AWS MULTI-TIER ARCHITECTURE

A CAPSTONE PROJECT REPORT
BY

TEAM NO. 45

BODDEDA JAYA SAI AVINASH (E17CSE018)
C R NITHESHWAR (E17CSE045)

SUBMITTED TO

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING
BENNETT UNIVERSITY

GREATER NOIDA, 201310, UTTAR PRADESH, INDIA

in

Partial fulfillment of the requirements
for the degree of

BACHELOR OF TECHNOLOGY

NOVEMBER 2020

CERTIFICATE

This is to certify that the capstone project report entitled "**AWS MULTI-TIER ARCHITECTURE**" is being submitted by **Mr. C R NITHESHWAR** (Enroll. No. E17CSE045) and **Mr. Boddeda Jaya Sai Avinash** (Enroll. No. E17CSE018) to the Department of Computer Science Engineering, Bennett University, Greater Noida, in partial fulfillment of the requirements for the award of Degree of Bachelor of Technology. It is an original research work carried out by him/them as the 7th Semester 12 credit course from June 2020 to November 2020.

The report has fulfilled all the requirements as per the regulations of this institute and has reached the standard needed for submission. The results embodied in this capstone project report has not been submitted to any other university or institute for the award of any other degree or diploma, degree elsewhere.

BODDEDA JAYA SAI AVINASH
(Enroll. No. E17CSE018)
C R NITHESHWAR
(Enroll. No. E17CSE045)

This is to certify that the above statement made by the candidate(s) is correct to the best of my knowledge.

Dr. Indrajeet Gupta (Capstone Project Instructor)
Assistant Professor
Computer Science Engineering Department
Bennett University, Greater Noida, INDIA

DECLARATION

I/We hereby declare that the work which is being presented in the report entitled "**AWS MULTI-TIER ARCHITECTURE**", in partial fulfillment of the requirements for the Bachelor of Technology in Computer Science and Engineering is an authentic record of my/our own work carried out during the period from JUNE, 2020 to November, 2020 at Department of Computer Science and Engineering, Bennett University Greater Noida.

The matters and the results presented in this report has not been submitted by me/us for the award of any other degree elsewhere.

Signature of Candidate

BODDEDA JAYA SAI AVINASH
(Enroll. No. E17CSE018)

C R NITHESHWAR
(Enroll. No. E17CSE045)

ACKNOWLEDGEMENT

I/We would like to take this opportunity to express my/our deepest gratitude to my/our mentor, **Dr. Indrajeet Gupta** (Assistant Professor) for guiding, supporting, and helping me/us in every possible way. I/we was/were extremely fortunate to have him as my/our mentor as he provided insightful solutions to problems faced by me/us thus contributing immensely towards the completion of this capstone project. I/We would also like to express my/our deepest gratitude to VC, DEAN, HOD, faculty members and friends who helped me/us in successful completion of this capstone project.

Signature of Candidate

BODDEDA JAYA SAI AVINASH
(Enroll. No. E17CSE018)

C R NITHESHWAR
(Enroll. No. E17CSE045)

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

Abbreviation	Explanation of the Abbreviation
API	Application programming interface
CSP	Cloud Service Provider
DNS	Domain Name System
IAM	Identity and Access Management
DB	Database
ML	Machine learning
VPN	Virtual private network
EC2	Elastic Cloud Computing
AWS	Amazon web services

ABSTRACT

For large companies, customer churn is one of the most critical issues. The telecommunications industry always suffers from extremely high churn rates when one industry provides a better plan than the previous one it is very difficult to prevent losses, but we can hold it to a minimal level by prediction. Because of the direct impact on the company's sales, companies aim to predict the churn of potential customers. We built a prediction model using an XGboost algorithm and achieved 73% accuracy. The XGBoost algorithm is an algorithm for machine learning and is based on a decision tree using a gradient boosting method. To interact with the model, we created a website using flask and secured with AWS multi-tier Architecture. Total work was done in the AWS cloud we used various services of cloud, like Sagemaker for Machine learning with help of the API we created a user interface for the model and The architecture provides better security to secure a web application. We create multiple tiers. For load balancing we create a separate subnet for publicly accessing servers we create another subnet for databases, and we create a backup of EC2 instances as snapshots with the help of AWS lambda we monitor a website with AWS cloud watch. This helps to secure the website if anyone of the services compromised while an attacker attacks the website, he can't access the other services which are there in other subnets. Still, if any attack happening a notification alert will be sent to the admin. To secure the admin access we use a VPN for admin.

1. INTRODUCTION

The Telecom industry is the main industry in developing countries. The level of competition has been boosted by technological innovation and the rising number of operators. Depending on different tactics, businesses work hard to succeed in this dynamic environment. To generate more money, It has three strategies

- gain new customers
- upselling current clients
- increase the customers' retention period

However, the analysis of these strategies, taking into account the value of return on investment (RoI) for each, showed that the third strategy is the most successful strategy, indicating that in addition to being considered much simpler than the upselling strategy, retaining a current customer costs much lower than acquiring a new one. In order to enforce the third strategy, companies have to minimize customer churn power, which is characterized as the transition of customers from one supplier to another.

1.1. Problem Statement

For large companies, customer churn is one of the most critical issues. The telecommunications industry always suffers from extremely high churn rates when one industry provides a better plan than the previous one it is very difficult to prevent losses, but we can hold it to a minimal level by prediction. Because of the direct impact on the company's sales, companies aim to predict the churn of potential customers.

Therefore, it is crucial to find ways that increase customer churn to take the appropriate measures to minimize it.

2. BACKGROUND RESEARCH

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/concepts.html

In this documentation, we can get the information about Ec2 instance how to create the instance and AMIs(amazon machine images) and the website should available for all regions we can know that through this documentation and to secure the EC2 instance we have to monitor and configure some security and this documentation will help us how to store the data from this instance and this documentation helps us to make EC2 instance available in the network by assigning an IP address and creating VPC.

https://d0.awsstatic.com/whitepapers/aws-web-hosting-best-practices.pdf

This PDF helps us how to host a web application in AWS it gives an idea of how to host a website and what security it needs. Aws can solve common web application hosting issues. It gives us guidance on how to balance the traffic and how to minimize the cost when resources are not in use and how to balance the load.

https://www.techtruffle.com/blog/aws/three-tier-architecture/

This blog helps us to build architecture as the architecture provides better security to secure a web application. We create multiple tiers. For load balancing we create a separate subnet for publicly accessing servers we create another subnet for databases, and we create a backup of EC2 instances as snapshots with the help of AWS lambda we monitor a website with AWS cloud watch. This helps to secure the website if anyone of the services compromised while an attacker attacks the website, he can't access the other services which are there in other subnets. Still, if any attack happening an SMS alert will be sent to the admin. To secure the admin access we use a VPN for admin.

https://docs.aws.amazon.com/wellarchitected/latest/security-pillar/welcome.html

This documentation helps us to understand security which is required for this project we can learn all the security issues and handle them with this documentation. This documentation provided all the design principles and how to operate Your Workload Securely And helps in creating IAM users, so the services can’t be interrupted unnecessarily. This documentation helps us to protect the data which is stored database it encrypts the network also for admin it will be more secure to access all the services.

https://www.kaggle.com/shrutimechlearn/churn-modelling

We have collected the dataset from this website this dataset includes information about the customers of a bank and the target variable is a binary variable that represents whether the customer has left the bank (closed his account) or is still a customer. It has 10000 Rows and 14 columns, Customer Id - Unique identification for customers, Surname- customer's last name It has 2932 unique values, Credit Score – credit score of the customer lowest score of a customer is 350, highest score of a customer is 850 and an average score of a customer is 652, In geography column, it has country from where the customer belongs. It contains 3 countries and mostly fifty percent of the data is from France. In the gender column, it has 45 percent of females and 55 percent of males. The lowest age of the customer is 18 and the highest is 92 mostly the age of the customer is 37. In the tenure section, it has the lowest value of 0, the highest value of 10, and an average value of 5. In the Bank balance of the customer section the lowest value of 0, the highest value of 251k, and the average value of 97.2k. In several products, the customer is utilizing the lowest value of 1, the highest value of 4, and an average value of 2.

https://aws.amazon.com/getting-started/hands-on/build-train-deploy-machine-learning-model-sagemaker/

From this blog, we have learned about Sagemaker and how to build the model, train the model, and deploy a Machine learning model. In this blog they used the XGBoost algorithm so we got an exercise kind of example to understand better about the algorithm by following this blog you will know how to create and use a Sagemaker notebook instance it launches a Jupyter notebook most of the libraries are preinstalled then you will prepare the data in Sagemaker where you can use your preprocess techniques then to give the result you have to train the model to learn from the dataset and then deploy the ML model after doing that you will get an endpoint where you can use this for interacting with other cloud services and evaluate your model performance.

https://docs.aws.amazon.com/lambda/latest/dg/lambda-python.html

From documentation you will learn how to use lambda after creating an endpoint in Sagemaker to trigger the model lambda will help it automatically triggers whenever the user wants to interact. You will connect this lambda and API to do that.

https://docs.aws.amazon.com/lambda/latest/dg/services-APIgateway.html

From this documentation you will learn how to use API and How to combine with lambda you will create a rest API and with the help of lambda you will communicate with your model.

2.1. Proposed System

In this project, we have predicted the churn of the company's customers by using machine learning techniques. With the help of the API and Lambda, we created a user interface and provided security for the website with AWS multi-tier Architecture. Our project is more focused on the architecture to provide better security and high availability for the website. Our main backend part was a Machine learning model which predicts the churn of customers. We first preprocessed the data by cleaning the data and checked any duplicate or null values are present or not then visualized the data then we used label encoder to convert the strings to numerical values then we used feature scaling and created an endpoint to interact with other services and we created lambda function to trigger the ML model and we created rest API to create the website. This whole thing was secured with the architecture.

The architecture provides better security to secure a web application. We create multiple tiers. For load balancing we create a separate subnet for publicly accessing servers we create another subnet for databases, and we create a backup of EC2 instances as snapshots with the help of AWS lambda we monitor a website with AWS cloud watch. This helps to secure the website if anyone of the services compromised while an attacker attacks the website, he can't access the other services which are there in other subnets. Still, if any attack happening an SMS alert will be sent to the admin. To secure the admin access we use a VPN for admin.

2.2. Goals and Objectives

Table 1: Goal and Objectives

#	Goal or Objective
1	Build a cloud Native application
2	Make the system secure and high availability
3	Use various AWS services to build better application
4	Learn about AWS cloud for better career in AWS
5	Integrate all services without errors

3. PROJECT PLANNING

3.1. Project Lifecycle

As the cloud is more focused on the agile approach it becomes easy to follow an agile approach. Agile mainly has four phases in the initiation phase you will define your project defining your problem statement for the project and the constraints. In the second phase, we need detailed planning like cost estimations and scheduling of tasks. In the third phase, we need to execute we need to develop and manage the project with the project team. In the fourth phase, we need to monitor and control the project and we close the project and take reviews. In the cloud, we have a lifecycle that has Tenant provisioning, Tenant configuration, User provisioning, Resource provisioning, User de-provisioning, and Tenant de-provisioning.

3.2. Project Setup

Table 2: project setup

#	Decision Description
1	Windows 10, Python, AWS, Git, Flask, Postman
2	Standards that must be followed (default Capstone coding standard, etc.)
3	Register for Domain Name System
4	A virtual server image will be set up in AWS that will host the website

3.3. Stakeholders

Table 3: Stakeholders

Stakeholder	Role
Dr. Indrajeet Gupta	Mentor & Instructor
C R Nitheshwar	Team member
B.J.S Avinash	Team member

3.4. Project Resources

Table 4: project resources

Resource	Resource Description	Quantity
Capstone Team	Our team of students who will be the primary developers of the project.	2
Dr. Indrajeet Gupta	The mentor who will be able to provide us with technical assistance.	1
Windows virtual server	To host a website	1
Postman	A software where gives the access to the AWS API gateway to give the result	1

3.5. Assumptions

Table 5: Assumptions

#	Assumption
A1	The capstone team and mentors will be able to meet face to face once a week.
A2	AWS will be available for the team to work with as a trial for the first year of the project.
A3	Team members will be able to familiarize themselves with the various AWS services
A4	DNS will be available as a free trail
A5	Integrate all services without errors

4. PROJECT TRACKING

4.1. Tracking

Table 6: Tracking

Information	Description	Link
Code Storage	Project code will be stored in AWS sagemaker and github.	Link
Project Documents and Assignments	Weekly reports, specification and design documents, etc. will be stored in our Onedrive.	Link
Dataset	Project dataset was stored in S3 bucket and github	Link

4.2. Communication Plan

Table 7: Regularly Scheduled Meetings

Meeting Type	Frequency/Schedule	Who Attends
Conference Call/Skype	Weekly	Project team and Instructor
Team Meeting	Weekly	Project team
Sprint Planning Meeting	Start of each sprint	Project team and mentor
Review Meeting	End of each milestone	Project team, ***instructor***

Table 8: Information to Be Shared Within Our Group

Who?	What Information?	When?	How?
Project team	Task assignments & General scrum information	Weekly	Team meetings,call.

Table 9: Information to Be Provided to Other Groups

Who?	What Information?	When?	How?
Instructor	Final deliverables	At completion of project	Project specification doc., code, Power Point presentation
Instructor and mentor	Monthly milestone report	monthly	Email and google forms

Table 10: Information Needed From Other Groups

Who?	What Information?	When?	How?
Instructor and mentor	Requirement changes	Start of each sprint	Conference call or meeting with Instuctor and mentor.
Hariharan	Web development	start of third sprint	Email

4.3. Deliverables

Table 11: Deliverables

#	Deliverable
1	Code
2	Test and test results
3	Install process documents
4	Web application
5	Final report (final PowerPoint presentation, 3 minute video, and final sprint)

5. SYSTEM ANALYSIS AND DESIGN

5.1. Overall Description

Our project is more focused on the architecture to provide better security and high availability for the website. Our main backend part was a Machine learning model which predicts the churn of customers. We first preprocessed the data by cleaning the data and checked any duplicate or null values are present or not then visualized the data then we used label encoder to convert the strings to numerical values then we used feature scaling and created an endpoint to interact with other services and we created lambda function to trigger the ml model and we created rest API to create the website. This whole thing was secured with the architecture.

The XGBoost algorithm is an algorithm for machine learning and is based on a decision tree using a gradient boosting method. To interact with the model, we created a website using flask and secured with AWS multi-tier Architecture. Total work was done in the AWS cloud we used various services of cloud, like Sagemaker for Machine learning with help of the API we created a user interface for the model and The architecture provides better security to secure a web application. We create multiple tiers. For load balancing we create a separate subnet for publicly accessing servers we create another subnet for databases, and we create a backup of EC2 instances as snapshots with the help of AWS lambda we monitor a website with AWS cloud watch. This helps to secure the website if anyone of the services compromised while an attacker attacks the website, he can't access the other services which are there in other subnets. Still, if any attack happening a notification alert will be sent to the admin. To secure the admin access we use a VPN for admin.

5.2. Users and Roles

Table 12: Users and Roles

User	Description
Developer	A capstone team member or mentor who is tasked with managing the test data, creating initial machine learning models, and ultimately generating a firm process for applying these techniques to future user data. This is used for sub-stories and task needed to fulfill the true end user use cases.
Architecture	A capstone team member who is tasked with building a architecture for security and high availability

5.3. Design diagrams/ UML diagrams/ Flow Charts/ E-R diagrams

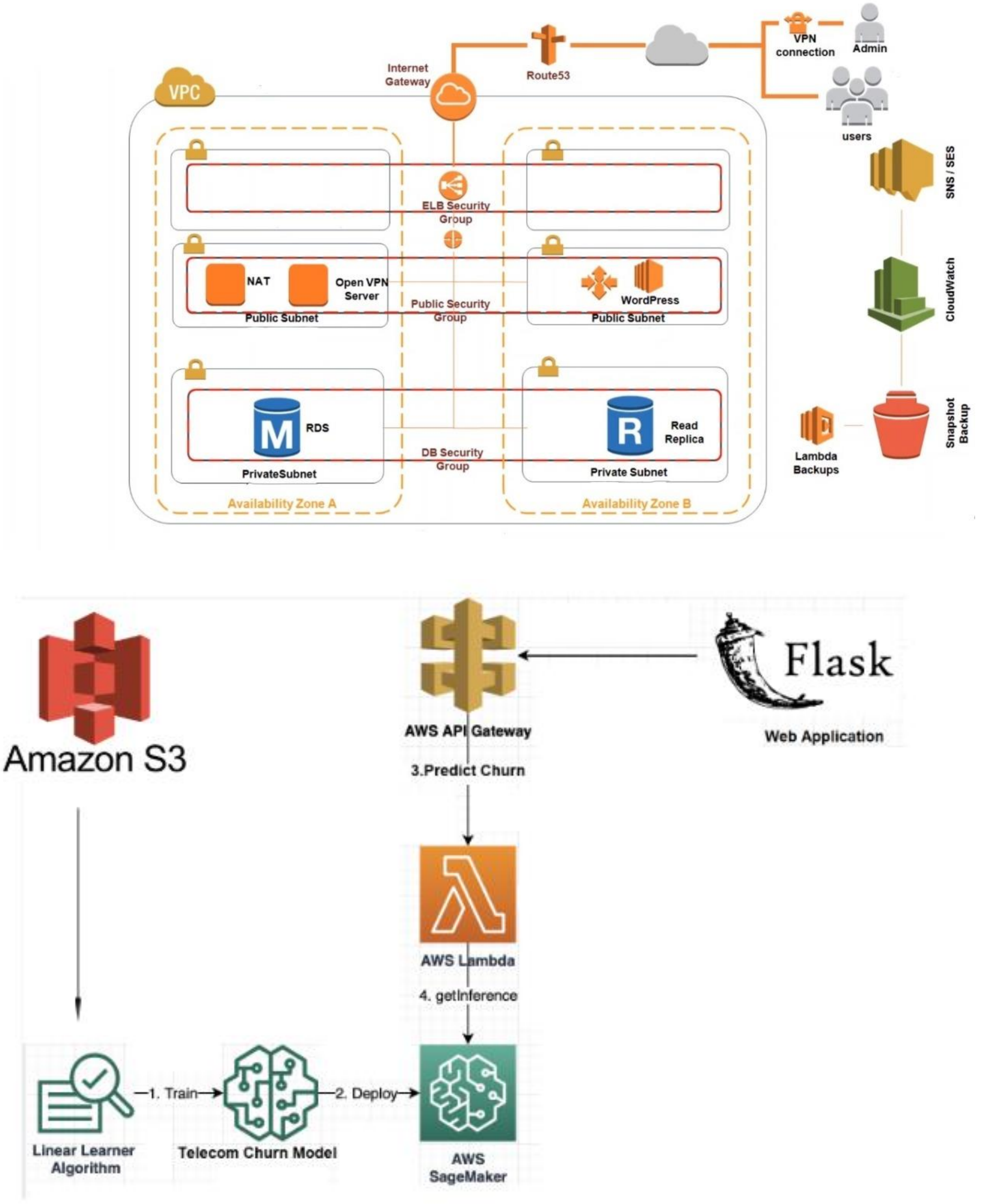


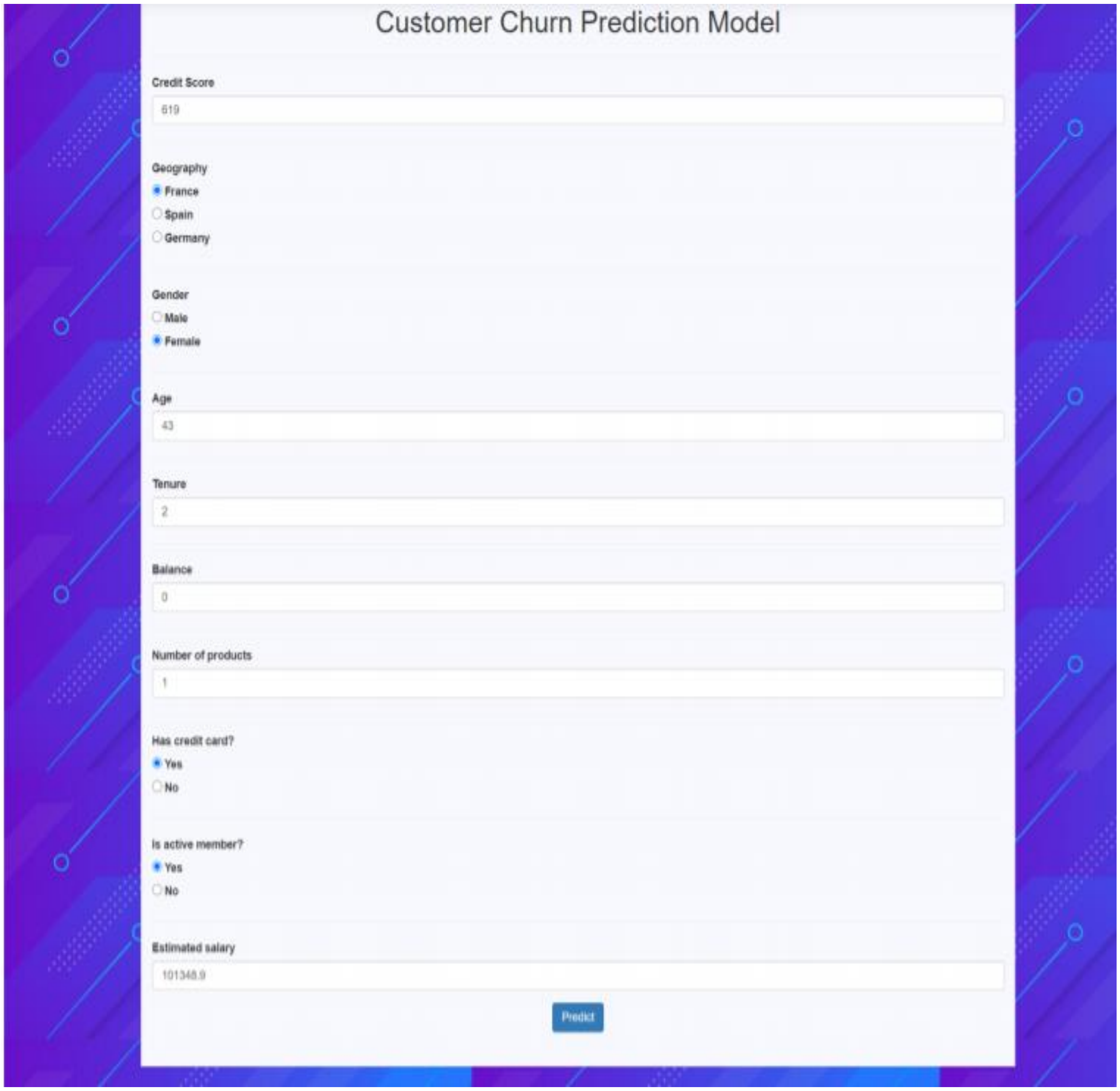
Figure 1 : Architecture diagram

6 USER INTERFACE

6.1 UI Description

We are creating a project using Python and AWS Sagemaker with the help of the flask we have developed a website It has a user interface we have to give the details of the customer to predict the value whether the customer will leave the company or not. It was a form used simple CSS and HTML after filling the form it shows the output by communicating through API.

6.2 UI Mockup



Customer Churn Prediction Model

Credit Score

619

Geography

● France
○ Spain
○ Germany

Gender

○ Male
● Female

Age

43

Tenure

2

Balance

0

Number of products

1

Has credit card?

● Yes
○ No

Is active member?

● Yes
○ No

Estimated salary

101348.9

Figure 1: User Interface

7 ALGORITHMS/PSEUDO CODE

```
# Importing container to run linear learner
from sagemaker.amazon.amazon_estimator import get_image_uri
container = get_image_uri(boto3.Session().region_name, 'xgboost','1.0-1')
output_location = 's3://{}/{}/output'.format(bucket, prefix)
s3_input_train=sagemaker.s3_input(s3_data='s3://{}/{}/train'.format(bucket
, prefix),content_type='text/csv')
print(output_location)
print(s3_input_train)
sess = sagemaker.Session()


linear=sagemaker.estimator.Estimator(container,
 role,
 train_instance_count=1,
 train_instance_type='ml.m5.large',
 output_path=output_location,
 sagemaker_session=sess)


linear.set_hyperparameters(max_depth=5,
 beta=0.2,
 gamma=4,
 min_child_weight=6,
 subsample=0.8,
 silent=0,
 objective='binary:logistic',
 num_round=100)


linear.fit({'train':s3_input_train})
```

8 PROJECT CLOSURE

8.1 Goals / Vision

Our original goals for this project were to build the architecture that provides better security to secure a web application. We create multiple tiers. For load balancing we create a separate subnet for publicly accessing servers we create another subnet for databases, and we create a backup of EC2 instances as snapshots with the help of AWS lambda we monitor a website with AWS cloud watch. This helps to secure the website if anyone of the services compromised while an attacker attacks the website, he cannot access the other services which are there in other subnets. Still, if any attack happening an SMS alert will be sent to the admin. To secure the admin access we use a VPN for admin.

8.2 Delivered Solution

Our solution delivered an architecture with most priority to security and high availability then we developed a website with a Machine learning model using AWS Sagemaker. We first preprocessed the data by cleaning the data and checked any duplicate or null values are present or not then visualized the data then we used label encoder to convert the strings to numerical values then we used feature scaling and created an endpoint to interact with other services and we created lambda function to trigger the ml model and we created rest API to create the website. This whole thing was secured with the architecture.

8.3 Remaining Work

We created a website with a simple UI using HTML and CSS we want to make the website design more attractive and user friendly and we got a low accuracy while training the model so we want to increase the accuracy of the model and our main goal was to use as many as services of aws to build the better application so we want to add other services like AI-powered attendance system in our website it uses the AWS recognition service we have to train the images of people which we want to detect to automatically mark attendance we have to use lambda function it triggers whenever it scans and uploads in dynamo DB to record and count the person's attendance.

REFERENCES

1.Gerpott TJ, Rams W, Schindler A. Customer retention, loyalty, and satisfaction in the German mobile cellular telecommunications market. Telecommun Policy. 2001; 25:249–69.

2. NGURAH PUTU OKA H, AJIB SETYO ARIFIN, "TELECOMMUNICATION SERVICE SUBSCRIBER CHURN LIKELIHOOD PREDICTION ANALYSIS USING DIVERSE MACHINE LEARNING MODEL", *MECHANICAL ELECTRONICS COMPUTER AND INDUSTRIAL TECHNOLOGY (MECNIT) 2020 3RD INTERNATIONAL CONFERENCE ON*, PP. 24-29, 2020.

3. Wouter Verbeke, David Martens, Christophe Mues, Bart prediction models with advanced rule induction (2011) 2354–2364.

4. Yu W, Jutla DN, Sivakumar SC. A churnstrategy alignment model for managers in mobile telecom. In: Communication networks and services research conference, vol. 3. 2005. p. 48–53.

5. MAKHTAR M, NAFIS S, MOHAMED M, AWANG M, RAHMAN M, DERIS M. CHURN CLASSIFICATION MODEL FOR LOCAL TELECOMMUNICATION COMPANY BASED ON ROUGH SET THEORY. J FUNDAM APPL SCI. 2017;9(6):854–68.

6. Zhao Y, Wang G, Yu PS, Liu S, Zhang S. Inferring social roles and statuses in social networks. In: KDD 201319th ACM SIGKDD international conference on knowledge discovery and data mining. Association for Computing Machinery; 2013. p. 695–703

7. Xie J, Rojkova V, Pal S, Coggeshall S. A combination of boosting and bagging for kdd cup 2009fast scoring on a large database. J Mach Learn Res Proc Track. 2009; 7:35–43.