

Zadanie

Stworzenie dokumentacji architektonicznej do systemu
Maze Solver, udostępniającego możliwość
rozwiązywania *labirytnu*.

Pytania do zaadresowania

- Ile użytkowników będzie korzystać z systemu?
- Jak użytkownik wprowadzi labirynt? Jaki jest format danych wejściowych (obraz, pliki z danymi)?
- Czy system ma działać w czasie rzeczywistym? Czy odpowiedź będzie natychmiastowa, czy też po pewnym czasie?
- Jaki format ma mieć odpowiedź? Czy użytkownik ma otrzymać obraz, czy też informacje tekstową dotyczącą kroków do rozwiązania labiryntu?

Następne kroki

Po otrzymaniu odpowiedzi przygotowuje odpowiednie diagramy, rejestr decyzji architektonicznych oraz finalną prezentację

`architecture.pdf`

Projekt Rozwiązania

<https://github.com/lukevito/maze>

Model opisu architektury

Prezentacja architektury systemu Maze Solver w
modelu **C4**

<https://c4model.com/>

Dokumentacja Maze Solver

<https://github.com/lukevito/maze/architecture.pdf>

Repozytorium dokumentacji architektonicznych systemu **Maze Solver**, dostępna na różnych poziomach C4.

Funkcjonalność dokumentacji systemu

(Dokumentacja Dokumentacji Maze Solver)

Generowanie diagramów

Możliwość generowania diagramów modelu architektury (zobacz katalog **doc\diagrams**)

<https://diagrams.mingrammer.com/>

Formaty: png, img

Polecenie generowania diagramów

```
python.exe .\architecture.py
```

Przykład:

```

from diagrams import Cluster, Diagram
from diagrams.gcp.analytics import BigQuery, Dataflow, PubSub
from diagrams.gcp.compute import AppEngine, Functions
from diagrams.gcp.database import BigTable
from diagrams.gcp.iot import IotCore
from diagrams.gcp.storage import GCS

with Diagram("Message Collecting", show=False):
    pubsub = PubSub("pubsub")

    with Cluster("Source of Data"):
        [IotCore("core1"),
         IotCore("core2"),
         IotCore("core3")] >> pubsub

    with Cluster("Targets"):
        with Cluster("Data Flow"):
            flow = Dataflow("data flow")

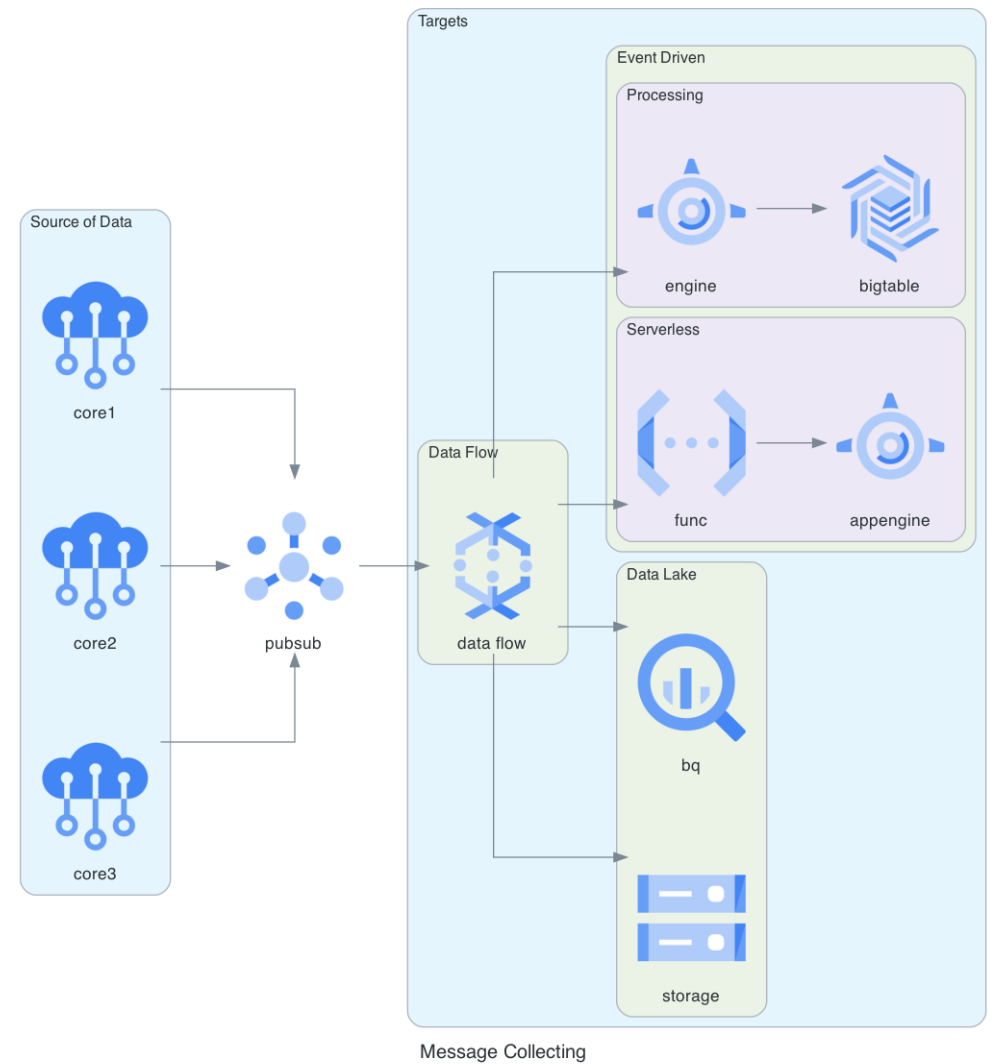
        with Cluster("Data Lake"):
            flow >> [BigQuery("bq"),
                    GCS("storage")]

        with Cluster("Event Driven"):
            with Cluster("Processing"):
                flow >> AppEngine("engine") >> BigTable("bigtable")

            with Cluster("Serverless"):
                flow >> Functions("func") >> AppEngine("appengine")

    pubsub >> flow

```



Architecture Decision Record

Każda zmiana architektoniczna, powinna mieć powiązanie z **Architecture Decision Record** (zobacz katalog **doc\adr**)

Polecenie generowania wpisu w ADR

```
adr-new utworzenie dokumentacji pierwsze kroki
```

Utworzy plik zawierający decyzje architektoniczną

2. Utworzenie dokumentacji pierwsze kroki

Date: 2024-02-12

Status

Do Akceptacji

Context

Potrzeba utworzenia projektu do dokumentacji systemu Maze Solver

Założenia:

- dokumentacja będzie posiadać rejestr ADR (DONE)
- dokumentacja będzie dostępna w różnych formatach
- dokumentacja będzie zwinna i lekka w modyfikacji, z możliwością *wersjonowania* i *code review*.

The issue motivating this decision, and any context that influences or constrains the decision.

Decision

Wybrane narzędzia:

Python

Diagrams (python lib)

Marp

Docelowy model:

C4

Consequences

- Wymagana znajomość podstaw języka python
- Wymagana integracja z Enterprise Architect

0002-utworzenie-dokumentacji-pierwsze-kroki.md

Dodatki

Prezentacja architektury

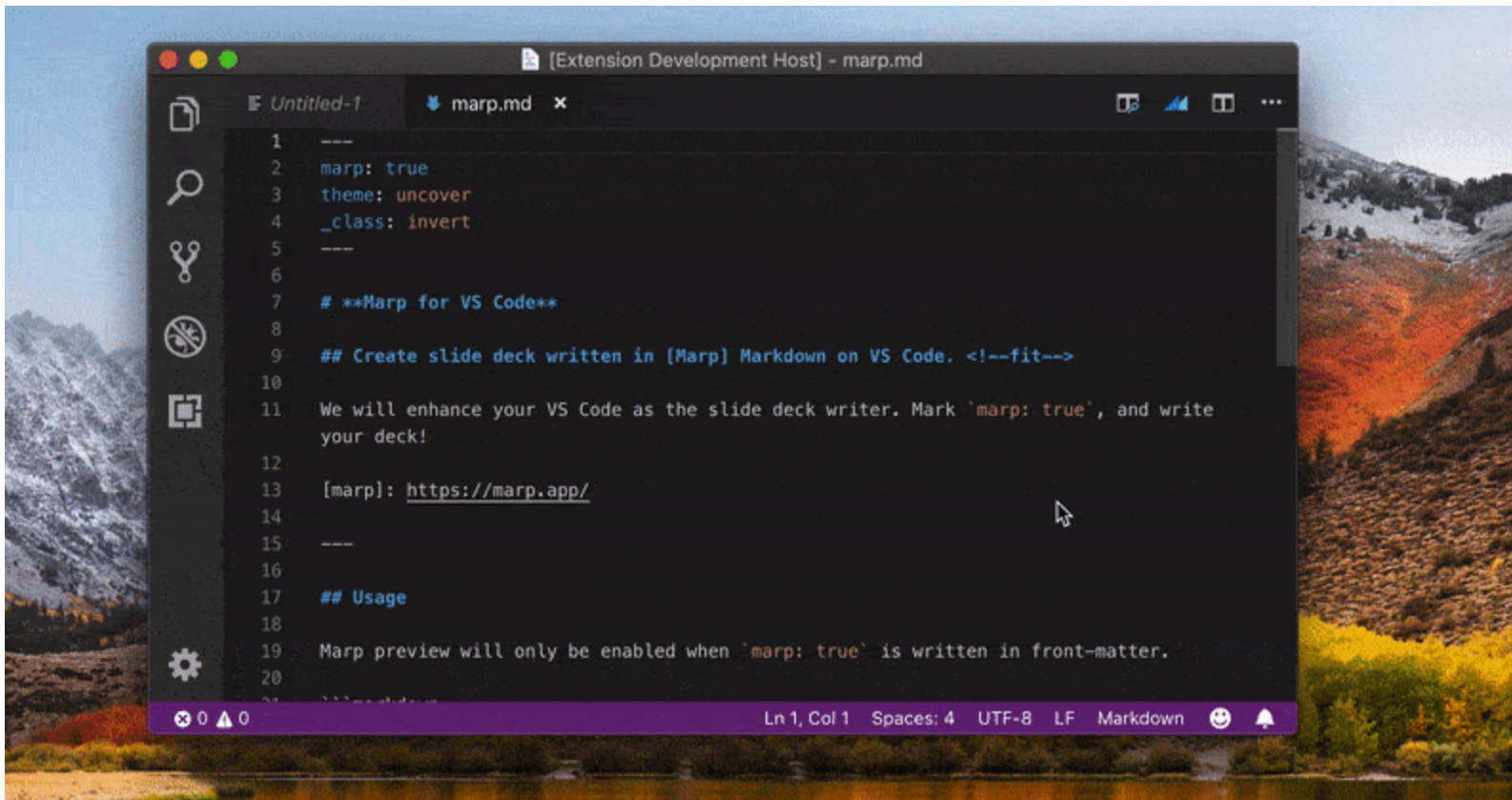
Możliwość generowania prezentacji z modelu architektury

<https://marp.app/>

Formaty: PowerPoint, HTML, PDF, png

(zobacz katalog **doc/presentations/export**)

Generowanie prezentacji (doc/presentations/zadanie.md)



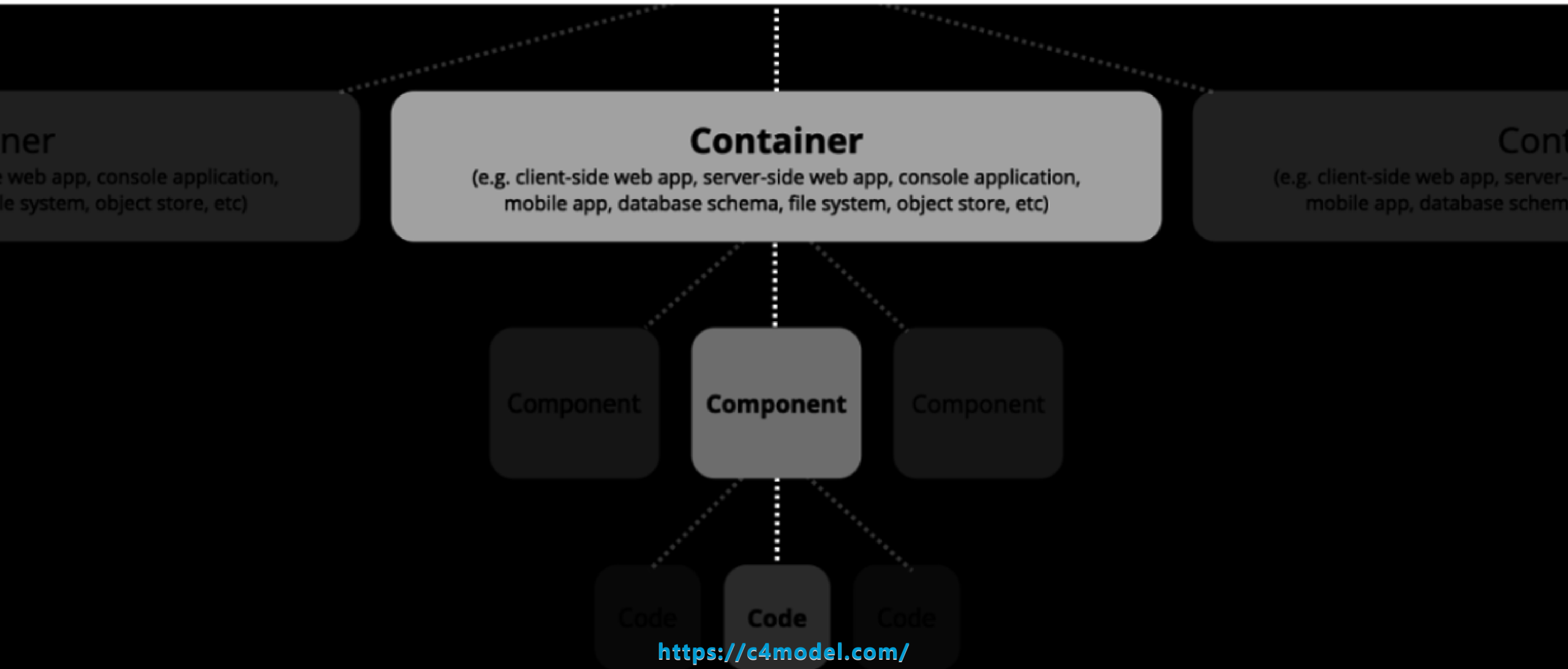
The screenshot shows a VS Code editor window titled "[Extension Development Host] - marp.md". The editor displays the content of a file named "marp.md". The file contains front-matter for Marp and instructions on how to use it.

```
1 ---
2 marp: true
3 theme: uncover
4 _class: invert
5 ---
6
7 # **Marp for VS Code**
8
9 ## Create slide deck written in [Marp] Markdown on VS Code. <!--fit-->
10
11 We will enhance your VS Code as the slide deck writer. Mark 'marp: true', and write
12 your deck!
13
14 [marp]: https://marp.app/
15
16 ---
17
18 ## Usage
19
20 Marp preview will only be enabled when 'marp: true' is written in front-matter.
```

The status bar at the bottom indicates "Ln 1, Col 1", "Spaces: 4", "UTF-8", "LF", and "Markdown".

Prezentacja na którą patrzysz powstała przy użyciu
tego narzędzia

Software System



FileEditSelectionViewGoRunTerminalHelp

EXPLORER

OPEN EDITORS

GROUP 1

README.md

GROUP 2

Preview README.md

CD_RED

.vscode

images

.gitignore

README.md

README.pptx

temp.md

856
857 ---
858 Journey Mapping
859 ![bg 80% right:60% invert:100%](images/map.png)
860
861 ---
862 ## **\#Product** and more, more
863
864
865 ## **Dokumentacja**
866 - Repozytorium dokumentacji architektonicznych w jednym miejscu, dostępna na różnych poziomach, czytelna dla osób nie związanych z IT.
867 - Prezentacja architektury w modelu *C4*
868 Tam gdzie to wymagane odpowiednio szczegółowa.
869
870 - Zwinna i lekka w modyfikacji, z możliwością wersjonowania i code review.
871
872 <!-- _footer: '**\#AGILE**' -->
873
874 ---
875 ![bg % invert:100%](images/c4.png)
876
877 ---
878 ## **\#Product** and more, more, more
879
880 - Każda zmiana architektoniczna, powinna mieć powiązanie z inicjatywą/projektem/epikiem biznesowym
881 - Każdy powinien mieć możliwość zaproponowania nowego podejścia, bez angażowania się w skomplikowane narzędzia.
882 - Wiele formatów prezentacji. Do pobrania w formacie w którym pracuje się najprzyjemniej.
883 - Dokumentacja specjalistyczna jak najbliżzej systemu. Preferowane systemy samodokumentujące się. Powiązania dwustronne (ogólna do szczegółowej i szczegółowa do ogólnej)
884
885 <!-- _footer: '**\#AGILE**' -->
886
887 ---
888 ## **\#Iterate** a lot
889
890 *Continuous Architecture*
891
892 - Review Architektury
893 - Definition of Done review
894 opracowane ze wszystkimi stakeholderami
895 - Iteracyjna praca nad doбором narzędzi do dokumentacji oraz struktury zarządzania projektem architektury
896 - Retrospekcje na temat współpracy
897
898 <!-- _footer: '**\#AGILE**' -->
899
900 ---
901
902 ## **\#Methodology** it's fun
903
904 - *Event Storming*
905 jako metoda do wydobywania informacji dostępna dla wielu bardzo odmiennych stakeholderów
906 ![w:1030 invert:100%](images/storm.png)
907
908 <!-- _footer: '**\#AGILE**' -->
909
910 ---
911 ## **\#Methodology** it's fun, fun

wiązanie które inAa .ab.*No results↑↓≡×

Preview README.md

Journey Mapping

SPECIFIC USER + SCENARIO + GOALS

PHASE 1	PHASE 2	PHASE 3
1. _____ 2. _____ <div></div>	3. _____ 4. _____ 5. _____ 6. _____ <div></div>	7. _____ 8. _____ 9. _____ <div></div>

OPPORTUNITIES + INTERNAL OWNERSHIP

#Product and more, more

Dokumentacja

- Repozytorium dokumentacji architektonicznych w jednym miejscu, dostępna na różnych poziomach, czytelna dla osób nie związanych z IT.
- Prezentacja architektury w modelu *C4*
Tam gdzie to wymagane odpowiednio szczegółowa.
- Zwinna i lekka w modyfikacji, z możliwością *wersjonowania* i *code review*.

#AGILE

Software System

ner

Container
(e.g. client-side web app, server-side web app, console application, mobile app, database schema, file system, object store, etc)

Cont
(e.g. Client-side web app, server-mobile app, database schem

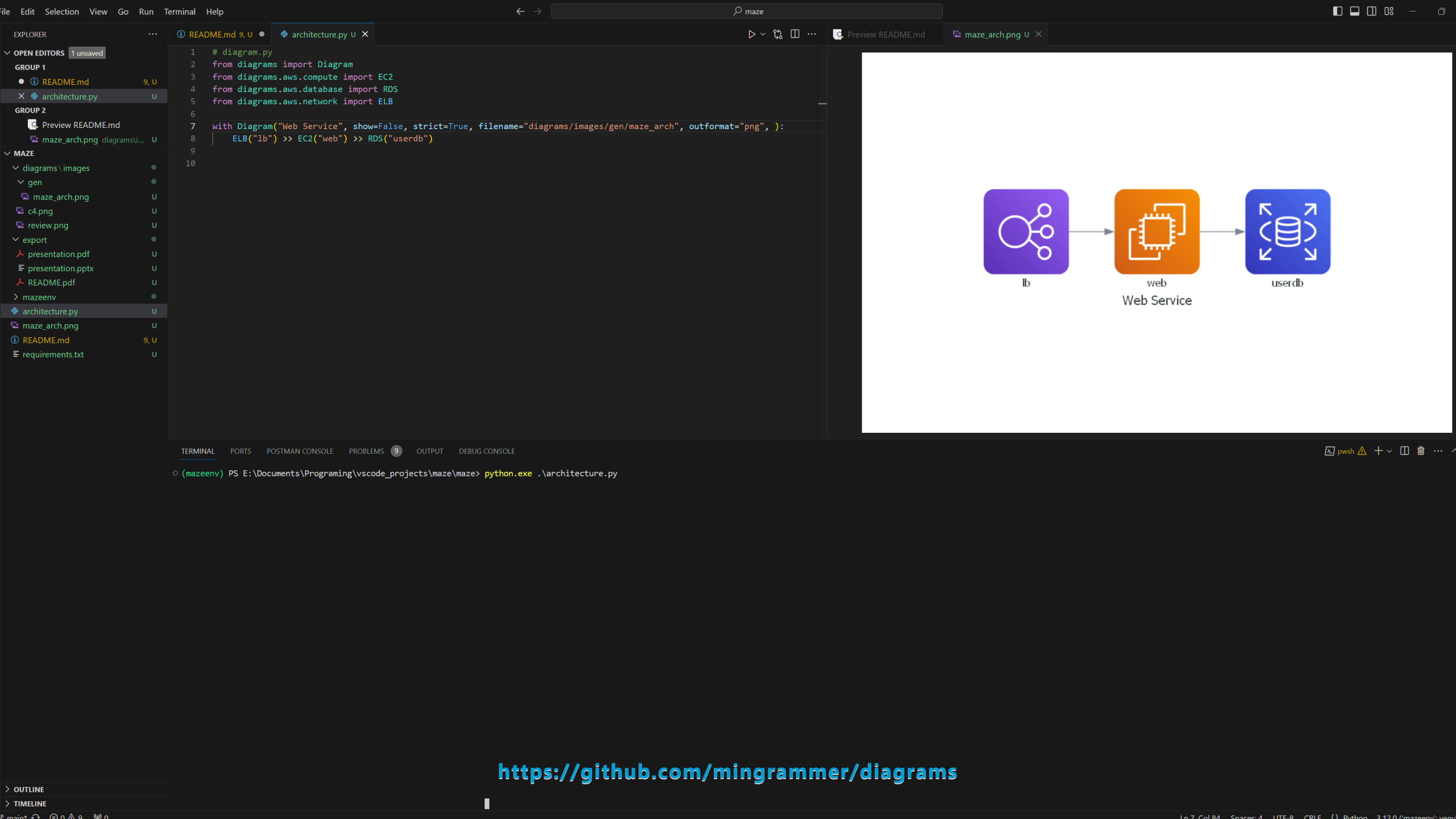
Component

Component

Component

https://marp.app/

Ln 870, Col 79 (32 selected) Spaces: 2 UTF-8 CRLF Markdown 41 Spell



<https://github.com/mingrammer/diagrams>