**Software Requirements Specification for**

**“Online monitoring system for automatic locks”**

**Version 1.0**

**Prepared by Trifan Filip-Adrian, Ghibuci Andrei-Gabriel,Manea Dragos-Adrian , Cretan Alexandru-Florin**

**March 8, 2020**

**Table of Contents**

* **Introduction**
* **Purpose**
* **Product Scope**
* **Intended Audience and Reading Suggestions**
* **Definitions, acronyms, and abbreviations**
* **References**
* **Overall Description**
* **Product Perspective**
* **Product Function**
* **User Classes and Characteristics**
* **Operating Environment**
* **Design and Implementation Constraints**
* **User Documentation**
* **Assumptions and Dependencies**
* **External Interface Requirements**
* **User Interfaces**
* **Hardware Interfaces**
* **Software Interfaces**
* **Communications Interfaces**
* **System Features**
* **Account Login**
* **Account Logout**
* **Admin Add Member**
* **Admin Edit Member**
* **Other Non-functional Requirements**
* **Performance Requirements**
* **Safety Requirements**
* **Security Requirements**
* **Software Quality Attributes**
* **Business Rules**
* **Other Requirements**

**1.Introduction**

**1.1 Purpose**

* This document describes the requirements for an online monitoring of an automatic lock.
* **Product Scope**

Our product is a web application that serves as an online monitoring of an automatic lock. This kind of locks are a vital part of any building nowadays. So in this scope we will develop an application that will manage the access to an automatic lock. The lock will be simulated with an Arduino UNO V3 and a rfid sensor, and will be emulated with four rfid cards.

* **Intended Audience and Reading Suggestions**

This document is for developers, project managers and users.

In the first chapter you will find the purpose and references used to make product.

In the subchapter 2.2 are enumerate the functions realized by the product and o short description of each function, information about the components of each class (lock, administrator, employees of administration etc.). The attribute and operations of each are presented in subchapter 2.3 through an image.

Chapter 4 describes the functional requirements.

* **Definitions, acronyms, and abbreviations**

This Document was created based on the IEEE template for System Requirement Specification Documents.

N.A – Not available

Font – Arial

Dimension – 11

The document is structured on chapters and each chapter in more subchapters.

The language used for this document is English.

**1.5 References**

IEEE. IEEE Std 830-IEEE Recommended Practice for Software Requirements

<https://github.com/>

<https://www.sourcetreeapp.com/>

<https://angular.io/>

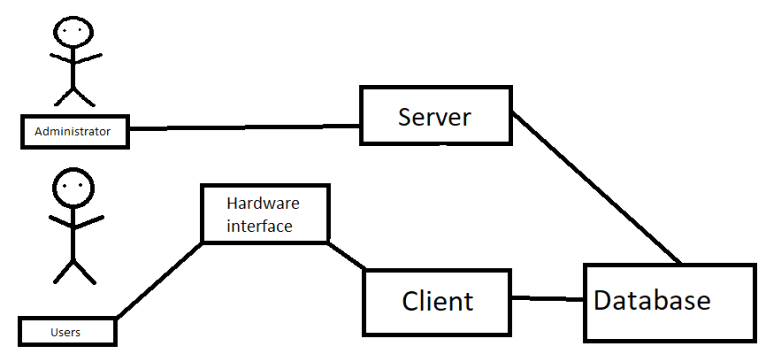
<https://material.angular.io/>

<https://dotnet.microsoft.com/apps/aspnet>

<https://www.microsoft.com/en-us/sql-server/sql-server-downloads>

**2. Overall Description**

**2.1 Product Perspective**



UnlockCells is an online web application that is used to manage an automatic lock system. Bassically, this web application is emulated by the users that access the lock with a card. Everytime an action is done on the lock it will be registered in the database and sent forward to the web page. Another managing actions are the posibility to block, remove,add and see users of the cards.

**2.2 Product Function**

* **Login**
* **User interface**
* **Administrator interface**
* **Poll for different services**
* **Daily report for the usage of the lock**
* **Administrators manage the users of the cards**

**2.3 User Classes and Characteristics**

The application users will be standard employees ( hardware interface part) as they will use their cards to enter a location, but also the employees that work as administrators for a building ( software interface part) as they manage who can enter in the building and also generates reports ( for executive).

**2.4 Operating Environment**

As mentioned before the app will require users to have access to a web browser on their workstation computer like Google Chrome(v 82.0) or Mozilla Firefox(v 73.0), Internet Explorer(v 11, 10, 9), Edge(v 80.0.361.62)

**2.5 Design and Implementation Constraints**

The communication between client and server must be quick as the access in a building should be granted immediately, so it needs efficient programming to be taken in consideration. If at any point there will be many users that will access the database at the same time there will be errors, as the sensor will not suport this stress and nor the database.

Backend language:C#, Arduino

Frontend: HTML, CSS,JavaScript

Framework: JSNode

**2.6 User Documentation**

There are two type of users:

- the administrator user is the one who manages the automatic lock and the people whom use it( he restrict the access and can even add or replace the user of a card).

- the standard user is the one who actually tries the lock with it's card and see if the access is granted or not.

**2.7 Assumptions and Dependencies**

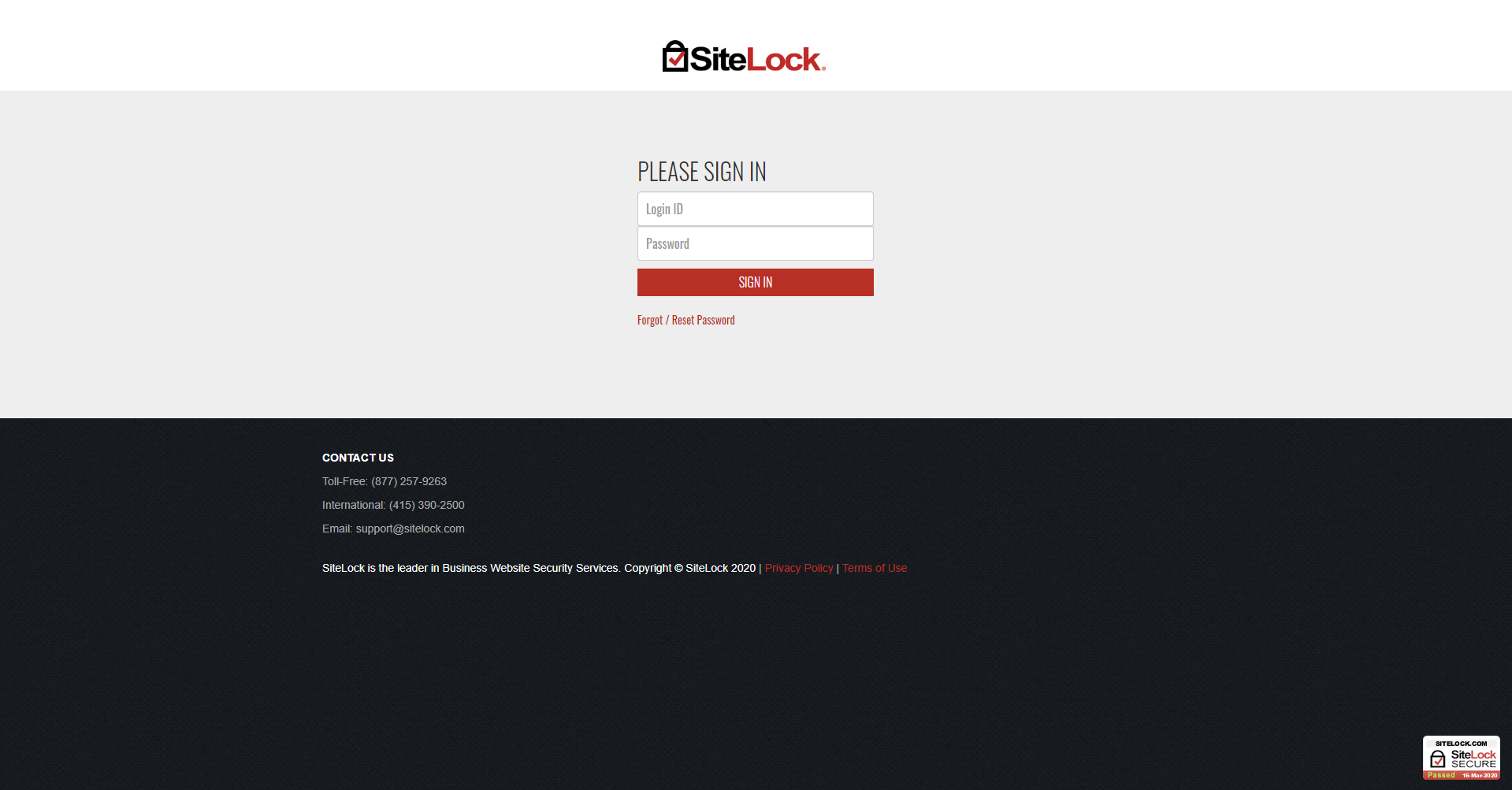
The web application assumes that the user has a computer with an Internet connection and a web browser to access the app. The system may not behave correctly when used with internet browsers other than Firefox and Google Chrome. The hardware interface assumes that any user shall have its own card to scan on the lock.

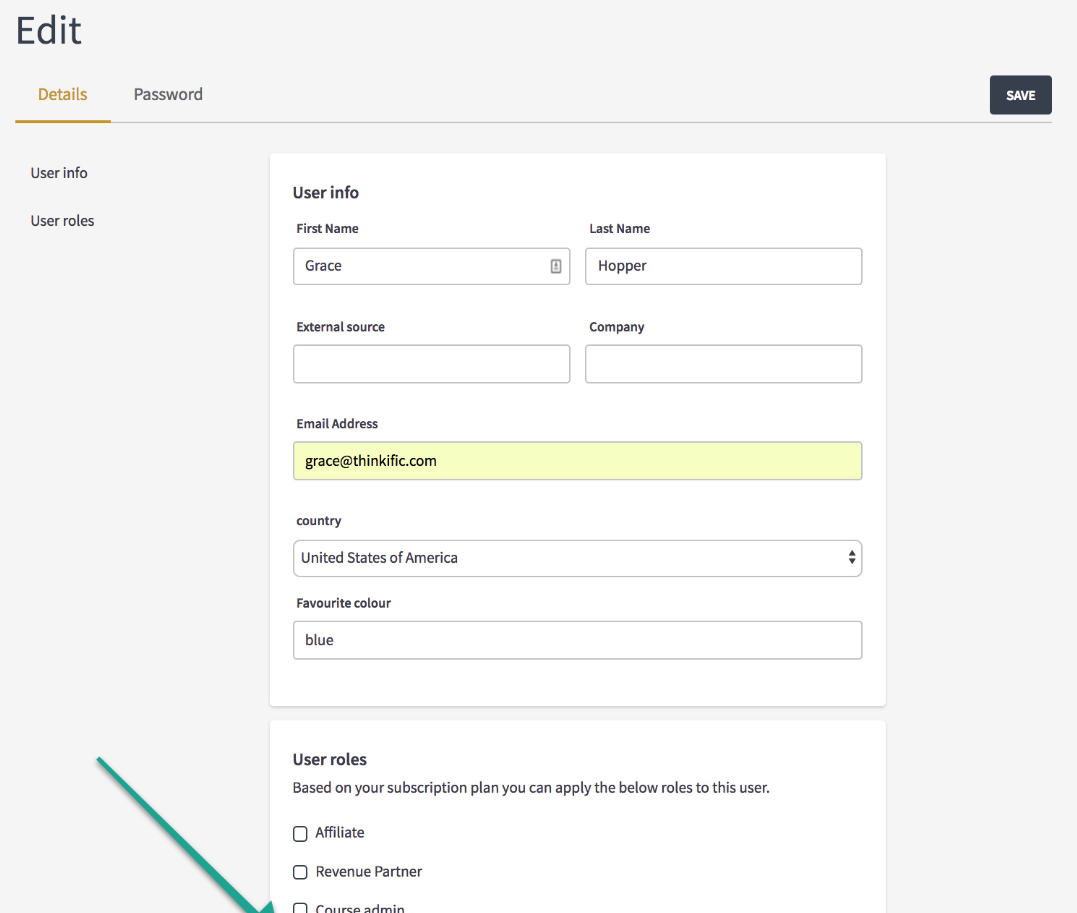
**3. External Interface Requirements**

**3.1 User Interfaces**

The layout should be as follows:

* The header which displays:
* the logo
* The body, which displays
* sign-in window
* after sign-in, a menu with options
* edit options
* logout
* The footer, which displays
* contact
* copyright







There will be two different user interfaces: one for the administrators(website) and one for the users(hardware interface).

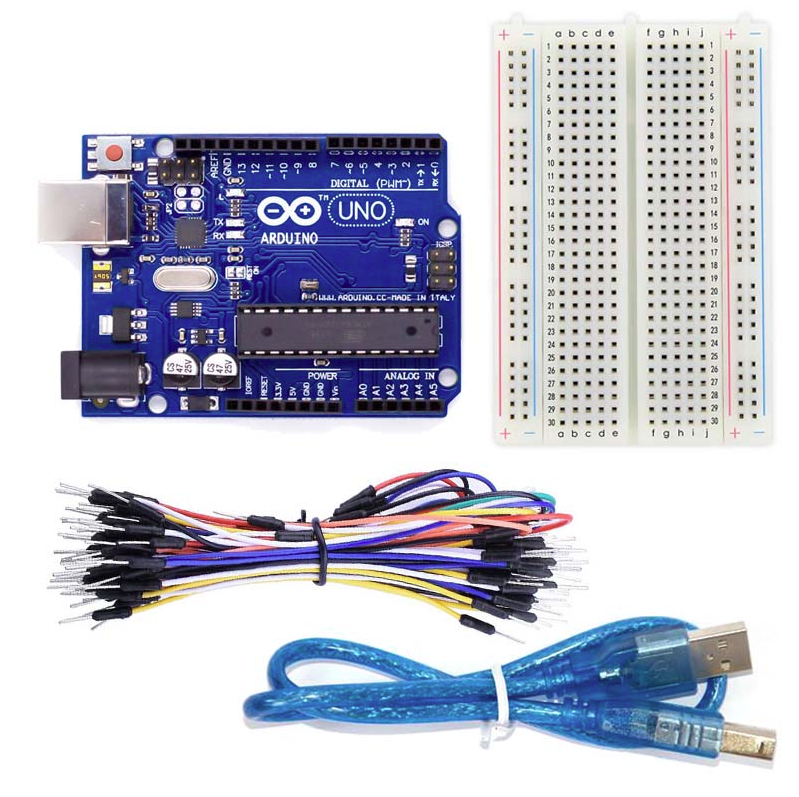
The administrators will be able to add users and remove them, to block them, to generate raports of the usage of the lock.

The users will be able to use the lock if they have the permission to do so.

**3.2 Hardware Interfaces**

* RFID Card
* RFID Sensor



* Arduino UNO v3
* Jumper cables
* USB cable
* 

**3.3 Software Interfaces**

The application is connected to a database (SQL Server). The database will communicate with the server, the information the server receives comes from the database and it is registered there with the help of the client that collects information from the usb serial port. The information comes through the serial port everytime a standard user tries the lock with a RFID Card.

**3.4 Communications Interfaces**

The communication protocols that our app will use:

* HTTP
* Port serial communication protocol