First of all, why I write this to you ...
I am a SA (System Analyst), worked in this field for over 10 years. Most of my career is developing new software, more precisely, coding. Actually, I dont think I am 'old enough' to write any suggestion on IT Industry to you. But the situation is getting too worse these years that push me to write some observation of me here. I am not good at writing, so, I keep this document as simple as possible. Hope you can get my points and lead us (IT workers in HK) to a better future. If you want to discuss with me face to face, it is totally welcome.

IT Industry Current Status in HK
- **Misperception** : Hong Kong IT is not working well now. I am not tell you this because we have high work load. I say this because the quality of IT in HK now is really bad. You can ask around, most people think 'IT System' is terrible and unhelpful. The only successful part is the infrastructure (network and bandwidth).

- **Many system is built, but bad** : after all these years 'computerizing', many companies now have software to do their business, the coverage is good. But, at the same time, many of these systems are built with low quality, make the business headache. As a result, many systems are planing to have system revamp (re-build existing system). But how to ensure these revamp is improving the company? They do not have clear direction.

- **Computer familiarity in key age range is still poor** : as time goes by, the 80s and 90s are start going into industry, they are very familiar with computer which is very good for IT industry glow, but unfortunately, most of the management level staff are still detached with computer as usual. These people make the decision of how the company adopting computer systems, automation and improve productivity. But as they are detached to computer and therefore, they dont understand how computer help them.

Why we need IT in HK
- **IT is suitable for low education but smart people** : programming do not require cert., they only need people that is smart, think with creativity and have good logical thinking. Actually, even people that English is poor, can be an excellent programmer.

- **Finance is main industry in HK, and it need computer system** : finance needs computer system to help them success. Because finance is keep changing, they need tailor-made system. Also, tailor-made system help them become more successful, compared with others.

Need more smart way to guide IT, better division
People trends to think IT Industry as a whole, but actually, we have many sectors. To guide the IT industry in HK, we must have better category on them first. Then, the glow in the entire industry can be balanced. Otherwise, like the current situation, some sectors are always neglected. And people still think the entire IT industry is good. Here is some simple categories:
- Hardware reselling
- Software reselling
- Software development
- Network/Infrastructure
- Software/hardware support

I dont know other sectors, but I know 'Software development' now is poor. But actually, software is the heart (or soul) of computer. **If our goal is to push the IT industry to a mature level, software development can not be ignored.** Imagine all companies in HK are using locally

developed systems, compare with all using system bought from international firm, it is a big market !

Main problem of software development in HK
- For those already working :
    - **Unfair pay** : Programmer in HK is getting low salary compared with other countries. And they always working overtime. They are expected to be IT expert, know many things but still get low pay. They need to learn many new tech within days, but they dont get a fair reward for that.
    - **Work without glory** : as users will not know things beyond the face of the system, effort of programmer trends to be hidden. In contrast, when the engine have problem, users will complaint. In order words, always blame, but never praise. This is very frustrated to many programmers. As a result, most programmer simply dont care about the engine. Make users even more likely to blame them.
    - **No discipline, bad reputation** : IT industry (including people that is not writing codes) in HK is still in early state. Most of the programmers dont think themselves a professional. They use illegal software themselves and enjoy teaching others how to do so. Besides license issue, they are teaching others to hack/crack systems, they simply ignore the security issue and many things that need to care. To conclude it, they dont have discipline, they have no idea why dont do something.

- New force :
    - **Always hard to hire programmer** : I dont have the statistic on this. But in my experience, we always hard to hire programmer. And most interviewers have very limited skill set. Seems, many so-called IT worker are indeed working as operator, help desk and 'meeting machine'. Very small portion of us have hands on experience in coding.

- Low Quality systems :
    - **Hard to measure** : Quality of a computer system is hard to be measured, especially, for those outsider. As a result, people trends to sacrifice quality to save time (or cost).

    - **Low IT knowledge users** : Most users (the management level) have too low IT knowledge, therefore, they do not know how to use the system to help them.

    - **Too harsh deadline** : because users do not understand IT, the only think they know is 'how much it cost', as a result, they simply choose the quickest and cheapest one. This make the system quality even worse.

    - **User suffering** : after the development period, the system will still come out, but in poor quality. Many users will suffer, the system dont help them finish their jobs. Some flows are not handled, or even many bugs exists and still not getting fix. Sometimes, IT support will 'use the system' instead of the users themselves. This is obviously silly but it is a fact. For the managers, they think this solved the problem, but actually, this is another waste of productivity. If the system is built with well planing, this kind of waste can be avoided entirely.

    - **Lack of user involvement** : for a system development, users have to conduct a UAT to confirm the system is work as they expected. Because users are always too busy, they are always too late to involved in the development. They are the one who use the

system, but they do not providing any idea/comments until late stage of system development. In most of the cases, they found many problem during UAT. This is a nightmare but is often happening. As a result, programmers are asked to change the system greatly after the UAT is started. It is easy to imagine, the system will not be stable, and the design will look like a 'no design at all' system.

- **Dummy Technical Document** - in software development cycle, there are many technical documents to be wrote. But people think they are just homework, no one really look into it. But these documents, in fact, are very important. It can clearly define the system and avoid misunderstanding between users and IT.

The Good Side

- Create Demand : many companies in HK need computer systems to help them, including for daily operation work, analysis and financial aspect. Because we, HK people, always think with creativity, software developed by other countries often cant meet their needs. Overall, software needs in HK is great.

- Suitable Labor / potential labor : people in HK are smart, compared with other countries, especially India. To build great system, programmer and their skill are the keys. HK people, in this aspect, show a great potential, compared with other countries.

Suggestions

Next Objective

As IT industry is already not new to HK, we must focus on what is the problem and what should we achieve next.

- **Increase Quality**
  - **Teach them how to measure**
    To increase system quality, the key is to let the users know what is a good system and what is bad. Actually, many IT people know what is good when they are learning in University. But because no one are measuring on their codes, they start ignoring these standards. I highlight some of them here:
    - Donot hardcode
    - Write logs
    - Write code with good readibiliy
    - Test with test case
    - Code must be reviewed by others
    (Actually, all of the above can be easily measured by some audit)

  - **Narrow the gap between users and IT**
    - **Better Document**
      Document should be wrote in more objective sentence because it is too clearly define the system. And too late revise is unacceptable, if user keep changing the Functional Spec, it is clearly a sign of bad planned project.

    - **Education on Adult / Managers**
      It is quite simple, just grep the manager to use the system.

    - **Better Testing**
      Test with plan, test cases, result should be well filed. Of cause, use objective sentence.

- **Funding for Good**
  After all, how to increase quality without corresponding funding? When asking IT / users to meet higher standard, you have to reward them correspondingly.
  I suggest to establish some funding to reward who join the scheme. For IT professional, if they meet the standard, they will be rewarded by the fund. But they must be audit on it. If they violated some IT discipline, the funding is reduced. On other hand, if their users violated some standard, for example, late revise on Functional Spec, the IT will be compensated with some extra funding. This is the same as the users side.

- **Repackage IT Professional**
  - **Advertisement on IT** : image of IT professional is too low, making new comer dont join. Some Ad or marketing work should be done on IT.

- **Encourage software house**
  - **Funding for purchases / network expense**
  - **Own their codes**
    Many company will buy the source code as well as the software. This make the software house hard to build up their 'library of codes'. Government should encourage software house to own their own codes. Then, they can use these codes for other customers, saving their cost and making their code more powerful. This will be good for both sides.

  - **Encourage usage/development of framework**
    Same as the previous point, software house should try to increase the rate of re-use of their work. Framework is what that is for. For model software development, usage of framework is a key to success. But because it is totally a hidden technical term for users, many IT professionals simply donot embrace this approach of writing codes.

  - **Encourage Open Source**
    IT professionals always hiding technical detail of the system from users. And many systems are unstable, even after many fixes. To solve this situation, IT world already have a good solution in long time ago, which is Open Source. Open Source is too make the codes visible to the public. Then, all hidden nasty codes are unhide. This can greatly increase the quality of codes. But the precondition is, the code must be released. For any individual, or company, it is hard to force this standard. But for government, rules or laws can help.
    By the way, there are many misunderstanding in Open Source. Many thinks this will introduce security issue, indeed, most of the security standard is open standard. Many don't understand, in Open Source, how software house earns ? But actually, there are many software development company in the world that mainly sell their service based on Open-Source softwares.

- **Encourage provision of web service**
  Web service is providing a API for external parties to directly get information from the server. This information is formatted in standard format. Therefore, others can easily develop 3rd party software for the backend (who providing the web service). For example, casual developer can create mobile app to show weather information, bus route information. This can greatly reduce the load from the backend company. And

make the software quick available to the market. And quick delivery is very important for software.
By the way, I know HK Gov already have some website that is said to be providing web service. But after looking into it, I am very disappointed. It is not conformed with standard, and the available service set is incrediblily small.

Name: Liu Chun Yu