VERSION 1.0

CODEFLIX

SOFTWARE REQUIREMENT SPECIFICATION

ANMOL GUPTA
BSC (H) COMPUTER SCIENCE II-YEAR

Table of Contents

Introduction

1. Purpose

   The purpose of this specification document is to provide the detailed description about the software application/project ‘**codeflix**’. This document also explains the different features of the system, the constraints under which the system will operate and how the system reacts on using the system. This document is intended for the development team, the testing team, the design team and the users of the application/system.

2. Scope

   The software application/system is **Educational Content Consuming and Progress Tracking System** for the end user. The project aims to provide the following benefits:

   a. Aggregation of OpenCourseWares, Tutorials and Archives of Lectures.
   b. Providing better browsing and consumption of OpenCourseWares, Tutorials and
   c. Archives of Lectures.
   d. Creating an Interactive Community of Learners using the System as well as Community
   e. for each course/tutorial.
   f. Providing ability to keep track of courses/tutorials.
   g. Providing the ability to add User-Defined courses/tutorials.
   h. Providing the ability to publish/share courses/tutorials developed by the User of the System.
   i. Provide recommendation of courses/tutorials.

3. Definition, Acronyms, and Abbreviations

   a. Definition

      i. YouTube : **YouTube** is an online video platform owned by Google.
      ii. MIT OCW: MIT OpenCourseWare is an initiative of the Massachusetts Institute of Technology to publish all of the educational materials from its undergraduate- and graduate-level courses online, freely and openly available to anyone, anywhere.
      iii. Netflix: Netflix is a streaming service that offers a wide variety of award-winning TV shows, movies, anime, documentaries, and more on thousands of internet-connected devices
      iv. Khan Academy: Khan Academy is an American non-profit[1] educational organization created in 2006 by Sal Khan, with the goal of creating a set of online tools that help educate students. The organization produces short lessons in the form of videos. Its website also includes supplementary practice exercises and materials for educators. All resources are available for free to users of the website and application.
      v. Hosting Services: Provided with the Hardware on subscription bases for the deployment of Application. The Application runs on their Hardware and can be accessed by anyone.

vi. Web Browser: A web browser (commonly referred to as a browser) is application software for accessing the World Wide Web. When a user requests a web page from a particular website, the web browser retrieves the necessary content from a web server and then displays the page on the user's device.

vii. HTML Tags like Input, List, Button etc. are used in User Interface Section for these refer this link

viii. Course/Tutorial refers to an entity that is collection of information about a specific Playlist (Learning) from YouTube.

4. References

   a. IEEE Guide to Software Requirements Specifications (Std 830-1993).

5. Overview

The rest of this document describes the various system requirements, interfaces, features and functionalities in details

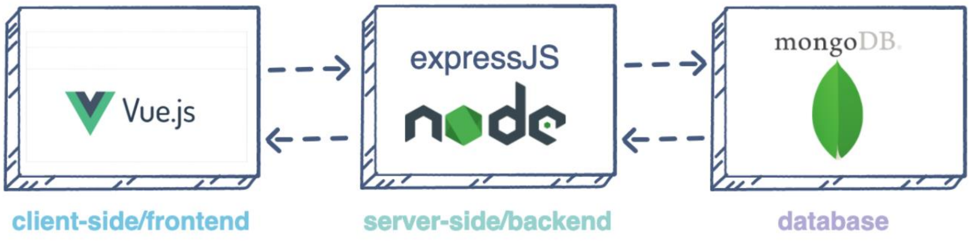
Overall Description

There are a lot of OpenCourseWares, Tutorials and Archives of Lectures previously taught available on the Internet through sites like YouTube, MIT OCW, University Websites etc. These OCWs, Tutorials or Lecture Archives are a great way to learn but due to the current state of their scattered existence in most cases sometimes makes them hard to follow and even complete although having quality material. There is also no sense of achievement like certificate etc. after completion. Some OCWs have top-notch Problem Sets and Assignments that have not been solved and because of not having a community of learners are hard to follow. Although Sites like YouTube provide great way to consume courses or tutorials there is always the distraction factor on YouTube because of availability of non-educational content on YouTube.

The aim of this System (**codeflix**) is to utilize the best feature of Netflix, Khan Academy and YouTube to provide a better and interactive self-learning environment or system and “Community of Learners”.

1. <u>Product perspective</u>

   The application is a Web-based Application which is an independent software that also uses information from different Content Providers like YouTube. The System has the following structure:



   i. <u>System Interface</u>

   ii. <u>User Interface</u>

   The application provides a user-friendly web-based interface with different parts/component of each screen reacting on user’s interaction. The application has the following screens:

   1. The Login Screen: This is where the User/Admin enters their Credentials (Username, Password). On Successfully authentication the User/Admin is granted access to rest of the screens or the whole system.
   2. The Registration Screen: This is where the User enters the Information (Name, Email, Username, Password etc.) to get themselves registered to the System. On successful registration the User now has the ability to access the rest of the system after authenticating themselves using the Login Screen

3. The Home Screen: This is where the User will be redirected after authenticating with the System. This screen will allow user to search and navigate to different Courses/Tutorials (each have their own separate screen) that System provides. This Screen will also display the recommended Courses/Tutorials.
4. The Profile Screen: This is where the User will be presented with the Courses/Tutorial they are taking, current progress on taken the Courses/Tutorial and User Profile. The screen will also allow for changing the User Profile Information and changing/adding/removing Courses/Tutorial that User is taking
5. The Forum Screen: This is where the User can ask for Question and get Answers. Each Course/Tutorial will have their own Forum Screen. The User will be presented with Forum that includes Questions, Answers and Discussion of the Course/Tutorial they selected.
6. The Course Screen: This is where the User can view a Specific Course/Tutorial that the System has.
7. The Course Management Screen: This is where the Admin adds/remove/modifies the various Course/Tutorial or their Information (Video URL, Problem Sets etc.).
8. The User Management Screen: This is where the Admin adds/removes/modifies the Information of the Users of the System.

The application will generate the following reports:

1. The User's Report: This will be generated to show the total number of Registered Users of the System.
2. The Course/Tutorial's Report: This will be generated to show the total number of Course/Tutorial-s and their Information that the System has.
3. The User Profile Report: This will be generated to show the various statistics about the various Course/Tutorial the User has opted-in.

iii. Hardware Interfaces

No Specific Hardware Interface. The System is a web-application that runs on a Server provided by Hosting Services.

iv. Software Interfaces

1. Node.JS (≥15.x): For Creating the API for accessing the Data from the Database.
2. Express.JS ((≥4.16.x).JS: For Creating the Web Server.
3. Vue.JS (≥2.0): For creating the Front-End of the System (Web Application) and Integrating the API made using Node.JS to the Front-End.
4. MongoDB (≥4.0.x): For Database of the System.
5. Mongoose.JS (≥5.10.x): For Accessing Database in Node.JS

6. Visual Studio Code: For Development Purpose of the System.
7. Web Browser: Any Browser that supports the latest HTML, JavaScript.
8. Any Desktop based Operating System for Development.

v. Communication Interfaces
The System will use the Communication Interface provided by the Operating System and the Web Browser.

vi. Memory Constraint
512 MB of RAM for running any Modern Web-Browser that will be used to access our System/Web-Application.

2. Product Functions
The system only allows access to those who are registered to the System. The User/Admin will be able to access the following functions:
    a. The User/Admin will be able to register and login to the System to get access to all the other functions of the System
    b. The User will be able to Browse and Search for the various Course/Tutorial that the System provides.
    c. The User will be able to View all the Information (Number of Lectures, a Specific Video Lecture, Problem Sets, Forum etc.) of about a specific Course/Tutorial.
    d. The User will be able to access the Video of the Lecture provided by the various Course/Tutorial.
    e. The User will be able to access the Problem Sets (if any) provided by the various Course/Tutorial.
    f. The User will be able to access the Forum of a various Course/Tutorial.
    g. The User will be able to add/modify/remove the various Course/Tutorial provided by the System from their Profile.
    h. The User will be able to modify/delete their Profile.
    i. The Admin will be able to add/remove/modify the various Course/Tutorial provided by the System.
    j. The Admin will be able to add/remove the User form the System.
    k. The Admin will be able to modify the Information of the User stored in the System like Name, Password.
3. User Characteristics
    a. The User/Admin should be **Internet-Literate** to be able to use the System. More specifically User/Admin should know how to perform basic operation on a Computer as well as should have knowledge about Internet and Web Browsers.
    b. The Admin should also be experienced about the how YouTube and OCW platforms works to be able to obtain information from these sources.
    c. The Admin should also know about the structure of how the Course/Tutorials and User Information are stored in the System.
4. Constraints

    a. The System is only supported by the Modern Browsers like Google Chrome, Firefox etc. and they must have JavaScript Enabled.
    b. The number of Users and the Course/Tutorial-s are limited by the MongoDB which allows only certain number of records for their free-tier package.
    c. The Internet Connection is the compulsory for accessing the System.
    d. The Log-in Credentials needs to be encrypted before storing them in the Database.

5. Assumption and Dependencies
    a. The System is available to the User/Admin assuming that the Service it was hosted on is Operational.
    b. The User/Admin will have a basic idea of what the System is trying to provide them.
    c. The availability of the System is also dependent on Database and the API to access the Database are Operational or Not.
    d. The System is also dependent on the YouTube and the OCW Platforms if they block access to their Content the System will not be operational.

Specific Requirements

1. External Interfaces Requirements
    a. User Interfaces
        i. Login Screen: This is the Screen that will be presented to the User/Admin when they will go to the URL of the System (Web Application) on the Browser. The User/Admin will be able to enter their Credentials to get Access to Rest of the System. The Screen will have following components:
            1. Username – Input - alphanumeric of length 5 – 25
            2. Password – Input - Alphanumeric with a Special Character of minimum 8
            3. Login – Button - to initiate the Login to the System
            4. Registration – Link - to the Registration Screen

        ii. Registration Screen: This Screen will be presented when the User will navigate to here using the Link in the Login Screen. The User will be able to Register themselves to the System and obtain Credentials to Access the System. The Screen will have following components:
            1. Username – Input - Alphanumeric of length 5 – 25
            2. Password – Input - Alphanumeric with a Special Character of minimum length 8
            3. Name – Input - Alphanumeric of length 5 – 25
            4. Email – Input - Alphanumeric with compulsory '@' of minimum length 5
            5. Age – Input - Numeric of minimum length 2 and minimum value of 13
            6. Login – Button - to initiate the Register to the System

        iii. Home Screen: This is the Screen which is presented to the User after they authenticate to the System. It displays a Search Bar through which User can search various Courses/Tutorials that the System Offers and also displays the recommended Courses/Tutorials. The Screen will have the following components:
            1. Search Bar – Input – Alphanumeric of minimum length 1
            2. Courses/Tutorial – Grid – Recommended Courses/Tutorials. On Clicking any item in the Grid, the User is sent to the Course Screen.
            3. User Profile – Link- to go the User Profile Screen.

        iv. Course Screen: This is the Screen that is specific to each course. It is a template that fills according to the Course/Tutorial User selects/clicks on the Home Screen of recommended Course/Tutorial or Search Results. The Screen will have the following components:

1. Lectures – List – a list of Lectures of specific Course/Tutorial, on clicking/selecting any one item of this list a Pop-Up Modal appears that plays that specific Lecture.
2. Problem Sets – List – a list of Problem Sets of specific Course/Tutorial, on clicking/selecting any one item of this list a Pop-Up Modal appears that displays the specific Problem Set.
3. Forum – Button- to go to the Forum Screen of the specific Course/Tutorial.

v. Profile Screen: This is the screen that the User enters after clicking the Link in the Home Screen. This Screen displays the various Courses/Tutorials that the User has opt-in and Information about the User. The Screen will have the following components:

1. Courses – List – a list of Course/Tutorial opted in by the User. Each List-Item is also accompanied with a Cross Button to remove the Course/Tutorial from the List.
2. Name – Editable Input – Alphanumeric of length 5 – 25
3. Username – Editable Input - Alphanumeric of length 5 – 25
4. Password – Editable Input - Alphanumeric with a Special Character of minimum length 8
5. Email – Editable Input - Alphanumeric with compulsory '@' of minimum length 5
6. Age – Editable Input - Numeric of minimum length 2 and minimum value of 13
7. Save – Button – to Save the modification made by the User.

vi. Forum Screen: This is the Screen that User enters after clicking the Forum Link in a Specific Course/Tutorial Screen they were in. The Screen allows the User to ask Question and get Answers to their specific Question. The User also has the ability to Search Q/A already asked in Forum. The Screen will have the following components:

1. Q/A – List – a list of recently asked Q/As. On clicking an Item in the List a Pop-Up Modal appears which Displays the Question and Answers with that other Users of the System has Submitted. The User is also provided with an Input (Alphanumeric with a minimum length of 5) Area and Submit Button from where they can submit their Answers if they wish to.
2. Question- Button- opens up a Pop-Up Modal in which User is provided with an Input (Alphanumeric with a minimum length of 5) Area and Submit Button to post Question to Forum.
3. Search Bar – Input – to search for a Questions. Alphanumeric with a minimum length of 5

vii. Course Management Screen: This Screen can only be accessed by the Admin. This Screen provide the Admin with all the Course/Tutorial-s that the System has and provides the ability to add/modify/delete Course/Tutorial-s and their Content. The Screen has the following Components:

1. Courses – List – a list of the most recently Courses/Tutorials added to the System. Each List Item is Clickable that opens a Pop-Up Modal that displays editable information about the Course/Tutorial and provides buttons for modifying/removing Information.
2. Search – Input – to search for a specific Course/Tutorial. Alphanumeric with a minimum length of 5. The Output is again a List same as 1.
3. Channel/Playlist URL – Input - Alphanumeric with a minimum length of 5 with compulsorily including the ':'.
4. Add – Button – gets the Data from the provided URL and adding it to the Database.

viii. User Management Screen: This Screen can only be accessed by the Admin. This Screen provide the Admin with all the Users that the System has and provides the ability to modify/delete Users and their Profile Information. The Screen has the following Components:

1. Users – List – a list of the most recently Users added to the System. Each List Item is Clickable that opens a Pop-Up Modal that displays editable information about the User and provides buttons for modifying/removing Information.
2. Search – Input – to search for a specific User. Alphanumeric with a minimum length of 5. The Output is again a List same as 1.

b. Hardware Interfaces
Same as in Overall Description Section.

c. Software Interfaces
Same as in Overall Description Section.

d. Communication Interfaces
Same as in Overall Description Section.

2. Use Case Description/Functional Requirements

➢ **Login/Register**

- Description: This use case describes how the actors Logs in and Register to the System.
- Actors:
  - User
  - Administrator

- Flow of Events:
    - Basic Flow –
        - This use case starts when the actors want to Login or Register to the System.
        - The System asks the actors to select a specific function they want to perform. (Login or Register functions)
        - After the actors select a certain function the following sub-flows are executed according to the selection:
            - On Selection of “Login” function, **Login** sub-flow is executed
            - On Selection of “Register” function, **Register** sub-flow is executed.
        - **Login** sub-flow
            - The System request the actors to enter the following information:
                - Username
                - Password
            - The System validates the entered information by the actors.
        - **Register** sub-flow
            - The System request the actors to enter the following information:
                - Name
                - Username
                - E-mail
                - Age
                - Password
            - The System also select a predefined Profile Picture for the actors and validates the entered information by the actors.
    - Alternative Flow-
        - Invalid Information
            - If the actors enter invalid information in the **Register** or **Login** sub-flow, they will be displayed with an error message.
            - The actors will then go to the beginning of the Basic Flow.
- Pre-conditions:
    - None
- Post-conditions:
    - If the use case was successful the actors will have access to the system, otherwise the state of the System will be remained unchanged.
    - If the actor was the **User,** he/she will have access to the **Browse Courses/Tutorials**, **View a Course/Tutorial** and **View Profile, Track Progress** modules or subprocesses of the System.

        - If the actor was the **Administrator,** he/she will have access to the **User Management** and **Course Management System** modules or subprocesses of the System.

- **Browse Courses/Tutorials**
    - Description: This use case describes how the actors will browse and search for different Courses/Tutorials the System provides.
    - Actors: User
    - Flow of Events:
        - Basic Flow:
            - The use case starts when the actors want to browse and search Various Courses/Tutorials the System displays.
            - The System asks actors to select a specific Course/Tutorial either by selecting from recommended Courses/Tutorials or through "Search" function (calls the **Search** sub-flow).
                - **Search** sub-flow:
                    - The System asks the actors to enter Search Query to search for the various Courses/Tutorials provided by the System.
                    - The System validates the Search Query and then take the users to the Search Result.
                    - The Actor can then select from the Courses/Tutorials according to the Search Results.
        - Alternative Flow:
            - Unavailable Courses/Tutorials
                - If the actors provide invalid Course/Tutorial or a something that doesn't exist in the **Search** sub-flow, then the user is displayed with an error message of "No Courses/Tutorials found". The actor can now either
    - Pre-conditions:
        - The actors must be logged in the System before they can use this Module or Sub-Process.
    - Post-conditions:
        - If the use case is successful the user can now access the Specific Course/Tutorial he/she selected by the **View Course/Tutorial** Module or Sub-Process of the System.
    - Extension:
        - **View Course/Tutorial** Use Case.
- **View Course/Tutorial:**
    - Description: This use case lets the actor access a certain Course/Tutorial provided by the System

- Actors: User
- Flow of Events:
    - Basic Flow:
        - The actors have the access to Use Case after they perform the **Browse Courses/Tutorials** successfully
        - The System shows the actors the Course/Tutorial Information and asks actors to perform a specific function according to which a sub-flow will be executed.
            - If the actors select "Show Lectures" function, **Show Lectures** sub-flow is executed.
            - If the actors select "Show Problem Sets" function, **Show Problem Sets** sub-flow is executed.
            - If the actors select "Forum" function, **Forum** sub-flow is executed.
            - If the actors select "Manage Course/Tutorial", **Manage Course/Tutorial** sub-flow is executed.
        - **Show Lectures** sub-flow
            - The System displays all the Lecture of the Course/Tutorial
            - The actors can click (or select) on any Lecture to watch them.
        - **Show Problem** sub-flow
            - The System displays all the Problem Sets of the Course/Tutorial if any exists.
            - The actors can click (or select) any Problem Set to view it.
        - **Forum** sub-flow
            - The System lets actors perform QA.
                - The System asks actors that asks Questions the following information:
                    - Question
                    - Description
                - The System asks actors that Answer questions raised by other actors the following information:
                    - Answer.
        - **Manage Course** sub-flow
            - The System provides the ability to *add/remove* Course/Tutorial to the User Profile.
    - Alternative Flow:
        - If the actor enters Invalid data in the **Forum** sub-process, he/she will get an error message and the sub-process will start from the beginning.
- Pre-conditions:

        - The actors must be logged in the System and the use case must have the **Browse Course/Tutorial** resulting data before they can use it.
    - Post-conditions:
        - If the use case is successful than the actors have access to the Lectures, Problem Set provided by the System. They also have ability to add/remove a Course/Tutorial from their Profile and also have access to the Forum of the Course/Tutorial provided by the System.

- **Course Management**
    - Description: This use case describes how an actor manages Course/Tutorial Provided by the System. (Add, Remove, Modify).
    - Actors: Administrator, [YouTube, OCWs, Websites]
    - Flow of Events:
        - Basic Flow:
            - The System lets the actors to Add and Modify Courses/Tutorials.
            - The System asks the actor to select a specific function according to which a sub-flow is executed.
                - If the actor selects "Add Course/Tutorial by YouTube Playlist" function, **Add Course by YouTube Playlist** sub-flow is executed.
                - If the actor selects "Add Courses/Tutorials by YouTube Channel" function, **Add Course by YouTube Channel** sub-flow is executed.
                - If the actor selects "Add Course/Tutorial by Manually" function, **Add Course/Tutorial by Manually** sub-flow is executed.
                - If the actor selects "Modify Course" function, **Modify Course** sub-flow is executed.
            - **Add Course by YouTube Playlist** sub-flow
                - The System asks the actor to enter *valid YouTube Playlist URL.*
                - The System then extracts the Data from the entered URL and add the Course/Tutorial to the Course/Tutorial Database.
            - **Add Courses by YouTube Channel** sub-flow
                - The System asks the actor to enter *valid YouTube Channel URL.*
                - The System than extracts Data from the entered URL and adds all the Playlist from the Channel as Courses/Tutorials to the Course/Tutorial Database.
            - **Add Course/Tutorial by Manually** sub-flow
                - The System asks the actor the following information:
                    - Course/Tutorial Name

- Description
- Lectures
- Problem Set (if exists)

- The System validates the information entered by the User and then adds Course/Tutorial to the Courses/Tutorials Database.

- **Modify Course** sub-flow
    - The System first asks the actor to select a Course/Tutorial from the Courses/Tutorials that the System has
    - After selecting a Course/Tutorial the System asks the actor the following information:
        - Course/Tutorial Name
        - Description
        - Lectures
        - Problem Set (if exists)
    - On validating the data entered by the actors the selected course by the actor is modified.

- Alternative Flow:
    - Invalid URL
        - If the actor enters *invalid* data/information in the **Add Course by YouTube Channel, Add Courses by YouTube Channel** sub-flow, the System displays an Error Message.
        - The sub-flow is restarted
    - Cancellation
        - If the actors wish to cancel the any sub-flow the use case will be restarted.
    - Invalid Information
        - If the actor enters Invalid Information in **Add Course Manually, Modify Course** sub-flow, then use case is restarted.

- <u>Pre-conditions:</u>
    - The actor must be logged in to the System and must be an **Administrator** before he/she can access/use this sub-process or module of the System.
- <u>Post-conditions:</u>
    - If the use case is successful than the actor has Added or Modified the Courses/Tutorial Databases.

3. <u>Performance Requirements</u>
    a. The System shall be Operational for at least 90% of the day.
    b. The System shall process 90% of the transaction to the Database in at-most 2-4 seconds.

   c. The System shall load time of different Screens of the System should be at-most 5-9 seconds.
   d. The System shall support at least 500 concurrent User for Version 1.
   e. The System shall be scalable in future.
   f. The System shall try to use as minimum resources.

4. Design Constraint

   All the Development norms of the Interfaces specified in the Software Interfaces, Hardware Interfaces and Communication Interfaces should be followed.

5. System Attributes
   a. Reliability
      i. The System shall keep the Database Information consistent across all User.
      ii. The System shall be able to keep User Session i.e., they should not be asked to Log In each time they visit the System.
      iii. The System shall display Error Messages whenever an Error or Failure occurs to keep the User/Admin informed about what went wrong.
   b. Security
      i. The Password of the User should be hashed before storing them in the Database. A Proper Salt should be selected for Hashing.
      ii. User shall not be able to access the Modules of the System for the Admin
   c. Maintainability

      The System shall have a Maintenance Mode where the whole System goes down for Maintenance Purpose. The User shall not be able to access the System during the Maintenance Mode and shall be informed before the System goes in Maintenance Mode. The Maintenance Task performed during this period shall not break the system and the changes performed shall be reflected after the end of Maintenance Mode.

6. Database Design Constraint

   The Following Schemas will be stored in the Database:
   a. Playlist – For Storing Course/Tutorial Information.
   b. User – For Storing User Information.