Life

Version 1.0

Sefanit Urgessa

Mohammed Allibalogun

Rob Garcia

Jamal Bourne

Augustin Mwamba

Charles Baisie

Meron Debela

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 6/9/2020 | 1.0 | Initial SRS document | Meron Debela |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Introduction 4

1.1 Purpose 4

1.2 Intended Audience 4

1.3 Definitions, Acronyms and Abbreviations 5

1.4 Overview 5

2. Scope 5

2.1 Scope Description 5

2.2 Out of Scope 6

2.3 Future Phase 6

3. Overall Description 6

3.1 Use-Case Model Survey 7

3.1.1 Use-Case Model Survey 7

3.2 Assumptions 8

4. Specific Requirements 8

4.1 Use-Case Reports 8

4.2 Supplementary Requirements 15

5. Nonfunctional Requirements 15

Software Requirements Specification

# Introduction

The Letter of Life is a paper form that provides first responders with a list of important medical information, such as, medical conditions, medications and allergies regarding a patient in need. Currently, this form is completed by a patient and placed inside the residence at a location easily accessible by EMS personnel. However, when/if an emergency occurs outside of patients’ home and if the patient is unresponsive, it is difficult for EMS personnel to have access to the patient’s medical information. Thus, we are proposing a solution that is a Web-based version of this letter, Virtual Letter of Life. Virtual Letter of Life will provide the same level of patient’s medical information to first responders.

## Purpose

The purpose of this Software Requirement Specification (SRS) is to collect requirements for Virtual Life of Letter in conjunction with the client, to achieve an effective alignment between client and project team - EMSPlus for the proposed solution.

The proposed software will implement the following 15 use cases:

|  |  |
| --- | --- |
| 1. Admin Add | 9. User Edit |
| 1. Admin Edit | 10. User View |
| 1. Admin Delete | 11. User Delete |
| 1. Admin View | 12. Log In |
| 1. Admin Listing | 13. Log Out |
| 1. User Listing | 14. Contact |
| 1. Scan/Search | 15. About |
| 1. User Add |  |

## Intended Audience

|  |  |  |
| --- | --- | --- |
| **#** | **Name** | **Role** |
| 1 | Chris L. Truitt | Client |
| 2 | Roy Gordon | External Project Manager |
| 3 | Arezo Sanie | External Project Manager |
| 4 | Dr. Mir Assadullah | Professor |
| 5 | DevOps Team | DevOps |
| 6 | Sefanit Urgessa | Project Manager / Developer |
| 7 | Mohammed Allibalogun | Project Manager / Developer |
| 8 | Rob Garcia | Product Owner / Developer |
| 9 | Jamal Bourne | UI/UX Developer |
| 10 | Augustin Mwamba | Database Developer |
| 11 | Charles Baisie | Security & Deployment Specialist |
| 12 | Meron Debela | Documentation |

## 

## Definitions, Acronyms and Abbreviation

The terms and abbreviations used on this SRS relating to the software being developed:

* EMS – Emergency Medical Services.
* QR code – Quick Response code.
* EMT – Emergency Medical Technician.
* EMR – Electronic Medical Record.
* CRUD – Create, Read, Update and Delete.
* CRISP – Chesapeake Regional Information System for our Patients.
* Sys Admin – The system administrator who has a super user role.
* SQL – Structured Query Language
* HTML – Hypertext Markup Language
* NIST SP – National Institute of Standards and Technology Special Publication
* PII – Personal Identifiable Information
* 2FA – 2 Factor Authentication
* AC – All Controls
* AU – Audit
* IA – Information Assurance
* SC – Systems and Communications
* SI – System & Information Integrity
* W3C – World Wide Web Consortium

## Overview

This section describes the overview of the entire document and provides a brief introduction for each section:

**Section 1:** This section introduce the problem, the purpose of the project, intended audience and definitions/abbreviations.

**Section 2:** This section discusses the scope of the project, what is out of scope and future phases.

**Section 3:** This section discusses the overall description of the project, assumptions and dependencies.

**Section 4:** This section describes specific requirements of the system - use cases.

# Scope

The goal of this project is to develop a web-based software that will allow emergency medical responders to retrieve medical information, such as allergies, medication and medical conditions, which is contained in a patient’s Letter of Life, from any location using a web interface.

## Scope Description

* Provide EMS personnel access to a patient’s Letter of Life information, from a laptop and mobile devices through a secure web interface.
* Allow EMS personnel to scan QR code to retrieve Letter of Life information for non-responsive patients.
* Allow patients and authorized users to create, update, read and delete patients’ medical records through a secure interface.

## Out of Scope

* Integration of product with the current Information Technology infrastructure of City of Salisbury, MD System.
* Hands-on training of end users.

## Future Phase

The following are to be completed in a future phase – timeline is to-be-determined upon the agreement of Version II development of Virtual Letter of Life:

Future functionalities

* Integration with CRISP (Chesapeake Regional Information System for our Patients) to update Virtual Letter of Life automatically.
* Reminder email: ability for the system to send reminder emails for patient to update their medical record.

# Overall Description

The basic functionality of the system is to provide EMS personnel access to patient’s Letter of Life information from a mobile device by scanning QR code or performing a search. The system will also allow patients and authorized users to create, read, update and delete records using web interface.

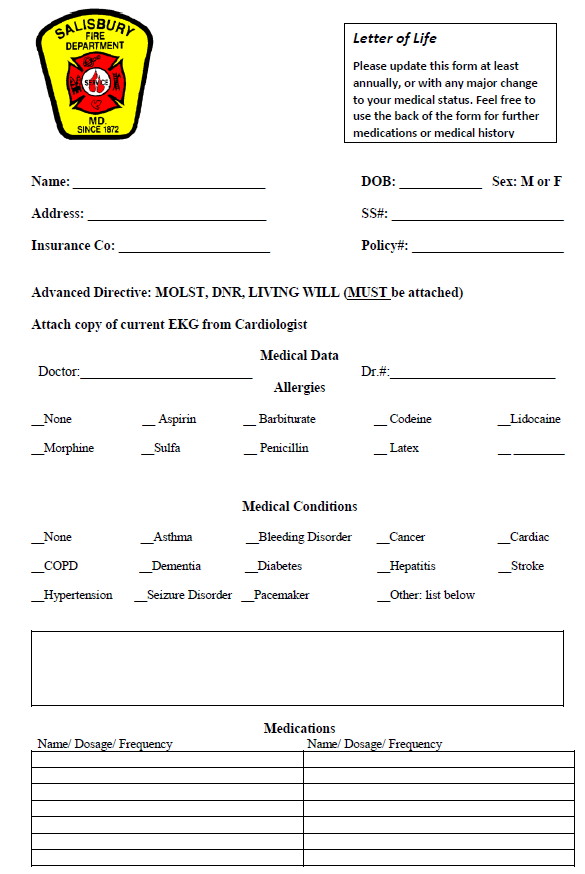
## Use-Case Model Survey

The table below lists the use cases that make up the basic requirements for the virtual letter of life, which will be constructed based on the existing paper Letter of Life form. The details of each use case are located in section 4 of this document.

|  |  |
| --- | --- |
| **Use Case** | **Description** |
| Admin Add | Creates user account. |
| Admin Edit | Edits existing account. |
| Admin Delete | Removes user from the system. |
| Admin View | Displays user information. |
| Admin Listing | Lists existing users of the system. |
| User Listing | Lists existing patients. |
| Scan/ Search | Allows EMT personal to scan barcode or search for patient and retrieve letter of life of a patient. |
| User Add | Adds patient into the system. |
| User Edit | Edits existing patient information. |
| User View | Displays patient information. |
| User Delete | Removes patient information from system. |
| Log In | Enables user to login to the system. |
| Log Out | Enables user to log out of the system. |
| Contact | Displays contact information. |
| About | Displays an introduction of the web interface. |

* + 1. **Required Data**

The required data is directly derived from this existing Letter of Life that is currently in use by EMS (see below).



## Assumptions and Dependencies

This project will be focusing on providing the Virtual Letter of Life, electronic medical record (EMR) to emergency medical responders. So our assumption is that internet bandwidth is always up and functioning on the devices that will be used to access the virtual letter.

# Specific Requirements

There are 15 use cases that makes up the basic functionality of the virtual letter of life web interface. The detail for each is described as use case in the section below.

## Use-Case Reports

**Use Case Name: Admin Add**

**Summary:** The Admin Add feature allows a new user to get added into the system.

|  |  |
| --- | --- |
| **Actors** | System Administrator |
| **Trigger** | The system admin triggers the Admin Add event. |
| **Precondition** | System administrator has a super user role access. |
| **System Response** | The system prompts the system admin to enter information of the person that is being added. Once the required information is provided, the admin selects Save. |
| **Post Condition** | The new user gets added into virtual letter of life database and new account gets generated. User now have the ability to login. |
| **Alternative Path** | The system rejects input information. No account gets created. |

**Use Case Name:** **Admin Edit**

**Summary:** The Admin Edit feature allows the sys admin to modify an existing user profile.

|  |  |
| --- | --- |
| **Actor** | System Administrator |
| **Trigger** | The system admin triggers the Edit feature event. |
| **Precondition** | System administrator has a super user role access and a user has an existing profile. |
| **System Response** | System displays current user information and prompt admin to make change as needed. |
| **Post Condition** | Modified user information displayed. |
| **Alternative Path** | Cancel editing user information. |

**Use Case Name: Admin** **Delete**

**Summary:** The Admin Delete feature enables the sys admin to remove an existing user form the system.

|  |  |
| --- | --- |
| **Actor** | System Administrator |
| **Trigger** | The system admin triggers the Admin Delete event. |
| **Precondition** | System administrator has a super user role access and a user have existing profile. |
| **System Response** | System displays existing user information and prompt admin to delete user. |
| **Post Condition** | User removed from the system. |
| **Alternative Path** | Cancel deletion. |

**Use Case Name: Admin** **View**

**Summary:** The Admin View feature allows the sys admin to view a specific user information.

|  |  |
| --- | --- |
| **Actor** | System Administrator |
| **Trigger** | The system admin triggers the Admin View event. |
| **Precondition** | System administrator has a super user role access and there are existing user profiles. |
| **System Response** | System displays existing user information. |
| **Post Condition** | User information displayed. |
| **Alternative Path** | None. |

**Use Case Name: Admin** **Listing**

**Summary:** The Admin Listing feature allows the sys admin to view lists of users that are in the virtual letter of life database.

|  |  |
| --- | --- |
| **Actor** | System Administrator |
| **Trigger** | The system admin triggers the Admin Listing event. |
| **Precondition** | System administrator has a super user role access. |
| **System Response** | System displays list of users. |
| **Post Condition** | List of existing users displayed. |
| **Alternative Path** | None. |

**Use Case Name: User** **Listing**

**Summary:** The User Listing feature allows EMT personnel to view lists of patients’ that are in the virtual letter of life database.

|  |  |
| --- | --- |
| **Actor** | EMT personnel |
| **Trigger** | EMT personnel triggers the User Listing event. |
| **Precondition** | EMT personnel authenticated and patients’ information is in the virtual life of letter database. |
| **System Response** | System allows user to view list of patients. |
| **Post Condition** | List of existing patients displayed. |
| **Alternative Path** | None. |

**Use Case Name: Scan/Search**

**Summary:** The Scan/Search feature allows EMT personnel to scan barcode or QR code and search patient’s letter of life from the virtual letter of life database.

|  |  |
| --- | --- |
| **Actor** | EMT personnel |
| **Trigger** | EMT personnel triggers the Scan/Search event. |
| **Precondition** | EMT personnel authenticated and patient information is in the virtual life of letter database. |
| **System Response** | System prompts user to scan barcode or QR code. |
| **Post Condition** | Patient letter of life displayed on the screen. |
| **Alternative Path** | Cancel search. |

**Use Case Name: User Add**

**Summary:** The User Add feature allows individuals (patients) to create an account.

|  |  |
| --- | --- |
| **Actor** | Patient |
| **Trigger** | Patient triggers the User Add event. |
| **Precondition** | None. |
| **System Response** | System displays list of questions for the patient to complete and save. |
| **Post Condition** | Patient information is added into the database. |
| **Alternative Path** | Cancel and no account gets created. |

**Use Case Name: User** **Edit**

**Summary:** The User Edit feature allows patient to edit existing information.

|  |  |
| --- | --- |
| **Actor** | Patient |
| **Trigger** | Patient triggers User Edit event. |
| **Precondition** | Patient have an existing account. |
| **System Response** | System displays existing information and prompt user to modify information as needed. |
| **Post Condition** | The modified information gets stored in the database. |
| **Alternative Path** | Cancel change and nothing gets modified. |

**Use Case Name: User** **View**

**Summary:** The User View feature allows patient to view his/her account information.

|  |  |
| --- | --- |
| **Actor** | Patient |
| **Trigger** | Patient triggers User View event. |
| **Precondition** | Patient have an existing account. |
| **System Response** | System displays existing account information. |
| **Post Condition** | Patient account information gets displayed. |
| **Alternative Path** | None. |

**Use Case Name: User** **Delete**

**Summary:** The User Delete feature allows patient to delete his/her account from the system.

|  |  |
| --- | --- |
| **Actor** | Patient |
| **Trigger** | Patient triggers the User Delete event. |
| **Precondition** | Patient have an existing account. |
| **System Response** | System prompt patient to confirm and proceed with deletion. |
| **Post Condition** | Account gets deleted from the system. |
| **Alternative Path** | Cancel deletion. |

**Use Case Name**: **Log In**

**Summary:** The Log In feature allows users (EMT personnel, system admins and patients) to login into the system.

|  |  |
| --- | --- |
| **Actor** | EMT personnel, system admins and patients. |
| **Trigger** | User triggers Login option. |
| **Precondition** | EMT personnel must be authenticated and have an existing account. |
| **System Response** | System prompts user to enter login information. |
| **Post Condition** | User is logged into virtual letter of life interface. |
| **Alternative Path** | Cancel login. |

**Use Case Name**: **Log Out**

**Summary:** The Logout feature allows user to log out of the system.

|  |  |
| --- | --- |
| **Actor** | EMT personnel, system admins and patients. |
| **Trigger** | User triggers Logout event. |
| **Precondition** | User is logged into the system. |
| **System Response** | System logs out user from account. |
| **Post Condition** | User is logged out of virtual letter of life interface. |
| **Alternative Path** | None. |

**Use Case Name**: **Contact**

**Summary:** The Contact feature provides user with contact information.

|  |  |
| --- | --- |
| **Actor** | EMT personnel, system admins and patients. |
| **Trigger** | User triggers Contact event. |
| **Precondition** | User is on virtual letter of life website. |
| **System Response** | System displays contact information. |
| **Post Condition** | Use views contact information page. |
| **Alternative Path** | None. |

**Use Case Name**: **About**

**Summary:** The About feature provides introduction of the web-interface.

|  |  |
| --- | --- |
| **Actor** | EMT personnel, system admins and patients. |
| **Trigger** | User triggers About event. |
| **Precondition** | User is on virtual letter of life website. |
| **System Response** | System displays an introduction of the site. |
| **Post Condition** | Use views the introductory page. |
| **Alternative Path** | None. |

## Supplementary Requirements

* Software Specifications:
  + Operating System: Windows Server 2019.
  + Web Server: Microsoft Internet Information Services (IIS) 10.0.
  + Database Server: Microsoft Structured Query Language (SQL) Server 2019.
  + Language and Framework: C# 8.0 / .NET Core 3.1. / HTML / CSS/ JavaScript
  + Repository: GitHub / UMGC Repository

* Upon completion, this web-based interface shall be compatible to run on iOS/Mac and Window/Android.

# Nonfunctional Requirements

Nonfunctional requirements of the virtual letter of life interface are:

* Limit Create-Read-Update-Delete (CRUD) access to authenticated and authorized users only, to preserve the confidentiality and integrity of Personally Identifiable Information (PII).
* New registrants shall verify their accounts via a link sent via email (2FA) to mitigate the creation of fake accounts.
* All non-verified accounts to be deleted within 24 hours to prevent database bloating.
* All user input should be validated to prevent injection and scripting attacks.
* User sessions should timeout within 30 minutes to prevent inadvertent disclosure of PII.
* Limit multi-account access to authorized devices only (MAC Whitelisting) to preserve the confidentiality and integrity of PII, as well as the availability of the system.
* The following NIST SP 800-53 controls enforced throughout the application to preserve the confidentiality and integrity of PII, as well as the availability of the system:
  + AC-2: ACCOUNT MANAGEMENT
  + AC-3: ACCESS ENFORCEMENT
  + AC-7: UNSUCCESSFUL LOGON ATTEMPTS
  + AC-8: SYSTEM USE NOTIFICATION
  + AC-11: SESSION LOCK
  + AC-12: SESSION TERMINATION
  + AU-2: AUDIT EVENTS
  + AU-8: TIME STAMPS
  + IA-2: IDENTIFICATION AND AUTHENTICATION (ORGANIZATIONAL USERS)
  + SC-13: CRYPTOGRAPHIC PROTECTION
  + SC-23: SESSION AUTHENTICITY
  + SI-10: INFORMATION INPUT VALIDATION
  + SI-11: ERROR HANDLING
* All SQL queries to use prepared statements to prevent SQL injection attacks.
* All tables to be normalized to 3rd Normal Form to improve data integrity and reduce data redundancy.
* Unit test all functions and methods to ensure boundary conditions are not violated.
* Incorporate static code analysis, using both automated and manual methods to ensure there are no weaknesses due to code syntax.
* Incorporate dynamic code analysis and penetration testing to ensure there are no vulnerabilities due to code semantics.
* Style and comment all code per the appropriate style guide to properly maintain and turn-over the application, as necessary.
* Incorporate version control to audit and track code generation.
* All issues and risks shall be identified during static and dynamic analysis, corrected before uploading to a repository to ensure the repository's code meets the bug bar.
* If identified issue and risk cannot be corrected immediately, it should be recorded in a bug log to ensure the issue or risk is corrected before deployment.
* All presentation code to be HTML5 and W3C compliant to ensure the application provides the same functionality across different platforms.