

Спецификација пројекта

Имплементирати информациони систем за подршку продавници и сервису рачунара и рачунарских компоненти.

Функционалности које систем треба да пружи дате су у наставку.

Функционални захтеви

1. Предлагање рачунара и рачунарских компоненти на основу задатих својстава.

- Својства зависе од рачунарске компоненте.
- Пример: за задате вредности (или опсег вредности) фреквенције радног такта и броја језгара процесора, систем предлаже конкретне моделе процесора.

2. Процена ваљаности предложеног рачунара и рачунарских компоненти за одређену намену.

- Врсте намене су: развој апликација, играње видео игара, рударење криптовалута, кућна употреба, пословна употреба и хостинг.
- Пример: за задате вредности својстава компоненти, систем даје меру ваљаности датих рачунарских компоненти за одређену намену.

3. Израчунавање вероватноће узрока кvara предложеног рачунара или компоненте.

- Потребно је упознати се са симптомима кварова различитих рачунарских компоненти.
- За сваку рачунарску компоненту омогућити одговор на питање, колика је вероватноћа да је квар одређене компоненте узрок кvara рачунара.

4. Преглед рачунара са компонентама, који су најсличнији предложеном.

Технички захтеви

1. Креирати модуларизовану онтологију која формално специфицира знање о рачунарима и рачунарским компонентама.

- Модуларизација подразумева раздвајање онтологије у више датотека – бар једну датотеку која садржи класе и својства и бар једну датотеку која садржи индивидуе.
 - Потребно је да укупан број класа буде најмање 150 (од којих најмање 30 да буде дефинисано помоћу конструкта језика OWL), а број индивидуа најмање 210.
 - Свака класа мора бити везана за неко својство.
 - Онтологија треба да задовољава стандард OWL 2, профила RL (*rules logic*).
 - Као помоћ при креирању класа рачунара и рачунарских компоненти, искористити отворени граф знања DBpedia (<https://dbpedia.org/page/Computer>). У обзир узети и софтверске и хардверске компоненте.
 - Као помоћ при креирању инстанци рачунара и рачунарских компоненти, искористити следеће веб-сајтове: <https://www.cpubenchmark.net/>, <https://www.videocardbenchmark.net/>, <https://www.harddrivebenchmark.net/>, <https://www.memorybenchmark.net/>, <https://www.pcbenchmarks.net/>, <https://cpu.userbenchmark.com/>, <https://gpu.userbenchmark.com/>,
-

<https://ssd.userbenchmark.com/>, <https://hdd.userbenchmark.com/>, <https://ram.userbenchmark.com/>, <https://usb.userbenchmark.com/>.

– Препорука је да се за моделовање онтологије користи програм Protege (<https://protege.stanford.edu/>), а за програмску манипулацију онтологијом – апликациони програмски интерфејс OWL базиран на технологији Java (<https://owlcs.github.io/owlapi/>).

– Сваки члан тима треба да имплементира један модул који садржи класе и својства.

2. За потребе процене ваљаности предложеног рачунара за одређену намену, имплементирати фази системе.

– Имплементирати посебан фази систем за сваку врсту намене.

– Препорука је да се за улазне променљиве наведу својства компоненте рачунара.

– За излазну променљиву фази система (процену ваљаности рачунара за одређену намену) у процесу дефазификације навести термине који градацијски означавају ваљаност (нпр. довољно, добро, врло добро, одлично) и у складу са тим дефинисати фази скупове.

– Приликом дефинисања фази скупова, водити рачуна о јединицама мере својстава рачунарских компоненти и њиховим минималним и максималним вредностима (у процесу фазификације).

– Потребно је да свака рачунарска компонента дефинисана у онтологији буде део скупа улазних променљивих и да се свака рачунарска компонента са својим својствима нађе у неком од фази правила.

– Сваки члан тима треба да имплементира два фази система.

3. Вероватноћу узрока кvara рачунара или рачунарске компоненте израчунати користећи Бајесово расуђивање.

– Потребно је да укупан број случајних променљивих буде најмање 90.⁶

– Нека је вероватноћа појаве симптома кварова код појединачних рачунарских компоненти једнако распоређена.

– Сваки члан тима треба да имплементира једну подмрежу.

4. Предлагање рачунара најсличнијих датом имплементирати технологијом расуђивања по случајевима.

– Потребно је да база случајева садржи најмање 120 рачунара описаних вредностима својстава њихових компоненти.

– Приликом одређивања функција сличности водити рачуна о типовима својстава.

– Резултат представити са 5 најсличнијих рачунара.

Нефункционални захтеви

1. Пројекат се имплементира у трочланим тимовима.

– Потребно је поделити посао тако да сваки члан тима подједнако допринесе креирању онтологије, фази система, Бајесове мреже и базе случајева.

2. Систем за управљање пројектом је GitHub.

- Сваки тим треба да креира јавни репозиторијум.
- Допринос чланова тима мериће се на основу доприноса на GitHub репозиторијуму.

3. Рок за предају пројекта је недељу дана пре заказаног термина одбране.

- Измене кода на репозиторијуму након истека рока за предају пројекта нису дозвољене.
 - Након истека рока за предају, сви пројекти биће подвргнути провери постојања плагијата.
-