Szoftver Specifikációt és Analízist segítő eszköz

A készítendő szoftver alap koncepciója, hogy az UML-ben megjelenő modellelemek kapcsolata mellett, igyekezne egy saját modell keretében összekapcsolni, a különböző absztrakciókon, illetve diagramokon, megjelenő modelleket, illetve egyes modellelemeit képes lenne összekötni, az adott elemet leíró specifikációs szöveggel. Ezt a jobb érthetőség kedvéért egy példával reprezentálnám:  
A Felhasználó, kijelöl egy specifikációs szövegrészt, majd egy menüből kiválaszt, egy a szöveghez már létrehozott diagramot, ekkor a rendszer oda navigál. Lényegében egy gráf épül föl melynek minden csomópontja vagy egy szövegrész, vagy egy diagram, esetleg egyszerűen egy UML modell elem.

A kapcsolatok, és szövegek szemléletesebbé tétele érdekében számos nézetet biztosítana. Az utóbbira példa lehet, a szövegben az Aktorok, illetve használati esetek továbbá, szintén a szoftver által biztosított Domain, illetve Project szótárak, szavainak megjelölése. Lehetne olyan nézete is, ami megjelöli azokat a szövegrészeket, amikhez még nem készült diagram. (Használati eset diagram mellett/helyett módot adhat User Story-k elkészítésére is)

A szakdolgozat keretein belül kísérletet tennék egyes magas absztrakciós diagramok a szövegből való generálására is. Ezt esetleg az adott fejlesztéshez kapcsolható szakirányok szótára segítségével, esetlegesen szövegbányászati eszközökkel kívánok elérni.

## Az irodalomkutatás keretein belül:

* Lehetséges, hogy említés szintjén visszanyúlnék Abbot (’83), aki a szöveg nyelvtani elemzését tartotta az analízis egy lehetséges kiinduló pontjának (pl: minden főnév lehet objektum, minden ige egy lehetséges művelet)
* Esetleg arra is kitérnék, hogy az OMT, milyen lépéseket dolgozott ki.
* Megvizsgálnám, hogy „a három amígó” hogyan/ miért helyezte át a hangsúlyt a használati esetekre.
* Végül szakirány vezérelt (Domain Driven) módszertanok rövid összefoglalásával alapoznám meg a szöveg alapján történő diagram készítés elméleti hátterét.
* Esetleges szövegbányászati irodalmak feldolgozása.

## Esetleges funkciók vázlatosan:

1. Specifikáció
   1. vizualizációs segítség
   2. szövegezés segítség
   3. formázás segítség, szabványos formának való megfelelés hitelesítése.
   4. régebbi specifikációs szövegek, illetve egyéb szöveges dokumentumok, az adott szöveghez való hasonlítása segítségével, segíthetne már kész modulok felderítésében.
   5. diagram generálás
2. Analízis
   1. entitások, feladatok kijelölése
   2. feladat részekre bontásában, illetve esetleges priorizálásukban való segítség.
   3. diagram generálás
   4. a specifikációs szövegrészek, és az analízis diagram összefüggő részeinek összekapcsolása, együtt mutatása, köztük egyszerű navigáció biztosítása
3. Céges szabványok betartatása
   1. metodika
   2. könyvtár szerkezet
4. Esetleges verziókövetés
5. Csapat munka, csapatos diagramkészítés

## Metodikák

Az elkészítendő szoftvernek nem célja egy bizonyos metodika támogatása, ellenben modellező eszköz (modelling tool) is, így az összes modellvezérelt metodikával jól használható. Ellenben tartalmazhat sablonokat, melyek használatát megköveteli, betartatja és például generálásnál be is tartja.

## Technológia

Az eszközt vastagkliens-szerver architektúrában valósítanám meg.

A szerver oldalon feltehetőleg Java EE, vagy Spring MVC-t használnék.

A kliens oldalon napjainkban egy, egyre inkább elterjedő, és szinte natív alkalmazáshoz hasonló élményt nyújtó, SPA (Single Page Application) webes felület lenne. Ezt minden bizonnyal HTML5, CSS, Java Script (+ Durandal framework), illetve Type Script segítségével valósítanám meg.