

**Analýza technológií, dekompozícia a dátový  
model k projektu  
Smelý zajko sprievodcom**  
Tvorba informačných systémov

**Autori:** Natália Ďurisová, Dominik Knechta,  
Martin Mudroch, Miroslav Gregorec

## Obsah

1	Úvod .....	3
2	Analýza technológií .....	4
2.1	Technológie na správu robota: .....	4
2.2	Technológie pre webové rozhranie : .....	5
3	Dekompozícia.....	6
3.1	Komponentový diagram.....	7
3.1.1	Popis komponentov.....	8

## 1 Úvod

Cieľom tohto dokumentu je analyzovať a popísať použité technológie, ktoré budú použité pri tvorbe projektu smelý zajko sprievodcom. Dané technológie budú v dokumente popísané slovne a taktiež znázornené pomocou komponentového diagramu. Keďže náš projekt je robot, ktorý nemá žiadne front-endové prostredie rozdeliteľné na komponenty, naša dekompozícia bude rozdelená podľa naprogramovaných tried (class), ktoré spravujú robota. Náš projekt taktiež **nepracuje s databázou, preto neobsahuje dátový model**. V dokumente sú okrem technológií na správu robota, popísané aj technológie použité pre webové prostredie a vzájomnú komunikáciu medzi robotom a webom.

## 2 Analýza technológií

Technológie môžeme rozdeliť na 2 skupiny podľa využitia:

- Technológie na správu robota
- Technológie pre webové rozhranie

### 2.1 Technológie na správu robota:

#### 1. Programovací jazyk (C++)

Podstatou celého projektu je programovací jazyk, v ktorom budú naprogramované všetky metódy na riadenie robota. Keďže úlohou robota je aj rozpoznávať postavy rozhodli sme pre jazyk C++, pretože najlepšie spolupracuje s knižnicou OpenCV bližšie opísanou nižšie. C++ je viacparadigmaticý programovací jazyk vyššej úrovne na všeobecné použitie, ktorý umožňuje pracovať aj s prostriedkami nízkej úrovne. Má statickú typovú kontrolu, podporuje procedurálne programovanie, dátovú abstrakciu, objektovo orientované programovanie, ale aj generické programovanie.

#### 2. Knižnica na rozpoznávanie postáv (OpenCV)

Počas jazdy robot sníma prostredie pomocou webkamery. Obrázok z webkamery sa odosiela na rozpoznávanie či sa na ňom nachádza nejaká postava. Takúto úlohu dokáže spracovať vybraná knižnica OpenCV. OpenCV je tzv. open-source, multiplatformová knižnica určená pre manipuláciu s obrazom. Je zameraná predovšetkým na počítačové videnie a spracovanie obrazu v reálnom čase.

#### 3. Mapy vo formáte SVG

Nato aby robot mohol správne navigovať po informatickom pavilóne potrebuje poznať polohu miestností a taktiež kde sa práve nachádza. Na tento účel použijeme už vopred pripravené SVG mapy informatického pavilónu, v ktorých sú zaznamenané súradnice potrebných miestností. Scalable Vector Graphics (SVG) je značkový jazyk z rodiny značkových jazykov XML, ktorý je určený na opis dvojrozmernej, statickej alebo animovanej vektorovej grafiky.

#### 4. Hlasový syntetizátor pre komunikáciu s človekom

Keďže robot komunikuje s človekom aj verbálne, potrebujeme na daný problém takzvaný TTS (text-to-speech) program. Robot komunikuje s človekom slovenským jazykom, preto bude potrebné použiť syntetizátor, ktorý podporuje daný jazyk.

## 2.2 Technológie pre webové rozhranie :

### 1. Hypertext Markup Language (HTML)

HTML je jednoduchý značkovací jazyk bežne používaný na tvorbu statických webových stránok zobrazovaných vo webovom prehliadači. Tvorí základ pre rozšírenie stránok o dynamické prvky pomocou jazykov ako napr. PHP. Jeho výhodou je jednoduchosť a bežná podpora zo strany internetových prehliadačov. Veľmi často je používaný spolu s kaskádovými štýlmi (CSS). Jazyk bude použitý pre statický návrh webového prostredia k robotovi, ktorý má informačný aj spravovací charakter.

### 2. Cascading Style Sheets (CSS)

Kaskádové štýly alebo CSS je všeobecné rozšírenie (X)HTML. Organizácia W3C označuje CSS ako jednoduchý mechanizmus na vizuálne formátovanie internetových dokumentov. Štýly umožňujú oddeliť štruktúru HTML alebo XHTML od vzhľadu. Pre náš projekt bude CSS použité na design webovej stránky.

### 3. Hypertext Preprocessor (PHP)

PHP je populárny open source skriptovací jazyk, ktorý sa používa najmä na programovanie klient-server aplikácií (na strane servera) a pre vývoj dynamických webových stránok. Pre náš projekt bude PHP využitý na prijímanie údajov o stavoch robota, ich následné spracovanie a taktiež na prihlasovanie do administrátorského prostredia.

## 3 Dekompozícia

Projekt bude rozdelený na dve časti:

### 1. Program pre robota.

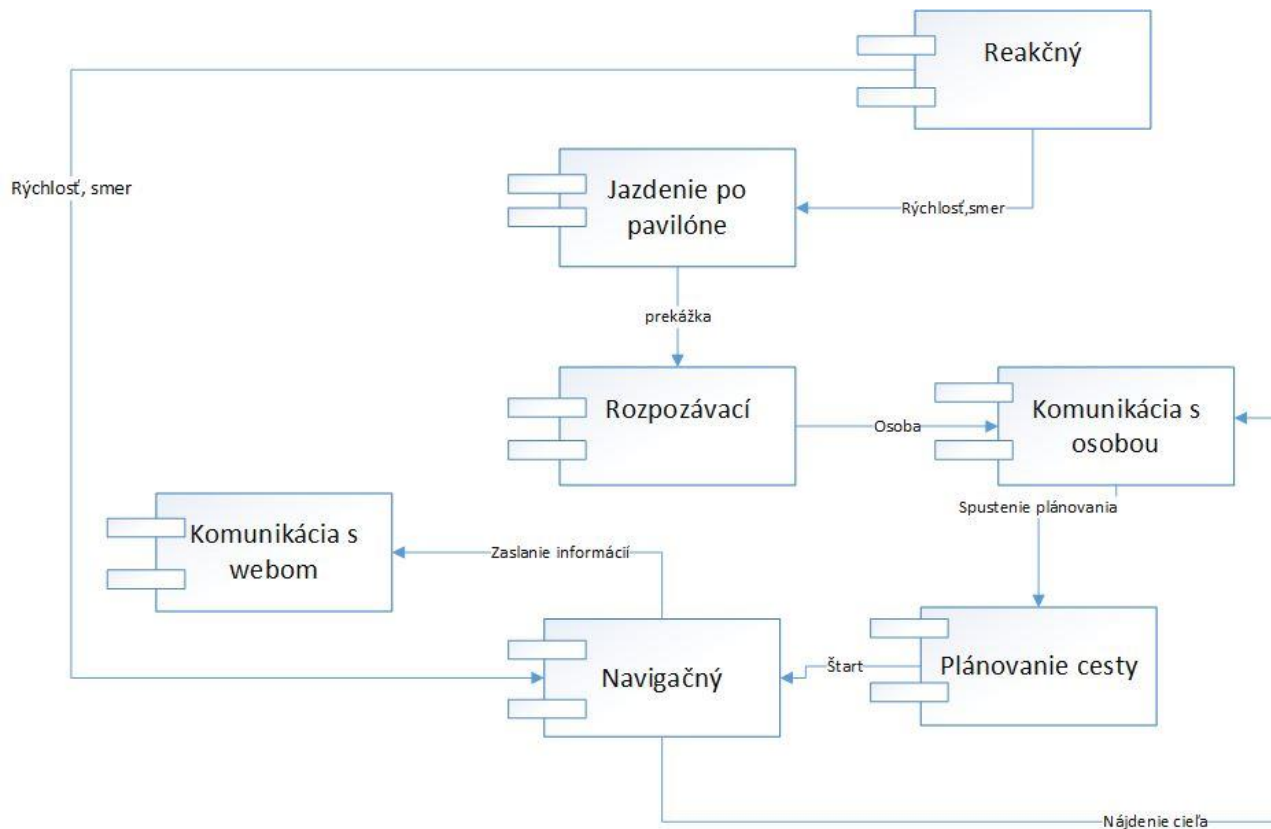
Tento program bude obsahovať triedy potrebné pre jazdu robota po pavilóne, komunikáciu s človekom, komunikáciu so serverom a vyhľadávanie miestností.

- Jazda:  
Má metódy na korigovanie smeru a rýchlosti jazdy, výpočet aktuálnej pozície, kontrolu výskytu osoby alebo prekážky. Pamätá si svoju aktuálnu pozíciu a cieľ jazdy.
- Rozpoznávanie prítomnosti človeka:  
Pomocou OpenCV sa spracúva výstup z kamery a reaguje na prítomnosť človeka.
- Komunikácia s človekom:  
Obsahuje metódy na hlasový výstup robota, spracovanie a kontrolu vstupu a časovač.
- Komunikácia so serverom:  
Metódy na odosielanie dát na server a na prijímanie dát zo serveru.

### 2. Webová stránka.

Na webovej stránke budú bežnému užívateľovi k dispozícii údaje o momentálnom stave robota a história jeho úloh. Admin má možnosť robota vypnúť alebo zapnúť.

### 3.1 Komponentový diagram



Obrázok 1 Komponentový diagram

## 3.1.1 Popis komponentov

### 3.1.1.1 Reakčný komponent

Komponent je potrebný pre samotný štart robota, po spustení sa zapnú senzory kamera a spustí sa aj samotný program prechádzania robota. V reakčnom komponente sa ďalej sleduje vzdialenosť robota od stien a zaisťuje sa správny pohyb po chodbe.

### 3.1.1.2 Komponent Jazdenie po pavilóne

Po spustení robota sa zapína program, ktorý zabezpečuje samotnú jazdu po pavilóne, čo zahŕňa pamätanie pozície robota, ako aj kontrolovanie prekážok pri robotovi.

### 3.1.1.3 Rozpoznávací komponent

Tento komponent využíva predovšetkým knižnicu OpenCV, pomocou ktorej sa snaží rozpoznať objekt nachádzajúci sa pred robotom. Po zistení, že pred robotom sa nachádza osoba, komponent odošle informáciu a spustí sa komunikačný komponent.

### 3.1.1.4 Komponent komunikácia s osobou

V tomto komponente sa prevažne využíva hlasový syntetizátor, ktorý oboznámi používateľa o postupe navigácie.

### 3.1.1.5 Komponent plánovanie cesty

Po zadaní vstupu od užívateľa sa v tomto komponente vypočíta najkratšia cesta k zvolenému miestu a následne sa spustí komponent navigácie.

### 3.1.1.6 Komponent Navigácia

Po úspešnom zadaní vstupu od rozpoznannej osoby sa spustí modul navigácia, ktorý bude postupovať podľa vypočítanej najkratšej cesty.

### 3.1.1.7 Komponent komunikácia s webom

Tento komponent slúži na komunikáciu s web stránkou na ktorú po ukončení navigácie odošle informácie o navigovaní. Po odoslaní sa znova spúšťa program na jazdenie po pavilóne.