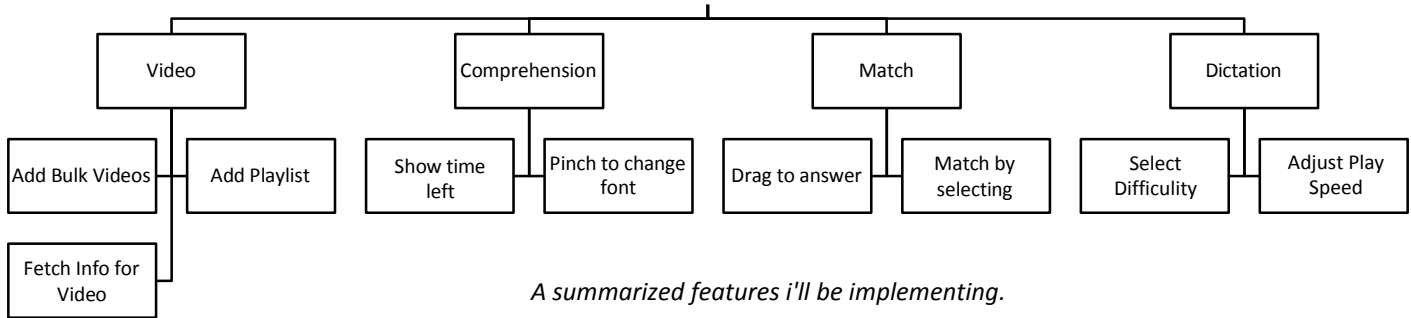


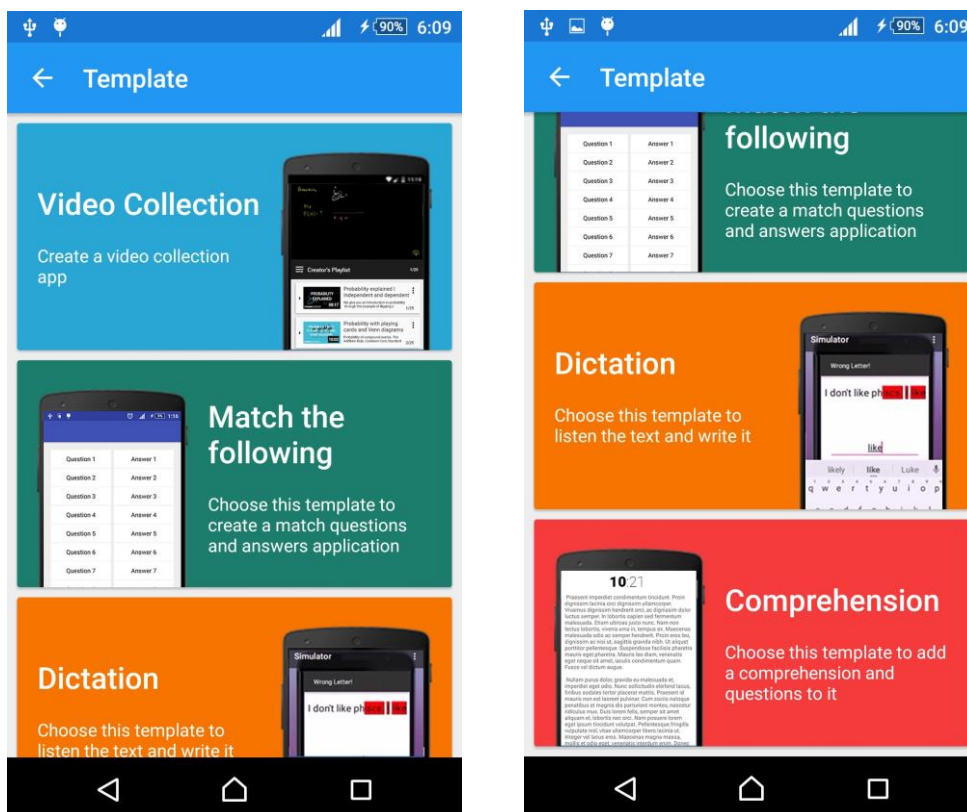
# Adding more templates to BuildMLearn.



## Abstract

These four new plugins will be added on the existing plugin architecture. Also I will be working on the old 4 templates and tweak them and update them to material design specs and improve them.

While adding new templates this is how it will look.



The whole document is in respect to two people

- Students - The student which are seeking to learn through this application.
- Teacher - The one who is creating the template for the students.

[Full Resolution Mockups Here.](#)

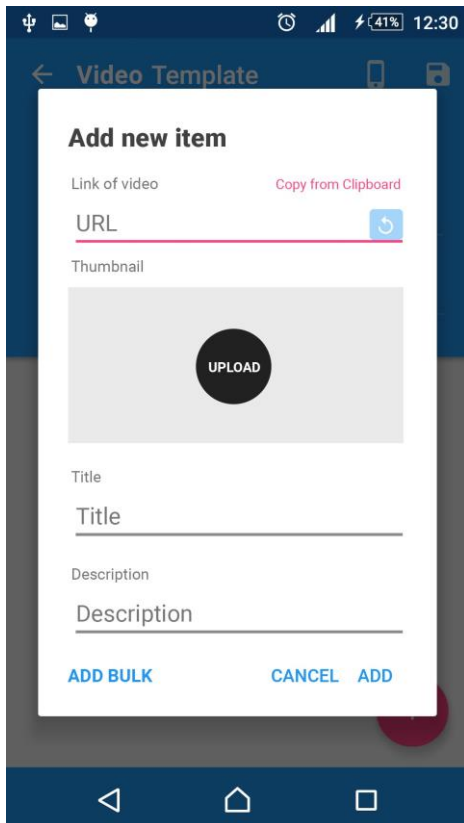
# Video Collection Template-

## Creation -

Two ways to add videos –

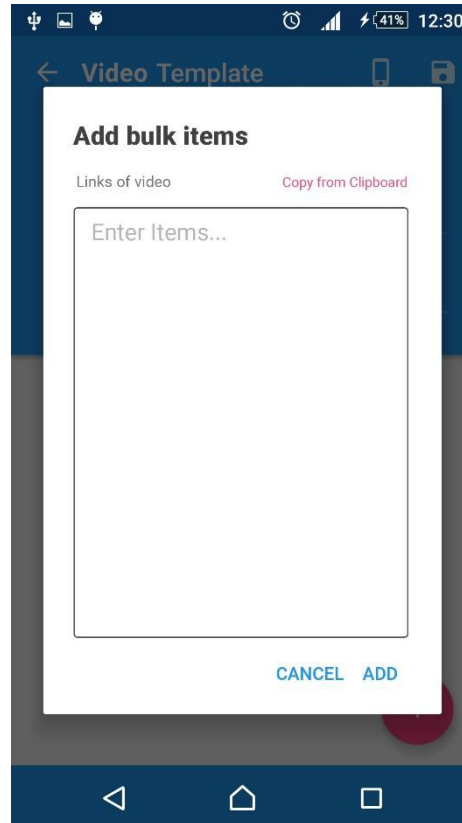
### Adding Single Mode

If a playlist is given there then it will be processed like bulk items

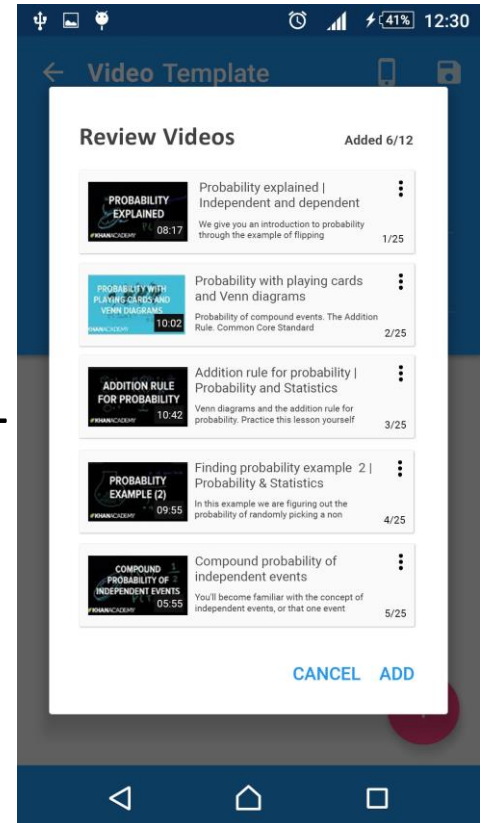


### Adding Bulk Video

After adding playlist or videos in the textarea given below they all will be processed to fetch the information and be reviewed and finally added to the playlist.



+

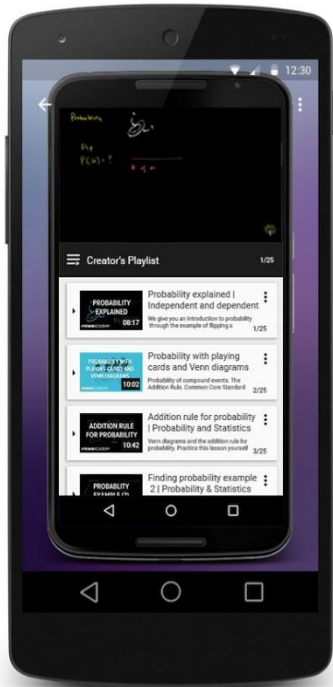


## Other Features –

- If the URL is from youtube, vimeo or dailymotion I'll automatically fetch the thumbnail, title and length of video.  
How will I process the URL to get the information?
  - YouTube – API provided by youtube.
  - Vimeo – Vimeo API.
  - Dailymotion – Dailymotion API.
- A URL can also be a playlist of either of these 3 sites, from where I can get the single videos and then process the URL's to get information.
- For Downloading the thumbnail I'll use Fresco API by Facebook for downloading images.
- Creator will be able to drag and drop to change the position of the video
- OverFlow Menu-
  - Remove the Video.
  - Move video to top of playlist.

## Simulation -

All videos in a template will be displayed in a form of playlist.



## Features

- Drag and drop to change the order of videos playing.
- When orientation changed to landscape then full screen activates.
- OverFlow Menu-
  - **Play Next** - Play the video next.
  - **Remove** - Remove the video.
  - **Share** - Share the complete playlist.

## Comprehension Template –

### Creation –

#### Add New Passage

Creator will have to add a title, passage.

#### Add New Questions

Adding a new multiple choice question.

### Simulation –

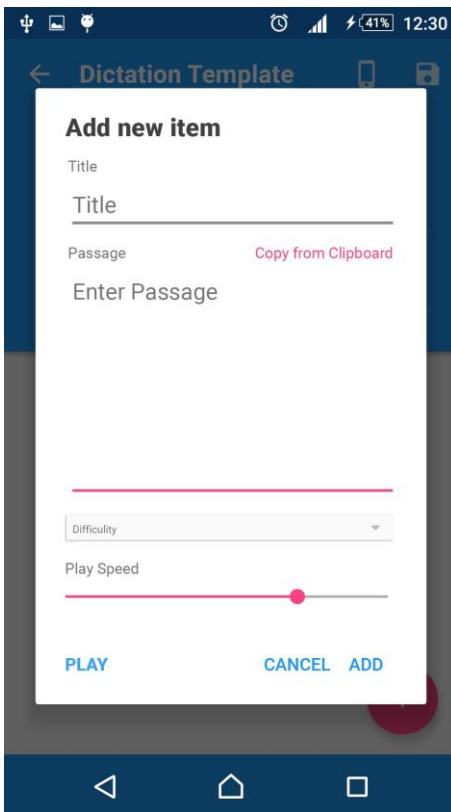


#### Other Features-

- Pinch at TextView to change the font size, to increase the readability.
- Timer will be displayed at the top.

## Dictation Template –

### Creation –



Add New Passage – Here users will have to enter a title, passage, difficulty (Learning, Pro mode and User Selectable) and the play speed.

Teacher can also listen to the speed also by Play Button.

TTS Library which I'm going use is this one – Google Text to Speech

### Difficulty

- Learning mode – This will be the easiest mode where the students will be corrected in real time more explained in the simulation part below!
- Pro Mode – This won't be real time and the student will be given marks at the end.

### Simulation –



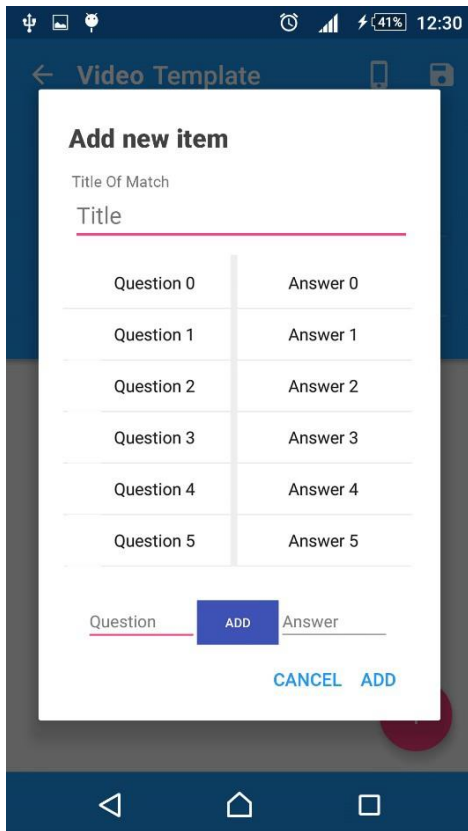
### Learning Mode

Students being notified that they are writing the wrong text hence they should go back and make the text correct, meanwhile while they are erasing the TTS dictation will also stop and wait till the user has corrected this mistake and again starting from where the student did the mistake.

[I coded this simulation and uploaded a video for it](#)

# Match Template

## Creation-



## Add New Match

Teacher need to add question and a corresponding answer to it.

## Simulation –

Students can match in two ways –

- Click on one question part and select the corresponding answer.
- They can drag the question part to the corresponding answer.

I will use AutoFitTextView to fit the questions or answers in two lines.



[I have added a video displaying the demo of the simulation of the matching template.](#)

## Changes required in the Desktop Application –

- **Video Collection Template –**

- Implementing a feature to add List of videos and parse their information and add to the Active Videos List.
- Change Parsing of HTML to use API's for fetching information of the videos.

- **Dictation Template –**

- Adding a difficulty option to select the difficulty.
- Implementation of highlighting wrong words, punctuation etc. in real time if learning mode is selected.

- **Comprehension Template –** Remains to be same.

- **Match Template –**

- Adding feature to select a question and select a corresponding answer, which will position the answer in front of question.
- Adding feature to drag and drop a question to corresponding answer to it, which will position the answer in front of question

## About me

I am Kunal Gupta, an engineering undergraduate pursuing Information Technology at Manipal Institute of Technology.

I've been developing for android past 2 years, have made contributions to the BuildMLearn Toolkit as well!

Here are some of my applications-

- **Crowdspell**- A game built for [Android](#)/Windows Phone
- **EntryScreenManager** - Built a library for importing Intro/Entry screens posted at [materialup.com](http://materialup.com) as well!
- **TextBuilder**- Built a [Standalone App](#)/[Plugin for IntelliJ\(Android studio\)](#) for generating automated text, [Details!](#)
- **IONAutoLogin**- Android app to automatically login to my college Wifi.

I have also worked as a freelancer for Android Projects, also worked as a Web-Developer in a WebSite Dev company! **Contact Information –**

Email – [kunall.gupta17@gmail.com](mailto:kunall.gupta17@gmail.com)

Phone No. - +91 9795197787

## Road-map

GSOC goes for about 13 weeks in which 1 month (4 weeks) for Community Bonding period. Timeline –

### Community Bonding Period

- Discuss the project, changes, enhancements with the mentor and the organization.
- Gather Information and API's required for the project.
- Changes to be made in the 4 old templates.

Task Name	Start Date	End Date	Time (Days)
<b>Google Summer Of Code</b>	<b>25<sup>th</sup> April</b>	<b>23<sup>rd</sup> August</b>	<b>116</b>
<b>Community Bonding</b>	<b>22<sup>nd</sup> April</b>	<b>22<sup>nd</sup> May</b>	<b>30</b>
Gather Information	22 <sup>nd</sup> April	25 <sup>th</sup> April	3
Documentation	25 <sup>th</sup> April	1 <sup>st</sup> May	7
Discussion of project	1 <sup>st</sup> May	11 <sup>th</sup> May	10
Convert PSD to XML	11 <sup>th</sup> May	21 <sup>st</sup> May	10
UI of Video Collection	21 <sup>st</sup> May	26 <sup>th</sup> May	5
Implement fetch information through API's	26 <sup>th</sup> May	29 <sup>th</sup> May	3
Playback of Videos	29 <sup>th</sup> May	2 <sup>nd</sup> June	4
Make Changes in the Desktop Application	2 <sup>nd</sup> June	7 <sup>th</sup> June	5
UI of Comprehension	7 <sup>th</sup> June	14 <sup>th</sup> June	7
Simulation of Comprehension	14 <sup>th</sup> June	21 <sup>st</sup> June	7
<b>Midterm Evaluations</b>	<b>21<sup>st</sup> June</b>	<b>28<sup>th</sup> June</b>	<b>7</b>
Writing TODOS for the left part of the app	21 <sup>st</sup> June	22 <sup>nd</sup> June	1



Testing and Submitting the app for the midterm Evaluations.	22 <sup>nd</sup> June	28 <sup>th</sup> June	6
Modifying the Basic Learning Template	28 <sup>th</sup> June	1 <sup>st</sup> July	4
Modifying the Learn Spelling Template	1 <sup>st</sup> July	5 <sup>th</sup> July	4
Modifying the Quiz Template	5 <sup>th</sup> July	9 <sup>th</sup> July	4
Modifying the Flash cards Template	9 <sup>th</sup> July	13 <sup>th</sup> July	4
Implementation of Match Template	13 <sup>th</sup> July	17 <sup>th</sup> July	4
Implementation of Match Simulation	17 <sup>th</sup> July	21 <sup>nd</sup> July	4
Changes in Desktop for Match Template	21 <sup>nd</sup> July	25 <sup>th</sup> July	5
Implementation of Dictation UI	25 <sup>th</sup> July	28 <sup>th</sup> July	3
Developing the TTS Engine	28 <sup>th</sup> July	31 <sup>st</sup> July	3
Simulation of Dictation	31 <sup>st</sup> July	3 <sup>rd</sup> August	3
Changes in Desktop for Dictation Template	3 <sup>rd</sup> August	8 <sup>th</sup> August	5
Competition of Remaining Part	8 <sup>th</sup> August	13 <sup>th</sup> August	5
Implement Espresso Testing	13 <sup>th</sup> August	14 <sup>th</sup> August	1
Thorough Testing	14 <sup>th</sup> August	18 <sup>th</sup> August	4
Documentation	18 <sup>th</sup> August	20 <sup>th</sup> August	2
<b>Submission of the Final application</b>	<b>21<sup>th</sup> August</b>		

## Some Quick Questions

### Q. Why will I be the best suited for this project?

Ans. Firstly, I am an open source lover and love to contribute for the project. I do not have any other projects to work in the coming summer. I do have experience with android and professional development as well, I have done some freelancing projects.

**Q. How much time will I be able to give per week?**

A. I am into writing code past 5-6 years. So I have the stamina to work at least 10 hours a day. So I can give about 50+ hours per week.

**Q. Am I comfortable working independently under a supervisor or mentor who is just available virtually?**

A. Yes, I am absolutely comfortable with both the ways. As I have worked with bosses over the internet (As freelancer).

**Q. Have I submitted proposals for any other organizations except BuildMLearn? If yes, which orgs have I applied for?**

A. Yes, I have submitted a proposal for Zulip origination as well.

**Q. Have I participated in any previous GSoC programs? If yes, tell BuildMLearn about the experience briefly and if I completed successfully.**

A. No, This is my first time applying for GSOC.

**Q. What is my primary motivation to submit a proposal for this project? Am I interested because I am getting a stipend as a part of GSoC or would I be interested to develop this project for BuildMLearn even outside GSoC?**

A. Honestly, I am working for GSOC for the stipend and for the Certification from Google. Plus I'll be learning so many new things.

Yeah, I would definitely like to contribute if I don't have any current projects of mine going on as I like the idea of the application and have an optimism that this application will be very popular for learning once it has been fully published and advertised.