

1. Product analysis

Ideation -> collaboration -> goal -> market analysis -> feature analysis(Kano) -> risk

2. Agile methodology -a way to do work, good for small teams

*waterfall methodology, good for big teams with yearly goals

Trello Board (cards): track task/progress for management

DSU: **Daily set up** 10min (10am): - yesterday

- today

- obstacles

Actively give feedback: what you gonna do? what to improve? be proactive.

SDLC: **Software development life cycle**

1) product analysis

2) design (UI:user interface/UX:user experience)*

Between 1) and 2), user's story map(general activity -> task -> priority)

3) development**

4) testing - developers do some; **UAT**(internal people): user acceptance
test(software bugs, not coding bugs)

3), 4) create a repeating cycle with reviews

5) launch/go live

6) marketing - gain users
- retain users

7) feedback and review - larger scale than UAT

1) to 7) create a software development life cycle

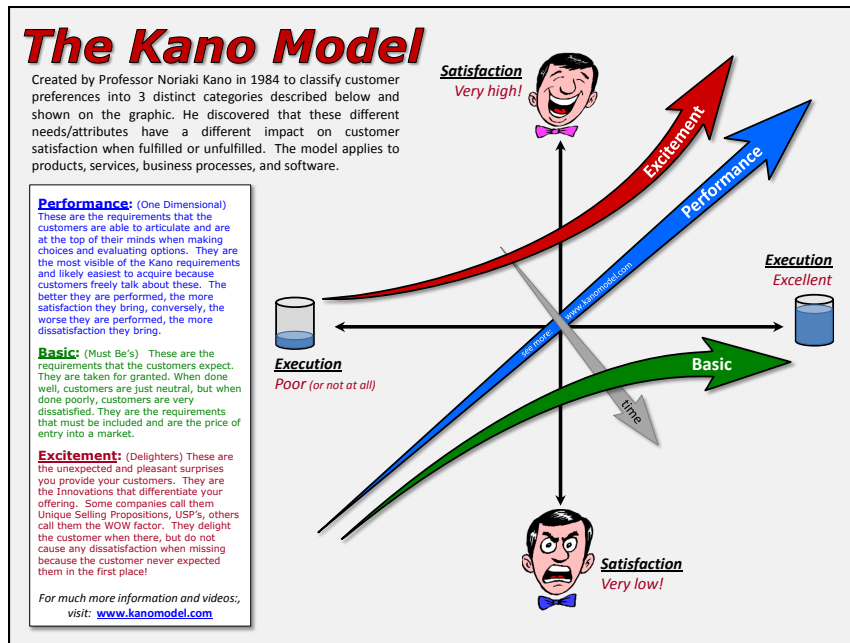
*UI/UX designing softwares: XD, invision studio, framer

Designers: think about where to put the buttons/how the buttons should look like/
prototype(animations, linking all the screens)

**Coding process: robotics(AI), *website application*, data science(self-driving car),
cyber security, games development

Website application: front-end(what you see; HTML, CSS and javascript) and
back-end(processes behind; python, Dhp, C++, Ruby)

2. Kano model -a way to determine whether this feature is good or not



Three types of features: basic(login), performance(archiving chat), excitement(animated stickers), time(aka shitty feature)

*features are changing over time!

- analysis of features over time

Today's tasks: 1) BooMail

-Feature analysis

-Market analysis -> why we need this feature

-> who, how

-User story: general activity -> screenshot -> task -> priority

2) Marketing

-Google Ads

-Facebook Ads

-Automation tools (AMS: **advertising management system**)

-Why people use this for marketing and how

-Presentation

3) Boogle Academy

-Website (**Wordpress, wix**)

-Feature analysis

-Marketing analysis

-User story

-Development

3. Blockchain 101

- 1) Definition: A database or a ledger that maintains a list of records. That is decentralised and distributed(without a centralised server).
- 2) Advantage: If one particular source is down, users cannot retrieve their documents in a centralised server. However, decentralised database has information stored more than one nodes.

Shared publicly, decentralised, secure, trusted automated — accessed by all stakeholders it cannot be manipulated

- 3) Conventional: trusted party—bank, charges transaction fees/administrative fee
Blockchain: directly from the buyer to the seller, no middle man (Project Ubin: cross boarder payment)
- 4) Why is it secure? Every record is written in blocks. Every block is chained to each other. For anyone to change a record, they will have to make changes to all the blocks before it. In contrast, hackers just need to hack the centralised server.

- 5) Types of blockchain: *A digital currency can only exist with a blockchain.

Bitcoin: 2008, uses proof-of-work consensus algorithm.

Open-end: transaction!

Back-end: verify it is a real transaction — system throw out a mathematical question — miners(people who solve the mathematical problems use their computers) — transaction fixed and stored in the blockchain.

Ethereum: 2015, uses proof-of-stakes consensus algorithm, uses **smart**

contract(when all requirements are met) to ensure people are paid their insurance money with a claim(commonly used by insurance companies; is there delay? is there sufficient money?)

- * Use cases for smart contracts: main focus is on transparently and accountability of transactions(ripple, stellar, filecoin, vechain, civic etc.)

eg. In Estonia and Malta? identity is stored in blockchain(transparent to the public), don't need an identity card.

- * Consensus protocol: how does the network validate

Proof-of-work: the first miner to **solve** the problem will benefit and receive block rewards. the rest of teh miners will then **validate** the solution.
10 min every time, more lucrative

Proof-of-stake: the richest(with the most stakes) miner get the chance to validate transactions. Miners will get “transaction fee” but not rewards)
40-50 cents every time

4. Encryptions

*encryption refers to the mathematical process of making a message unreadable except to a person who has the key to decrypt it into readable form.

Encrypting data **in transit** vs at rest:

1) transport-layer encryption(third-party encryption):

Because your messages can be seen by (and are often stored on) company servers, they may be vulnerable to law enforcement requests or leaking if the company's servers are compromised. (eg. VPN, https)

2) end-to-end encryption:

Private key encryption: a form of encryption where only a single private key can encrypt and decrypt information. It is a fast process since it uses a single key. However, the private key may be stolen or leaked. Key management requires prevention of these risks and necessitates changing the encryption key often.

Public key encryption: uses public key and private key;

Traditional encryption systems use the same key to encrypt and decrypt a message. Public key encryption uses two keys: one to encrypt, and another to decrypt. It means that you can **use public key to encrypt messages to you**, and **use private key to decrypt the messages**.

**Protect the contents of your messages, text and files from being understood by anyone except intended recipients.*

Prove that a message came from a particular person and has not been altered.








PGP(pretty good privacy) encryption: uses digital signature, requires a PGP passphrase; PGP encryption uses a mix of data compression, hashing, and public-key cryptography. It also uses symmetric and asymmetric keys to encrypt data that is transferred across networks. It combines features of private and public key cryptography. Each step uses a different algorithm, and each public key is associated with a username and an email address.

(<https://koolspan.com/private-key-encryption/>)

(<https://ssd.eff.org/en/module/what-should-i-know-about-encryption>)

(<https://digitalguardian.com/blog/what-pgp-encryption-defining-and-outlining-uses-pgp-encryption>)

5. The Business Model: not writing concepts, but specifically what you gonna do

The Business Model Canvas		Designed for:	Designed by:	Date:	Version:
Key Partners  <p>Who are our Key Partners? Who are our key suppliers? Which Key Resources are we acquiring from partners? Which Key Activities do partners perform?</p> <p>NOTATIONS FOR PARTNERSHIP Specialization and synergy Reduction of risk and uncertainty Acquisition of particular resources and activities</p> <p>Can they market us? Can they provide products/service?</p>	Key Activities  <p>What Key Activities do our Value Propositions require? Our Distribution Channels? Customer Relationships? Revenue Streams?</p> <p>CATEGORIES Production Problem Solving Platform/Network</p> <p>How we recruit members? What do we need to produce?</p>	Value Propositions  <p>What value do we deliver to the customer? Which one of our customer's problems are we helping to solve? What bundles of products and services are we offering to each Customer Segment? Which customer needs are we satisfying?</p> <p>CHARACTERISTICS Novelty Performance Customization "Selling the pig down" Design Price Self-Reduction Risk reduction Accessibility Convenience/Usability</p> <p>eg. IOS provides a common platform for users to develop</p>	Customer Relationships  <p>What type of relationship does each of our Customer Segments expect us to establish and maintain with them? Which ones have we established? How are they integrated with the rest of our business model? How costly are they?</p> <p>EXAMPLES Personal assistant Dedicated Personal Assistance Self-service Automated Services Communities Co-creation</p> <p>1. List down our potential costumers(a broad or a small one) Send email server to companies(eg. Go Daily, Gmail/BooMail) 2. Maintain good relationship with the partners(Free demo/trial, chat box, dedicated package) ==If customers help you, how can you give back?</p>	Customer Segments  <p>For whom are we creating value? Who are our most important customers?</p> <p>Mass Market Niche Market Diversified Strategic Multi-sided Platform</p>	
	Key Resources  <p>What Key Resources do our Value Propositions require? Our Distribution Channels? Customer Relationships? Revenue Streams?</p> <p>TYPES OF RESOURCES Physical Intellectual Human Financial</p> <p>Who is the person that can do...?</p>		Channels  <p>Through which Channels do our Customer Segments want to be reached? How are we reaching them now? How are our Channels integrated? Which ones work best? Which ones are most cost-efficient? How are we integrating them with customer routines?</p> <p>CHANNEL STRATEGIES 1. Awareness How do we raise awareness about our company's products and services? 2. Distribution How do we help customers evaluate our organization's value Proposition? 3. Purchase How do we allow customers to purchase specific products and services? 4. Delivery How do we deliver a Value Proposition to customers? 5. After sales How do we provide post-purchase customer support?</p> <p>ISP: internet server provider (a better bandwidth to host the server)</p>		
Cost Structure <p>What are the most important costs inherent in our business model? Which Key Resources are most expensive? Which Key Activities are most expensive?</p> <p>IS YOUR BUSINESS MODEL Cost Driven: Minimal cost structure, low price value proposition, maximum automation, extensive outsourcing Value Driven: Based on value creation, premium value proposition</p> <p>SAMPLE CHARACTERISTICS Fixed Cost: Salaries, rents, utilities Variable costs Economies of scale Economies of scope</p>	Cost to manpower/welfare/monetary cost <p>*financial/website developers/designers...</p>	Revenue Streams <p>For what value are our customers really willing to pay? For what do they currently pay? How are they currently paying? How would they prefer to pay? How much does each Revenue Stream contribute to overall revenues?</p> <p>PRICE Asset sale Usage fee Subscription fees Licensing/renting/leasing Advertising</p> <p>OTHER REVENUE List Price Product features dependent Customer segment dependent Usage dependent Volume dependent</p> <p>SPONSORED REVENUE Registration (Sponsoring) Field Management Real-time Market</p>	From users who want to pay for the service(the site itself is its revenue stream)		

6. Product Testing

- Clear all caches for the old version
- Read release notes — by developers
- Test different phone models
- User Acceptance Test:

Test Case Document — by testers, version control

V1.6.9: V1.6.8 unresolved bugs and V1.6.9 new bugs