

# Intro to Engineering

Baibhav Bista

# Who am I?

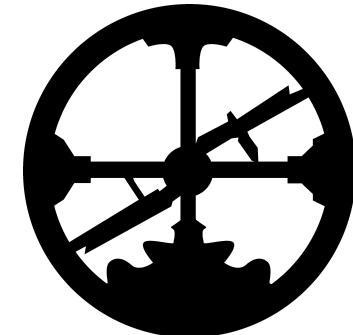
Baibhav Bista

Work as Software Engineer at Roam Research

073BCT

Other relevant bits:

- Software Coordinator at LOCUS 2020
- Got the NCELL award for topping my class 3 out of 4 times



# **Who are you?**

# **Who is this talk for?**

THIS GUY!



# Broad Outline of this session

- My Journey
  - Before BE
  - BE
  - After BE
- What to expect from your Engineering Degree
- Advice
  - Preamble: Some blind spots
- Appendix
  - Recommended materials

# **My Journey**

# My Journey: before BE

- Always loved computers



# My Journey: **before BE**

- To “keep my options open” studied
  - Accountancy instead of CS in Class 9-10
  - Biology instead of Physics/CS in Plus-two
- At some point in class 12, realized I wanted to study computers and then started preparing for BE entrance prep
  - Went all in
  - Got in! (rank 4 😊)

# My Journey: before BE: CS50

After entrance result & before  
classes started, did Harvard  
CS50 course online  
(yes, all assignments too)

I believe this gave me a solid  
foundation and head start



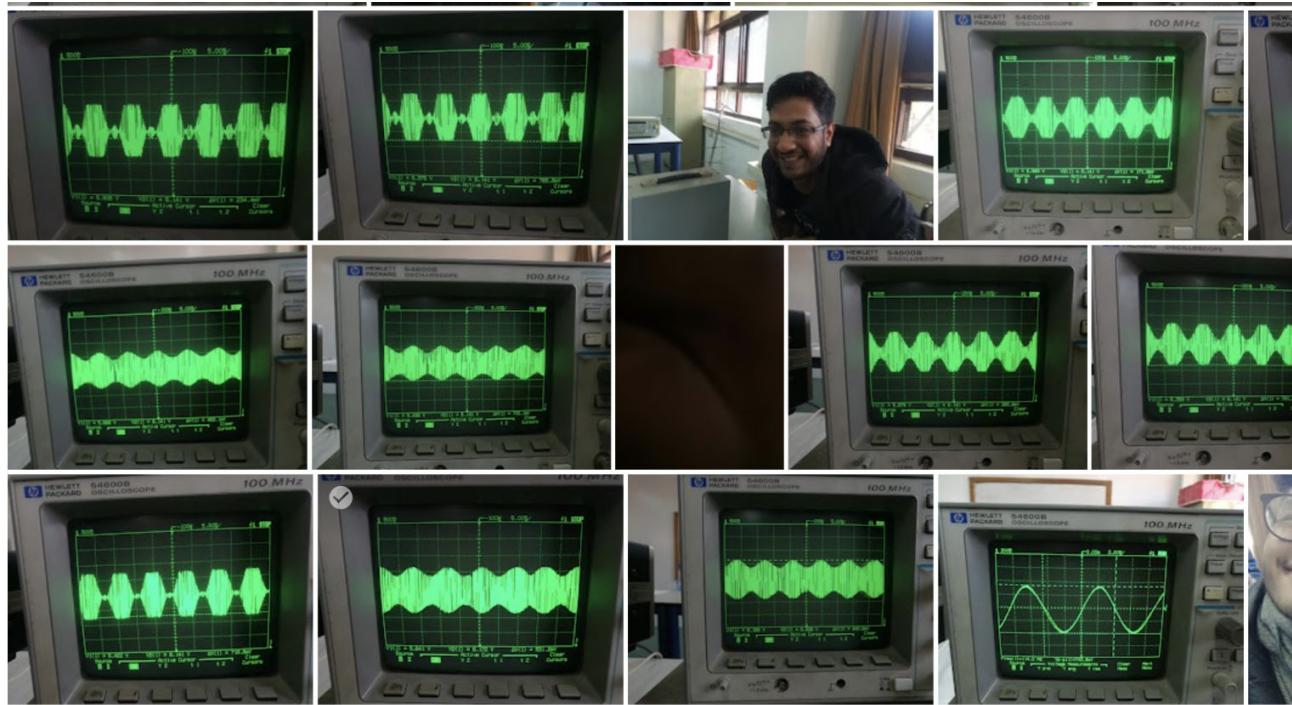
# My Journey: BE: Academics

- Boring section so let's keep this short
- (We'll get into details later in the Advice section)
- Studied well (paid attention in classes & worked hard the 3 days before exams)
- Topped my class & got NCELL scholarship 3 out of my 4 years



# My Journey: BE: Academics

Random glimpse at my camera roll: Oscilloscopes galore!



# My Journey: BE: CS50 revisited

Head start to the core subjects:

- C
- Data structures & algorithms
- Python
- & even a bit of web dev



If you feel a bit rusty/behind in your core CS skills, highly recommend you go through the course (& do the exercises)

# My journey: BE: Competitive Programming

- Got into it in 2nd year, because some friends were organizing class-wide contests
- Made a team with 2 classmates - “pulchowk\_oders”
- 2018 - my team won the national level contest and went to the ICPC Asia Regionals in Dhaka, Bangladesh to participate



# My journey: BE: Competitive Programming

three day workshop Dec 2018



# My Journey: BE: Hackathons and other contests

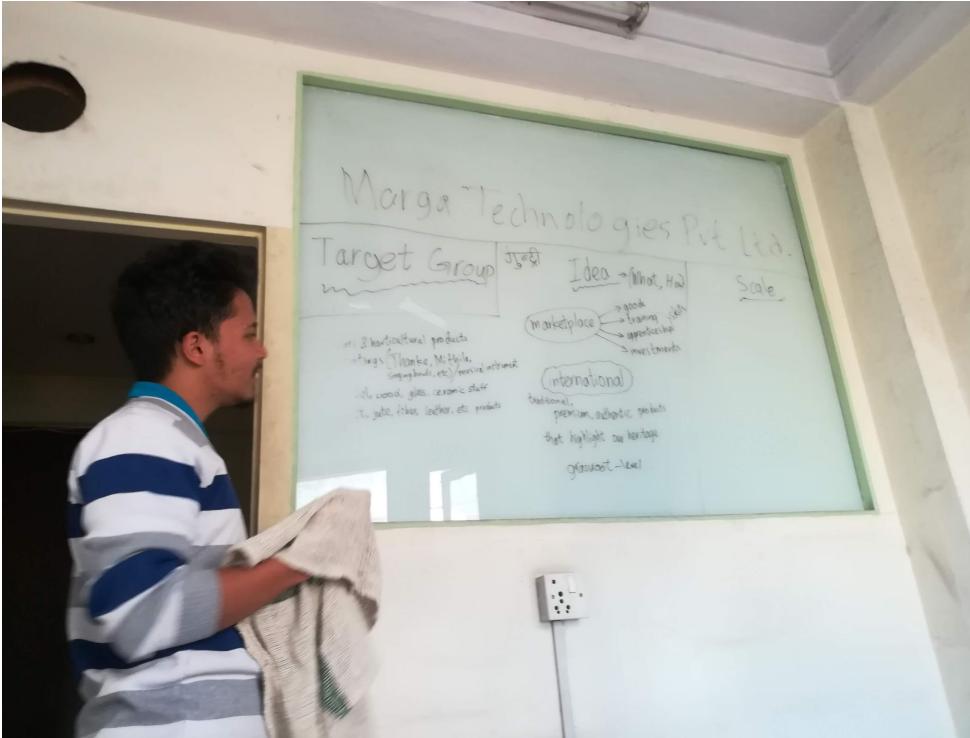


# My Journey: BE: Hackathons and other contests

- June 2018 an Assistive Technology hackathon organized in Thapathali Campus
- “Maarga” a map features for helping wheelchair-bound people
- Got second place and met my cofounders!



# My Journey: BE: Startup!



# My Journey: BE: LOCUS 2020

As Software Coordinator

Organized (& helped with):

- Software Fellowship
- HackAWeek
- CodeCamp
- CodeJam
- Software projects in the exhibition



# My Journey: BE: LOCUS 2020



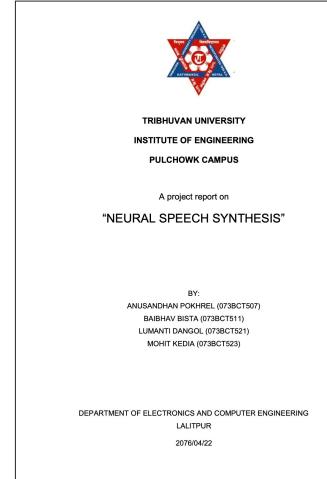
# My Journey: BE: Minor Project

End-to-end trained Text to Speech model

Guided by Paaila Technology

Recreated English model and had similar performance

Some work on the Nepali model

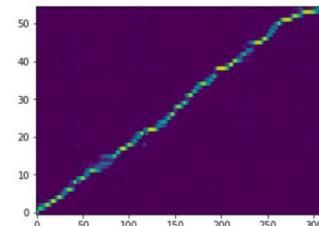
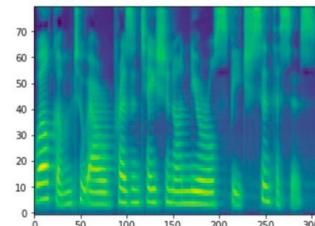


## Samples

```
In [9]: mel_outputs = generate_mel("Welcome to our presentation on Neural Speech Synthesis.")
with torch.no_grad():
    audio = wavglow.infer(mel_outputs, sigma=0.666)
    ipd.Audio(audio[0].data.cpu().numpy(), rate=hparams.sampling_rate)
```

Out[9]:

▶ 0:03 / 0:03 ⏸ ⏴ ⏵ ⏴

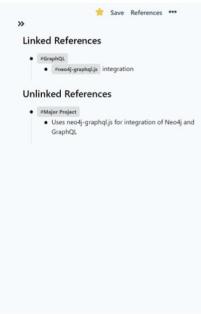
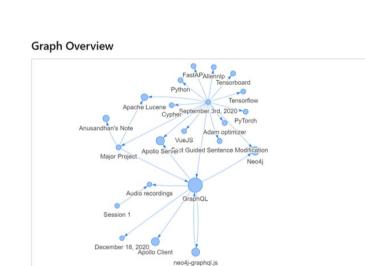
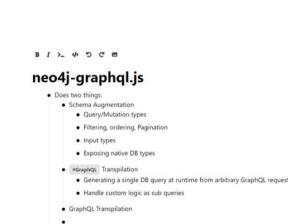


# My Journey: BE: Major Project

# “Wiki with Fact Verification and Fact-Guided Sentence Modification”

A full featured note-taking outliner app, with AI features like:

- ability to verify a fact based on stored data
  - change text (sentences) in the notes if incorrect



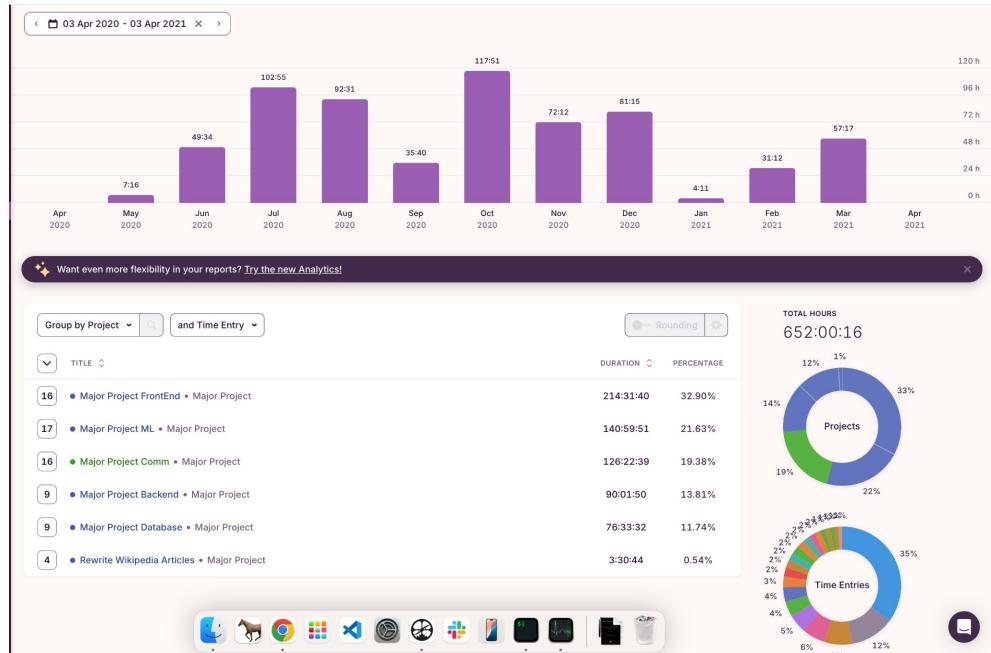
# My Journey: BE: Major Project

By far the most effort I'd ever applied on a project

Roughly 650 hours (from May 2020 to March 2021)

(It was lockdown and I had nothing else to do 😅)

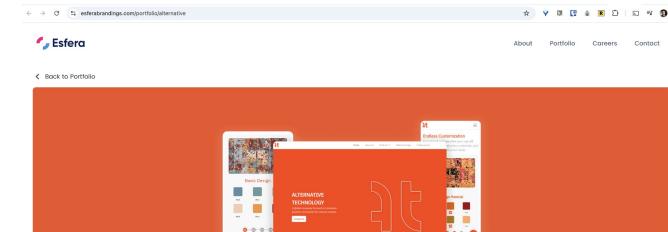
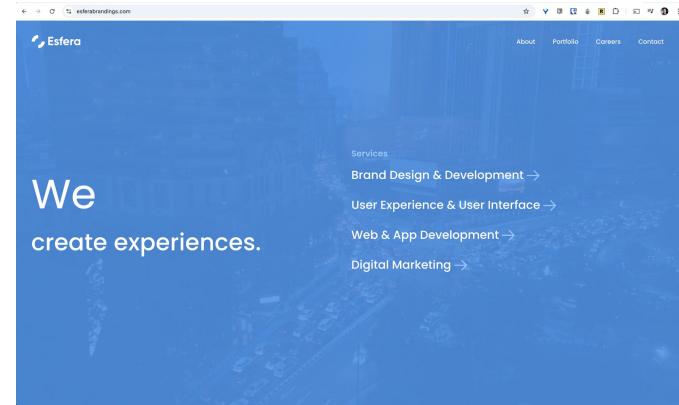
Surprisingly ended up paying off



# My Journey: After BE: Esfera (2019 - 2022)

Continued working on the startup I started with my friends in my second year

Lots of projects: starting from a data collection app to websites to AR apps



Client  
Alternative Technology

Technologies

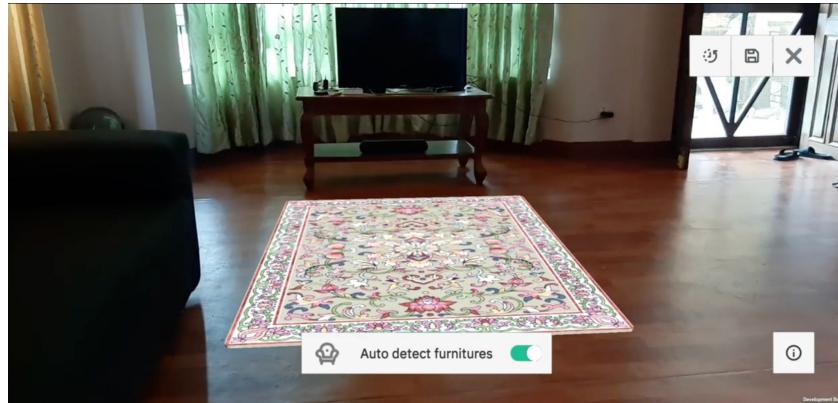


Project  
Corporate Website Re-design

Deliverables

Branding, Web Development

# My Journey: After BE: Esfera (2019 - 2022)



Top 5 Finalist at ICT Award 2021 in the  
Rising Star Innovation Category

# My Journey: After BE: Roam Research (2021 - Present)

Software Engineer

Full Stack (& mobile)

Small team so wear a lot of hats

(Major project paid off)



A note-taking tool  
for networked thought.

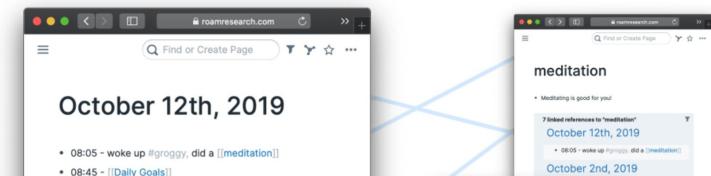
As easy to use as a document. As powerful as a graph database.  
Roam helps you organize your research for the long haul.

[Sign Up](#)

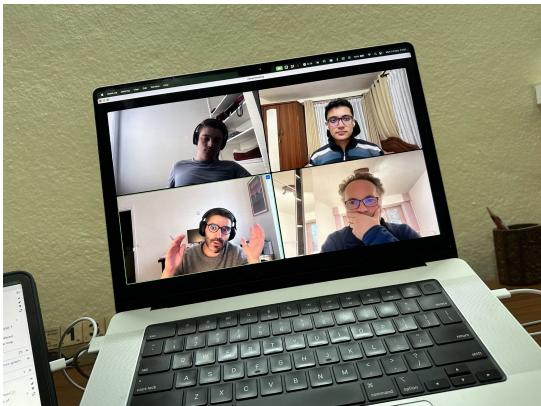


For Macs with Apple Silicon chips (2020 on), click [here](#) to download

Also available on [Windows](#) and [Linux](#)



# My Journey: After BE: Roam Research (2021 - Present)



# What to expect from your engineering degree

(He finally got into the main “Intro to engineering” part!)

**Important mindset:** Expect that you will have to learn everything yourself

- Good experience also for future because you will need to keep learning all throughout your career (this is more the case for our field than other fields)

# Advice: Preamble: My blind spots

- I have not done any academic research
  - Except for Minor project but that didn't work out for reasons stated before
  - So, I do not have good advice on getting papers accepted
- I have only worked at startups and small companies.
  - Not sure what is important for medium-scale/large companies
- Internships
  - I haven't done any.
  - I do think they are a good idea and you should do them though

## Advice: Go All In

- What's actually not a bad idea is GOING ALL IN on Computer/Electronics Engineering. Say to yourself I've dedicated the next 4 years to this, I will not try to look for greener pastures somewhere else (I have enough time in the future for that)
  - for 4 years, this is your job!
  - you have the benefit of focusing on this being the objectively best thing you can be doing with your time right now. After you start your career you will not have that

# Advice: How to study

- Key terms:
  - Active Recall
  - Ebbinghaus Forgetting Curve
  - Interleaving
- Simple tip
  - every day after class (or even between classes), try to recall (without looking at any reference material) what was taught
  - Before class, do the same for the class
  - Feynman Technique?
- General advice: Desirable difficulty

# Advice: How to study

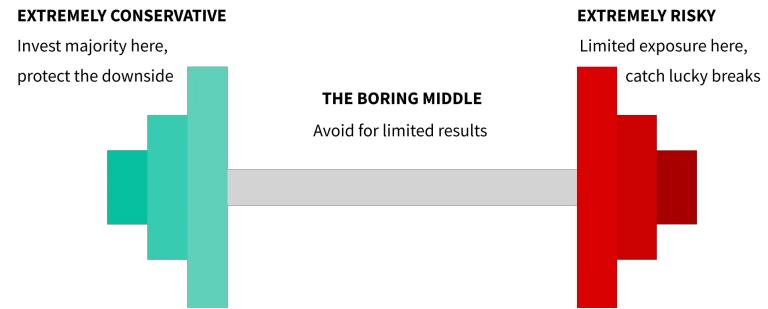
Barbell strategy

Two types of subjects:

- Your core subjects
- Everything else

Different strategies for them

Caveat: a lot of the courses (& what you think of as outdated subjects) are probably more important than you're assuming right now



## **Advice: Try not to fail subjects (the dreaded “backs”)**

- If you do get backs, clear them as soon as possible
- too much mental pressure, particularly as they start to accumulate

# Advice: Learn the history of our field

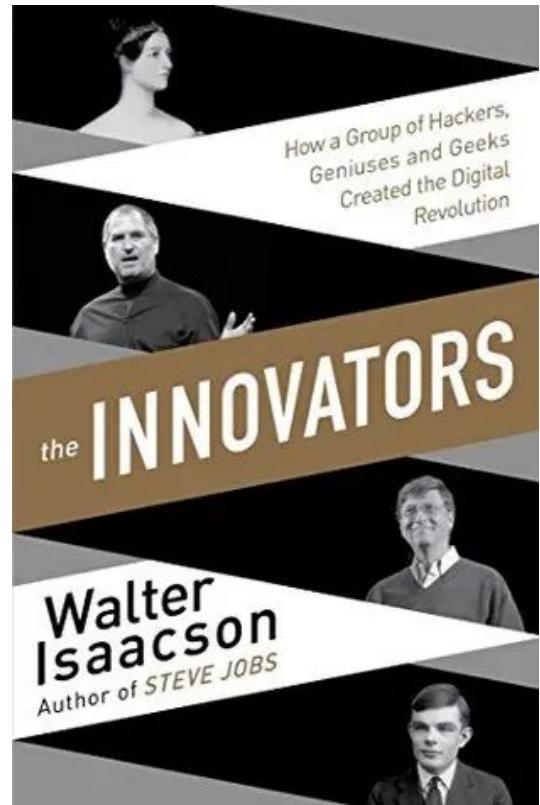
Read the book “The Innovators” by Walter Isaacson

Ada Lovelace, Charles Babbage

Alan Turing, Von Neumann

Robert Noyce, Gordon Moore

Bill Gates, Steve Jobs, Tim Berner's Lee, Larry Page



# Advice: Projects (do a lot of them!)

- Try to do a project for each core class
- Examples
  - DSA class
    - Code up some obscure data structure like a piece table
  - OS class
    - Perfect opportunity to build a mini File System (you learn inodes, directories, permissions, etc)
  - DBMS class
    - What better what to learn about data representations than to code up your own key value store?

# Advice: Projects (do a lot of them!)

- Lean towards what you find **interesting**
- Common pitfall is to make very simple React webpages as projects
  - Good to do at the beginning to learn
  - But please, think bigger! (I know you can do it!)



## Advice: **Extra-curriculars**

- One of the things I regret is not doing more of these
- Realize that some classes are .. ahem .. skippable

# Advice: If you're worried about AI

- AI is concerning, and the problem is that no one knows what the situation will be in 4 years
- TBH I would love to be in your position
- When learning to code, it is probably not a good idea to highly leverage AI
- That being said, do not shy away from AI after you have learned the basics.
  - Most old people have a bias against AI, because it is risky and they are concerned for their jobs and skills they've developed over a long period of time. By the virtue of just starting right now, you have a psychological edge and none of this baggage. Take advantage of this!

# Advice: Do not worry about choosing your sub-field immediately

- Premature optimization is not good, not in code and neither in life
- Learn what interests you
- Instead of aiming at a particular thing and only working towards that:
  - Arm yourself with a broad range of skills/tools
  - Note how the landscape is changing
  - And embrace the opportunities as they come along
  - Steve Jobs: “You can't connect the dots looking forward; you can only connect them looking backward”
- This does NOT excuse you from working hard!

# **Q&A Session**

# Thank you!

If further questions, you can contact me at:

<https://www.facebook.com/baibhavbista5/>

<https://x.com/baibhavbista>

# References: Books I recommend

- The Innovators - Walter Isaacson
- Compilers - Crafting Interpreters
- Algorithms - How to think about algorithms - Jeff Edmonds
- Distributed systems - Understanding Distributed Systems
- So good they can't ignore you - Cal Newport