https://www.facebook.com/images/emoji.php/v6/f8c/1/16/1f4bb.pngWhat Software Developers Need to Know About DevOps

Software တခုကုိစျပီးေကာင္အထည္ေဖာ္ျပီဆုိရင္ develop လုပ္တဲ့အခ်ိန္ ကေနစျပီး release လုပ္တဲ့အထိဟာ အျမဲလုိလုိ code ခ်ည္းပဲေရးေနရတာ ထက္ပုိပါတယ္။

မ်ားေသာအားျဖင့္ software ေတြဟာ data ေတြသိမ္းမဲ့ data storage ေတြ၊ build လုပ္ဖုိ့ေတြ၊ testing လုပ္ဖုိ့ေတြ၊ တေနရာရာမွာ deployed လုပ္ဖုိ့ေတြလုိပါမယ္။

က်ြန္ေတာ္တုိ့ေတြဟာ အဲ့တာေတြကုိ ကုိင္တြယ္ျပီး ေျဖရွင္းနုိင္မယ့္ လူေတြကိုလုိအပ္ေနတာ ျဖစ္တယ္။

အဲ့တာေၾကာင့္မုိ့လုိ့ DBAs (database administrator) ေတြ DevOps roles ေတြရွိလာတာပါ။ (အခုတေခါက္ေတာ့ DevOps နဲ့ပက္သက္တာပဲေရးသြားပါမယ္။)

ကုိယ့္အေနနက္အကုန္လုံးကုိ (see “What this mean to you” section:) သိစရာလုိလုိ.လား၊ code ခ်ည္းပဲေရးေနလုိ့မရဘူးလားလုိ့ေမးစရာ၇ွိေနပါလိမ့္မယ္။ Code မွ Code ပဲ Java မွ Java ပဲေရးခ်င္တဲ့လူေတြအမ်ားၾကီး၇ွိပါတယ္။

အေျဖကေတာ့ ရပါတယ္ code ခ်ည္းပဲေရးေနလုိ့ Java ခ်ည္းပဲေရးေနလုိ့၊ ဒါေပ့မယ့္ အခုေနာက္ပိုင္းနွစ္ေတြမွာ company ေတြ အဖြဲ့အစည္းေတြဟာ တစ္ခုတည္းကုိပဲလုပ္ရတယ့္ပုံစံမ်ူိးမဟုတ္တဲ့ဘက္ကုိသြားေနၾကပါျပီ။ ဆိုလိုတာက ကုိယ့္ေနနဲ့ language မ်ိူးစုံလုပ္ရင္လုပ္ရမယ္၊ software တစ္ခုအားလံုးျပီးသြားျပီး delivery ပို့ျပီး production ဘက္ကုိေရာက္သြားတဲ့အထိ လိုတဲ့ေနရာတုိင္းမွာ၀င္ျပီးလုပ္ရင္လုပ္ရမယ္။ အဲ့တာေၾကာင့္ coding ခ်ည္းပဲေရးမယ္ဆုိတဲ့လူမ်ိူးမဟုတ္တဲ့ software developers ေတြပုိျပီးလုိလာတာပါ၊ အနည္းဆံုးေတာ့ ေနရာတုိင္းကုိ၀င္ျပီးျဖည့္ေပးနုိင္မယ့္လူမ်ိုးေတြပါ။

ဘလုိပဲျဖစ္ျဖစ္ software developer တစ္ေရာက္အေနနက္ ဘယ္လုိ roles မ်ိူးမဆုိလုပ္ဖုိ့ေမးခံရနုိင္တယ္၊ အထူးသျဖင့္ DBAs or operation team လုိမ်ိူးသက္သက္မ၇ွိတက္ေနရာမ်ိူးမွာဆုိရင္ေပါ့။ ကုိယ့္အေနနက္ application code တခုလုံးရဲ့ process ကုိကုိင္တြယ္ဖုိ့၊ build လုပ္ဖုိ့၊ testing လုပ့္ဖုိ့၊ stagin-server ေတြ production-server ေတြ ထ္ိန္းခ်ူပ္ဖုိ့ေတြေမးခံရနုိင္တယ္။

အဲ့တာေၾကာင့္ ဘလုိပဲျဖစ္ျဖစ္ ကုိယ္နဲ့သက္ဆုိင္ေနတဲ့ technologies ေတြအကုန္ေလ့လာနုိင္ရင္ပုိေကာင္းပါတယ္။

►DevOps: A New Role

IT ေလာကမွာ role အသစ္အေနနက္ထက္ထြက္တာရွားသလို၊ ရွိျပီးသား roles ေတြနက္ပဲလည္ပတ္ေနတာမ်ားပါတယ္။

DevOps ရဲ့ role ကေတာ့တျခား role ေတြနက္မတူသလုိ၊ က်ယ္လည္းက်ယ္ျပန့္တယ္၊ သူ့ကုိေသေသျခာျခာသက္မွက္ထားတာမရွိဘူးလုိ့ေတာင္ေျပာလုိ့ရတယ္။

★ဒီလုိဆို DevOps ရဲ့ အတိအက်ကဘာျဖစ္မလည္း?

အတိအက်ကေတာ့ software တစ္ခု၇ဲ့ development and operations အစအဆုံးမွာလုပ္ရမယ့္အရာေတြအကုန္လံုးကုိေျပာတာပါ။

တကယ္လုိ့အဲ့တာကုိတျခားလူေတြဆီမွာေမးၾကည့္မယ္ဆုိရင္ အေျဖအမ်ိူးမ်ိုးရနုိင္ပါတယ္၊ ဒါေပမယ့္ DevOps နဲ့ပက္သက္ျပီးအားလုံးလက္ခံထားတယ့္ေျဖက code build ေတြ၊ testing ေတြ၊ deployed ေတြနက္၊ production မွာ applied လုပ္ဖို့ေတြလုိအပ္တဲ့အရာေတြအားလုံးအတြက္ လုပ္ေပးရတာပဲျဖစ္တယ္။

►Operations: How We Used to Do It

DevOps မတုိင္ခင္က်ြန္ေတာ္တုိ့ဆီမွာရွိတာက developers ေတြ၊ operations ေတြ (တခါတေလ IT လုိ့လည္းေခါာ တယ္)။

အရင္တုန္းကနက္ လက္ရွိ(အခ်ိူ့) က်ြန္ေတာ္တုိ့ေတြဘယ္လုိလုပ္ၾကတာလည္းဆုိေတာ့၊ developers ေတြက code ေတြေရးမယ္ QA ေတြပုိ့မယ္၊ QA team or QA ပုိ့ရမယ့္လူဆီ။

QA ကၾကည့္ျပီး အဆင္ေျပတာေတြ မေျပတာေတြျပန္ေျပာမယ္။

Developers ေတြကျပန္ျပင္မယ္ ကုိယ့္ဘာ့ကုိ testing ျပန္လုပ္မယ္၊ ၀ါ operation team ကေန testing လုပ္မယ္ production ကုိမပုိ့ခင္။

Operation team က complain ေတြတက္မယ္ coding standard ေတြ၊ server control ေတြ၊ security issue ေတြ ေျပာျပီး developers ေတြွဆီျပန္ပုိ့မယ္။

Developers ေတြကျပန္ျပင္မယ္၊ operation လုပ္တဲ့လူေတြနဲ့အဆင္မေျပနည္းနည္းျဖစ္မယ္။

ေနာက္ဆုံးတာ၀န္ရွိတယ့္လူက final code ယူမယ္၊ server deployed လုပ္ျပီး production ကုိပုိ့မယ္။

တေယာက္နဲ့တေယာက္အဆင္မေျပမျဖစ္ပဲ၊ အေပါာကက်ြန္ေတာ္ေျပာတဲ့အတုိင္းလုပ္လာျပီး အားလုံးျပီးသြားေတာ့မွပုိ့လုိက္တဲ့အရာက production ဘက္မွာအဆင္ေျပေျပျဖစ္သြားဖုိ့ဆုိတာရွားပါတယ္။

အဲ့ေတာ့ဒီအပုိဒ္မွာေျပာခ်င္တဲ့အေရးၾကီးဆုံးခ်က္က က်ြန္ေတာ္တုိ့ေတြဟာ software တစ္ခုအစအဆုံးလုပ္ရမယ့္အရာေတြကုိ တစ္ေယာက္နဲ့တစ္ေယာက္၊ တစ္ဖြဲ့နဲတစ္ဖြဲ့ ဆက္သြယ္မူွေကာင္းေကာင္းမရွိပဲလုပ္ျဖစ္သြားၾကတယ္ဆုိတာပါပဲ။

အဲ့တာေတြေၾကာင့္ေနာက္ပုိင္းေတာ္ေတာ္မ်ားမ်ား agile software development ဘက္ကုိေျပာင္းလာၾကပါပီ။ (အခြင့္ေရးရရင္အေျခခံေလာက္ေတာ့ဆက္ေရးပါမယ္)

အရင္ကလုိ ၆လေနမွတခါ ၁နွစ္ေနမွတခါ delivery လုပ္တာမ်ိူးမဟုတ္ပဲ၊ ၁လေနတခါ ၂လေနတခါမ်ိူးေတြ production ဘက္ကုိပုိ့ေနၾကပီ။

Operations ေတြ procedures ေတြအကုန္လုံးကုိတခါတည္းလုပ္တာမ်ိူးမဟုတ္ပဲ agile နည္းနဲ့တက္နုိင္သေလာက္လုပ္ေနၾကပီ။

►What Is DevOps?

အိုေက, အရင္ကလုပ္ေနတဲ့ေၾကာင္းေတြနည္းနည္းသိသြားျပီဆုိေတာ့၊ DevOps ကဘလုိမ်ိူးလည္း?။

“DevOps is about taking full ownership at every step along the way”.

ေစာနက်ြန္ေတာ္ေျပာတဲ့အကုန္လုံးကုိလုံး၀တာ၀န္ယူေပးရမယ္ဆုိတာမ်ိူးပါ။

Company ေတြမွာ DevOps roles ေတြသက္သက္ရွိလာပီ developer ေတြပဲ ဒါေပမယ့္ build ေတြ၊ testing ေတြ၊ automated test ေတြ၊ deployed ေတြ အကုန္လုံးကိုအတက္နုိင္ဆုံးလုပ္နုိင္မယ့္လူေတြရွိလာပီ။

အခ်ူိ့ company ေတြမွာမတူတဲ့ role ေတြသက္သက္ရွိတုန္းပဲ၊ developer ေတြ၊ software engineer ေတြ၊ tester ေတြ၊ ဒါေပမယ့္ သူတုိ့ေတြအကုန္လုံးကအတူတူပူေပါင္းျပီး DevOps role ကုိျဖည့္ေနၾကပီ။

DevOps ရဲ့အေရးၾကီးဆုံး၊ အေရးၾကီးဆုံး အခ်က္ကေတာ့ software တစ္ခုကုိ production ဆီေရာက္ေအာင္ပို့ဖို့အတြက္အေကာင္းဆုံးျဖစ္ေအာင္စဥ္းစားျပီးကုိင္တြယ္တာပဲ။

►What This Means to You

တကယ္ေတာ့ နည္းပညာအရၾကည့္မယ္ဆုိရင္ software developer တစ္ေရာက္အေနနက္ သင္ဟာ DevOps ရဲ့အစိတ္အပုိင္းတခုပါပဲ။ Welcome. Congratulations! ☺

ယေန့မွာ software developers ေတြဟာ code ခ်ည္းပဲေရးေနရတာထက္ကုိပုိျပီးသိဖို့လုပ္ေနၾကပါျပီ။

Code မွ code သမားေတြေတာ့ရွိပါေသးတယ္ (အၾကံေပးခ်င္တာကအဲ့တာသိပ္မေကာင္းပါဘူး)။

ဆုိလုိတာက ကုိယ္က small company မွာျဖစ္ျဖစ္ startup company မွာျဖစ္ျဖစ္ ကိုယ့္ရဲ့ code ကုိ deliver လုပ္ျပီး တျခားတေယာက္ရဲ့ျပသနာမ်ိူးျဖစ္ေအာင္မလုပ္သင့္ဘူး။ ကုိယ္ကုိတုိင္က development process ေတြ tools ေတြ တျခားလုပ္စရာရွိတဲ့အရာေတြအားလံုးကို ေသေသခ်ာခ်ာနားလည္ေနဖုိ့လုိတယ္။

★ကုိယ့္အေနနက္ေရာ software developer တေရာက္ေနနဲ့ေရာ သိထားရမွာေတြက

✔Source control ေတြ source integration ေတြအတြက္ဘယ္လုိလုပ္ရလည္းဆိုတာေတြ၊

✔Continuous integration ေတြ server ေတြအတြက္ဘယ္လုိလုပ္ရလည္းဆိုတာေတြ၊

✔Testing ေတြ integrated testing ေတြအတြက္ဘယ္လုိလုပ္ရလည္းဆိုတာေတြ၊

✔Development process ေတြနဲ့ configuration ေတြအတြက္ဘယ္လုိလုပ္ရလည္းဆိုတာေတြ၊

✔Monitoring ေတြ performance check ေတြအတြက္ဘယ္လုိလုပ္ရလည္းဆိုတာေတြ၊

လက္ခံခ်င္တာကအဲ့တာေတြအကုန္လုံးကအရမ္းမ်ားတယ္ဆိုတာပါ၊ ဒါေပမယ့္အကုန္လုံးကို in-depth knowledge သိစရာမလိုပါဘူး၊ နားလည္ျပီးေတာ့လုိအပ္တဲ့ေနာရာေတြမွာအသုံးခ်နုိင္ရင္ရပါတယ္။ (related: Specialization Vs Generalization: အခြင့္ေရးရရင္အဲ့ေၾကာင္းဆက္ေရးပါမယ္)

★ေအာက္ကဥပမာနဲ့စဥ္းစားၾကည့္ပါ၊

*ကုိယ္ကေဘာလုံးအသင္းတသင္း၇ဲ့ေ၇ွ့တန္းဆိုပါေတာ့၊ အဲ့ဒီမွာကိုယ္ကတျခား position ေတြအၾကာင္းသိစရာမလုိဘူးလုိ့ယူဆရင္ သိေအာင္လည္းမလုပ္ထားရင္ အဲ့ဒီ position ေတြရဲ့အဓိပယ္ရွိတဲ့ strategy ေတြအေၾကာင္းေဆြးေနြးတဲ့အခါ၀င္ပါနုိင္မွာမဟုတ္ဘူး။*

*Football Manager လုပ္ခုိင္းေနတာမဟုတ္ပါ တျခား role ေတြအကုုန္လုးံကုိသိေအာင္လုပ္ဖို့ပါ။*

Software development မွာလည္းဒီအတုိင္းပါပဲ။ Manager, CEO ျဖစ္ေအာင္အရင္ လုပ္ခုိင္းေနတာမဟုတ္ပါ၊ ကိုယ္လုပ္ရမယ့္ software နက္ပတ္သတ္တဲ့အရာအားလုံးကုိသိဖို့

လုပ္ရမွာျဖစ္တယ္။

►Conclusion

DevOps ဆိုတာ tool တခုလည္းမဟုတ္သလုိ technique တခုလည္းမဟုတ္ပါဘူး၊ culture change တခုပါ။ ေနာက္ပုိင္းနွစ္ေတြမွာလူေျပာမ်ားလာျပီး ေတာ္ေတာ္မ်ားမ်ားလည္း ေျပာင္းေနၾကပါပီ။ က်ယ္ျပန့္တဲ့ topic လည္းျဖစ္တဲ့အတြက္ ေလ့လာစရာေတြလည္းအမ်ားၾကီးက်န္ပါေသးတယ္ အတူတူဆက္ျပီးေလ့လာပါ။ ေနာက္ဆုံးေျပာခ်င္တာက တကယ္လုိ့သင္ဟာ DevOps ရဲ့အစိတ္အပိုင္းတခုမျဖစ္ေသးရင္ျဖစ္ေအာင္လုပ္ပါ ျဖစ္ျပီးသားဆုိရင္လည္းအစိတ္အပိုင္းေသးေသးေလးကေန အစိတ္အပိုင္း အၾကီးကေနပါ၀င္ပတ္နုိင္ေအာင္လုပ္ပါလို့ေၾကံေပးခ်င္ပါတယ္။

►Note

2016 မွာ programming နဲ့ပက္သက္ျပီး slide ေတြလည္းမေရးျဖစ္သလုိ FB မွာ status အေနနက္ေတာင္မတင္ျဖစ္ပါဘူး။ ☺

2017 မွာေတာ့ေရးသြားျဖစ္မယ္ထင္တယ္၊ အားေပးပါအုံး။

So, if you got something or agree with me about this post, please share. If you already share, please click the like button, let me know that you reached there.

Because I'm happy to know that people are reached my post. I will talk to you next time, take care!

☛***Reference: Self-Experience, Soft Skills Book***