Read Me

University Name: San Jose State University http://www.sjsu.edu/

Course: Cloud Technologies

Professor: Sanjay Garje

ISA: Divyankitha Urs

Student: Sourabh Namilikonda http://www.linkedin.com/in/sourabh-namilikonda-255b226a

Project Introduction:

Project Introduction:

- For most of us music is an indispensable part of our daily lives and at one point in life or other every one of us has dreamed of learning to sing or play an instrument. A lot of us have some musical instrument at home which we want to learn to play but cannot spare enough time to join a class.
- Tune world is a one stop common platform for all levels of musicians - be it beginner trying to learn or an expert creating new tunes – to access all the information about any song they want to learn and share or secure access to the new secret song they want to play at some concert at any random location or country.
- Global and reliable access to a variety of files separated by language and genre and complete song tutorials all in one place without the pain of searching all over the internet for that right tune (like me).
- Choose to share your content when you are ready and help others out.

Feature list of the application:

- **Search** – A search panel allowing search based on filename, genre, language and song name to display details of the files matching search filters and a link to download them.

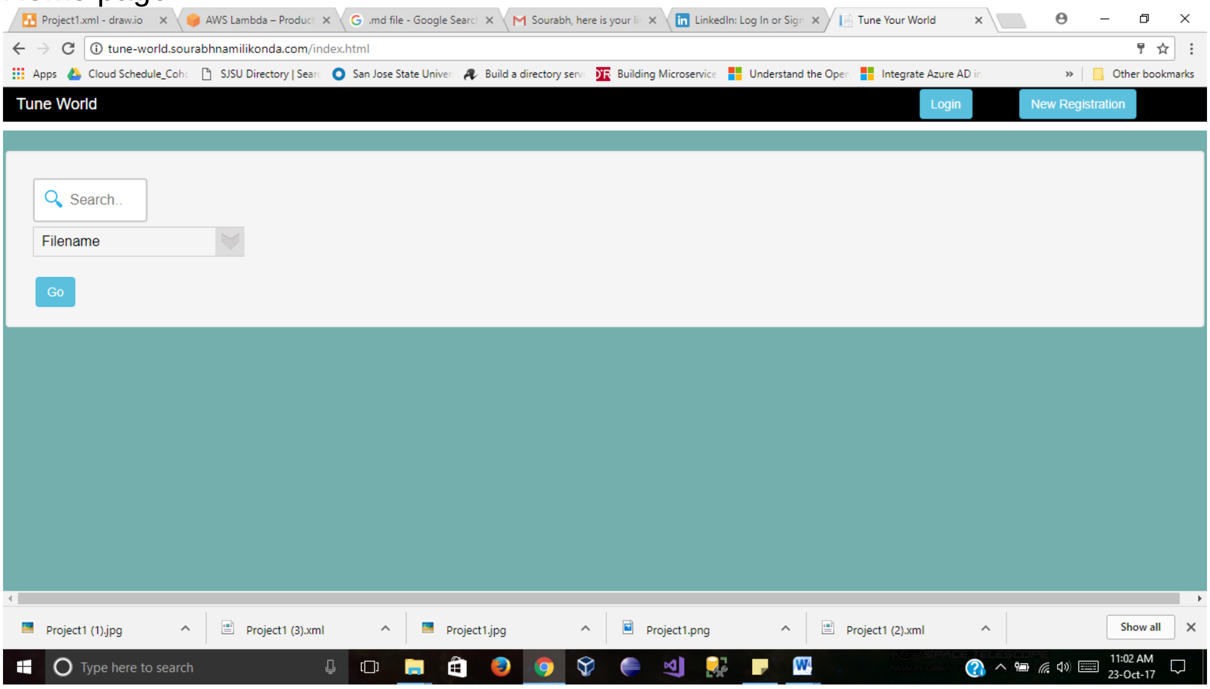
  Note**:** This search is only for files that are made publically accessible by the owner. No need to be a registered user to browse through them.
- **Signup** – Register as a user and then sign in securely with session management and receive a mail on successful registration. Validity checks on client side JavaScript and checks for unique Email ID and password encryption for privacy.

  Note: Password needs to be at least 12 characters with number, small and caps letters and symbols.
- **Login –** Login to your account to view, add, update and delete your personal files.
- **Add/Update Files** – Add your files by providing details like description, language, genre, song name and whether or not you want to make your file public. Files with same filename will update the existing files and overwrite the previous entries. Choose what you share. Limit of 10MB file size.
- **Delete –** Delete your files which cannot be later retrieved.

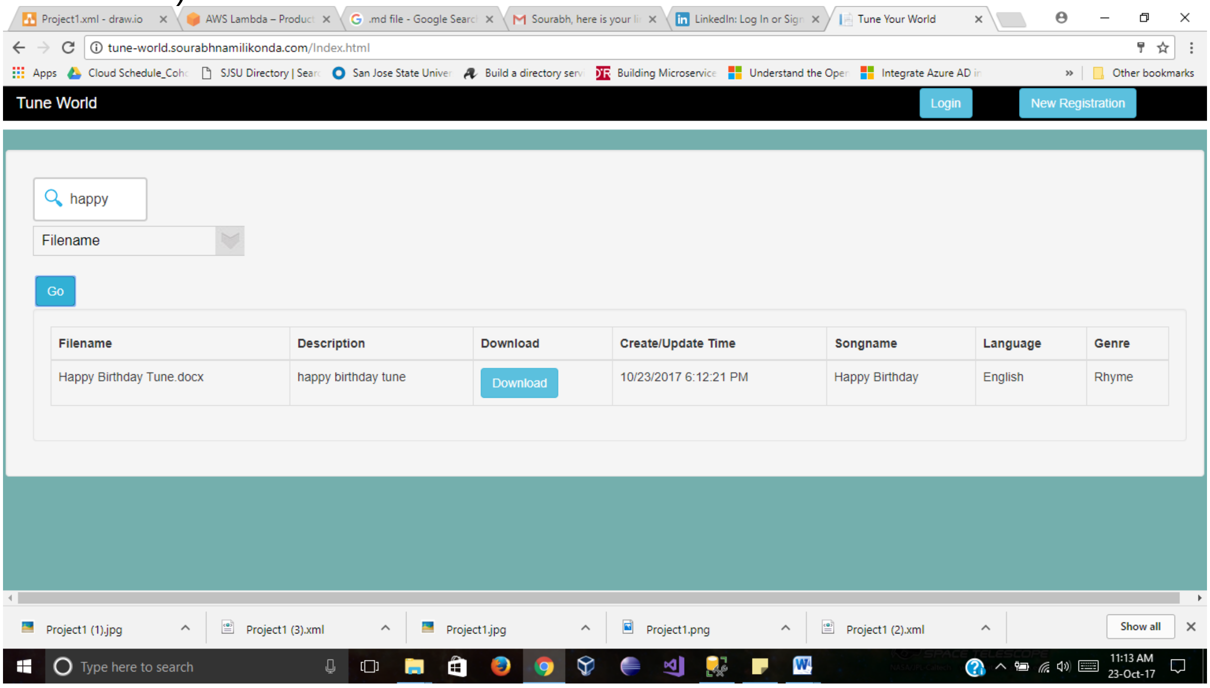
- **Download** – Download private and public files. Private files will have a Pre-Signed Link which will be valid only for a limited period of time before expiration.

Sample Demo Screenshots:

1. Home page.



2. Searching files that are made to be publically accessible (Lifetime active link to download)

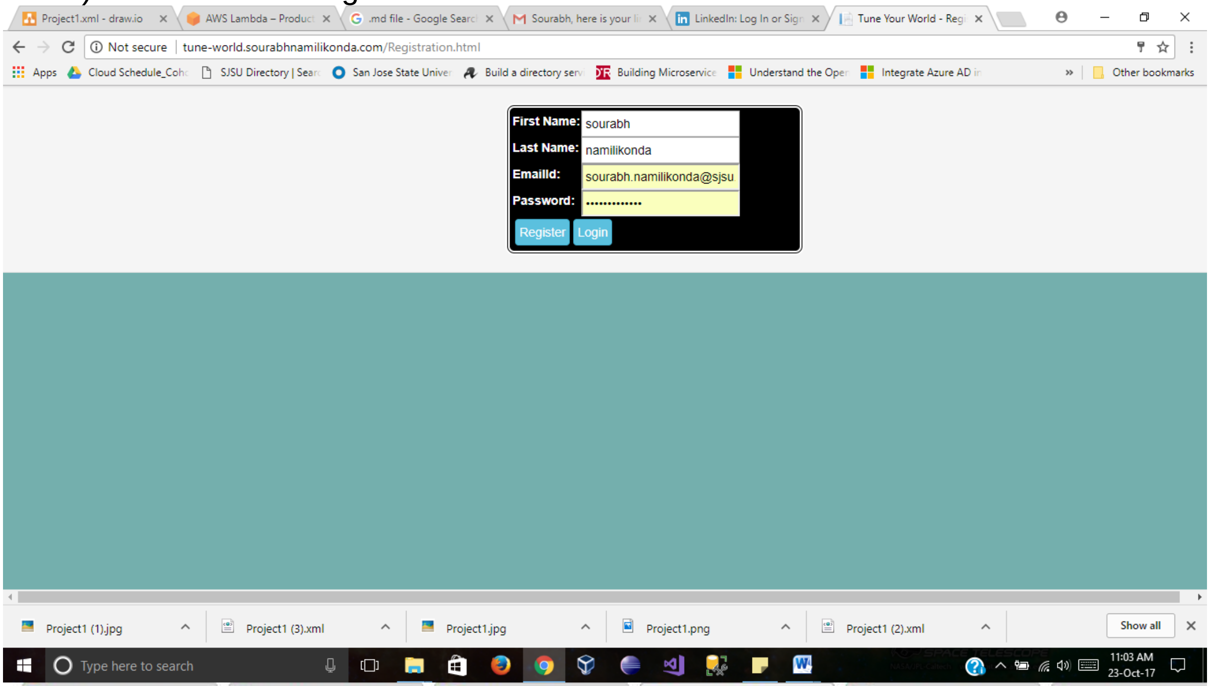


Search: happy

Filename

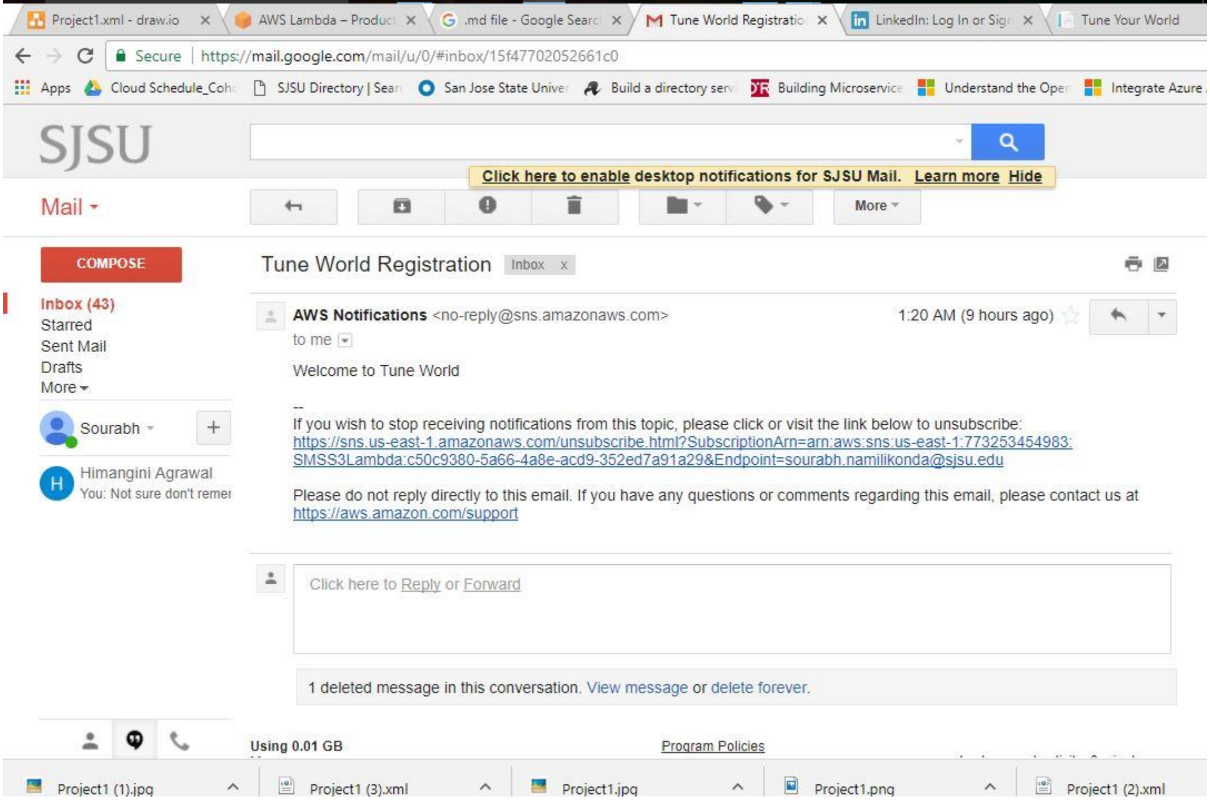
Filename	Description	Download	Create/Update Time	Songname	Language	Genre
Happy Birthday Tune.docx	happy birthday tune	Download	10/23/2017 6:12:21 PM	Happy Birthday	English	Rhyme

3. Registration (JS validation, Password Encryption, Username unique) Please use complex password with lower, upper case, letters and symbols (12 chars). Email sent on registration.



First Name:	sourabh
Last Name:	namilikonda
EmailId:	sourabh.namilikonda@sjsu
Password:	•••••••••••••

Register Login



Tune World Registration Inbox x

AWS Notifications <no-reply@sns.amazonaws.com> 1:20 AM (9 hours ago)

to me

Welcome to Tune World

--

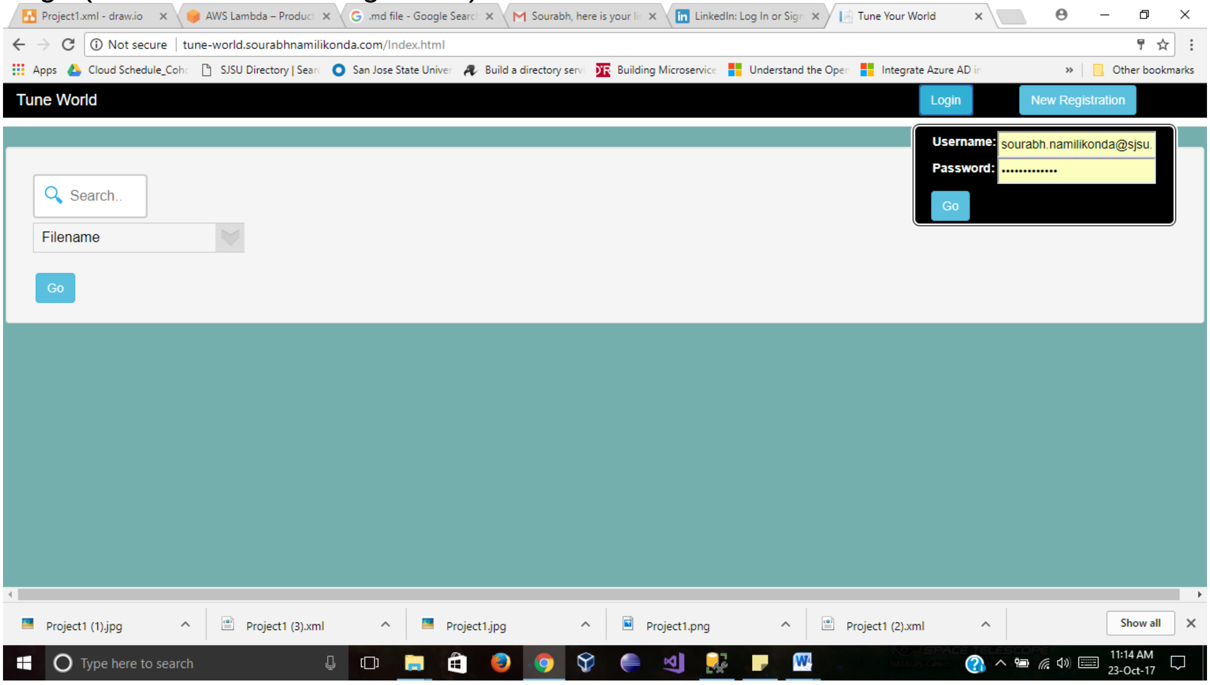
If you wish to stop receiving notifications from this topic, please click or visit the link below to unsubscribe:
https://sns.us-east-1.amazonaws.com/unsubscribe.html?SubscriptionArn=arn:aws:sns:us-east-1:773253454983:SMSS3Lambda:c50c9380-5a66-4a8e-acd9-352ed7a91a29&Endpoint=sourabh.namilikonda@sjsu.edu

Please do not reply directly to this email. If you have any questions or comments regarding this email, please contact us at https://aws.amazon.com/support

Click here to Reply or Forward

1 deleted message in this conversation. View message or delete forever.

4. Login(With session management)

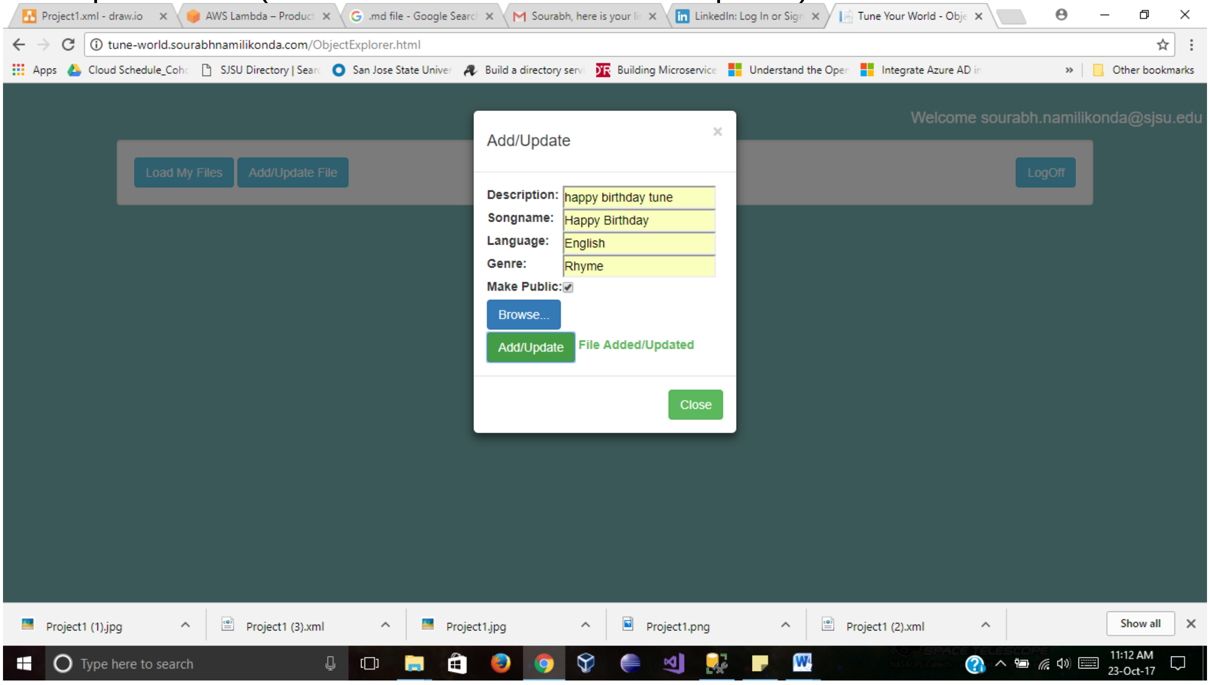


Tune World

Username: sourabh.namilikonda@sjsu.

Password: ....

5. Add/Update Files (Enter fields and check to make public)



Welcome sourabh.namilikonda@sjsu.edu

Add/Update

Description: happy birthday tune

Songname: Happy Birthday

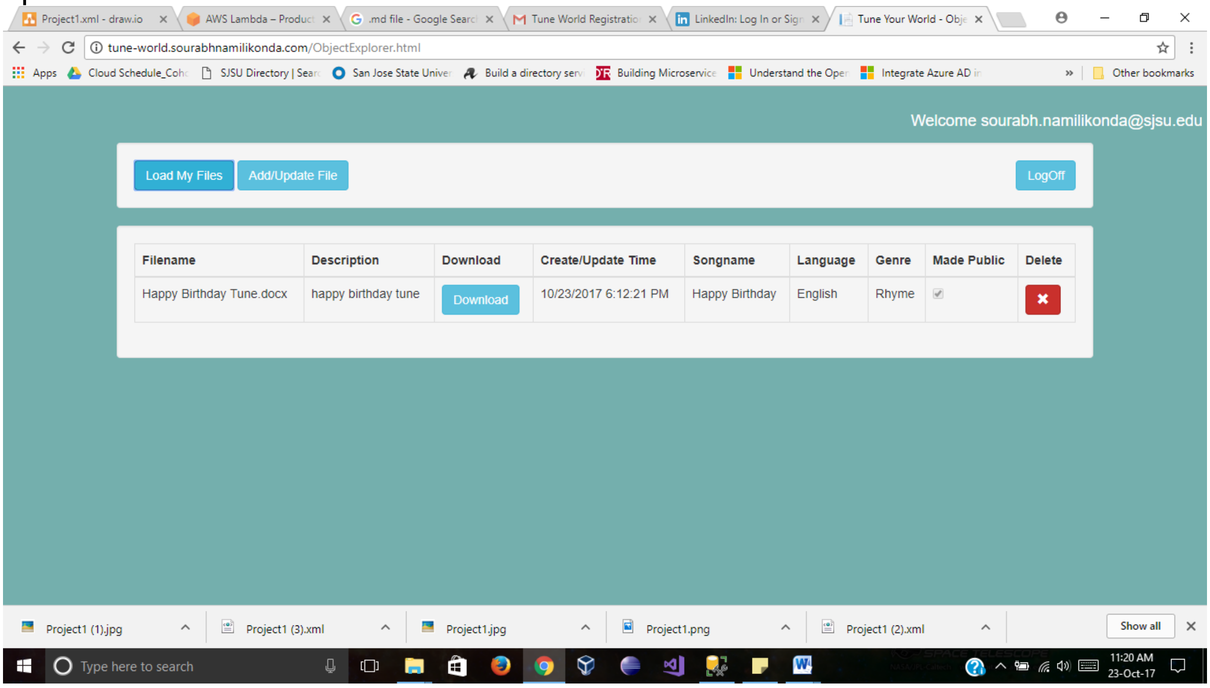
Language: English

Genre: Rhyme

Make Public: ☑

File Added/Updated

6. Loaded files with download (Signed url valid only for some time) and delete options

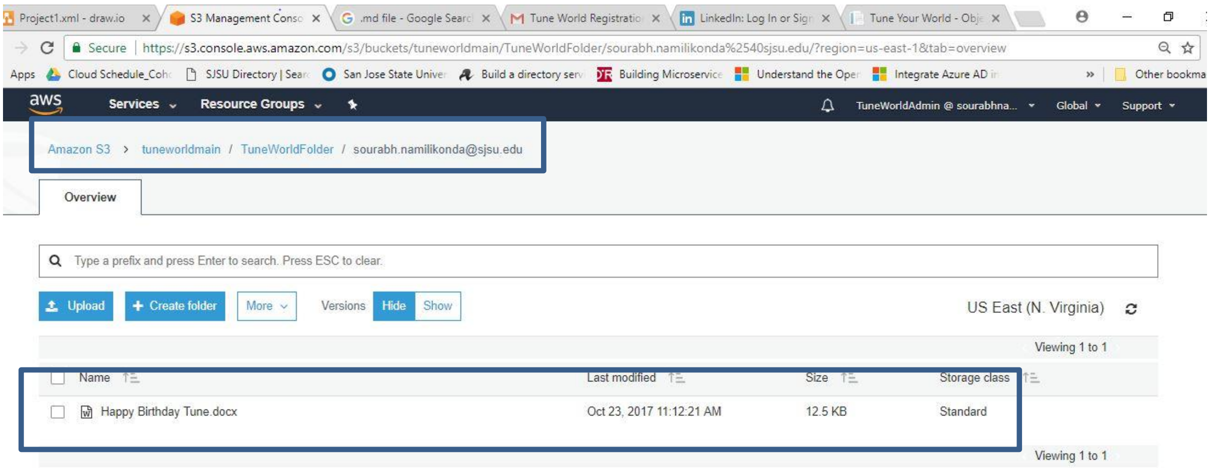


tune-world.sourabhnamilikonda.com/ObjectExplorer.html

Welcome sourabh.namilikonda@sjsu.edu

Filename	Description	Download	Create/Update Time	Songname	Language	Genre	Made Public	Delete
Happy Birthday Tune.docx	happy birthday tune	Download	10/23/2017 6:12:21 PM	Happy Birthday	English	Rhyme	☑	✖

7. Same files also shown in S3 bucket



https://s3.console.aws.amazon.com/s3/buckets/tuneworldmain/TuneWorldFolder/sourabh.namilikonda%2540sjsu.edu/?region=us-east-1&tab=overview

Amazon S3 › tuneworldmain / TuneWorldFolder / sourabh.namilikonda@sjsu.edu

Name	Last modified	Size	Storage class
Happy Birthday Tune.docx	Oct 23, 2017 11:12:21 AM	12.5 KB	Standard

Pre-requisites Set Up: (Please refer to AWS site for how to achieve individual functions)

1. On the Cloud Account first configure an IAM role and logout of root account and login using the IAM role after giving it enough privileges.
2. Choose a region and create S3 buckets giving it the proper settings for Transfer Acceleration, Versioning, Logging and tags. Assign a LifeCycle Policy to move to S3-IA after 75 days and to Glacier after a year. Make the content private.
3. Create another similar bucket in another region and assign as a backup to first using replication policy.

4. Configure CloudFront on top of S3 by creating a new distribution. Now use that link from Cloudfront distribution to download.
5. Create a key/pair to access the EC2 instances remotely. Keep the private key file that you download safely.
6. Create a relational database of choice (MSSQL used here) with a db.t2.micro instance and assign it a username and password. Currently Multi AZ deployment is not turned on but can be done in the detail section.
7. Create Elastic Beanstalk(EB) Application with a new environment with load balancing and auto scaling feature enabled and giving also associate the key which you created earlier. Upload the code in a zip file and start the environment. This creates a Load Balancer and EC2 t2.micro instances along with an AutoScaling Group which will spin up new instances if required. Please explore all possible configurations.
8. Create a time based scaling (optional) to set up a new instance on 1st of every month at 9 am using a CRON job from the scaling section of configuration tab.
9. Add CloudWatch alarms to check if cost of resources exceeds a certain value give your mail id to send notification.
10. Ensure that the Database is able to connect to the EC2 instances by configuring both to be in the same security group.
11. Create a new topic in SNS, subscribe an email and publish a message to check if it is working.
12. Create a lambda function and choose python code and use the given code to copy files of particular type from s3 bucket to another.
13. Test the link provided by EB and check if the uploaded code is running. If there is any change in code then Visual Studio has the option of republishing directly to EB after you install AWS for Visual Studio SDK. Versions are maintained for eac of your changes.
14. Create and link a CloudFront distribution to the webserver giving origin source as the loadbalancer. Also ensure that you allow all types of requests to the Cloudfront and whitelist the Authorization headers and parameter query strings.
15. Next to be able to see the website publically register a domain either using Route53 or some other source and link Route53 to it.
16. Create a HostedZone and Record Set and link the CloudFront distribution to it after giving it a proper name. Then test the record set.
17. You can now check the link and ensure it works.
18. Please refer to the Project 1 document for more information and architecture diagram.

List of required software to download locally:

- MS SQL Management Studio
- IIS (Internet Information Server) to host
- Visual Studio 2017
- Browsers

Local configuration and how to deploy application locally:

1. Install VS 2017, MS SQL Server Management Studio, IIS.
2. Download the project from Git and Open project (Solution sln file) on VS and change the following items in the web.config file:
    a. Connection string to database.
    b. AWS Access and Secret keys.
    c. Bucket Name and Object Key Name.
    d. Region name.
    e. SNS topic name.
3. Build (Ctrl + Shift + B) and Run (F5) the project.
4. Main page will be available under /index.html.
5. Deploy to IIS to host locally by first publishing the project and then configuring a new website in IIS.
6. MSSQL Server Management Studio is used to connect to the database on cloud or locally.

Note: Product still under development. So some modifications necessary:

1. Check if App_Data folder is created with proper permissions to IIS in deployed folder.
2. Lambda modifications imminent.