MOOCs RECOMMENDER BASED ON LEARNING STYLES

Software Requirement Specification

Project ID: 18-036

Saugat Aryal

Porawagama A.S

Hasith M.G.S

Thoradeniya S.D

Bachelor of Science Special (Honors) in Information Technology

Specializing in Software Engineering

Department of Software Engineering

Sri Lanka Institute of Information Technology

Sri Lanka

Date of Submission: 2018-05-15

MOOCs RECOMMENDER BASED ON LEARNING STYLES

Project ID: 18-036

Saugat Aryal

Porawagama A.S

Hasith M.G.S

Thoradeniya S.D

Supervisor: Mr. Nuwan Kodagoda

Co-Supervisor: Ms. Kushnara Suriyawansa

Date of Submission: 2018-05-15

DECLARATION

We hereby declare that the project work entitled "MOOCs Recommender Based on Learning Styles" submitted to the Sri Lanka Institute of Information Technology, is a record of original work done by our group under the guidance of Mr. Nuwan Kodagoda (Supervisor) and Ms. Kushnara Suriyawansa (Co- Supervisor), and this project work is submitted in the fulfillment for the award of the Bachelor of Science (Special Honors) in Information technology Specialization in Software Engineering. The results embodied in this report have not been submitted to any other University or Institute for the award of any degree or diploma. The diagrams, research results and all other documented components were developed by us and we have cited clearly any references we have made.

Name	Student ID	Signature
Saugat Aryal	IT 14 146602	
Porawagama A.S	IT 14 142024	
Hasith M.G.S	IT 14 140280	
Thoradeniya S.D	IT 14 138232	

Supervisor's Name: Mr. Nuwan Kodagoda Signature:

Co-Supervisor's Name: Ms. Kushnara Suriyawansa Signature:

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

1.1 Purpose

The purpose of this document is to present a detailed description of requirements for our research project system, MOOCs Recommender Based on Learning Styles. The document will explain the purpose and features, the interfaces, what the system will do and the constraints under which it must operate. In general, it provides a complete and comprehensive description of requirements for the explained solution.

This document is mainly targeted to project supervisor, co-supervisor and the research team members. The document is written in a form that any person can read and understand the content. Also, it will be useful for former researches who are interested to implementing this kind of applications.

1.2 Scope

This document is intended for both stakeholders and the developers of the system. This document covers the requirements for the initial release of the proposed “MOOCs Recommender Based on Learning Styles” application. Our research project is based on correlating two concepts: Learning Styles and MOOCs. Identifying the learning style of learner based on Felder and Silverman Learning Style Model and mapping it with the MOOCs to recommend the most suitable courses, is the prime objective of the system. Besides that, the system can also be used as an advanced search engine for courses based on standard filters like topics, keyword, accent of speaker and transcript complexity. Above all, we intend to provide a unified educational platform where the learner can easily access his/her personalized learning materials.

1.3 Definitions, Acronyms and Abbreviation

MOOCs	Massive Open Online Courses
FSLSM	Felder and Silverman Learning Style Model
ILS	Index of Learning Style
SRS	Software Requirement Specification
NLP	Natural Language Processing
MEAN	MongoDB, Express, Angular and Node JS
OS	Operating System

1.4 Overview

The final product is targeted to be used by any learner who wants to learn computer programming through MOOCs. The learners can use the system to find the most appropriate and suitable MOOC courses based on their learning styles.

This SRS document intends to cover all the functional and non-functional requirements of the proposed system. Each of them has been discussed clearly in detail. All are described under three chapters. The first chapter provides an overall description of the component. The purpose and scope of the SRS is also mentioned.

The second chapter will provide an in-depth overview of the functionalities as focusing on users. The section will focus on comparing the software application to the existing systems in the market. It then moves on to explain several interfaces, constraints and operation of users to provide the reader with a better perspective of understanding the product. The chapter then explains about the summary of major functions, characteristics of users, constraints of system, assumptions and dependencies and finally conclude with apportioning of requirements.

The third chapter includes the specific requirements of the system and will be written primarily for developers. This section will describe the technical aspects and in-depth detail of the functionalities mentioned in the previous chapter. Both sections of the document describe the same software product in its entirety but are intended for different audiences and thus use different languages. The document will conclude with providing any supporting information regarding the content of the document.

2 OVERALL DESCRIPTIONS

Massive Open Online Courses (MOOCs) have taken a giant leap in the field of e-Learning, specifically open education. Over the years, various platforms such as edX[1], Coursera[2], Udacity[3], Futurelearn[4] etc. have emerged with the intent of providing massive educational resources to its users. One of the major and fundamental component in MOOCs is the video lectures and its production style. There are standard styles, like talking head, presentation slides with voice-over, animation, screencast etc. that are most commonly used. Along with the video lectures, transcripts, quizzes and other learning resources are also made available to help in the learning process.

In the recent years, there has been increasing attention towards the characteristics of learners such as learning styles. Different learners have the ability to learn in different ways and hence poses their own style of learning. Because of this behavior, the learner mostly explores through different MOOCs platform to find the most optimal learning resources that best fit their needs, preferences and learning style. Until now, no system has been developed that takes learner's learning style into consideration when recommending the best available MOOCs across different platforms.

Hence, we will be focusing on combining two different aspects: learning styles and MOOCs. Our sole purpose throughout the research project is to integrate both the factors together and develop an educational platform where learners from any domain can benefit from personalized learning services.

[1] https://www.edx.org/
[2] https://www.coursera.org/
[3] https://www.udacity.com/
[4] https://www.futurelearn.com/

2.1 Product Perspective

MOOCs have evolved as big players in the field of online learning and its unique characteristics makes it an effective Technology-Enhanced Learning (TEL) model in the modern era of education and technology. Lots of prominent research have been carried out considering the state-of-art and describing an overview of MOOCs [1][2]. According to the survey from Class Central [3], until 2017, around 81 million students are registered worldwide, where 23 million were new learners that signed up only in 2017. Similarly, the total number of MOOCs surpassed 9400 contributed from over 800 universities in 2017 [3]. This exponential growth rate is promising to believe that MOOCs are a new revolution that can help people distributed all over the world to gain access to diversified quality education in a more convenient way.

Learning style refers to the way a learner receives and processes information [4]. Therefore, different learners have different learning styles [4]. Considering learners learning style when designing a course has been found effective and shown positive results. It has been stated that providing learners with learning materials and activities that suit their preferences and learning style makes learning easier for them [5][6]. More evidence to support the statement is provided by studies which showed students can achieve better learning outcomes and higher scores [7], and can also master the course in less time [8].

Several studies have proposed to integrate learning style into the open learning environment (MOOCs), to provide adaptive and personalized support for learning[6], [9], [10]. Other studies reveal the use of data mining and machine learning algorithms to automatically identify the learners' learning styles. However, there has been no significant research to support the direct mapping of a specific learning style with MOOC based on characteristics such as, video lecture production style, available quizzes, resources etc.

Researches regarding MOOCs search engine and recommenders are also being carried out at a rapid rate. A recommender system using Case Based Reasoning (CBR) approach is proposed in [11]. User's query is described by five attributes where each attribute is assigned a weight value based on user's preference. "Courducate" is another system proposed for personalized search engine with two functionalities : multi-site search and multi-filed search [12]. Besides using BM25 ranking function, a novel ranking function is used to rank the sites upon query. A different approach is taken in [13] where the authors proposes to associate MOOCs with learning outcomes. Hence, allowing learners to discover the most suitable MOOCs for their learning objectives. [14] proposes two contributions: Using attribute and attribute value weight of resources to get specific user preferences; A new algorithm to overcome the shortcomings of the Collaborative Filtering (CF) and provide more accurate personalized recommendations on MOOCs. In the similar context, Content based and collaborative filtering recommendation approaches are used to accommodate several undergraduate characteristics when recommending MOOCs [15].

Some existing products available in the market are discussed below:

1. **Class Central**

Figure 1: Class Central

It[16] is one of the most popular search engine and reviews site for MOOCs. It also provides various statistics or MOOC report on yearly basis. It allows filtering on factors such as: start date, language, certificate allowed, university and provider. It provides courses from 41 providers and 811 universities. Users can create account and are also recommended courses based on their activities and history.

2. **My Mooc**

Figure 2: MyMooc

My Mooc[17] is an international platform which several thousands of free courses available in French, English and Chinese. With more than 450 course providers listed, it helps to find the best courses according to the learners' needs. As an European leader for online courses browsing, it maintains user profile, badges and leaderboard tables. It also allows filtering on basis of factors like category, duration, language, accessibility etc.

3. MoocLab

Figure 3: MoocLab

MoocLab[18] is a free community website which provides forums to connect about online learning and offers suggestions, help, reviews and guidance with online learning. Besides serving as a search engine for MOOCs it mainly acts as an e-Learning "hub" where people could interact about online learning in a centralized place. It provides unique features of finding a study buddy and jobs with MOOC-friendly employers.

4. Coursetalk

Figure 4: Coursetalk

Coursetalk[19] is another platform for MOOCs search engines. It is a well-organized website with thousands of courses from more than 30 subjects to choose from. Like other systems, users can create account and get recommendations for their courses of preference. It provides feature of course tracker which allows users to keep tabs of all their courses – past, present and future all in one convenient location.

From the literature survey and research activities carried out, we discovered that there is quite a research gap that needs to be addressed. The following table shows a comparison of features between existing products and the proposed solution.

Table 1: Comparison of existing products in market

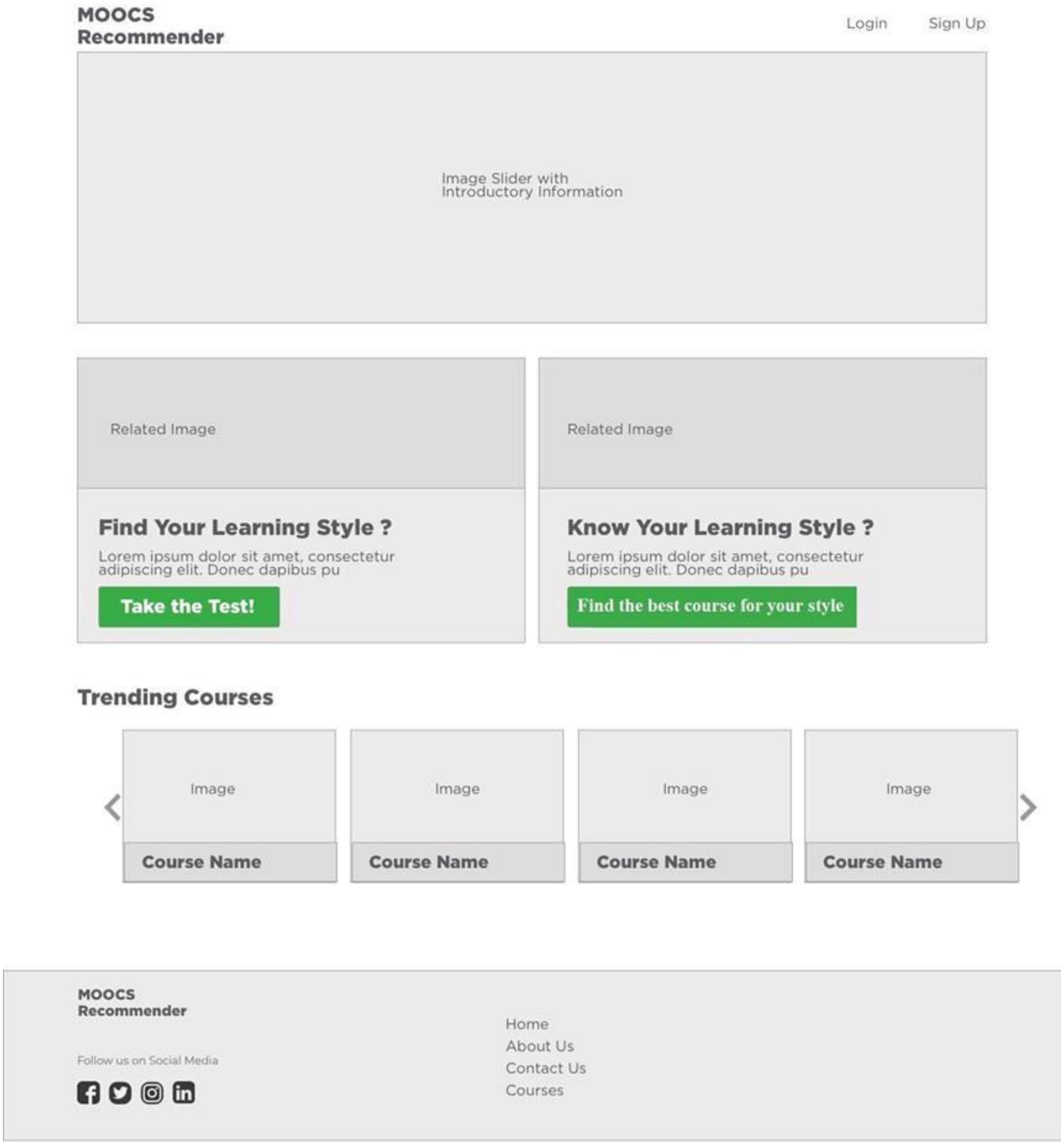
Features	Class Central	MyMooc	MoocLab	Coursetalk	Proposed solution
Learner’s Learning Style	✗	✗	✗	✗	✓
Video Production Style of MOOC videos	✗	✗	✗	✗	✓
Search filter based on English accent of speaker	✗	✗	✗	✗	✓
Search filter based on specific keywords / topics	✗	✗	✗	✗	✓
User Profile and Dashboard	✓	✓	✓	✓	✓

2.1.1 System Interfaces

- PyMongo Interface to connect MongoDB database and Python backend
- Web-Desktop Connectivity

2.1.2 User Interfaces

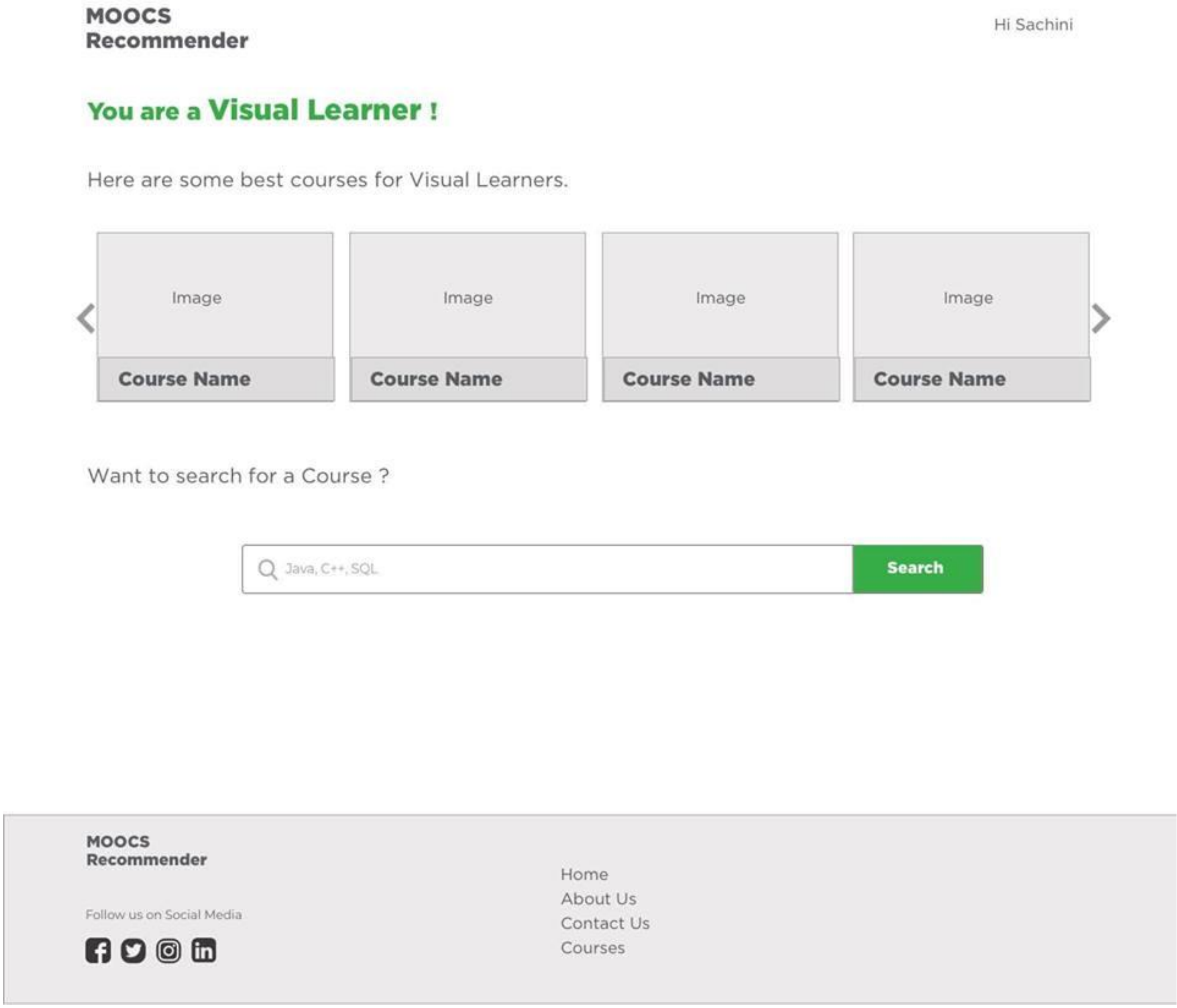
Sample User Interfaces of the system are shown below:



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec dapibus pu

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec dapibus pu

Figure 5: Home Screen



MOOCS Recommender

Hi Sachini

You are a **Visual Learner !**

Here are some best courses for Visual Learners.

Want to search for a Course ?

Figure 6: Learning Style Screen

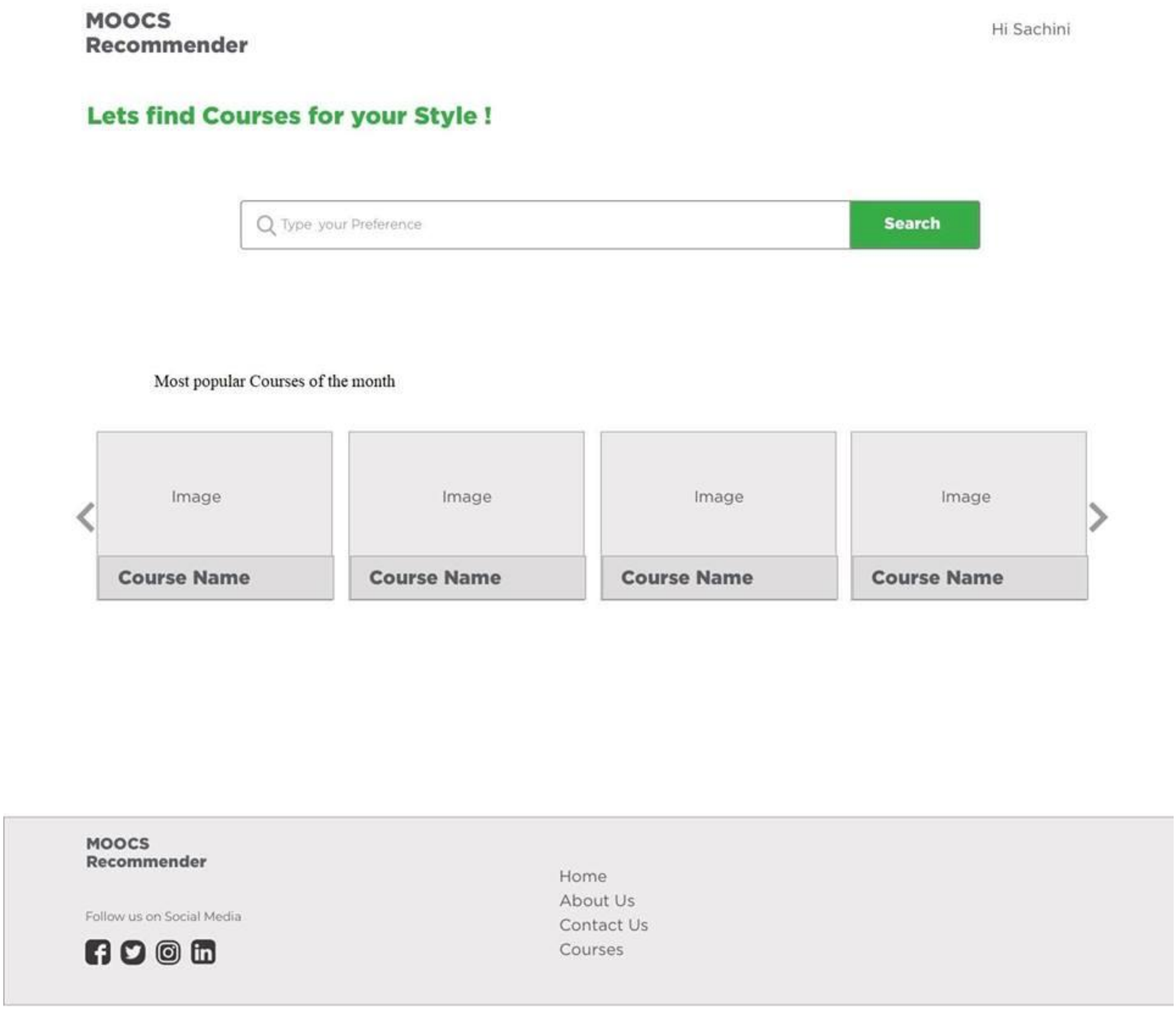
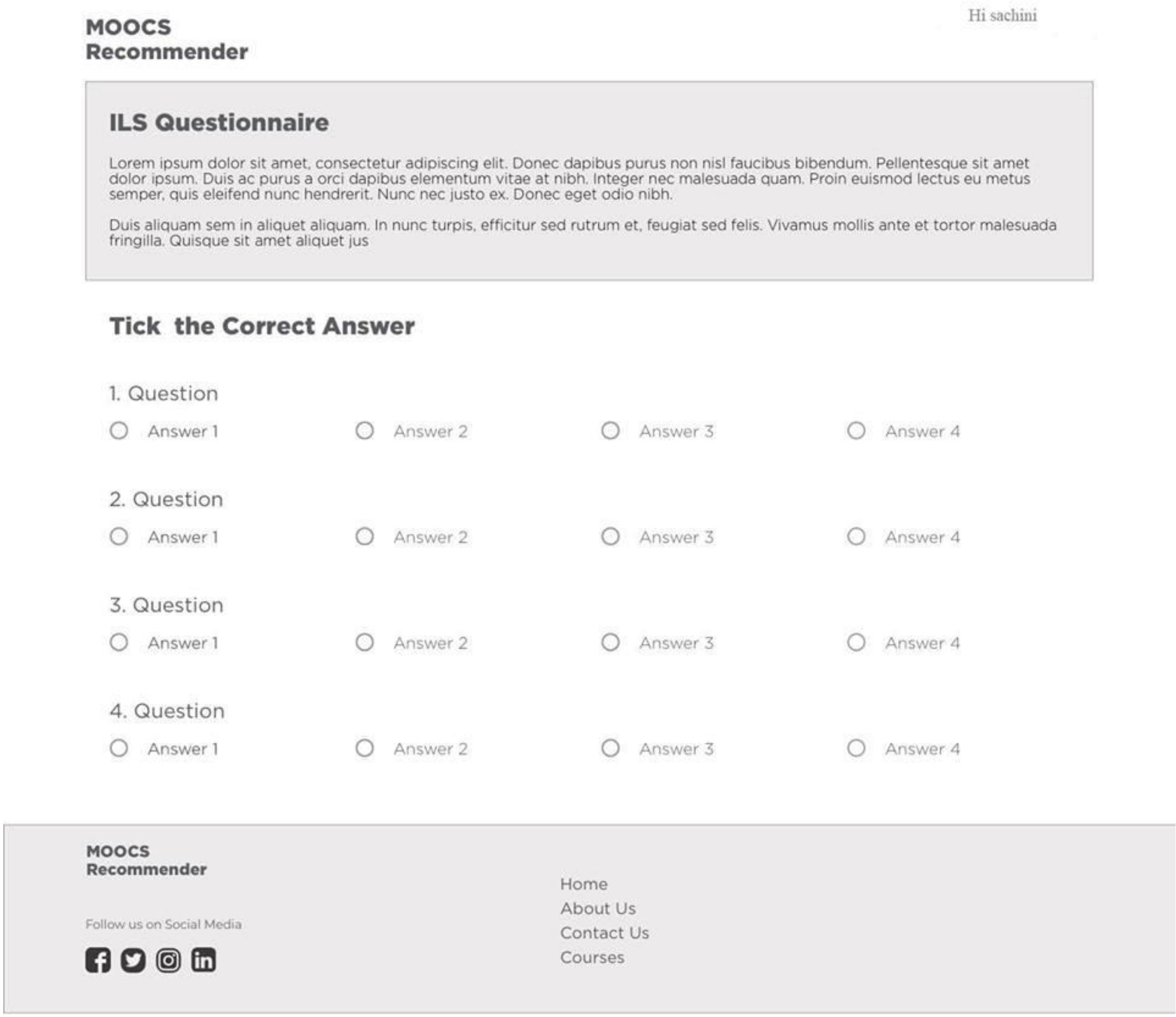


Figure 7: Search Screen



MOOCS
Recommender

Hi sachini

ILS Questionnaire

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec dapibus purus non nisl faucibus bibendum. Pellentesque sit amet dolor ipsum. Duis ac purus a orci dapibus elementum vitae at nibh. Integer nec malesuada quam. Proin euismod lectus eu metus semper, quis eleifend nunc hendrerit. Nunc nec justo ex. Donec eget odio nibh.

Duis aliquam sem in aliquet aliquam. In nunc turpis, efficitur sed rutrum et, feugiat sed felis. Vivamus mollis ante et tortor malesuada fringilla. Quisque sit amet aliquet jus

Tick the Correct Answer

1. Question

○ Answer 1 ○ Answer 2 ○ Answer 3 ○ Answer 4

2. Question

○ Answer 1 ○ Answer 2 ○ Answer 3 ○ Answer 4

3. Question

○ Answer 1 ○ Answer 2 ○ Answer 3 ○ Answer 4

4. Question

○ Answer 1 ○ Answer 2 ○ Answer 3 ○ Answer 4

MOOCS
Recommender

Follow us on Social Media

Home
About Us
Contact Us
Courses

Figure 8: ILS Questionnaire

2.1.3 Hardware Interfaces

No special hardware devices are needed other than the usual PC or Laptop.

2.1.4 Software Interfaces

- Google Colaboratory
- Jupyter Notebook
- Google Tensorflow
- Keras – Deep Learning Library
- MongoDB and AngularJS
- PhpStorm

2.1.5 Communication Interfaces

The user's laptop/desktop should support high speed data communication methods such as 3G (HSPDA) or 4G (LTE).

2.1.6 Memory Constraints

- RAM of 6 GB or higher

2.1.7 Operations

- Users should have access to internet to use the system.
- Login to the system and fill the ILS questionnaire to identify the learning styles.
- Users can take the MCQ test to get to know their knowledge and understanding in a programming language.
- They can search for custom MOOCs based on various filters as required.

2.1.8 Site Adaption Requirements

Since the user can be of any nationality the user interface must be created for English language.

2.2 Product Functions

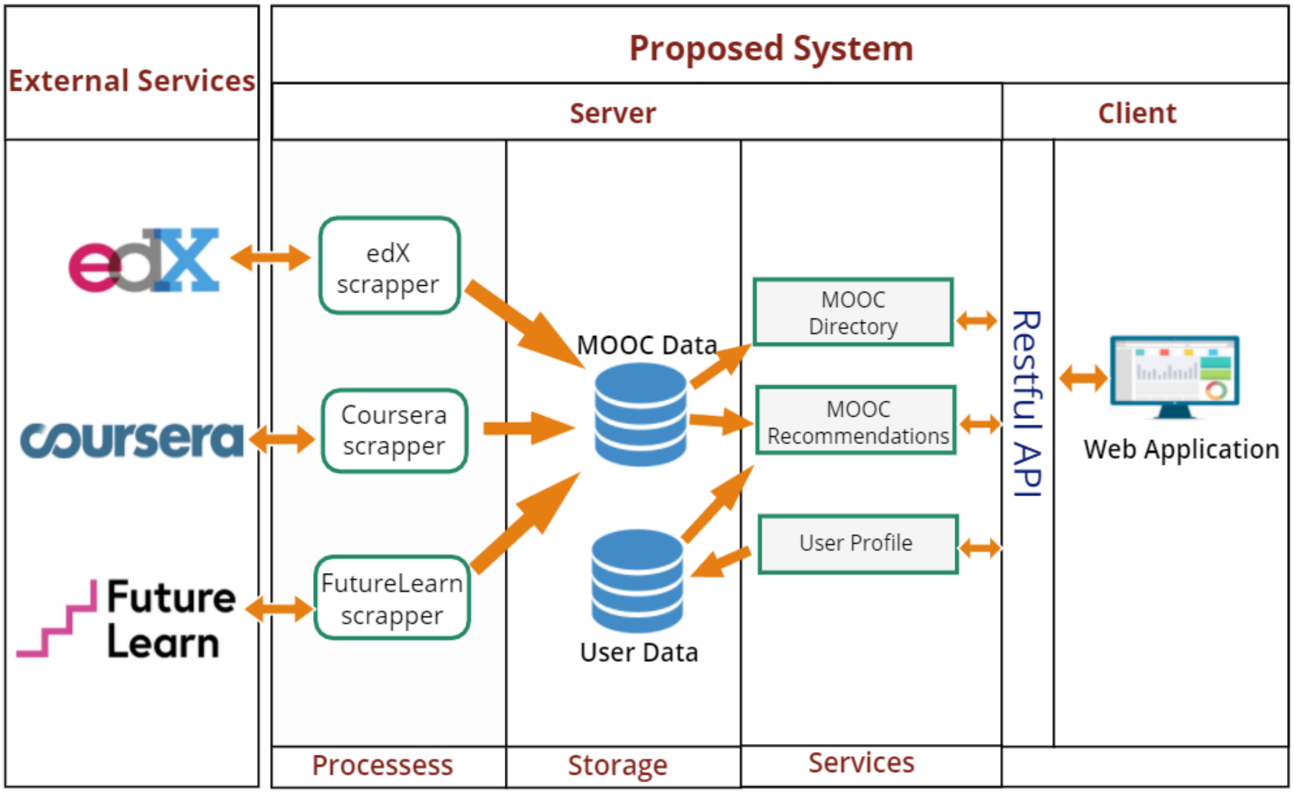


Figure 9: High Level System Architecture

The four distinct modules of the system are discussed as follows:

1. Web Scraping, Crawling and Audio Classification
   - Develop different web scraper to extract MOOCs information from different platforms: edX, Coursera, Futurelearn.
   - Implement a web crawler to periodically scan through given platforms to check for new courses.
   - Classify the audio of a given MOOC video as native or non-native English speaker.

2. Topic Modeling and Transcripts Complexity
    - For a given MOOC video, perform topic modelling to discover the abstract topics for the specific video.
    - Calculate the transcript complexity (lexical and syntactic).

3. Video Style Classification
    - Split a given MOOC video into set of frames and classify each frame into a specific video style category.
    - Identify what percentage of each video style is contained in a single MOOC video.
    - Calculate the aggregate composition of the video styles for the overall course.

4. Learner Profile Modeling and Recommendation
    - Mapping/Correlating MOOCs with the Learning Styles by comparing the characteristics of both aspects based on literature review.
    - Implement the search engine based on standard filters such as, specific keyword, accent, transcript complexity etc.
    - Natural Language Processing (NLP) based searching of MOOCs.

2.2.1 Use Case Diagram

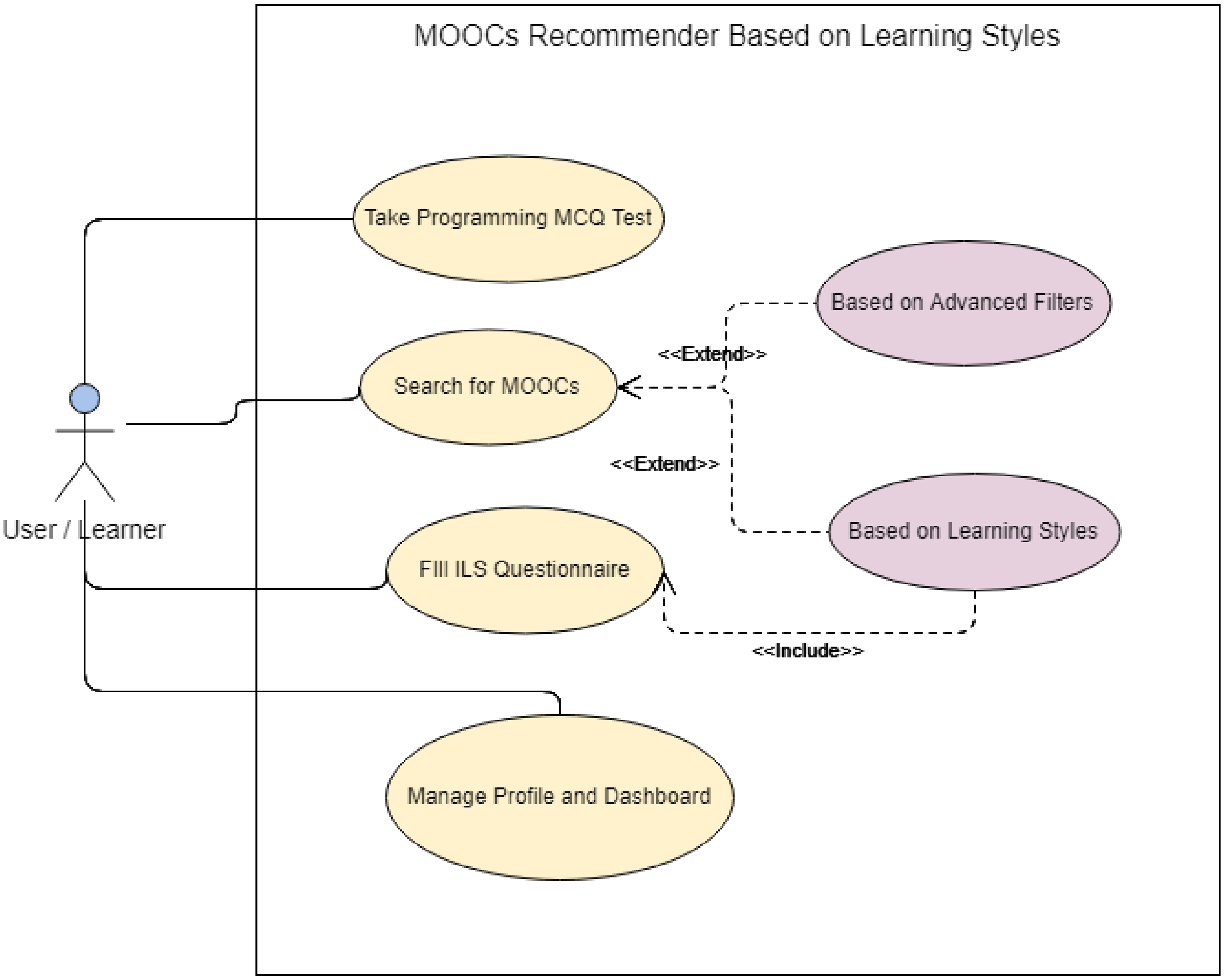

MOOCs Recommender Based on Learning Styles
Take Programming MCQ Test
Search for MOOCs
Based on Advanced Filters
<<Extend>>
<<Extend>>
Based on Learning Styles
User / Learner
FIll ILS Questionnaire
<<Include>>
Manage Profile and Dashboard

Figure 10: Use Case Diagram

2.2.2 Use Case Scenarios

Use case Name	Fill the ILS Questionnaire
Pre –Condition	User must be logged in to the system
Post-Condition	User identifies his/her learning style
Actor	User (Learner)
Main Success Scenarios	1. Use case starts with user opening the application. 2. User selects the type of learning path from the home screen. 3. User chooses “Through Learning Style” option and clicks “Next”. 4. System loads the ILS questionnaire to the screen. 5. User fills out the questions as required and click on “Finish”. 6. The system evaluates the answers and evaluate the score of learning style. 7. The use case ends with the system successfully displaying his/her learning style on the screen.
Extension	5a. User do not fill all questions. 5b. The system prompts message box to fill all the required questions.

Table 2: Use Case Scenario 1

Use case Name	Get Recommended Courses
Pre –Condition	Fill out ILS Questionnaire
Post-Condition	List of Recommended MOOCs
Actor	User(Learner)
Main Success Scenarios	1. Use case starts with the learner logging in to system. 2. He/She fills out the questionnaire and identifies the learning style. 3. The user clicks on “Next”. 4. The user then selects the type of course from dropdown and clicks “Show Courses”. 4. Based on the learning style, the system shows list of recommended courses to the user in a ranked order. 5. The use case ends with user clicking on any of the suggested courses.

Table 3: Use Case Scenario 2

Use case Name	Search for Custom Courses
Pre –Condition	User must be logged in to the system
Post-Condition	List of MOOCs
Actor	User (Learner)
Main Success Scenarios	1. Use case starts with user opening the application. 2. User logs in to the system by filling the details and click “Login”. 3. User selects the type of learning path from the home screen. 4. User chooses the option “Search Custom MOOCs” and

	clicks “Next”. 5. User selects the type of course, any specific keywords, accent and other available filters and clicks “Show Courses”. 6. The system evaluates the user’s request and displays the most relevant courses in a ranked order. 7. The use case ends with the user selecting any of the courses or clicks on “Start Again”.
Extension	5a. User do not select any course. 5b. The system prompts message box to choose the course from the dropdown options. 6a. There are no matches for any courses based on user’s request. 6b. The system prompts message “No Courses Found”

Table 4: Use Case Scenario 3

Use case Name	Take Programming MCQ Test
Pre –Condition	User must be logged in to the system
Actor	User (Learner)
Main Success Scenarios	1. Use case starts with user logging in the system. 2. On the left hand sliding menu, the use selects “Take Test”. 3. The user then selects the type of programming course from the dropdown options and clicks “Take Quiz”. 4. User takes the quiz filling out all the questions given and clicks “Sow Result”. 5. The system evaluates the user’s answers and displays the score on the screen along with the knowledge level, such as Beginner, Intermediate or Advanced. 7. The use case ends with the user clicking on “Finish”.

Table 5: Use Case Scenario 4

2.2.3 Activity Diagram

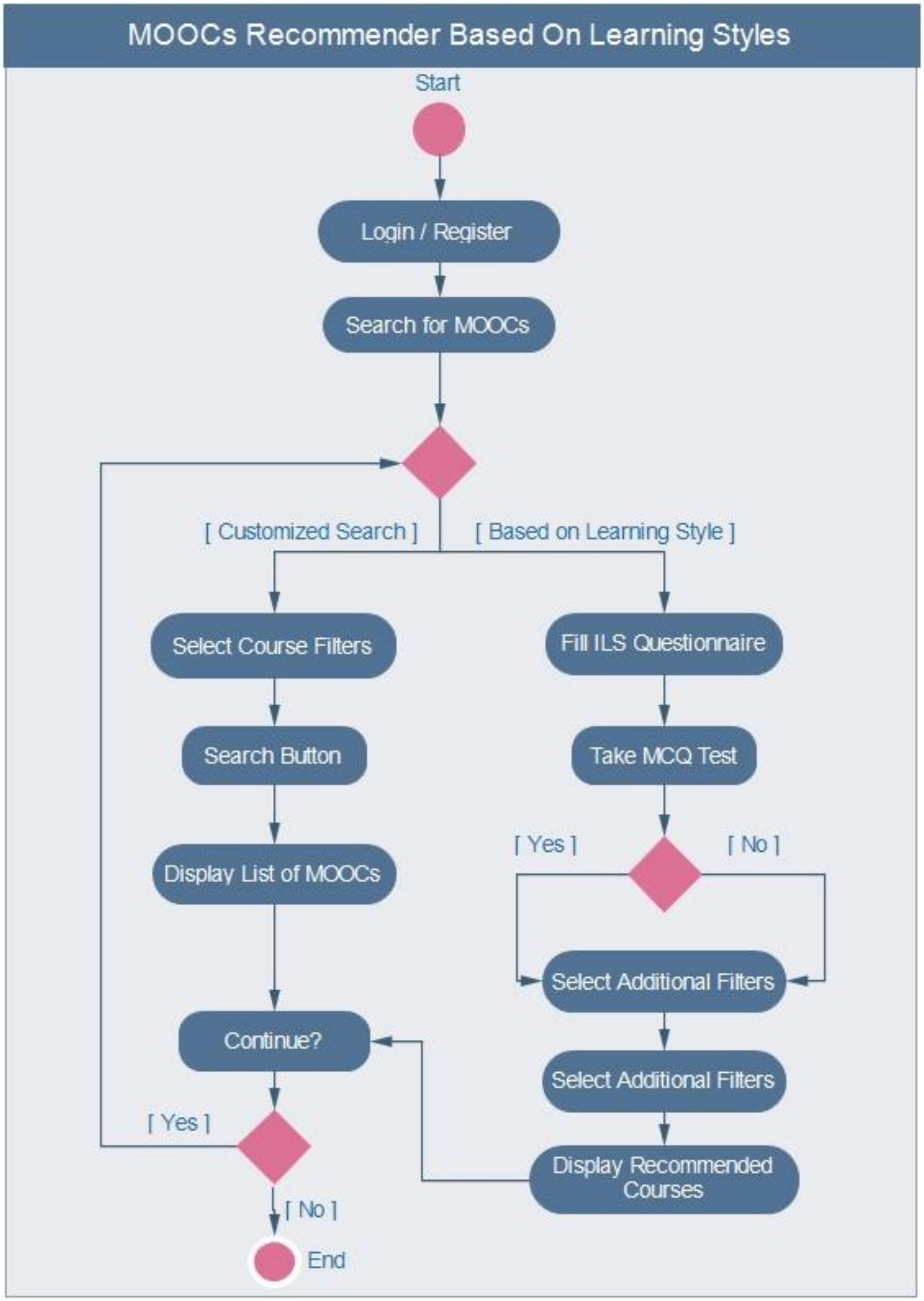


Figure 11: Activity Diagram

2.3 User Characteristics

The objective of the system is to provide personalized learning resources to any learner. The learner can be anyone who wants to learn computer programming. They can be either a student, lecturer, novice, software professional etc. The system is open to learners of all kind irrespective of their fields. There are no restrictions to have any specific technical knowledge to use the system besides general interactions with a web application.

2.4 Constraints

The proposed system is a web application, consisting of frontend client application and backend process. Backend system performs all the complicated and heavy computational tasks while the client application only interacts with the user and performs operations as requested. The constraints of system are discussed below:

- Internet connectivity is a must to interact between the frontend and backend processes.
- Only the reference and link to the recommended MOOC courses are displayed in the system. The learner must go to the original platform to take the course.
- Backend system must have minimum of 6GB RAM. Additional 2GB of GPU will be advantageous. Client's PC can have the usual specifications.
- MEAN stack is used to develop frontend client application while the backend processes are developed using Python and machine learning frameworks.
- After the MOOC videos are extracted by the scraping tool at the backend, they are used to perform video classification and transcript analysis. After performing the required operations and storing the necessary data in the database, the video files are removed from the system. It helps to optimize the performance.

2.5 Assumptions and Dependencies

- The operating system for the client application is preferred to be Windows. However, it can run on any OS with any browser.
- It is assumed that the internet connectivity can be established with ease on request.

2.6 Apportioning of Requirements

Essential Requirements:

1. Allow learner to fill the ILS Questionnaire to find out about their learning style.
2. Allow learner to search for courses based on their learning patterns.
3. Allow learner to search for custom MOOCs based on keyword and topics filters.
4. Allow learner to take the programming standard MCQ test to determine their knowledge and understanding.
5. Allow learner to track their history and manage their dashboard.
6. Allow learner to filter MOOCs based on accent of the speaker and transcript complexity.

Desirable Requirements:

1. Generate statistics of video styles of programming courses across various platforms.
2. Generate statistics report of learner's preferences of courses based on their search history.
3. Maintain learner profile based on their activities while interacting with the system.

Optional Requirements:

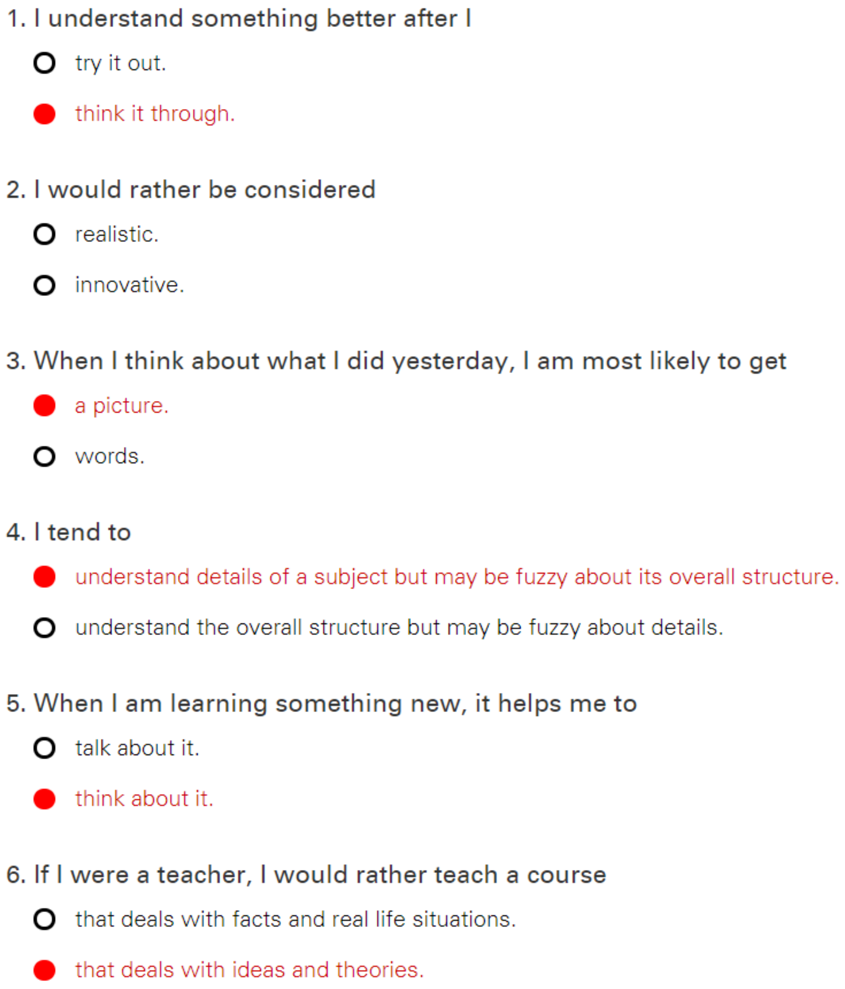
1. Allow learners to search for MOOCs related to other fields such as science, business etc.
2. Chatbot integrated within the application to interact with the learner regarding any queries.

3 SPECIFIC REQUIREMENTS

3.1 External Interface Requirements

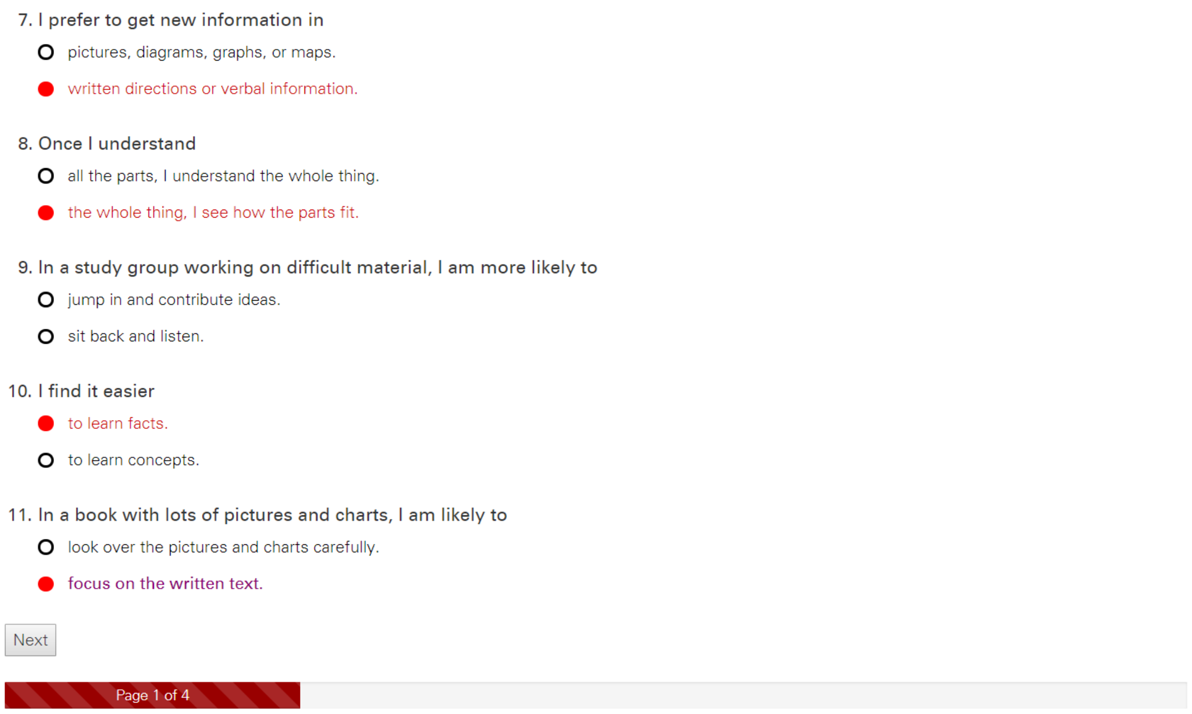
3.1.1 User Interfaces

Sample user interfaces are shown above in Section 2. These interfaces show how the learner can search for courses based on his/her learning style or also based on other preferences and filters. It also shows recommended courses to the user based on the learning style. Similarly, interface forthe ILS questionnaire is also shown. Below show some sample ILS questions and MCQs to test the proficiency in different programming languages.



1. I understand something better after I
   - ○ try it out.
   - ● think it through.
2. I would rather be considered
   - ○ realistic.
   - ○ innovative.
3. When I think about what I did yesterday, I am most likely to get
   - ● a picture.
   - ○ words.
4. I tend to
   - ● understand details of a subject but may be fuzzy about its overall structure.
   - ○ understand the overall structure but may be fuzzy about details.
5. When I am learning something new, it helps me to
   - ○ talk about it.
   - ● think about it.
6. If I were a teacher, I would rather teach a course
   - ○ that deals with facts and real life situations.
   - ● that deals with ideas and theories.

Figure 12: ILS Sample Questionnare-1



7. I prefer to get new information in

○ pictures, diagrams, graphs, or maps.

● written directions or verbal information.

8. Once I understand

○ all the parts, I understand the whole thing.

● the whole thing, I see how the parts fit.

9. In a study group working on difficult material, I am more likely to

○ jump in and contribute ideas.

○ sit back and listen.

10. I find it easier

● to learn facts.

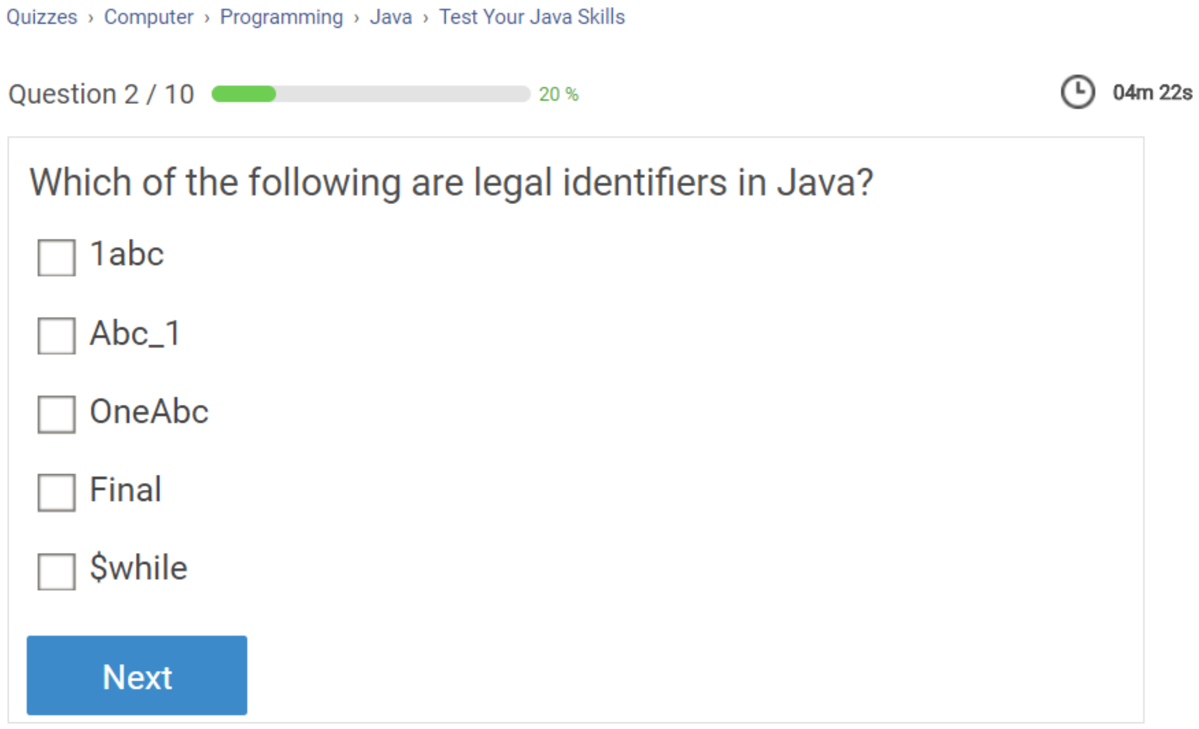
○ to learn concepts.

11. In a book with lots of pictures and charts, I am likely to

○ look over the pictures and charts carefully.

● focus on the written text.

Figure 13: ILS Questionnaire Sample -2



Quizzes › Computer › Programming › Java › Test Your Java Skills
Question 2 / 10
20 %
04m 22s

Which of the following are legal identifiers in Java?

□ 1abc

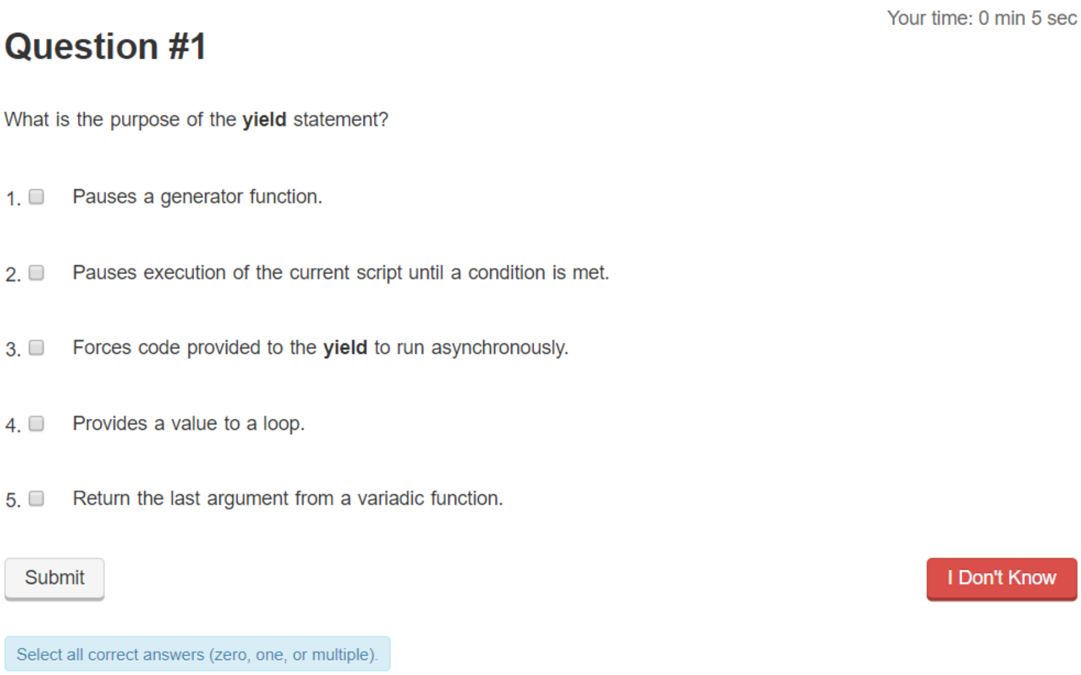
□ Abc_1

□ OneAbc

□ Final

□ $while

Figure 14: Sample Java MCQ


Your time: 0 min 5 sec

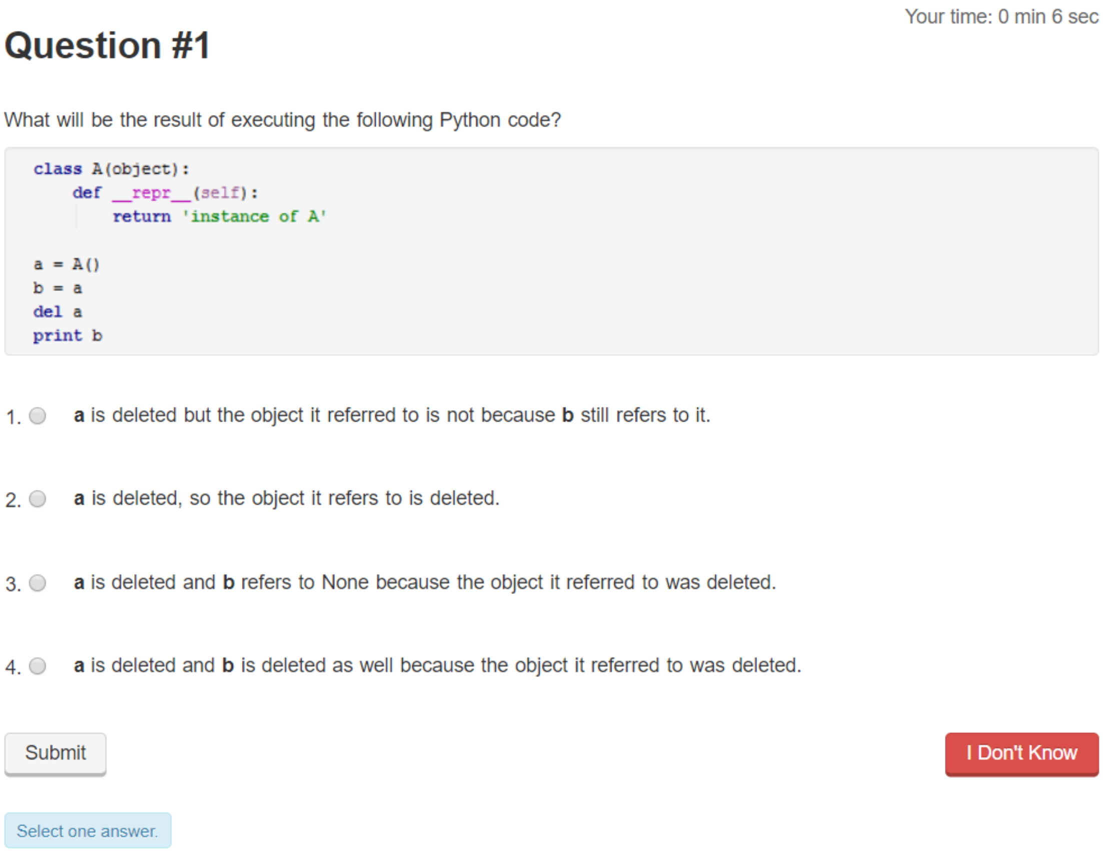
Question #1

What is the purpose of the **yield** statement?

1. □ Pauses a generator function.
2. □ Pauses execution of the current script until a condition is met.
3. □ Forces code provided to the **yield** to run asynchronously.
4. □ Provides a value to a loop.
5. □ Return the last argument from a variadic function.

Select all correct answers (zero, one, or multiple).

Figure 15: Sample PHP MCQ


Your time: 0 min 6 sec

Question #1

What will be the result of executing the following Python code?

```
class A(object):
    def __repr__(self):
        return 'instance of A'

a = A()
b = a
del a
print b
```

1. ○ **a** is deleted but the object it referred to is not because **b** still refers to it.
2. ○ **a** is deleted, so the object it refers to is deleted.
3. ○ **a** is deleted and **b** refers to None because the object it referred to was deleted.
4. ○ **a** is deleted and **b** is deleted as well because the object it referred to was deleted.

Select one answer.

Figure 16: Sample Python MCQ

3.1.2 Hardware Interfaces

- Graphics Processing Unit (GPU) – It can be used to speed up the computational tasks of training a neural network at the backend. It is all about leveraging the parallelism to perform mathematically compute intensive operations that CPU is not designed to handle. Training a CNN model can take days and even weeks when carried out in normal CPU, hence GPU can help to reduce the heavy load.
- PC with normal specifications such as 2-4 GB RAM, Intel Core Processor is required to access the application through internet browser.

3.1.3 Software Interfaces

- Google Colaboratory: It is free cloud service offered by Google that provides free GPU. That distinctive feature makes it different than other cloud services. Using this service, users can develop deep learning applications using popular libraries such as, Keras, TensorFlow, OpenCV etc.
- TensorFlow: It is a computational framework for building machine learning models. It provides wide variety of toolkits that allows the users to construct models at the preferred level of abstraction. It comes in-built with Google Colab.
- Keras: It is a very popular deep learning library written in Python. We can quickly build and test a neural network with minimal lines of code. It is capable of running on top of TensorFlow and also available in Google Colab.
- MongoDB and AngularJS: They are JavaScript platforms to develop modern web applications. MongoDB is selected as the database because of the structure of the data that will be stored. Angular JS is mostly used to develop SPA (Single Page Applications) and we will be using the similar approach for our web pages.
- RESTful API: The frontend application communicates with the backend process using the principle of REST services. It has grown quite popular over the years and proved to be a popular choice for implementing Web Services.

3.1.4 Communication Interfaces

The backend service along with the database is hosted on the cloud. Similarly, the frontend client application is also deployed in the web. Hence, the communication between them requires internet connectivity. The connection can range from minimum of 3G to fast 4G (LTE).

3.2 Classes/Objects

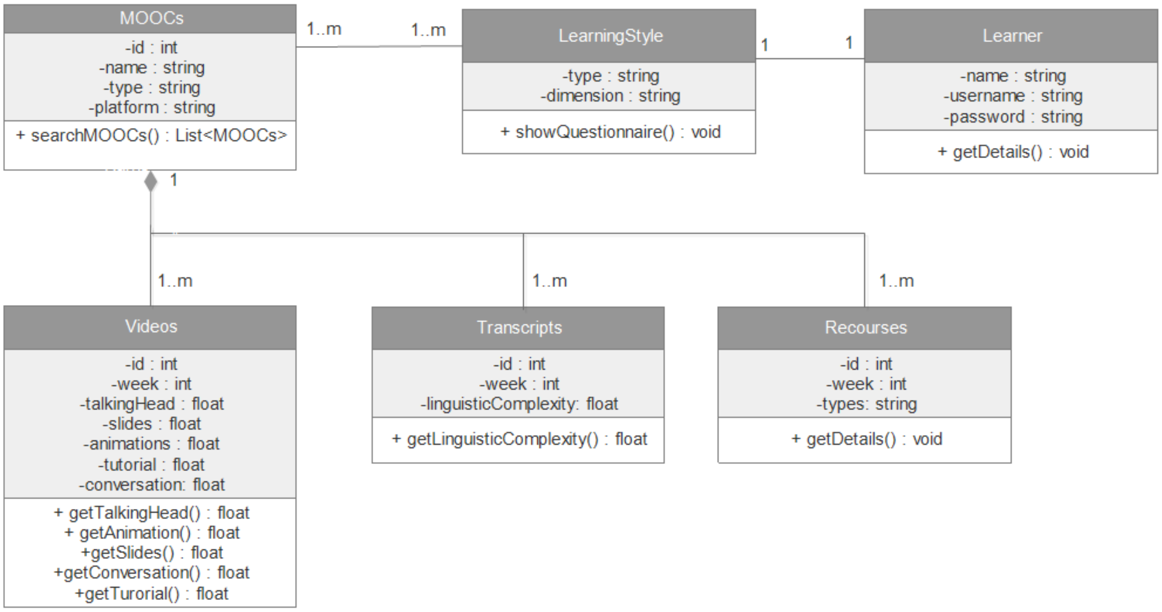


Figure 17: Class Diagram

3.3 Performance Requirements

It is expected that the proposed system will perform all the requirements stated under the product functions section. Some performance requirements identified are listed below:

- The system should be able to accommodate a minimum of 50,000 records in the database.
- Recommended Courses are displayed to the user in no more than 5 seconds upon any query.
- The results of ILS Questionnaire and Programming MCQ test should be displayed within 3 seconds of completion.
- Search query for any personalized courses should not take more than 5 seconds.
- The system should support more than 100 simultaneous users and their requests.

3.4 Design Constraints

- MOOC videos related to three platforms: edX, Futurelearn and Coursera are considered.
- Only computer programming courses are taking into perspectives.
- Felder and Silverman Learning Style Model is adopted to map the learning styles and MOOCs based on literature review.

3.5 Software System Attributes

3.5.1 Reliability

All the major operations of the system are performed in the backend including mapping of MOOCs with learning styles, storing MOOCs characteristics in database, ILS Questionnaire results, Programming MCQ test etc. The client application basically queries to the database to get the desired results. Hence, considering all the four different modules perform their responsibility successfully, the overall should be reliable enough (around 99%) to complete its sole objective without much failure.

3.5.2 Availability

As the backend process is hosted on the cloud server, the downtime of the system solely depends upon the cloud provider’s services and performance. However, the system should be available more than 95% of the time. There will be a huge system load to the backend as new course videos are made available. However, the system is designed to carry out the tasks simultaneously such that the workload on the server remains minimal and it is up and running most of the time. The frontend client application is also deployed in the web. But, since it does not carry any heavy computational tasks, it is expected to be available almost all the time.

3.5.3 Security

The system stores the user’s information regarding the learning styles and the knowledge of different programming languages. Even though, the details stored are not much sensitive, they should be secured enough to not get exploited by any third-party applications. Encryption methods are taken into consideration to maintain the confidentiality of the information. User needs to log in to the system with their credentials before using the system services. Passwords will be encrypted before saving to database.

3.5.4 Maintainability

The system is split into four different modules independent of each other. The functions that are expected to change soon or frequently are:

- New MOOCs statistics and characteristics are stored in the database once in a week.
- Out-dated courses are removed from the database once in a year.
- Upon arrival of new video styles, their features are mapped with the MOOCs characteristics and result is stored.

3.6 Other Requirements

- Extensibility and Modifiability: System should be able to accept and accommodate new features and customizations with ease.
- Accessibility: The user interface of the system should be simple, easily accessible and understood by all the users.

4 REFERENCES

[1] M. H. Baturay, “An Overview of the World of MOOCs,” *Procedia - Soc. Behav. Sci.*, vol. 174, pp. 427–433, Feb. 2015.

[2] A. Mohamed, F. Yousef, M. A. Chatti, U. Schroeder, M. Wosnitza, and H. Jakobs, “A Review of the State-of-the-Art.”

[3] “By The Numbers: MOOCS in 2017 — Class Central.” [Online]. Available: https://www.class-central.com/report/mooc-stats-2017/. [Accessed: 26-Mar-2018].

[4] R. M. Felder, R. M. Felder, and L. K. Silverman, “Learning and teaching styles in engineering education,” *Eng. Educ.*, 1988.

[5] S. Graf and T.-C. Liu, “Supporting Teachers in Identifying Students’ Learning Styles in Learning Management Systems: An Automatic Student Modelling Approach,” *Educ. Technol. Soc.*, vol. 12, no. 4, pp. 3–14, 2009.

[6] C. Li and H. Zhou, “Enhancing the Efficiency of Massive Online Learning by Integrating Intelligent Analysis into MOOCs with an Application to Education of Sustainability,” *Sustainability*, vol. 10, no. 2, p. 468, Feb. 2018.

[7] N. Bajraktarevic[1], W. Hall[1], and P. Fullick[2], “Incorporating learning styles in hypermedia environment: Empirical evaluation.”

[8] K. K. Graf Sabine, “Providing adaptive courses in learning management systems with respect to learning styles,” in *E-Learn: World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education. Association for the Advancement of Computing in Education (AACE)*, 2007, pp. 2576–2583.

[9] H. A. Fasihuddin, G. D. Skinner, and R. I. Athauda, “Boosting the Opportunities of Open Learning (MOOCs) through Learning Theories,” *GSTF J. Comput.*, vol. 3, no. 3, p. 31, Dec. 2013.

[10] H. A. Fasihuddin, G. D. Skinner, and R. I. Athauda, “Personalizing Open Learning Environments through the adaptation to Learning Styles.”

[11] F. Bousbahi and H. Chorfi, “MOOC-Rec: A Case Based Recommender System for MOOCs,” *Procedia - Soc. Behav. Sci.*, vol. 195, pp. 1813–1822, Jul. 2015.

[12] Q. Cheng and Y. Gao, “Courducate --An MOOC Search and Recommendation System.”

[13] C. D.-K. Israel Gutiérrez-Rojas, Derick Leony, Carlos Alario-Hoyos, Mar Pérez-Sanagustín, “Towards an Outcome-based Discovery and Recommendation of MOOCs

using moocrank," in *European MOOCs Stakeholders Summit*, 2014.

[14] Y. Wang, B. Liang, W. Ji, S. Wang, and Y. Chen, "An improved algorithm for personalized recommendation on MOOCs," *Int. J. Crowd Sci.*, vol. 1, no. 3, pp. 186–196, Sep. 2017.

[15] D. Fu, Q. Liu, S. Zhang, and J. Wang, "The Undergraduate-Oriented Framework of MOOCs Recommender System," in *2015 International Symposium on Educational Technology (ISET)*, 2015, pp. 115–119.

[16] "Class Central • #1 Search Engine for Free Online Courses &amp; MOOCs." [Online]. Available: https://www.class-central.com/. [Accessed: 03-Apr-2018].

[17] "Browse the best MOOC and free online trainings in the world - My Mooc." [Online]. Available: https://www.my-mooc.com/en/. [Accessed: 03-Apr-2018].

[18] "Home | MoocLab - Connecting People to Online Learning." [Online]. Available: https://www.mooclab.club/. [Accessed: 03-Apr-2018].

[19] "Student reviews of online courses on Computer Science, Business, Design, Data Science, Humanities and more | CourseTalk." [Online]. Available: https://www.coursetalk.com/. [Accessed: 03-Apr-2018].

5 APPENDIX