# Project Plan: (Easy-Author animator)

Written by:
(Kiran Kumar Lekkala)

Following is the plan for my passion project:

## 1 Introduction

Easy-Author tool makes it easy for people who want to teach students in their own resourceful style. This app is already in implementation and will be released soon. A framework which generates the easy-author tool is also underway. This animator is an front-end extension for the easy-author tool.

## 2 Overview

Easy-author tool takes in information from the users as of how they would like to teach and so on. Using all the given parameters and the content they would like to teach, easy-author tool generates an xml file. This xml file in-turn given as input to this animator app which then generates the necessary utilites which are used for teaching students.

### 2.1 Customers

2.1.1 Teacher: The Teacher is main user of this animator application. He/She will be the registered user for the easy-author app which will generate a xml file. When the set of server files are generated by this animator app will be used by the teacher as presentations to teach the students.

2.1.2 Student: The mobile app file which is generated by this application wil be used by the student as a study tool for animations

| | |
|---|---|
| ID: | UC-1 |
| Title: | Uploading the xml file |
| Description: | Giving input and the content with which the author teaches. |
| Primary Actor: | Teacher |
| Preconditions: | Teacher is logged into system |
| Main Success Scenario: | 1. The teacher firsts logs into the system.<br>2. System displays teacher's dashboard once he is logged in.<br>3. The teacher clicks on the button create a new course<br>4. All the courses created in the easy author app will be displayed and the teacher clicks on the course.<br>5. The system will then ask for an xml file which was generated by the easy author app. |
| Extensions: | 2a. No courses are available for this teacheer.<br>— 2a1. System displays error message saying no courses are available, and provides the reason & how to rectify if possible.<br>— 2a2. Teacher either backs out of this use case, or tries again after rectifying the cause.5a. Some courses could not be created.<br>— 5a1. System displays message showing which courses were created, and which courses were not created along with a reason for each failure.5b. None of the courses could be created.<br>— 5b1. System displays message saying none of the courses could be created, along with a reason for each failure. |

| | |
|---|---|
| ID: | UC-2 |
| Title: | Installing the app or the web-server files |
| Description: | Teacher |
| Primary Actor: | Teacher |
| Preconditions: | Teacher is logged into system |
| Postconditions: | Teacher has downloaded the animated files |
| Main Success Scenario: | 1. Teacher downloads the set of animated files.<br>2. He/She extracts the files in his/her local machine.<br>3. He follows the instructions in readme file and runs the make file which automatically installs them. |
| Extensions: | 2a. Error in installing the animated files<br>— 2a1. System displays error message saying no courses are available, and provides the reason & how to rectify if possible.<br>— 2a2. Teacher either backs out of this use case, or tries again after rectifying the cause by reporting it to the administrator or by posting on the forum. |
| Frequency of Use: | A few times every quarter |
| Status: | Pending Review |
| Owner: | Kiran Kumar Lekkala |
| Priority: | P3 – Medium |

| | |
|---|---|
| ID: | UC-3 |
| Title: | Edit user's profile |
| Description: | The user(Teacher or student) can edit his profile which consists of his personal description |
| Primary Actor: | User |
| Preconditions: | user is logged into system |
| Postconditions: | None |
| Main Success Scenario: | 1. User selects profile button on the right-top side of the menu.<br>2. System displays lthe current profile of the user.<br>3. User selects one or more courses he wants to register for.<br>4. User clicks “Submit” button.<br>5. System registers student for the selected courses and displays a confirmation message. |
| Extensions: | 2a. no/some description of the profile available for this user.<br>— 2a1. System displays warning message saying no /some desctiption of profile is available, and asks the user to fill in the necessary information. |
| Frequency of Use: | A few times every quarter |
| Status: | Pending Review |
| Owner: | Kiran Kumar Lekkala |
| Priority: | P3 – Medium |

ID: UC-4

Title: Student registers for courses

Description: Student registering for a specific course which has been created by the Teacher

Primary Actor: Student

Preconditions: Student is logged into system

Postconditions: Student is registered for course

Main Success Scenario: 1. Student selects “Register New Courses” from the menu.
2. System displays list of courses available for registering.
3. Student selects one or more courses he wants to register for.
4. Student clicks “Submit” button.
5. System registers student for the selected courses and displays a confirmation message.

Extensions: 2a. No courses are available for this student.
— 2a1. System displays error message saying no courses are available, and provides the reason & how to rectify if possible.
— 2a2. Student either backs out of this use case, or tries again after rectifying the cause.5a. Some courses could not be registered.
— 5a1. System displays message showing which courses were registered, and which courses were not registered along with a reason for each failure.5b. None of the courses could be registered.

Frequency of Use: A few times every quarter

Status: Pending Review

Owner: Kiran Kumar Lekkala

Priority: P3 – Medium

### 2.2 Functionality

Following are the list of some points which form the main functionality of this application:

1. This app must be able to generate graphical animations from the xml data file which is fed by the registered user.

2. Unlike the easyauthor app and easyauthor framework, this animator app will also enable a login for the student. The student will be able to view and download the animated material the Teacher has posted for him. Also, the student will be able to send a registration request for a specific course.

3. The graphical objects and modules should be already existing in the app and will be defined in the previous phase of this app. i.e. easyauthor app. The xml file will then specify what all and how the animations should by used.

### 2.3 Platform

This web-application will enable to users to browse and upload the xml file. A process will be run which will output the animated versions of server application files and an apk file which will be used for teaching in mobile phones. This app will be purely a web-app which can be accessed and used independent of any platform.

## 3 Goals and Scope

This application is being made in Ruby on Rails which is a Ruby development platform.

**Note**: Presently since we have not yet developed much, the scop of this application will be brought out when we roll the alpha version by the end of october.

## 4 Deliverables

At the end of this project:

1. This product will be able to generate a set of files which can be used when hosting a server for teaching people. It will also generate an apk file which can be installed on android phones.

**2.** Well documented readme file will be written for every sub-module and module of this application with which user will not have any trouble while running or installing an app or the set of files.

3. The user(Teacher) will have option by which he can vary some of the specific settings of the animated files even at runtime. If he wants to change the fontsize, animation pattern, time-intervals etc. in between different animations can be changed only in the web-server files which will be used by the teacher. The app will be made read-only.

## 6 Scheduling and Estimates

Following are the milestones which I have set to finish the project in time and to give time for fixing the bugs. Following is the rough sketch of the milestones:

| Milestone | Description | Release Date | Release Iteration |
|---|---|---|---|
| M1 | User management and scema design | September 26, 2015 | R1 |
| M2 | project management, basic distributed filesystem | October 26, 2015 | R1 |
| M3 | Backend part | November 14, 2015 | R1 |
| M4 | Final alpha version released | December 1, 2015 | R1 |
| M5 | Final beta version released | December 14, 2015 | R2 |

A more detailed description of the features for each milestone is already provided in the previous section of this requirements document (using use cases).

## 7 Technical Process

1. The GUI for setting up the application has to be configured which will provide an upload field asking the user for the xml file generated by the easy-author application.

2. Then the user might give some additional information for how the content can be divided in an organized manner. This will be the final stage of the user giving any input.

3. An XML parser must be installed in the ruby on rails framework which will enable the app to parse the input xml file which is output from the easyauthor app. The parser used for this framework is Nokogiri. Nokogiri is an HTML, XML, SAX, and Reader parser for Ruby on rails. Among Nokogiri's many features is the ability to search documents via XPath or CSS3 selectors. Nokogiri parses and searches XML/HTML quickly, and also has correctly implemented CSS3 selector support as well as XPath 1.0 support.

4. After parsing the xml file a process will start to fit creating the required animations and customizing them accordingly. After which, these will be enclosed into a tarball or zip which can be extracted by the user-end machine. An android app will also be made available for the teacher and the student.

5. For generating animations there will be a pre-defined library of animated objects and modules. These will be some basic gif objects in the application. The user will be selecting the required animations from the xml file by defining and specifiying in the easyauthor app.