MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION, MUMBAI

Project title: "**Subdomain Finder OSINT (Android Java)**"

Subject: MAD

Branch: Computer Engineering

Batch: 3rd Batch [CO3]

Class: CO-6I

Submitted By

Roll No	**Name of student**	**Enrollment no.**
49	Shingare Om Prashant	2101180366

A

Capstone Project On

"**Subdomain Finder OSINT (Android Java)**"

Submitted In Partial Fulfilment of the Requirement

For MAD

In "Computer Science and Engineering" of

Government Polytechnic Dharashiv

Affiliated to

Maharashtra State Board of Technical Education Mumbai

Submitted By

Shingare Om Prashant

Under The Guidance

Mrs. P. Vedpathak Ma'am

Department of Computer Science and Engineering

Government Polytechnic Dharashiv.

MAHARASTRA STATE BOARD OF TECHNICAL EDUCATION, MUMBAI

GOVERNMENT POLYTECHNIC DHARASHIV CERTIFICATE

"Subdomain Finder OSINT (Android Java)"

This to certify that **Mr. Om Prashant Shingare** Roll No. **49** of 6th Semester of diploma in Computer engineering has completed the term work satisfactorily in **MAD (22617)** For academic year 2023-2024 as prescribed in the curriculum.

Place: Dharashiv
Date: / /2024

Enrolment No. : 2101180366
Exam seat no. :

Subject Teacher	Head of Department	Principal
Prof. Gaikwad Sir	Prof.Gaikwad Sir	Prof. Pankhe Sir

ACKNOWLEDGMENT

I take this opportunity to express my profound gratitude and deep regards to my guide **Mrs. P. Vedpathak ma'am** (Computer dept., Government Polytechnic Dharashiv) for his exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessing, help and guidance given by him time to time shall carry me a long way in the journey of life on which I am about to embark.

I also take this opportunity to express a deep sense of gratitude to **Mr. A.B. Gaikwad** (**Head of Dept**.) for their cordial support, valuable information and guidance which helped me in completing this task through various stages.

I am obliged to staff members of Government Polytechnic Dharashiv, for the valuable information provided by them in their respective fields. I am grateful for their cooperation during the period of my assignment.

Lastly, I thank almighty, my parents and my classmates for their constant encouragement without which this assignment would not have been possible.

Your Sincerely
Om Shingare

INDEX

PART B:- MicroProject Report on Subdomain Finder OSINT (Android Java)

Rational

Android application development is one of the rising and growing trend in the industry of mobile. This course examines the principles of mobile application design and covers the necessary concepts which are required to understand mobile based applications and develop Android based Applications in particular. After completing this course students will design and build a variety of real-time Apps using Android.

Aim/Benefits of the Micro Project

Deepens Understanding of Internet Infrastructure: By working on this project, you enhance your comprehension of DNS (Domain Name System), which is fundamental to how the internet operates, including how domain names are resolved and managed.

Advances Java and Android Development Skills: Java is a versatile programming language, and Android is the most widely used mobile operating system globally. Developing an app in this environment improves your proficiency in both Java and Android development, making you more adept in the field of mobile app development.

Fosters API Integration Skills: Integrating third-party services or APIs for DNS lookups or subdomain enumeration exercises your ability to work with external data sources, a critical skill in modern software development.

Course Outcomes

- Interpreter features of Android operating system.
- Configure Android environment and development tools.
- Develop rich user Interfaces by using layouts and controls.
- Use User Interface components for android application development.
- Create Android application using database.
- Publish Android applications.

Literature Reviews

The field of mobile application development has witnessed substantial growth, with music player apps being a popular category. As smartphones become increasingly integral to our daily lives, users demand feature-rich and customizable music player applications.

INTRODUCTION TO MAD

In the ever-evolving landscape of technology, mobile application development stands as a cornerstone, shaping the way we interact with the digital world. As the prevalence of smartphones continues to soar, the creation of innovative and user-friendly mobile applications has become paramount. Mobile application development encompasses the process of designing, creating, and deploying software applications tailored specifically for mobile devices, such as smartphones and tablets.

This dynamic field involves a myriad of technologies, platforms, and frameworks, each contributing to the seamless functioning of applications that cater to a diverse array of user needs. From enhancing productivity to providing entertainment and facilitating communication, mobile apps have become integral to our daily lives.

The process of mobile application development typically involves key stages, including ideation, planning, design, development, testing, and deployment. Developers leverage programming languages such as Java, Kotlin (for Android), Swift (for iOS), and frameworks like React Native or Flutter to create cross-platform applications that can run on multiple devices.

User experience (UX) and user interface (UI) design play a pivotal role, ensuring that applications are not only functional but also visually appealing and intuitive. Security considerations, optimization for various screen sizes, and efficient use of resources are paramount to delivering a successful mobile app.

As the demand for innovative and feature-rich mobile applications continues to surge, the realm of mobile application development remains dynamic, promising endless possibilities for developers to shape the future of digital experiences on the go.

CODE

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools" >

    <uses-permission android:name="android.permission.INTERNET"/>
    <uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Subdiso"
        tools:targetApi="31" >
        <activity
            android:name=".MainActivity"
            android:exported="false" >
            <meta-data
                android:name="android.app.lib_name"
                android:value="" />
        </activity>
        <activity
            android:name=".IntroActivity"
            android:exported="true" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>

            <meta-data
                android:name="android.app.lib_name"
                android:value="" />
        </activity>

        <meta-data
            android:name="preloaded_fonts"
            android:resource="@array/preloaded_fonts" />
    </application>

</manifest>
```

```java
package com.rahadsec.subdisco;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class IntroActivity extends AppCompatActivity {

    Button getStartedButton;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_intro);

        getStartedButton = findViewById(R.id.getStarted);

        SharedPreferences preferences = getSharedPreferences("intro_prefs", MODE_PRIVATE);
        boolean introShown = preferences.getBoolean("intro_shown", false);

        if (introShown) {
            Intent intent = new Intent(IntroActivity.this, MainActivity.class);
            startActivity(intent);
            finish();
        }
        getStartedButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                SharedPreferences preferences = getSharedPreferences("intro_prefs", MODE_PRIVATE);
                SharedPreferences.Editor editor = preferences.edit();
                editor.putBoolean("intro_shown", true);
                editor.apply();

                Intent intent = new Intent(IntroActivity.this, MainActivity.class);
                startActivity(intent);
                finish();
            }
        });

    }
}
```

```
package com.rahadsec.subdisco;

import androidx.appcompat.app.AppCompatActivity;

import android.content.ClipData;
import android.content.ClipboardManager;
import android.os.Bundle;
import android.os.AsyncTask;
import android.view.ContextMenu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.io.IOException;
import java.net.URL;
import java.util.ArrayList;
import java.util.Scanner;


public class MainActivity extends AppCompatActivity {


    TextView subdomainList;
    private ArrayList<String> subdomains;
    private EditText domainName;
    private Button discoverButton;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        domainName = findViewById(R.id.domainName);
        discoverButton = findViewById(R.id.getSubdomains);
        subdomainList = findViewById(R.id.subdomains);

        discoverButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String baseDomain = domainName.getText().toString();
                // Perform subdomain discovery here using a library or API
                // and display the results in the subdomainsTextView
                if (baseDomain.length()>0){
                    new SubdomainDiscoveryTask().execute(baseDomain);
                    Toast.makeText(MainActivity.this, "Discovering Subdomains...",
Toast.LENGTH_SHORT).show();
                }else {
                    Toast.makeText(MainActivity.this, "Please enter a domain",
Toast.LENGTH_SHORT).show();
                }
            }
        });
    }

    private class SubdomainDiscoveryTask extends AsyncTask<String, Void, String> {
        @Override
        protected String doInBackground(String... params) {
            String baseDomain = params[0];
            String urlString = "https://www.virustotal.com/vtapi/v2/domain/report?domain=" +
baseDomain + "&apikey=1d0d5f5070c2a9154bc001264a231c9e12e58f78df26f6f19410d2052d330e85";
            String subdomainsString = "";
            try {
                URL url = new URL(urlString);
                Scanner scanner = new Scanner(url.openStream());
                String response = "";
                while (scanner.hasNext()) {
                    response += scanner.nextLine();
```

```
                }
                scanner.close();
                int counter = 1;
                JSONObject json = new JSONObject(response);
                JSONArray subdomains = json.getJSONArray("subdomains");
                for (int i = 0; i < subdomains.length(); i++) {
                    subdomainsString += counter + ". " + subdomains.getString(i) + "\n";
                    counter++;
                }
            } catch (IOException e) {
                e.printStackTrace();
            } catch (JSONException e) {
                e.printStackTrace();
            }
            return subdomainsString;
        }

        @Override
        protected void onPostExecute(String subdomains) {
            subdomainList.setText(subdomains);
            registerForContextMenu(subdomainList);

        }
    }


    @Override
    public void onCreateContextMenu(ContextMenu menu, View v, ContextMenu.ContextMenuInfo menuInfo) {
        super.onCreateContextMenu(menu, v, menuInfo);
        menu.add(0, v.getId(), 0, "Copy");
    }

    @Override
    public boolean onContextItemSelected(MenuItem item) {
        if (item.getTitle() == "Copy") {
            ClipboardManager clipboard = (ClipboardManager) getSystemService(CLIPBOARD_SERVICE);
            ClipData clip = ClipData.newPlainText("subdomains", subdomainList.getText().toString());
            clipboard.setPrimaryClip(clip);
            Toast.makeText(this, "Copied to clipboard", Toast.LENGTH_SHORT).show();
            return true;
        }
        return super.onContextItemSelected(item);
    }
}
```

ACTUAL METHODOLOGY

1. Searching the topic.
2. Collecting information about the topic.
3. Analysis the information.
4. Developing the code for the given topic.
5. Preparing the project report.
6. Finalizing the report.
7. Submission of report.

ACTUAL RESOURCES USED

Sr No.	Resources	Specs	Qty	Remarks
1.	Computer system	Ram: 8 GB, Rom: 512 SSD, OS: Windows	1	
2.	Software	Visual Studio, Github, Thonny IDE	1	
3.	Any other resources used	Maven Lib, ESP32, WiFi Adapters, Breadboard	1	

REFERENCES

Book: MAD With OM, Official docs of Andriod Studio

OUTPUT


8:22
5G* 12%
Discover Subdomain
Subdomain Finder is a tool designed by Om Shingare in java and Figma uses OSINT in order to enumerate subdomains of websites. It helps pen-testers in collecting and gathering subdomains for a domain which is their target
Get Started


8:22
5G* 12%
Discover Subdomain
example: canva.com
Discover

OUTCOMES

The successful completion of a Subdomain Finder project for Android, developed using Java, is expected to yield several tangible outcomes, both from a technical development perspective and from the standpoint of user benefits. Here's a detailed breakdown:

Technical Outcomes

1. **Functional Android Application**: A fully operational Android app capable of identifying and listing subdomains of a given domain. This involves performing DNS queries or utilizing third-party APIs to discover subdomains.
2. **User Interface (UI) Design**: A user-friendly interface that allows users to easily input domain names, initiate scans, and view results. The design should be intuitive, making efficient use of Android's UI components.
3. **Network Communication**: The app will include robust network communication components for interacting with DNS servers or APIs to fetch subdomain information. This demonstrates an understanding of Android networking capabilities.
4. **Data Handling and Storage**: Implementation of efficient data handling mechanisms for parsing, storing, and displaying the results of the subdomain searches. This might involve using SQLite databases or Android's Room Persistence Library for local data storage.
5. **Security Features**: Since the app deals with network communications, implementing security best practices to protect user data and prevent any malicious use of the app is crucial. This could include securing API keys, using HTTPS for network requests, and ensuring user privacy.

APPLICATIONS

The Subdomain Finder project, developed as an Android application using Java, has a broad spectrum of applications across different sectors, primarily due to the critical role of subdomain discovery in various IT and cybersecurity tasks. Here are some key applications of your project:

1. Cybersecurity and Ethical Hacking

- **Vulnerability Assessment**: Cybersecurity professionals can use the app to uncover subdomains as part of an initial reconnaissance phase. Hidden or forgotten subdomains might host outdated or vulnerable services that could be exploited by attackers.
- **Penetration Testing**: In ethical hacking, understanding the target's domain structure is crucial. The app can help simulate an attacker's footprinting phase to identify points of entry.
- **Security Audits**: Regularly identifying and reviewing all subdomains of an organization can be part of a comprehensive security audit, ensuring that no part of the domain is left unmonitored or unsecured.

2. IT and Network Administration

- **Domain Management**: IT professionals can use the app to keep track of the domain and subdomain structure within their organization, ensuring that DNS configurations are correctly implemented.
- **Troubleshooting and Maintenance**: The app can help in diagnosing network issues related to DNS configurations or in ensuring that redirects and subdomains are functioning as intended after updates or migrations.

3. Web Development and Administration

- **Website Organization**: Web administrators and developers can use the app to map out the structure of their sites, especially useful for large websites or when taking over an existing site without comprehensive documentation.
- **SEO Strategy**: SEO specialists might use the app to understand the scope of a website’s subdomains, optimizing each for search engines to improve visibility and traffic. Managing content and keywords across subdomains can be made more efficient with a clear subdomain structure.

CONCLUSION

The development of a Subdomain Finder application for Android, utilizing Java, marks the culmination of a project that intersects the domains of cybersecurity, network management, web development, and mobile application development. This endeavor highlights the growing necessity for mobile-accessible, security-focused tools in the digital landscape, addressing the needs of a diverse user base ranging from cybersecurity professionals to web administrators and enthusiasts.

Key Takeaways

- **Technological Proficiency**: The project underscores the versatility and power of Java in developing complex, network-based applications for the Android platform. It showcases the developer's ability to handle networking, UI/UX design, data management, and security considerations effectively within a mobile context.
- **Practical Utility**: By providing a tool for discovering subdomains, the application fills a crucial gap in mobile cybersecurity and IT tools, facilitating tasks such as vulnerability assessments, domain management, and SEO optimization. Its development recognizes the importance of on-the-go access to sophisticated network analysis tools in today’s fast-paced digital environment.
- **Educational Value**: For the developer, the project serves as a significant learning experience, enhancing their understanding of both the technical aspects of mobile app development and the practical cybersecurity applications. For the wider community, it offers a resource for learning about network security, domain structures, and app development.
- **Future Directions**: The project opens avenues for further development and innovation, such as integrating more advanced features, improving the app’s efficiency, or expanding its capabilities to include more comprehensive network analysis tools. The potential for open-source collaboration could further enhance its value and utility.

DEDICATION

Hello, I'm Om Shingare. Actually, I'm deeply passionate about creating projects that serve a purpose and have a meaningful impact on my college and the broader community. My dedication lies in the belief that technology should be a force for good, addressing real-world challenges and enhancing educational experiences.

In my journey as a developer, I have consistently focused on projects that are not just about code, but about their practical applications and the value they bring to others. I have a profound aversion to investing my time and skills in projects that do not contribute positively to society. Instead, my commitment is to craft innovative solutions that are helpful, accessible, and transformative for both my peers and educators.

My dedication stems from the conviction that every project I undertake should have a purpose beyond technical proficiency. It is not merely about coding; it is about creating solutions that matter. I am motivated by the prospect of making a difference, whether it is in the classroom, the learning experiences of my fellow students, or the educational landscape of my institution.

I strongly believe in the power of education and technology to drive positive change. By dedicating my efforts to projects that are genuinely helpful for my college and others, I aim to create an impact that resonates far beyond the confines of a computer screen. Each project I undertake is a testament to my commitment to meaningful innovation, and my dedication serves as a guiding principle in my journey as a developer and a lifelong learner.