**PLATFORM FOR IMPROVING SEARCHABILITY AND INTERACTIVITY OF RECORDED LECTURES**

Project ID: 19-087

Project Proposal Report

Karunaratne D. C.

Fernando S. S. M. S.

40 mm

20 mm

20 mm

25 mm

Hettiarachchi H. A. I. S.

Epa S. S.

Bachelor of Science Special (Honors) in Information Technology

Specializing in Software Engineering

Department of Software Engineering

Sri Lanka Institute of Information Technology

Sri Lanka

March 2019

**PLATFORM FOR IMPROVING SEARCHABILITY AND INTERACTIVITY OF RECORDED LECTURES**

Project ID: 19-087

Project Proposal Report

40 mm

20 mm

20 mm

25 mm

Bachelor of Science Special (Honors) in Information Technology Specializing in Software Engineering

Department of Software Engineering

Sri Lanka Institute of Information Technology

Sri Lanka

March 2019

**DECLARATION**

We declare that this is our own work and this proposal does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

|  |  |  |
| --- | --- | --- |
| Name | Student ID | Signature |
| Karunaratne D. C. | IT 16037434 |  |
| Hettiarachchi H. A. I. S. | IT 15146366 |  |
| Fernando S. S. M. S. | IT 16001862 |  |
| Epa S. S. | IT 16009646 |  |

The above candidates are carrying out research for the undergraduate Dissertation under my supervision.

Signature of the supervisor: Date:

Signature of the co-supervisor: Date:

**Abstract**

E-Learning has become commonplace and many leading Universities provide the facility to view pre-recorded lectures online. This approach gives learners the ability to follow lectures without time or location constraints and consume the lectures at their own pace. Despite their advantages, recorded lectures tend to be lengthy and tedious to watch. They also prove cumbersome when specific information needs to be extracted from them. Another drawback is that the lecture videos fail to show the connection between the lecture and its supporting material such as lecture slides and questionnaires.

Several platforms exist where videos can be edited to make them more interactive, however this is a time-consuming process. We propose a system which will automatically improve the interactivity and accessibility of recorded lectures in a few clicks. The system will take in raw lecture videos along with supporting material such as lecture slides and code samples. It will then carry out noise removal and optimizing on the raw video footage before matching the slides and code samples to occurrences in the video. Some novel features we plan on introducing are automatic generation and suggestion of questions based on the content and automatic video segmentation according to topics. The main objective of the system is to create a web platform which can add interactivity and accessibility to course material thereby improving learner engagement.

**TABLE OF CONTENTS**

[1 INTRODUCTION 1](#_Toc4232623)

[1.1 Background 1](#_Toc4232624)

[1.2 Literature Survey 2](#_Toc4232625)

[1.3 Existing Products 8](#_Toc4232626)

[1.3.1 LearnWorlds 8](#_Toc4232627)

[1.3.2 Echo360 8](#_Toc4232628)

[1.3.3 TechSmith Relay 9](#_Toc4232629)

[1.4 Research Gap 9](#_Toc4232630)

[1.5 Research Problem 10](#_Toc4232631)

[2 OBJECTIVES 11](#_Toc4232632)

[2.1 General Objective 11](#_Toc4232633)

[2.2 Specific Objectives 11](#_Toc4232634)

[3 METHODOLOGY 12](#_Toc4232635)

[3.1 System Design 12](#_Toc4232636)

[3.2 System Description 12](#_Toc4232637)

[3.2.1 Matching each line of code in a sample code file to its occurrence in a live-coding video 12](#_Toc4232638)

[3.2.2 Content-based Lecture Video Segmentation into topic units 14](#_Toc4232639)

[3.2.3 Automatic Question Generation from Lecture and Reference Materials 15](#_Toc4232640)

[3.2.4 Synchronization of slides with the lecture video and de-noising 16](#_Toc4232641)

[3.3 Software Development Life Cycle 18](#_Toc4232642)

[3.4 Gantt Chart 19](#_Toc4232643)

[3.5 Work Breakdown Structure 20](#_Toc4232644)

[4 BUSINESS POTENTIAL 21](#_Toc4232645)

[5 DESCRIPTION OF PERSONAL AND FACILITIES 22](#_Toc4232646)

[6 References 24](#_Toc4232647)

**LIST OF FIGURES**

[Figure 1.1: LearnWorlds 8](#_Toc4232648)

[Figure 1.2: echo360 8](#_Toc4232649)

[Figure 1.3: TechSmith Relay 9](#_Toc4232650)

[Figure 3.1: High-level system architecture 12](file:///C:/Users/USER/Documents/SLIIT/CDAP/Documents/Project_Proposal_Draft_1.docx#_Toc4232651)

[Figure 3.2: High-level system diagram of code matching process 13](#_Toc4232652)

[Figure 3.3: High-level system diagram of lecture video segmentation process 14](file:///C:/Users/USER/Documents/SLIIT/CDAP/Documents/Project_Proposal_Draft_1.docx#_Toc4232653)

[Figure 3.4: High-level system diagram of automatic question generation process 15](#_Toc4232654)

[Figure 3.5: High level diagram of slide matching process 16](#_Toc4232655)

[Figure 3.6: High level diagram of audio de-noising process 17](file:///C:/Users/USER/Documents/SLIIT/CDAP/Documents/Project_Proposal_Draft_1.docx#_Toc4232656)

[Figure 3.7: Agile Scrum Process 18](#_Toc4232657)

[Figure 3.8: Gantt Chart 19](file:///C:/Users/USER/Documents/SLIIT/CDAP/Documents/Project_Proposal_Draft_1.docx#_Toc4232658)

[Figure 3.9: Tasks schedule 19](file:///C:/Users/USER/Documents/SLIIT/CDAP/Documents/Project_Proposal_Draft_1.docx#_Toc4232659)

[Figure 3.10: Work Breakdown Structure 20](#_Toc4232660)

**LIST OF TABLES**

[Table 1.1: Comparison of existing products 9](#_Toc4232661)

[Table 5.1: Description of Personal and Facilities 23](#_Toc4232662)

**LIST OF ABBREVATIONS**

|  |  |
| --- | --- |
| CNN | Convolutional Neural Network |
| IDE | Integrated Development Environment |
| LMS | Learning Management System |
| MCQ | Multiple Choice Question |
| MOOC | Massive Open Online Course |
| NLP | Natural Language Processing |
| OCR | Optical Character Recognition |
| SaaS | Software as a Service |
| TDT | Topic Detection and Tracking |

# INTRODUCTION

## Background

Today, e-learning has become an essential component of higher education for both teachers and students. According to a study on the effectiveness of e-learning on education, it was found that students nowadays are more satisfied with web enhanced learning when compared to a traditional classroom environment [1]. Therefore, it is common to see universities and higher education institutes adopting some form of e-learning to assist their students. Many institutes use their own customized version of a Learning Management System (LMS) to provide online course material.

Online education is beneficial for both students and teachers in many ways. For instance, as lecture content is always available online, the possibility of missing a lecture is low and teachers can ensure that students have access to course material irrespective of time and location [1]. Recorded lecture videos also enable students with different styles of learning and different levels of understanding to obtain a better grasp of the subject. For example, those who are familiar with the work can skip ahead to a section of interest while those that need more time to understand the concepts can pause and rewind to digest the lecture at their own pace [2, 3].

Usually, many LMSs enable lecturers to upload course material such as tutorials, lab sheets, lecture slides and recorded lecture videos. Whilst videos are more effective because they address both visual and auditory aspects of teaching [4], many students find it tedious to watch recorded lecture videos because of its duration, which normally lasts around 1 - 3 hours and its lack of interconnectivity and relevance to other course material [5]. Although there are many video-creation and editing platforms which allow users to create interactive course material, these methods are time consuming as they require the lecturer to spend a lot of time editing and making the video interactive.

At the time of writing this proposal, to the best of our knowledge, there is no system which automatically identifies the relationship between different types of course material and enables the creation of interactive courses in a few steps. Hence, our main focus will be on improving two aspects of course material: accessibility and interactivity. Our goal is to develop an intelligent system capable of improving the interactivity and learner engagement of course material in just a few clicks.

## Literature Survey

The following section will review existing literature on the individual research areas assigned to each team member. Four areas will be examined.

1. Matching each line of code in a sample code file to its occurrence in a live-coding video.
2. Content-based Lecture Video Segmentation into topic units
3. Automatic question generation from lecture and reference materials
4. Synchronization of slides with the lecture video and denoising

**Matching each line of code in a sample code file to its occurrence in a live-coding video**

Live coding lectures, often seen in the field of Information Technology, usually feature a screen capture in which the instructor types code into a text editor or Integrated Development Environment (IDE) while narrating. A study conducted by Marc. J. Rubin, on the effectiveness of live-coding lectures, found that students exposed to live-coding lectures performed better when tackling large programming assignments [6].

Although lectures of this type have their merits, students often need to revisit a specific point in a lecture and to do this they must search through the video until the relevant section is found. This has been identified as a drawback and many researchers have tried to address this problem with varying degrees of success. In a research conducted in 2011, Kambathula and Iyer suggested a system which would enable automatic tagging of lecture videos to enable easy identification of the different sections. It achieves this by first performing text analysis on the audio extracted from the video and then creating a database of tags from the resulting transcript [7]. Their system can highlight portions in each video in response to user queries, allowing the user to navigate to an exact location in the video. Luca Ponzanelli *et al* in 2016 introduced CodeTube [8], a similar search engine which when given a query, returns self-contained fragments of the corresponding lecture videos. Their system can identify Java code in video frames by applying image processing techniques such as Optical Character Recognition (OCR) and shape detection as well as text analysis methods such as island parsing on subsections of each frame to generate a Heterogenous Abstract Syntax Tree (H-AST) which is used to identify coding constructs [9]. In a research conducted in 2018, a deep-learning approach which leverages Convolutional Neural Networks (CNNs) to classify the presence or absence of Java code in video frames is proposed [10]. Their system is able to achieve an average accuracy of 98% for this binary classification task using a trained VGG16 [11] Neural Network and represents a more scalable solution to identifying code in videos. However it is limited to identifying code in the Java language and cannot be successfully applied in a system which would analyze videos in many programming languages. Research on algorithms such as ResNet50 [12] and InceptionV3 [13] prove they are good candidates for this purpose.

As illustrated above, much research has been conducted on identifying source code in video and image files. Although the topic area (matching each line of code in a sample code file to its occurrence in the lecture video) has not been widely researched and would most likely have an algorithmic solution, research that has already been conducted on source code mining and text detection can be used as a basis to create a system which identifies relevant portions of a live-coding video which correspond to the source code file.

**Content-based Lecture Video Segmentation into topic units**

Segmentation of long lecture videos into cohesive topic units is highly beneficial since it makes the search for information easier, minimizes learning time and improves the overall learning experience. Research has been carried out in relation to segmenting videos based mainly on three areas - visual content, audio and text (transcripts). However, limited research has been conducted in relation to the specific domain of segmenting lecture videos into topic units.

The most commonly used mechanism for segmenting videos is based on scene changes. A research conducted in 2000 used color histogram distance computation between successive images to detect scene changes [14]. Another research done by Zhang and Smoliar, proposed a system for detecting progressive transitions based on both motion and statistical analysis [15]. However, this mechanism of segmentation based on scene changes is not applicable to lecture videos, as they have very few scene changes and even these scene changes do not match with topic transitions [4].

Another feature considered by researchers for lecture video segmentation is slide matching. A research done in 2013, focused on segmenting lecture videos into topics by analyzing its supplementary synchronized slides. They used OCR to extract content from lecture slides and identify different subtopics according to their logical relevance. Slides were then synchronized with the video stream to identify different topic changes. As stated in their paper [16] the mechanism was approximately 90% accurate, except that it always assumes that the slides are synchronized with the video streams, which, in practice might not always be the case. Furthermore, since their method was solely based on matching slide content with the video, its accuracy is limited only for certain lecture video types.

Moreover, segmentation methods utilizing transcribed text or closed captions have also been researched on. The main motivational factor for work in this area was the Topic Detection and Tracking (TDT) initiative conducted in 1998 [14]. TDT is defined as the task of segmenting transcribed speech into topically cohesive stories. Their algorithm is trained mainly using broadcast and news domain data sets where, formal presentation format and cue phrases are used to improve segmentation accuracy. Unlike in the broadcast domain, speeches in lecture videos are often unscripted and spontaneous. Furthermore, a large training dataset is used for many methods in TDT, which is not available for lecture videos [4, 16, 17].

Inspired by the work on TDT, in 2004 Michael Chau, and his team members, conducted another research to identify topic changes based on multiple linguistic features like noun phrases, topic noun phrases, verb classes, word stems, combined features, cue phrases, and pronouns [4]. Using automatic speech recognition software, they retrieved timestamps that synchronize with the video stream and then the results from transcribed text segmentation is then mapped back to video segmentation. However, they only developed a set of algorithms each considering one specific feature out of the list of segmentation features mentioned above. They then compared the results from these algorithms to identify the most salient feature for lecture video segmentation. Work in this area can be considered as a potentially successful solution and can be improved further as audio and the transcribed text extracted from lecture videos provide rich content information for topic change detection [4].

**Automatic question generation from lecture and reference materials**

The main goal of any lecture is that the learner should achieve knowledge on a specific subject or area. This specific knowledge gained from a lecture is known as its learning outcome. One of the primary and most effective ways to evaluate whether the learner has achieved these learning outcomes is quizzes. However, formulation of questions for quizzes can be a time-consuming task.

Research has been carried out on ways to automatically generate questions. Shah et al suggest a method of generating Multiple Choice Questions (MCQs) for an input text passage with the aid of a one-time trained knowledge base developed using Wikipedia articles [18]. During the execution of the system words in sentences are mapped to a predefined dictionary and an Inverse Document Frequency score is used to choose the word that serves as a blank. Distractors are generated by the paradigmatic discovery on a self-made corpus and dictionary.

TEDQuiz [19] is a system that generates MCQs for TED Talks video clips using a graph-based algorithm. The system generates two types of questions. The first type is gist-content questions which ask about the overall theme of the content that is generated by identifying the most important sentence using LexRank [20] and creating distractors by less important sentences. The second type is detailed questions which use Heilman and Smith’s work [21, 22] to create question stem and selecting words for distractors using WordNet and similar corpus.

A similar system which analyzes a text transcript of a video lecture to suggest self-assessment items at runtime is seen in [23]. The process is carried out by identifying discourse boundaries from the lecture and retrieving Wikipedia text segments related to identified discourse boundaries for further well-formed and formal discourse to generate question items: MCQs generated using Heilman’s work [24] and distractors generated on an ontology-based strategy using Wikipedia category taxonomy as a replacement for the ontology.

Apart from this, automatic question generation was carried out using ontology-based strategies. SeMCQ [25] is a Protégé plugin created for automatic ontology-driven multiple question test generation. OntoQue [26] is an automatic question generation engine based on domain ontologies which can generate MCQ, true/false questions and fill-in-the-blank questions. Papasalouros et al. suggest an approach of generating MCQs based on domain-specific ontologies that use simple natural language generation techniques [27].

**Synchronization of slides with the lecture video and denoising**

Most of the time, lecturers upload PowerPoint presentations along with the lecture video for reference. For easy access, it is highly beneficial to have the ability to point out the occurrences where the slide is discussed in the lecture video, when a particular slide is selected rather than manually going through the entire lecture video to find out the occurrences in which the particular slide is discussed. In addition, removing unnecessary noises like the noise of breathing and time intervals with no audio would optimize the time taken for the learning process.

Several research projects have been carried out related to synchronization of lecture slides with the video. Among them, a research has been done by the Hongkong university and science technology and City university of Hongkong in which a system using OCR is proposed [28]. In the system, geometry-based approach for text detection which works well with images with less noise and Super Resolution Reconstruction approach is used to enhance the visual quality of the video texts to make the video images appropriate for commercial OCR systems.

Zentation [29] is an online tool that can be used to synchronize the slides with the video. But it is not a perfect tool when it comes to synchronization since most of the time this works well with presentations that have a smaller number of slides. Therefore, this kind of tool cannot be used for matching lecture videos with the slides as most of the times lecture slide decks contain a considerable number of slides.

When it comes to denoising audio enhancement studies have been carried out reduce the background noise and to enhance intelligibility [30, 31, 32, 33, 34]. Removal of high frequency noise for speech enhancement with frequency response masking (FRM) has also been implemented. A FIR filter has been designed to have impulse responses associated with various cut off frequencies to minimize the error when comparing the original speech signal and the filtered speech signal [35].

Audacity [36] is a tool that can be used to remove background noise of an audio and it can be used to enhance the audio quality also. It can be used to remove regular noises like static, hum, hiss and other constant background noises. In addition, this tool enables the user to change the pitch without any changes in the tempo. But Audacity is not capable of removing irregular noises like traffic sounds and sounds of an audience which can occur in lecture videos. As a result, tools like Audacity cannot be used to denoise lecture videos in e-learning platforms as it contains irregular noises rather than constant noises.

## Existing Products

### LearnWorlds



Figure 1.1: LearnWorlds

LearnWorlds [37] is a platform that enables users to create and sell online courses. It allows content creators (instructors) to upload video clips, documents, quizzes and other reference materials to an online course and customize by using drag and drop tools to show course outline, branding, etc.

### Echo360



Figure 1.2: echo360

Echo360 [38] is an active learning platform that analyses student engagement data and enables teachers to create and distribute videos. It has features for online and offline video recording as well as streaming. It also includes a built-in video editor. Echo360 provides auto transcribing for lecture videos with automatic speech recognition, linking class presentations and student note-taking to specific moments in the video. Major LMS platforms also can be integrated to Echo360.

### TechSmith Relay



Figure 1.3: TechSmith Relay

TechSmith Relay [39] is a video creation and sharing platform that maximizes student engagement in online courses. It allows instructors to create engaging and interactive videos using a web-based editor and integration with TechSmith Camtasia. Relay also provides methods to add captions and embed questions in the video. Integration with popular LMS is an added feature in TechSmith Relay.

## Research Gap

Even though there are several products already available with similar objectives, they mostly focus on the use of manual processes that involve human intervention to make the content more interactive and increase the searchability. Our proposed solution aims to reduce the amount of human interaction needed for this process by introducing a platform which will analyze and augment the content automatically. Table 1.1: Comparison of existing products is a comparison of the proposed system with existing systems in the market.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Features | LearnWorlds | Echo360 | TechSmith Relay | Our Solution |
| Matching lines in code samples to occurrence in recorded lectures | ✗ | ✗ | ✗ | ✔ |
| Automated segmentation of lecture video into topic units | ✗ | ✗ | ✗ | ✔ |
| Matching slides with the lecture video | ✗ | ✔ | ✗ | ✔ |
| Automated noise removal from the video | ✗ | ✗ | ✗ | ✔ |
| Automatic question generation | ✗ | ✗ | ✗ | ✔ |

Table 1.1: Comparison of existing products

## Research Problem

Nowadays, video lectures have become increasingly popular and many education institutes use Learning Management Systems that support video content. Whilst video lectures have benefits such as giving learners remote access to lectures, there are a few drawbacks such as poor searchability through the video and less interaction with the learner.

To overcome these drawbacks many lecture platforms have introduced tools such as web-based video editors that allow lecturers to add captions, divide the video into discussed topics, link lecture slides and embed questions into the video. However, these tools require human intervention which is time-consuming. When considering the domain of computer science, none of the available platforms provide a tool to map programming language code segments with their occurrences in the video.

Because there is a need for video lectures to be more interactive and searchable, but that a significant amount of time is taken by lecture creators to make them so, we conclude that a platform which will analyze and augment the content automatically will be useful.

# OBJECTIVES

The proposed system is a research study to improve the method of delivery for video lectures and increase the engagement of the learner. The objectives of the research are as follows.

## General Objective

To develop an automated platform which provides a quick and efficient way for lecture creators to deliver video lectures which are more interactive and have increased searchability.

## Specific Objectives

* Automatically tag specific points in the video which correspond to a given lecture slide or line of code in a given supporting code sample.
* Automatically identify main topic transitions of a given lecture video and segment the video according to the identified positions.
* Automatically generate question items related to the lecture video and present to end user.
* Automatically remove unnecessary background noises and enhance the quality of the audio.

# METHODOLOGY

## System Design

Figure 3.1: High-level system architecture



Unedited lecture video, slides and reference materials

Noise removal and optimizing

Optimized Video

Transcript

Code Matching

Question Generation

Slide Matching

Topic Segmentation

Metadata



Question Items

Proposed System

Processes

Database

Web-Platform

## System Description

### Matching each line of code in a sample code file to its occurrence in a live-coding video

For the purpose of this research we have decided to limit the scope to videos where i) code is typed incrementally on screen or ii) code is scrolled through and explained line by line. To match a code sample to its occurrence in the lecture video, text detection must be carried out on each video frame to extract a textual representation of it. However, existing research in the fields of OCR and Text detection suggests that it is more efficient to use a Machine Learning model to detect candidate frames before running the OCR algorithm [8, 10] instead of wasting CPU cycles on frames which do not contain code. Based on this research we propose to train a machine learning algorithm to detect such frames. Considering the time constraints of the project we will apply Transfer Learning techniques [40, 41] to repurpose an established machine learning model for this classification.

Once a video is uploaded, the system will first reduce the number of frames to be analyzed by sampling the video at a rate of one frame per second. The extracted set of frames will then be passed through the neural network so that candidate frames can be identified. Each candidate frame will be analyzed by an OCR tool to obtain a textual representation of the code contained within it. The extracted code will then be passed into an algorithm along with the source code file to identify a suitable timestamp in the video for each the frame in which the code occurs. Audio taken from the lecture can be used to make sure that the video is not indexed mid-sentence. Figure 3-2 illustrates this process.

A close up of a sign

Description generated with high confidence

Figure 3.2: High-level system diagram of code matching process

Measurable outcomes for this component are:

* Accuracy: accuracy for this component can be measured by comparing percentage of lines identified against a baseline determined by examining the video manually
* Speed: The time taken by the system to identify a line of code in the video can be compared with the average time taken to scroll through the video and find the lines manually.

### Content-based Lecture Video Segmentation into topic units

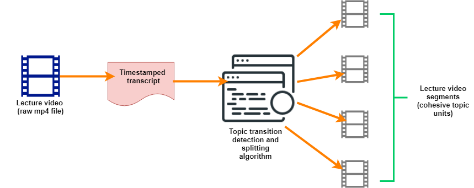
Out of the 3 different types of segmentation features available we choose the linguistic approach based on video transcripts. Video transcripts provide rich content information and will be ideal for detecting topic transitions. Also, generally computing audio and visual features is a very time-consuming process, while computing text takes comparatively less amount of time. In our approach we will extract time stamped transcripts from the raw mp4 video files and then these transcripts will be input into an algorithm, similar to the TextTilting approach [42], which is able to detect topic transitions based on linguistic features like noun phrases and word frequency. The identified timestamps where topic transitions occur will then be mapped back to the video for splitting. The challenging part in this component is to identify the topic transitions and extracting the transcription as accurately as possible even for domain specific words. The high-level view of the overall procedure is shown in **Error! Reference source not found.**

Figure 3.3: High-level system diagram of lecture video segmentation process

Measurable outcomes for this component are:

* Accuracy: accuracy for this component can be measured by comparing the number of topic transitions identified and the position of these transitions identified by the algorithm to that of the actual number of topic transitions and their positions which can be determined manually.
* Speed: Speed can be measured by the time taken for the whole segmentation process and can be compared with the time taken for an average human to perform segmentation manually.

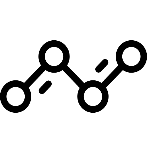
### Automatic Question Generation from Lecture and Reference Materials

For automatic question generation we use a strategy which combines NLP techniques and an ontology-based approach. Important sentences that can be used to generate factual questions and important word phrases will be extracted from a transcript of the video lecture and other provided reference materials for this process. Using the extracted sentences factual questions will be generated by applying Heilman’s work [21]. Domain specific ontologies will be used for distractor generation. Extracted word phrases will be mapped with the ontologies to identify knowledge represented by specific subject areas. This identified knowledge will be used to generate more questions based on the connectivity and instances presented in ontologies using an algorithmic method.



Transcript

Extracted Sentences and Words



Ontology

Reference materials

Question generation and distractor selection

Question Items

Figure 3.4: High-level system diagram of automatic question generation process

Measurable outcome of this component:

* Accuracy: Accuracy of automatically generated questions can be measured by validating number of valid questions (related to lecture content) against the number of all questions generated.

### Synchronization of slides with the lecture video and de-noising

In order to match the electronic slide with the relevant time stamp in the lecture video in which that particular slide is discussed, the proposed system will be using Optical Character Recognition (OCR). A Geometry based approach can be used to locate the exact text in the uploaded lecture video and the quality of the images can be enhanced using a Super Resolution Approach. In the algorithm, Laplacian of Gaussian (LOG) is used to detect the edges of the frames. It identifies the rectangular area covering the edges of the frames and an attribute set is recorded for each rectangle. This attribute set is used to identify the text regions and non-text regions. Video text is sent through an algorithm which will enhance the resolution. Before sending the text to the OCR system, the texts are binarized and the binarized text boxes are fed to the OCR system. The texts from slides and videos are separated into titles and content and matched through an algorithm which will compute the edit distance between two strings.

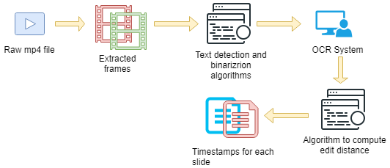


Figure 3.5: High level diagram of slide matching process

Measurable outcomes of this sub component:

* Speed: Speed of this component can be measured by the time taken by the proposed system to point out the place in the lecture video in which the selected slide is discussed. This can be compared with the time taken to locate the places in the video in which a selected slide is discussed manually.
* Accuracy: Accuracy of this component can be measured by the places pointed out by the system given a selected lecture slide to the actual frames in which the selected slide is discussed.

In the process of removing the unwanted background noises (denoising), the proposed system will use spectral subtraction. This algorithm will be similar to the one defined in [35] and will utilize Inverse Fast Fourier Transform (IFFT) to remove noise and reconstruct the result.



Figure 3.6: High level diagram of audio de-noising process

Measurable outcome of this component:

* Performance: Performance of this sub component can be measured using the Speech to Noise Ratio (SNR).

## Software Development Life Cycle

At the beginning of a research project requirements may be unclear or undefined and will be subjected to lot of changes throughout the development cycle. The Agile Scrum model is an ideal methodology for this kind of a project because it adds more flexibility to the software development life cycle and encourages requirement changes throughout the process of development. Agile follows an incremental and iterative development approach, and each iteration will focus on delivering a working product. As our team consists of four members, having daily scrum meetings will allow each member to have a general understanding of the whole project and be aware of problems faced by other members. Also, this will improve the collaboration between team members encouraging better team work. Figure 3-5 represents the scrum process.



Figure 3.7: Agile Scrum Process

## Gantt Chart

Figure 3.8: Gantt Chart

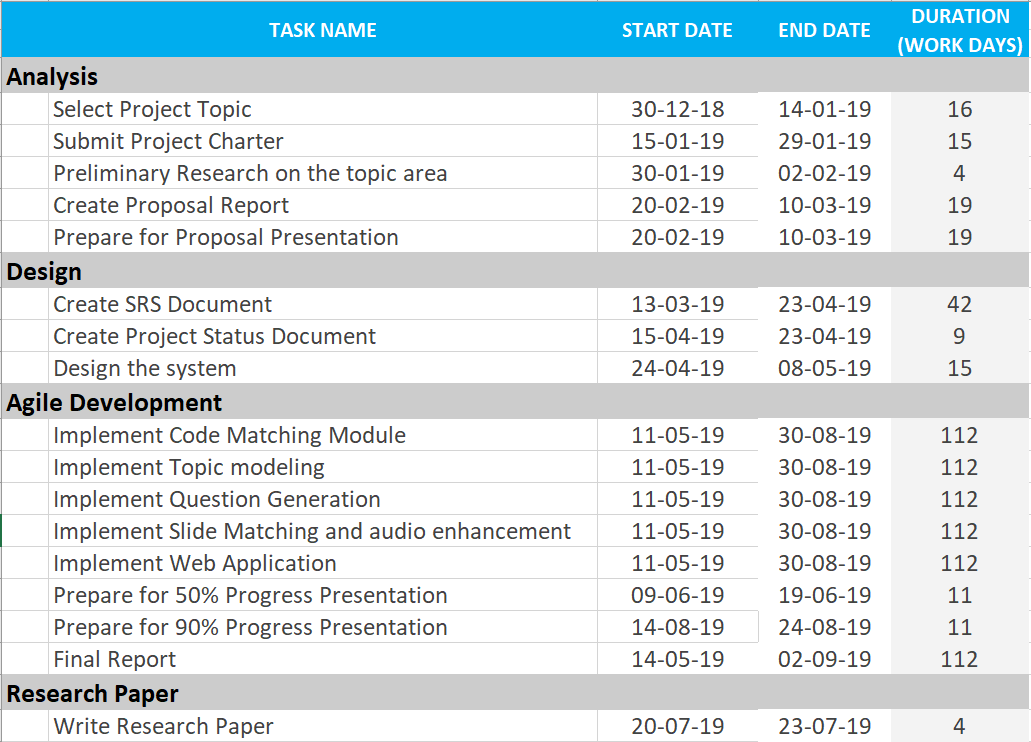
****

Figure 3.9: Tasks schedule

## Work Breakdown Structure

Figure 3.10: Work Breakdown Structure

# BUSINESS POTENTIAL

The web-based nature of the platform we propose has several advantages when considering its potential business value.

* It can be provided as a plugin for several existing MOOCs and LMSs.
* It can be hosted on a cloud platform and provided as a *Software as a Service* (SaaS) product, where the customer will pay a one-time fee or a subscription to use it.
* It can be developed as a website and advertisements can be incorporated into the system.
* It can be developed as a Freemium model, where certain services are provided free of charge and certain premium services can be provided for a fee.

# DESCRIPTION OF PERSONAL AND FACILITIES

|  |  |  |
| --- | --- | --- |
| Registration No | Name | Task Description |
| IT 16037434 | Karunaratne D. C. | * Train a machine learning algorithm for the classification of frames which contain source code. * Develop algorithm for identification of the video frame which corresponds to the first frame where the code was typed. * Documentation * Testing |
| IT 15146366 | Hettiarachchi H. A. I. S. | * Develop algorithm for identify and extract important sentences and words from video transcript. * Develop algorithm for generate questions and distractors using extracted words and domain ontologies. * Documentation. * Testing. |
| IT 16001862 | Fernando S. S. M. S. | * Extract transcription along with timestamps from raw mp4 video files * Develop an algorithm to identify topic transitions, by giving the transcription as an input. * Segment the video file according to identified topic transitions. * Documentation * Testing |
| IT 16009646 | Epa S. S. | * Develop an algorithm to point out the time stamps where a particular slide is discussed. * Develop an algorithm to compute the noise spectrum and noisy speech spectrum of a noisy speech and to remove the noise * Documentation * Testing |

Table 5.1: Description of Personal and Facilities

# References

|  |  |
| --- | --- |
| [1] | "Importance and Effectiveness of E-learning," 1 December 2015. [Online]. Available: https://higheredrevolution.com/importance-and-effectiveness-of-e-learning-9513046ed46c. [Accessed 6 March 2019]. |
| [2] | M. Brandsteidl, T. Mayerhofer, M. Seidl and C. Huemer, "Replacing traditional classroom lectures with lecture videos: an experience report," in *Proceedings of the 8th edition of the Educators' Symposium*, 2012. |
| [3] | M. Maher, H. Lipford and V. Singh, "Flipped classroom strategies using online videos," UNC Charlotte, North Carolina, 2013. |
| [4] | M. Chau, M. Lin, J. F. Nunamaker and H. Chen, "Segmentation of Lecture Videos Based on Text: A Method Combining Multiple Linguistic Features," in *Proceedings of the 37th Hawaii International Conference on System Sciences*, Hawaii, 2004. |
| [5] | Z. Woolfitt, "The effective use of video in higher education," 2015. [Online]. Available: https://www.inholland.nl/media/10230/the-effective-use-of-video-in-higher-education-woolfitt-october-2015.pdf. [Accessed 6 March 2019]. |
| [6] | M. J. Rubin, "The effectiveness of live-coding to teach introductory programming," in *Proceeding of the 44th ACM technical symposium on Computer science education - SIGCSE ’13*, Denver, Colorado, USA, 2013. |
| [7] | V. K. Kamabathula and S. Iyer, "Automated Tagging to Enable Fine-Grained Browsing of Lecture Videos," in *2011 IEEE International Conference on Technology for Education*, 2011. |
| [8] | Luca Ponzanelli et al, "CodeTube: Extracting Relevant Fragments from Software Development Tutorials," in *Proceedings of the 38th International Conference on Software Engineering Companion - ICSE ’16*, 2016. |
| [9] | A. M. a. M. L. L. Ponzanelli, "StORMeD: Stack Overflow ready made data," in *Proceedings of MSR 2015 (12th Working Conference on Mining Software Repositories)*, 2015. |
| [10] | J. Ott, A. Atchison, P. Harnack, A. Bergh and E. Linstead, "A deep learning approach to identifying source code in images and video," in *Proceedings of the 15th International Conference on Mining Software Repositories - MSR ’18*, 2018. |
| [11] | K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *CoRR,* vol. abs/1409.1556, 2014. |
| [12] | K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. |
| [13] | C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. |
| [14] | M. Lin, C. B. R. Diller, Ming, N. Forsgren, Y. Huang and J. F. Nunamaker, Jr, "Segmenting Lecture Videos by Topic: From Manual to Automated Methods," in *Eleventh Americas Conference on Information Systems*, Omaha, NE, 2005. |
| [15] | H. J. Zhang and S. W. Smoliar, "Developing power tools for video indexing and retrieval," in *SPIE'94 Storage and Retrieval for Video Databases*, San Jose, CA, USA, 1994. |
| [16] | X. Che, H. Yang and C. Meinel, "Lecture Video Segmentation by Automatically Analyzing the Synchronized Slides," in *MM’13*, Barcelona, Spain, 2013. |
| [17] | J. Allan, J. Carbonel, G. Doddington, J. Yamron and Y. Yang, "Topic detection and tracking pilot study: Final report," in *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*. |
| [18] | R. Shah, D. Shah and L. Kurup, "Automatic Question Generation for Intelligent Tutoring Systems," in *2017 2nd International Conference on Communication Systems, Computing and IT Applications (CSCITA)*, Mumbai, 2017. |
| [19] | Y.-T. Huang, Y.-M. Tseng, Y. S. Sun and M. C. Chen, "TEDQuiz: Automatic Quiz Generation for TED Talks Video Clips to Assess Listening Comprehension," in *2014 IEEE 14th International Conference on Advanced Learning Technologies*, Athens, 2014. |
| [20] | G. Erkan and D. R. Radev, "LexRank: Graph-based Lexical Centrality as Salience in Text Summarization," in *Journal of Artificial Intelligence Research, Vol 22*, 2004. |
| [21] | M. Heilman, "Automatic Factual Question Generation from Text," Doctoral thesis, Carnegie Mellon University, Pittsburgh, 2011. |
| [22] | M. Heilman and N. A. Smith, "Question Generation via Overgenerating Transformations and Ranking," in *Technical report, Language Technologies Institute, Carnegie Mellon University Technical Report*, 2009. |
| [23] | A. Krishna, P. Bhowmick, K. Ghosh, A. Sahu and S. Roy, "Automatic Generation and Insertion of Assessment Items in Online Video Courses," in *IUI Companion '15 Proceedings of the 20th International Conference on Intelligent User Interfaces Companion*, Atlanta, Georgia, 2015. |
| [24] | M. Heilman and N. A. Smith , "GoodQuestion! StatisticalRankingforQuestionGeneration," in *HLT '10 Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, Log Angeles, California, 2010. |
| [25] | M. Tosic and M. Cubric, "SeMCQ – Protégé Plugin for Automatic Ontology-Driven Multiple Choice Question Tests Generation," in *Procs of the 11th International Protege Conference*, 2009. |
| [26] | M. Al-Yahya, "OntoQue: A Question Generation Engine for Educational Assesment Based on Domain Ontologies," in *2011 IEEE 11th International Conference on Advanced Learning Technologies*, Athens, 2011. |
| [27] | A. Papasalouros , K. Kanaris and K. Kotis, "Automatic Generation Of Multiple Choice Questions From Domain Ontologies," in *IADIS International Conference e-Learning 2008*, Amsterdam, 2008. |
| [28] | F. Wang, C. W. Ngo and T. C. Pong, "Synchronization of lecture videos and electronic slides by video text analysis," in *Proceedings of the eleventh ACM international conference on Multimedia - MULTIMEDIA ’03*, Hong Kong, 2003. |
| [29] | "Zentation.com - Webinar software," Zentation.com, 2019. [Online]. Available: http://www.zentation.com/. [Accessed 08 March 2019]. |
| [30] | K. Hymavathy and P. Janardhanan, "Noise Filtering in Speech Using Frequency Response Masking Technique," *International Journal of Emerging Trends in Engineering and Development,* 2013. |
| [31] | S. Muangjaroen and T. Yingthawornsuk, "A Study of Noise Reduction in Speech Signal Using FIR Filtering," in *Proceedings of the International Conference on Advances in Electrical and Electronics Engineering*, Pattaya, 2012. |
| [32] | T. L. Kumar, K. S. Rajan and ., "Noise Suppression in Speech Signals Using Adaptive Algorithms," *International Journal of Engineering Research and Applications,* vol. 2, pp. 718-721, 2012. |
| [33] | R. S. J. G. V. R. S. T. M. a. K. A. Aggarwal, "Noise Reduction of Speech Signal Using Wavelet Transform with Modified Universal Threshold," *International Journal of Computer Applications,* vol. 20, pp. 15-19, 2011. |
| [34] | E. Verteletskaya and B. Simak, "Speech Distortion Minimized Noise Reduction Algorithm," in *Proceedings of theWorld Congress on Engineering and Computer Science*, San Francisco, 2010. |
| [35] | M. Karam, H. F. Khazaal, H. Aglan and C. Cole, "Noise Removal in Speech Processing Using Spectral Subtraction," *Journal of Signal and Information Processing,* vol. 5, no. 2, pp. 32-41, 2014. |
| [36] | "Audacity - Free Open source, cross platform audio software," Audacity, 2019. [Online]. Available: https://www.audacityteam.org/. [Accessed 05 March 2019]. |
| [37] | "Learnworlds: Create and Sell Online Courses from Your Own Website," Learnworlds, 2019. [Online]. Available: https://www.learnworlds.com/. [Accessed 01 March 2019]. |
| [38] | "About Echo360 - The Smarter Video Platform for Education," Echo360, 2019. [Online]. Available: https://echo360.com/about/. [Accessed 01 March 2019]. |
| [39] | "TechSmith | Create high-quality images and videos for better communication," TechSmith, 2019. [Online]. Available: https://www.techsmith.com. [Accessed 01 March 2019]. |
| [40] | J. Brownlee, "A Gentle Introduction to Transfer Learning for Deep Learning," Machine Learning Mastery, 2019. [Online]. Available: https://machinelearningmastery.com/transfer-learning-for-deep-learning/. [Accessed 1 March 2019]. |
| [41] | W. Nowak, "How to Train Your Model (Dramatically Faster)," Towards Data Science, 2019. [Online]. [Accessed 1 March 2019]. |
| [42] | M. A. Hearst, "TextTilting: Segmenting Text into Multi-Paragraph Subtopic Passages," *Computational Linguistic,* 1994. |
| [43] | "Importance and Effectiveness of E-learning," 1 December 2015. [Online]. Available: https://higheredrevolution.com/importance-and-effectiveness-of-e-learning-9513046ed46c. [Accessed 6 March 2019]. |