

*Tvorba informačných systémov, diagramy UML, 12.11.2019.*

Čerství absolventi študijného programu Bioinformatika dokázali svoju profesionalitu a vyvinuli nanoroboty, ktoré sa dostávajú do tela subjektu vdýchnutím, následne preniknú do krvného obehu a napok obsadia nervovú sústavu a úpravou chemického prostredia v mozgu modifikujú emocionálne i racionálne reakcie subjektu na deje okolo neho podľa požiadavky vzdialeného operátora. Politické strany neváhali a jedna zo zelených strán získala recept na výrobu týchto nanozariadení. Len vďaka tomu sa podarilo otupiť odpor a presadiť nové prísne environmentálne pravidlá a tak sa každý rok rozširujú chránené a bezzásahové územia o desiatky tisícov hektárov. To však spôsobilo nárast agendy a tak sa obrátili na vás.

Navrhňte informačný systém, ktorý bude evidovať chránené územia. Používateľmi budú: 1) občania, ktorí si budú môcť pozerieť informácie o územiach, súvisiacu legislatívu i nariadenia, verejné poriadky v parkoch, flóru a faunu (vrátane možnosti nahrávania záznamov získaných z fotopascí), mapy a turistické mapy, nahlasovať porušenia ochrany prírody alebo akýchkoľvek zákonov a predpisov aj cez mobil, dokonca budú môcť navrhovať nové územia, ktoré majú byť chránené, alebo opatrenia, ktoré majú byť prijaté; 2) Štátna ochrana prírody SR, ktorá celý systém spravuje, do systému vkladá všetky informácie a zároveň reaguje na podnety od občanov; 3) Ministerstvá a štátna správa, ktorá získava informácie pre tvorbu legislatívy i rozpočtu, výber daní, keďže systém eviduje aj všetky komerčné subjekty, ktoré na chránených územiach existujú, čo má na ich podnikanie daňové dopady; 4) Samotná komerčná sféra, ktorá získava informácie ovplyvňujúce ich podnikanie na chránených územiach; 5) Správy jednotlivých chránených území (napr. správa národného parku Vydrica), pričom jej členovia monitorujú situáciu priamo v teréne a cez systém nahlasujú zistené nedostatky, udelené sankcie, resp. vytvárajú úlohy, ktoré treba plniť.

Nakreslite use-case diagram, v ktorom je zachytená celková funkcionálna aplikácie, stavový diagram pre nejakú entitu (entita GUI, ani celá aplikácia nevyhovuje), sekvenčný diagram pre niektorý use-case tak, aby nebol triviálny (využite v ňom nejaké bloky), triedny diagram nejakej zmysluplnej časti aplikácie, v ktorom využijete všetky tri základné druhy vzťahov medzi triedami a deployment diagram pre netriviálnu časť celého systému.