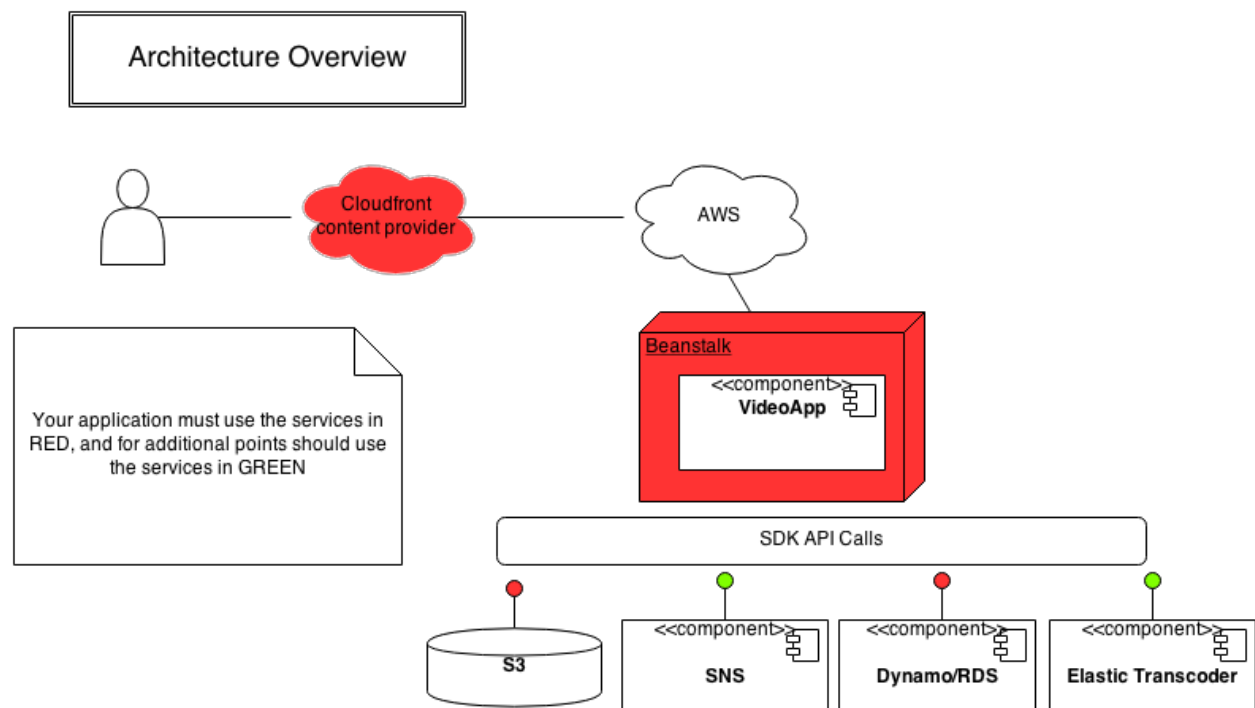


Assignment 1 - Video Conversation/Messages

The assignment has the following objectives:

1. You will develop a web app that allows people to exchange video messages as broadcasted messages and in turn they might get broadcast responses to generate open conversations.
2. Your web app should be deployed using Elastic Bean Stalk on AWS. We will provide some starting code for this.
3. Your application should make extensive use of the AWS designed for distributed video services like CloudFront, S3, RDS.
4. Your web app should be friendly to mobile devices.

Please find the architecture overview diagram below:



The web app will have 2 pages at least as below -

1. Home page where there's a list with the broadcasted conversations. Here you should be able to create a new broadcast conversation by uploading a video.
2. When you click on a conversation reply button, second page opens which will list the exchanged video messages over this conversation, you should be able to reply to a conversation by uploading a video.
3. User should be able to click on a video message and play it.

Evaluation:

Basic device-friendly web application 100 Points

Use of Elastic Transcoder and SNS 50 Points (25 Each)

Have fun coding!

PS

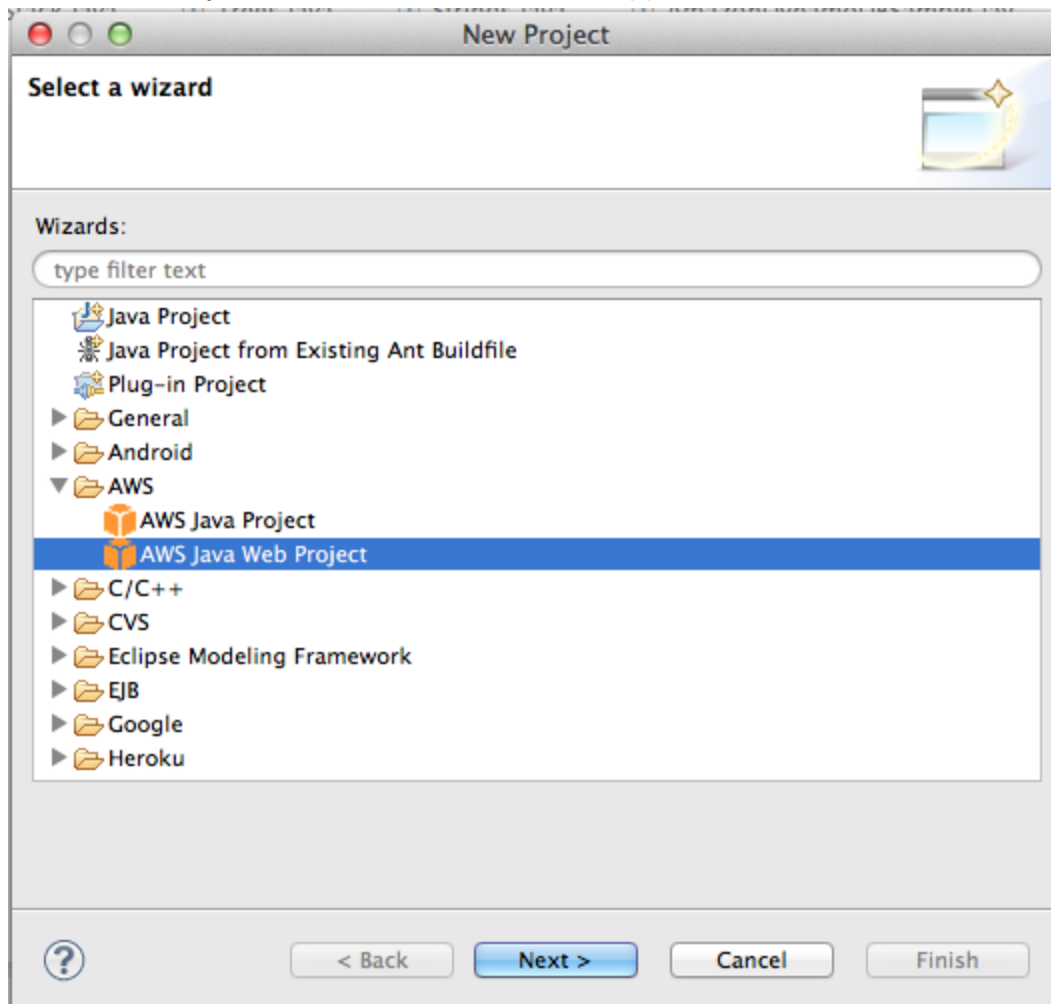
Check this out [chatwala](#)

Walkthrough and Code Snippets:

We are providing Code Snippets for DynamoDB and RDS so you can choose your persistence service. Those are found for download [here](#) and [here](#).

Also this is a brief walkthrough on creating a beanstalk app directly into your AWS account from Eclipse.

The easiest way to create an AWS Beanstalk app is to create a new AWS Java Web Project



Then, select the worker code example:

New AWS Java Web Project

Configure the options for creating a new AWS Java web project.

Project name: CloudTest

Select Account: ColumbiaCloud [Configure AWS accounts...](#)

Start from:

- ☐ Basic Java Web Application
A simple Java web application with a single JSP.
- ☒ Basic Amazon Elastic Beanstalk Worker Tier Application
A simple Amazon Elastic Beanstalk Worker Tier Application that accepts work requests via POST request.

Amazon DynamoDB Session Management:

☐ Store session data in Amazon DynamoDB
This option configures your project with a custom session manager that persists sessions using Amazon DynamoDB when running in an AWS Elastic Beanstalk environment.
This option requires running in a Tomcat 7 environment and is not currently supported for Tomcat 6.
[More information on the Amazon DynamoDB Session Manager for Tomcat](#)

? < Back Next > Cancel Finish

This will generate a base application that will help you understand the flow of information between HTTP Requests and the AWS backend services.

Finally, we are also providing visual mockup as reference. Please note this is a **suggestion**, feel free to improve and be creative.

