**Software Requirements Specification**

**for**

LMS Project

**Version 1.0 approved**

**Prepared by**

Luis Sierra

Cuong Pham

Kevin Martinez

Shawn Brautigam

**University Of Houston Downtown**

**04/25/2018**

**Table of Contents**

[**Introduction**](#_9owxyylhhoxb) **4**

[**Purpose**](#_3znysh7) **4**

[**Document Conventions**](#_itiry3xkqaca) **4**

[**Software Development Life Cycle (SDLC)**](#_olnogkgua07o) **4**

[**Intended Audience and Reading Suggestions**](#_p9gupj9oij3) **4**

[**Product Scope**](#_smy3m7abqof) **4**

[**References**](#_cwfctdk7lcck) **4**

[**Overall Description**](#_gb9lp97e6qv2) **5**

[**Product Perspective**](#_2s8eyo1) **5**

[**Product Functions**](#_po0e6niy8ykn) **5**

[**User Classes and Characteristics**](#_jg9dibvepm5c) **5**

[**Operating Environment**](#_duwpull4zhht) **6**

[**Hardware requirements (Minimum)**](#_seciz9wv5l5z) **6**

[**Installation requirements**](#_2ei877id5w2k) **6**

[**Design and Implementation Constraints**](#_8mi6ohq9g12z) **6**

[**User Documentation**](#_619y21h2np8a) **7**

[**Assumptions and Dependencies**](#_meq94eduopaj) **7**

[**External Interface Requirements**](#_mwo3f1vy6kpl) **7**

[**User Interfaces**](#_2kc21jkld659) **7**

[**Hardware Interfaces**](#_1gw0vsna35py) **11**

[**Software Interfaces**](#_r5gifoal6h4q) **11**

[**Communications Interfaces**](#_8qnzxjoylx9i) **11**

[**System Features**](#_nt8ca9p9q11) **11**

[**System Features**](#_4y63091vnwxj) **12**

[**Performance Requirements**](#_2bn6wsx) **12**

[**Safety Requirements**](#_1ccgl5pfx0nc) **12**

[**Security Requirements**](#_colp7ygoqtm3) **12**

[**Business Rules**](#_6gp4g28dysaz) **13**

[**Appendix A: Glossary**](#_fephlbk9k6d0) **13**

[**Appendix B: Analysis Models**](#_s9u0gouzrg1) **13**

[**UML Class Diagram**](#_wa9y6lu14b56) **13**

[**Use Case Diagram**](#_txsnox511ac) **14**

[**Architecture Diagram**](#_hehej13ueigx) **15**

**Work History**

|  |  |  |
| --- | --- | --- |
| **Name** | **Dates** | **Reason For Changes** |
| Luis Sierra | April 4, 2018 to April 24, 2018 | Created Google docs based on IEEE Std 830-1998 template, edited hardware requirements and software installation requirements. Added User Interfaces, added security limitations, reviewed/tested application, reviewed specifications. |
| Shawn Brautigam | April 4, 2018 to April 24, 2018 | Added several description to the specifications for the purpose, Document conventions, intended audience reading, Product scope, product perspective, product functions, etc Reviewed specifications. |
| Kevin Martinez | April 4, 2018 to April 24, 2018 | Created GitHub repository. Created UML Diagrams, Database Diagram, use case diagram, flow diagram. |
| Cuong Pham | April 4, 2018 to April 24, 2018 | Created UML Diagrams, Database Diagram, use case diagram, user interface, flow diagram, coded application. |

# **Introduction**

## **Purpose**

The purpose of this document is to outline the specifications of the Learning Management System. It will explain the features, purposes, and design goal of the program. The document is intended to be used by both the users and developers of the product.

## **Document Conventions**

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

## Software Development Life Cycle (SDLC)

Interactive and and Incremental Development

## **Intended Audience and Reading Suggestions**

This document is intended for students and professors who are using the Learning Management System. The document contains a list of user permissions, software functions, and a reference guide for how to use the Learning Management System. The recommended reading order for a given user of the management system is to find out intended user permissions in section 2, then proceed to sections 3 and 4 for specifications on how to accomplish a given task. System administrators should read the full document to gain an understanding of other users’ permissions and the full scope of the software’s capabilities and functions.

## **Product Scope**

The Learning Management System is a program intended to allow for students, professors, and administrators of a college or university create, assign, and keep track of courses, assignments, and grades.

## **References**

“IEEE Std 830-1998 - IEEE Recommended Practice for Software Requirements Specifications.” *IEEE 830-1998 - IEEE Recommended Practice for Software Requirements Specifications*, 25 June 1998, https//standards.ieee.org/findstds/standard/830-1998.html.

Dates Accessed April 2, 2018 - April 24, 2018

“.NET Framework 4.7, 4.6, and 4.5.” *.NET Framework 4.7, 4.6, and 4.5 | Microsoft Docs*, https//docs.microsoft.com/en-us/dotnet/framework

Dates Accessed April 2, 2018 - April 24, 2018

Version 11.0.2 - Tweet. “Json.NET.” *Newtonsoft*, www.newtonsoft.com/json.

Dates Accessed April 2, 2018 - April 24, 2018

“.NET Framework System Requirements.” *Microsoft Docs*, docs.microsoft.com/en-us/dotnet/framework/get-started/system-requirements.

Dates Accessed April 2, 2018 - April 24, 2018

# **Overall Description**

## **Product Perspective**

The Learning Management System is an independently created software intended to be used as a student/class database tool for college and university students and professors. The system can keep track of student information and track class information for students, and allows for management of assignments and grades by professors.

The software will allow students to access their classes to view their assignments, their grades, and be able to calculate overall GPA for a semester. Professors will be able to create assignments, assign grades, and look at basic information of students. Administrators can add and remove courses, can assign professors to courses

## **Product Functions**

The completed LMS will allow a student to:

* Retrieve basic student information
* Keep track of student courses in a semester
* View grades for the student’s course
* Calculate student’s grade point average for a given semester

The LMS will allow a professor to:

* Retrieve basic student information
* Add/modify assignments within their classes
* Add/modify assignment grades in their classes

The LMS will allow an administrator to:

* View and modify student information
* Add/remove available courses

## **User Classes and Characteristics**

Student users will be able to view information, observe assignments/grades, and calculate grade point average.

Professor users will be able to view student information, create/grade assignments, and view student grades in a course.

Administrators will be able to view and modify student information, create/delete courses, add/remove students from a course, assign professors to a course.

## **Operating Environment**

* Windows 7 and Up with .net 5.6.2
* All other OSs with .NET Framework support

## Hardware requirements (Minimum)

|  |  |
| --- | --- |
|  |  |
| **Processor** | 1 GHz |
| **RAM** | 512 MB |
| **Disk space (minimum)** |  |
| 32-bit | 4.5 GB |
| 64-bit | 4.5 GB |

## Installation requirements

The .NET Framework requires administrator privileges for installation. If you don't have administrator rights to the computer where you'd like to install the .NET Framework, contact your network administrator.

## **Design and Implementation Constraints**

Framework Limitations: The LMS is designed and run in .net framework, no other frameworks can be use to run the LMS

Security Limitations:

* The LMS uses .json file to store all the data. The Json file can be accessed via installation path. All the information in the json file can be read or altered using a json reader/ writer.
* Login form will not lock the username if there are many attempts to enter the correct password. This software is not protected against brute force attacks.

## **User Documentation**

*End User Manual had not been generated for this Learning Management System until 04/24/2018*

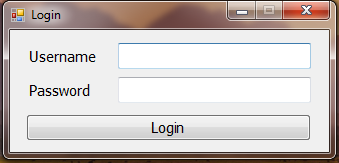
## **Assumptions and Dependencies**

The LMS is programmed in C#, so it is assumed that the system implementing the software will have C# readily available.

# **External Interface Requirements**

## **User Interfaces**

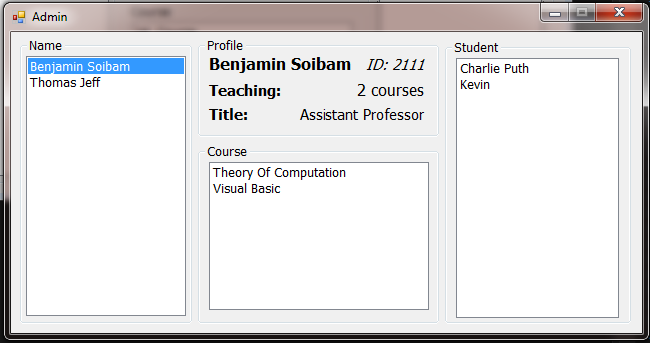
1. **Login Screen**
   1. Window Title: login
   2. User name
      1. Label: Username
      2. User Name Text Box
   3. Password
      1. Label: Password
      2. Password Text Box
   4. Login button
      1. Label: Login
   5. Windows Form buttons
      1. Minimize button
         1. When this button is click the window minimizes and remain the taskbar
      2. Close button
         1. When this button is click the window closes
            1. There is not confirmation window
      3. Maximize button
         1. Is is a disabled button
   6. Functionality
      1. If Username and/or Password are incorrect
         1. Display: “Please try again”
      2. Login screen is not resizable
      3. User can type as many passwords and the system will not lock the username
   7. **Login Screen UI:**

****

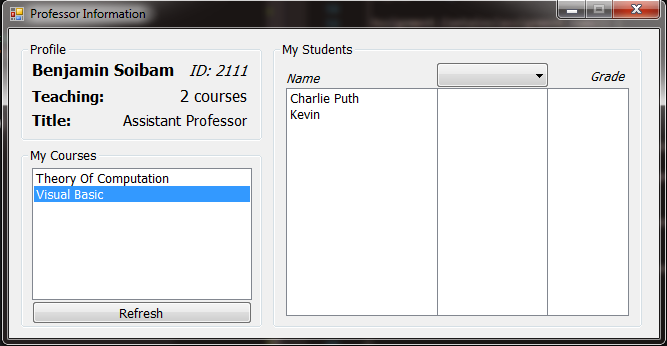
1. **Administration**
   1. Administration Tab
      1. Tab label: Administration
      2. Name (Contain the professor names list)
         1. Label: Name
         2. Contain a Listbox
         3. List all the professor names
      3. Profile (Contain the professor profile)
         1. Label: Profile
         2. Shows:
            1. Professor name and ID
            2. Number of courses that are been teach
            3. Professor Title
            4. List of courses
      4. Student (Contain the list of students enroll in a particular course)
      5. Windows Form buttons
         1. Minimize button
            1. When this button is click the window minimizes and remain the taskbar
         2. Close button
            1. When this button is click the window closes

There is not confirmation window

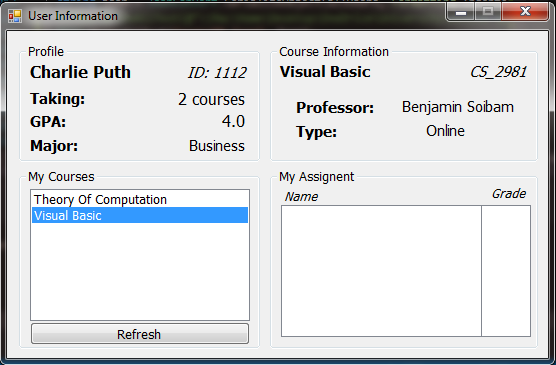
* + - 1. Maximize button
         1. Is is a disabled button
    1. Functionality
       1. When professor name is selected
          1. The professor profile is displayed
       2. When the Course name is Selected
          1. Display all the students enroll in the course
  1. **Administration UI**

****

1. **Professor Information Screen**
   1. Profile (Contain the professor profile)
      1. Label: Profile
      2. Shows:
         1. Professor name and ID
         2. Number of courses that are been teach
         3. Professor Title
         4. List of courses
   2. My Students screen (Contain the list of students)
      1. Student name
         1. label: Name
      2. Grade
         1. label: Grade
   3. Functionality:
      1. When the professor select a course name, the students list enrolled for that course is shown
   4. **Professor Information Screen UI**

****

1. **User Information window**
   1. Profile Group Box(Contain the student profile)
      1. Label: Profile
      2. Shows:
         1. Student name and ID
         2. Number of courses taken
         3. GPA
         4. Major
         5. List of courses
   2. Course Information Group Box
      1. Course name
      2. Professor Name
      3. Type
      4. My Assignment
         1. Name
         2. Grade
   3. Functionality
      1. Clicking on the course, the course information is displayed
   4. **User Information UI**



## **Hardware Interfaces**

*The Learning Management system will interact with the computer hard drive and the computer memory memory. There will not be other interaction with any other device regarding hardware.*

## **Software Interfaces**

*Databases: Not databases are been used, data is stored using json file*

*Operating System: Use windows 7 and up*

*.NET framework 4.6.1*

*Tools: Newtonsoft (.json file) version 11.0.2*

*Data will not be shared between computer or moved over the network*

*Application is installed on a standalone environment*

## **Communications Interfaces**

*Communications Interfaces are not been use, the Learning Management System operates in a standalone environment, therefore; communications protocols like http(s)/FTP are not been used*

# **System Features**

1. *Login Screen*
   1. *It is the access point to the application*
   2. *Authenticate the user*
2. *Administrator:* 
   1. *Insert and Update records*
3. *User*
   1. *View Details of the student*
   2. *Unable to do any modification*

## **System Features**

4.1.1 Description and Priority

*Login Screen: High Priority*

*Administration interfase: High Priority*

*User Interface: High Priority*

4.1.2 Stimulus/Response Sequences

*Login Screen: If user enter a valid Username and Password then user can enter the application*

*Administrations: Add/remove students, teachers and classes*

*User Interface: View students/courses*

4.1.3 Functional Requirements

*Login Screen: If user enter an invalid Username and Password then user can not log in*

*Administration: There are not screens designed to show system or user errors*

*User Interface: There are not screens designed to show system or user errors*

## **Performance Requirements**

*No performance requirements are available, this application is small and does not consume too much computer resources. Data is stored in json file and over time the file size could become large.*

## **Safety Requirements**

*This application does not meet any government standard and it can not be used in a productions environment. This product was developed to meet the requirements of a computer science course.*

## **Security Requirements**

*This product is GPL license, the authors/developers of this application are not responsible for data lost or any other damage that can result during the use of this application.*

*This application was not tested for reliability and robustness. The application was only tested by developers. The application is easy to use and to learn.*

## **Business Rules**

*Administrator: Insert and Update records*

*User: View student Details*

# Appendix A: Glossary

*Label: String used to name a button or a input box*

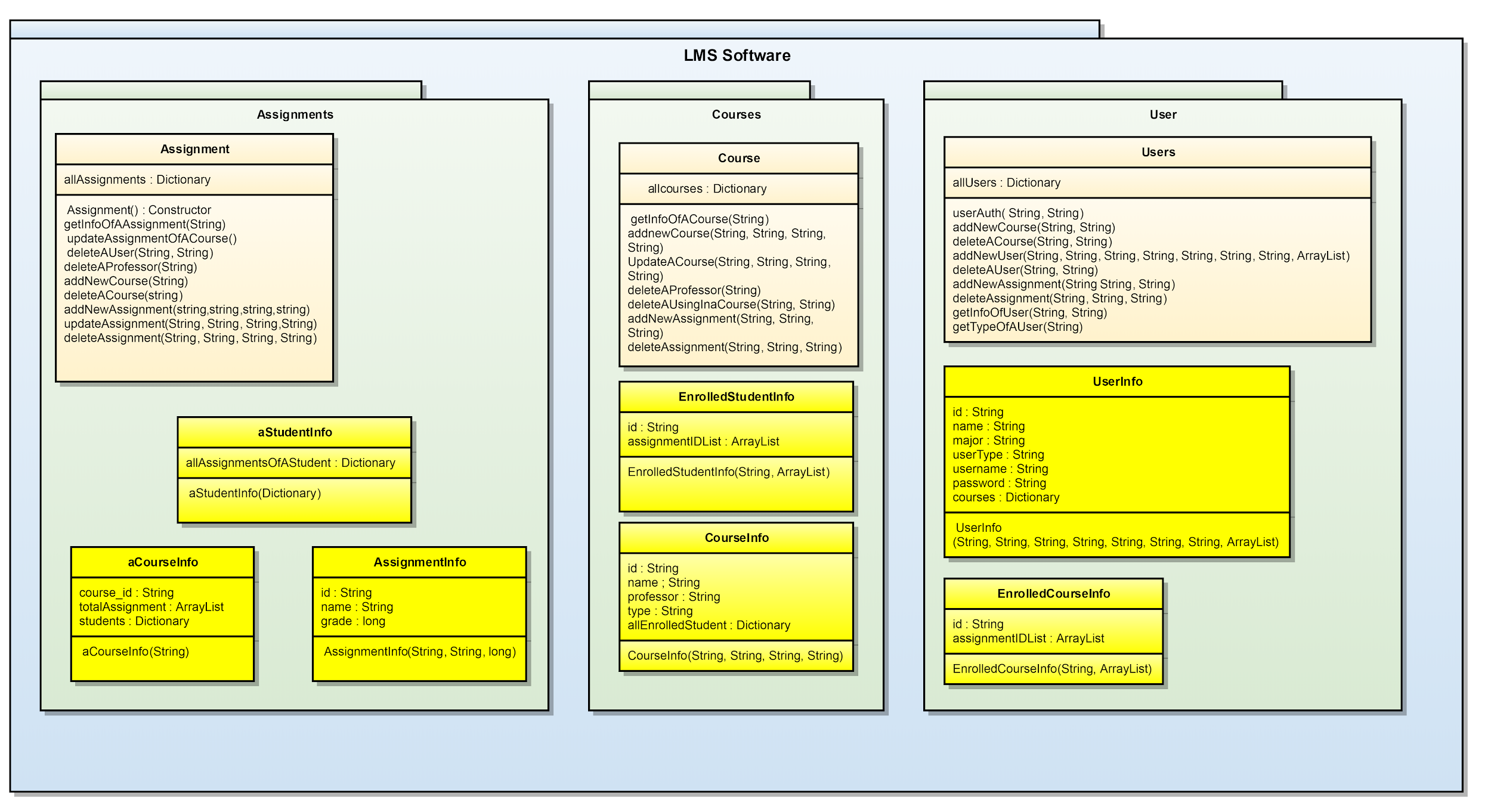
*LMS: Learning Management System*

*SDLC: Software Development Life Cycle*

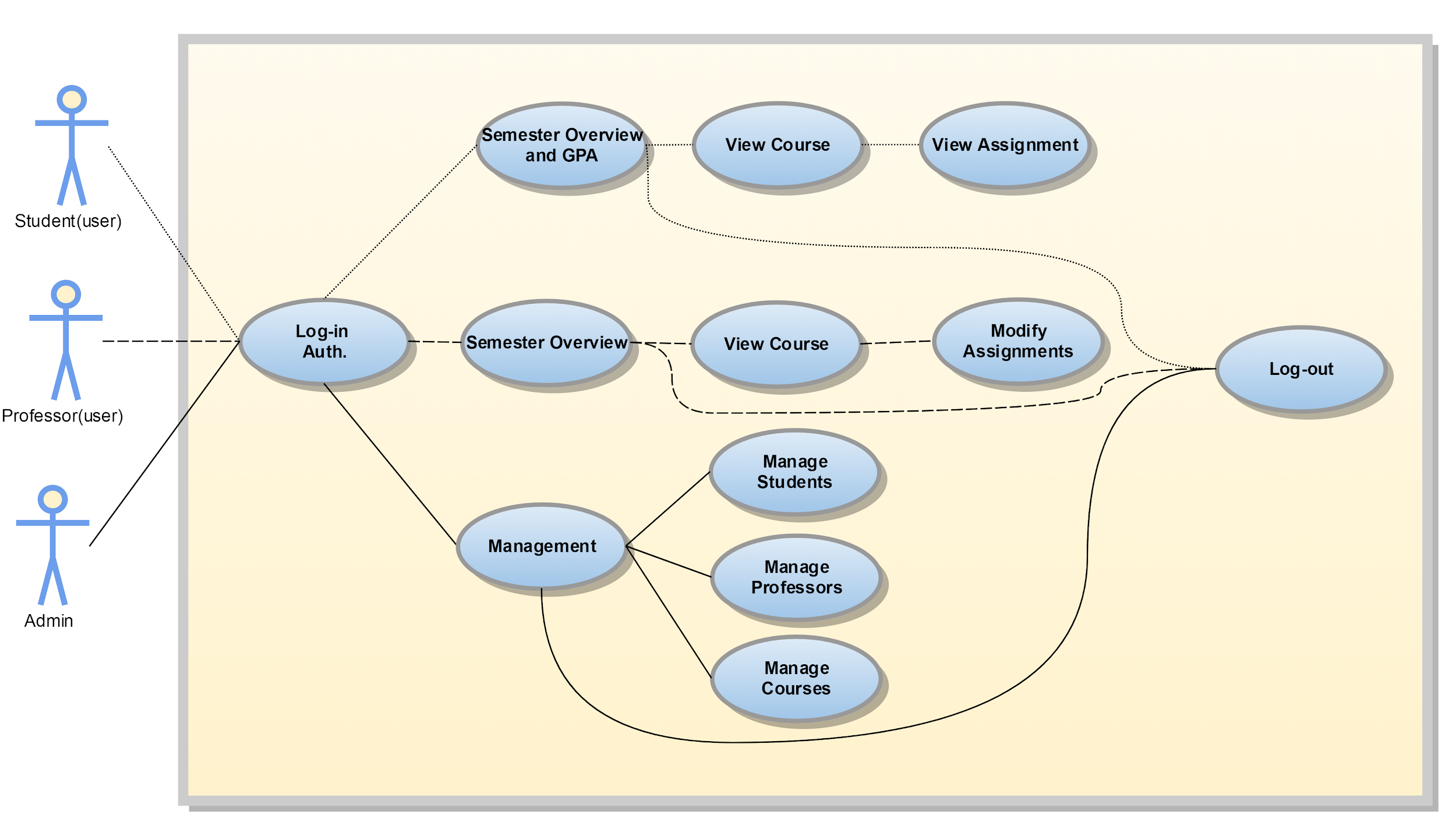
# Appendix B: Analysis Models

*<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams*.>

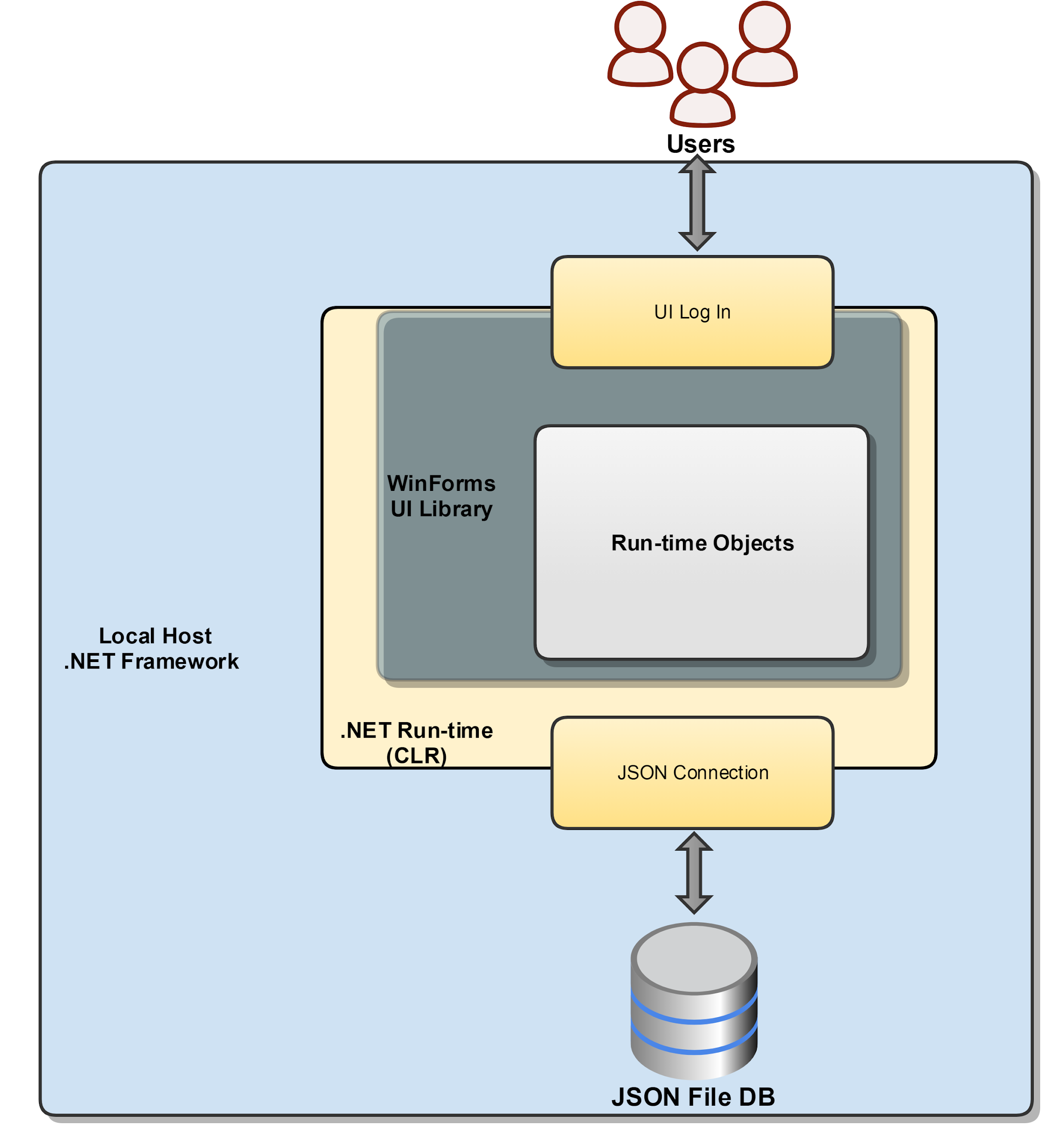
## UML Class Diagram



## Use Case Diagram



## Architecture Diagram

****