

Zkušenosti a rady z projektu Agonot

V následujících několika bodech sepíšeme rady, které se nám po implementaci a obhájení projektu Agonot zdají užitečné. Může se zdát, že některé rady jsou omílané pořád dokola i v sepsaných zkušenostech jiných projektů, ale je to zřejmě především z toho důvodu, že jsou velmi důležité a je třeba jim věnovat pozornost.

- Rozhodně je výhodné nepodcenit analýzu projektu. Je to stokrát omílaná věta, ale není účelem domluvit v 5 minutách téma projektu a 9 měsíců něco zběsile bušit do klávesnice. Důkladná analýza (klidně i několik měsíců dlouhá) strašně pomůže rozmyslet, co a jak bude uděláno, navrhnout důkladně architekturu, strukturu projektu a vyřešit (zatím nanečisto) nalezené problémy. Následně je implementace o mnoho jednodušší a věci, které byste 3x předělávali, uděláte nyní hned a pořádně. Analýze Agonotu bylo věnováno opravdu mnoho a také to bylo oceněno vedoucím, oponentem a komisí. Je to důležité i pro celý tým, během analýzy se ujistíte, že opravdu všichni rozumí problematice a tomu, jak bude celá aplikace fungovat. Dá se tím předejít spoustě problémům, nedorozumění a přepisování již hotových částí (i když určitě ne 100%). Při analýze se snažte nic nepodcenit, i zdánlivě jednoduché věci mohou být ve vašem kontextu problematické (u Agonotu to byla například serializace).
- Udělejte si časový plán, který se budete snažit dodržet. K tomuto vás bude nejspíše částečně nutit i vedoucí, ale je dobré mít již rovnou od začátku představu, co vše bude potřeba udělat, kolik času to tak může zabrat a v jakém pořadí. Na začátku jen ve formátu určitých milestoneů s tím, že následujících pár období máte rozdělené detailně. Snažte se plán dodržet, abyste nemuseli dodělavat spoustu věcí na poslední chvíli (i když k tomu stejně dojde, jde o to, v jaké míře). V časovém plánu nezapomeňte na dokumentaci (ideálně tak měsíc). Také si tam dejte tak měsíc na řešení problémů, které nedokážete odhadnout na začátku. Určitě k problémům dojde a budete rádi za čas, který jste si tam dali jako rezervu.
- Ať už zvolíte jakékoli rozložení týmu, je důležité, aby existovala nějaká autorita, která v případě potřeby zakročí. Někjaký „project manager“, který je v roli kritika a "stále nespokojeného" člena týmu. Neustále popohání členy týmu k práci nebo kritizuje, co již bylo uděláno. Je to velmi složitá role a ten kdo se toho ujme, si musí uvědomit, že přestože členové týmu jsou (nejspíše) vaši kamarádi, je potřeba se k nim v určité chvíli chovat jako k podřízeným, kteří neplní svou práci. A stejně tak na druhou stranu každý člen týmu musí chápat, že i když je vedoucí týmu jeho kamarád, nemůže za něho pravidelně dělat a přepisovat práci. PM týmu by měl i chválit dobře odvedenou práci a dávat to najevo i ostatním členům.
- Je nejspíše nezbytné používat nějakou správu zdrojových kódů, aby bylo možné sledovat změny (mimo jiné i PM týmu, který tak může snadno kontrolovat průběh práce). Také je nutné mít systém na správu bugů a tasků (je dobré si domluvit, kdo je bude vytvářet, ideálně PM, který řídí průběh vývoje). Při vývoji Agonotu jsme používali TFS server, který umožňuje pohodlně oboje (včetně provázání commitů s tasky, takže je možné poměrně pohodlně vyhledávat).
- Doporučujeme testovat během vývoje. Drobné chyby se vám mohou během vývoje nakupit a není nic moc příjemného, když je začnete opravovat na poslední chvíli. Asi měsíc před

odevzdání jsme byli přesvědčeni, že máme hotovo. Začali jsme intenzivně testovat a nakonec jsme byli rádi, že jsme za měsíc stihli opravit (snad) všechny chyby. Především proto, že opravou jedné chyby se objevila jiná (například se nám měnilo pořadí prvků po deserializaci – v automatickém testu velmi snadno odhalitelné, my jsme si toho za celou dobu nevšimli, až kvůli přidání testu k jiné chybě jsme to odhalili).

- Po implementaci projektu určitě není hotovo - zbývá ještě dokumentace a ta je v případě projektů na MFF hodně důležitá. Vězte, že se detailně kontroluje i gramatika, formátování, věcnost a slohová korektnost. Je důležité, aby dokumentace byla napsána srozumitelně, a mnohdy velmi pomůže několik krátkých úvodních odstavců na začátku kapitoly, aby čtenář pochopil souvislosti. Ideální jsou různé krátké ukázky, které naznačí, co se snažíte slovy sdělit. V uživatelské dokumentaci je potřeba nejdříve rozepsat, co vlastně aplikace umí, aby člověk získal představu, k čemu se to vlastně dostal. Na druhou stranu v dokumentaci není třeba psát romány, snažte se psát stručné a lehce pochopitelné věty a vysvětlit vše, co je potřeba.
- Pokud se rozhodnete psát dokumentaci až nakonec, je dobré si psát bokem (např. do nějaké wiki), proč jste učinili daná rozhodnutí během vývoje projektu, i když Vám v danou chvíli přijdou jasná, abyste pak nemuseli dlouze vzpomínat proč to tak bylo. Navíc budete i dlouze vzpomínat na to, jaké rozhodnutí jste vůbec dělali.
- Jednou z výhod SW projektu je i to, že si můžete vyzkoušet nové technologie na větším projektu. Na druhou stranu, nepřeženejte to, spousta technologií a knihoven má i své černé stránky. Je dobré, aby každé technologii někdo z týmu rozuměl, někdo, kdo ji bude mít na starost, na koho se mohou případně ostatní obrátit (i kdyby to pak měl dodatečně zjišťovat).
- Poslední částí práce na SW projektu je obhajoba. Je formou prezentace, během které se ukazuje i aplikace v praxi. V řadě sepsaných zkušeností jsme se dočetli, že čas na prezentaci je 20 minut. My jsme dostali 15 minut, což už je opravdu hodně krátký čas, navíc bylo potřeba trochu improvizovat a připravené slidy projet o něco rychleji. Podle mého by určitě měl mluvit jen jeden člověk, ale v momentě kdy se ukazuje něco v aplikaci, tak je vhodnější, aby u počítače byl někdo další. Určitě by nepůsobilo dobře, kdyby mluvící člověk mluvil směrem do monitoru. 15 minut je opravdu krátká doba, když chcete aplikaci ukázat a říct i něco k tomu, jak je to naprogramované. Proto je vhodné si vyzkoušet předem to, co budete chtít říkat. Nahlas a se stopkami. Tím zjistíte, co vše musíte přeskočit a vyhnete se tím zbytečnému zabíhání do detailů (na které opravdu není čas, ačkoliv jsou třeba zajímavé).