

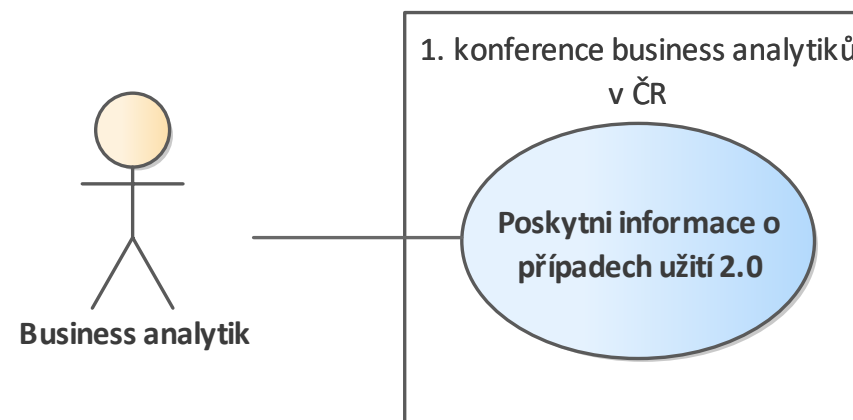
# Use Case 2.0: Případy užití v agilním prostředí



Slávek Rydval, Capricorn Pro

# Případ užití a případ užití 2.0

- **Případ užití:** všechny cesty, jak použít systém k dosažení konkrétního cíle konkrétním uživatelem.
- **Případ užití 2.0:** škálovatelná agilní technika
  - pro zachycování požadavků a
  - jejich řízení v inkrementální dodávce změny.



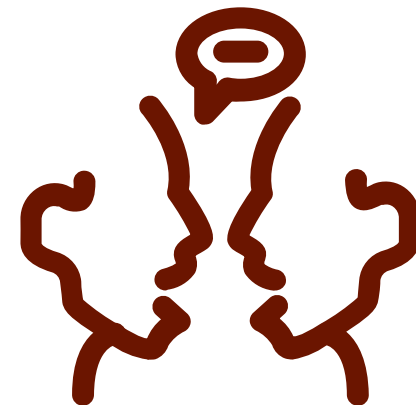
# Šest principů případů užití 2.0

1. Vyprávějte **jednoduché příběhy**.
2. Pochopte **celkový obraz**.
3. Zaměřujte se na **přínos**.
4. Budujte **postupně** (v částech).
5. Dodávejte **inkrementálně**.
6. Adaptujte se na **potřeby týmu**.



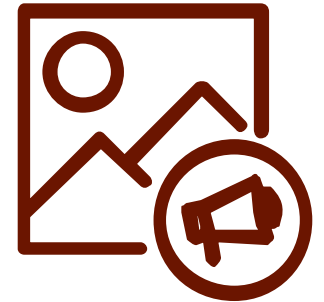
# 1. Vyprávějte jednoduché příběhy

- Příběhy jsou základem každé kultury.
- Je to nejjednodušší a nejefektivnější cesta k *předávání znalostí*.
- Pro pochopení případu užití také vyprávíme příběh.
  - Příběh pak můžeme převádět na podrobné požadavky.
- Pozor při *zaznamenávání* příběhů: je třeba, aby podle nich bylo možné postup *zopakovat a otestovat*.



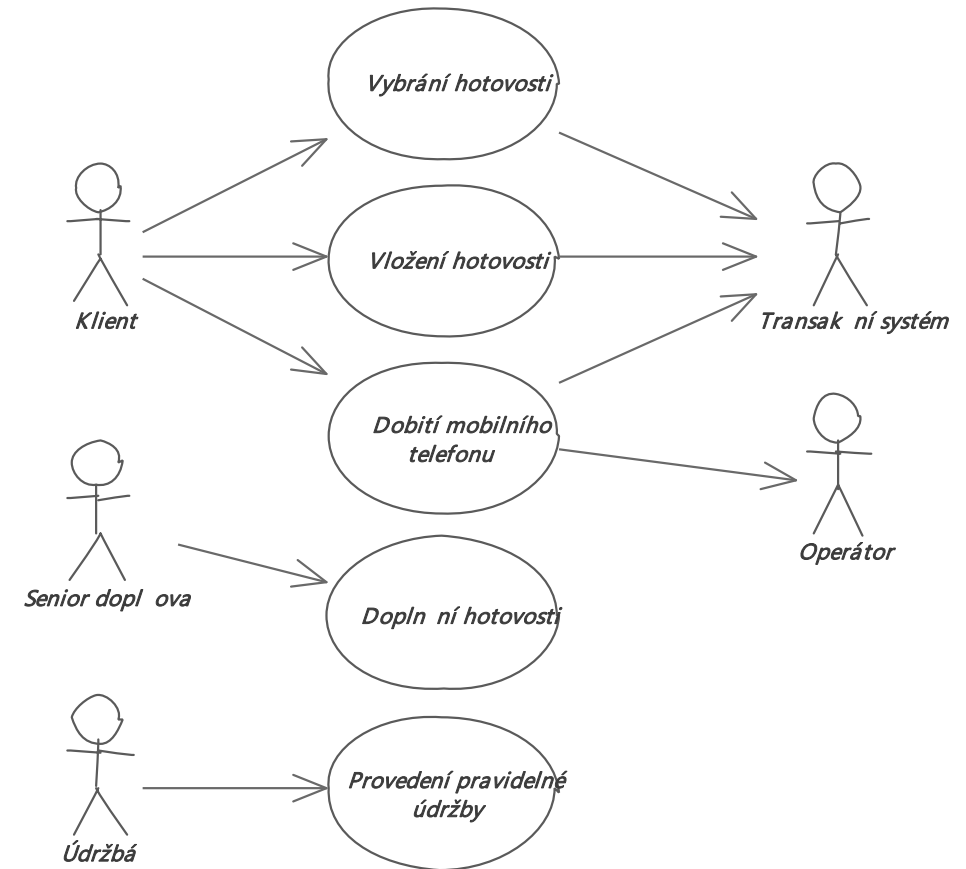
## 2. Pochopte celkový obraz

- Je třeba **chápat celkový obraz** daného řešení, a to bez ohledu na:
  - jeho velikost,
  - to, zda jde o softwarové, hardwarové či jiné řešení.
- **Bez znalosti celku je nemožné** dělat správná rozhodnutí typu:
  - Co je ještě je a co už není součástí celku.
  - Co je nutné zvážit a čím se nezabývat.
  - Kolik peněz bude řešení stát.
  - Jaké výhody systém přinese.
  - Jaké požadavky jsou správné.



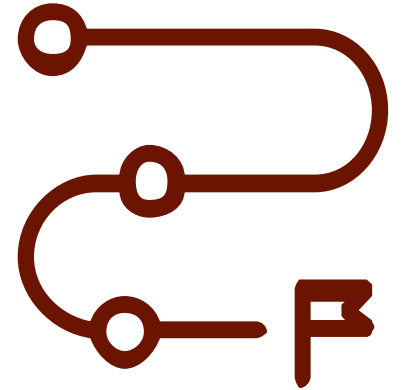
## 2. Pochopíte celkový obraz

- **Diagram případů** užití je jednoduchý způsob **prezentace přehledu o systému**.
  - To ale neznamená, že na jeden diagram dáte všechny informace!
  - Diagram je pouze **pohled do modelu**.
- Model případů užití představuje **přehled o poskytovaných službách** systému vůči aktérům.
  - Nesoutěžte však v počtu UC v modelu.
  - Jde o **konceptuální model**.



# 3. Zaměřujte se na přínos

- CO může systém **přinést** zvolenému **stakeholderovi**?
- Jaká je **nejrychlejší cesta k cíli**?
- Až poté:
  - Co se může cestou **pokazit**?
  - Je **více možností**, jak cíle dosáhnout?
- Určitě ne:
  - Co všechno systém může umět?



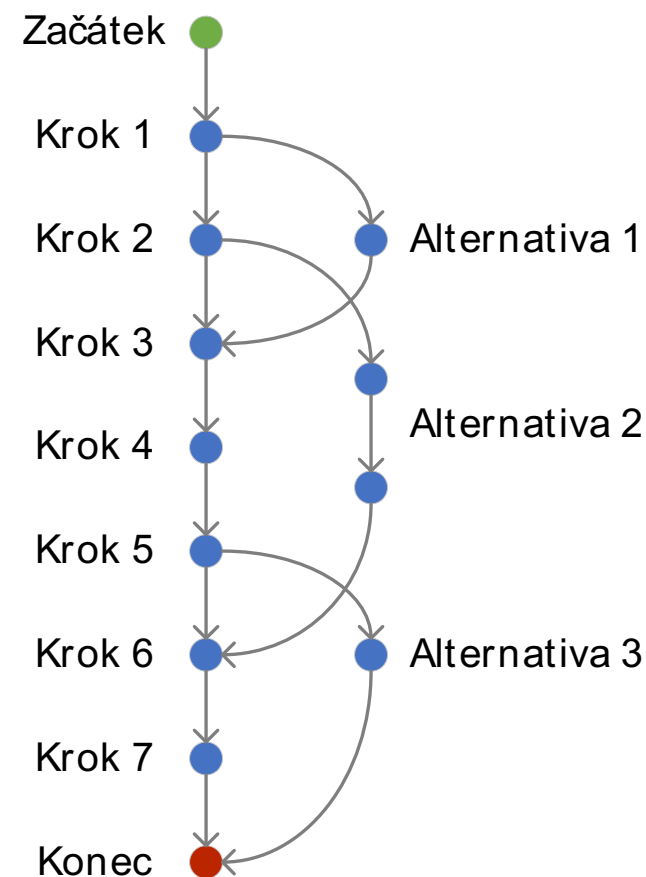
# 3. Zaměřujte se na přínos

## Základní scénář

- Uživatel vloží platební kartu.
- Systém validuje kartu.
- Uživatel zadá částku pro výběr.
- Systém vrátí kartu.
- Systém vydá peníze.

## Alternativy

- Nevalidní karta.
- Nestandardní částka k výběru.
- Požadavek na potvrzení výběru.
- Nedostatek hotovosti v bankomatu.
- Neodebrané peníze.
- ...





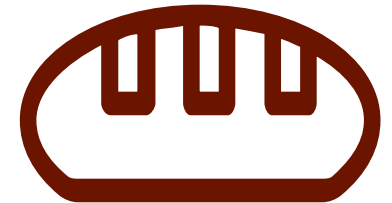
# 3. Zaměřujte se na přínos

- **Není třeba** najít a popsat **všechny** možnosti.
- Postupně:
  - Sestavte kroky základního scénáře.
  - Ptejte se, co se může v každém kroku stát dalšího.

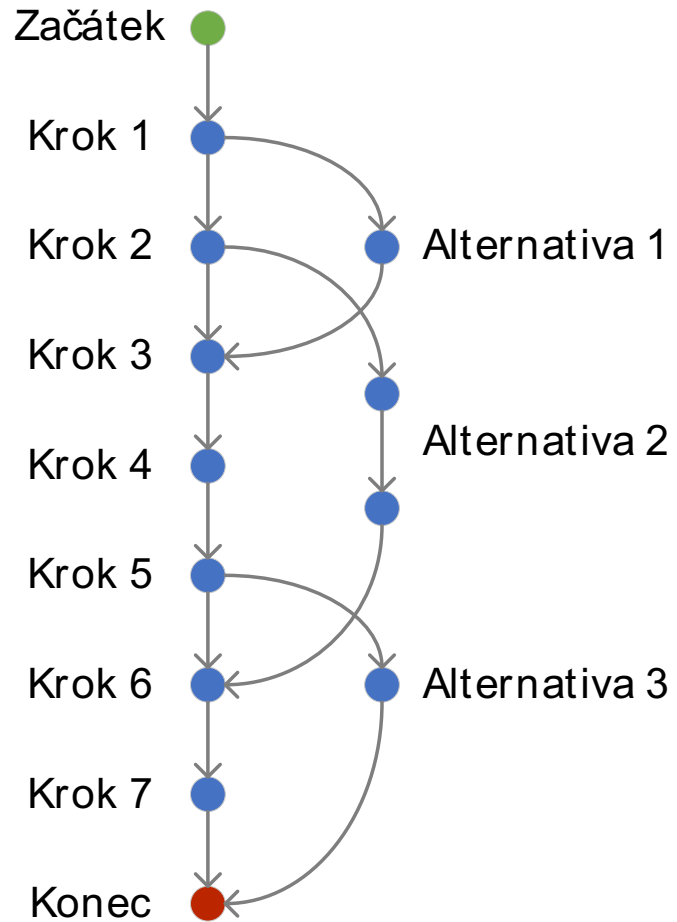


## 4. Budujte postupně (v částech, slices)

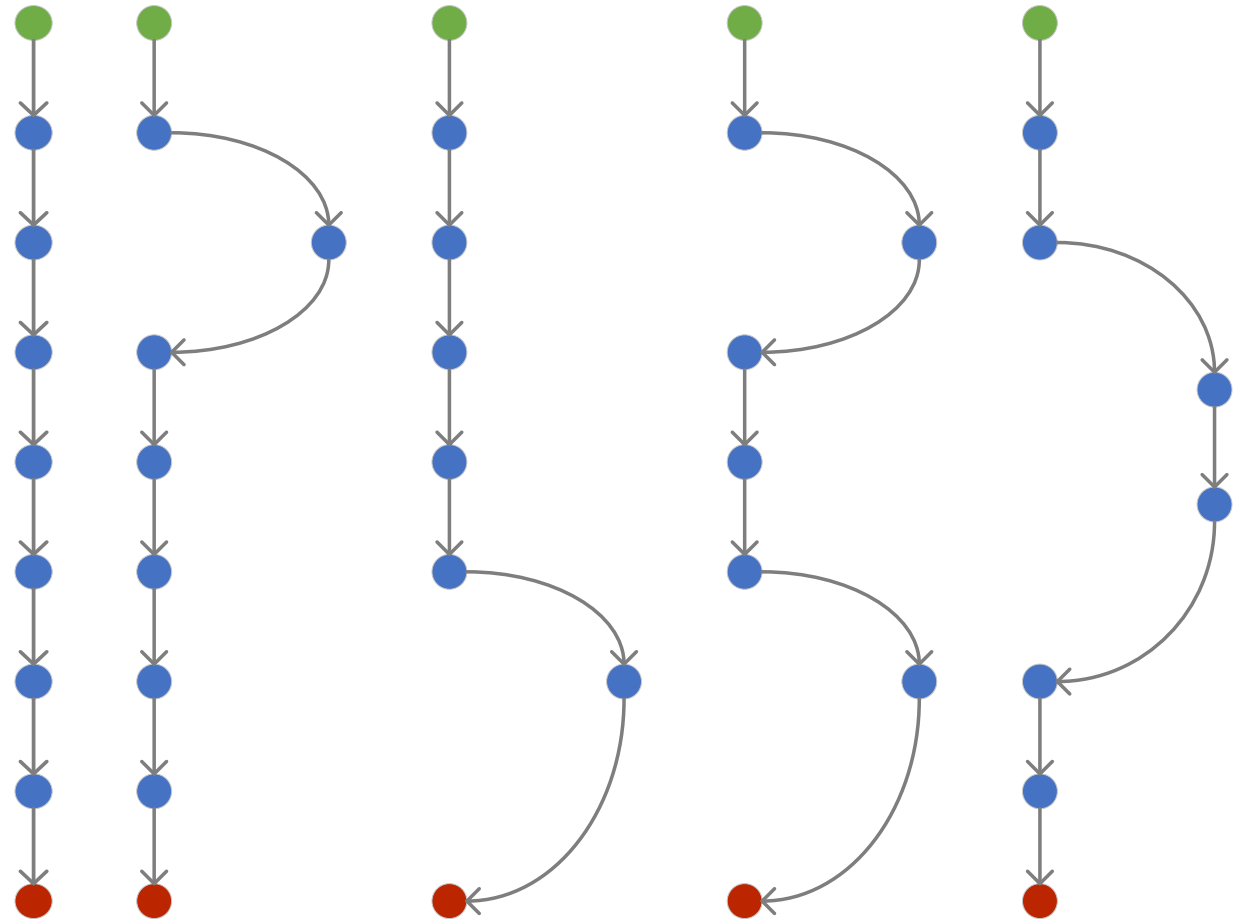
- **Nedělejte vše najednou**, ale postupně.
- Rozdělení na části je podle přínosu uživateli:
  - **Najděte jednotlivé případy užití.**
  - **Prioritizujte podle největšího přínosu.**
  - **Případy užití „rozkrájejte“** na jednotlivé části.
- Případ užití s alternativami **je možné procházet různě.**
  - Část (slice) je jeden takový průchod.
- Díky rozdělení na případu užití na části si automaticky **dělíte na části i způsob(y) řešení.**



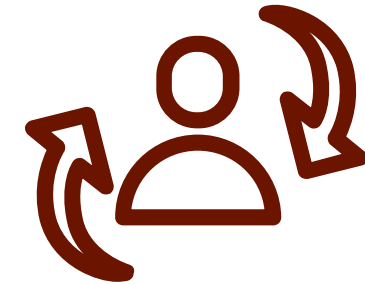
### Jeden případ užití



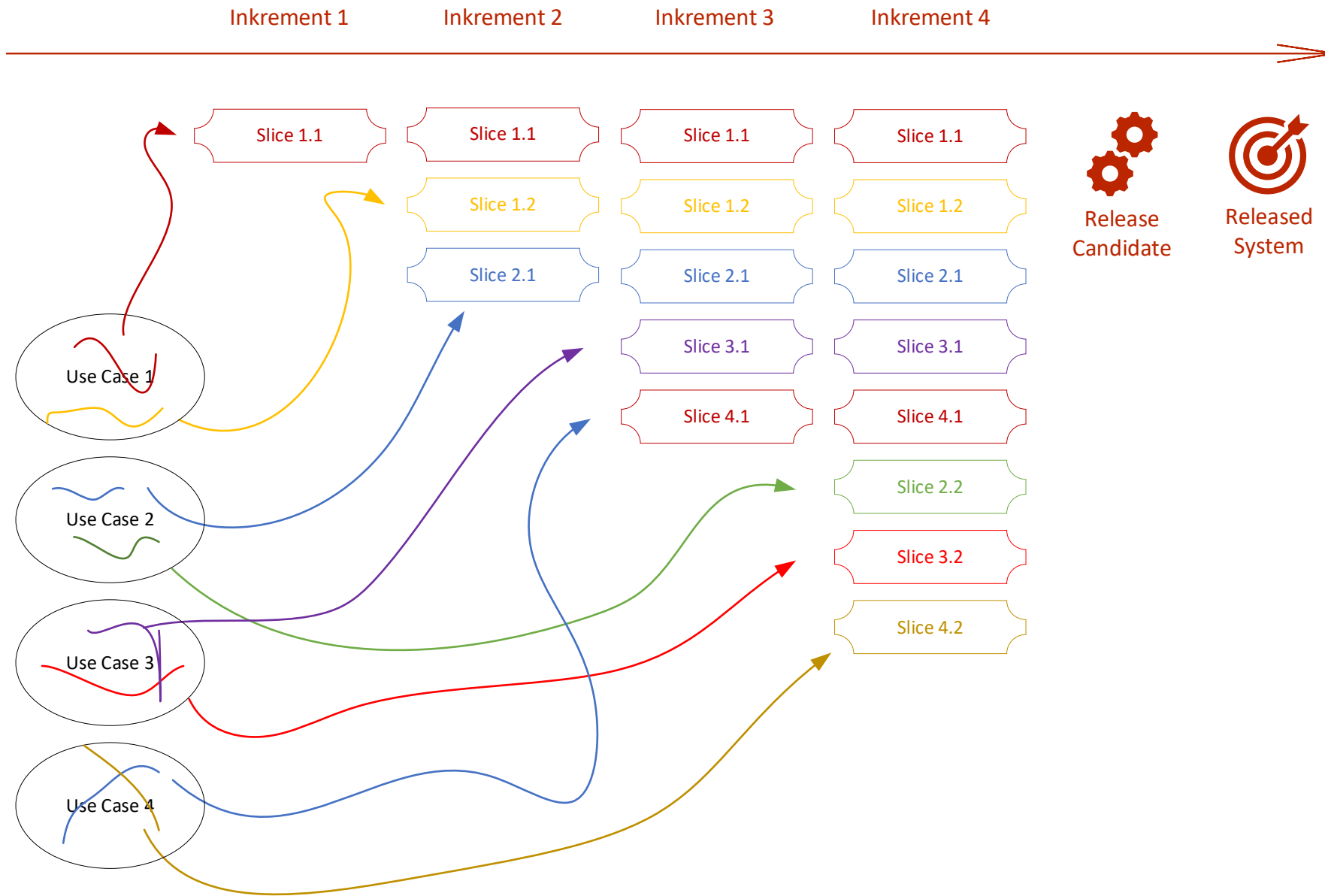
### Mnoho příběhů (stories)



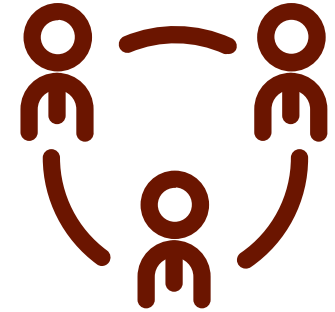
# 5. Dodávejte inkrementálně



- Podporujte postupné rozšiřování vlastností.
- Každý inkrement poskytuje **použitelnou verzi řešení**.
- Každý inkrement **staví na těch předchozích** (pochopitelně kromě toho prvního).
- Dodat celý případ užití v jednom inkrementu může být náročné: dodávejte po částech (slices).
  - Části dobře fungují i pro definování obsahu releaseu.

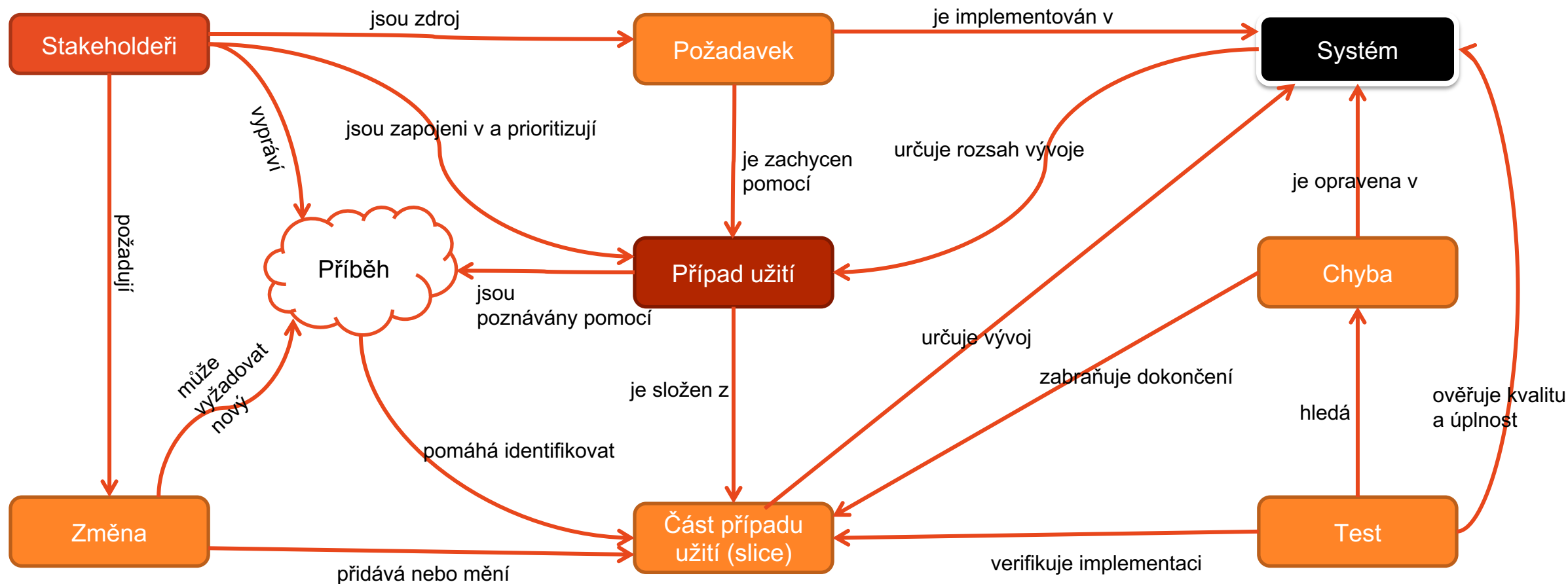


## 6. Adaptujte se na potřeby týmu



- **Kontext každé dodávky řešení je jiný:**
  - Lidé, nástroje, regulatorika, míra detailu...
- Adaptujte si (libovolnou) **techniku** tak, aby vám byla **užitečná**.
  - Např. odlehčené požadavky mohou fungovat při úzké spolupráci vývojového týmu s uživateli.
  - Případy a jejich části nemusí být plně modelované pomocí diagramů, ale (je-li to užitečné), stačí textový zápis.

# Dosah případu užití 2.0



# Use Case 2.0:

- je agilní technika,
- rozděluje UC na části,
- každá část má přínos pro uživatele.

**Slávek Rydval**

+420 603 404 318

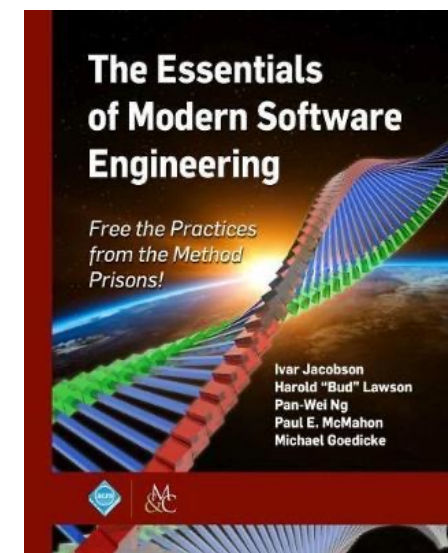
slavek.rydval@capricornpro.com





# Další zdroje informací

- Use-Case 2.0 ebook
  - <https://www.ivarjacobson.com/publications/white-papers/use-case-ebook>
- eLearning
  - <https://iji.ivarjacobson.com/>
- The Essentials of Modern Software Engineering
  - ISBN 978-1-947487-27-7



# Pořadatelé

**CAPRICORN PRO**



**Projectman**  
connecting experts

goodea >

# Partneři

**ČESKÁ**   
spořitelna

 metada