

# Aplikační vývoj – Návrh a Implementace

Návrh a implementace jsou jedny z nejdůležitějších fází vývoje aplikačního software. Dobrý návrh by měl stavět na výstupech z analýzy a měl by především umět zodpovědět základní otázku: „Jak bude systém vyroben?“ Následná implementace pak respektuje nejen požadavky zákazníka, ale také návrh. Pouze komplexní a profesionální přístup k návrhu a implementaci zajistí, že vývoj, údržba a následný rozvoj požadovaného řešení bude skutečně efektivní.

Value proposition	Přínosy pro zákazníka	
<p>Mnoho firem se přesvědčilo, že podcenění fáze návrhu či dokonce implementace při vývoji informačního systému se rozhodně nevyplácí. Výsledkem bývá přečerpaný rozpočet a systém, který nesplňuje základní požadavky na funkcionalitu, je obtížně ovladatelný, chybový, nestabilní nebo nemá požadovaný výkon. Přemluvit uživatele, aby přešli na nový systém, pak není jednoduché a investované prostředky vycházejí nazmar. Jen použití kvalitní metodiky, seřazeného týmu zkušených odborníků, odpovídajících nástrojů a technologií při návrhu a implementaci umožňuje většinu z těchto rizik eliminovat a zajistit tak splnění projektových cílů ve stanoveném čase a při optimálním využití zdrojů.</p> <p>Adastra využívá ve všech etapách aplikačního vývoje vlastní metodiku APV.360, která je postavená na metodice IBM Rational Unified Process.</p>	<p>Kvalitní provedení návrhu a implementační fáze zajistí především:</p> <ul style="list-style-type: none"> <li>■ Zvolení nejvhodnější strategie implementace – za pomoci správného posouzení požadavků na systém je možné zvolit optimální implementační strategii, což šetří náklady a zkracuje dobu trvání projektu.</li> <li>■ Snížení rizika volby nevhodné technologie – Kvalitně a včas provedený umožní provést včasnou volbu vhodných technologií. Technologie tak skutečně budou podporovat požadovanou funkcionalitu a budou optimálně dimenzované potřebám zákazníka.</li> <li>■ Snížení rizika chybné interpretace požadavků – Adastra využívá už v prvotních fázích projektu prototypování. Možnost vyzkoušení základní funkcionality na prototypu (často využívané např. při specifikaci uživatelského rozhraní) nebo ověření proklamovaných vlastností zvolených technologií snižuje riziko pozdních a nákladných oprav nedostatků.</li> </ul>	<ul style="list-style-type: none"> <li>■ Snížení rizika nesplnění požadavků zákazníka - ve všech fázích aplikačního vývoje používá Adastra iterativní přístup (zakotvený v metodice Adastra APV.360). Není tak nutné provádět radikální a nákladné změny; dílčí výstupy jsou v průběhu vývoje průběžně konzultovány a schvalovány zákazníkem. Ten má tak během celého projektu podrobný přehled o stavu řešení a při finální dodávce nedochází k nepříjemným překvapením.</li> <li>■ Zajištění optimálního výkonu, stability a bezproblémového chodu nového informačního systému je jedním z klíčových úkolů oborů Návrh a Implementace.</li> </ul>
<b>Formy spolupráce</b>	<b>Proč tým Adastra</b>	
<p>Adastra vám nabízí následující formy spolupráce při vývoji aplikačního SW:</p> <ul style="list-style-type: none"> <li>■ Vývoj a dodání kompletního řešení.</li> <li>■ Team leasing celého týmu analytiků, programátorů, designerů a softwarových architektů nebo jejich částí k doplnění vaší interní kapacity.</li> <li>■ Expertní konzultace v oblasti návrhu architektury informačních systémů a jejich zapojení do IT architektury zákazníka</li> <li>■ Školení a workshopy</li> </ul>	<p><b>Obecně:</b></p> <ul style="list-style-type: none"> <li>■ Schopnost porozumět zákazníkovi.</li> <li>■ Pečlivost důslednost a zodpovědnost.</li> <li>■ Schopnost týmové práce a komunikace.</li> <li>■ Schopnost přizpůsobit se pracovní zátěži.</li> <li>■ Spolehlivost.</li> <li>■ Aktivní přístup k sebevzdělávání.</li> </ul>	<p><b>Technologicky:</b></p> <ul style="list-style-type: none"> <li>■ Znalost různých typů architektury SW.</li> <li>■ Znalost metodik vývoje softwaru (např. standardní metodiky Adastra APV.360 a RUP).</li> <li>■ Orientace v oblasti návrhu a implementace software na platformě Microsoft .NET a Java. Zkušenosti a znalosti v oblasti databázových systémů</li> <li>■ Zkušenosti s dodávkou komplexních mission critical systémů.</li> </ul>

## Obor Návrh

Základem činnosti oboru Návrh je transformace specifikace předmětu dodávky (funkční a jiné požadavky získané oborem Analýza) na popis, jak bude systém vyroben. Tento popis zahrnuje zejména:

- definici architektury systému
- popis integrace systému s okolními spolupracujícími systémy
- dekompozici systému na moduly a komponenty a jejich detailní návrh.
- definici použitých technologií
- definici potřebné infrastruktury (hardware, software)
- návrh uživatelského rozhraní.

Návrh se v projektu vývoje IS objevuje již v počátečních fázích při definici možných architektur, návrhu prvotních prototypů a v dalších fázích je postupně rozpracováván a zpřesňován až po detailní návrh všech komponent systému. Metodika APV.360 dělí obor Návrh do čtyř základních procesů:

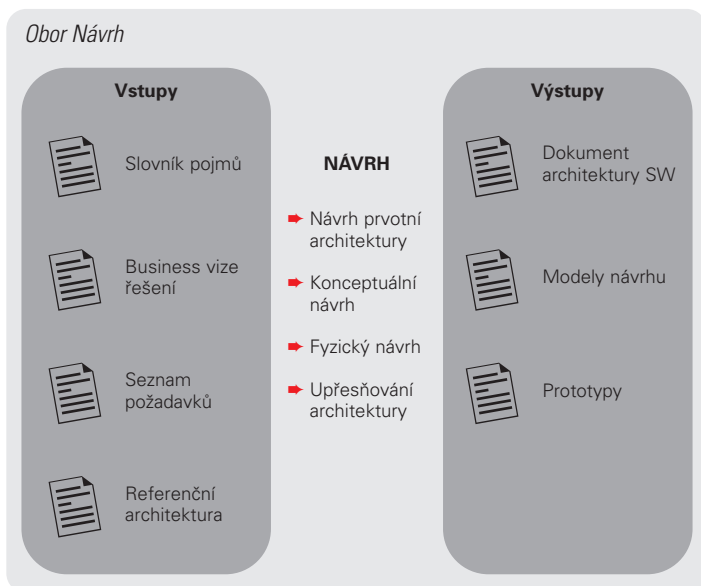
- **Návrh prvotní architektury** – vznikají první návrhy architektury systému.
- **Konceptuální návrh** – zaměřuje se na vytvoření první verze Design Modelu, kdy se bere v potaz především splnění všech funkčních požadavků a mohou být odstíněny ostatní požadavky (technologické požadavky, výkonové požadavky, implementační prostředí apod.).
- **Fyzický návrh** – vychází z konceptuálního návrhu systému, bere v potaz ostatní požadavky na systém (výkon, stabilita, robustnost apod.) a omezení systému (technologie, kontext prostředí nasazení systému).
- **Upřesňování architektury** – slouží k dalšímu rozpracovávání a upřesňování prvního návrhu architektury na základě dalších poznatků z analýzy, zkušeností získaných z prototypů a zpětné vazby od návrhářů, programátorů a testerů.

## Obor Implementace

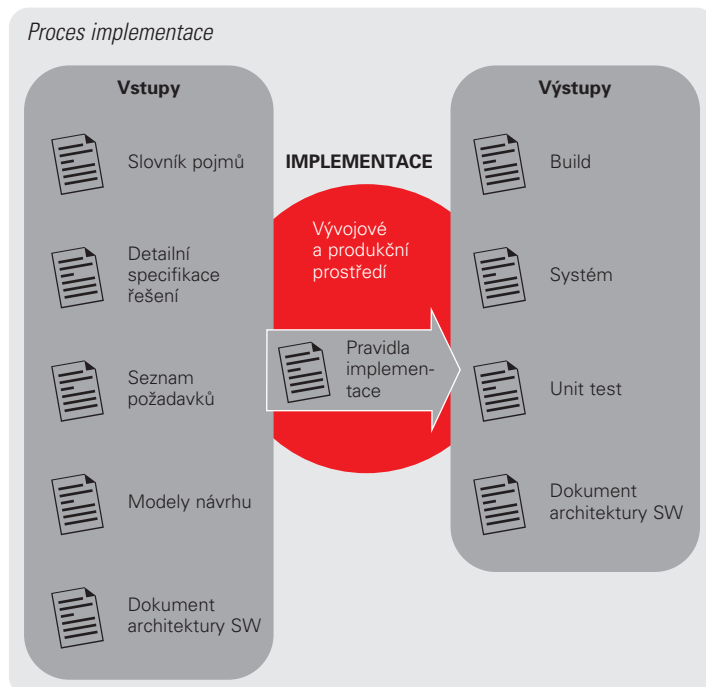
Úkolem oboru Implementace je vyrobit koncový produkt podle požadavků zákazníka při dodržení návrhu systému. Zahrnuje především následující činnosti:

- definice struktury vytvářeného kódu ve smyslu implementovaných vrstev a subsystémů
- implementace jednotlivých komponent (zdrojové soubory, binární soubory, apod.)
- otestování vytvořených komponent jako nezávislé jednotky (unit testování)
- integrace části systému vyvinutého jednotlivými vývojáři a týmy do celku.

### Obor Návrh



### Proces implementace



**ADA**STRA

ADA STRA, s.r.o.  
Nile House  
Karolinská 654/2  
186 00 Praha 8 – Karlín  
Tel.: +420 271 733 303  
sales@adastra.cz  
www.adastra.cz