

Name: Yining Tang
Andrew ID: yiningt

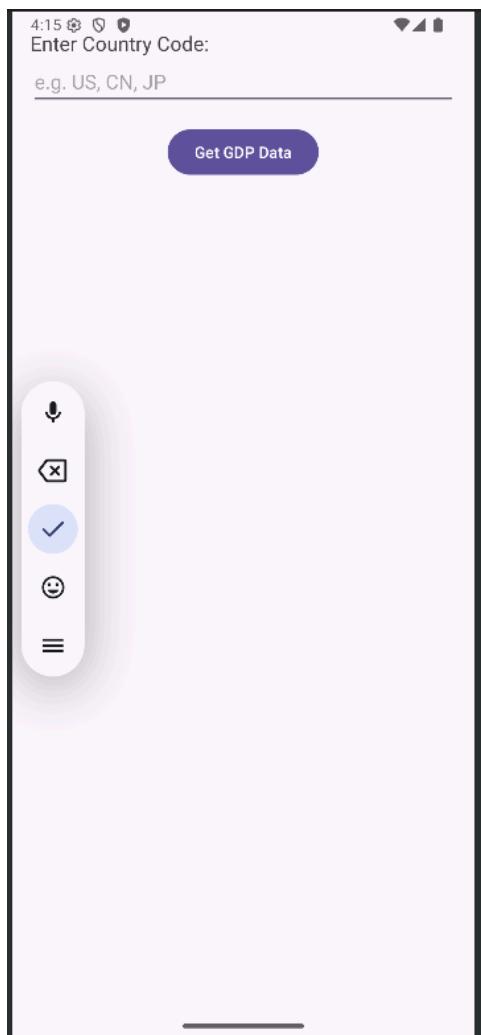
In this project, I want to show the GDP of different countries by year. Users can enter the country code, and GDP will display by year.

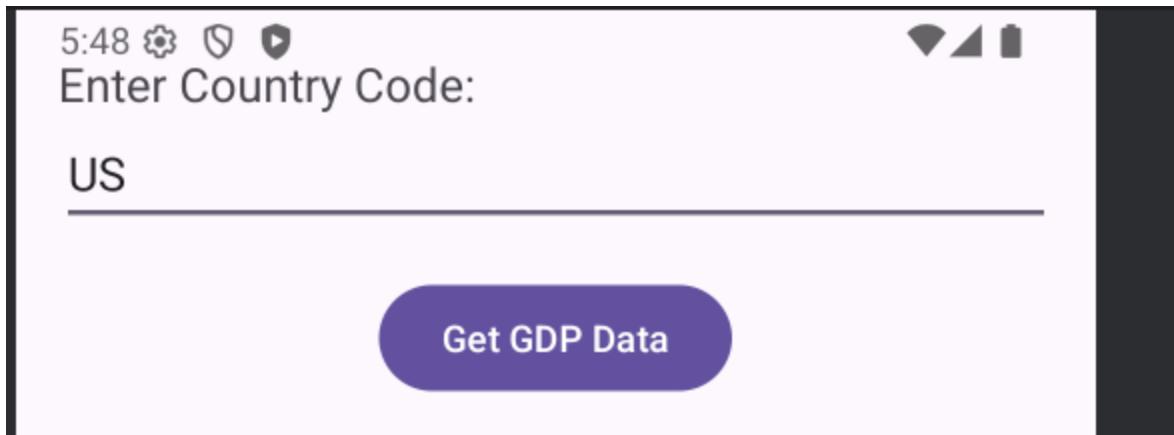
1. Implement a native Android application

a) Multiple View Types

Implemented EditText for country code input; Button to trigger data fetch; TextView to display results; ScrollView with dynamic TextView elements for multi-year GDP display; ProgressBar for loading state

b) Ask for user Input





c) HTTP Requests

[https://\[codespace-url\]/gdp/\[countryCode\]](https://[codespace-url]/gdp/[countryCode])

Codespace url:<https://shiny-space-happiness-4v6x5jxjg94fqpg7.github.dev/>

Users need to enter a country code and can get each year's GDP of that country.

The request validates 2-letter country codes and makes GET requests to web service via AsyncTask in MainActivity.java:

```
new FetchGDPTask().execute(countryCode);
```

And it runs in the background as required in the guideline.

d) JSON Parsing & Display

World Bank API Parsing:

```
JSONArray jsonResponse = new JSONArray(result);
JSONArray dataArray = jsonResponse.getJSONArray(1);
JSONObject yearData = dataArray.getJSONObject(i);
```

Parses World Bank API response format::

```
[{"page":1,...}, [{"indicator":..., "value":X, "date":"2023"}, ...]]
```

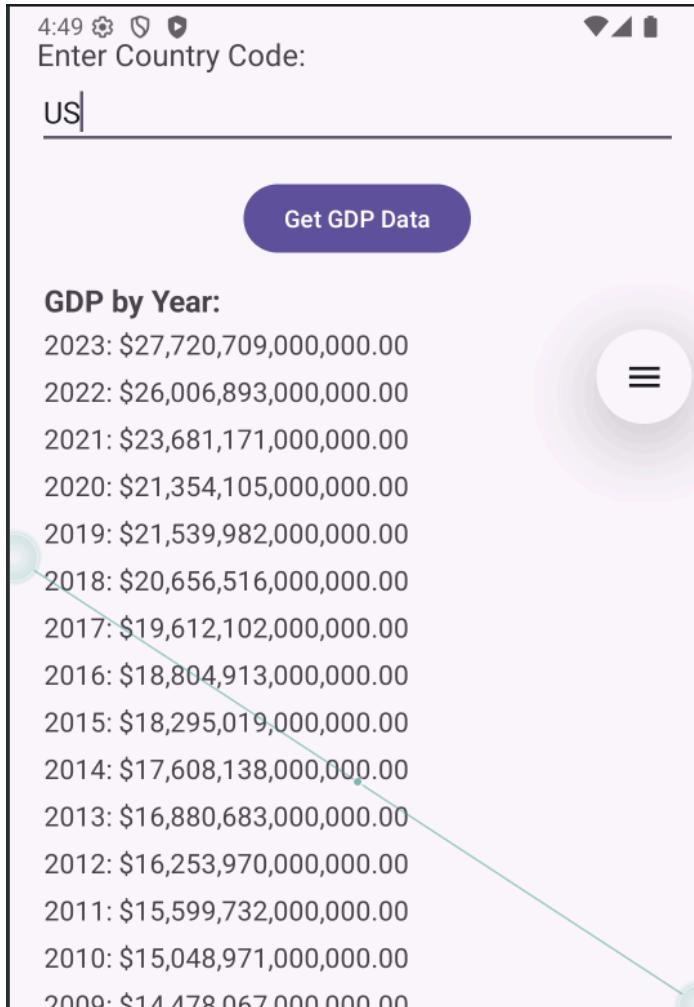
Examples:

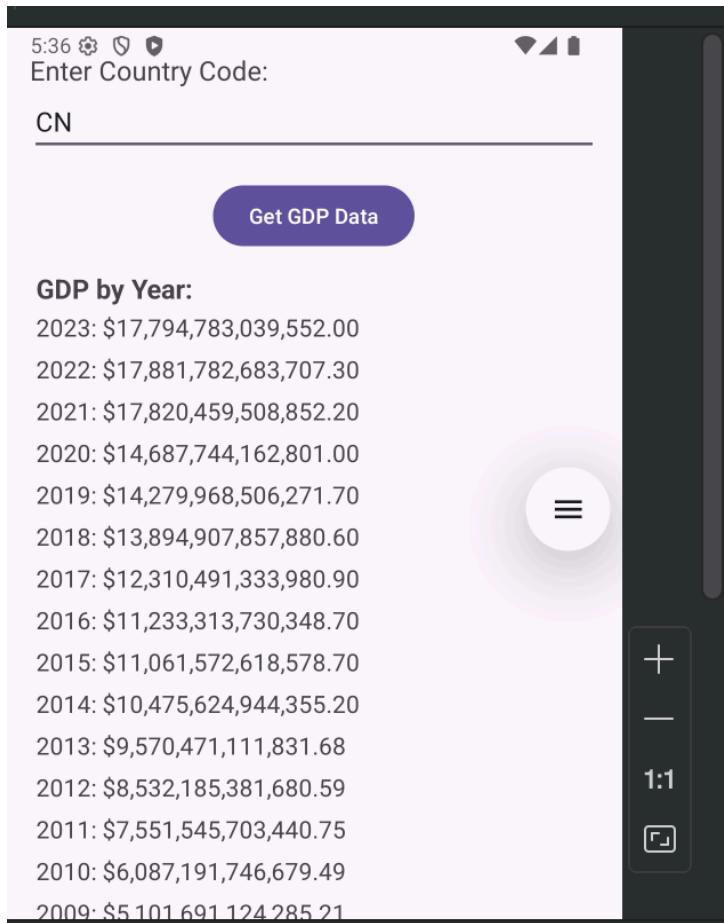
```
[{"page":1,"pages":2,"per_page":50,"total":64,"sourceid":"2","lastupdated":"2025-03-24"},[{"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2023", "value":27720709000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2022", "value":26006893000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2021", "value":23681171000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2020", "value":21354105000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2019", "value":19705000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2018", "value":18455000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2017", "value":17355000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2016", "value":16355000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2015", "value":15455000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2014", "value":14655000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2013", "value":13905000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2012", "value":13205000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2011", "value":12555000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2010", "value":11955000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2009", "value":11355000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2008", "value":10755000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2007", "value":10155000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2006", "value":9555000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2005", "value":8955000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2004", "value":8355000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2003", "value":7755000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2002", "value":7155000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2001", "value":6555000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"2000", "value":5955000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"1999", "value":5355000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"1998", "value":4755000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"1997", "value":4155000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"1996", "value":3555000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"1995", "value":2955000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"1994", "value":2355000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"1993", "value":1755000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"1992", "value":1155000000000, "unit":"",
"obs_status": "", "decimal":0}, {"indicator":{"id":"NY.GDP.MKTP.CD","value":"GDP (current US$)"}, "country":{"id":"US","value":"United States"}, "countryiso3code":"USA", "date":"1991", "value":555000000000, "unit":"",
"obs_status": "", "decimal":0}], [{"text": "Page 1 of 2"}]
```

```
US$)"}, "country": {"id": "US", "value": "United States"}, "countryiso3code": "USA", "date": "2019", "value": 21539982000000, "unit": "", "obs_status": "", "decimal": 0}, {"indicator": {"id": "NY.GDP.MKTP.CD", "value": "GDP (current US$)"}, "country": {"id": "US", "value": "United...}}
```

The app processes World Bank API response format and displays GDP history in scrollable format as shown below. It also formats currency values properly.

e) display new information





f) repeatable

Users can query multiple countries without app restart. E.g. I entered CN after entering the US as shown above and it gave me answers accordingly.

2. Implement a web service

a.

Use World Bank API:

```
// Construct the World Bank API URL
String apiUrl = "https://api.worldbank.org/v2/country/" + countryCode +
"/indicator/NY.GDP.MKTP.CD?format=json";
```

My servlet endpoint:

```
@WebServlet("/gdp/*") // Handles /gdp/US, /gdp/CN, etc.
protected void doGet(HttpServletRequest req, HttpServletResponse resp) {
    // Fetches data from World Bank API
    // Logs to MongoDB
    // Returns JSON array
}
```

B.

Android → Web Service:

```
// Android code (MainActivity.java)
String apiUrl = https://[codespace-url]/gdp/[countryCode]
```

Web Service Handling:

```
// GDPWebService.java
String countryCode = req.getPathInfo().substring(1); // Country Code Extraction
```

c.

My API design:

It accepts country code via URL path, and returns HTTP 200 with JSON array on success; HTTP 404/500 with error message on failure

Business Logic:

- 1) Fetches GDP data from World Bank API
- 2) Processes response to extract relevant fields
- 3) Implements server-side input validation

Third-Party API:

Uses World Bank API (approved, not banned)

Fetches JSON directly:

```
String apiUrl = "https://api.worldbank.org/v2/country/" + countryCode +
"/indicator/NY.GDP.MKTP.CD?format=json";
```

No screen scraping (pure JSON API).

Processing:

I forward raw API response without unnecessary computation:

```
resp.getWriter().write(apiResponse);
```

No over-fetching: Returns only what Android needs (GDP values by year).

d.

Response Format:

```
resp.setContentType("application/json"); // Explicit JSON
resp.getWriter().write(apiResponse); // Original World Bank JSON
```

Android parses only what it needs:

```
// parses only "date" and "value" fields
JSONObject yearData = dataArray.getJSONObject(i);
String year = yearData.optString("date");
double gdpValue = yearData.optDouble("value");
```

4.

```
Document logEntry = new Document()
.append("country", countryCode)
.append("api_url", apiUrl)
.append("api_latency_ms", apiLatency)
.append("response", apiResponse)
.append("client_ip", clientIP)
.append("user_agent", userAgent)
.append("timestamp", requestTime);
```

All the needed information are extracted and store in log:

- Information about the request from the mobile phone:

Client IP Address: The IP address of the client (mobile phone) making the request is logged.

User Agent: The user agent is captured, giving insight into what type of phone or device made the request.

Country: This information is extracted from the request, indicating which country is entered.

Timestamp of Request: The exact time the request was received by the server is logged, helping track when the request was made.

- Information about the request and reply to the 3rd-party API:

API Request URL: The URL that was used to send the request to the 3rd-party API is logged, providing insight into the actual service being called.

API Latency: The time it took for the 3rd-party API to respond is captured, indicating the latency involved in the interaction.

API Response: The full response received from the 3rd-party API is stored, allowing you to analyze the data received and use it in the response to the mobile phone.

- Information about the reply to the mobile phone:

Data Sent to the Mobile Phone: The same response that was sent to the mobile phone, which includes the information retrieved from the 3rd-party API, is logged. This provides insight into the data the user actually receives.

Picture of the log dashboard:

Recent Requests

5.

Cloud Atlas Connection:

```
public void init() {
    String uri = "mongodb+srv://yiningt:(passwd
here)@cluster0.ij87b.mongodb.net/?retryWrites=true&w=majority&appName=Cluster0";
    this.mongoClient = MongoClients.create(uri);
}
```

CRUD Operations:

Insert: collection.insertOne(logEntry)

Query: Used in dashboard analytics.

The screenshot shows the Atlas interface with the following details:

- Header:** Atlas, yining's Org ..., Access Manager, Billing, All Clusters, Get Help, yining
- Left Sidebar:** Project 0, Overview, DATABASE (selected), Clusters, SERVICES, Atlas Search, Stream Processing, Triggers, Migration, Data Federation, SECURITY (selected), Backup, Database Access, Network Access, Advanced, Goto.
- Central Area:** Data Services tab selected. A search bar for "Search Namespaces" is present. The collection "gap_data.logs" is selected under "DATABASE".
 - Logs Sub-Collection:** "country_gdp" and "logs" are listed.
- Logs Collection View:** Title "gap_data.logs", Storage Size: 88KB, Logical Data Size: 427.21KB, Total Documents: 41, Indexes Total Size: 36KB.
 - Actions: Find, Indexes, Schema Anti-Patterns, Aggregation, Search Indexes.
 - Buttons: INSERT DOCUMENT, Filter, Reset, Apply, Options.
 - Text: Generate queries from natural language in Compass.
 - Results:
 - Document 1: _id: ObjectId('67ff544a915e4a22339a4c423'), country: "us", api_url: "https://api.worldbank.org/v2/country/us/indicator/NY.GDP.MKTP.CD?forma...", api_latency_ms: 781, response: "[{"page":1,"pages":2,"per_page":50,"total":64,"sourceid":"2","lastupda..." client_ip: "172.17.0.1", user_agent: "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, timestamp: 2025-04-08T15:45:44.910+00:00
 - Document 2: _id: ObjectId('67ff544ea15e4a22339a4c424'), country: "us", api_url: "https://api.worldbank.org/v2/country/us/indicator/NY.GDP.MKTP.CD?forma...", api_latency_ms: 167
 - Navigation: PREVIOUS, 21-40 of many results, NEXT.

6.

<http://localhost:8080/Project4Task2-1.0-SNAPSHOT/gdp/US>

<http://localhost:8080/Project4Task2-1.0-SNAPSHOT/dashboard.jsp>

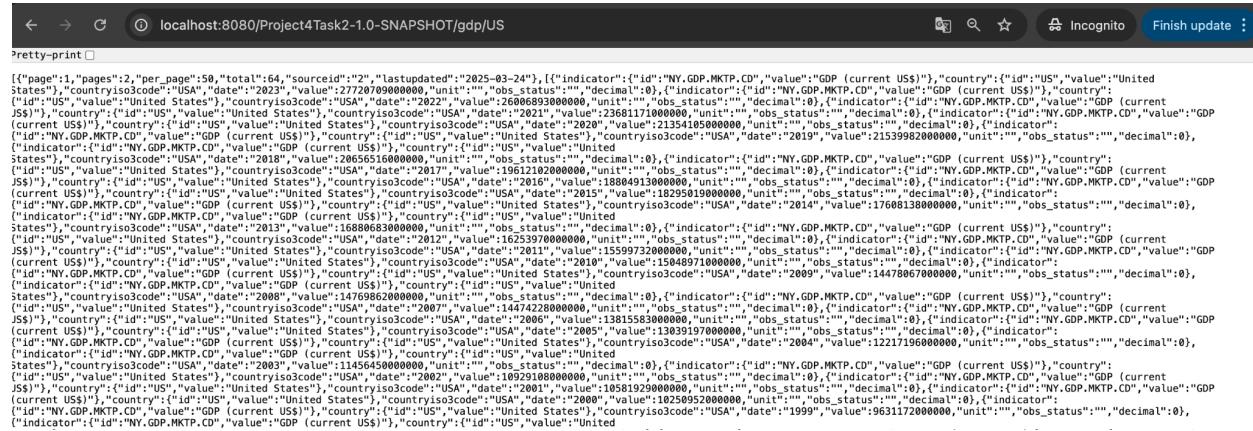
[{"page":1,"pages":2,"per_page":50,"total":64,"sourceid":2,"lastupdated":"2025-03-24"}, {"["indicator":{"id":"NY_GDP_MKTP_CD","value":"GDP (current US\$)"}, "country": "CHN","value":1,"countryiso3code": "CHN","date": "2023","value":1779478303952,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","value": 2,"countryiso3code": "CHN","date": "2022","value":17881726836707.3,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","value": 3,"countryiso3code": "CHN","date": "2021","value":17820459508852.2,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","value": 4,"countryiso3code": "CHN","date": "2020","value":14687744162801,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","value": 5,"countryiso3code": "CHN","date": "2019","value":14279968506271.7,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","value": 6,"countryiso3code": "CHN","date": "2018","value":13894907857880.6,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","value": 7,"countryiso3code": "CHN","date": "2017","value":12310491333980.9,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","value": 8,"countryiso3code": "CHN","date": "2016","value":11233313730346.7,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","value": 9,"countryiso3code": "CHN","date": "2015","value":11061572618578.7,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","value": 10,"countryiso3code": "CHN","date": "2014","value":10475624944355.2,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","value": 11,"countryiso3code": "CHN","date": "2013","value":9507471111831.68,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","value": 12,"countryiso3code": "CHN","date": "2012","value":8532185381680.59,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","value": 13,"countryiso3code": "CHN","date": "2011","value":7551545703440.75,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","value": 14,"countryiso3code": "CHN","date": "2010","value":6087191746679.49,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","value": 15,"countryiso3code": "CHN","date": "2009","value":5101691124285.21,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","value": 16,"countryiso3code": "CHN","date": "2008","value":4594336785737.67,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","value": 17,"countryiso3code": "CHN","date": "2007","value":3550327802998.4,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","value": 18,"countryiso3code": "CHN","date": "2006","value":2752118657184.02,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","value": 19,"countryiso3code": "CHN","date": "2005","value":2285961149879.85,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","date": "2004","value":1955346768721.39,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","date": "2003","value":1660280543846.79,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","date": "2002","value":1470557654799.95,"unit": "", "obs_status": "", "decimal": 0}, {"["indicator":{"id": "NY_GDP_MKTP_CD","value": "GDP (current US\$)"}, "country": "CHN","date": "2001","value":1339400887105.11,"unit": "", "obs_status": "", "decimal": 0}]}]

GDP Analytics Dashboard

Total Requests 38	Top Countries <ol style="list-style-type: none">1. US (17 requests)2. CN (13 requests)3. us (8 requests)	Last Updated Tue Apr 08 11:23:22 EDT 2025
-----------------------------	--	---

Recent Requests

Incognito Window to confirm it can be reached without authentication:



7.

E.g.

<https://shiny-space-happiness-4v6x5xjg94fqpg7-8080.app.github.dev/gdp/us>

<https://shiny-space-happiness-4v6x5xjg94fqpg7-8080.app.github.dev/dashboard.jsp>

shiny-space-happiness-4v6x5jxg94fqpg7-8080.app.github.dev/gdp/us

Incognito window to confirm it can be reached without authentication:

Incognito window to confirm it can be reached without authentication:

Github:

The screenshot shows a GitHub repository page for 'Project4Task2'. At the top, there's a navigation bar with links for Code, Issues, Pull requests, Actions, Projects, Security, Insights, and Settings. The repository name 'Project4Task2' is displayed, along with a 'Private' badge. Below the repository name, there's a message encouraging users to help improve GitHub Codespaces. The main content area shows a list of commits from 'YiningTang0'. The commits are as follows:

Commit	Message	Time
AndroidApp.zip	Add files via upload	yesterday
Dockerfile	ROOT.war and Dockerfile added	last week
Project4Task2WriteUp.pdf	writeup uploaded	yesterday
README.md	Initial commit	last week
ROOT.war	Add files via upload	1 hour ago
WebService.zip	Add files via upload	1 hour ago
devcontainer.json	Add files via upload	5 days ago

At the bottom, there's a preview of the 'README' file.

Codespace picture:

The screenshot shows a VS Code interface within a Codespace. The left sidebar has an 'EXPLORER' tab open, displaying the project structure: 'PROJECT4TASK2 [CODESPACE]' containing 'AndroidApp.zip', 'devcontainer.json', 'Dockerfile', 'Project4Task2WriteUp.pdf', 'README.md', 'ROOT.war', and 'WebService.zip'. The 'README.md' file is currently selected. The main editor area shows the content of 'README.md': '# Project4Task2'. At the bottom, the 'PORTS' tab is active, showing a table with one port entry:

Port	Forwarded Address	Running Process	Visibility	Origin
8080	https://shiny-space...	/usr/bin/docker-proxy -proto tcp -host-ip ...	Public	Auto Forwarded

Codespace terminal running:

```
@YiningTang0 ➔ /workspaces/Project4Task2 (main) $ docker build -t project4_task2 .
[+] Building 28.6s (8/8) FINISHED
docker:default
=> [internal] load build definition from Dockerfile
0.1s
=> => transferring dockerfile: 462B
0.0s
=> [internal] load metadata for docker.io/library/tomcat:11.0.0-M24-jdk21-temurin-noble
0.8s
=> [auth] library/tomcat:pull token for registry-1.docker.io
0.0s
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load build context
=> => transferring context: 1.99MB
0.0s
=> [1/2] FROM
docker.io/library/tomcat:11.0.0-M24-jdk21-temurin-noble@sha256:df2de3c15547dd359672c6f35
78c16a8809a2ca9547ccf 25.9s
=> => resolve
docker.io/library/tomcat:11.0.0-M24-jdk21-temurin-noble@sha256:df2de3c15547dd359672c6f35
78c16a8809a2ca9547ccf0 0.0s
=> => sha256:c61d10e3f45ca5ceba713ad4ff57202ff5536e200d3b5ed9fcdf10b36b8d9c6
12.04kB / 12.04kB 0.0s
=> => sha256:eb993dc6942ffcb7633f2cb271bd4b0a275fc9bcd8f5100c5b4d24694cacf50
30.57MB / 30.57MB 1.8s
=> => sha256:facf7d693a40c7b09d711beecddef9142e6074a9d517b5b2f44a16299654b031
158.59MB / 158.59MB 5.9s
=> => sha256:df2de3c15547dd359672c6f3578c16a8809a2ca9547ccf024ee145294a5c488c
5.34kB / 5.34kB 0.0s
=> => sha256:a282954f64fb7a659ba705dfc994784a1d247a3c17b1577e86d2be676a13c2ae
2.72kB / 2.72kB 0.0s
=> => sha256:ef6d67df44ebaf68af30549648235de86e15355c7fc9afc496d8d73df71a7a3d
19.77MB / 19.77MB 1.8s
=> => sha256:bb3fe075c198ddd7a31193d47754cdb43d384b3bcd7585284f478d8bd5fa5107
175B / 175B 1.9s
=> => sha256:ad28f0ce2ee5faf000467e3072dbaa09ec07d716c591a6871791efa036d05e9d
2.11kB / 2.11kB 1.9s
=> => sha256:4f4fb700ef54461cfa02571ae0db9a0dc1e0cdb5577484a6d75e68dc38e8acc1
32B / 32B 2.1s
=> => sha256:d4f114cd17ef6b8759027a2e1cec31a4f6e7116e77f55128c844a5935fe21bde
139B / 139B 2.0s
```

```
=> => extracting
sha256:eb993dc6942ffcb7633f2cb271bd4b0a275fc9bdc8f5100c5b4d24694cacf50
2.9s
=> => sha256:7fabf86c44bea842ac9aadff69f67c5a23d664717a1382281ad21573575ec6ae
29.87MB / 29.87MB          6.9s
=> => extracting
sha256:ef6d67df44ebaf68af30549648235de86e15355c7fc9afc496d8d73df71a7a3d
1.8s
=> => extracting
sha256:facf7d693a40c7b09d711beecddef9142e6074a9d517b5b2f44a16299654b031
4.8s
=> => extracting
sha256:bb3fe075c198ddd7a31193d47754cdb43d384b3bcd7585284f478d8bd5fa5107
0.0s
=> => extracting
sha256:ad28f0ce2ee5faf000467e3072dbaa09ec07d716c591a6871791efa036d05e9d
0.0s
=> => extracting
sha256:d4f114cd17ef6b8759027a2e1cec31a4f6e7116e77f55128c844a5935fe21bde
0.0s
=> => extracting
sha256:4f4fb700ef54461cfa02571ae0db9a0dc1e0cdb5577484a6d75e68dc38e8acc1
0.0s
=> => extracting
sha256:7fabf86c44bea842ac9aadff69f67c5a23d664717a1382281ad21573575ec6ae
1.3s
=> [2/2] COPY ROOT.war /usr/local/tomcat/webapps/
0.6s
=> exporting to image          1.1s
=> => exporting layers          0.9s
=> => writing image
sha256:fecd2583a1aa1e2b57b9c0ab02ba52345f6a912ca9894fc99e2406fe12c48169
0.1s
=> => naming to docker.io/library/project4_task2
0.0s
@YiningTang0 ➔ /workspaces/Project4Task2 (main) $ docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
project4_task2  latest   fecd2583a1aa  6 minutes ago  510MB
@YiningTang0 ➔ /workspaces/Project4Task2 (main) $ docker run --rm -it -p 8080:8080
project4_task2
Using CATALINA_BASE: /usr/local/tomcat
Using CATALINA_HOME: /usr/local/tomcat
Using CATALINA_TMPDIR: /usr/local/tomcat/temp
Using JRE_HOME:      /opt/java/openjdk
```

Using CLASSPATH: /usr/local/tomcat/bin/bootstrap.jar:/usr/local/tomcat/bin/tomcat-juli.jar
Using CATALINA_OPTS:
08-Apr-2025 15:45:29.158 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
Server version name: Apache Tomcat/11.0.0-M24
08-Apr-2025 15:45:29.175 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
Server built: Aug 2 2024 13:16:30 UTC
08-Apr-2025 15:45:29.175 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
Server version number: 11.0.0.0
08-Apr-2025 15:45:29.175 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
OS Name: Linux
08-Apr-2025 15:45:29.175 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
OS Version: 6.8.0-1021-azure
08-Apr-2025 15:45:29.175 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
Architecture: amd64
08-Apr-2025 15:45:29.175 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
Java Home: /opt/java/openjdk
08-Apr-2025 15:45:29.176 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
JVM Version: 21.0.4+7-LTS
08-Apr-2025 15:45:29.176 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
JVM Vendor: Eclipse Adoptium
08-Apr-2025 15:45:29.176 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
CATALINA_BASE: /usr/local/tomcat
08-Apr-2025 15:45:29.176 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
CATALINA_HOME: /usr/local/tomcat
08-Apr-2025 15:45:29.199 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
Command line argument: -Djava.util.logging.config.file=/usr/local/tomcat/conf/logging.properties
08-Apr-2025 15:45:29.200 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
Command line argument: -Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager
08-Apr-2025 15:45:29.201 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
Command line argument: -Xmx300m
08-Apr-2025 15:45:29.201 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
Command line argument: -Djdk.tls.ephemeralDHKeySize=2048
08-Apr-2025 15:45:29.202 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
Command line argument: -Dorg.apache.catalina.security.SecurityListener.UMASK=0027
08-Apr-2025 15:45:29.202 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
Command line argument: --add-opens=java.base/java.lang=ALL-UNNAMED
08-Apr-2025 15:45:29.203 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
Command line argument: --add-opens=java.base/java.io=ALL-UNNAMED
08-Apr-2025 15:45:29.203 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
Command line argument: --add-opens=java.base/java.util=ALL-UNNAMED
08-Apr-2025 15:45:29.203 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
Command line argument: --add-opens=java.base/java.util.concurrent=ALL-UNNAMED
08-Apr-2025 15:45:29.204 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
Command line argument: --add-opens=java.rmi/sun.rmi.transport=ALL-UNNAMED

08-Apr-2025 15:45:29.204 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
Command line argument: --enable-native-access=ALL-UNNAMED
08-Apr-2025 15:45:29.205 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
Command line argument: -Dcatalina.base=/usr/local/tomcat
08-Apr-2025 15:45:29.205 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
Command line argument: -Dcatalina.home=/usr/local/tomcat
08-Apr-2025 15:45:29.205 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
Command line argument: -Djava.io.tmpdir=/usr/local/tomcat/temp
08-Apr-2025 15:45:29.222 INFO [main]
org.apache.catalina.core.AprLifecycleListener.lifecycleEvent Loaded Apache Tomcat Native
library [2.0.8] using APR version [1.7.2].
08-Apr-2025 15:45:29.237 INFO [main]
org.apache.catalina.core.AprLifecycleListener.initializeSSL OpenSSL successfully initialized
[OpenSSL 3.0.13 30 Jan 2024]
08-Apr-2025 15:45:29.727 INFO [main] org.apache.coyote.AbstractProtocol.init Initializing
ProtocolHandler ["http-nio-8080"]
08-Apr-2025 15:45:29.770 INFO [main] org.apache.catalina.startup.Catalina.load Server
initialization in [876] milliseconds
08-Apr-2025 15:45:29.840 INFO [main] org.apache.catalina.core.StandardService.startInternal
Starting service [Catalina]
08-Apr-2025 15:45:29.841 INFO [main] org.apache.catalina.core.StandardEngine.startInternal
Starting Servlet engine: [Apache Tomcat/11.0.0-M24]
08-Apr-2025 15:45:29.863 INFO [main] org.apache.catalina.startup.HostConfig.deployWAR
Deploying web application archive [/usr/local/tomcat/webapps/ROOT.war]
08-Apr-2025 15:45:30.606 INFO [main] org.apache.jasper.servlet.TldScanner.scanJars At least
one JAR was scanned for TLDs yet contained no TLDs. Enable debug logging for this logger for
a complete list of JARs that were scanned but no TLDs were found in them. Skipping unneeded
JARs during scanning can improve startup time and JSP compilation time.
08-Apr-2025 15:45:30.667 INFO [main] org.apache.catalina.startup.HostConfig.deployWAR
Deployment of web application archive [/usr/local/tomcat/webapps/ROOT.war] has finished in
[802] ms
08-Apr-2025 15:45:30.671 INFO [main] org.apache.coyote.AbstractProtocol.start Starting
ProtocolHandler ["http-nio-8080"]
08-Apr-2025 15:45:30.683 INFO [main] org.apache.catalina.startup.Catalina.start Server startup
in [910] milliseconds
08-Apr-2025 15:45:44.775 WARNING [http-nio-8080-exec-3]
com.mongodb.diagnostics.logging.Loggers.shouldUseSLF4J SLF4J not found on the classpath.
Logging is disabled for the 'org.mongodb.driver' component

//ai appendix: chagpt and deepseek are used for brainstorming, syntax,
grammar, sentence flow, debugging