



## Ke smlouvám v IT, resp. smlouvám týkajících se software

V praxi se setkávám, a to i v dnešní době, že zákazník, a mnohdy i dodavatel profesionál, při objednání, vývoji, implementaci, resp. koupi, software netuší, co vše se k dané problematice pojí. Smlouvy jsou řešeny přes obecné smlouvy o dílo, kde právní úprava a podpora IT oblasti schází a v případě nejasností, či sporů, dochází k nepříjemným situacím, které mohou být problematické pro obě strany v určité fázi vývoje, či domnělého dokončení. Tento text by měl, alespoň obecně, seznámit čtenáře s nejčastějším úskalím dané problematiky. Autor doporučuje jak dodavatelům software, tak objednatelům, nepodcenit tvorbu smluv a řešit ji s odborníky, resp. právními zástupci a advokáty.

1. Jak již jsem již zmínil, tak jednou z prvních obvyklých vad je typ smlouvy, kde se často vychází z konceptu „klasické“ smlouvy o dílo, která je ale určena spíše pro díla hmotná. Z hlediska např. vývoje software a implementace software je potřeba si uvědomit specifické potřeby kombinace více typů smluv, jako například smlouvu licenční, např. EULA (end user license agreement), smlouvu kupní, o dílo, NDA (Non-Disclosure Agreement - dohoda, v níž se člověk zavazuje, že získané informace použije jen na dohodnuté účely), atp., dle povahy dodávky software a případně i hardware, či smlouvy následné servisní. U tohoto bodu je dále třeba zmínit potřebu podchycení další práce s dílem, možné úpravy, záruky funkčnosti, možné uložení zdrojových kódů, přístup k datům pořizovatele, atp.
2. Jako spíše zajímavost mohu uvést kupříkladu jednu drobnost z praxe, kdy u vyplněných smluvních stran IT smlouvy, resp. podepisujících zastupujících osob, nebyly uvedeny jejich funkce ve firmě. Bylo poté dovozováno, že uvedenou smlouvu neuzavřely uvedené právnické subjekty, ale uvedené a podepsané fyzické osoby. Mohu tedy jen doporučit jasnou identifikaci všech zúčastněných stran.
3. Smlouva musí být jasná a srozumitelná. Spíše právní kolegové, při tvorbě smluv, se velmi často setkávají s potřebou jedné, či druhé strany smlouvy, aby některé články a ujednání nebyly zcela určité, či přímo „využili“ nejasností, či více významnosti některých ustanovení. Není neobvyklé, když se toto „otočí“ i proti svému tvůrci v případě sporu (v případě soudního sporu je při nejasném, resp. vyložitelném více výklady, použito v tom významu, který je výhodnější pro protistranu toho, kdo toto ustanovení navrhl). Důrazně tedy doporučuji jasnou smlouvu a formulaci pro obě strany a to s definicí užitých pojmů. Z pohledu své praxe mohu uvést, že i jako technici máme v IT velmi obtížné problémy se, v některých termínech a jejich definicích, dohodnout. Příkladem je například slovo implementace. Pokud ji jasně nedefinujeme, pak není jasné, co vlastně obsahuje a co je její součástí. Osobně v některých případech nešetřím, čas a naše lesy, v definici pojmů. Nejdůležitější přesnou definicí je pak hlavně předmět smlouvy.
4. Předmět smlouvy je v software na samostatnou kapitolu. Nicméně je potřeba si uvědomit, že v případě vývoje software, pokud nejsou jasně ohraničeny a popsány



- parametry software, resp. jeho funkcionalita, tak dodavatel na straně jedné bude špatně prokazovat řádnou dodávku v plném rozsahu a na straně druhé se bude objednatel bránit úhradě za stále nedokončený, či nepřesný předmět plnění. Rada v tomto spočívá v uvědomění si řádného postupu vývoje software. Software by měl být dodáván na základě analýzy prostředí a požadavků zákazníka. Následuje obvykle definice a podklady pro řízení projektu z pohledu fází realizace, načasování, definice projektového týmu, užití metodiky vývoje, atp., ale hlavně je specifikována funkcionalita software a postup nasazení, akceptace zákazníka, atd. Zmíněné podklady, hlavně zmíněná funkcionalita, by mohla být pokládána a uvedena za exaktní předmět smlouvy, resp. dodání software.
5. V okamžiku, kdy jsme si uvedli hrubý postup vývoje software, je dobré také zmínit připravenost objednatele. Zde mohu uvést obvyklou chybu, kdy objednatel si řádně nepřipraví přehled o svých potřebách a možných dopadech, rizicích, plynoucích z realizace. Z hlediska této absence poté vyplývá i nejasnost při specifikacích sankcí, jejich opodstatnění a výše. Sankce jsou obvykle důležité i z hlediska platnosti, právní účinnosti a právní vymahatelnosti smlouvy samotné. Z pohledu zákona je potřeba si uvědomit, jak se bude například postupovat v okamžiku odstoupení od smlouvy, nedodržení určitých podmínek, ujednání, apod.
  6. Zajímavým dalším bodem je odpovědnost za škodu, resp. smluvní pokuta. Škoda je obvykle těžce prokazatelná, kdy smluvní pokuta je vymahatelná téměř okamžitě. Z hlediska praxe je také zajímavé sledovat přístup k omezení výše škody, kdy ve většině případů se přistupuje k tomuto limitování tak, že toto není v našem právním řádu přípustné. Žádné řešení není zcela ideální, kdy východiskem může být definice předvídatelné výše škody, kdy se vracíme k předchozímu bodu připravenosti objednatele.
  7. V tomto místě bych rád zmínil velmi často se zaměřující odpovědnost za vady. (V předchozím článku „škoda vs. pokuta a jejich limitace“ se věnovala spíše ochraně dodavatele, kdy z pohledu zákona je výše škody neomezena a v případě například informačního systému může jít i o škody způsobené a vymahatelné z pohledu zákazníků objednatele. Může jít tedy o naprosto likvidační záležitost IT společnosti). V tomto případě jde ale o vady v software. Jedním z nejzákladnějších chyb je nespecifikování reakční doby a lhůt pro opravu, resp. odstranění vady. Každý čtenář si umí představit situace v informačním systému, či klíčovém softwarovém řešení, kdy dodavatel by prováděl reklamaci v zákonné lhůtě třiceti dnů. Doporučuji v počátku roztřídit možné problémy alespoň na kritické a standardní, se stanovenou reakční lhůtou dodavatele, jejich způsobem nahlášení a zaznamenáním (například ticket systém, apod.). Je možné doplnit i sankce za výpadky, zpoždění, apod. Dále je ale potřeba odlišit tyto možné vady na prostředí, což znamená např. na update, či změnách operačního systému, změny infrastruktury, možné „zavirování“ a jiné skutečnosti, které mohou způsobit vadu, ale bez zavinění a odpovědnosti dodavatele. Výše uvedené tedy naznačuje, že



v některých případech by dodavateli měla např. náležet odměna, za odstranění vady, namísto sankce.

8. Změny. Nic není jistějšího, nežli to, že v průběhu realizace bude docházet k různým změnám. Ať již z pohledu změn objednatele na účelu, resp. předmětu dodávky, prostředí a potřeb objednatele, tak i z pohledu dodavatele, jiných technických řešení, vylepšení, či jejich návrhy, praktická limitace technologií při realizaci, atp. Z tohoto důvodu je potřeba v prvotní smlouvě definovat jasně tzv. změnové řízení. Obvykle se každá změna v projektu dotýká předmětu smlouvy, termínů plnění, tak samozřejmě následné ceny. Doporučuji tedy jasně stanovit druhy a způsob komunikace, kompetence jednotlivých osob s návazností na postup akceptace dohodnutých změn, atp.
9. Dalším pohledem o potřebném obsahu smlouvy by mohl být i pohled na úspěšné zakončení, čili předání software objednateli. Jednoduše jde o předání a převzetí software. Zde bych doporučil postupovat cestou např. UAT (user acceptance testing), kdy dochází nejdříve k testovacímu provozu u objednatele, resp. zákazníka. Zde doporučuji předem stanovit klíčovou funkcionalitu, řekněme podružnou funkcionalitu a méně důležité součásti software. Jedná se zde o stanovení lhůt, možnosti slevy na vývoji a dodávce software, atp. Testovací provoz ukáže možné vady (z praxe prakticky vždy), je potřeba je odstranit a to v přiměřené lhůtě. Na rozdíl od klíčových a případně i podružných funkcí software, nevidím důvod proč neuzavřít převzetí software, resp. akceptační protokol, i v případě vad na méně důležitých částí software, s dohodou o odstranění. Na straně jedné jde o ochranu objednatele, že opravdu za své finance získá co objednal, na stranu druhou jde o ochranu IT dodavatele, aby mu nebylo prodlužováno dodání software a řešila se opravdu problematika důležitá.
10. Z pohledu ukončení a předání software je také třeba zmínit možnost, kdy obvykle spíše objednatel může při realizaci vývoje, či implementace, získat chuť smlouvu napadnout, obejít, resp. ukončit a řešit i jeho součinnost. Zde je potřeba zmínit doporučení kolegů právních zástupců, kteří radí tzv. taxativně vymezit důvody pro odstoupení od smlouvy a to i s lhůtou pro jejich možnou nápravu.
11. V této úvahové části bych rád zmínil fakt, že doporučuji se věnovat i problematice ukončení smlouvy a vyrovnání smluvních stran. Jako doposud jednou z nejschůdnějších cest se jeví akceptace jednotlivých fází projektu, resp. vývoje software a jejich průběžného účtování, resp. akceptace.
12. Poslední zmínkou by již snad mohla být jen navazující servisní smlouva. Ta na straně jedné zákazníkovi zaručuje chod software, na druhé straně IT zajistí další stabilní vytížení zdrojů, resp. fakturaci. Zde je dobré se věnovat opět systému hlášení, jejich třídění a doby reakce (servis), případné customizaci, „do“vývoji částí systému a rozšíření funkcionality s ohledem na zkušenosti s provozem, apod.