

Minor-Project Report
On
“Weather Detection App using Cloud Techniques”

Project report submitted
in partial fulfilment of the requirement for the degree of

**Bachelor of
Technology**
Submitted to

KIIT (Deemed to be University)

Submitted By-

Ayush Agarwal	1505105
Kishor Kumar Nanda	1505113
Shashank Awasthi	1505143
Sonu Kumar Tiwari	1505076
Suraj Verma	1505162

Under The Guidance of

Prof. N. Biraja Isac



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY
Deemed to be University, BHUBANESWAR
30th April 2018

CERTIFICATE

It is certified that the work contained in the project report titled **“Weather detection app using cloud techniques”** by **“Ayush Agarwal (1505105), Kishor Kumar Nanda (1505113), Shashank Awasthi (1505143) ,Sonu Kumar Tiwari (1505076) , Suraj Verma (1505162) ”** has been carried out under my/our supervision and that this work has not been submitted elsewhere for a degree

Signature of Supervisor
Prof. N. Biraja Isac
School of computer science
KIIT Bhubaneswar
April, 2018

Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Ayush Agarwal

Kishor Kumar Nanda

Shashank Awasthi

Sonu Kumar Tiwari

Suraj Verma

30th april 2018

Acknowledgements

We are profoundly grateful to **Prof N. Biraja Isac** for his expert guidance and continuous encouragement through to see that this project reaches its target since its commencement to its completion.....

Ayush Agarwal

Kishor Kumar Nanda

Shashank Awasthi

Sonu Kumar Tiwari

Suraj Verma

ABSTRACT

User will interact with the application to know the current weather conditions of any location he /she wants to gather information about. The user can make plans according to the weather information gathered of the place. For example, a long dry period is forecast, the watering system authorities can react appropriately and available sun protection systems can be brought into position. If a storm is forecast, the necessary protective measures can be taken accordingly.

The application that we have developed gathers information of weather conditions of a particular place based on the user's query. The application displays information about temperature, humidity, pressure, cloud cover over a particular place. Our application is deployed on AWS cloud Platform. Our application utilizes the AWS services for backend features such as messaging and analytics. Amazon pinpoint service provides detailed analytics of our application usage and provides visual representation via graphs. It also enables us to send notifications to our targeted group or a single individual based on configured campaign or direct messaging to notify users.

Keywords: Android, Weather, Cloud Computing, AWS, Mobile Hub, Pinpoint, Firebase

Contents

1 Introduction

1.1 Terminologies	7
1.1.1 Cloud computing.....	7
1.1.2 Android Application.....	8
1.1.3 Amazon Web Services(AWS).....	9

2 Software Requirement Specification

2.1 Tools and Languages	
2.1.1 Android Studio.....	10
2.1.2 Mobile Hub and Pinpoints	11
2.1.3 Java Programming Language	12

3 Requirement Analysis

3.1 Functional Requirements.....	13
3.2 Non-Functional Requirements.....	14

4 System Design

4.1 Screenshots	15
4.2 UML Diagrams.....	19

5 Steps Involved

5.1 Connecting to AWS Backend	20
5.1 Capturing Analytics and Monitoring.....	21
5.1 Adding Notifications.....	21

6 Conclusion and Future Scope

6.1 Conclusion.....	23
6.2 Future Scope	23

References

Chapter 1

Introduction

1.1 Terminologies

1.1.1 Cloud Computing

Cloud computing is an information technology (IT) paradigm that enables ubiquitous access to shared pools of configurable system resources and higher-level services that can be rapidly provisioned with minimal management effort, often over the internet. Cloud computing relies on sharing of resources to achieve coherence and economies of scale, similar to a public utility.

Third-party clouds enable organizations to focus on their core business instead of expending resources on computer infrastructure and maintenance. Advocates note that cloud computing allows companies to avoid or minimize up-front IT infrastructure costs. Proponents also claim that cloud computing allows enterprises to get their applications up and running faster, with improved manageability and less maintenance, and that it enables IT teams to more rapidly adjust resources to meet fluctuating and unpredictable demand. Cloud providers typically use a "pay-as-you-go" model, which can lead to unexpected operating expenses if administrators are not familiarized with cloud-pricing models.

Benefits of cloud computing

- Trade capital expense for variable expense
- Benefit from massive economies of scale
- Stop guessing capacity
- Increase speed and agility
- Stop spending money on running and maintaining data centers
- Go global in minutes

1.1.2 Android Application

Android is an open source and Linux-based Operating System for mobile devices such as smartphones and tablet computers. Android was developed by the Open Handset Alliance, led by Google, and other companies. Android offers a unified approach to application development for mobile devices which means developers need to develop only for Android, and their applications should be able to run on different devices powered by Android. The first beta version of the Android Software Development Kit (SDK) was released by Google in 2007, whereas the first commercial version, Android 1.0, was released in September 2008. On June 27, 2012, at the Google I/O conference, Google announced the next Android version, 4.1 Jelly Bean. Jelly Bean is an incremental update, with the primary aim of improving the user interface, both in terms of functionality and performance. The source code for Android is available under free and open source software licenses. Google publishes most of the code under the Apache License version 2.0 and the rest, Linux kernel changes, under the GNU General Public License version 2.

Benefits of Android Development

- Open Source
- Customizable user interface
- Low investment and High ROI
- Multiple sales channels
- Easy to adopt

1.1.3 Amazon Web Services

Amazon Web Services (AWS) is a subsidiary of Amazon.com that provides on demand delivery to individuals, companies and governments, on a paid subscription basis. The technology allows subscribers to have at their disposal a full-fledged virtual computer clusters, available all the time, through the Internet. AWS's version of virtual computers have most of the attributes of a real computer including hardware (CPU(s) & GPU(s) for processing, local/RAM memory, hard-disk/SSD storage); a choice of operating systems; networking; and pre-loaded application software such as web servers, databases, CRM, etc. Each AWS system also virtualizes its console I/O (keyboard, display, and mouse), allowing AWS subscribers to connect to their AWS system using a modern browser. The browser acts as a window into the virtual computer, letting subscribers log-in, configure and use their virtual systems just as they would a real physical computer. They can choose to deploy their AWS systems to provide internet-based services for their own and their customers' benefit.

The AWS technology is implemented at server farms throughout the world, and maintained by the Amazon subsidiary. Fees are based on a combination of usage, the hardware/OS/software/networking features chosen by the subscriber, required availability, redundancy, security, and service options. Based on what the subscriber needs and pays for, they can reserve a single virtual AWS computer, a cluster of virtual computers, a physical (real) computer dedicated for their exclusive use, or even a cluster of dedicated physical computers. As part of the subscription agreement, Amazon manages, upgrades, and provides industry-standard security to each subscriber's system. AWS operates from many global geographical regions including 6 in North America.

In 2017, AWS comprised more than 90 services spanning a wide range including computing, storage, networking, databases, analytics, application services, deployment, management, mobile, developer tools, and tools for the Internet of Things. The most popular include Amazon Elastic Compute Cloud (EC2) and Amazon Simple Storage Service (S3). Most services are not exposed directly to end users, but instead offer functionality through APIs for developers to use in their applications. Amazon Web Services' offerings are accessed over HTTP, using the REST architectural style and SOAP protocol.

Chapter 2

Software Requirement Specification

2.1 Tools and Languages

2.1.1 Android Studio

Android Studio is the official integrated development environment (IDE) for Google's Android operating system, built on JetBrains' IntelliJ idea software and designed specifically for Android Development. It is available for download on Windows, macOS and Linux based operating systems. It is a replacement for the Eclipse Android Development Tools (ADT) as primary IDE for native Android application development.

Android Studio was announced on May 16, 2013 at the Google I/O conference. It was in early access preview stage starting from version 0.1 in May 2013, then entered beta stage starting from version 0.8 which was released in June 2014. The first stable build was released in December 2014, starting from version 1.0. The current stable version is 3.1 released in March 2018.

Features

The following features are provided in the current stable version:

- Gradle-based build support
- Android-specific refactoring and quick fixes
- Lint tools to catch performance, usability, version compatibility and other problems
- Template-based wizards to create common Android designs and components
- A rich layout editor that allows users to drag-and-drop UI components.
- Built-in support for Google Cloud Platform, enabling integration with Firebase Cloud Messaging (Earlier 'Google Cloud Messaging') and Google App Engine.
- Android Virtual Device (Emulator) to run and debug apps in the Android studio.

2.1.2 AWS_MOBILE HUB and PINPOINT

The AWS platform handles demanding requirements for virtually any application, and for our project we used variety of services provided by the AWS such as the mobile services of AWS known as AWS Mobile Hub for the deployment of our Android application on the AWS cloud. AWS Mobile Hub gives you a single place to easily configure AWS services. It generates a cloud configuration file, which stores information about configured services. The 'awsconfiguration.json' file you download contains the configuration of backend resources that Mobile Hub enabled in our project. We add the backend service configuration file to our app which enables us to connect to AWS services and use the AWS SDKs. The AWS Mobile Android SDK helps to build high quality mobile apps quickly and easily through the integration of various AWS services for better functionality and wide variety of usages. They provide easy access to a range of AWS services, including Amazon Cognito, AWS Lambda, Amazon S3, Amazon Kinesis, Amazon DynamoDB, Amazon Pinpoint and many more.

Amazon Pinpoint helps to engaging by sending email, SMS, and mobile push messages. Whether you're a developer, marketer, or business user, you can use Amazon Pinpoint for target messages (such as promotional alerts), as well as direct messages (such as order confirmations and password reset messages) to your customers.

You can integrate Amazon Pinpoint into your apps to check usage, which helps you better understand how customers interact with your apps. Amazon Pinpoint also tracks the ways that your customers respond to the messages you send.

You can send direct messages—such as order confirmations, welcome messages, and one-time passwords—using the console or the Pinpoint REST API. You can also use the API to build custom applications that deliver campaign and transactional messages across multiple channels.

2.1.3 Java Programming Language

Java is a general-purpose computer –programming language that is concurrent, class based, object oriented, and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers “write once read anywhere” (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to bytecode that can run on any Java Virtual Machine (JVM) regardless of computer architecture. As of 2016, Java is one of the most popular programming language in use, particularly for client-server web applications, with a reported 9 million developers. Java was originally developed by James Gosling at Sun Microsystems (which has since been acquired by Oracle Corporation) and released in 1995 as a core component of Sun Microsystems' Java platform. The language derives much of its syntax from C and C++, but it has fewer low-level facilities than either of them.

The original and reference implementation Java compilers, virtual machines, and class libraries were originally released by Sun under proprietary licenses. As of May 2007, in compliance with the specifications of the Java Community Process, Sun relicensed most of its Java technologies under the GNU General Public License. Others have also developed alternative implementations of these Sun technologies, such as the GNU Compiler for Java (bytecode compiler), GNU Classpath (standard libraries), and IcedTea-Web (browser plugin for applets).

Chapter 3

3.1 Functional Requirements

R1: Android App

Input: Enter Location

Output: Displays temperature, Humidity, Pressure, Cloud Cover

Description: The android app calculates the weather condition based on latitudes and longitudes. The app displays the weather condition of the entered position.

R2: AWS Server

Input: Deployment of the application on its server.

Output: Graph Visualizations of application

Description: Amazon Web Services provides backend services and analyses the application and gives various graph visualization.

R1: User

Input: Enters Location

Output: Gets temperature, Humidity, Pressure, Cloud Cover

Description: The android app calculates the weather condition based on latitudes and longitudes. The app displays the weather condition of the entered position.

3.2 Non-Functional Requirements

3.2.1 Performance Requirements

Performance of the system depends on the response time from the server and also the latency of packet transmission, effectiveness and internet speed. Since, the application is hosted on the AWS cloud, the application is totally dependent on a proper internet connectivity and speed and reliability of the AWS cloud.

3.2.2 Safety Requirements

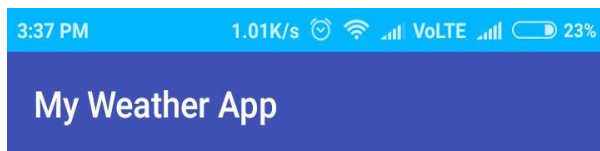
Android application has to check whether the user enters the correct location or not. This would lead to redundant results. AWS Cloud provides a secure deployment of the application code.

Chapter 4

System Design

4.1 Screenshots

Android Application

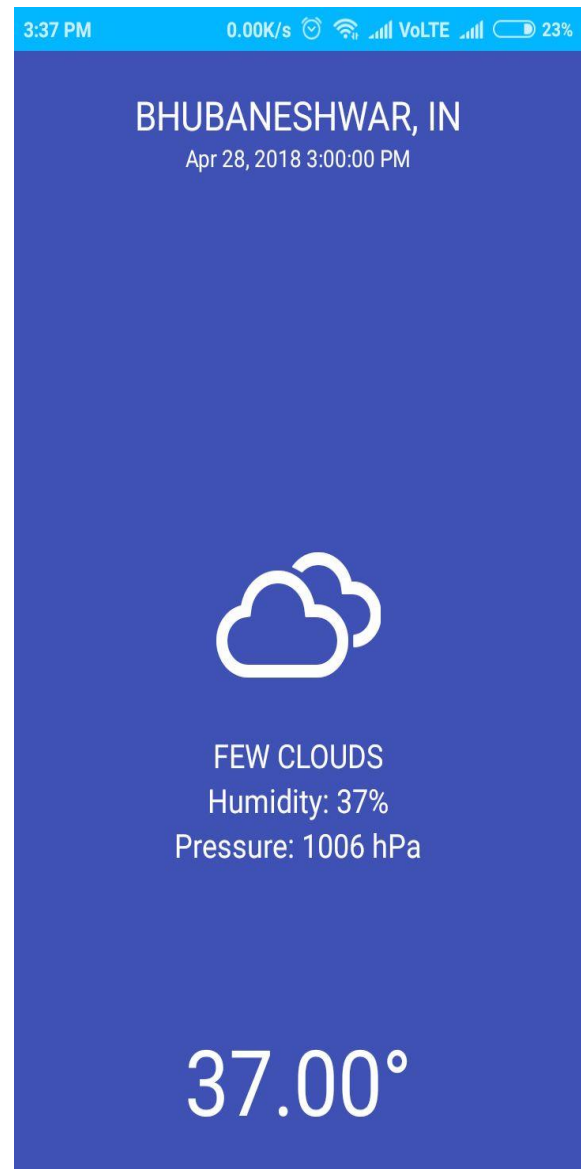


Weather Forecast

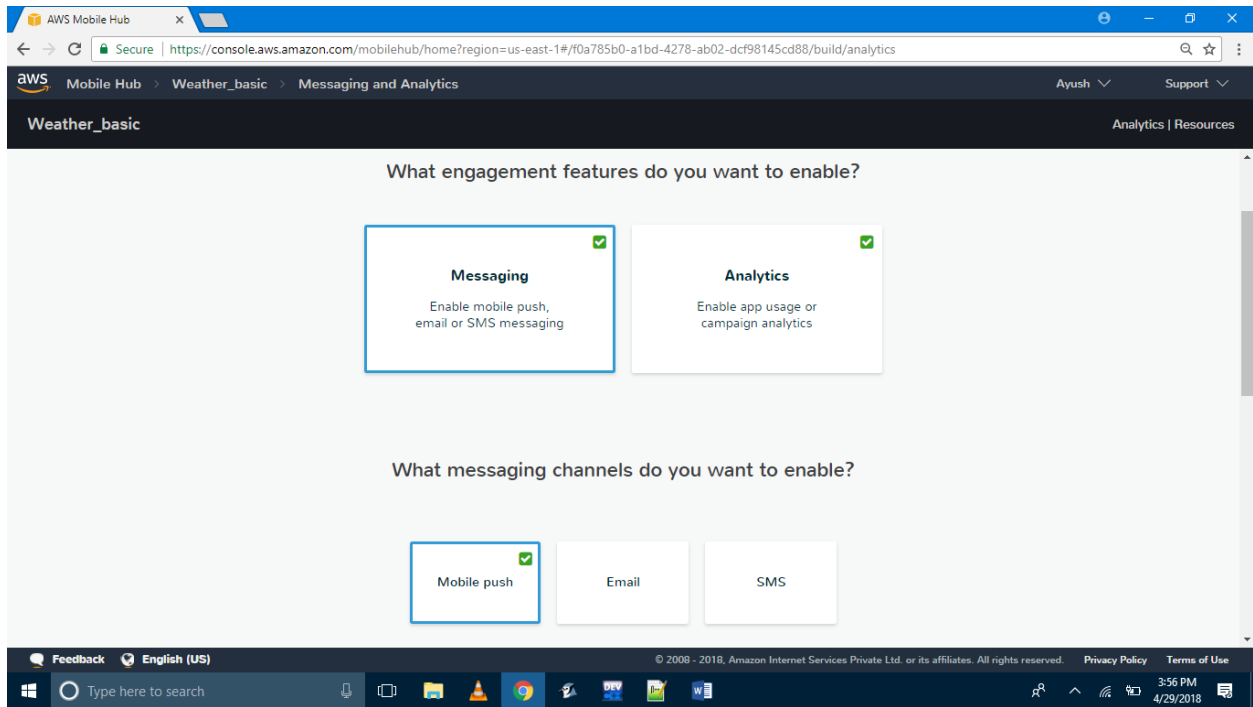


Weather Forecast can provide you with forecast and alerts fo your exact location as you move.Please allow using Location Services to enable this.

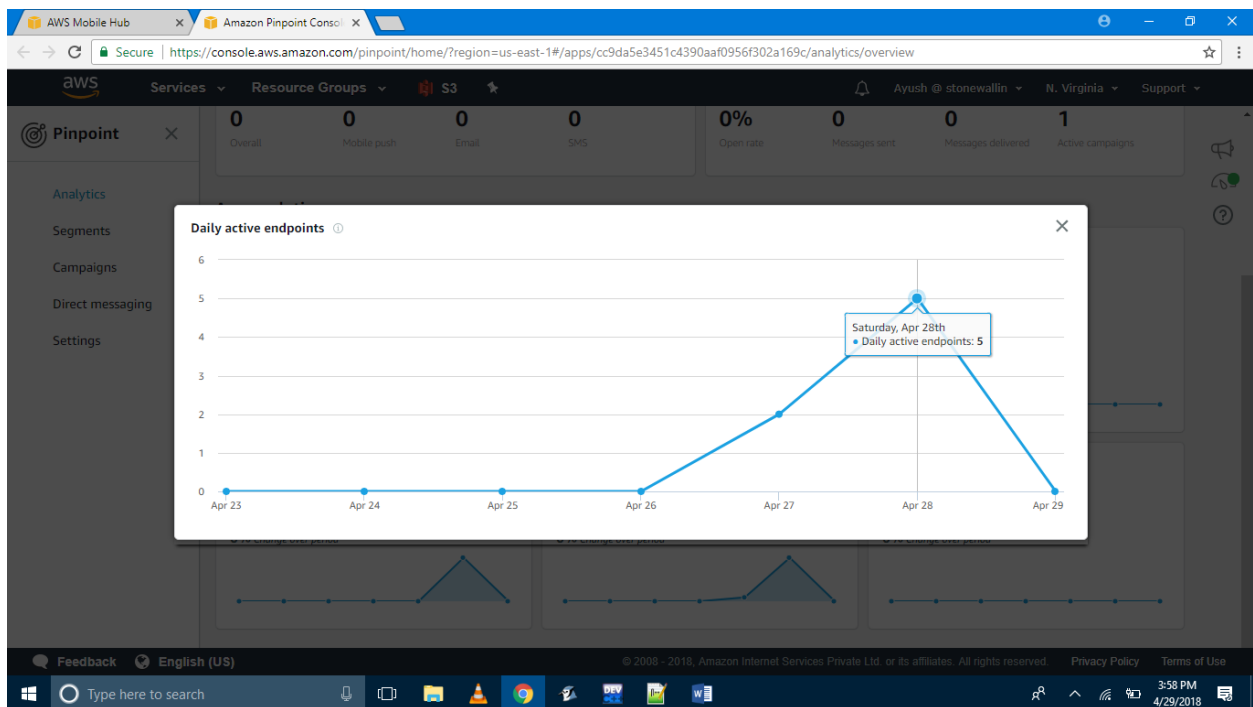
OK LET'S MOVE



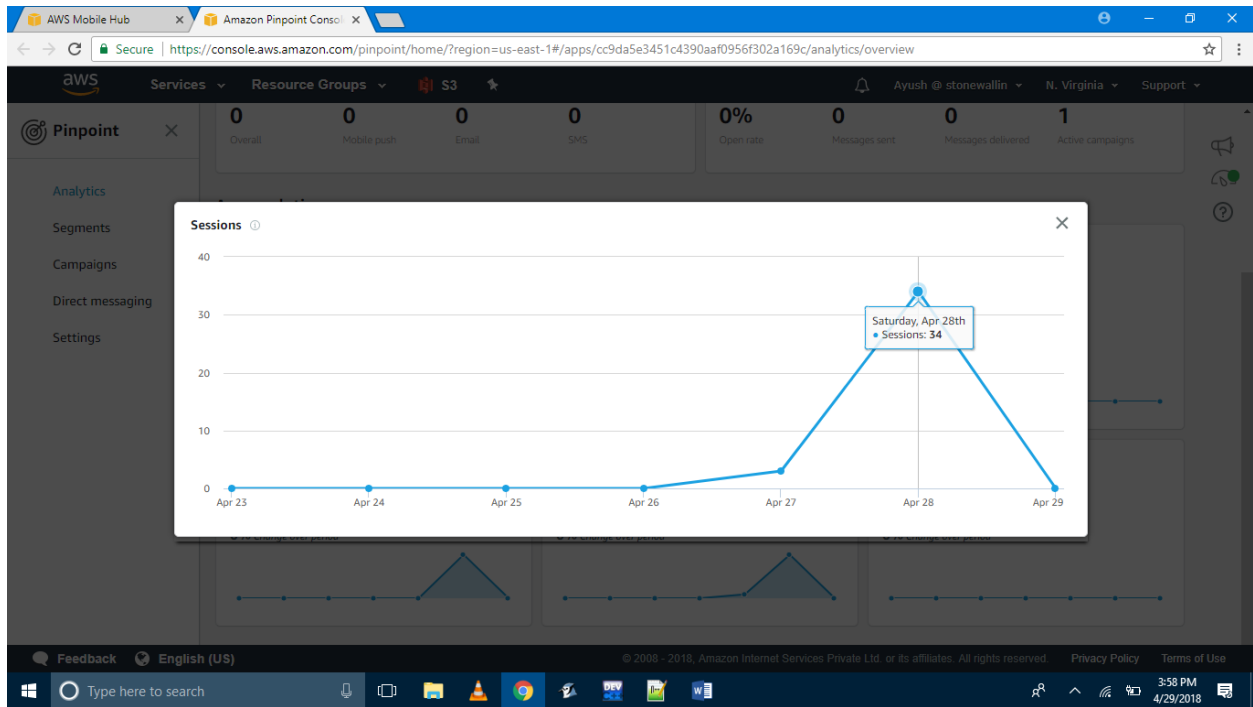
AWS Mobile Hub- Backend and Analytics enabled



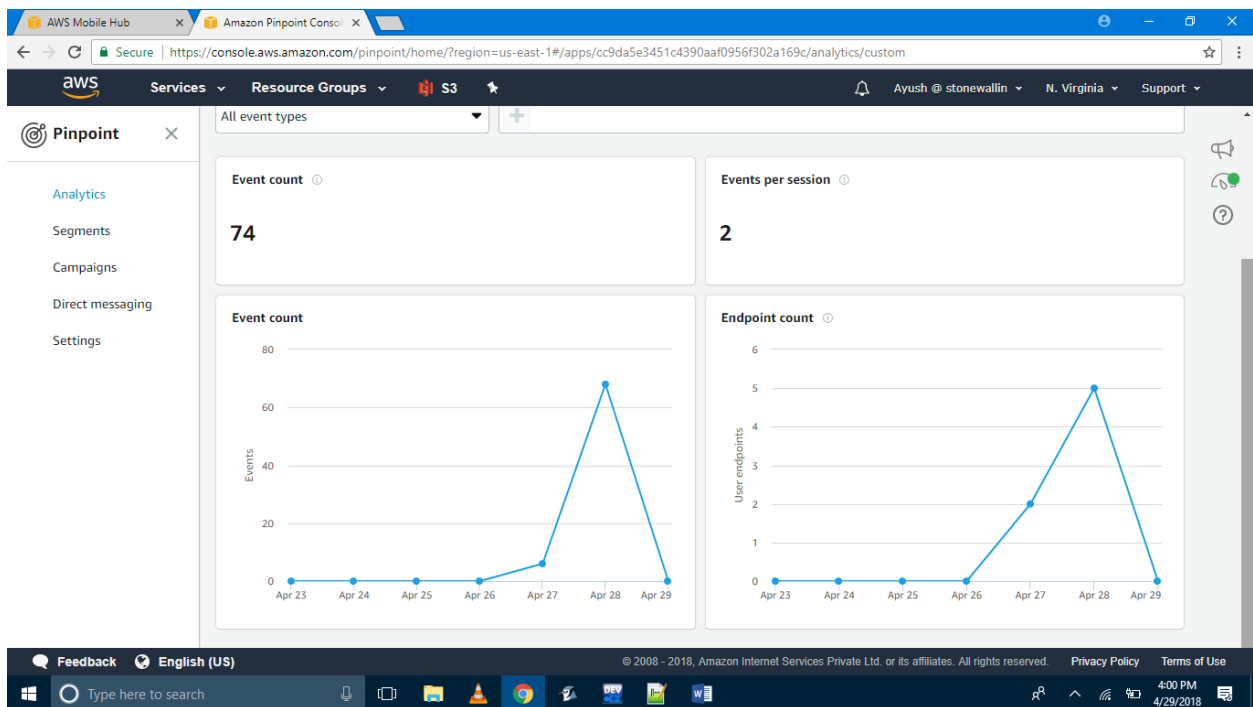
AWS Pinpoint Analytics- Daily Active Endpoint



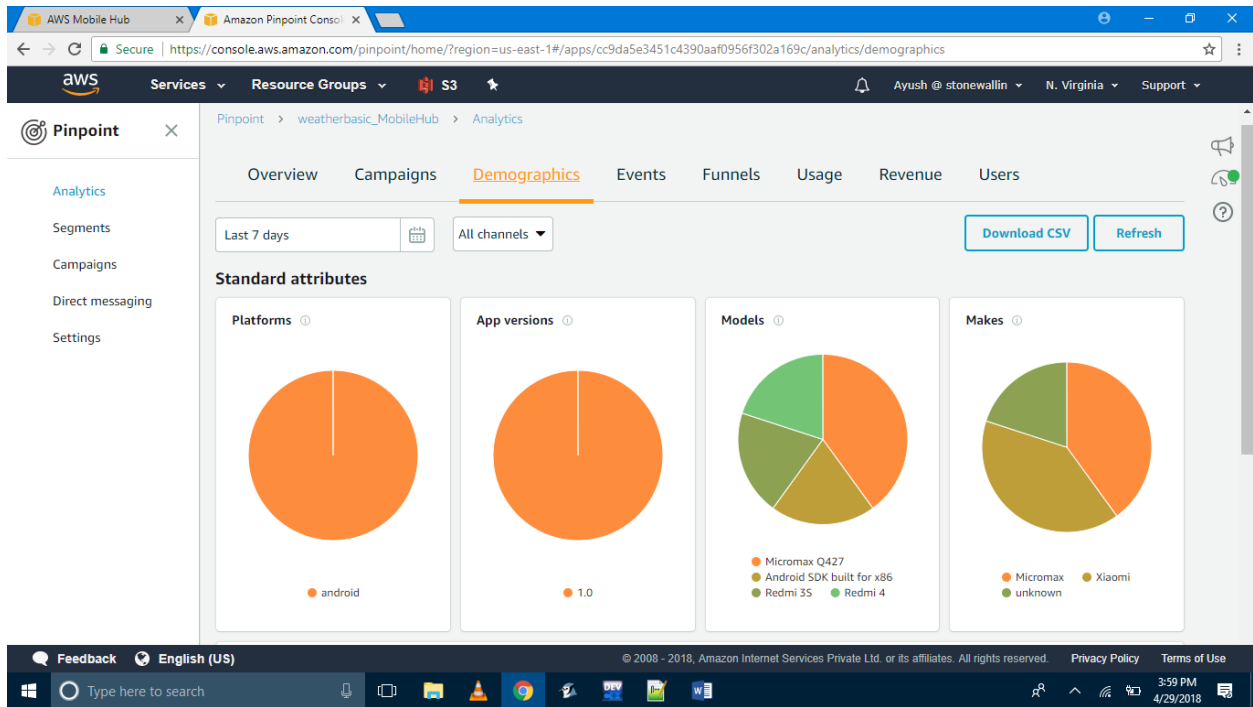
AWS Pinpoint Analytics- User Sessions



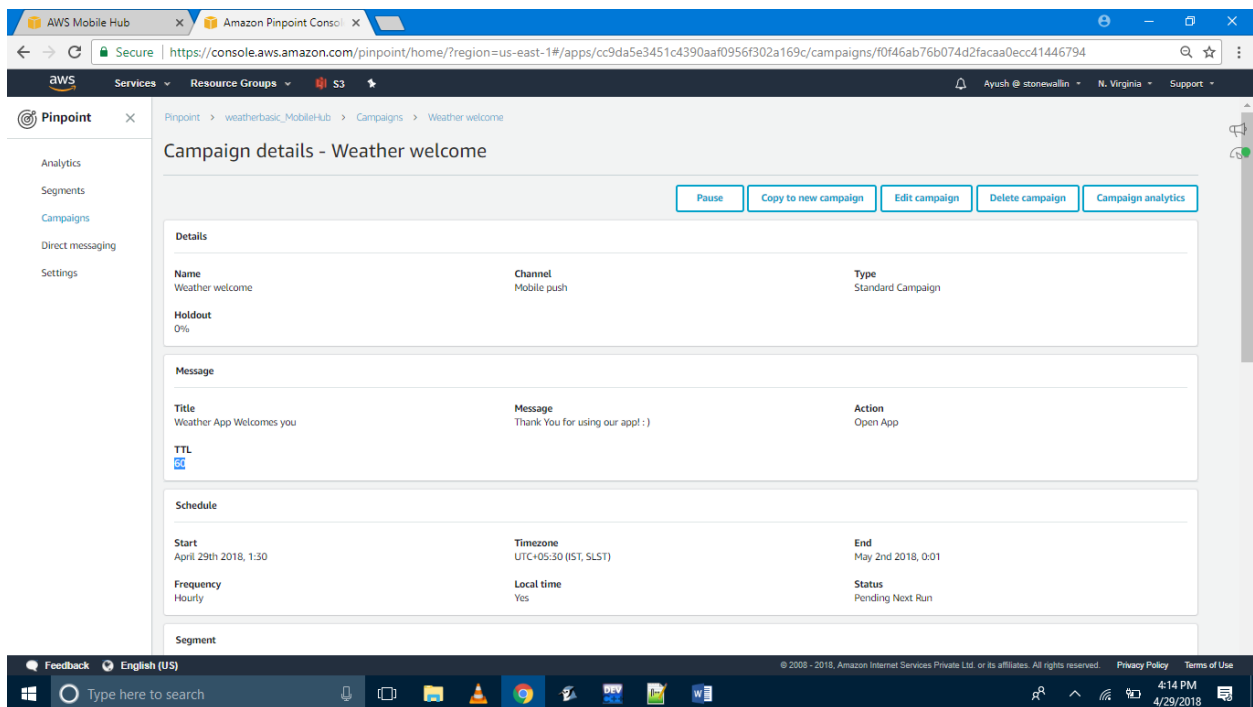
AWS Pinpoint Analytics- Events



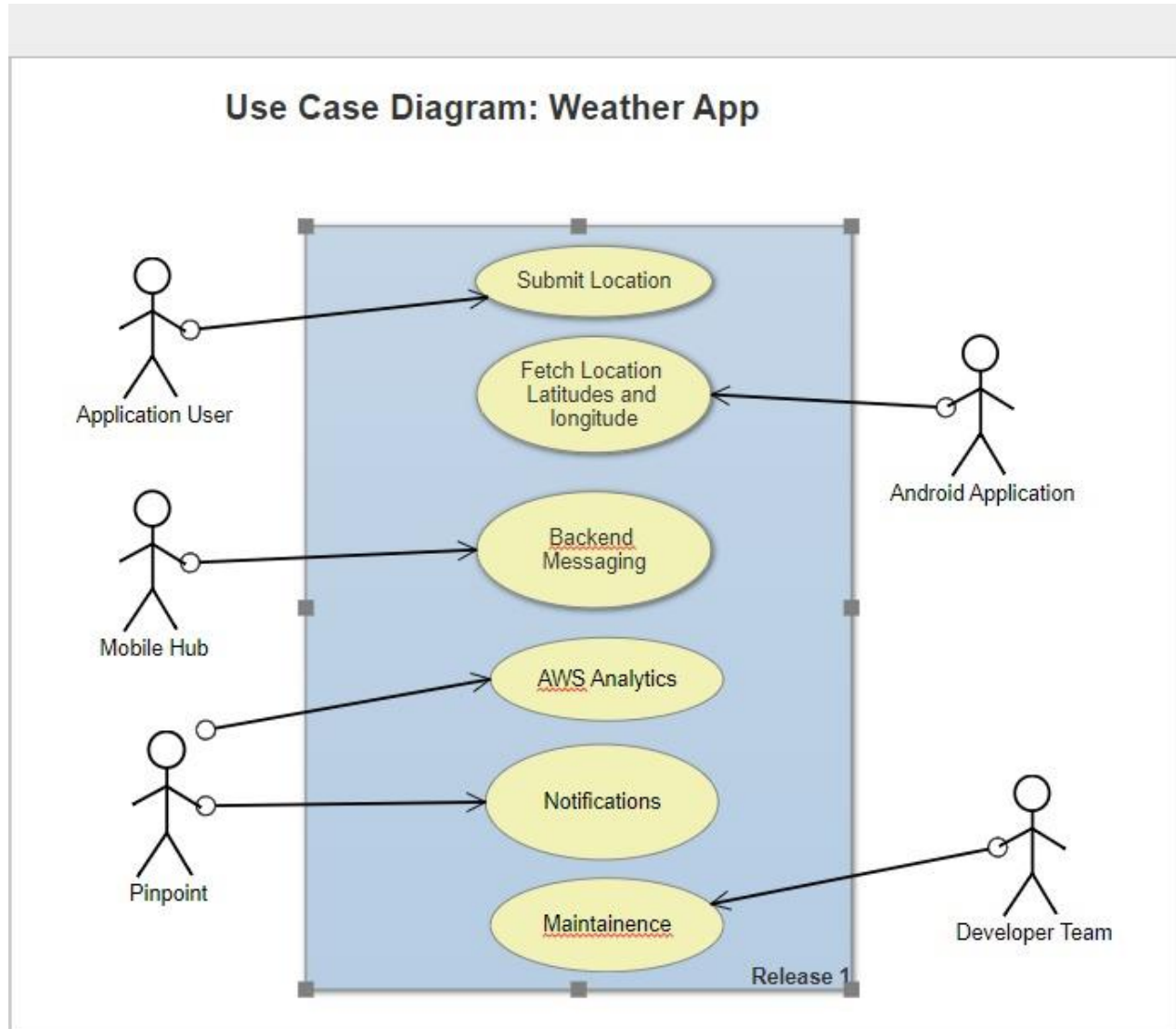
AWS Pinpoint Analytics-Demographics



AWS Pinpoint – Campaign



4.2 UML Use Case Diagram



Chapter 5

Steps Involved

5.1 CONNECTING TO AWS BACKEND

After downloading and adding the cloud configuration file in the 'raw' directory of 'res' of our android project for backend, we connect to the backend services to utilize the benefits of the cloud. The AWS SDK provided code has to be added to the application source code to connect.

Uses permission had to be added to the AndroidManifest.xml file to allow Internet and Network State Access.

Dependencies were added to our app/build.gradle and synched by choosing Sync Now in Android Studio. These dependency libraries enable basic AWS functions, like credentials, and analytics.

AWS provided code had to be added to the onCreate function of our MainActivity and AWSMobileClient class had to be imported.

AWSMobileClient is a singleton that establishes your connection to and acts as an interface for our services.

And the app is set up to interact with the AWS services configured in our Mobile Hub project. Various features can be easily added through the source code and by using the AWS Mobile Hub Management Console.

To check the connectivity to the AWS cloud, we run the application and in our logcat, we search for 'Welcome to AWS!' which ensures our app is successfully able to connect to cloud and can utilize its benefits.

5.2 CAPTURING ANALYTICS AND MONITORING THROUGH PINPOINT ANALYTICS

To gather the data that helps improve our app's usability, sessions, and engagement with our users. Mobile Hub deploys our analytics backend when enabling the Messaging and Analytics feature, which uses the Amazon Pinpoint service. Analytics were by default enabled when adding and connecting to the backend. To ensure that the analytics are enabled or not we check out Mobile Hub console of our application, and a green tick present there ensures it.

Calls to start and stop session for analytics were added using AWS SDK code. Analytics session is started at the beginning of the onCreate() method of our MainActivity. To stop the session, we use stopSession() and submitEvents() at the last point in the session we wanted to capture. PinpointManager and PinpointConfiguration classes were imported for the analytics session capture and sending reports to the console.

Metrics for the application usage can be checked on the console by going to the 'Analytics' tab in Mobile Hub window of our project which opens up the Pinpoint services where our metrics are sent. The metrics that were monitored were 'Daily Active endpoints', 'New Endpoints', 'Monthly Active endpoints', 'Sessions', '7-day retention period' and 'Revenue'.

5.3 ADDING PUSH NOTIFICATIONS THROUGH PINPOINT CAMPAIGN AND DIRECT MESSAGING

To add push notification to our application we use the 'Messaging and Analytics' feature of Amazon Pinpoint Service. Amazon Pinpoint enables apps to receive mobile push messages sent from the Apple (APNs) and Google (FCM/GCM) platforms. For android we have the option of using either the Google Cloud Messaging(GCM) or the Firebase Cloud

Messaging(FCM). We require that in order to obtain the application API key and Sender ID as it is required to send notification to the devices where our application is installed.

We create a new Firebase project for our application that provide us with required application key and Sender ID. After obtaining the required credentials from the Firebase console, we add the application API key and Sender ID under the 'Android' section which is in 'Push notification' option in Messaging. When the operation is complete, an alert will pop up saying "Your Backend has been updated", prompting to download the latest copy of the cloud configuration file. After completion of configuring the feature, we choose the pop-up banner to return to the project details page and click on the 'Integrate' button.

We have to update our application with the latest configuration file and then we need to connect to the backend service using the AWS Mobile SDKs. AndroidManifest.xml file needs to be edited and uses-permission have to added to be able to receive notification. Dependencies have to be added in app/build.gradle and also project level build.gradle. We have to create an Amazon Pinpoint client in the location of your push notification code and it is done so by adding the SDK code into our onCreate() function and importing the GoogleCloudMessaging class and the InstanceID class. A new PushListener class is created and making required updates on AndroidManifest file.

Amazon Pinpoint console enables you to target your app users with push messaging. We can send individual messages or configure campaigns that target a group of users that match a profile that you define. For instance, we could email users that have not used the app in 30 days, or send an SMS to those that frequently use a given feature of our app and these updates of features can be managed easily through the Pinpoint Console. Through the console we can add title and body of our notification, add endpoints i.e., the users of our application to whom we send notification to, and the function that it notification will perform when user interacts with it. The 'Direct Message' can be used to send a direct message. A direct message is a one-time message that can be send to a limited audience without creating a campaign.

We configure a campaign based on certain criteria based on which our users will receive notifications and interact with the app. These campaign can be edited any time easily through the console. Some of the basic settings include how frequently user will receive notification, the criteria on which user will receive the notification, quiet time when no notification will be sent, and obviously the title and the body of the notification and what action that takes place when users click on it.

Chapter 6

Conclusion and Future Scope

6.1 Conclusion

In this developing world, where cloud deployment of applications made the scalability, reliability, availability and integrating additional features very easy. Using AWS Mobile Hub and Pinpoint for android application deployment has made adding features and their management very easy to implement. The android application provides information about weather conditions and can send notifications to the user to help in any possible way.

6.2 Future Scope

Cloud Computing is a fast emerging business standard. Enterprises find it beneficial in several ways. Cloud Computing simplifies accessibility, provides virtual storage space, addresses backup issues, it provides security against unauthorized access and loss of data. Key advantage is that users can pay only for the resources they have used on 'the cloud' and do away with the major investments for data storage, software licenses, servers and hardware. Our application implements the cloud computing and android app to give user information about weather and can be used in large scale.

References

- 1) AWS Documentation (<https://aws.amazon.com/documentation/mobile-hub>)
- 2) AWS Developer's Guide (<https://docs.aws.amazon.com/aws-mobile/latest/developerguide/getting-started.html>)
- 3) Wikipedia (www.wikipedia.com)
- 4) Android Developer's guide (<https://developer.android.com>)