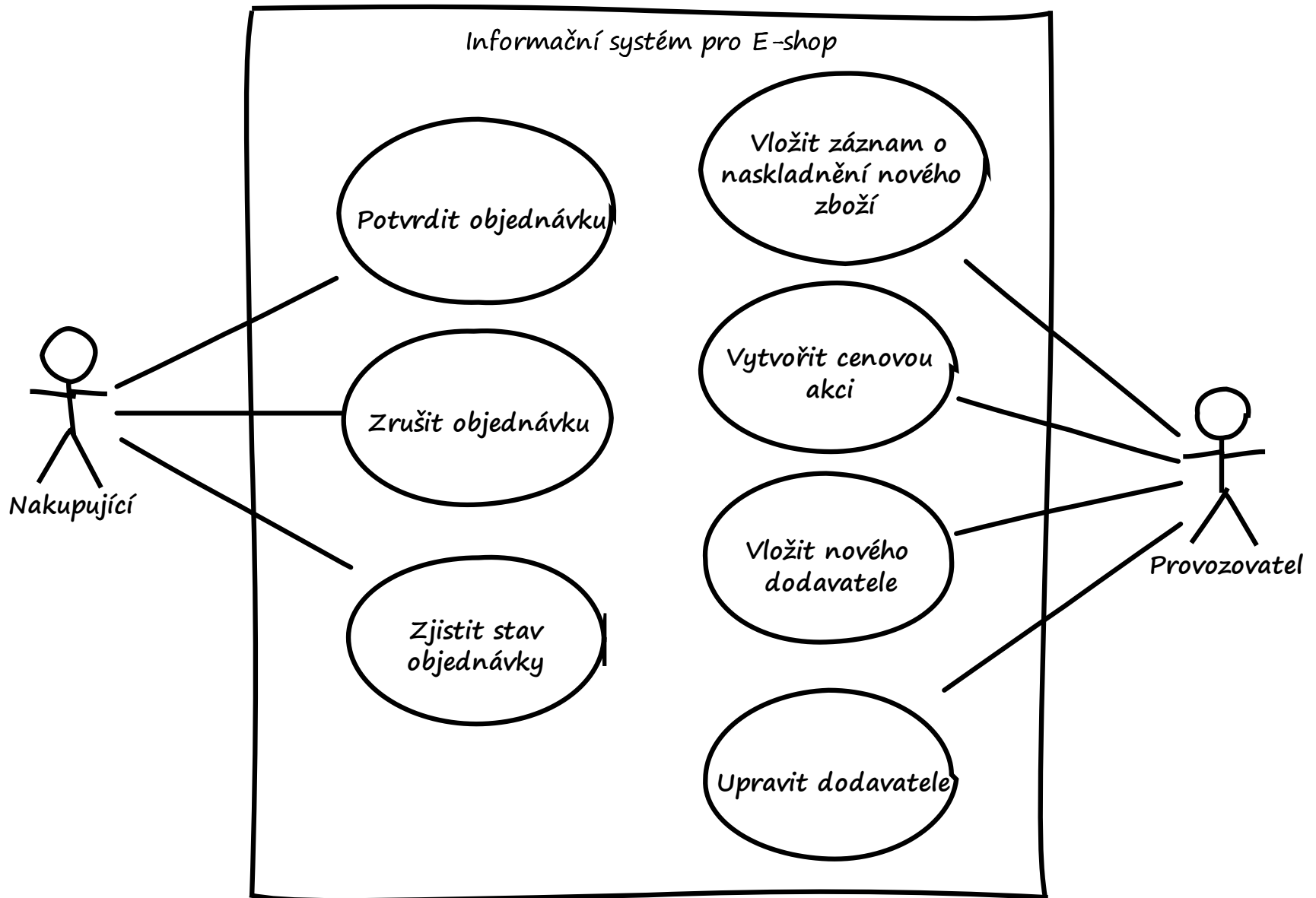


Model případu užití

Martin Komárek

Ukázka diagramu případů užití



Model případů užití = USE CASE MODEL

- Vyjadřuje, kdo bude jakým způsobem používat systém = uživatelé systému a jejich práva
- Cíl modelování: **Najít hranice systému**
 - Co je vně systému?
Aktéři = Kdo bude systém přímo používat/ovlivňovat?
 - Co je součástí systému?
Případy užití = Co bude systém umožňovat?
- Modelování případů užití, je způsob zachycení **pouze funkčních požadavků**. Pouze na základě UC modelu nelze správně odhadnout cenu vývoje SW.

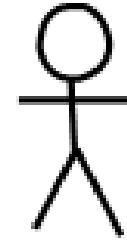
Proč vůbec modelovat případy užití?

- Může výrazně přispět k omezení efektu bobtnání řešení.
- Velmi vhodné pro zachycení detailu jednotlivých funkcionalit.
- Usnadňuje odhady pracnosti řešení.
- Podporuje efektivnější tvorbu uživatelské dokumentace.
- Vynikající podklady pro funkcionální testování.

Případy použití systému (USE CASE)

- Případ užití popisuje jednu **systemem podporovanou aktivitu** obvykle jednoho účastníka v jednu chvíli.
- Je vždy iniciován účastníkem.
- Pojmenovává funkce systému z pohledu účastníků. Tedy název odpovídá na otázku: „**Co SYSTÉM umožňuje uživateli?**“.
- Název má vždy **slovesnou vazbu!**
 - *Př. : Zadat objednávku, Zjistit stav objednávky, ...*
- Každý případ užití popisuje pouze jednu ze základních aktivit v systému (Vytvořit, zobrazit, upravit nebo smazat.)

Účastníci = aktéři = actors

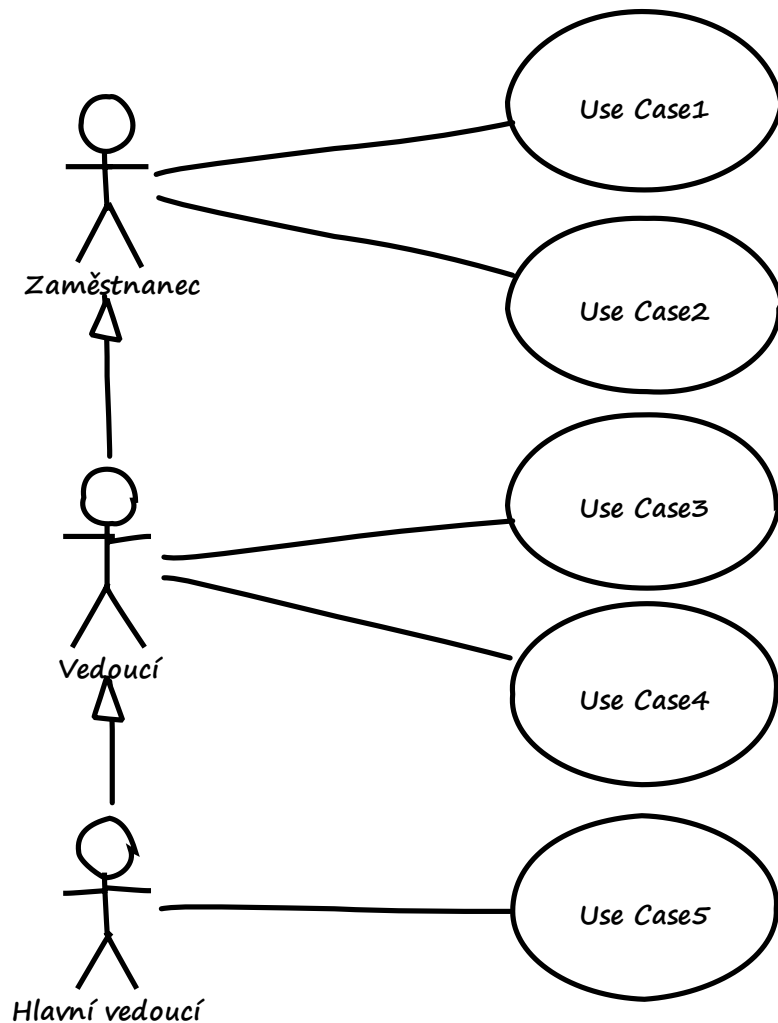
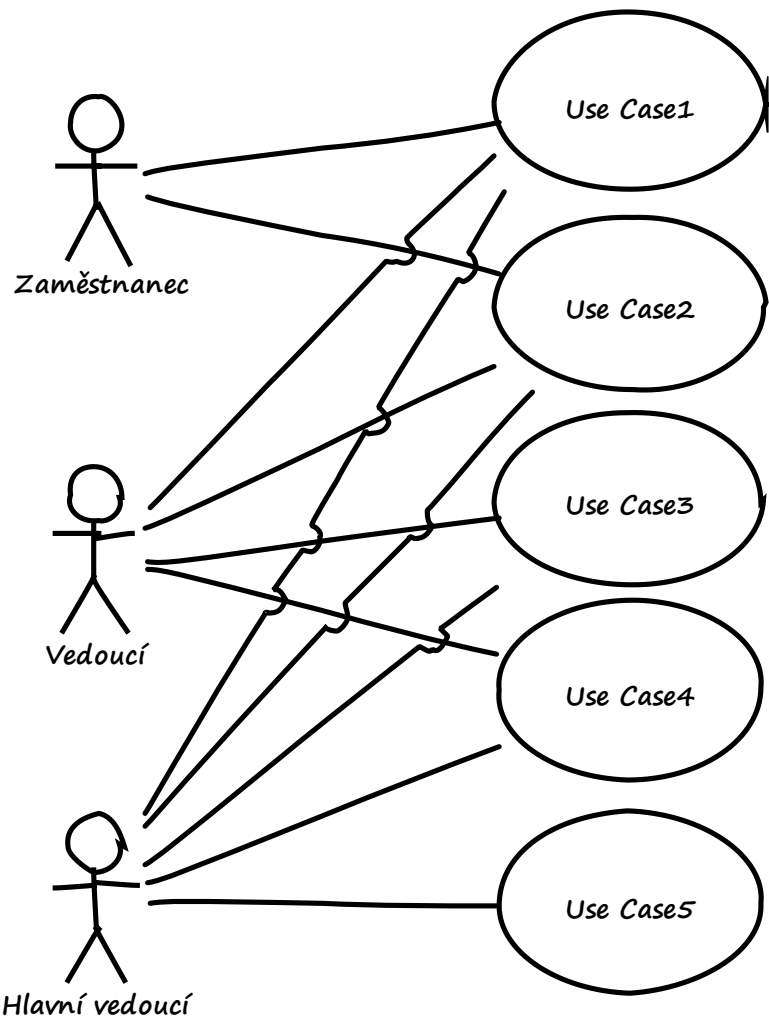


Jednotlivý účastník:

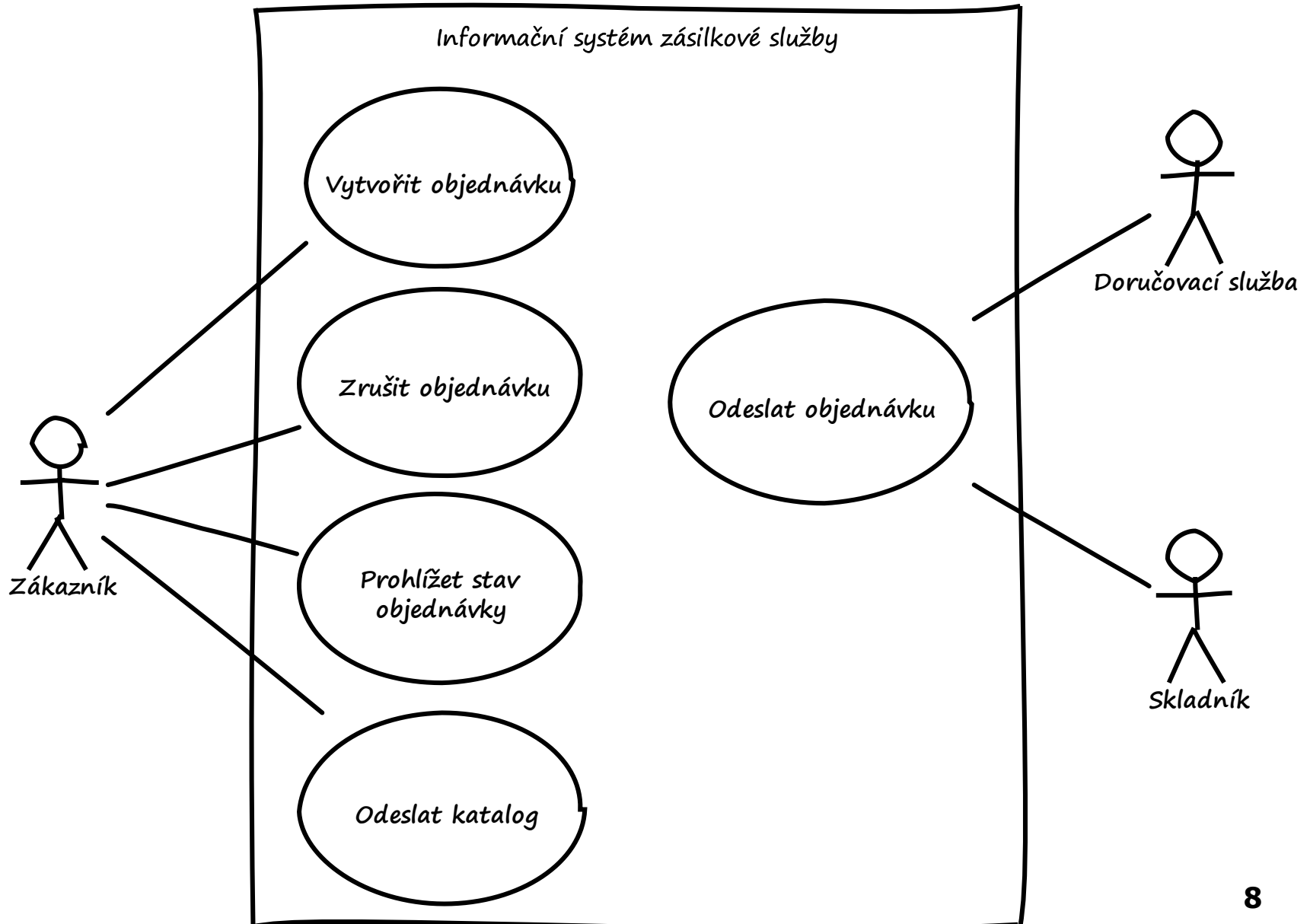
- je vůči systému externí entitou, která systém využívá nebo ho ovlivňuje.
- většinou účastníkem reálná osoba (uživatel), nebo častěji spíše **role**, kterých konkrétní osoby nabývají (př. obchodní zástupce, lékař, hlavní účetní,)
- účastníkem ale může být například i "čas" (spouštění záloh atd..)
- účastníkem dále může být i jiný systém

Generalizace účastníků

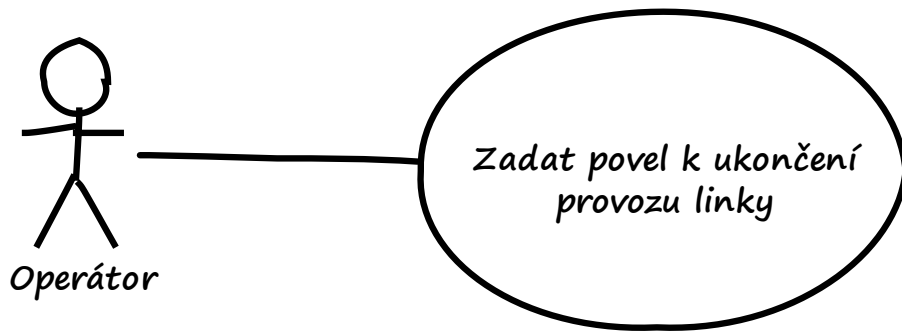
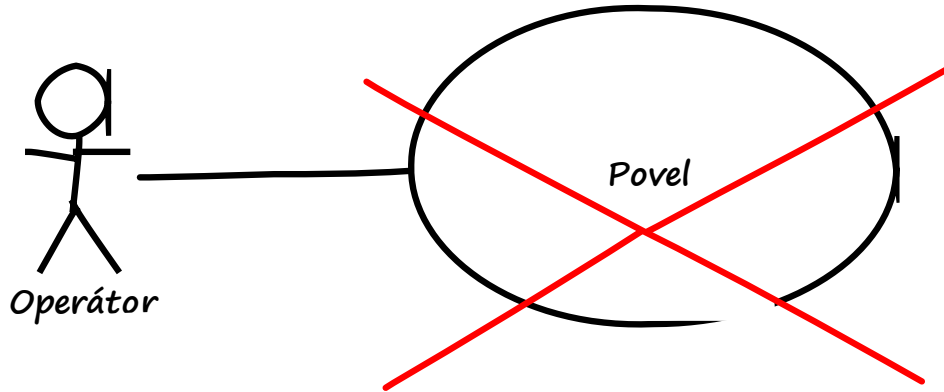
Slouží k zachycení hierarchie uživatelských práv



Co je špatně?



Název USE CASE vždy **slovesnou vazbou!**



Stačí název? Aneb detailnější specifikace případu užití

- Název
- Stručný popis = pár vět
- Detailní popis vnitřního běhu
 - Pouze pro složitější průchody
 - Varianty
 - Textové scénáře případu užití (nejčastější)
 - Digramy aktivit (pro hodně větvené scénáře)

Co se děje po spuštění případu užití? Scénář!

Popisuje interakci systému a uživatele!

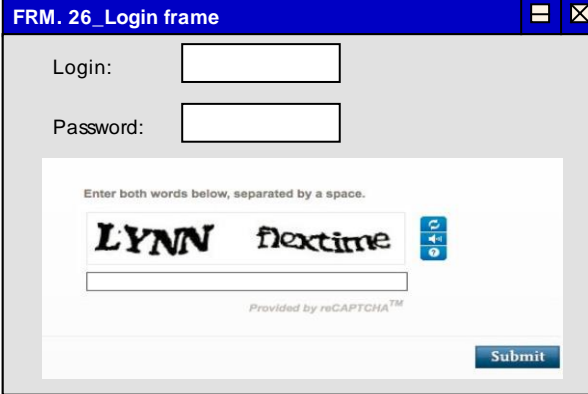
UC 112 – Přihlášení uživatele do systému

1. **System** zobrazí přihlašovací formulář (FRM. 26_Login frame) včetně CAPTCHA.
2. **Uživatel** vyplní požadované údaje a potvrdí je.
3. IF **System** ověří správnost vyplněných údajů THEN zobrazí všechny role (FRM.27_Role offer), které má uživatel v systému na výběr ELSE ...
4. **Uživatel** vybere jednu z nabízených rolí.
5. **System** ...

Textové scénáře je vhodné doplnit návrhy obrazovek

UC 112 – Přihlášení uživatele do systému

1. **System** zobrazí přihlašovací formulář (FRM. 26_Login frame) včetně CAPTCHA.



FRM. 26_Login frame

Login:

Password:

Enter both words below, separated by a space.

LYNN nextime

Provided by reCAPTCHA™

Submit

2. **Uživatel** vyplní požadované údaje a potvrdí je.
3. IF **System** ověří správnost vyplněných údajů THEN zobrazí všechny role (FRM.27_Role offer), které má uživatel v systému na výběr ELSE ...
4. **Uživatel** vybere jednu z nabízených rolí.
5. **System** ...



FRM.27_Role offer

Choose one of your role!

Admin

Teacher

Ph.D. student

Scénář případu užití = tok událostí

- **Hlavní scénář** (Main/Basic/Happy flow)

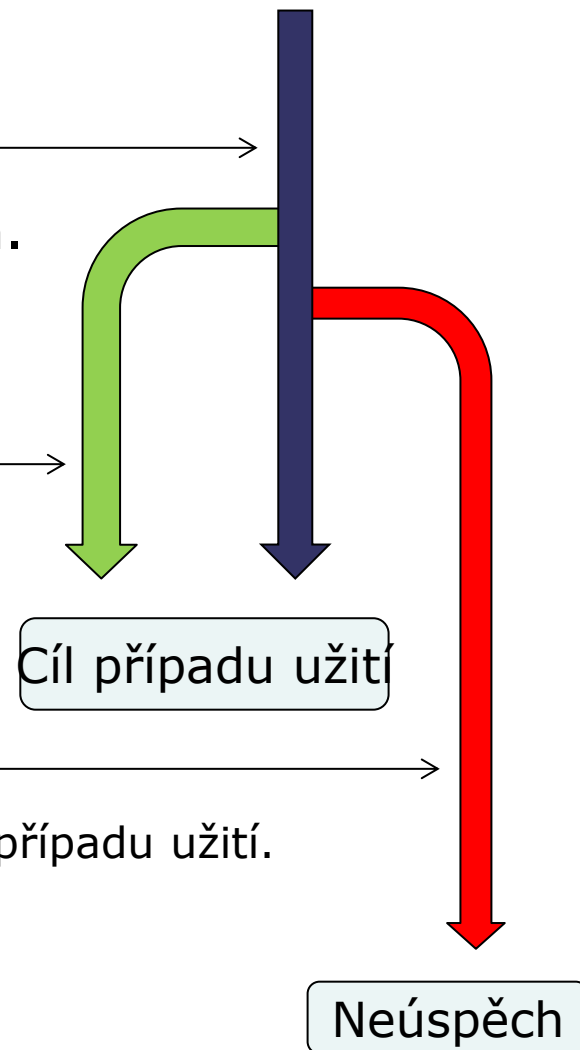
- Popisuje kroky případu užití, pokud vše jde jak má.

- **Alternativní scénář** (Alternative flow)

- Zachycuje **odchyly** od hlavního scénáře

- **Výjimečný scénář** (Exceptional flow)

- Popisuje varianty, kdy se nepodaří dosáhnout cíle případu užití.



Postup tvorby modelu případů užití

1. Nalezení a vhodného pojmenování hlavních UC a aktérů.
2. Stručný popis jednotlivých UC. Stačí pár vět.
3. Prvotní odhad složitosti realizace UC.
4. U netriviálních UC napsání prvotních scénářů (Hlavní scénář a identifikace vedlejších).
5. Detailní dopracování
 - Nové UC připojené vazbami <<include>> a <<extend>>
 - Dopracování alternativních scénářů
6. Revize celého modelu

(Ne)Používání uživatelských rolí ve scénářích

Upravit záznam

Umožňuje upravit jednotlivé položky u vybraného záznamu v katalogu. Seznam položek jednotlivých záznamů v katalogu je uveden v popisu tohoto balíčku.

Tok událostí:

1. Příklad užití začíná, když chce **lékař** upravit některý ze záznamů v katalogu.
2. INCLUDE (Vybrat katalog)
3. Systém požádá **lékaře** o výběr záznamu z katalogu, který chce upravovat.
4. INCLUDE (Zobrazit položky katalogu).
5. Systém zobrazí formulář umožňující upravit veškeré položky u vybraného záznamu.
6. **Lékař** upraví požadované údaje.
7. Systém uloží do záznamu všechny změny provedené **lékařem**.

Poznámka: Pozor! Jedná se o příklad z reálné analýzy, která obsahuje chyby různé chyby, které budou postupně vysvětleny.

Doporučuji nepoužívat uživatelské role ve scénářích

Upravit záznam

Umožňuje upravit jednotlivé položky u vybraného záznamu v katalogu. Seznam položek jednotlivých záznamů v katalogu je uveden v popisu tohoto balíčku.

Tok událostí:

1. Příklad užití začíná, když chce **uživatel** upravit některý ze záznamů v katalogu.
2. INCLUDE (Vybrat katalog)
3. Systém požádá **uživatele** o výběr záznamu z katalogu, který chce upravovat.
4. INCLUDE (Zobrazit položky katalogu).
5. Systém zobrazí formulář umožňující upravit veškeré položky u vybraného záznamu.
6. **Uživatel** upraví požadované údaje.
7. Systém uloží do záznamu všechny změny provedené **uživatelem**.

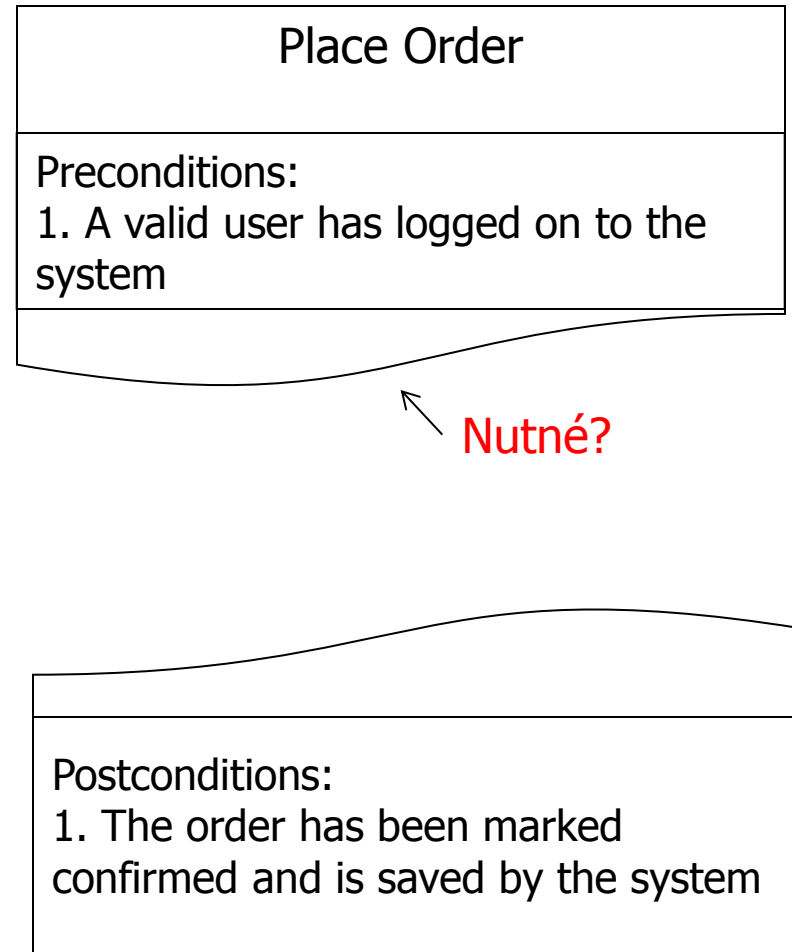
Doporučuji systémové role spíše nepoužívat, protože případ užití může být později přidělen k jiné uživatelské roli nebo více různým uživatelským rolím. Další možností je použití systémových rolí.

Poznámka: Pozor! Jedná se o příklad z reálné analýzy, která obsahuje chyby různé chyby, které budou postupně vysvětleny.

Pre a postconditions

dle Arlowa


- Preconditions obsahují stav systému před spuštěním UC (co musí platit, aby šel UC spustit)
- Postconditions obsahují stav systému po spuštění UC (efekt UC)



Hlavní scénář (Happy scenario)

dle Arlowa

<číslo> <aktér/systém> <akce>

- Posloupnost akcí
- Vždy začíná akcí aktéra  Dle M. Komárka není vždy nutné, protože akce začíná většinou rozhodnutím aktéra akci spustit.
 - Např:
 - 1) Use Case začíná když <actor> <function>
- Odpovídá situaci, kdy vše jde bez problémů a aktér dosáhne svého cíle
- Alternativy je možné vyjádřit větvením nebo alternativními scénáři

Větvení pomocí If

Use case: ManageBasket
ID: 2
Brief description: The Customer changes the quantity of an item in the basket.
Primary actors: Customer
Secondary actors: None.
Preconditions: 1. The shopping basket contents are visible.
Main flow: 1. The use case starts when the Customer selects an item in the basket. 2. IF the Customer selects "delete item" THEN The system removes the item from the basket. 3. IF the Customer types in a new quantity THEN The system updates the quantity of the item in the basket.
Postconditions: None.
Alternative flows: None.

Opakování pomocí For

Use case: FindProduct
ID: 3
Brief description: The system finds some products based on Customer search criteria and displays them to the Customer.
Actors: Customer
Preconditions: None.
Main flow: <ol style="list-style-type: none">1. The use case starts when the Customer selects "find product".2. The system asks the Customer for search criteria.3. The Customer enters the requested criteria.4. The system searches for products that match the Customer's criteria.5. If the system finds some matching products then<ol style="list-style-type: none">5.1 For each product found<ol style="list-style-type: none">5.1.1. The system displays a thumbnail sketch of the product.5.1.2. The system displays a summary of the product details.5.1.3. The system displays the product price.6. Else<ol style="list-style-type: none">6.1. The system tells the Customer that no matching products could be found.
Postconditions: None.
Alternative flows: None.

Vztahy mezi případy použití

- <<include>>

- <<extend>>

$$A \times B + A \times C + A \times D = ???$$

$$A \times B + A \times C + A \times D = A \times (B + C + D)$$

Vazba <<include>>

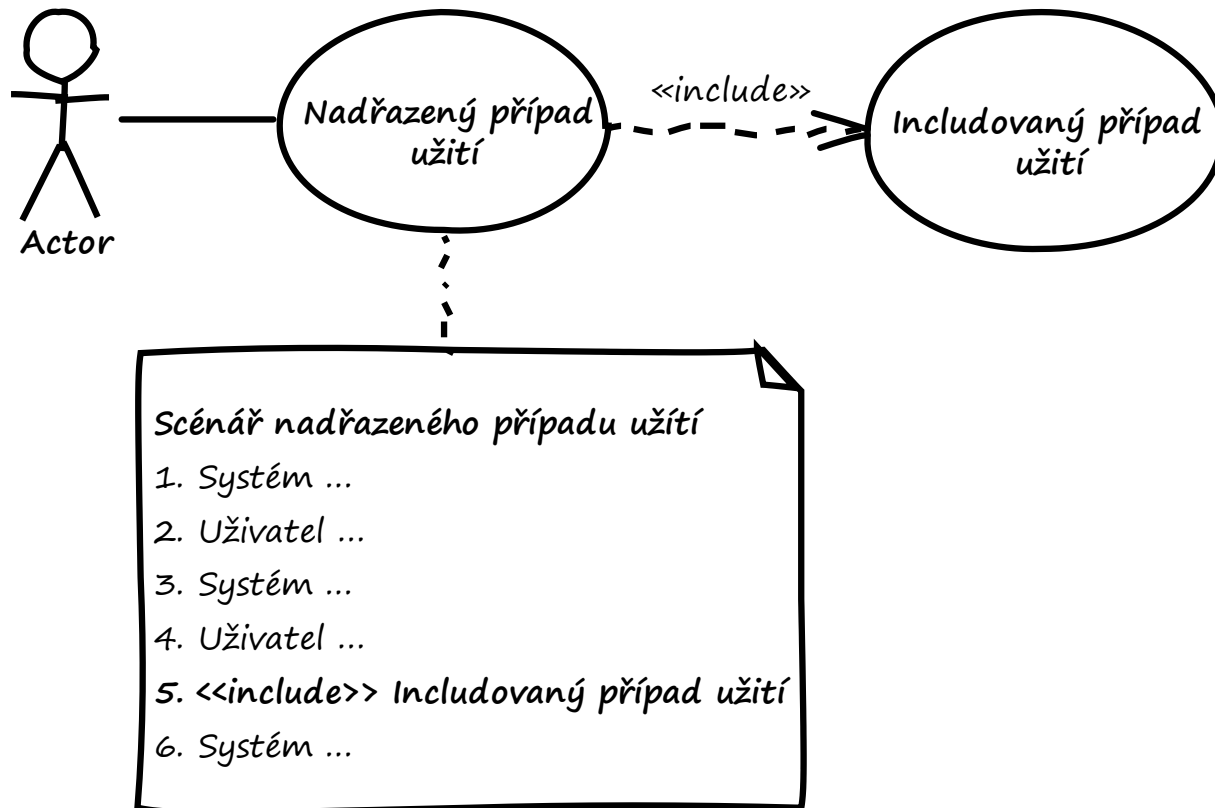
„Vytknutí“ několika navazujících kroků scénáře, pokud se opakují v několika případech užití.

=>

Zamezení duplicitám ve scénářích.

Vazba <<include>>

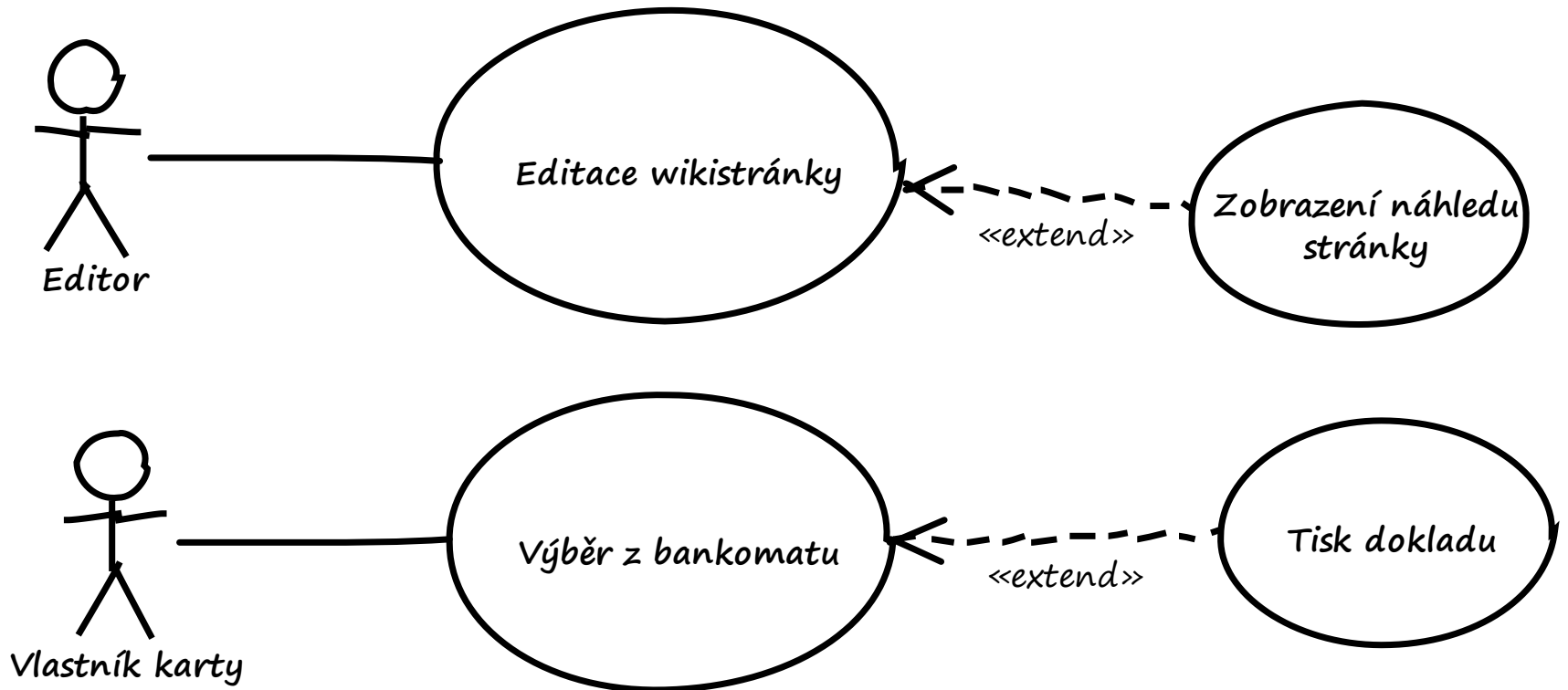
- „includovaný“ případ užití je vyvolán vždy, když dojde k vyvolání „nadřazeného“ případu užití
- kdy přesně bude „includovaný“ případ užití vyvolán musí být uvedeno ve scénáři „nadřazeného“ případu užití



Vazba <<extend>>

- Někaký případ užití rozšiřuje možnosti hlavního případu užití.
 - Hlavní případ užití funguje i bez rozšiřujícího případu užití.
 - Rozšiřující případ užití obvykle nemá samostatně žádný význam = je kontextově závislý na hlavním případě užití.
- Rozšiřující případ užití může být spuštěn:
 - Uživatelovým rozhodnutím.
 - Splněním nějaké podmínky = podmíněný <<include>>

Vazba <<extend>>



Chyby v UC - Vágní pojmenování UC

Upravit záznam

Umožňuje upravit jednotlivé položky u vybraného záznamu v katalogu. Seznam položek jednotlivých záznamů v katalogu je uveden v popisu tohoto balíčku.

Tok událostí:

1. Příklad užití začíná, když chce lékař upravit některý ze záznamů v katalogu.
2. INCLUDE (Vybrat katalog)
3. Systém požádá lékaře o výběr záznamu z katalogu, který chce upravovat.
4. INCLUDE (Zobrazit položky katalogu).
5. Systém zobrazí formulář umožňující upravit veškeré položky u vybraného záznamu.
6. Lékař upraví požadované údaje.
7. Systém uloží do záznamu všechny změny provedené lékařem.

Jaký „záznam“? Těch je v systému mnoho. Nutno dospecifikovat!

Poznámka: Pozor! Jedná se o příklad z reálné analýzy, která obsahuje různé chyby, které budou postupně vysvětleny.

Chyby v UC - Popis navigace ve scénářích

Upravit záznam

Umožňuje upravit jednotlivé položky u vybraného záznamu v katalogu. Seznam položek jednotlivých záznamů v katalogu je uveden v popisu tohoto balíčku.

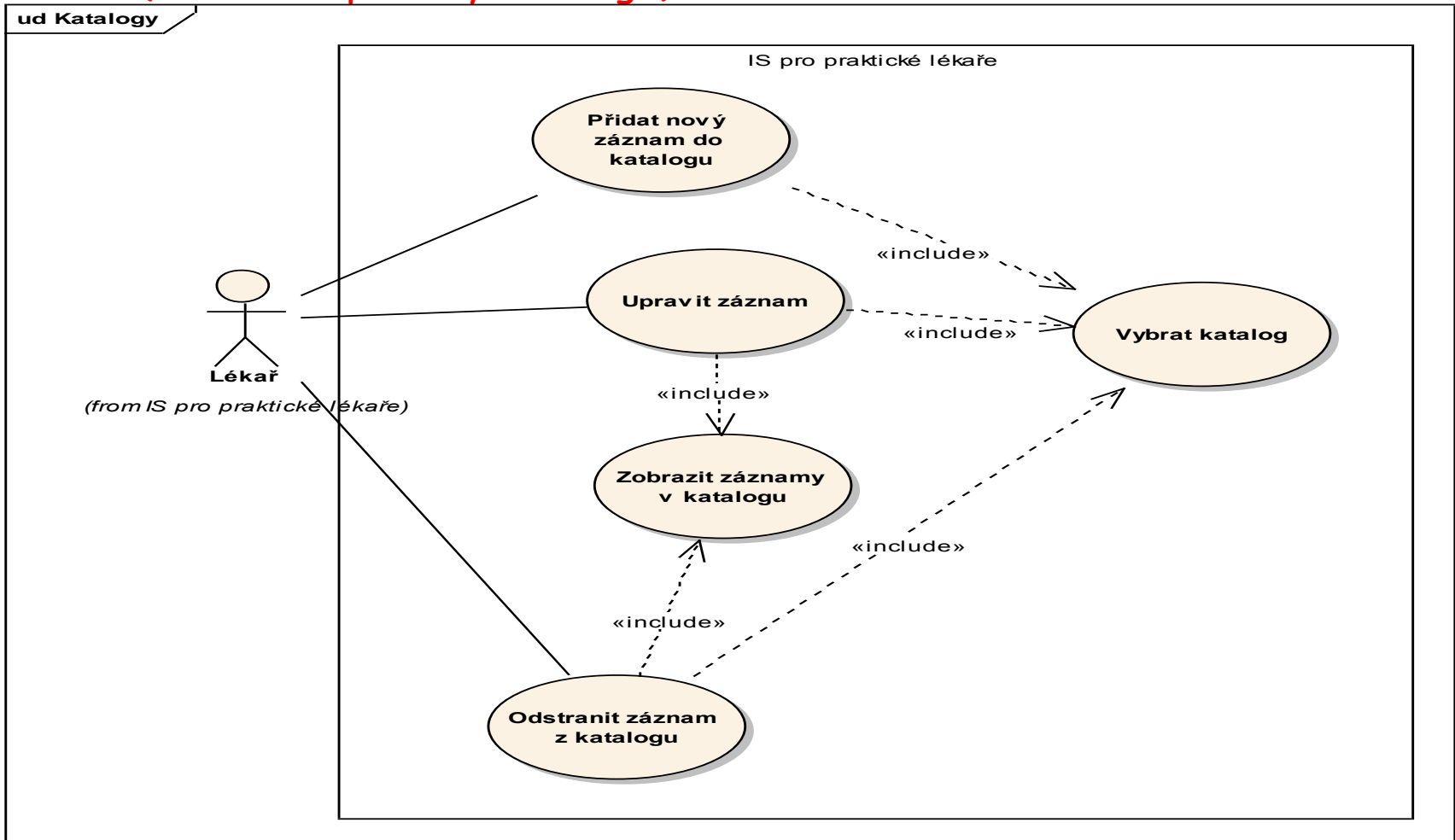
Tok událostí:

1. Příklad užití začíná, když chce lékař upravit některý ze záznamů v katalogu.
2. INCLUDE (Vybrat katalog)
3. Systém požádá lékaře o výběr záznamu z katalogu, který chce upravovat.
4. INCLUDE (Zobrazit položky katalogu).
5. Systém zobrazí formulář umožňující upravit veškeré položky u vybraného záznamu.
6. Lékař upraví požadované údaje.
7. Systém uloží do záznamu všechny změny provedené lékařem.

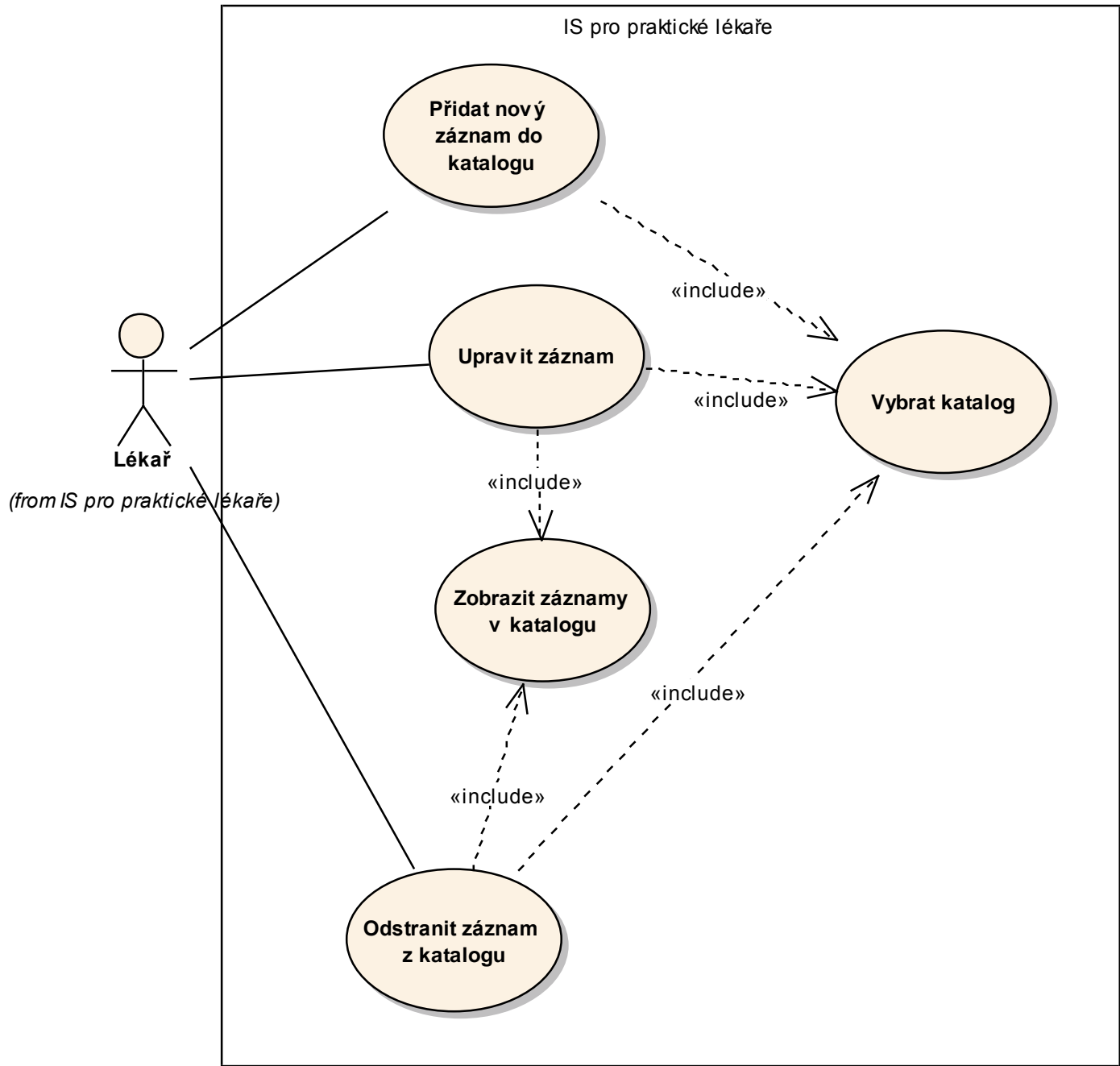
Poznámka: Pozor! Jedná se o příklad z reálné analýzy, která obsahuje různé chyby, které budou postupně vysvětleny.

Chyby v UC - Popis navigace ve scénářích

1. Příklad užití začíná, když chce lékař upravit některý ze záznamů v katalogu.
2. INCLUDE (Vybrat katalog)
3. Systém požádá lékaře o výběr záznamu z katalogu, který chce upravovat.
4. INCLUDE (Zobrazit položky katalogu).
5. ...



Poznámka: Pozor! Jedná se o příklad z reálné analýzy, která obsahuje různé chyby, které budou postupně vysvětleny.



Případy použití - nejčastější chyby

- Model zachycuje navigaci v systému.
- Model zachycuje časovou souslednost.
- Model obsahuje aktéra System místo aktéra Čas.
- Nesprávně používaný směr vazeb <<include>> a <<extend>>.

Matice požadavků a UC

- Sledovatelnost / Traceability požadavků.
- Ukázka v EA.