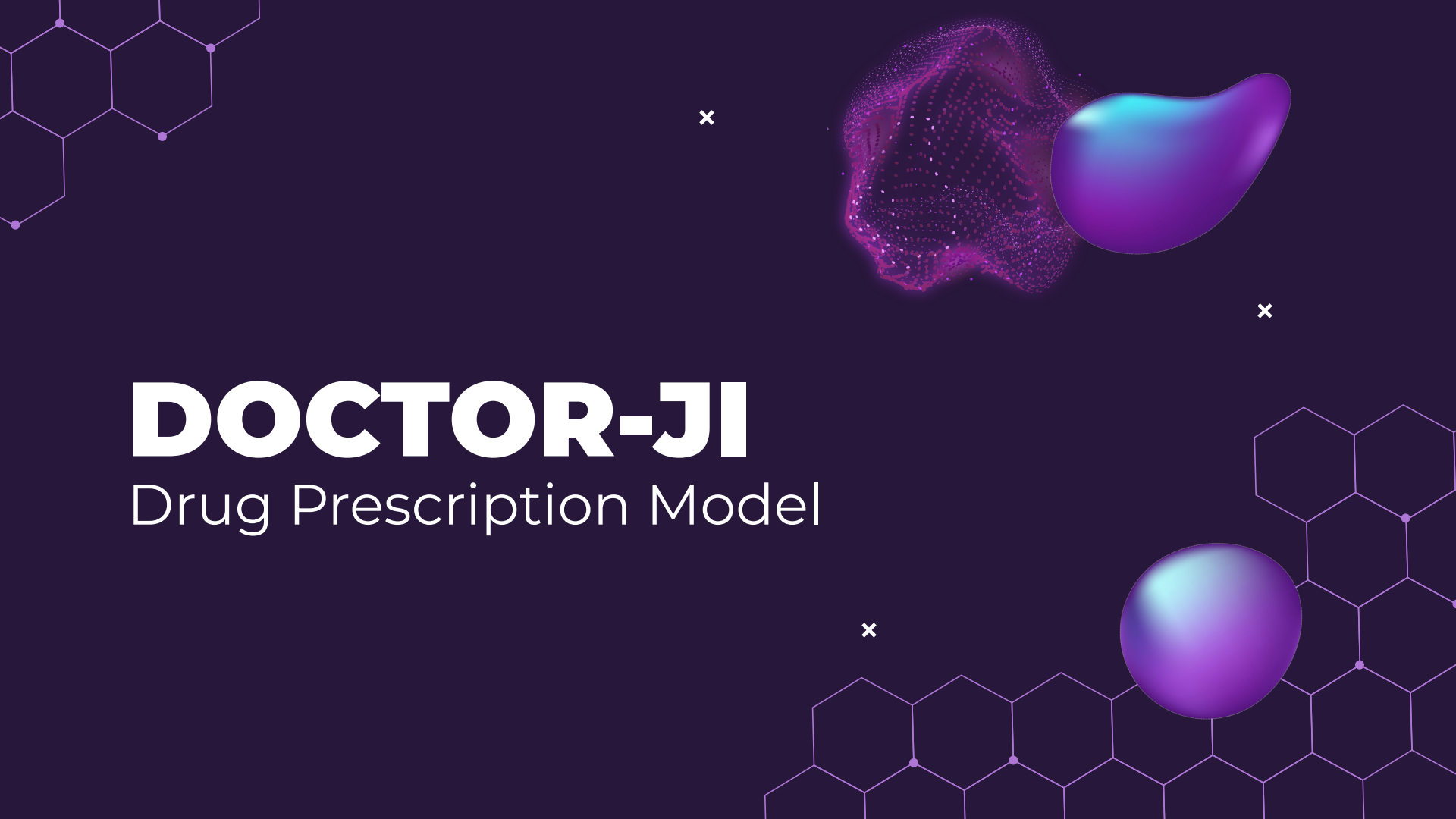


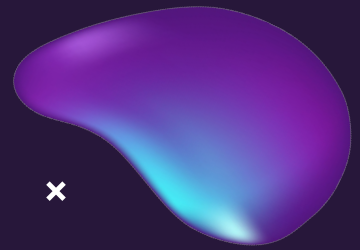
# DOCTOR-JI

Drug Prescription Model





# TABLE OF CONTENTS



- 01 OVERVIEW**
- 02 ABOUT THE PROJECT**
- 03 PROBLEM STATEMENT**
- 04 ABSTRACT**
- 05 SNAP-SHOTS**

# WHAT ARE WE WORKING ON?

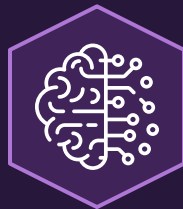
x



## ASK

Please feel free to share what you're feeling or any specific symptoms or questions you have

+



## FEED

You can provide me with the details of the question and any relevant context

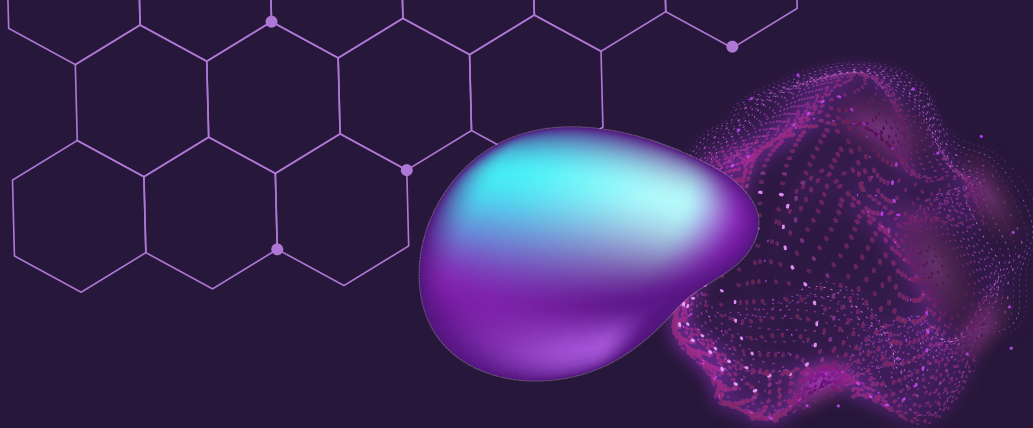


x

## IDENTIFY

I'd be happy to help, generate, provide information to the best of my knowledge.





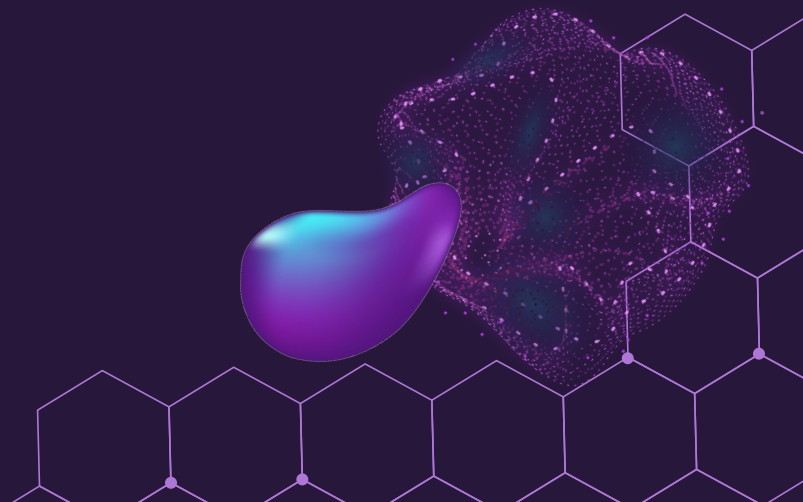
x

# ABOUT THE PROJECT

**Drugs for Common Treatment**



+





## 1. Data processing:

x

Clean and organize your dataset. Ensure that data is accurate and consistent. Handle missing values, if any, using appropriate techniques.


## 2. Text Processing and Natural Language Understanding:

Implement natural language processing techniques to understand user queries.


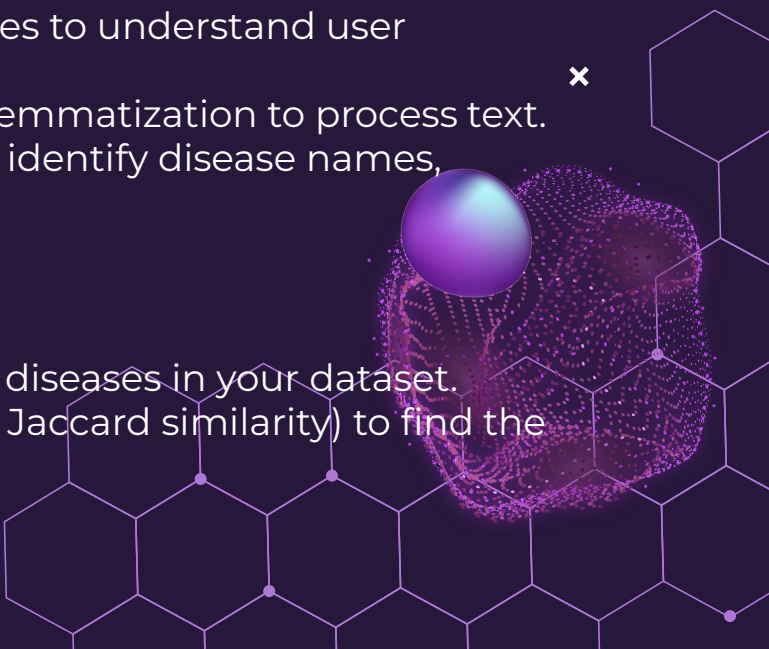
Use techniques like tokenization, stemming, and lemmatization to process text. Consider using named entity recognition (NER) to identify disease names, medicine names, and other relevant entities.

x

## 3. Query Matching and Retrieval:



Develop a mechanism to match user queries with diseases in your dataset. Use text similarity measures (e.g., cosine similarity, Jaccard similarity) to find the most relevant diseases based on user input.



#### 4. Medicine Recommendation:

For the selected disease, retrieve the associated remedies and medicines from your dataset.

You could use a rule-based approach, where you pre-define which medicines are associated with each disease.

Alternatively, you could use machine learning or deep learning models to learn associations between diseases and medicines based on your dataset.

#### 5. Testing and Validation:

Test your system with a variety of queries to ensure accurate disease matching and medicine recommendations.

Validate the recommendations with medical experts or professionals to ensure the correctness and safety of the suggestions.

#### 6. Continual Improvement:

Gather user feedback and continuously improve your system based on user interactions and suggestions.






# PROBLEM STATEMENT

**Drugs for Common Treatment**




+


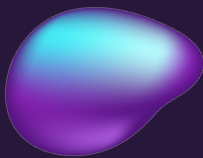





In the world of healthcare, it's really important to give people the right medical advice at the right time. But there's a tricky problem we need to tackle called the "cold start." This happens when we don't have enough past information about a person to suggest the best medicine for their illness. Imagine if you just started using a health app, and it doesn't know anything about your medical history yet – that's the "cold start."



Now, there's this cool tool called the "Bolt" package. It's like a super-smart helper that understands what people say in everyday language. We want to use this tool to help us understand what people are asking about their health.



So, our project is all about using a big collection of information about common diseases. We have details about what each disease is like, what precautions to take, and what remedies can help. With this info, we want to make a system that can suggest the right medicine for a disease, even if we don't know much about the person asking.

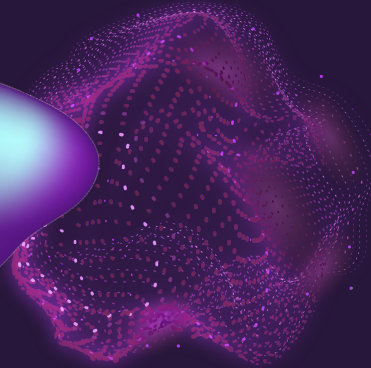
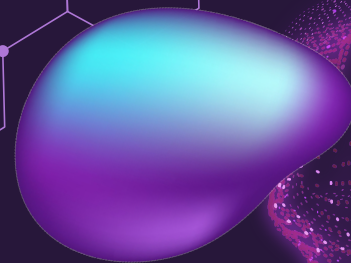


# ABSTRACT

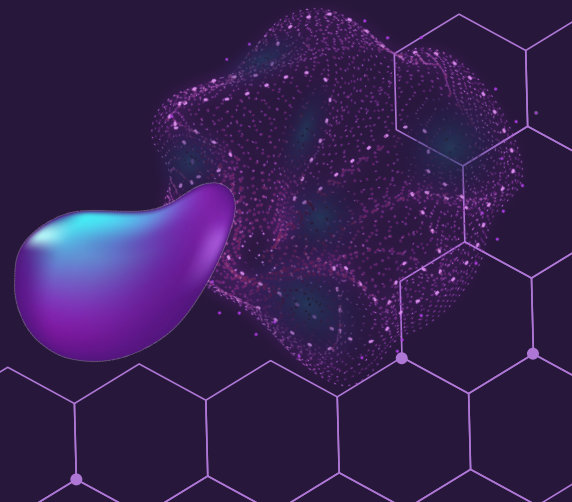
**Drugs for Common Treatment**

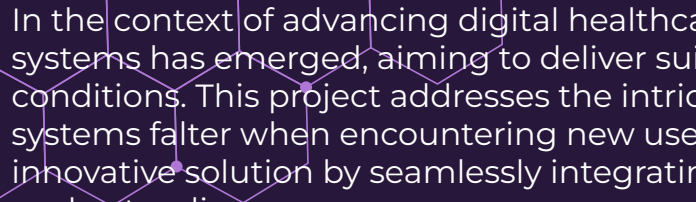


+



x



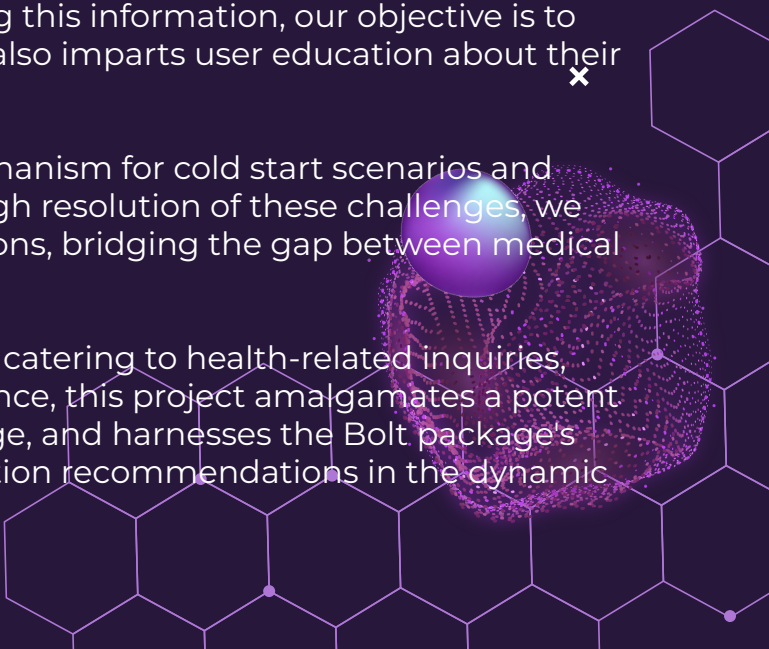


In the context of advancing digital healthcare, the demand for intelligent medical recommendation systems has emerged, aiming to deliver suitable medication guidance based on user symptoms and conditions. This project addresses the intricate "cold start" dilemma, where conventional recommendation systems falter when encountering new users or diseases with limited historical data. We propose an innovative solution by seamlessly integrating the versatile "Bolt" package for natural language understanding.

Our extensive dataset encompasses diverse general diseases, accompanied by comprehensive descriptions, precautions, and recommended remedies. Leveraging this information, our objective is to develop a system that not only offers medication suggestions but also imparts user education about their conditions and available treatments.

Key challenges involve constructing a robust query matching mechanism for cold start scenarios and effectively integrating Bolt's capabilities to grasp user input. Through resolution of these challenges, we strive to provide accurate personalized medication recommendations, bridging the gap between medical expertise and technological innovation.

By meticulous testing, we aim to establish a user-friendly interface catering to health-related inquiries, continually enhancing the system based on user feedback. In essence, this project amalgamates a potent medical dataset, addresses the intricacies of the cold start challenge, and harnesses the Bolt package's potential, culminating in an intelligent solution for precise medication recommendations in the dynamic landscape of digital healthcare.

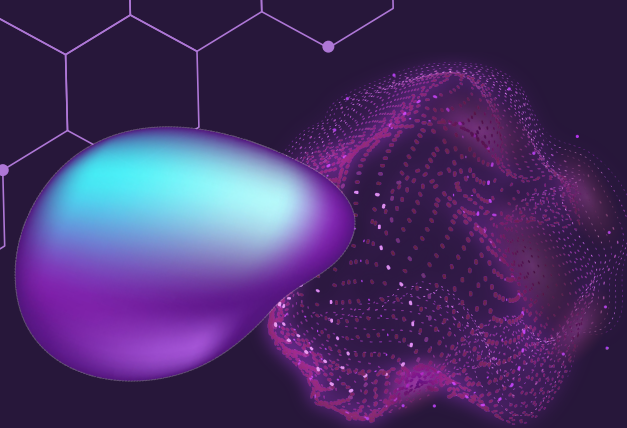


# SNAP-SHOTS

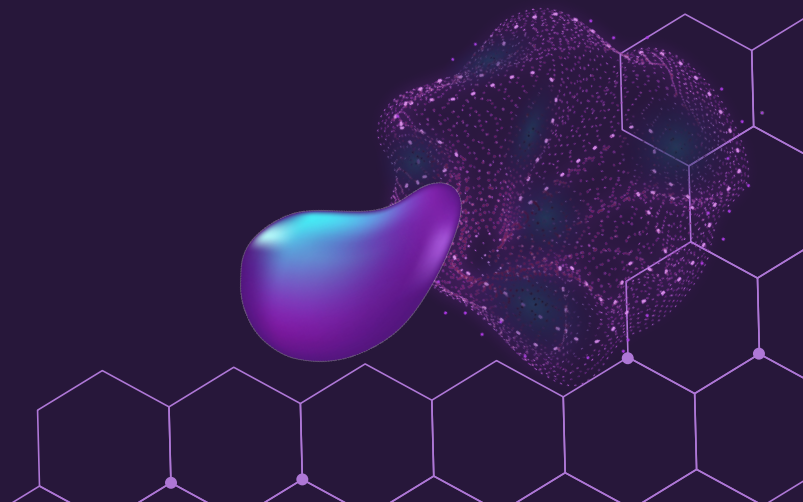
**Drugs for Common Treatment**



+



x



## ▼ Importing dataset and preprocessing data



```
import pandas as pd
import os

catalog_file="/content/drive/MyDrive/ThirdAI_Hackathon/Main Project Files/drugs_for_common_treatments.csv"

def sample_catalog(catalog_file, percentage=0):
    df = pd.read_csv(catalog_file)
    df["alcohol"]=df["alcohol"].fillna(0)
    df["alcohol"]=df["alcohol"].replace("X",1)

    df = df.sample(frac=percentage, random_state=0)
    df["DOC_ID"] = [i for i in range(df.shape[0])]
    df["drug_name"] = df["drug_name"].str.lower()
    df["rating"]=df["rating"].fillna(df.rating.mean())
    df.rating=df.rating.astype("str")
    df.alcohol=df.alcohol.astype("str")
    df["medical_condition"] = df["medical_condition"].str.lower()
    df["medical_condition_description"] = df["medical_condition_description"].str.lower()
    df["rx_otc"]=df["rx_otc"].str.lower()
    df["rating"]=df["rating"].str.lower()
    df["alcohol"]=df["alcohol"].str.lower()
    #
    sampled_catalog_file = f"/content/drive/MyDrive/ThirdAI_Hackathon/Main Project Files/drugs_for_common_treatments-{percentage}.csv"
    df.to_csv(sampled_catalog_file, index=False)
    #
    return sampled_catalog_file, df

sampled_catalog_file, dataframe = sample_catalog(catalog_file, 0.75)
```

## Calling the Function



```
result=top_k_drug('i am having cancer ', 5)
```

```
{'DOC_ID': 1276, 'drug_name': 'ifosfamide', 'medical_condition': 'cancer'}
```

```
*****
```

```
{'DOC_ID': 974, 'drug_name': 'carboplatin', 'medical_condition': 'cancer'}
```

```
*****
```

```
{'DOC_ID': 2849, 'drug_name': 'vincristine', 'medical_condition': 'cancer'}
```

```
*****
```

```
{'DOC_ID': 2029, 'drug_name': 'etopophos', 'medical_condition': 'cancer'}
```

```
*****
```

```
{'DOC_ID': 2616, 'drug_name': 'thiotepa', 'medical_condition': 'cancer'}
```

```
*****
```

# TOP 5 RESULTS

## ▼ Most Recommended Medicine

✓  
0s



```
drug_name=dataframe.iloc[result[0],1]  
print(drug_name)
```

ifosfamide

**MOST  
EFFECTIVE  
DRUG**

## Showing Information For the Most Recommended Medicine

```
import wikipedia
summary = wikipedia.summary(drug_name, sentences=5)
sentences = summary.split('. ')
print("Information about the Drug :",drug_name)
for sentence in sentences:
    print(sentence + '.')
```

Information about the Drug : ifosfamide

Ifosfamide (IFO), sold under the brand name Ifex among others, is a chemotherapy medication used to treat a number of types of cancer. This includes testicular cancer, soft tissue sarcoma, osteosarcoma, bladder cancer, small cell lung cancer, cervical cancer, and ovarian cancer. It is administered by injection into a vein. Common side effects include hair loss, vomiting, blood in the urine, infections, and kidney problems. Other severe side effects include bone marrow suppression and decreased level of consciousness. Use during pregnancy will likely result in harm to the baby..

# DETAILED DESCRIPTION



# OUR TEAM

**Dhruval Patel**

<https://www.linkedin.com/in/dhruval-patel-7926b3213/>

×

**Khush Fadadu**

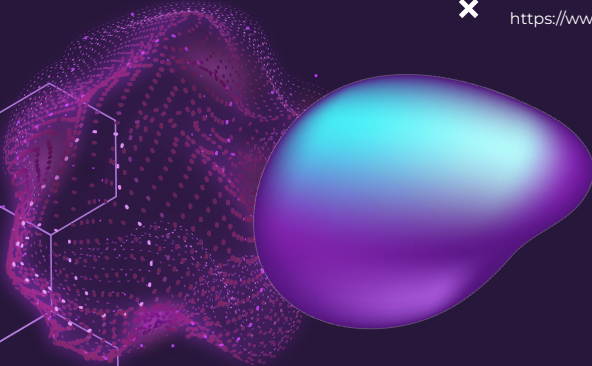
<https://www.linkedin.com/in/khush-fadadu-237925201/>

×

**Shlok Dhokai**

<https://www.linkedin.com/in/shlok-dhokai-29053b213/>

×







THANKS!!

