Programska podrška izradi izvješća sa sastanka projektnog tima

Breški, Tomislav

Master's thesis / Diplomski rad

2018

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Mechanical Engineering and Naval Architecture / Sveučilište u Zagrebu, Fakultet strojarstva i brodogradnje**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:235:873297

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-03-28**

Repository / Repozitorij:

Repository of Faculty of Mechanical Engineering and Naval Architecture University of Zagreb

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

DIPLOMSKI RAD

Tomislav Breški

Zagreb, 2018.

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

DIPLOMSKI RAD

Mentori:

Prof. dr.sc. Neven Pavković

Student:

Tomislav Breški

Zagreb, 2018.

Izjavljujem da sam ovaj rad izradio samostalno koristeći znanja stečena tijekom studija i navedenu literaturu.

Zahvaljujem se obitelji na neupitnoj podršci tijekom studiranja, te svim kolegama s kojima je vrijeme na faksu postalo poprilično zabavno iskustvo. Također, zahvaljujem se djevojci Maji koja je bila nepokoriva potpora tokom cijelog studija.

Također se zahvaljujem mentoru prof. Nevenu Pavkoviću, te kolegi Jasminu Juraniću koji su pomogli u izradi ovog rada konkretnim prijedlozima i kvalitetnim usmjeravanjem omogućili izradu ovog rada.

Tomislav Breški

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE
Središnje povjerenstvo za završne i diplomske ispite
Povjerenstvo za diplomske ispite studija strojarstva za smjerove:
procesno-energetski, konstrukcijski, brodostrojarski i inženjersko modeliranje i računalne simulacije

Sveučilište u Zagrebu Fakultet strojarstva i brodogradnje	
Datum	Prilog
Klasa:	
Ur. broj:	

DIPLOMSKI ZADATAK

Student: **Tomislav Breški** Mat. br.: 0035191278

Naslov rada na hrvatskom jeziku: **Programska podrška izradi izvješća sa sastanka projektnog tima**

Naslov rada na engleskom jeziku: **Software support for preparing reports from the project team meeting**

Opis zadatka:

U multidisciplinarnom timskom radu na razvoju proizvoda, koordinacija i komunikacija među članovima tima predstavlja problematiku od primarnog interesa. Na sastancima projektnih timova donose se važne odluke i raspodjeljuju zadaci te na temelju diskusija utvrđuju daljnji pravci razvoja. U velikim tvrtkama odnosno na većim projektima izvješće projektnog sastanka obično sadrži veliku količinu informacija koje treba strukturirati i proslijediti odgovarajućim članovima tima.
U uvodnom dijelu rada potrebno je proučiti rezultate istraživanja postojećeg procesa pripreme izvješća koje je provedeno u tvrtki Daimler AG. Također treba proučiti literaturu koja predlaže različite metodologije unaprjeđenja upravljanja informacijama tijekom i nakon sastanaka projektnog tima.
Nadalje, prema specifičnostima upravljanja razvojnim procesom u tvrtki Daimler AG u radu je potrebno napraviti sljedeće:

1. Izdvojiti i strukturirati podatke koji se najčešće pojavljuju u izvješćima – primjerice nazive konstrukcijskih aktivnosti i objekata na kojima se aktivnosti provode, podatke o članovima tima i zadacima i sl. Podatke treba spremiti u relacijsku bazu, te izraditi sučelje za održavanje.
2. Predložiti načine primjene tako strukturiranih podataka u procesu „polu-automatizirane" izrade izvješća s projektnog sastanka. Izvješće treba strukturirati tako da se omogući biranje elemenata spremljenih u bazu podataka te njihovo kombiniranje i „slaganje" u nizove teksta.
3. Cilj rada je razviti programsku aplikaciju koja će ubrzati i olakšati izradu izvješća sa sastanka projektnog tima. Pri tome treba posebnu pažnju posvetiti formalizaciji zapisa da bi se omogućila što efikasnija obrada zapisa drugim programskim alatima koji se koriste u procesu upravljanja projektom jer razvijena aplikacija treba biti jedan dio složenog sustava programske podrške.

U radu je potrebno navesti korištenu literaturu i eventualno dobivenu pomoć."

Zadatak zadan:	Datum predaje rada:	Predviđeni datum obrane:
8. ožujka 2018.	10. svibnja 2018.	16., 17. i 18. svibnja 2018.

Zadatak zadao: Predsjednica Povjerenstva:

Prof. dr. sc. Neven Pavković Prof. dr. sc. Tanja Jurčević Lulić

SADRŽAJ

1. UVOD 1

2. ANALIZA POSTOJEĆIH ISTRAŽIVANJA O UPRAVLJANJU PODATCIMA GENERIRANIH TIJEKOM I NAKON SASTANAKA PROJEKTNIH TIMOVA 2

2.1. SASTANAK PROJEKTNOG TIMA 3

2.2. ONTOLOGIJA AKTIVNOSTI U RAZVOJNOM PROCESU [4] 7

2.2.1. Klasifikacija aktivnosti definiranja 9

2.2.1.1. Apstrahiranje 10

2.2.1.2. Generiranje 10

2.2.1.3. Dekomponiranje 11

2.2.1.4. Povezivanje 11

2.2.1.5. Komponiranje 12

2.2.1.6. Strukturiranje 12

2.2.1.7. Detaljiranje 13

2.2.1.8. Standardiziranje 13

2.2.2. Klasifikacija aktivnosti ocjenjivanja 14

2.2.2.1. Donošenje odluka 14

2.2.2.2. Ocjenjivanje 15

2.2.2.3. Odabir 16

2.2.2.4. Analiziranje 16

2.2.2.5. Modeliranje 17

2.2.2.6. Simuliranje 17

2.2.2.7. Testiranje 18

2.2.3. Klasifikacija aktivnosti upravljanja 18

2.2.3.1. Ograničavanje 19

2.2.3.2. Istraživanje 19

2.2.3.3. Identificiranje 20

2.2.3.4. Skupljanje informacija 20

2.2.3.5. Rješavanje 21

2.2.3.6. Pretraživanje 21

2.2.3.7. Prioritiziranje 22

2.2.3.8. Planiranje 22

2.2.3.9. Zakazivanje 22

2.3. PETRI MREŽE [5] 23

2.3.1. Obojene Petri mreže [6] 26

3. PREGLED STUDIJE SLUČAJA PROVEDENE UNUTAR TVRTKE DAIMLER AG..... 30

4. SOFTVERSKO RJEŠENJE UPRAVLJANJA INFORMACIJAMA DOBIVENIH TIJEKOM INŽENJERSKIH SASTANAKA 33

4.1. BAZA PODATAKA..... 34

4.1.1. Hijerarhijska struktura aktivnosti na sastancima..... 36

4.1.2. Određivanje odgovornosti za pojedini zapis sastanka 38

4.1.3. Održavanje baze podataka 38

4.2. APLIKACIJA „DAIMLER AG DESIGN REVIEWS“ 40

4.2.1. Osnovne postavke za rad aplikacije..... 43

4.2.2. Rezervacija sastanka 43

4.2.3. Održavanje sastanka..... 44

5. ZAKLJUČAK 50

POPIS SLIKA

Slika 1. Shematski prikaz razvoja procesa s kontrolnim točkama [2] 4
Slika 2. Formalizacija aktivnosti u razvoju proizvoda [4] 8
Slika 3. Taksonomija aktivnosti definiranja [4] 10
Slika 4. Shematski prikaz aktivnosti ocjenjivanja [4] 14
Slika 5. Shematski prikaz aktivnosti upravljanja razvojnim procesom [4] 19
Slika 6. Osnovna Petri mreža s dvije pozicije i jednom tranzicijom 24
Slika 7. Ostvareni uvjeti za razrješenje tranzicije 25
Slika 8. Petri mreža [5] 25
Slika 9. CPN model slanja tekstualnih paketa u inicijalnom stanju M_0 [6] 26
Slika 10. CPN model nakon aktiviranja tranzicije *SendPacket* [6] 27
Slika 11. CPN model nakon aktiviranja tranzicije *TransmitPacket [6]* 28
Slika 12. CPN model nakon aktiviranja tranzicije *ReceivePacket [6]* 28
Slika 13. Stanje CPN modela nakon prvog uspješnog prijenosa informacije [6] 29
Slika 14. CPN model s modulima pošiljatelja [6] 29
Slika 15. Primjer izvješća sa sastanka unutar tvrtke Daimler AG 31
Slika 16. Prijedlog toka informacija u ciklusu između sastanaka 33
Slika 17. Grafički prikaz baze podataka 35
Slika 18. Obrazac za uređivanje baze zaposlenika 39
Slika 19. Obrazac za uređivanje timova 40
Slika 20. Struktura baze timova i zaposlenika 40
Slika 21. Struktura obrazaca 42
Slika 22. Glavni meni aplikacije 42
Slika 23. Obrazac za uređivanje postavki aplikacije 43
Slika 24. Obrazac za rezervaciju sastanka 43
Slika 25. Obrazac za izbor zapisničara 44
Slika 26. Obrazac za izbor sastanka 44
Slika 27. Obrazac za vođenje sastanka 45
Slika 28. Aktivnosti provedene tokom sastanka 46
Slika 29. Odabir odgovornih osoba i roka izvršenja aktivnosti 46
Slika 30. Evidencija zapisa na sastanku 47
Slika 31. Predložak izvještaja sa sastanka 47

POPIS TABLICA

Tablica 1. Pregled postojećih istraživanja u području analize razvojnih sastanaka 2
Tablica 2. Postotni prikaz učestalosti pojedinih razloga organiziranja sastanka..... 3
Tablica 3. Apstrahiranje 10
Tablica 4. Generiranje 11
Tablica 5. Dekomponiranje 11
Tablica 6. Povezivanje 12
Tablica 7. Komponiranje..... 12
Tablica 8. Strukturiranje..... 13
Tablica 9. Detaljiranje 13
Tablica 10. Standardiziranje..... 14
Tablica 11. Donošenje odluka..... 15
Tablica 12. Ocjenjivanje 15
Tablica 13. Odabir..... 16
Tablica 14. Analiziranje 16
Tablica 15. Modeliranje 17
Tablica 16. Simuliranje 18
Tablica 17. Testiranje..... 18
Tablica 18. Ograničavanje..... 19
Tablica 19. Istraživanje 20
Tablica 20. Identificiranje 20
Tablica 21. Skupljanje informacija 21
Tablica 22. Rješavanje 21
Tablica 23. Pretraživanje..... 21
Tablica 24. Prioritiziranje..... 22
Tablica 25. Planiranje..... 22
Tablica 26. Zakazivanje 23
Tablica 27. Osnovni elementi Petri mreže 24
Tablica 28. Hijerarhijski popis aktivnosti 36
Tablica 29. Osnovni podatci o održanom sastanku..... 48
Tablica 30. Zapisi na sastanku 48

SAŽETAK

Inženjerski sastanci jedan su od bitnih čimbenika koji utječu na smjer razvoja odabranog proizvoda. Informacije i odluke koje proizlaze iz inženjerskih sastanaka često nisu zapisane na adekvatan način, što onemogućuje kasnije kvalitetno analiziranje provedenih sastanaka. U uvodnom dijelu ovog rada detaljno su opisani aspekti i vrste sastanaka, aktivnosti u procesima razvoja proizvoda, te Petri mreže koje služe za simuliranje stvarnih procesa. Na temelju studije slučaja unutar tvrtke Daimler AG formirana je hijerarhijska struktura aktivnosti koja pomaže u procesu generaliziranja aktivnosti vođenih tokom sastanaka. U ovom radu detaljno je opisana struktura baze podataka te aplikacija za izradu izvješća sa sastanka povezana s bazom podataka. Nakon opisa aplikacije prikazan je proces izrade izvješća koji drastično smanjuje potrebno naknadno uređivanje izvješća.

Ključne riječi: inženjerski sastanak, aktivnost, Petri mreža, baza podataka, izvješće sa sastaka

SUMMARY

Design reviews are one of the most important factors which influence on product design development course. Information and decisions which are generated during design reviews often are not adquately noted, which makes following analysing of held meetings quite a difficult task. In itroductory part of this thesis detailed design reviews aspects and types, design process activities, and Petri nets, which are used for simulating real processes, are described. Based on case study held in company Daimler AG meeting activities hyerarchy is developed. In this thesis detailed descriptions of database and linked application for creating meeting report are given. Following the application descriptions, process of creating the meeting report is shown.

Key words: design review, activity, Petri net, database, meeting report

1. UVOD

U trenutnom razdoblju industrije 4.0 naglasak je stavljen na sveprisutnu dostupnost informacija. U skladu s time, očigledan je eksponencijalan porast kompleksnosti ne samo proizvoda nego i popratne računalne i logističke podrške. U razvoj modernih proizvoda uključeni su brojni timovi inženjera, koji su u posljednje vrijeme i geografski odijeljeni, te je jedan od najutjecajnijih faktora koji utječu na proteklo vrijeme do izlaska na tržište (eng. Time-to-market), dostupnost točno određenih informacija i podataka točno određenoj osobi u određenom trenutku. Iako ova konstatacija djeluje naizgled poprilično banalno, da bi se postigli takvi uvjeti u timskom razvoju proizvoda potreban je veliki napor i veliko znanje pri samoj organizaciji timskog razvoja proizvoda.

Težnja prilikom razvoja novog proizvoda je maksimiziranje korištenja modularnosti pojedinih komponenti, što dovodi do uske međuovisnosti velikog broja komponenti, te je potrebno prilikom izmjene pojedine komponente voditi računa o posljedičnim izmjenama na odgovarajućim komponentama koje su u sprezi s inicijalno odabranom komponentom. Moderni PLM (Product Lifecycle Management) sustavi nisu izričito fokusirani na komunikaciju i koordinaciju između timova, te je poprilično otežano adekvatno upravljanje spregnutim parametrima.

Prilikom razvoja proizvoda pojedini članovi tima generiraju veliku količinu informacija, no najvažnije odluke i konstrukcijski parametri definiraju se na timskim sastancima. Uobičajena praksa na sastancima projektnih timova je da se odabere adekvatna osoba koja pokušava u najkraćem mogućem obliku zapisati maksimalnu količinu informacija koja je generirana tokom sastanka. Takav pristup u konačnici generira nepotpune informacije koje je naknadno potrebno dopunjavati. Cilj ovog rada je ubrzanje zapisa informacija sa sastanaka projektnih timova koje se postiže formalizacijom pojedinih zapisa, što bi u konačnici moglo dovesti do implementacije softverske automatizacije koja će na osnovu svojih logičkih procesa pravilno odlučiti koji članovi tima dobivaju koje informacije u adekvatnom razdoblju razvojnog procesa. Istraživanja u ovom području naizgled nemaju velik utjecaj na sam razvoj proizvoda, no potencijalna ušteda vremena na obradi informacija skupljenih sa timskih sastanaka, može se posljedično transformirati u pojeftinjenje razvojnog procesa, što u konačnici dovodi do jeftinijeg konačnog proizvoda, koji je prihvatljiviji nama kao kupcima proizvoda.

2. ANALIZA POSTOJEĆIH ISTRAŽIVANJA O UPRAVLJANJU PODATCIMA GENERIRANIH TIJEKOM I NAKON SASTANAKA PROJEKTNIH TIMOVA

U svrhu kvalitetnijeg strukturiranja i formaliziranja podataka generiranih na projektnim sastancima proučena je literatura navedena u tablici 1. Na osnovu empirijskih podataka skupljenih sa više različitih sastanaka detaljno su opisane pojedine faze timskih sastanaka, te uobičajene aktivnosti koje se provode prije, tokom i nakon sastanka. Na temelju navedenih članaka određena je hijerarhijska struktura mogućih odabira aktivnosti tokom sastanka, te su odabrani odgovarajući atributi koji se zapisuju u samom zapisniku sastanka.

Tablica 1. Pregled postojećih istraživanja u području analize razvojnih sastanaka

Autori	Naziv znanstvenog članka	Kratak opis
Romano, N.C. Jr., Nunamaker, J. F. Jr.	*Meeting analysis: Findings from research and practice*	Emprijski podaci vezani uz projektne sastanke
Olson, G. M., Olson, J. S., Carter, R. M., Storrøsten, M.	*Small group design meetings: An analysis of collaboration*	Struktura sastanka projektnih timova, Kolaborativni razvoj novog proizvoda
Huet,G., Culley, S. J., McMahon, C. A., Fortin, C.	*Making sense of engineering design review activities*	Strukturiranje podataka generiranih na sastancima
Sim, S. K., Duffy, A. H. B.	*Towards an ontology of generic design activities*	Strukturiranje aktivnosti razvojnog procesa
Radošević, D.	*Petri mreže i njihov odnos prema drugim poznatim mrežnim tehnikama*	Osnove Petri mreža
Jensen, K., Kristensen, L M., Wells, L.	*Coloured Petri Nets and CPN Tools for Modelling and Validation of Concurrent Systems*	Obojene Petri mreže

2.1. Sastanak projektnog tima

Da bismo pravilno shvatili prednosti i mane sastanaka projektnih timova potrebno je adekvatno definirati sam pojam sastanka projektnog tima. Iz više različitih izvora slijedi da je timski sastanak planirana ili slučajna fokusirana interakcija kognitivne pozornosti, gdje se više osoba složi sastati u svrhu ostvarenja zajedničkog cilja, bilo na istom mjestu u isto vrijeme ili na više različitih lokacija i u različitim trenutcima [1]. Ukoliko se pozornije prouči navedena definicija vidljivo je da se unutar same definicije nalazi više dimenzija sastanaka prikazanih u nastavku:

- Fokusirana interakcija
- Grupe ljudi
- Zajednički cilj
- Razina formalnosti
- Vremenska i fizička disperzija

Opisane dimenzije pojedinog sastanka u konačnici definiraju sam sastanak, te je vidljivo da definicija pokriva sve vrste sastanaka u rasponu od službenih sastanaka vodećih ljudi tvrtke, ležernih razgovora u hodnicima, do internetskih sastanaka s velikom količinom ljudi različitih struka raspoređenih na više lokacija. Djelovanjem na pojedine dimenzije sastanaka može se drastično promijeniti kvaliteta samog sastanka, te je potrebno osigurati odgovarajuću podršku za svaku pojedinu dimenziju sastanka. Prikazani koncept obuhvaća informacije koje ljudi međusobno dijele u svrhu rješavanja problema pojedinca ili ostvarenja zajedničkog cilja [1].

Neki od najčešćih razloga sastanaka u postotnom prikazu prikazani su u tablici 2 [1]. Ukoliko se pozornije promotri tablica 2. vidljivo je da su najučestaliji razlozi sastanaka vrlo složeni grupni procesi za koji je potreban konsenzus većeg broja ljudi unutar skupine [1].

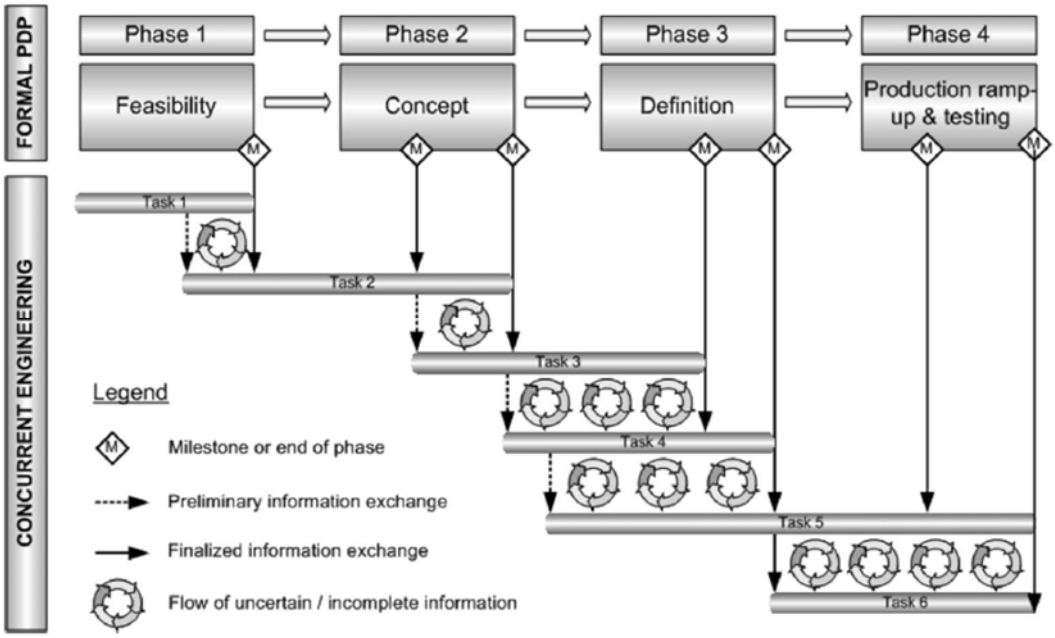
Tablica 2. Postotni prikaz učestalosti pojedinih razloga organiziranja sastanka

%	Razlog sastanka
29	Rješavanje konflikata
26	Donošenje zajedničkog suda ili odluke
11	Rješavanje potencijalnih problema
11	Osiguranje razumijevanja svih članova grupe
5	Poboljšanje međusobne komunikacije zaposlenika
4	Ostvarenje podrške za pojedine ideje

4	Istraživanje novih ideja i koncepata
2	Izlaganje izvješća
2	Predstavljanje projekta ili sustava

Kao što je i prethodno opisano, inženjerski projektni sastanci, koji se odvijaju u prethodno definiranim trenutcima razvojnog procesa, su temeljni elementi za evaluaciju i kontrolu svih inženjerskih procesa. Takvi sastanci omogućuju pojedincima da se uključe u cjelokupan složeni proces razvoja finalnog proizvoda. Podatci skupljeni tokom samih sastanaka naizgled nemaju poprilično velik značaj, no najbitnije odluke o smjeru razvoja, te pojedina specifična znanja su eksplicitno prikazana tokom tih sastanaka.

Sam razvojni proces možemo podijeliti na pojedine faze, koje su odijeljene odgovarajućim kontrolnim točkama na kojima se vrši kontrola razvojnog procesa i donose odluke za sljedeću fazu[2]. Detaljni shematski prikaz razvojnog procesa prikazan je na sljedećoj slici.



Slika 1. Shematski prikaz razvoja procesa s kontrolnim točkama [2]

Ukoliko se pozornije promotri slika 1 vidljivo je da inženjerski sastanci održavaju na kraju svake faze razvojnog procesa, te da se tokom tih sastanaka generiraju nove informacije koje služe kao ulazne informacije u sljedećoj fazi procesa. Razvojni proces je kontinuiran proces, no uvođenjem kontrolnih točaka postiže se diskretizacija procesa, što omogućuje bolju kontrolu pojedine faze tokom cijelog procesa. Kao što je i vidljivo na slici 1 tokom pojedine faze moguće

je organizirati više sastanaka u svrhu bolje kontrole troškova i same kvalitete procesa razvoja proizvoda, no nekontroliranim povećanjem količina sastanaka unutar jedne faze može dovesti do pada efikasnosti sastanaka i dolazi do zasićenja što u konačnici drastično utječe na kvalitetu rada pojedinih zaposlenika. Također, potrebno je i adekvatno odrediti trenutak sastanka, jer ukoliko se sastanak održi prerano postoji mogućnost pojavljivanja prevelike količine nepotpunih informacija, te odluke donesene na osnovu takvog skupa informacija su potencijalno neoptimalne. Ukoliko se sastanak održi prekasno, postoji mogućnost donošenja određenih odluka koje nisu potvrđene od svih osoba, što ukoliko je potrebno naknadno mijenjati, dovodi do nepotrebnog kašnjenja koje se moglo izbjeći. Kao što je i prikazano na slici 1. unutar pojedine faze iterativnim procesom se kombinacijom nepotpunih informacija dolazi do željene potpune informacije. Iz tog razloga se razvojni sastanci i smatraju jedinstvenom točkom sinkronizacije nepotpunih informacija. Da bi se adekvatno mogle koristiti informacije stvorene tokom sastanka, potrebno ih je zapisati na način koji omogućuje maksimalnom broju zaposlenika jednaku mogućnost pristupa toj informaciji [3].

Ovisno u kojoj fazi razvojnog procesa se trenutno nalazimo prema [3] slijede vrste sastanaka koje se održavaju u većini tvrtki:

- **Pregled zahtjeva** (eng. Requirements review). Na ovom sastanku potrebno je osigurati potpunu i jasnu definiciju inicijalnih zahtjeva i ograničenja. Pregled zahtjeva također se izvodi i tokom preliminarnog pregleda.
- **Pregled razvoja velikih sustava** (eng. Systems design review). Ukoliko je potrebno razvijati velik sustav s velikom količinom konfiguracija, na ovom sastanku osigurava se pravilna alokacija zahtjeva pojedinim konfiguracijama sustava.
- **Preliminarni pregled razvoja** (eng. Preliminary design review). Evaluacija koncepata na osnovu izvedivosti, tehničke smislenosti i općenitom usklađenosti s prethodno definiranim zahtjevima i ograničenjima. Opisuju se pretpostavke i proračuni na osnovu kojih su donesene odgovarajuće odluke, te ukoliko je moguće diskutira se o preliminarnim prototipovima ili skicama koncepata. Također, naglašavaju se potencijalni rizici koji se mogu javiti, te se pokušava kreirati plan za izbjegavanje pojedinih rizika.
- **Kritičan pregled razvoja** (eng. Critical design review). Sastanak koji se održava nakon faze detaljiranja proizvoda i prije početka izrade fizičkog prototipa proizvoda ili modela koji služi za uhodavanje proizvodnje. Provjerava se zadovoljava li proizvod sve detaljno

definirane uvjete. Po strukturi je vrlo sličan preliminarnom pregledu razvoja, no dodatno se raspravlja o problematici procesa proizvodnje.

- **Pregled spremnosti za testiranje** (eng. Test readiness review). Na ovom sastanku određuju se planovi za testiranje prototipova u skladu s postavljenim zahtjevima. Potrebno je osigurati da su konačni planovi za testiranje proizvoda pouzdani, ponovljivi i da pokrivaju sve postavljene zahtjeve.
- **Konačan pregled razvoja (**eng. Final design review**).** Nakon testiranja prototipova raspravlja se o problemima nastalima tokom testiranja, te potencijalnim rješenjima nastalih problema. Osigurava se da su sve nužne promjene proizvoda donesene u skladu s postavljenim zahtjevima, te eventualno nastalim kašnjenjima i troškovima koje navedene promjene nose sa sobom.
- **Pregled spremnosti za proizvodnju (**eng. Production readiness review**).** Osiguravanje da su sve značajke proizvoda u skladu s svim zahtjevima proizvodnosti. Ovakav tip sastanaka održava se više puta tokom procesa razvoja, te se ovisno u kojoj se fazi proces razvoja nalazi raspravlja o detaljnim zahtjevima proizvodnog procesa.
- **Ad hoc sastanak.** U bilo kojem trenutku razvojnog procesa mogu se pojaviti problemi koji mogu drastično promijeniti smjer razvoja proizvoda, te je potrebno u tim trenutcima održati sastanke na kojima se donose kritične odluke za usmjeravanja razvojnog procesa.

Jedan od ključnih faktora za održavanje kvalitetnog sastanka je odabir prisutnih. U svrhu maksimiziranja efikasnosti, preporučen je broj prisutnih u rasponu od 5 do 10 ljudi. Prilikom sastanaka tokom razvoj velikih sustava preporučuje se da samo nekoliko ključnih ljudi prisustvuje cijelom sastanku, dok se ostatak ljudi mijenja ovisno o njihovom djelokrugu rada te trenutnoj temi rasprave. Nerijetko se pozvani zaposlenici ne mogu odazvati na sastanak, te je potrebno predvidjeti alternativne zaposlenike koji mogu nadomjestiti inicijalno pozvane. Karakteristike ključnih ljudi na sastanku slijede:

- **Predsjedavajući.** Odgovoran za koordinaciju priprema za sastanak i nadzor toka sastanka, te nadzor aktivnosti koje se održavaju nakon sastanka. U nekim tvrtkama predsjedavajući je iskusan inženjer s temeljnim razumijevanjem razvojnog procesa, no bitno je da nije uključen u razvojne aktivnosti što mu omogućuje objektivan pristup svakoj temi sastanka. Kupac proizvoda može također postaviti dodatnog predsjedavajućeg čija je uloga nadzor svih aktivnosti na sastanku s gledišta kupca.

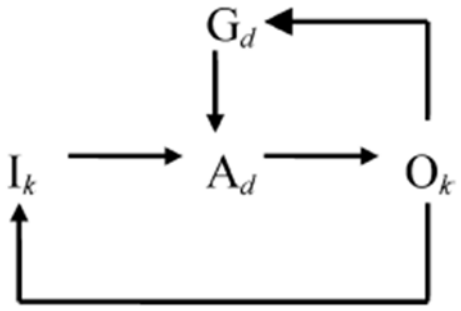
- **Razvojni tim.** Sudjeluje u sastanku u raspravama vezanim uz sam razvoj i razvojni proces u koji su uključeni. U pravilu nije praktično pozvati sve članove razvojnih timova u velikim razvojnim projektima, te su iz tog razloga cijelom sastanku prisutni samo vođe timova, dok se u specifičnim temama pozivaju dodatni zaposlenici.
- **Stručnjaci za specifične teme.** U svakom individualnom razvojnom procesu izbor neovisnih stručnjaka koje se poziva na sastanak ovisi o izboru tema. Odabrani stručnjaci nisu direktno povezani s projektom, no posjeduju znanja koja uključeni timovi ne posjeduju u toj mjeri
- **Kupac.** Pri razvoju proizvoda za specifičnog kupca, nerijetko predstavnici kupca postaju članovi sastanaka. Također, ukoliko u razvojnim timovima nema članova s potrebnim specifičnim znanjima, kupac može pozvati stručnjake za pojedine probleme iz svoje tvrtke na sastanak.

2.2. Ontologija aktivnosti u razvojnom procesu [4]

Trenutno u znanstvenom svijetu postoje brojni modeli i teorije koje fenomenološki opisuju razvojni proces, no gledano na svjetskoj razini ne postoji konsenzus u izboru odgovarajućeg modela koji bi najbolje opisivao stanje u industriji. Neki istraživači razvojnog procesa predlažu teorije koje se zasnivaju na kognitivnoj znanosti, no trenutno, navedene teorije nisu u potpunosti primjenjive zbog manjka teorijskog razumijevanja kognitivnih sposobnosti korištenih tokom razvoja proizvoda. Jedan od prijedloga koji autor predlaže je uvođenje teorije razine znanja (eng. knowledge level theory), gdje se razvojni proces smatra skupom konceptualno različitih faza ili aktivnosti koje transformiraju konstrukciju od skupa zahtjeva do konačnog konstrukcijskog rješenja. Iz tog razloga definirana je ontologija aktivnosti koja daje uvid u veze između pojedinih aktivnosti koje se mogu pojaviti tokom razvojnog procesa. Autor smatra razvojne aktivnosti kao razina apstrakcije na koju se racionalna kognitivna aktivnosti može dekomponirati, te slijedi da se općenita aktivnost u razvoju proizvoda može prikazati kao proces promjene znanja tokom odabrane aktivnosti.

Definicijom aktivnosti na razini znanja, aktivnost se smatra svakom radnjom koja generira neko novo znanje na osnovu ulaznog znanja, u svrhu postizanja nekog prethodno definiranog cilja. Na ulazno znanje utječe percepcija inženjera trenutnog konteksta razvoja, te njegovo eksplicitno i prešutno znanje u odgovarajućoj domeni. Izlazno znanje može biti simbolička struktura koja inženjeru predstavlja rješenje ili djelomično rješenje početnih zahtjeva aktivnosti. Izlaz aktivnosti u pravilu pokreće novu aktivnost, te se na taj način nastavlja

razvojni proces. Također, moguće je da neka aktivnost pokreće nekoliko novih aktivnosti izvedenih istovremeno ili naknadno. Razvojni proces se prekida u trenutku kada su zadovoljeni svi postavljeni zahtjevi i ograničenja. Osnovni elementi aktivnosti prikazani su na sljedećoj slici.



Slika 2. Formalizacija aktivnosti u razvoju proizvoda [4]

Prema slici 2. osnovni elementi aktivnosti u razvojnom procesu su:

- Postojeće znanje kao ulazno znanje, I_k
- Aktivnost u razvojnom procesu, A_d
- Cilj razvojnog procesa, G_d
- Izlazno znanje, O_k

Promotri li se slika 2. vidljivo je da znanje inženjera i postavljeni zahtjevi služe kao ulazni parametri aktivnosti koja generira novo znanje, sa svrhom postizanja konačnog cilja koji zadovoljava postavljene zahtjeve.

Ulazno znanje se može kategorizirati u eksplicitno i tacit znanje. Prema autoru članka tacit znanje je svo osobno znanje inženjera koje je vrlo teško formalizirati i zapisati. Postoje tri općenite vrste tacit znanja, no u cjelini, tacit znanjem se može smatrati svo znanje koje je inženjer iskustveno generirao:

- Deklarativno znanje
- Proceduralno znanje
- Neformalno znanje

Eksplicitno znanje je svo znanje koje se može prenijeti formalnim i sistematiziranim kanalima, kao što su znanstveno i tehnološko znanje.

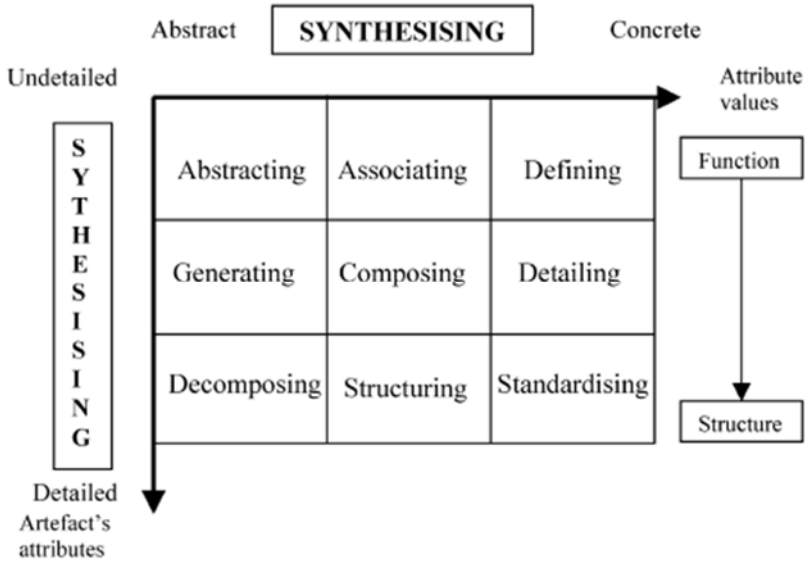
Cilj razvojnog procesa u pravilu se zaključuje na temelju početnih zahtjeva koje je potrebno zadovoljiti. U svakoj fazi razvojnog procesa određuju se novi ciljevi, što može dovesti

do poprilično složene hijerarhijske strukture ciljeva u svrhu ostvarivanja optimalnog konačnog proizvoda. Na primjer, cilj aktivnosti donošenja odluke može biti odabir optimalnog koncepta proizvoda. No, da bi se odluka mogla donijeti, potrebno je provesti ocjenjivanje koncepata na osnovu definiranog skupa kriterija, što je zasebna aktivnost s zasebnim ciljem.

Izlazno znanje aktivnosti u razvojnom procesu ovisi o izboru odgovarajuće aktivnosti na temelju ulaznog znanja, u svrhu ostvarivanja konačnog cilja, konačnog proizvoda. Stoga, izlazno znanje pojedine aktivnosti doprinosi cjelokupnoj promjeni znanja, koja omogućuje inženjerima približavanje konačnom cilju razvoja proizvoda.

2.2.1. Klasifikacija aktivnosti definiranja

Vrhuncem aktivnosti definiranja smatra se sintetiziranje, iz razloga što se sintetiziranje sastoji od skupa aktivnosti koje uključuju pretraživanje i istraživanje parcijalnih rješenja s konačnim ciljem generiranja zadovoljavajućeg proizvoda. Konfiguracijom postojećih entiteta razvoja proizvoda postiže se izvediv sustav postojećih struktura koje zadovoljavaju početne zahtjeve. Taksonomija aktivnosti definiranja prikazana je na slici 3. gdje je vidljivo da se sintetiziranjem postiže prijelaz iz apstraktnog područja u konkretno, te od općenite konstrukcije do konačnog detaljiranog proizvoda. Također, na osnovu prikazane taksonomije moguće je vrlo lako zaključiti koje aktivnosti se provode u kojoj fazi razvojnog procesa. U ranim faza razvoja proizvoda apstrahiranje je aktivnost koja se najčešće provodi, dok je standardiziranje aktivnost koja se provodi u završnim fazama razvoja proizvoda. DFX (Design for X) metoda jedna je od najkorištenijih metoda unutar aktivnosti sintetiziranja, gdje je fokus na razvoju proizvoda koji u najvećoj mogućoj mjeri ispunjava početne postavljene uvjete.



Slika 3. Taksonomija aktivnosti definiranja [4]

2.2.1.1. Apstrahiranje

Apstrahiranje kao aktivnost smanjuje važnost specifičnih informacija, te se fokusira na opće. Apstrakcija određenog problema mora obuhvatiti sve bitne odnose između pojedinih atributa proizvoda, što omogućuje inženjeru da donosi pravilne odluke u ranim fazama razvojnog procesa. Korištenjem apstrakcija i modela olakšavaju se početne odluke gdje nije bitno poznavati točne veze između pojedinih komponenti i njihovih funkcija. U tablici 3. opisani su elementi aktivnosti apstrahiranja.

Tablica 3. Apstrahiranje

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Smanjenje kompleksnosti objekta u razvojnom procesu	- Tipovi apstrakcije - Znanje o odgovarajućoj domeni	- Odgovarajuće apstrakcije objekta (skice, sheme)	- Apstrakcije koje prikazuju korisne veze između pojedinih objekata

2.2.1.2. Generiranje

Generiranje kao aktivnost ima svrhu zadovoljavanja funkcijskih zahtjeva koji su često izvedeni iz zahtjeva kupca. Generiranje je složena aktivnost koja se sastoji od aktivnosti generiranja novih koncepata, komponiranja i kombiniranja postojećih konstrukcijskih rješenja. Za svaku pojedinu funkciju proizvoda generira se koncept koji zadovoljava parcijalnu funkciju, te nakon kombiniranja dobivenih koncepata generira se konačan koncept koji zadovoljava sve

zahtjeve. S obzirom da se generiranje koncepata provodi u ranim fazama razvojnog procesa kada još ne postoje potpune informacije, dobivene informacije su kvalitativne prirode. Jedan od zahtjeva za opisanu aktivnost je provođenje aktivnosti dekompozicije gdje je potrebno osigurati funkcijsku dekompoziciju proizvoda.

Tablica 4. Generiranje

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Generiranje rješenja koje zadovoljava hijerarhiju funkcija	- Znanje o hijerarhijskoj strukturi funkcija	- Povezivanje utjecaja rješenja na konačnu fukciju komponente	- Mapiranje hijerarhije fukcije u ovisnosti o rješenju

2.2.1.3. Dekomponiranje

Dekomponiranje je aktivnost pri kojoj se složeni problem ili objekt rastavljaju na parcijalne probleme niže kompleksnosti koje je jednostavnije riješiti. Stoga, dekomponiranje se može smatrati aktivnošću koja dijeli određeni problem na više jednostavnih entiteta. Postoji više vrsta dekompozicije, gdje je strukturna dekompozicija rastavljanje proizvoda na sklopove i podsklopove, dok se modularnom dekompozicijom omogućuje opisivanje veza između neovisnih sklopova unutar nekog sustava.

Tablica 5. Dekomponiranje

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Određivanje strukture proizvoda	- Postojeća rješenja s sličnim zahtjevima - Znanje o zahtjevima pojedinih funkcija	- Specifična taksonomija konstrukcije s obzirom na strukturu (sustav - podsustav)	- Određivanje konačne hijerarhijske strukture proizvoda - Znanje o cjelokupnoj funkciji proizvoda na temelju podfunkcija

2.2.1.4. Povezivanje

Inženjeri povezuju ideje i koncepte pomoću kognitivne aktivnosti povezivanja. Opisanom aktivnošću postiže se grupiranje međusobno povezanih ideja, što u konačnici omogućuje drukčiji pogled na zadani problem. Stoga je cilj opisane aktivnosti stvaranje novih ideja na osnovu povezivanja koncepata s postojećim rješenjima sličnih problema. Jedna od najčešće korištenih metoda za poticanje stvaranja novih ideja je tzv. Brainstorming metoda, gdje se

potiče stvaranje maksimalne količine različitih koncepata koji su na neki način povezani, što omogućuje kvalitetnije naknadno ocjenjivanje i izbor optimalnog koncepta.

Tablica 6. Povezivanje

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Generiranje novih ideja / koncepata	- Znanje o odgovarajućoj domeni - Metode za generiranje ideja (brainstorming)	- Ideje, koncepti i njihove veze	- Znanje o novim načinima povezivanja ideja i koncepata

2.2.1.5. Komponiranje

Kao jedna od bitnih aktivnosti u fazi koncipiranja proizvoda, komponiranje služi za povezivanje parcijalnih koncepata i modula. Korištenjem odgovarajućih tabličnih metoda za povezivanje parcijalnih rješenja postiže se kvalitetan pregled potencijalnih kombinacija, što u konačnici omogućuje olakšani izbor optimalnog rješenja

Tablica 7. Komponiranje

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Kombiniranje ideja i koncepata koji zadovoljavaju cjelokupnu funkciju	- Znanje o odgovarajućoj domeni - Tablice kombinacija	- Koncepti ili moduli koji zadovoljavaju cjelokupnu funkciju	- Modularna arhitektura proizvoda

2.2.1.6. Strukturiranje

Postojeći proizvod može se opisati s obzirom na funkcionalne i fizičke elemente, gdje su funkcionalni elementi opisani u shematskoj formi prije implementacije u specifičnim tehnologijama, dok se fizički elementi koji služe za ostvarivanje određene funkcije spajaju u module. S obzirom da navedeni moduli mogu na različite načine biti u vezi s ostalima, definirane su dvije vrste interakcije: osnovna koja je shematski opisana, te definira raspored elemenata u modulima, te incidentna koja definira geometrijsku međuovisnost modula

Tablica 8. Strukturiranje

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Optimalna arhitektura proizvoda koja minimizira kompleksnost koordinacije potrebne za razvoj proizvoda	- Znanje o vezama između komponenti - Znanje o specifikacija pojedinih komponenti	- Znanje o arhitekturi proizvoda, te vezama između pojedinih komponenti	- Grupiranje znanja o vezama između komponenti unutar sustava

2.2.1.7. Detaljiranje

Detaljiranje je aktivnost čija je svrha povišenja specifičnosti odgovarajućeg konstrukcijskog rješenja. Završetkom navedene aktivnosti postiže se konstrukcijsko rješenje spremno za proizvodnju na način da su definirani svi proizvodni parametri (dimenzije, tolerancije, materijal, svojstva površine), te se osigurava da postoje sva potrebna tehnička dokumentacija za samu proizvodnju (nacrti, tehnički opisi). Bitan čimbenik u realizaciji samog proizvoda je kvalitetno provedeno detaljiranje, gdje se može uvidjeti fenomen u kojem loš koncept ma kako kvalitetno detaljiranje ne može spasiti, dok odličan koncept može loše detaljiranje upropastiti.

Tablica 9. Detaljiranje

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Eliminiranje svake dvosmislenosti i nesigurnosti vezanih uz konstrukciju	- Znanje o odgovarajućoj domeni - Zahtjevi proizvoda	- Detaljni crteži - Dokumentacija o proizvodu	- Znanje o konačnoj konstrukciji prikazana crtežima i dokumentacijom

2.2.1.8. Standardiziranje

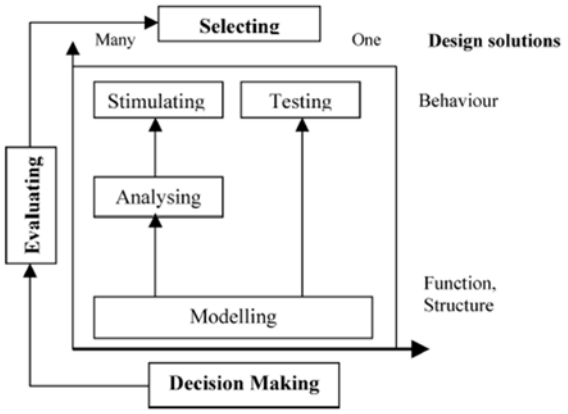
Standardiziranje je posebna vrsta aktivnosti koja je specifična za fazu detaljiranja proizvoda, te se može smatrati kao jedna od presudnih strategija za smanjenje troškova proizvoda. Cilj inženjera je ovom aktivnošću maksimizirati korištenje standardnih komponenti u svrhu postizanja uniformnosti unutar konstrukcijskog rješenja. Standardiziranje se može smatrati podaktivnošću detaljiranja s obzirom da joj je cilj identičan detaljiranju, no naglasak se stavlja na uniformnost sa svrhom snižavanja troškova proizvodnje. Standardiziranje u pravilu umanjuje kompleksnost proizvoda.

Tablica 10. Standardiziranje

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Smanjenje kompleksnosti s obzirom na broj komponenti u konstrukciji	- Znanje o detaljima konstrukcije - Znanje o standardnim komponentama	- Odabran skup standardnih komponenata - Kriterij standardizacije	- Znanje o prirodi standardizacije i razlogu izbora

2.2.2. Klasifikacija aktivnosti ocjenjivanja

Jedan od ciljeva aktivnosti ocjenjivanja je smanjenje kompleksnosti prostora mogućih rješenja. Provođenjem aktivnosti kao što su ocjenjivanje ili donošenje odluka neizvediva ili neoptimalna rješenja se odbacuju, što smanjuje prostor preostalih mogućih konstrukcijskih rješenja. Na sljedećoj slici prikazan je shematski prikaz aktivnosti ocjenjivanja gdje su vidljive veze između pojedinih aktivnosti.



Slika 4. Shematski prikaz aktivnosti ocjenjivanja [4]

Donošenje odluka je skup aktivnosti koji obuhvaća ocjenjivanje predstavljenih alternativa i odabir odgovarajuće s obzirom na postavljene zahtjeve. Modeliranje i simuliranje su aktivnosti koje predstavljaju rješenja problema s obzirom na njihove funkcije i strukture. Modeliranjem i simuliranjem omogućuje se vjerodostojnija analiza predstavljenih koncepata.

2.2.2.1. Donošenje odluka

Donošenje odluka kao aktivnost svodi se na izbor pojedinih mogućnosti unutar postojećeg skupa mogućnosti. Postoje dvije vrste donošenja odluka, gdje su procesno orijentirane odluke one odluke koje određuju trenutak napredovanja razvojnog procesa, a proizvodno orijentirane odluke su one odluke koje se tiču samog proizvoda tokom nekog perioda razvoja proizvoda. Tokom donošenja odluke, inženjer identificira potencijalne probleme, određuje kriterije

ocjenjivanja, te nakon rasprave eksplicitno definira izbor odgovarajuće alternative koja maksimalno zadovoljava kriterije ocjenjivanja. Posebnu pozornost potrebno je obratiti na odluke donesene u ranim fazama razvojnog procesa iz razloga što se uvođenjem naknadnih promjena parametara definiranih u ranim fazama razvoja donose katastrofalni dodatni troškovi koji u konačnici i mogu zaustaviti proces razvoja proizvoda.

Tablica 11. Donošenje odluka

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Izbor najbolje alternative iz skupa na osnovu postavljenih kriterija	- Znanje o zahtjevima proizvoda - Znanje o alternativama rješenja - Znanje o odgovarajućim metodama analiziranja	- Znanje o danim kriterijima - Znanje o donesenim temeljnim odlukama - Znanje o izboru odgovarajuće metode analize	- Znanje o razlozima izbora odgovarajuće alternative

2.2.2.2. Ocjenjivanje

Ocjenjivanje kao aktivnost smatra se svako procjenjivanje da li određeni koncept zadovoljava postavljene zahtjeve u definiranim specifikacijama. Ova aktivnost je također složena aktivnost koja se sastoji od aktivnosti identificiranja, mjerenja i uspoređivanja. Razlozi ocjenjivanja uključuju sprečavanje kvarova tokom eksploatacije, usporedbu predloženih rješenja sa svrhom izbora najboljeg, te usporedbu predloženih rješenja s „idealnim“ rješenjem. U nekim slučajevima ocjenjivanje je moguće tek nakon provedenih ispitivanja pojedinih komponenti. Da bi se aktivnost ocjenjivanja mogla provesti potrebno je definirati kriterije, izjave o zadovoljavajućoj sposobnosti ispunjavanja željenih svojstava. Neki kriteriji su definirani vrijednostima i matematičkim odnosima, no neki mogu biti subjektivni, te ovise o ljudskim odlukama.

Tablica 12. Ocjenjivanje

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Određivanje mjere kvalitete ili iznosa parametra s obzirom na definirane kriterije	- Znanje o specifikacijama proizvoda i konačnim ciljevima - Znanje o odgovarajućim metodama ocjenjivanja	- Znanje o ponašanju sposobnosti objekta da zadovolji definirane kriterije	- Znanje o razlozima procjene kvalitete pojedinih konstrukcijskih rješenja

2.2.2.3. Odabir

Odabir je aktivnost koja uključuje izbor objekta u procesu razvoja proizvoda koji zadovoljava zahtjeve proizvoda iz skupa specifičnih alternativa. Pojam objekta korišten je sa svrhom generaliziranja, jer u izbor mogu ulaziti razni objekti kao što su izbor principa rada koncepta, vrste materijala, komponente iz kataloga, funkcijskog modula ili konačnog cilja u razvojnom procesu. Odabir se vrši u svim fazama razvojnog procesa.

Tablica 13. Odabir

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Izbor izvedivog rješenja ili aktivnosti iz skupa alternativa	- Znanje o atributima alternativa - Znanje o zahtjevima koji definiraju atribute - Znanje o kriterijima izbora	- Znanje o odabranom objektu - Znanje o korištenom kriteriju	- Znanje o optimalnom izboru specifičnog konstrukcijskog rješenja ili procesa

2.2.2.4. Analiziranje

Analiziranje je aktivnost koja uključuje korištenje modela poznatog fizičkog fenomena u svrhu odgovora na pitanja u vezi određenog proizvoda i njegovog ponašanja. Prema autoru, analizom se predviđa ponašanje analizirane komponente proizvoda. Analiziranje se sastoji od međusobno povezanih aktivnosti, koje su modeliranje, simuliranje i ocjenjivanje. Kao primjer može se uzeti FEM model određene komponente, gdje se predviđa ponašanje komponente pod određenim opterećenjem. Analiziranje se može podijeliti u tri kategorije: kvalitativne tehnike, aproksimativne tehnike te detaljne tehnike. Kvalitativne tehnike analiziranja predviđaju ponašanje nekog fizičkog sustava, na temelju temeljnih inženjerskih zakona. Aproksimativnim tehnikama detaljnije se analiziraju informacije dobivene kvalitativnim tehnikama, korištenjem formula i korelacija u svrhu ostvarivanja kvantitativnih procjena ponašanja. Detaljne tehnike su obično vezane s preciznim numeričkim tehnikama koje daju precizna kvantitativna predviđanja ponašanja.

Tablica 14. Analiziranje

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Predviđanje ponašanja	- Znanje vezano uz fizičke fenomene i teorije	- Znanje o ponašanju konstrukcijskog rješenja	- Znanje da analizirano konstrukcijsko

konstrukcijskog rješenja	- Znanje o strukturi konstrukcijskog rješenja - Znanje o radnom okruženju rješenja - Znanje o metodama analize fizičkih fenomena		rješenje se ponaša na previđeni način u skladu postavljenih kriterija

2.2.2.5. Modeliranje

Modeliranje je jedna od aktivnosti koja se pojavljuje u svim fazama razvojnog procesa. Modeliranjem se predstavljaju željeni aspekti proizvoda na koje je potrebno usmjeriti pažnju inženjera. Model u nastajanju je kognitivni zapis koji se naknadnim aktivnostima transformira u stvarni entitet. Postoje mnoge vrste modela, kao što su funkcionalni modeli koji opisuju parcijalne funkcije modela, matematički modeli, koji pomoću poznatih formula definiraju fizikalno ponašanje odabranog konstrukcijskog rješenja, strukturalni modeli itd. Glavni cilj modeliranja je apstraktna reprezentacija konstrukcijskog rješenja koje služi kao način komunikacije između inženjera te kontroli i predviđanju ponašanja konačnog proizvoda. Modeliranje kao aktivnost može se provoditi tek nakon što je definirana forma i struktura proizvoda.

Tablica 15. Modeliranje

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Prikladan model konstrukcijskog rješenja ovisno o zahtijevanoj perspektivi	- Znanje o prikladnim tehnikama modeliranja za pojedine vrste analiza	- Znanje o prikladnim modelima	- Detaljniji opis konstrukcijskog rješenja koje ovisi o tipu provedene analize

2.2.2.6. Simuliranje

Simuliranjem je aktivnost koja formira sliku ili imitaciju ponašanja pojedinih svojstava konstrukcijskog rješenja. Provođenjem navedene aktivnosti dolazi se do očekivanog ponašanja u stvarnoj eksploataciji proizvoda. Postoje razne vrste modela koji služe za simuliranje kao što su matematički modeli, te replike stvarnih proizvoda korištenih u stvarnim situacijama. Obično, dobiveni skupovi ponašanja ocjenjuju se prema početno postavljenim uvjetima. Ukoliko je moguće, simulacije se izvode u periodu između sinteze i ocjene predloženih koncepata.

Tablica 16. Simuliranje

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Formiranje slike ili imitacije ponašanja i svojstava konstrukcijskog rješenja korištenjem prikladnih modela	- Znanje o postojećim prikladnim simulacijskim modelima - Znanje o početnim zahtjevima proizvoda	- Znanje o očekivanom ponašanju pod određenim testnim okolnostima	- Znanje o uvjetnim predviđanjima ponašanja proizvoda

2.2.2.7. Testiranje

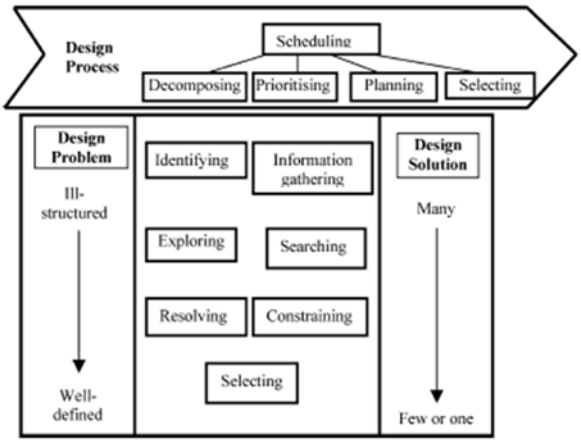
Većina proizvoda prije izlaska na tržište podliježe velikom broju ispitivanja kojima se osigurava da konačan proizvod zadovoljava sve početno postavljene uvjete. Za razliku od analiziranja, kada se ponašanje predviđa na osnovu modela i simulacija, kod testiranja ponašanje se opisuje mjerenjem odgovarajućih parametara u eksploataciji.

Tablica 17. Testiranje

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Osiguranje da je predviđeno ponašanje u skladu s stvarnim	- Znanje o početnim specifikacijama - Znanje o kriterijima ispitivanja	- Znanje o rezultatima ispitivanja	- Znanje o usklađenosti konstrukcijskog rješenja s definiranim specifikacijama

2.2.3. Klasifikacija aktivnosti upravljanja

Aktivnosti upravljanja mogu se podijeliti u dvije skupine: aktivnosti koje upravljaju napretkom zadanog problema do konačnog konstrukcijskog rješenja, te aktivnosti koje upravljaju procesom razvoja proizvoda njegovim tokom. Glavna svrha aktivnosti upravljanja je osiguranje poštivanja vremenskih rokova, te upravljanje kompleksnošću prostora mogućih konstrukcijskih rješenja. Shematski prikaz opisanih aktivnosti prikazan je na slici 5. Vidljivo je da se provođenjem određenog skupa aktivnosti loše definiran i strukturiran problem pretvara u dobro definiran, te dodatno se sužava prostor mogućih konstrukcijskih rješenja.



Slika 5. Shematski prikaz aktivnosti upravljanja razvojnim procesom [4]

2.2.3.1. Ograničavanje

Glavni cilj aktivnosti ograničavanja je smanjenje prostora pretraživanja konstrukcijskih rješenja. Ograničenja se mogu postaviti na osnovu početnih specifikacija, inženjerskih kodova, standarda , procedure konstrukcijskog ureda, te na osnovu znanja i iskustva odgovarajućih inženjera. Bitno je naglasiti da je potrebno pravilno tempirati trenutak postavljanja ograničenja, jer preranim ograničavanjem pojedinih rješenja nepotrebno se ograničava prostor rješenja, te potencijalno dobro rješenje može biti izostavljeno iz razmatranja.

Tablica 18. Ograničavanje

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Smanjenje kompleksnosti prostora mogućih konstrukcijskih rješenja	- Znanje o specifičnim zahtjevima konstrukcije - Znanje o vrstama primjenjivih ograničenja	- Znanje o specifičnim primijenjenim ograničenjima - Znanje o razlozima primjene određenih ograničenja	- Znanje o izvedivom prostoru konstrukcijskih rješenja

2.2.3.2. Istraživanje

U početnim fazama razvojnog procesa prostor mogućih rješenja loše je definiran, te je u svrhu napretka razvoja proizvoda potrebno definirati strukturu prostora, što omogućuje smanjenje broja potencijalnih rješenja. Istraživanje se svodi na dekompoziciju prostora svih mogućih rješenja u međusobno nepovezane module, uz sprečavanje opredjeljenja za neko specifično rješenje. Cilj istraživanja je strukturiranje prostora mogućih rješenja određivanjem nedostupnih informacija, te njihovim naknadnim uobličavanjem u prostor mogućih konstrukcijskih rješenja.

Tablica 19. Istraživanje

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Istraživanje prostora mogućih rješenja bez opredjeljenja za neko specifično rješenje	- Znanje o postojećim konstrukcijskim rješenjima	- Znanje o strukturi problema - Znanje koje nedostaje	- Znanje o potencijalnim rješenjima kao kandidatima konačnog rješenja - Znanje o strategijama izbora rješenja

2.2.3.3. Identificiranje

Identificiranjem se upravlja kompleksnošću problema izborom relevantnih domena potrebnog znanja za trenutno aktualni problem. Inženjeri identificiraju koje aspekte konstrukcijskog rješenja je potrebno staviti u fokus tokom trenutno aktualnog procesa razvoja proizvoda. Također se identificiraju metode konstruiranja i metodologije koje se koriste u aktivnostima analize i ocjenjivanja

Tablica 20. Identificiranje

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Označavanje relevantnog i bitnog u svrhu upravljanja kompleksnosti konstrukcijskog problema	- Znanje o domeni problema - Znanje o postojećim rješenjima - Znanje o metodama i metodologijama konstruiranja	- Znanje o bitnoj domeni - Specifičan konstrukcijski slučaj - Specifične konstrukcijske metode i metodologije	- Znanje o bitnim problemima koji utječu na konačnu konstrukciju koje vodi do bolje strategije konstruiranja

2.2.3.4. Skupljanje informacija

U svim konstrukcijskim zadatcima, informacije vezane uz odgovarajuće zadatke moraju se skupiti iz različitih izvora. Prema autoru, 30% ukupnog inženjerskog vremena se svodi na pretraživanje i pristupanje inženjerskim informacijama na određenim mjestima. Takvo vrijeme se smatra neproduktivno vrijeme, te je težnja inženjera da iskoriste znanje koje posjeduju, no takav pristup može dovesti do loših rješenja zbog manjka kvalitetnih informacija. Skupljanje informacija omogućuje inženjerima pristup novim informacijama koje su vezane uz njihov konstrukcijski problem.

Tablica 21. Skupljanje informacija

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Pružanje aktualnih znanja koje mogu unaprijediti konstrukcijsko rješenje u sljedeću fazu	- Skup znanja/informacija unutar tvrtke	- Specifično znanje vezano za konstrukcijsko rješenje	- Dodatno znanje o konstrukcijskom rješenju ili procesu

2.2.3.5. Rješavanje

Tokom razvoja proizvoda često dolazi do konfliktnih situacija, te je potrebno u svrhu razvoja samog procesa osigurati maksimalno brzo rješavanje potencijalnih konflikata,

Tablica 22. Rješavanje

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Rješavanje konfliktnih interesa, zahtjeva i pogleda	- Znanje o mogućim tipovima konflikata - Znanje o strategijama rješavanja konflikata	- Znanje o ograničenjima nastalim nakon razrješenja konflikta	- Specifično znanje o rješenju konflikta vezanog uz proizvod ili proces

2.2.3.6. Pretraživanje

Pretraživanje je vrlo slična aktivnosti istraživanja s razlikom da je prilikom pretraživanja čvrsto definiran očekivan rezultat unutar dobro definiranog prostora mogućih rješenja. Uspješno pretraživanje često je vezano s strategijama pretraživanja koje upravljaju poljima pretraživanja. Predložene su dvije vrste pretraživanja koje su: diskurzivne metode (analogije, skupljanje, zakoni sličnosti, strukturiranje, itd) te intuitivne metode (brainstorming).

Tablica 23. Pretraživanje

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Zadovoljavanje pojedinih zahtjeva konstrukcijskog rješenja ili problema	- Znanje o strategijama pretraživanja - Znanje o izvorima informacija	- Rezultati pretraživanja	- Znanje koje pridonosi definiciji konstrukcijskog rješenja

2.2.3.7. Prioritiziranje

Prioritiziranje je aktivnost kojom se određuje redoslijed ciljeva na osnovu njihove važnosti i na osnovu opće strategije u razvoju procesa

Tablica 24. Prioritiziranje

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Fokusiranje na bitne ciljeve koji imaju utjecaj na aktivnosti u kasnijim fazama razvojnog procesa	- Znanje o relativnim važnostima ciljeva - Znanje o zahtjevima pojedinog podzadataka	- Znanje o redoslijedu ciljeva na temelju njihove važnosti	- Znanje o specifičnom redoslijedu zadataka i podazadataka unutar nekog procesa

2.2.3.8. Planiranje

Planiranjem se postiže organizacija resursa potrebnih za razvoj proizvoda na osnovu redoslijeda zadataka, rasporeda članova tima na pojedini zadatak i na osnovu aktivnosti koje su vezane uz proizvodnju.

Tablica 25. Planiranje

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Minimiziranje vremena do izlaska na tržište strukturiranjem zadatka u razvoju proizvoda	- Znanje o zadatcima razvoja, resursima - Znanje o metodama i algoritmima planiranja	- Sekvence razvojnih zadataka i alokacija resursa i alata	- Specifična sekvenca zadataka za dani projekt koja minimizira broj iteracija

2.2.3.9. Zakazivanje

Glavna razlika između zakazivanja i planiranja je činjenica da se kod zakazivanja točno definiraju vremenski atributi za svaki pojedini zadatak. Svaki zadatak dobiva odgovarajuće vremenske atribute koji definiraju početak i trajanje tog zadatka.

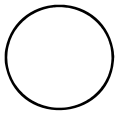
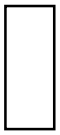
Tablica 26. Zakazivanje

Cilj aktivnosti, G_d	Ulazno znanje, I_k	Izlazno znanje, O_k	Ostvarena promjena znanja
Minimiziranje vremena do izlaska na tržište strukturiranjem zadatka u razvoju proizvoda	- Znanje o zadatcima razvoja, resursima - Znanje o metodama i algoritmima planiranja	- Sekvence razvojnih zadataka i alokacija resursa i alata s pripadajućim vremenskim atributima	- Specifična sekvenca zadataka za dani projekt koja minimizira vrijeme do izlaska na tržište

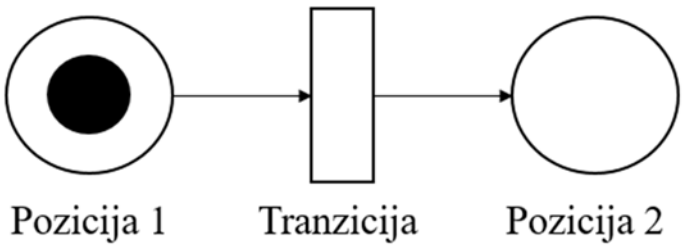
2.3. PETRI MREŽE [5]

Petri mreže se u posljednje vrijeme većinom koriste kao matematička metoda za opisivanje stvarnih procesa. Poznato je da je matematičko modeliranje stvarnih dinamičkih procesa vrlo složen problem zbog velikog broja poznatih i nepoznatih varijabli, te da bi se takvi procesi zadovoljavajuće matematički opisali razvijene su Petri mreže. Korištenjem Petri mreža mogu se na zadovoljavajuć način opisati procesi s velikim brojem istovremenih operacija, ciklički procesi, potencijalne promjene smjera razvoja procesa, te razni vremenski odnosi unutar nekog procesa. Prema [5] Petri mreža je bipartitni usmjereni multigraf. Pojam bipartitni opisuje dva osnovna elementa (poziciju i tranziciju), dok pojam multigraf predstavlja činjenicu da svaka veza između osnovnih elemenata ima pripadajući težinski faktor koji u stvarnosti predstavlja uvjete za izvršenje tranzicije. Jedna od prednosti Petri mreža je što se samo modeliranje procesa izvodi kombinacijom grafičkih i programskih značajki, te kao takav predstavlja vrlo praktičan pristup modeliranju procesa s diskretnim događajima. Jedan od potencijalnih problema prilikom korištenja Petri mreža je činjenica da u nekim slučajevima preglednost nije ostvarena kao kod nekih drugih mrežnih metoda. Kao primjer, proces se ne mora nužno pokrenuti na početku mreže, no takva fleksibilnost u modeliranju omogućuje simuliranje stvarnih cikličkih neusmjerenih procesa. U tablici 27. prikazani su osnovni elementi Petri mreže. Odgovarajućom kombinacijom osnovnih elemenata postiže se željeni model stvarnog procesa. U samom modelu mogu se razlikovati pomični i nepomični elementi, od kojih su jedini pomični elementi marke (eng. tokens), te se one pomiču prema prethodno definiranim uvjetima, i njihova uloga je određivanje jesu li ostvareni uvjeti za nastavak procesa u definiranom smjeru.

Tablica 27. Osnovni elementi Petri mreže

Grafički prikaz elementa	Naziv
	Pozicija
	Tranzicija
	Luk
	Marka

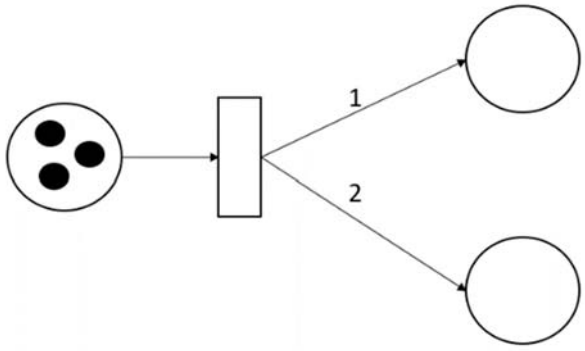
Grafički element pozicije može predstavljati pojedine objekte, uvjete ili sama stanja, dok tranzicije mogu predstavljati neke događaje u samom procesu, određenu aktivnosti ili potencijalnu promjenu određenih parametara u sustavu. Činjenica da su marke pomične omogućuje vizualno praćenje tijeka modeliranog procesa . Princip rada Petri mreža opisan je na osnovnoj Petri mreži s dvije pozicije, i jednom tranzicijom vidljivoj na sljedećoj slici.



Slika 6. Osnovna Petri mreža s dvije pozicije i jednom tranzicijom

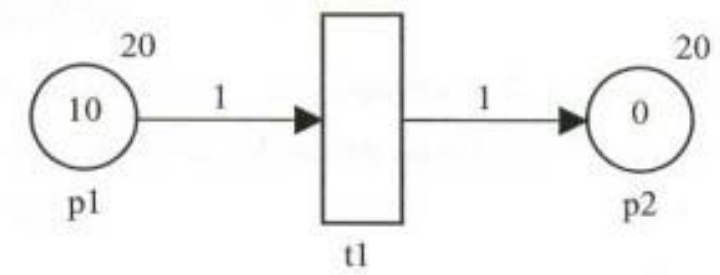
Temeljni princip rada Petri mreže zasniva se na ideji da su za razvoj procesa ili događaja potrebni određeni resursi. Ovisno o modelu, resursi mogu biti raznih vrsta, kao što su materijalni resursi, financijski, informacijski i mnogi drugi. Prema tome, da bi se iz jedne pozicije prešlo u drugu potrebno je osigurati definiranu količinu resursa za prijelaz. Količina resursa grafički se prikazuje brojem marki unutar pojedine pozicije. Pozicije su uvijek međusobno odijeljene tranzicijama, čija je uloga da dopuste prijelaz marki iz jedne pozicije u drugu ukoliko su ostvareni definirani uvjeti broja potrebnih resursa za prijelaz iz početne pozicije. Nadalje, postojanje dovoljne količine resursa za tranziciju naziva se postojanje razrješenja tranzicije, te je potrebno naglasiti da samim postojanjem dovoljne količine resursa ne mora nužno značiti da će marke i prijeći preko tranzicije. Prolaskom marki preko tranzicije

dobiva se novi raspored resursa unutar modela, te su tako iznova definirani novi uvjeti za razrješenje i napuštanje pojedine tranzicije. Ukoliko promotrimo sljedeću sliku vidimo da su u danoj Petri mreži ostvareni uvjeti za prijelaz marki u nove pozicije. Također, nakon što se izvrši tranzicija u početnoj poziciji više nema marki, što sprečava novu tranziciju marki. Za svaku tranziciju definiraju se odgovarajuća pravila, te je bitno da se definirana pravila ne mijenjaju tokom rada sustava. Na primjer, pri modeliranju elektrotehničkog sustava, razmještaj marki definira postoji li na određenoj lokaciji električne struje, dok tranzicije predstavljaju sklopke kojima se ostvaruje usmjeravanje električne energije unutar sustava. Prilikom inicijalizacije Petri mreže izvršava se proces početnog markiranja, te je razmještaj marki proizvoljan, što znači da se proces može aktivirati na bilo kojoj poziciji, te da jedan model može pokrivati mnoge potencijalne varijacije u procesu.



Slika 7. Ostvareni uvjeti za razrješenje tranzicije

U matematičkom zapisu Petri mreža se prikazuje kao uređena šestorka pojedinih pozicija, tranzicija, ulaznih i izlaznih događaja, te težinske matrice pojedinih lukova i konačno vektora stanja marki [5].



Slika 8. Petri mreža [5]

Matematički zapis Petri mreže vidljive na slici 8 slijedi:

$P = \{p_1, p_2\}$, neprazan skup pozicija

$T = \{t_1\}$, neprazan skup tranzicija

$F = \{(p_1, t_1), (t_1, p_1)\}$, neprazan skup lukova

$K = \{(p_1, 20), (p_2, 20)\}$, neprazan skup kapaciteta pozicija

$W = \{((p_1, t_1), 1), ((t_1, p_2), 1)\}$, neprazan skup težinskih faktora lukova

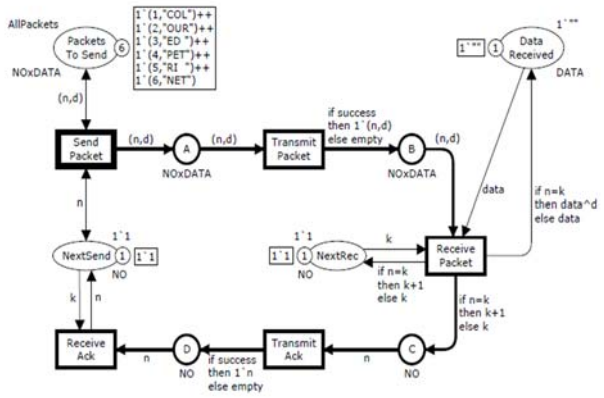
$M_0 = \{(p_1, 10), (p_2, 10)\}$, početni razmještaj marki

Prilikom korištenja klasičnih Petri mreža određena su dva glavna nedostatka koji otežavaju njihovo korištenje:

1. Da bi se ostvarila pravilna manipulacija s ulaznim i izlaznim podatcima potrebno je definirati veliki broj pozicija što u konačnici dovodi do vrlo velikih i nepreglednih mreža
2. Nije moguće izvesti Petri mrežu s više podmreža (hijerarhijska dekompozicija)

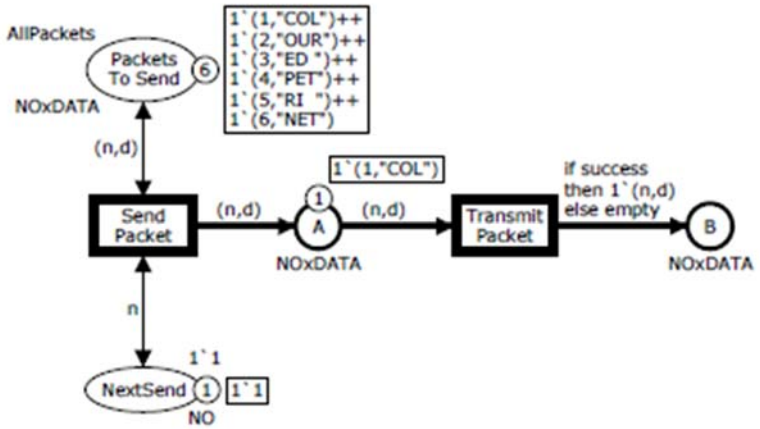
2.3.1. Obojene Petri mreže [6]

Sa svrhom rješenja prethodno navedenih problema klasičnih Petri mreža, korištenjem programskih jezika novije generacije (*STANDARLD ML, JAVA*), razvijene su obojene Petri mreže. One uključuju sve mogućnosti klasičnih Petri mreža s mogućnostima strukturiranja ulaznih i izlaznih podataka, te je moguće konstruirati modele s hijerarhijski strukturiranim podmrežama, kao i modele s vremenski definiranim parametrima. U obojenim Petri mrežama tokeni prenose vrijednosti podataka te su označen vremenskim žigom, koji definira najraniji mogući prijelaz iz jedne pozicije u drugu. Princip rada obojene Petri mreže, u nastavku korištenog zapisa CPN, objašnjen je uz pomoć sljedeće slike gdje je prikazan model prijenosa tekstualnih paketa od pošiljatelja do primatelja.



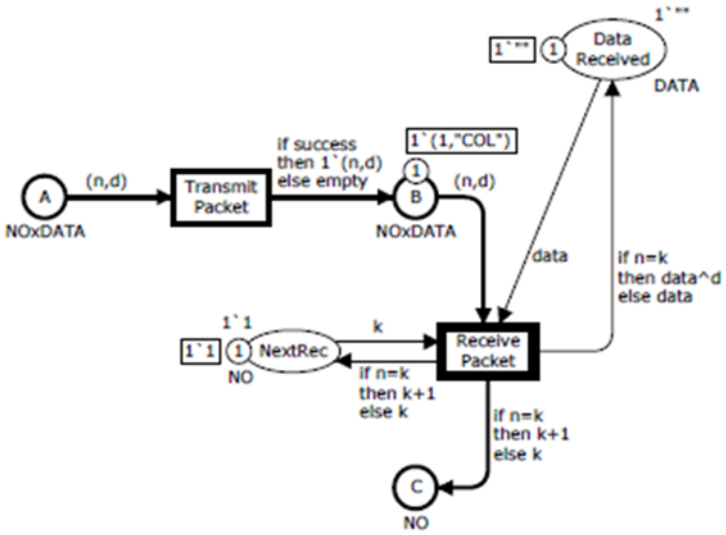
Slika 9. CPN model slanja tekstualnih paketa u inicijalnom stanju M_0 [6]

Prikazani CPN model sastoji se od 8 pozicija koje su grafički prikazane kao krugovi ili elipse, 5 tranzicija prikazanih pravokutnicima, usmjerenim lukovima koji definiraju smjer prijelaza tokena, te određenih natpisa koji definiraju uvjete tranzicije tokena. Pojam boje tokena odnosi se na podatke koji prenosi token. Svaka pozicija može primiti više tokena, te svaki token može prenositi više podataka. Specifičan položaj kombinacije boja tokena i samih tokena na određenim pozicijama definiraju trenutno stanje modela. Za svaku poziciju definira se dozvoljena boja tokena, te ukoliko pozornije promotrimo sliku 10. možemo vidjeti da neke pozicije zahtijevaju skalarni produkt cijelog broja i podatka (*NOxDATA*), dok neke pozicije zahtijevaju samo cijeli broj(*NO*). Također, inicijalno stanje tokena M_0, prikazano je u odgovarajućim pravokutnicima pokraj pozicije. Generalni princip prikazanog modela uzima u obzir stanje pošiljatelja koje je definirano pozicijama ***Packets To Send*** i ***NextSend***, stanje mreže između primatelja i pošiljatelja definirano je pozicijama ***A***, ***B***, ***C***, ***D***, te stanje primatelja pozicijom ***Data Recieved***. Prikazani model simulira prijenos dijelova teksta od pošiljatelja do primatelja, na način da pojedina boja tokena definira na koji način se prenose dijelovi teksta. U početnom stanju vidljivo je da je tranzicija ***Send Packet*** omogućena, što znači da odgovarajući token iz pozicije ***Packets to Send*** može prijeći u poziciju A. Odabir tokena vrši se uz pomoć tokena koji se nalazi na poziciji ***NextSend***, te nakon evaluacije slijedi da nakon izvršenja prve tranzicije model prelazi u novo stanje prikazano na sljedećoj slici.



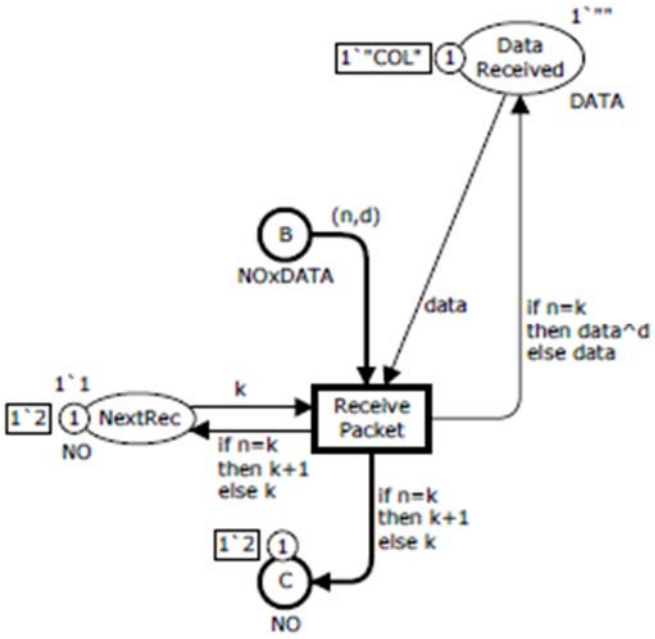
Slika 10. CPN model nakon aktiviranja tranzicije *SendPacket* [6]

Dodatno, promotre li se lukovi vezani uz tranziciju ***SendPacket*** vidljivo je da su obostrani što znači da se aktivacijom tranzicije stvara kopija tokena koji se šalje u mrežu, dok u početnim pozicijama ostaju svi tokeni, što omogućuje ponovno slanje istog tokena ukoliko se ne uspije izvršiti tranzicija ***Transmit Packet***. Ukoliko se tranzicija uspješno izvrši sustav prelazi u novo stanje prikazano na sljedećoj slici.



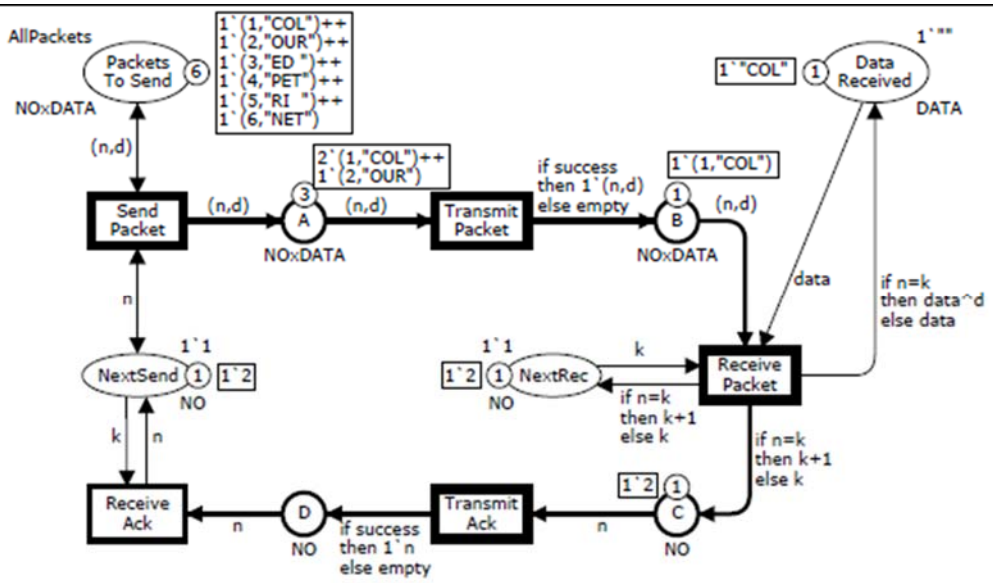
Slika 11. CPN model nakon aktiviranja tranzicije *TransmitPacket [6]*

U novom stanju sustava, provjerava se da li je dobiven token onaj koji je je primatelj zatražio, te ukoliko je dobiven token zadovoljavajuće boje, aktivira se tranzicija ***Receive Packet*** koja omogućuje prijelaz dijela teksta u konačnu poziciju ***Data Received***. Također, pokreće se proces koji je zaslužan za izmjenu boje tokena u poziciji ***NextSend*** čime se mijenja koji se token šalje iz pozicije ***Packets to Send***. Opisano stanje prikazano je na slici 12.



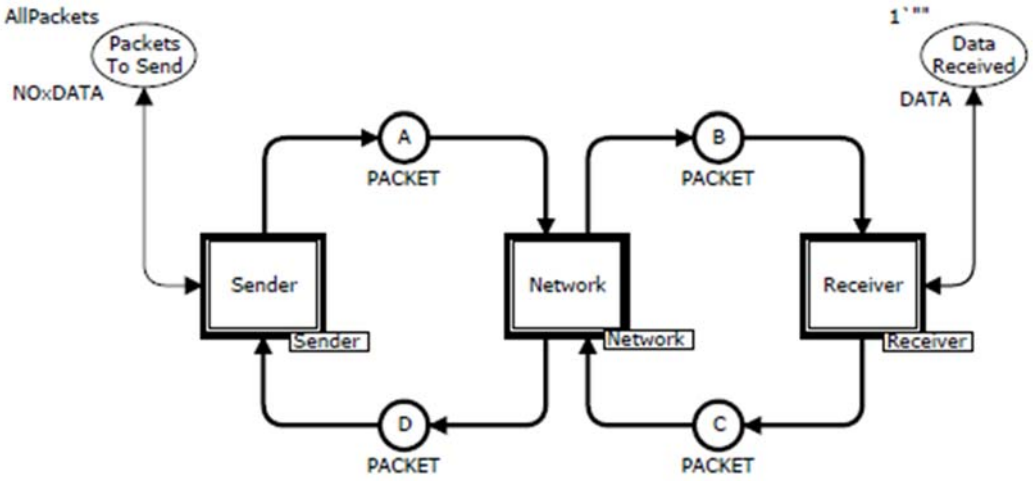
Slika 12. CPN model nakon aktiviranja tranzicije *ReceivePacket [6]*

Nakon aktiviranja preostalih tranzicija sustav prelazi u novo stanje, koje predstavlja završetak prve petlje operacija i prikazan je na slici 13.



Slika 13. Stanje CPN modela nakon prvog uspješnog prijenosa informacije [6]

Jedna od bitnih prednosti CPN modela je sposobnost izvršavanja paralelnih procesa, tako da ukoliko je ostvaren potreban broj resursa na odgovarajućim pozicijama, mogu se aktivirati više tranzicija, što uvelike ubrzava simulaciju procesa i omogućuje kvalitetan pregled razvoja procesa. Kao što i većina današnjih računalnih aplikacija koristi module, tako i CPN modeli mogu imati više hijerarhijski vezanih modula, od kojih svaki modul može biti zaseban CPN model. Modularni prikaz opisanog CPN modela prikazan je na slici 14.



Slika 14. CPN model s modulima pošiljatelja [6]

3. PREGLED STUDIJE SLUČAJA PROVEDENE UNUTAR TVRTKE DAIMLER AG

U ovom poglavlju korišteni su podatci dobiveni unutar tvrtke Daimler AG koja je jedan od najutjecajnijih tvrtki svjetske autoindustrije. Potrebno je dodati da je detaljnija analiza provedena u sklopu izrade doktorske disertacije koja je u završnoj fazi izrade. Cilj ovog dijela rada je strukturirati podatke dobivene na osnovu prezentacija i izvješća sa sastanaka razvojnih inženjera unutar tvrtke. Poznato je da se vrlo velika količina podataka prenosi preko elektroničkih poruka između zaposlenika, no zbog unutrašnjih pravila tvrtke pristup tim informacijama nije odobren, te iz tog razloga ovaj skup informacija nije uključen u ovom radu.

Sastanci inženjera održavaju se na tjednoj bazi za svaki pojedini aktualni projekt unutar tvrtke, te se u svrhu planiranja sastanaka prethodno definiraju sljedeći parametri:

- Lokacija sastanka
- Vrijeme održavanja sastanka
- Prisutni zaposlenici
- Dnevni red sastanka

Popis tema jedan je od najvažnijih parametara koji je potrebno dostaviti pozvanim zaposlenicima čim prije moguće, što omogućuje zaposlenicima kvalitetniju pripremu za pojedinu temu sastanka. Tokom samog sastanka prisutni zaposlenici zajednički rješavaju dane probleme, raspravljaju, te izmjenjuju ideje i misli. Provedene diskusije zapisničar pokušava maksimalno formalizirati, te ih strukturirano zapisati u predložak kojim se osigurava jednako zapisivanje podataka kroz sve održane sastanke. S obzirom da zapisničari mijenjaju, mogu se i očekivati male razlike u načinu zapisivanja pojedinih fraza (korištenje različitih glagolskih vremena, korištenje skraćenica), što u konačnici predstavlja problem ukoliko se izvješće sastanka softverski analizira. Izvješće sa sastanka sastoji se od tri glavna dijela:

- Osnovne informacije o sastanku (lokacija, projekt, vrijeme održavanja,…)
- Popis prisutnih zaposlenika
- Popis tema rasprava

Primjer izvješća sa sastanka unutar tvrtke Daimler AG prikazan je na sljedećoj slici, te je vidljivo da su pojedini stupci zamućeni zbog politike tvrtke. Teme sastanaka su podijeljene u grupe koje se često ponavljaju, dok se upisom podteme definira sama rasprava.

Subject	SubSubject	Current status / Next step	Deadline	Responsible Team	Responsible Person	Tag 1	Tag 2	Tag 3
Anforderungen	[redacted]	[redacted]	########	*Team 52*Team 19	*Employee55 *Employee26	Something is in PDM stored		
Vorgaben	[redacted]	[redacted]	########	*Team 52	*Employee74	Something should be stored in PDM		
Status	[redacted]	[redacted]	########	*Team 52*Team 19	*Employee55 *Employee121	Special meeting should be organised	Need to make agreement about something	Something should be stored in PDM
Bewertung	[redacted]	[redacted]	########	*Team 52	*Employee34	Something is presented to group	Some conflict in PDM	
Bewertung	[redacted]	[redacted]	########	*Team 52	*Employee34	Special meeting should be organised	Something is agreed	Something should be stored in PDM
Heckdeckelaufdrehung		[redacted]	########	*Team 52*Team 37*Team 17	*Employee124* Employee19 *Employee10	Need to make agreement about something	Something should be stored in PDM	

Slika 15. Primjer izvješća sa sastanka unutar tvrtke Daimler AG

Na osnovu provedene analize može se uočiti da većina najučestalije korištenih aktivnosti imaju veze s rutinskim poslovima komunikacije s PDM – sustavom (upisivanje ili čitanje podataka) te je popis generaliziranih provedenih aktivnosti od najčešće korištenih do najrjeđe prikazan u nastavku:

- Obavještavanje
- Dodjeljivanje vrijednosti
- Organiziranje sastanka
- Prezentiranje
- Prijenos podataka u PDM
- Evaluacija
- Donošenje odluke
- Provjera

Na osnovu provedenog istraživanja, te na temelju istraživanja opisanog u pregledu literature definiranja je generalizirana taksonomija aktivnosti koja će se koristiti kao referentna za izradu programske podrške vođenju inženjerskih sastanaka. Bitno je naglasiti da su referentne aktivnosti one aktivnosti koje po hijerarhijskoj strukturi nemaju dijete. Iz toga slijedi da je na temelju provedenog istraživanja i na uzorku od 560 tema na sastancima unutar tvrtke Daimler AG, definirano 18 općih aktivnosti koje u velikom dijelu pokrivaju sve moguće aktivnosti koje

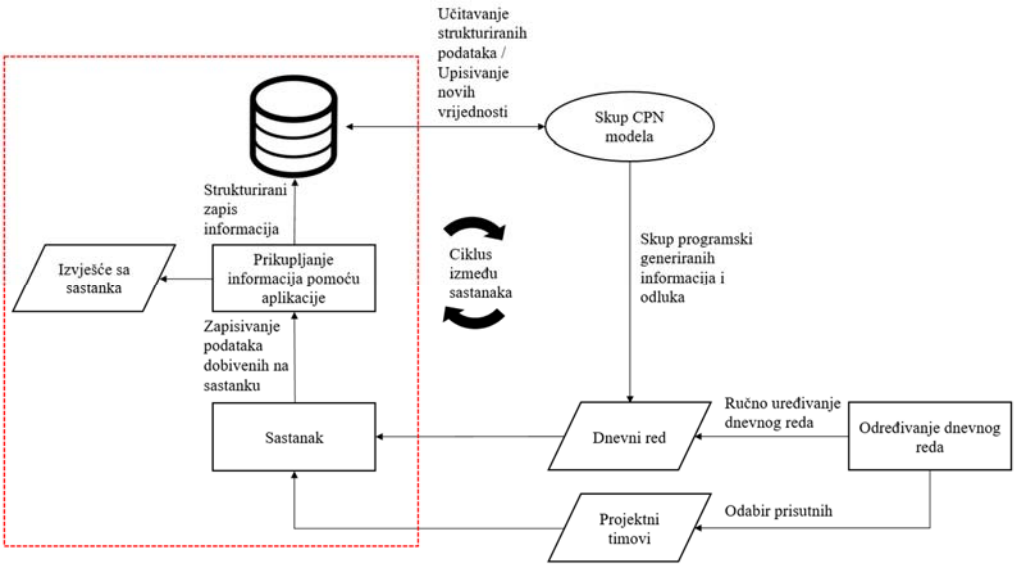
se javljaju tokom razvojnog procesa. Ovakvom taksonomijom postiže se zadovoljavajuća diskretizacija aktivnosti, te su time postavljeni čvrsti temelji za razvoj softverske podrške opisane u ovom radu.

Aktivnosti na inženjerskim sastancima:

- Zahtijevanje
  - Zahtijevanje informacije
  - Zahtijevanje ispitivanja
  - Prijenos podataka iz PDM sustava
    - Traženje međuovisnosti parametara
    - Traženje svojstava parametara
    - Traženje statusa parametra
    - Traženje vrijednosti parametra
- Izvještavanje
  - Informiranje
    - Informiranje o završenom zadatku
    - Informiranje o općim informacijama
    - Informiranje o promjeni statusa zadataka
    - Informiranje s drugog sastanka
  - Prezentiranje
    - Prezentiranje odluke
    - Prezentiranje rezultata
- Dodjeljivanje
  - Analiziranje
  - Provjeravanje
  - Ocjenjivanje
  - Donošenje odluke
  - Rješavanje
    - Rješavanje kolizija u 3D modelu
    - Rješavanje spregnutih parametara
    - Rješavanje općeg problema
  - Organiziranje sastanka
  - Ispitivanje
  - Prijenos podataka u PDM sustav
    - Izmjena međuovisnosti parametara
    - Izmjena svojstava parametra
    - Izmjena statusa parametra
    - Izmjena vrijednosti parametra
    - Unos novog parametra

4. SOFTVERSKO RJEŠENJE UPRAVLJANJA INFORMACIJAMA DOBIVENIH TIJEKOM INŽENJERSKIH SASTANAKA

U ovom poglavlju detaljno je objašnjena struktura predložene baze podataka, te način rada razvijene aplikacije u svrhu kvalitetnijeg i bržeg prikupljanja informacija stvorenih tokom projektnih sastanaka. Na sljedećoj slici prikazan je prijedlog ciklusa operacija tokom pojedine faze razvojnog procesa, koji uključuje korištenje razvijene aplikacije za prikupljanje informacija sa sastanaka, te skupa CPN modela koji upravljaju daljnjim tokom informacija unutar tvrtke. Općenita ideja ovog prijedloga unapređenja je maksimalno strukturirati i formalizirati sve informacije dobivene tokom sastanka, što omogućuje razvoj odgovarajućih CPN modela za pojedine aktivnosti koje su održane tokom sastanka. Na osnovu strukturiranih podataka, aktiviraju se odgovarajući CPN modeli za pojedinu aktivnost. U sklopu ovog rada razvijena je programska podrška koja rješava probleme definirane crvenim okvirom vidljivim na slici 16.



Slika 16. Prijedlog toka informacija u ciklusu između sastanaka

U ovom trenutku još nije fizički razvijen CPN model koji bi na odgovarajući način upravljao predloženim ciklusom, no postavljeni su temeljni zadaci koji će u budućnosti biti osnova za realizaciju odgovarajućih CPN modela:

- Analiza strukturiranih podataka u svrhu pokretanja odgovarajućih procesa
- Izvršavanje rutinskih prijenosa podataka (automatizirana komunikacija s PDM sustavom)

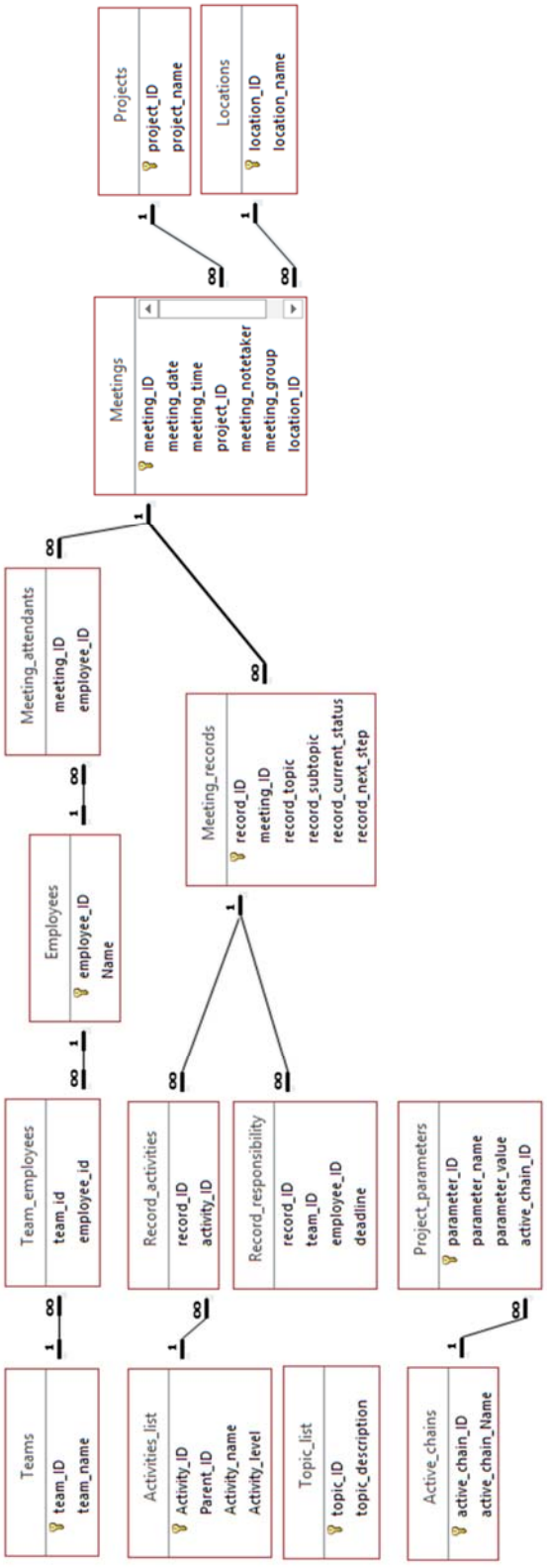
- Korištenje automatski generiranih informacija u svrhu podrške kolaborativnom razvoju proizvoda (određivanje odgovornosti pojedinih timova za odgovarajuće zadatke)
- Kreiranje novog dnevnog reda za neki sljedeći sastanak na osnovu podataka dobivenih na održanom sastanku
- Komunikacija s pojedinim članovima odgovarajućeg tima

4.1. Baza podataka

Za izradu baze podataka korišten je programski paket *MICROSOFT ACCESS 2016* iz razloga što je moguće na relativno jednostavan način uspostaviti komunikaciju baze podataka i vanjske razvijene aplikacije u programskom jeziku *VB.NET,* te je jedan od najkorištenijih programskih paketa za upravljanje bazama podataka. U ovom poglavlju detaljno je opisana struktura baze podataka, razlozi postavljanja odgovarajućih veza između pojedinih tablica baze podataka, te konačno, opisane su forme za održavanje (upisivanje novih slogova, izmjena atributa postojećih slogova) odgovarajućih tablica u bazi podataka. Komunikacija između aplikacije i baze podataka ostvaruje se pomoću OLEDB aplikacijskog programskog sučelja, koje omogućuje korištenje SQL (*Structured query language*) naredbi za manipulaciju s podacima baze podataka. Odabranim pristupom manipulacije s podacima osigurava se standardiziranost baze prema trenutno važećoj normi *ISO/IEC 9075:2016,* što u konačnici olakšava eventualne naknadne promjene baze podataka, te samo proširenje razvijene aplikacije ukoliko se prilikom korištenja otkriju potencijalna unapređenja predloženog sustava. Trenutno zapisani podatci u bazi podataka su generički, te nemaju nikakve veze s realnim parametrima u razvoju proizvoda, no cilj je pokazati princip rada, te za to nisu potrebni realni podatci.

Prema definiciji [7], baza podataka je skup međusobno povezanih podataka bez redundancije (nepotreban zapis istog podatka u više tablica), te je u skladu s tom definicijom realizirana struktura podataka vidljiva na slici 17. Središnja tablica ***Meetings*** sadrži osnovne podatke o samom sastanku, te je odgovarajućim relacijama vezana s ostalim tablicama. Za svaki sastanak zapisuju se informacije o vremenu održavanja sastanka, mjestu održavanja sastanka, vrsti sastanka, te naziv projekta u sklopu kojeg je sastanak organiziran. Ukoliko se pozornije promotre odnosi između pojedinih tablica u bazi podataka, vidljiv je način strukturiranja informacija sa sastanaka. Na nekom sastanku vezom jedan prema više povezani su odgovarajući zapisi (tablica ***Meeting_records***), od kojih je svaki pojedini zapis također vezom jedan prema više vezan uz tablicu u kojima su upisane aktivnosti vezane uz pojedini zapis (tablica

Record_activities), te tablicu u kojoj određeni zapis ima skup odgovornih osoba i timova (tablica ***Record_responsibility***).



Slika 17. Grafički prikaz baze podataka

Promotri li se na slici 12. tablicu ***Topic_list,*** vidljivo je da ona nije nikakvim odnosima vezana s ostalim tablicama, te je njezina uloga ubrzavanje procesa popunjavanja osnovnih podataka o pojedinom zapisu sastanka, što će detaljnije biti opisanu u poglavlju 4.2.3. Trenutni trendovi u razvoju proizvoda maksimiziraju korištenje aktivnih lanaca parametara, te je iz tog razloga definiran odnos između tablica ***Active_chains***, te ***Project_parameters***, gdje su povezani odgovarajući parametri unutar nekog aktivnog lanca. Tablica ***Projects*** vezana je jedino uz pojedini sastanak što nije dovoljno dobra definicija, jer pojedini projekti imaju odgovarajuće parametre i bilo bi točnije da su i te tablice povezane, no takvim vezama dolazi se do potencijalnih problema u radu baze podataka koji se javljaju prilikom definicije cikličkih odnosa između pojedinih tablica.

4.1.1. Hijerarhijska struktura aktivnosti na sastancima

Na osnovu provedene analize sastanaka unutar tvrtke Daimler AG, u prethodnom poglavlju definirana je hijerarhijska struktura aktivnosti na sastancima. Opisane aktivnosti u bazi podataka nalaze se u tablici ***Activites_list,*** te je njezin sadržaj vidljiv u tablici 28.

Tablica 28. Hijerarhijski popis aktivnosti

Activity_ID	Parent_ID	Activity_name	Activity_level
1		Requesting	1
2	1	Requesting information	2
3	1	Requesting testing	2
4	1	Transferring from PDM	2
5	4	Get parameter dependencies	3
6	4	Get parameter properties	3
7	4	Get parameter status	3
8	4	Get parameter value	3
9		Reporting	1
10	9	Informing	2
11	10	Informing about completed task	3
12	10	Informing about general information	3

Activity_ID	Parent_ID	Activity_name	Activity_level
13	10	Informing about task status update	3
14	10	Reporting from other meeting	3
15	9	Presenting	2
16	15	Presenting decision	3
17	15	Presenting results	3
18		Assigning	1
19	18	Analysing	2
20	18	Checking	2
21	18	Evaluating	2
22	18	Making decision	2
23	18	Resolving	2
24	23	Resolving 3D model collision	3
25	23	Resolving coupled parameters	3
26	23	Resolving general issue	3
27	18	Scheduling meeting	2
28	18	Testing	2
29	18	Transferring to PDM	2
30	29	Changing parameter dependencies	3
31	29	Changing parameter properties	3
32	29	Changing parameter status	3
33	29	Changing parameter value	3
34	29	Inserting new parameter	3

U stupcu Activity_level definirana je razina hijerarhije, dok je u stupcu Parent_ID definirana veza pojedine aktivnosti s odgovarajućom aktivnošću više razine. Ovakvim načinom zapisa omogućeno je jednostavno naknadno dodavanje novih aktivnosti ili izmjena postojećih. Također, moguće je i modificirati hijerarhijsku strukturu izmjenom odgovarajućih indeksa roditeljskih aktivnosti. Kao što je u prethodnom poglavlju opisano, nastoje se realizirati CPN

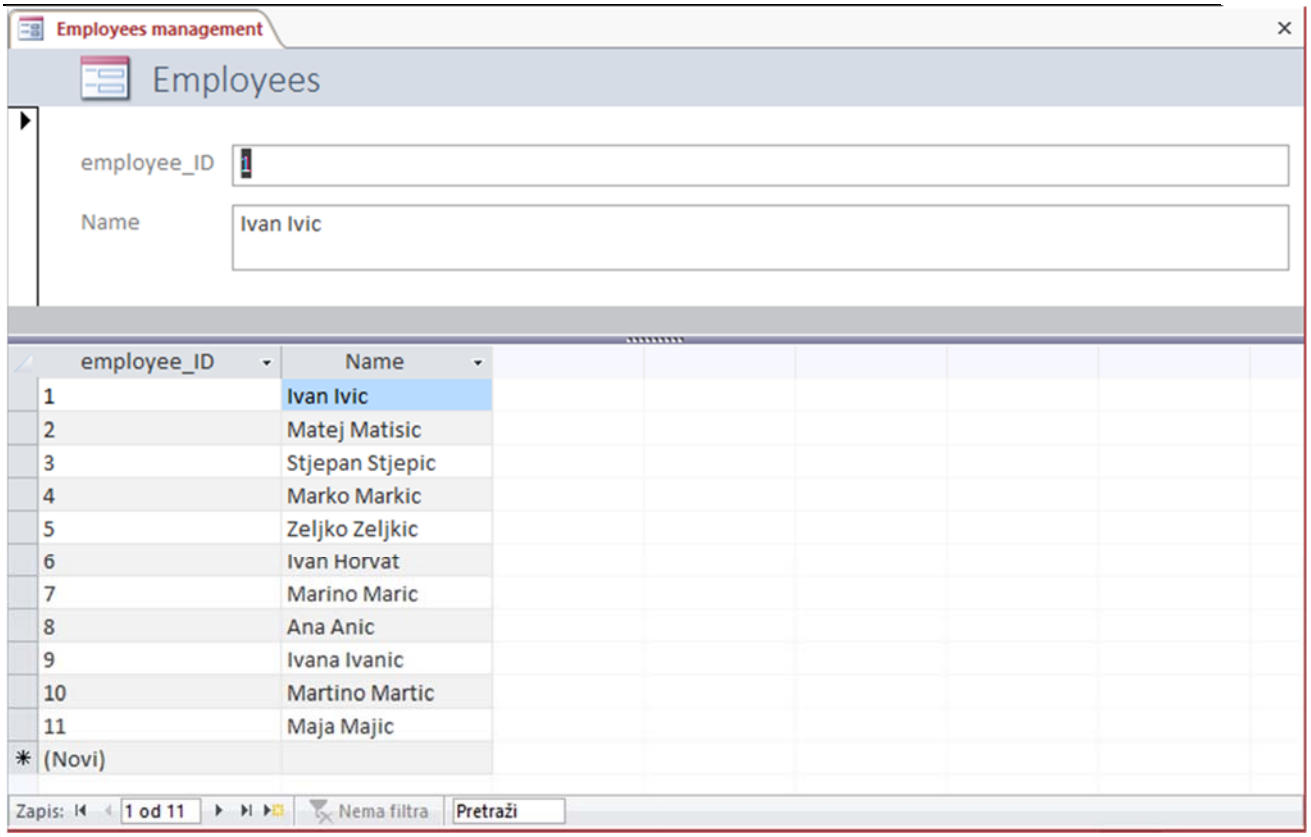
modeli za sve predložene aktivnosti na najnižoj hijerarhijskoj razini, te ukoliko se promotri tablica 28. vidljivo je da je potrebno razviti 18 CPN modela koji će se aktivirati ovisno o zapisima u tablici ***Record_activities***. Promotrimo li strukturu baze na slici 17. vidljivo je da su prethodno navedenoj tablici zapisani samo iznosi primarnih ključeva odgovarajućih zapisa sastanaka i aktivnosti, koji su po vrsti podatka pozitivni cijeli brojevi. Ovakav zapis odabran je sa svrhom da se prilikom budućeg razvoja CPN modela maksimalno pojednostavi izbor CPN modela ovisno o provedenoj aktivnosti na sastanku. Programska manipulacija s cijelim brojevima je izrazito jednostavna, te se odluka o aktivaciji odgovarajućeg CPN modela donosi jednostavnom usporedbom dvaju cijelih brojeva, indeksa CPN modela i primarnog ključa provedene aktivnosti.

4.1.2. Određivanje odgovornosti za pojedini zapis sastanka

Promotrimo li tablicu ***Record_responsibility*** na slici 17. vidljivo je da je primijenjen analogan pristup zapisa kao i u tablici ***Record_activities.*** Za svaki pojedini zapis sa sastanka vezani su primarni ključevi pojedinog zaposlenika i tima čiji je taj zaposlenik član što omogućuje analogni pristup aktivaciji CPN modela koji služe za komuniciranje s odgovarajućim zaposlenicima. Dodatno ograničenje postavlja se na vremenski rok koji je po vrsti podatka „Date/Time“, što se prilikom prijenosa pretvara u tip podataka „string“, što zahtijeva dodatne korake prilikom programskog odlučivanja CPN modela.

4.1.3. Održavanje baze podataka

Održavanje baze podataka vrši se unutar programskog paketa *MICROSOFT ACCESS 2016*, gdje se uz pomoć odgovarajućih obrazaca pristupa pojedinim tablicama, te se prema potrebi mijenjaju vrijednosti pojedinih atributa odabranog sloga, ili dodavanje novog sloga s odgovarajućim atributima. Odabir željenog obrasca vrši se klikom na ikonu obrasca u donjem lijevom uglu grafičkog sučelja programskog paketa. Kao reprezentativni primjer obrasca za uređivanje prikazan je obrazac za uređivanje zaposlenika tvrtke koji je vidljiv na slici 18. Na prikazanom obrascu moguće je pregledavanje atributa pojedinog zaposlenika, ili po potrebi dodavanje novog zaposlenika s svim potrebnim atributima. Također je vidljiv tablični prikaz zaposlenika, što je ukoliko se u obzir uzmu realne okviri, vrlo pregledan prikaz zbog često velikog broja timova i zaposlenika unutar neke tvrtke.

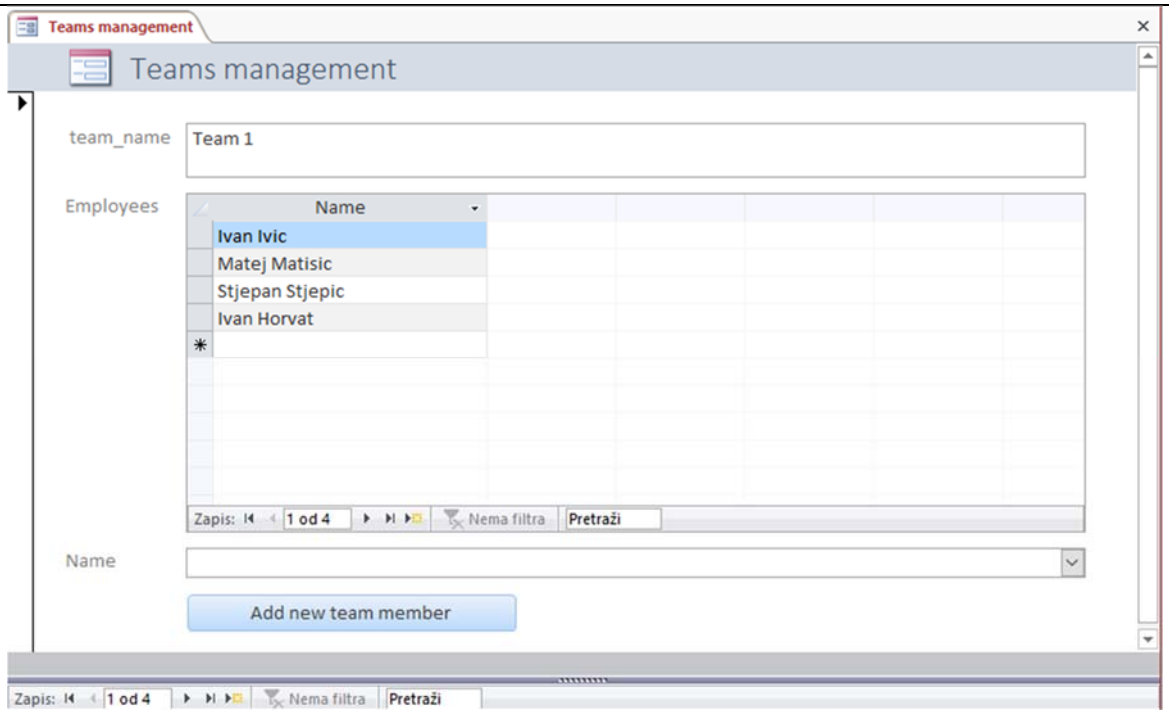


Slika 18. Obrazac za uređivanje baze zaposlenika

Na analogan način, te uz pomoć analogno izrađenih obrazaca održavaju se podatci sljedećih tablica:

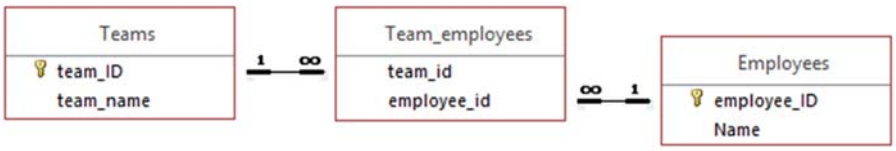
- Activities_list (Activities management) – uređivanje hijerarhijske strukture aktivnosti
- Locations (Locations management) – uređivanje podataka o lokacijama tvrtke
- Projects (Projects management) – uređivanje podataka o projektima

Upravljanje timovima vrši se uz pomoć obrasca prikazanog na slici 19., te je vidljivo da je prikazani obrazac relativno složeniji. Da bi se ostvarilo adekvatno upisivanje odgovarajućih podataka u željene tablice korišten je modul unutar programskog paketa koji omogućuje pisanje skupa naredbi u programskom jeziku *Visual Basic for Applications.* Obrazac se sastoji od podobrasca u kojem su tablično prikazani zaposlenici kao članovi odgovarajućeg tima. Također, po potrebi je moguće uređivati atribute željenog tima i dodavanje novog tima.



Slika 19. Obrazac za uređivanje timova

Raspored zaposlenika unutar pojedinih timova definiran je relacijom više prema više, te se indeksi primarnih ključeva timova i zaposlenika zapisuju u veznu tablicu ***Team_employees***, što je vidljivo na prikazu isječka iz strukture baze podataka prikazanom na slici 20.



Slika 20. Struktura baze timova i zaposlenika

U obrazac za uređivanje pojedinog tima dodani su objekti s kojima se vrši dodavanje odabranog zaposlenika u trenutno odabran tim. Korisnik odabirom željenog zaposlenika u padajućem izborniku ***Name***, te pritiskom na gumb ***Add new team member*** pokreće slijed naredbi, čija zadaća je određivanje indeksa primarnog ključa odabranog tima i odabranog zaposlenika, te nakon određivanja opisanih podataka, upisati ih u odgovarajuća polja tablice ***Team_employees.***

4.2. Aplikacija „Daimler AG Design Reviews“

U ovom poglavlju detaljno je opisana aplikacija za prikupljanje podataka sa inženjerskih sastanaka na temelju zahtjeva određenih iz analize sastanaka u tvrtki Daimler AG. Glavni cilj aplikacije je omogućiti strukturirano zapisivanje odgovarajućih podataka i atributa na način koji zahtjeva minimalnu količinu operacija korisnika prilikom rada s aplikacijom. Razvijena aplikacije treba biti podrška tokom sastanaka, te je potrebno osigurati da korisnik u kratkom

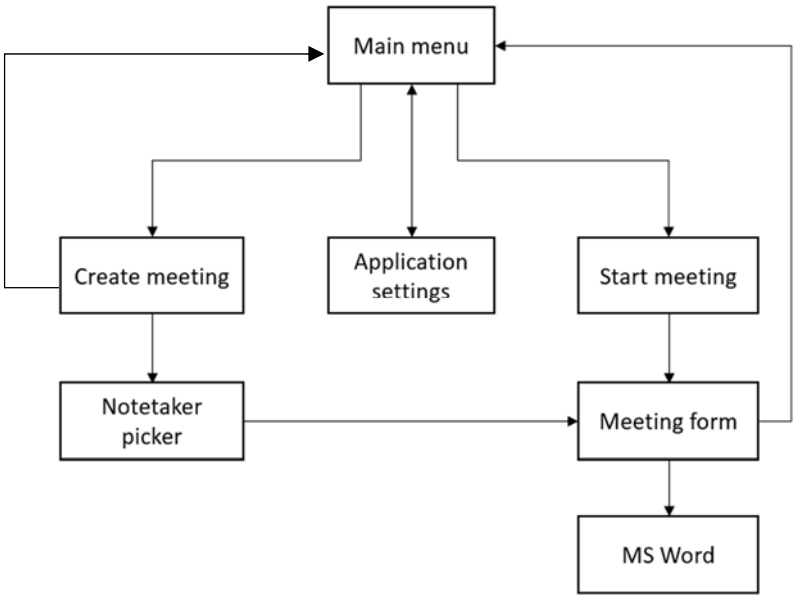
vremenu može zapisati točno određene informacije koje su vezane uz pojedinu temu sastanka, što omogućuje korisniku da se fokusira na sam sastanak te da kvalitetnije iskoriste vrijeme samog sastanka.

Poznato je da je jedan od najčešće korištenih operacijskih sustava *MICROSOFT WINDOWS,* te je iz tog razloga odabran programski paket za izradu aplikacija na *MS WINDOWS* platformi, *MICROSOFT VISUAL STUDIO 2016.* Odabrani programski jezik za izradu aplikacije je *VISUAL BASIC.NET.* S obzirom da je većina korisnika upoznata s Windows platformom, korištenje same aplikacije je vrlo jednostavno iz razloga što se pri izradi grafičkog sučelja koriste isti objekti koji se većim dijelom koriste u samom operativnom sustavu *MS WINDOWS*. Korištenjem navedenog programskog paketa osigurava se kompatibilnost aplikacije s svim računalima koji rade na platformi *MS WINDOWS,* no potrebno je osigurati da računalo na kojem se pokreće aplikacija ima pristup upravljačkim programima baze podataka, te softverskom okviru *.NET FrameWork.* Iz tog razloga slijede nazivi upravljačkih programa koje je potrebno instalirati prije same instalacije opisane aplikacije:

- Access Database Engine 2010 (ACE.OLEDB.12)
- .NET Framework 4.5.2

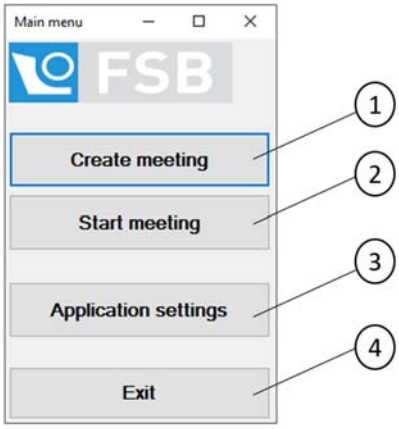
Navedeni upravljački programi su službeno izdani za platformu *MS WINDOWS,* te ih je vrlo lako pronaći korištenjem nekih od poznatijih tražilica. Također, ukoliko korisnik koristi neki moderniji operativni sustav (MS Windows 7,8,10) zahtijevani .NET Framework nije potrebno dodatno instalirati, jer navedeni sustavi dolaze s novijim softverskim okvirom koji je nazadno kompatibilan, te sadržava sve potrebne biblioteke koje su korištene u razvijenoj aplikaciji. Razvoj aplikacije proveden je s ciljem da korisnik ne treba imati nikakvog iskustva pri radu s bazama podataka, te da na vrlo jednostavan način prelazi s odgovarajućih obrazaca do sljedećih. Za sam rad aplikacije također je potrebno da računalo ima instaliran programski paket za uređivanje teksta *MS Word,* iz razloga što se nakon sastanka u navedenom programskom paketu programski generira izvještaj sa sastanka, na osnovu predloška koji je prikazan u nastavku ovog poglavlja.

Struktura obrazaca prikazana je na slici 21. gdje je vidljiv raspored razvijenih obrazaca, te redoslijed aktiviranja obrazaca ovisno o izboru odgovarajućih objekata na obrascu ***Main Menu.*** Skup naredbi vezanih uz pojedini obrazac vidljiv je u prilogu ovog rada, s napomenom da su glavne klase odgovarajućih obrazaca nazvane prema strukturi obrazaca vidljivoj na slici 21. što omogućuje jednostavno snalaženje prilikom pregleda programskog koda.



Slika 21. Struktura obrazaca

Prilikom pokretanja aplikacije prikazuje se glavni obrazac aplikacije ***Main menu*** vidljiv na slici 22.



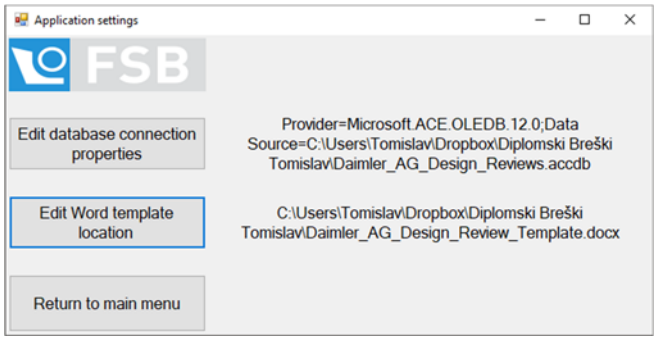
Slika 22. Glavni meni aplikacije

Ovisno o pritisnutom gumbu pokreće se odgovarajući procesi:

- Rezervacija sastanka sa svim potrebnim atributima – Gumb 1
- Pokretanje prethodno rezerviranog sastanka – Gumb 2
- Podešavanje postavki aplikacije – Gumb 3
- Zatvaranje aplikacije – Gumb 4

4.2.1. Osnovne postavke za rad aplikacije

Osnovni parametri za komunikaciju aplikacije s bazom podataka i aplikacijom ***MS Word*** definirani su u globalnoj klasi, te ih je moguće uređivati otvaranjem obrasca vidljivog na slici 23., pritiskom na gumb ***Application settings*** u glavnom meniju aplikacije.

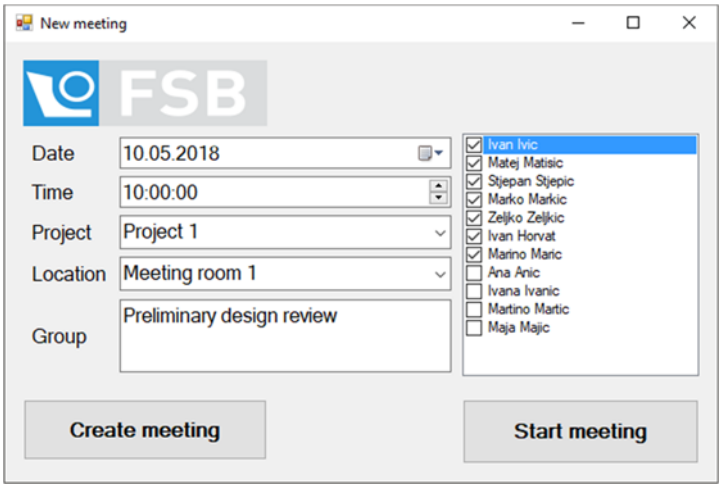


Slika 23. Obrazac za uređivanje postavki aplikacije

Pritiskom na gumb ***Edit database connection properties,*** otvara se standardan *Windows Explore*r obrazac, u kojem je potrebno odabrati izvorišnu datoteku baze podataka u formatu *.accdb. Pritiskom na gumb ***Edit Word template location*** također se otvara standardan *Windows Explorer* obrazac u kojem je potrebno odabrati izvorišnu datoteku predloška izvještaja u formatu *.docx

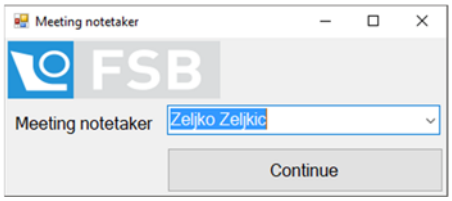
4.2.2. Rezervacija sastanka

Klikom na gumb 1 pokreće se proces rezervacije novog sastanka, te se otvara obrazac prikazan na slici 24. Korištenjem aplikacije moguće je unaprijed rezervirati sastanak, ili pokrenuti izvanredni sastanak, ovisno o izboru gumba koji pokreće željeni proces.



Slika 24. Obrazac za rezervaciju sastanka

Prilikom pokretanja obrasca, aplikacija uspostavlja vezu s bazom podataka, i ovisno o objektu na obrascu popunjava kolekciju odgovarajućeg objekta s podatcima iz odgovarajuće tablice u bazi podataka. Nakon odabira vremena održavanja, lokacije, projekta i grupe sastanaka, korisnik u popisu zaposlenika odabire željene zaposlenike koji će prisustvovati sastanku. Nakon što se korisnik osigura da su upisani podatci točni, odabirom gumba ***Create meeting*** zatvara prikazani obrazac, te vraća korisnika u glavni meni aplikacije. Osnovni podatci o sastanku upisuju se u tablicu ***Meetings***, dok se podatci o prisutnim zaposlenicima upisuju u tablicu ***Meeting_attendants***. Posljednje navedena tablica povezana je relacijama više prema više, te su iz tog razloga u toj tablici pohranjeni samo indeksi primarnih ključeva odgovarajućeg sastanka, te indeksi primarnih ključeva prisutnih zaposlenika. Ukoliko korisnik želi odmah pokrenuti izvanredni sastanak, nakon odabira gumba ***Start meeting*** pokreće se obrazac za odabir zapisničara prikazan na slici 25.

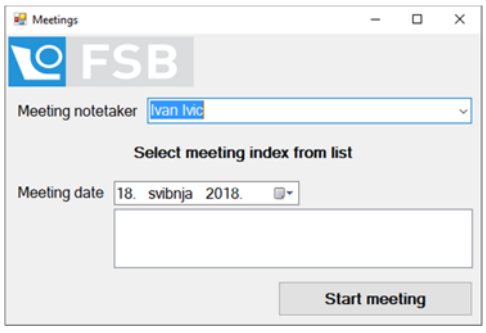


Slika 25. Obrazac za izbor zapisničara

Odabirom zapisničara i pritiskom na gumb ***Continue*** otvara se obrazac za vođenje sastanka, te se u pozadini zapisuju u bazu podataka podatci o zapisničaru u tablicu ***Meetings***. Uz sve odabrane atribute sastanka, također se dodaje atribut statusa sastanka, koji je nakon rezervacije statusa, „pending“, dok se nakon provedenog sastanka mijenja u „finished“.

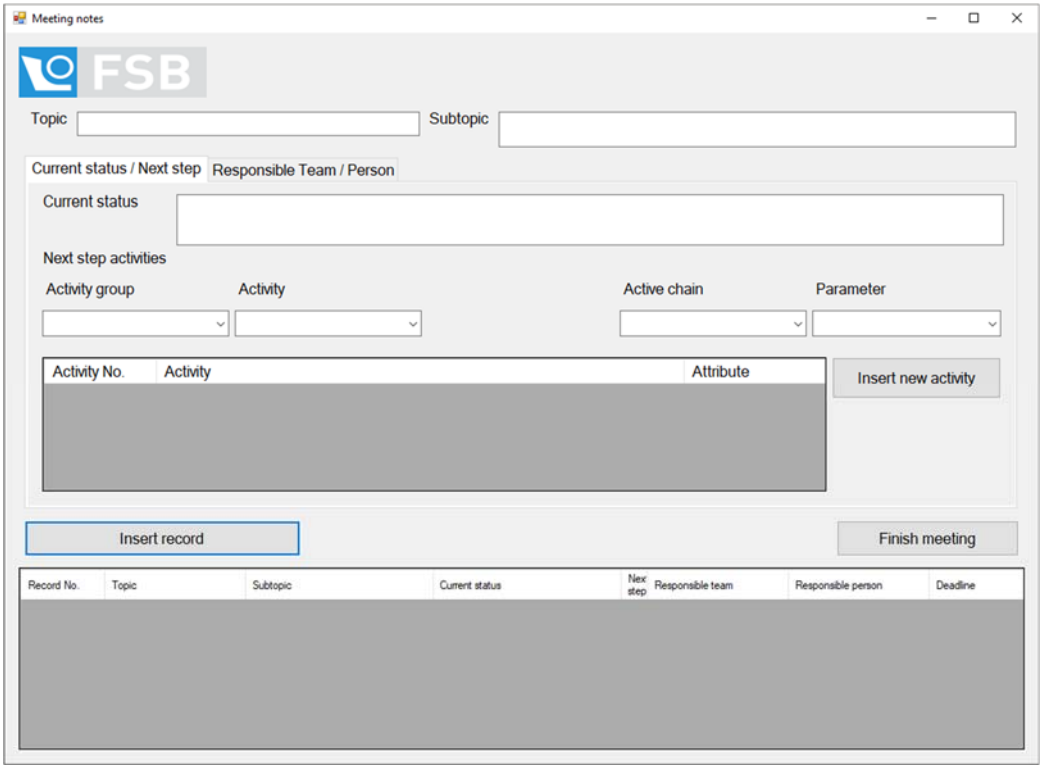
4.2.3. Održavanje sastanka

Izborom gumba za pokretanje sastanka u glavnom meniju otvara se obrazac za izbor sastanka prikazan na slici 26.



Slika 26. Obrazac za izbor sastanka

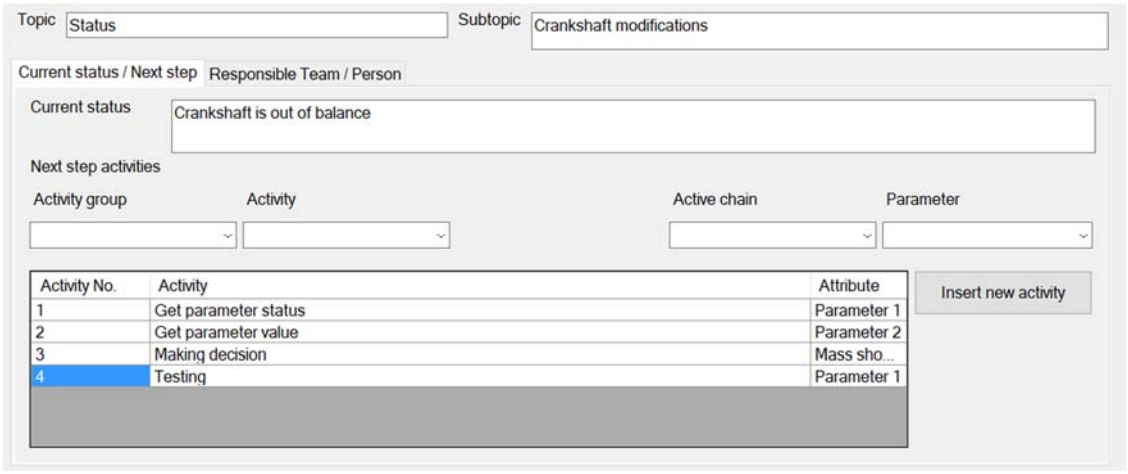
Odabirom odgovarajućeg datuma sastanka osvježava se popis sastanaka rezerviranih za odabrani datum, te je dodatan uvjet pri filtriranju sastanaka status sastanka, koji mora biti „pending“. Također, potrebno je odabrati zapisničara sastanka, i nakon pritiska gumba ***Start meeting*** otvara se obrazac za vođenje sastanka vidljiv na slici 27. Dizajn ovog obrasca je poprilično složen, te će se pojedini moduli obrasca detaljnije opisati u nastavku. Prilikom pokretanja obrasca aplikacija vrši komunikaciju s bazom podataka te popunjava kolekcije objekata za izbor atributa sastanka (hijerarhijska struktura aktivnosti, aktivni lanci parametara i odgovarajući parametri, te popis timova i zaposlenika). U svrhu ubrzavanja rada s obrascem korišteni su padajući izbornici, te u slučaju definiranja teme zapisa sastanka, tekstni okvir s predlaganjem tema na osnovu prethodno upisanih. Korisnik pri upisivanju željene teme u odgovarajući tekstni okvir dobiva ponuđen izbor prethodno upisanih tema sastanaka u bazi podataka upisanih u tablici ***topic_list.*** Ukoliko ne postoji slična tema, nakon prelaska na novi zapis sastanka, upisana tema sastanka se upisuje u prethodno navedenu tablicu.



Slika 27. Obrazac za vođenje sastanka

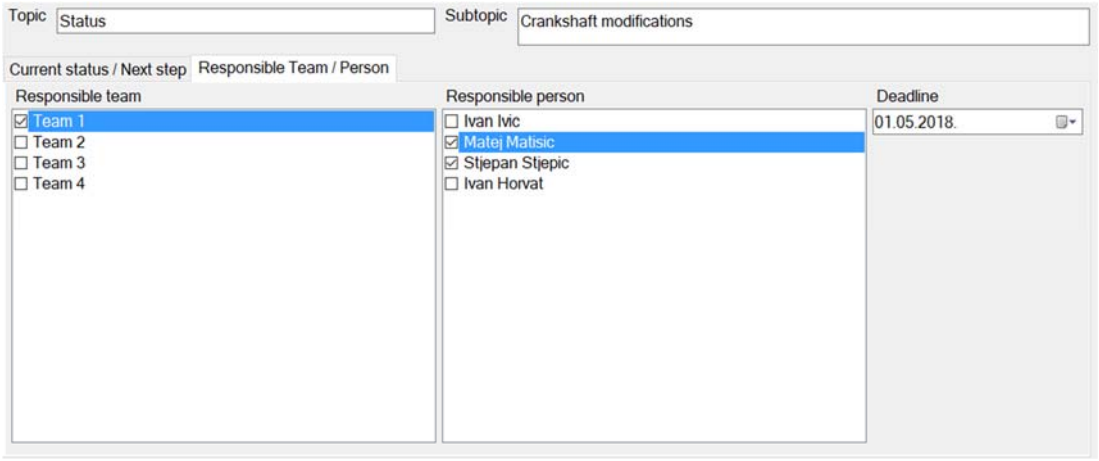
Svaka tema ima trenutno stanje, te nakon opisa trenutnog stanja u odgovarajućem tekstnom okviru, korisnik upisuje provedene aktivnosti tokom sastanka vezane uz odabranu temu. Odabirom odgovarajuće kombinacije zapisa unutar kombiniranih okvira za definiranje atributa

aktivnosti te pritiskom na gumb ***Insert new activity***, odabrani atributi se zapisuju u tablicu unutar obrasca, gdje je moguće na vrlo lak način uređivati atribute pojedinih aktivnosti. Nakon odabira aktivnosti, opisana tablica prikazana je na slici 28.



Slika 28. Aktivnosti provedene tokom sastanka

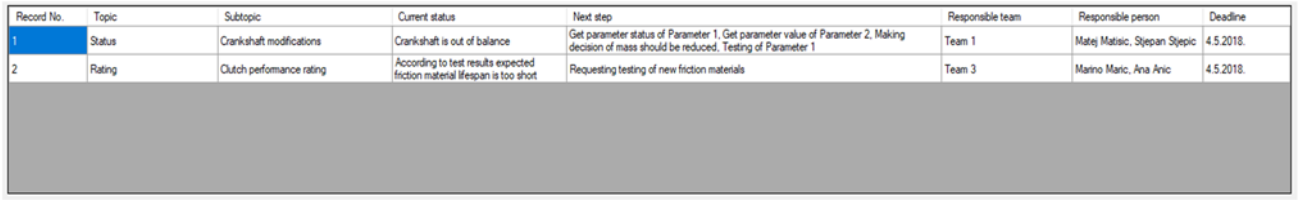
Za skup odabranih aktivnosti potrebno je definirati odgovorne timove i osobe, te krajnjeg roka izvršenja, što se definira unutar izbornika ***Responsible Team / Person***, koji je prikazan na slici 29.



Slika 29. Odabir odgovornih osoba i roka izvršenja aktivnosti

Odabirom prethodno opisanih atributa, te pritiskom gumba ***Insert new record*** ažurira se tablica zapisa sa sastanka, te se omogućuje korisniku prelazak na novu temu. Također, uz sve prethodno navedene radnje, upisuju se odabrane aktivnosti u bazu podataka u tablicu ***Meeting_activities***. Kao što je u prethodnom poglavlju opisano, upisom odgovarajućih podataka u navedenu tablicu omogućuje se naknadno automatsko aktiviranje odgovarajućeg CPN modela. Bitno je napomenuti da je potrebno definirati sve atribute zapisa, te ukoliko korisnik zaboravi definirati određeni atribut, a pokuša pokrenuti novu temu, aplikacija ga

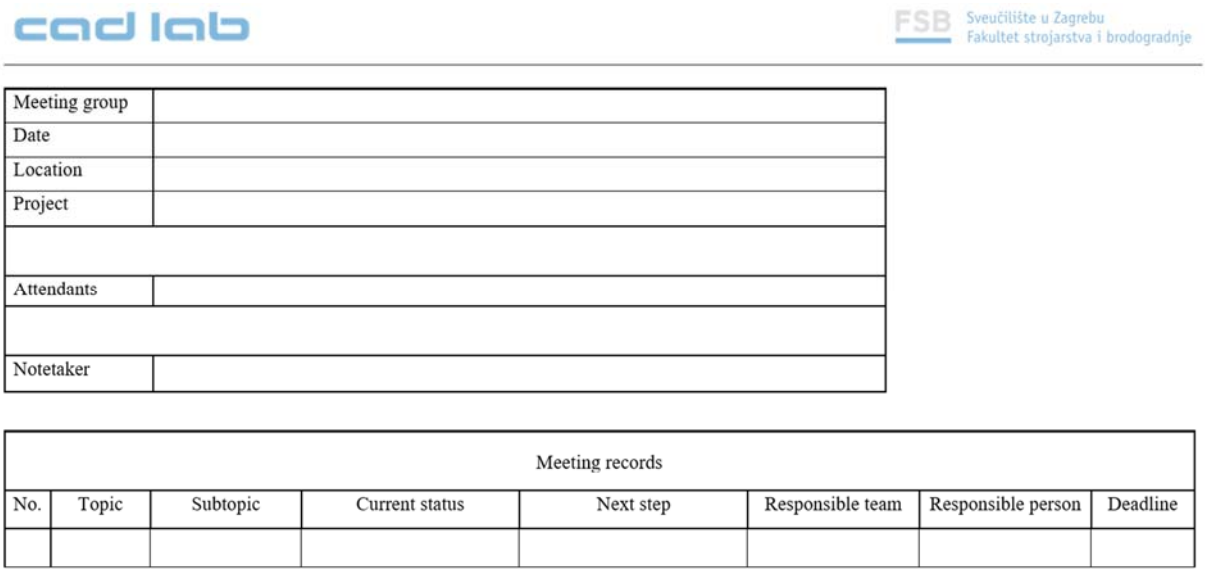
obavještava i spriječi u prelasku na novu temu, do onog trenutka kada su definirani svi potrebni atributi. Opisane radnje mogu se ponavljati po potrebi, te nakon nekoliko diskutiranih tema popis zapisa prikazan je na slici 30.



Record No.	Topic	Subtopic	Current status	Next step	Responsible team	Responsible person	Deadline
1	Status	Crankshaft modifications	Crankshaft is out of balance	Get parameter status of Parameter 1, Get parameter value of Parameter 2, Making decision of mass should be reduced, Testing of Parameter 1	Team 1	Matej Matisic, Stjepan Stjepic	4.5.2018.
2	Rating	Clutch performance rating	According to test results expected friction material lifespan is too short	Requesting testing of new friction materials	Team 3	Marino Maric, Ana Anic	4.5.2018.

Slika 30. Evidencija zapisa na sastanku

Prema potrebi, moguće je vršiti izmjene pojedinih ćelija u prikazanoj tablici, te nakon što je korisnik zadovoljan svim zapisima, klikom na gumb ***Finish meeting*** završava sastanak. Nakon završetka sastanka korisniku se otvara popunjeni izvještaj u *.docx formatu, te se u pozadini završava upis svih potrebnih atributa u bazu podataka. Također, status sastanka se mijenja iz „pending“ u „finished“.Aplikacija pomoću knjižnih oznaka u predlošku izvještaja upisuje odgovarajuće podatke na odgovarajuće pozicije unutar predloška, te je bitno naglasiti da je za pravilan rad aplikacije potrebno osigurati pravilno pripremljen predložak u ***MS Word*** aplikaciji koji je vidljiv na sljedećoj slici.



cad lab

FSB Sveučilište u Zagrebu
Fakultet strojarstva i brodogradnje

Meeting group	
Date	
Location	
Project	
Attendants	
Notetaker	

Meeting records							
No.	Topic	Subtopic	Current status	Next step	Responsible team	Responsible person	Deadline

Slika 31. Predložak izvještaja sa sastanka

U nastavku prikazan je izvještaj simuliranog sastanka, te su svi podatci generirani tokom održanog simuliranog sastanka prikazani u tablicama 29. i 30.

Tablica 29. Osnovni podatci o održanom sastanku

Meeting group	Preliminary design review
Date	7.5.2018.
Location	Meeting room 1
Project	Project 1
Attendants	Ivan Ivic, Matej Matisic, Stjepan Stjepic, Marko Markic, Zeljko Zeljkic, Ivan Horvat, Marino Maric
Notetaker	Ivan Horvat

Tablica 30. Zapisi na sastanku

Meeting records							
No.	Topic	Subtopic	Current status	Next step	Responsible team	Responsible person	Deadline
1	Status	Crankshaft modifications	Crankshaft is out of balance	Get parameter status of Parameter 1, Get parameter value of Parameter 2, Making decision of Mass should be reduced significantly	Team 1	Matej Matisic, Stjepan Stjepic	11.5.2018.
2	Rating	Clutch performance rating	According to test results expected friction material lifespan is too short	Requesting testing of new friction materials	Team 3	Marino Maric, Ana Anic	11.5.2018.

3	Assessment	Connection rods supplier	Current supplier has too many faulty delivered parts	Informing about general information for forged parts suppliers, Evaluating of test parts from new suppliers, Making decision - selecting new supplier	Team 2, Team 3	Zeljko Zeljkic, Marino Maric, Ana Anic	11.5.2018.
4	Requirements	Production requirements for new drivetrain platform	New platform is in final stages of design	Testing of production capabilities	Team 3	Ivana Ivanic	11.5.2018.
5	Status	Infotainment software updates	There are few existing bugs in navigation module	Requesting information of reported bugs, Resolving general issue of reported bugs	Team 1	Ivan Horvat	12.5.2018.
6	Assessment	1.9 TDI engine oil consumption	New engine oil consuption is not known	Requesting information of old engine oil consumption, Testing of new engine	Team 3	Marino Maric	16.5.2018.
7	Report	Seat upholstering materials	Seat upholestering materials are not yet decided	Reporting from other meeting, Making decision - new selected upholstering fabric	Team 2	Zeljko Zeljkic	10.5.2018.
8	Rating	Engine mounts stability	FEM simulations show too big displacements of engine mounts during high engine loads	Presenting results of FEM simulations of engine mounts, Evaluating of new engine mount design, Scheduling meeting – 18.05.2018. 8:00	Team 2	Ivana Ivanic	18.5.2018.

5. ZAKLJUČAK

U ovom radu postavljeni su čvrsti temelji za daljnji razvoj metodologije za upravljanje podatcima generiranima na sastancima i nakon sastanaka. U uvodnom dijelu su opisani aspekti sastanaka, aktivnosti na sastancima, te Petri mreže kao metoda simuliranja stvarnih procesa. U nastavku je opisana hijerarhijska struktura aktivnosti definirana na analizi studije slučaja održanoj unutar tvrtke Daimler AG, te su detaljno opisane aplikacija za izradu izvještaja sa sastanka te pripadajuća baza podataka.

Na temelju definirane strukture aktivnosti omogućeno je formiranje čitavog skupa CPN modula koji će se aktivirati ovisno o provedenim aktivnosti na sastanku. Postignuta je zadovoljavajuća struktura baze podataka koja omogućuje jednostavan programski pristup potrebnim podatcima za povezivanje odgovarajućeg CPN modula s provedenom aktivnošću na sastanku. Potencijalno unapređenje opisanog procesa je u implementaciji opisane baze podataka u sam PDM/PLM sustav koja će omogućiti pristup većoj količini informacija, što je i konačni cilj ove teme, a to je kreiranje polu automatskog sustava koji će samostalno upravljati svim informacijama koje se generiraju tokom cijelog procesa razvoja proizvoda.

Razvijena aplikacija zadovoljava postavljene zahtjeve, no da bi izvještaji bili kvalitetniji potrebno je razviti čitav skup lingvistički pravila koji će omogućiti kvalitetnije formiranje zapisa. O ovom stanju aplikacije postavljeni su temelji za uvođenje lingvističkih pravila prilikom formiranja zapisa, te je potrebno u bazi podataka kreirati odgovarajuće tablice u kojima će lingvistička pravila biti zapisana. Nakon uvođenja lingvističkih pravila, dodatno će se smanjiti potrebno uređivanje pojedinih zapisa.

Također je potrebno definirati strukturu atributa pojedinih aktivnosti na sastanku, što u ovom stanju aplikacije nije u potpunosti riješeno, no odabranim pristupom razvojnog sučelja aplikacije omogućeno je vrlo jednostavno dodavanje novih modula u samu aplikaciju.

LITERATURA

[1] Romano, N.C. Jr., Nunamaker, J. F. Jr., Meeting analysis: Findings from research and practice, University of Tulsa, 2001

[2] Huet,G., Culley, S. J., McMahon, C. A., Fortin, C., Making sense of design review activities, University of Bath, 2007.

[3] Olson, G. M., Olson, J. S., Carter, R. M., Storrøsten, M., Small group design meetings: An analysis of collaboration, University of Michigan, 1992

[4] Sim, S. K., Duffy, A. H. B., Towards an ontology of generic design activities, Springer-Verlag London Limited 2003

[5] Radošević, D.: Petri mreže i njihov odnos prema ostalim mrežnim tehnikama Fakultet organizacije i informatike, Varaždin 1991

[6] Jensen, K., Kristensen, L. M.,Wells L.: Coloured Petri Nets and CPN Tools for Modelling and Validation of Concurrent Systems

[7] Codd, E. F., The relational model for database management, Addison-Wesley Publishing Company, Reading, Massachusets, 1990

PRILOZI

I. CD-R disk

II. Programski kod aplikacije

```vb
Public Class application_settings_form
    'Prikazivanje trenutnih postavki aplikacije u otvorenoj formi.
    Private Sub application_settings_form_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        database_location_label.Text = Globals.db_connection_string
        word_template_location_label.Text = Globals.report_template_string
    End Sub

    ' Pritiskom na gumb "Edit database connection properties" pokreće se dijaloški okvir u kojem se odabire lokacija baze podataka, te se nakon toga upisuje odgovarajuća putanja baze podataka u odgovarajuću globalnu varijablu.
    Private Sub edit_database_connection_properties_button_Click(sender As Object, e As EventArgs) Handles edit_database_connection_properties_button.Click

        'Instanciranje objekta dijaloškog okvira za izbor željene datoteke.
        Dim database_picker_dialog As OpenFileDialog = New OpenFileDialog()
        With database_picker_dialog
            .Title = "Choose location of database"
            .InitialDirectory = "C:\"
            .RestoreDirectory = True
        End With

        'Ukoliko korisnik odabere željenu datoteku pritiskom na tipku OK mijenja globalne postavke veze između aplikacije i baze podataka.
        If database_picker_dialog.ShowDialog() = DialogResult.OK Then
            Globals.db_connection_string = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source=" & database_picker_dialog.FileName & ""
        End If
        database_location_label.Text = Globals.db_connection_string
    End Sub

    'Pritiskom na gumb "Edit Word template location" pokreće se dijaloški okvir u kojem se odabire lokacije Word predloška, te se nakon toga upisuje odgovarajuća putanja predloška u odgovarajuću globalnu varijablu.
    Private Sub edit_word_template_button_Click(sender As Object, e As EventArgs) Handles edit_word_template_button.Click

        'Instanciranje objekta dijaloškog okvira za izbor željene datoteke
        Dim word_template_dialog As OpenFileDialog = New OpenFileDialog()
        With word_template_dialog
            .Title = "Choose location of report template"
            .InitialDirectory = "C:\"
            .RestoreDirectory = True
        End With

        'Ukoliko korisnik odabere željenu datoteku i pritiskom na tipku OK mijenja globalne postavke veze između aplikacije i Word aplikacije
        If word_template_dialog.ShowDialog() = DialogResult.OK Then
            Globals.report_template_string = word_template_dialog.FileName
        End If
        word_template_location_label.Text = Globals.report_template_string
    End Sub

    'Povratak korisnika na glavni meni aplikacije.
    Private Sub main_menu_button_Click(sender As Object, e As EventArgs) Handles main_menu_button.Click
        My.Forms.application_settings_form.Close()
```

```
47          My.Forms.Meeting_report_generator.Show()
48      End Sub
49  End Class
```

```
1  Public Class Globals
2      Public Shared db_connection_string As String = ("Provider=Microsoft.ACE.OLEDB.12.0;Data
       Source=C:\Users\Tomislav\Dropbox\Diplomski Breški Tomislav\FSB_dummy_company.accdb")
3      Public Shared report_template_string As String = "C:\Users\Tomislav\Dropbox\Diplomski Breški
       Tomislav\Daimler_report_template.docx"
4      Public Shared meeting_ID As Integer = 0
5  End Class
6
```

```vb
Public Class Meeting_report_generator

    'Korisnik ovisno o izboru pokreće željeni proces.

    'Pokretanje procesa kreiranja novog sastanka.
    Private Sub create_meeting_button_Click(sender As Object, e As EventArgs) Handles
    create_meeting_button.Click
        Me.Hide()
        My.Forms.new_meeting_form.Show()
    End Sub

    'Pokretanje sastanka prema odabiru korisnika.
    Private Sub start_meeting_button_Click(sender As Object, e As EventArgs) Handles
    start_meeting_button.Click
        Me.Hide()
        My.Forms.meeting_picker_form.Show()
    End Sub

    'Zatvaranje aplikacije.
    Private Sub exit_button_Click(sender As Object, e As EventArgs) Handles exit_button.Click
        Application.Exit()
    End Sub

    'Pokretanje forme za definiranje parametara za rad aplikacije (Baza podataka i predložak izvješća)
    Private Sub application_settings_button_Click(sender As Object, e As EventArgs) Handles
    application_settings_button.Click
        Me.Hide()
        My.Forms.application_settings_form.Show()
    End Sub
End Class

```

```
1  Imports word = Microsoft.Office.Interop.Word
2  Public Class meeting_form
3      Dim connection As New OleDb.OleDbConnection
4      Dim command As New OleDb.OleDbCommand
5      Dim reader As OleDb.OleDbDataReader
6      Dim wordapp As word.Application
7      Dim worddoc As word.Document
8
9      Dim lvl_2_parent_ID As Integer
10     Dim lvl_3_parent_ID As Integer
11     Dim active_chain_ID As Integer
12     Dim team_ID As Integer
13     Dim employee_ID As Integer
14     Dim project_ID As Integer
15     Dim location_ID As Integer
16
17     Dim activity_counter As Integer
18     Dim record_counter As Integer
19     Dim activity_ID As Integer
20     Dim record_ID As Integer
21     Dim initial_record_ID
22
23
24     Private Sub meeting_form_Load(sender As Object, e As EventArgs) Handles MyBase.Load
25         activity_counter = 1
26         record_counter = 1
27
28         'Povezivanje aplikacije s bazom podataka.
29         With connection
30             .ConnectionString = Globals.db_connection_string
31             .Open()
32         End With
33
34         'Popunjavanje auto-complete kolekcije objekta za unos naziva željene grupe tema.
35         With command
36             .CommandText = "SELECT * FROM Topic_list"
37             .Connection = connection
38         End With
39
40         reader = command.ExecuteReader
41         While reader.Read
42             topic_tbox.AutoCompleteCustomSource.Add(reader("topic_description"))
43         End While
44         reader.Close()
45
46         'Popunjavanje kolekcije objekta za izbor aktivnosti na najvišoj hijerarhijskoj razini.
47         With command
48             .CommandText = "SELECT * FROM Activities_list WHERE [Activity_level] = 1"
49         End With
50         reader = command.ExecuteReader
51         While reader.Read
52             activity_group_cbox.Items.Add(reader("Activity_name"))
53         End While
54         reader.Close()
55
56         'Sakrivanje objekta za izbor aktivnosti na najnižoj hijerarhijskoj razini. Da bi se aktivirao objekt
```

```
        potrebno je odabrati odgovarajući izbor koji na drugoj razini hijerarhije aktivnosti ima aktivnosti na nižoj razini.
57      lvl_3_activity_cbox.Hide()
58
59      'Popunjavanje kolekcije objekta za izbor aktivnog lanca parametara.
60      command.CommandText = "SELECT * FROM Active_chains"
61      reader = command.ExecuteReader
62      While reader.Read
63          active_chain_cbox.Items.Add(reader("active_chain_name"))
64      End While
65      reader.Close()
66
67      'Popunjavanje kolekcije objekta za izbor odgovornih timova za odgovarajući zadatak.
68      command.CommandText = "SELECT * FROM Teams"
69      reader = command.ExecuteReader
70      While reader.Read
71          responsible_team_checked_listbox.Items.Add(reader("team_name"))
72      End While
73      reader.Close()
74
75      'U bazi podataka onemogućeno je samonumeriranje primarnog ključa record_ID, te da bi se izbjeglo dupliciranje primarnog ključa potrebno je odrediti broj postojećih slogova u tablici Meeting_records.
76      command.CommandText = "SELECT COUNT(*) FROM Meeting_records"
77      record_ID = command.ExecuteScalar + 1
78      initial_record_ID = record_ID
79  End Sub
80
81  'Prilikom učitavanja forme objekti za odabir aktivnosti na nižim razinama su prazni, te ih je potrebno popunjavati prema hijerarhijskoj strukturi aktivnosti definiranoj u tablici Activities_list.
82  Private Sub activity_group_cbox_SelectedIndexChanged(sender As Object, e As EventArgs) Handles activity_group_cbox.SelectedIndexChanged
83
84      'Prilikom svake promjene objekta za izbor aktivnosti najviše razine objekti za izbor aktivnosti na drugoj i trećoj razini se prazne.
85      With lvl_2_activity_cbox
86          .Text = ""
87          .Items.Clear()
88      End With
89      lvl_3_activity_cbox.Text = ""
90
91      'Određivanje indeksa primarnog ključa odabrane grupe aktivnosti.
92      With command
93          .CommandText = "SELECT Activity_ID FROM Activities_list WHERE [Activity_name] = '" & activity_group_cbox.Text & "'"
94      End With
95      lvl_2_parent_ID = command.ExecuteScalar
96
97      'Popunjavanje kolekcije objekta za odabir aktivnosti na drugoj razini hijerarhije s aktivnostima kojima je indeks roditelja jednak indeksu odabrane grupe aktivnosti.
98      command.CommandText = "SELECT * FROM Activities_list WHERE [Parent_ID]= " & lvl_2_parent_ID & ""
99      reader = command.ExecuteReader
100     While reader.Read
101         lvl_2_activity_cbox.Items.Add(reader("Activity_name"))
102     End While
103     reader.Close()
```

```
104     End Sub
105
106     Private Sub lvl_2_activity_cbox_SelectedIndexChanged(sender As Object, e As EventArgs) Handles
        lvl_2_activity_cbox.SelectedIndexChanged
107
108         ' Prilikom svake promjene odabira na objektu za izbor aktivnosti na drugoj hijerarhijskoj razini briše
            se kolekcija odabira na objektu za izbor aktivnosti hijerarhijske razine 3.
109         With lvl_3_activity_cbox
110             .Text = ""
111             .Items.Clear()
112         End With
113
114         'Određivanje indeksa primarnog ključa odabrane aktivnosti na drugoj hijerarhijskoj razini.
115         With command
116             .CommandText = "SELECT Activity_ID FROM Activities_list WHERE [Activity_name] = '" &
                lvl_2_activity_cbox.Text & "'"
117         End With
118         lvl_3_parent_ID = command.ExecuteScalar
119
120         'Popunjavanje kolekcije objekta za odabir aktivnosti na trećoj hijerarhijskoj razini s aktivnostima
            kojima je indeks roditelja jednak indeksu odabrane aktivnosti na hijerarhijskoj razini 2.
121         command.CommandText = "SELECT * FROM Activities_list WHERE [Parent_ID]= " &
            lvl_3_parent_ID & ""
122         reader = command.ExecuteReader
123         While reader.Read
124             lvl_3_activity_cbox.Items.Add(reader("Activity_name"))
125         End While
126         reader.Close()
127
128         'Ako ne postoje odgovarajuće aktivnosti na trećoj razini hijerarhije, skriva se objekt za izbor aktivnosti
            na trećoj razini.
129         If lvl_3_activity_cbox.Items.Count = 0 Then
130             lvl_3_activity_cbox.Hide()
131         Else
132             lvl_3_activity_cbox.Show()
133         End If
134     End Sub
135
136     'Nakon odabira odgovarajuće komabinacije aktivnosti i atributa aktivnosti i pritiskom na gumb "Add new
        activity" popunjava se tablica activity_datagrid te nakon toga se brišu postojeći izbori u objektima.
137     Private Sub new_activity_button_Click(sender As Object, e As EventArgs) Handles
        new_activity_button.Click
138         activity_datagrid.Rows.Add()
139         activity_datagrid(0, activity_counter - 1).Value = activity_counter
140
141         'Ako ne postoji aktivnost na trećoj hijerarhijskoj razini u tablicu se upisuje odabrana aktivnost na
            drugoj razini, dok ukoliko postoji aktivnost na trećoj razini u tablicu se upisuje aktivnost treće razine.
142         If lvl_3_activity_cbox.Items.Count = 0 Then
143             activity_datagrid(1, activity_counter - 1).Value = lvl_2_activity_cbox.Text
144         Else
145             activity_datagrid(1, activity_counter - 1).Value = lvl_3_activity_cbox.Text
146         End If
147
148         activity_datagrid(2, activity_counter - 1).Value = parameter_cbox.Text
149         activity_counter += 1
150         activity_group_cbox.Text = ""
```

```
151         lvl_2_activity_cbox.Text = ""
152         lvl_3_activity_cbox.Hide()
153         active_chain_cbox.Text = ""
154         parameter_cbox.Text = ""
155     End Sub
156 
157     'U bazi podataka definirani su aktivni lanci parametara, te je iz tog razloga potrebno učitati odgovarajuće parametre u kolekciju objekta za izbor parametara, ovisno o indeksu primaranog ključa aktivnog lanca.
158     Private Sub active_chain_cbox_SelectedIndexChanged(sender As Object, e As EventArgs) Handles active_chain_cbox.SelectedIndexChanged
159 
160         'Svakom promjenom izbora aktivnog lanca briše se kolekcija objekta za izbor parametara.
161         With parameter_cbox
162             .Text = ""
163             .Items.Clear()
164         End With
165 
166         'Određivanje indeksa primarnog ključa odabranog aktivnog lanca parametra
167         command.CommandText = "SELECT active_chain_ID FROM Active_chains WHERE [active_chain_name] = '" & active_chain_cbox.Text & "'"
168         active_chain_ID = command.ExecuteScalar
169 
170         'Popunjavanje kolekcije objekta za odabir parametara ovisno o indeksu aktivnog lanca parametara.
171         command.CommandText = "SELECT * FROM [Project_parameters] WHERE [active_chain_ID] = " & active_chain_ID & ""
172         reader = command.ExecuteReader
173         While reader.Read
174             parameter_cbox.Items.Add(reader("parameter_name"))
175         End While
176         reader.Close()
177     End Sub
178 
179     'Kolekcija za odabir odgovornih članova tima mijenja se ovisno o odabranim timovima. Prilikom svake promjene odabranog tima mijenja se mogućnost izbora odgovornih zaposlenika.
180     Private Sub responsible_team_checked_listbox_SelectedIndexChanged(sender As Object, e As EventArgs) Handles responsible_team_checked_listbox.SelectedIndexChanged
181         responsible_person_checked_listbox.Items.Clear()
182 
183         'Petlja se ponavlja ovisno o broju odabranih odgovornih tima.
184         For Each item In responsible_team_checked_listbox.CheckedItems
185 
186             'Određivanje indeksa primarnog ključa odabranog tima.
187             command.CommandText = "SELECT team_ID FROM Teams WHERE team_name = '" & item.ToString() & "'"
188             team_ID = command.ExecuteScalar()
189 
190             'Popunjavanje kolekcije objekta za izbor odgovorne osobe ovisno o primarnom ključu odabranog tima.
191             command.CommandText = "SELECT * FROM Team_employees_joined WHERE team_ID = " & team_ID & ""
192             reader = command.ExecuteReader
193             While reader.Read
194                 responsible_person_checked_listbox.Items.Add(reader("Name"))
195             End While
196             reader.Close()
197         Next
```

```
198     End Sub
199
200     'Nakon popunjavanja svih atributa za pojedini zapis odabrane vrijednosti se nakon pritiska na gumb
        "Insert record" upisuju u bazu podataka, te se prikazuju u tablici meeting_records_table.
201     Private Sub insert_record_button_Click(sender As Object, e As EventArgs) Handles
        insert_record_button.Click
202
203         'Ukoliko korisnik zaboravi odabrati odgovorne timove i osobe spriječen je nastavak izvođenja
            sljedećeg skupa naredbi.
204         If responsible_person_checked_listbox.CheckedItems.Count = 0 Or
            responsible_team_checked_listbox.CheckedItems.Count = 0 Then
205             MsgBox("Please check responsible staff for this record")
206             Exit Sub
207         End If
208
209         'Slaganje teksta aktivnosti ovisno o odabranim aktivnosti pojedinog zapisa
210         Dim activity_string_builder As New System.Text.StringBuilder
211         For i As Integer = 0 To activity_datagrid.RowCount - 1
212             activity_string_builder.Append(activity_datagrid(1, i).Value & " of " & activity_datagrid(2,
                i).Value & ", ")
213         Next
214
215         'Slaganje teksta odabranih odgovornih timova
216         Dim responsible_team_string_builder As New System.Text.StringBuilder
217         For i As Integer = 0 To responsible_team_checked_listbox.CheckedItems.Count - 1
218
                responsible_team_string_builder.Append(responsible_team_checked_listbox.CheckedItems(i).ToStri
                ng() & ", ")
219         Next
220
221         'Slaganje teksta odabranih odgovornih osoba
222         Dim responsible_person_string_builder As New System.Text.StringBuilder
223         For i As Integer = 0 To responsible_person_checked_listbox.CheckedItems.Count - 1
224
                responsible_person_string_builder.Append(responsible_person_checked_listbox.CheckedItems(i).To
                String() & ", ")
225         Next
226
227         'Popunjavanje tablice meeting_records_table
228         meeting_records_table.Rows.Add()
229         meeting_records_table(0, record_counter - 1).Value = record_counter
230         meeting_records_table(1, record_counter - 1).Value = topic_tbox.Text
231         meeting_records_table(2, record_counter - 1).Value = subtopic_tbox.Text
232         meeting_records_table(3, record_counter - 1).Value = current_status_tbox.Text
233         meeting_records_table(4, record_counter - 1).Value = activity_string_builder.ToString.Substring(0,
            activity_string_builder.ToString.Length - 2)
234         meeting_records_table(5, record_counter - 1).Value =
            responsible_team_string_builder.ToString.Substring(0,
            responsible_team_string_builder.ToString.Length - 2)
235         meeting_records_table(6, record_counter - 1).Value =
            responsible_person_string_builder.ToString.Substring(0,
            responsible_person_string_builder.ToString.Length - 2)
236         meeting_records_table(7, record_counter - 1).Value = deadline_picker.Value.ToShortDateString
237
238         'Rezervacija indeksa primarnog ključa u tablici Meeting_records
239         With command
```

```
240         .CommandText = "INSERT INTO Meeting_records (record_ID, meeting_ID) VALUES (" &
            record_ID & ", " & Globals.meeting_ID & ")"
241         .ExecuteNonQuery()
242     End With
243
244     'Ubacivanje odabranog skupa aktivnosti u bazu podataka u tablicu Record_activities
245     For i As Integer = 0 To activity_datagrid.RowCount - 1
246         command.CommandText = "SELECT activity_ID FROM activities_list WHERE activity_name =
            '" & activity_datagrid(1, i).Value & "'"
247         activity_ID = command.ExecuteScalar
248         With command
249             .CommandText = "INSERT INTO Record_activities VALUES (" & record_ID & "," &
                activity_ID & ")"
250             .ExecuteNonQuery()
251         End With
252     Next
253
254     'Ubacivanje odabranog skupa odgovornih timova i osoba u tablicu odgovornih članova za pojedini
        zapis Record_responsibility
255     For i As Integer = 0 To responsible_person_checked_listbox.CheckedItems.Count - 1
256         command.CommandText = "SELECT employee_ID FROM Team_employees_joined WHERE
            [Name] = '" & responsible_person_checked_listbox.CheckedItems(i).ToString() & "' "
257         employee_ID = command.ExecuteScalar
258         command.CommandText = "SELECT team_ID FROM Team_employees_joined WHERE [Name] =
            '" & responsible_person_checked_listbox.CheckedItems(i).ToString() & "' "
259         team_ID = command.ExecuteScalar
260         With command
261             .CommandText = "INSERT INTO Record_responsibility VALUES (" & record_ID & "," &
                team_ID & "," & employee_ID & ", '" & deadline_picker.Value.ToShortDateString & "')"
262             .ExecuteNonQuery()
263         End With
264     Next
265
266     'Određivanje postoji li u bazi podataka unešena tema sastanka.
267     command.CommandText = "SELECT COUNT(*) FROM Topic_list WHERE topic_description = ?"
268     command.Parameters.AddWithValue("@p1", topic_tbox.Text)
269
270     'Ukoliko postoji u bazi slog s odabranom temom sljedeći skup naredbi se preskače, a ako ne postoji u
        bazu se upisuje odabrana tema, te će se prilikom sljedećeg korištenja pojaviti kao prijedlog teme
        sastanka.
271     If command.ExecuteScalar = 0 Then
272         With command
273             .CommandText = "INSERT INTO Topic_list ([topic_description]) VALUES (?)"
274             .Parameters.AddWithValue("@p1", topic_tbox.Text)
275             .ExecuteScalar()
276         End With
277     End If
278
279     'Povećanje brojača zapisa, te indeksa primarnog ključa zapisa, te vraćanje svih objekata forme u
        izvorno stanje za izbor atributa sljedećeg zapisa.
280     record_counter += 1
281     record_ID += 1
282     topic_tbox.Text = ""
283     subtopic_tbox.Text = ""
284     current_status_tbox.Text = ""
285     activity_group_cbox.Text = ""
```

```
286         lvl_2_activity_cbox.Text = ""
287         lvl_3_activity_cbox.Hide()
288         active_chain_cbox.Text = ""
289         parameter_cbox.Text = ""
290         activity_datagrid.Rows.Clear()
291         activity_counter = 1
292         tab_control.SelectTab(0)
293
294         'Brisanje odabira odgovornih osoba
295         For i As Integer = 0 To responsible_team_checked_listbox.Items.Count - 1
296             responsible_team_checked_listbox.SetItemChecked(i, False)
297         Next
298         For i As Integer = 0 To responsible_person_checked_listbox.Items.Count - 1
299             responsible_person_checked_listbox.SetItemChecked(i, False)
300         Next
301     End Sub
302
303     'Nakon što su upisane svi atributi pojedinih zapisa sastanaka kreira se Word dokument u kojem su
        navedeni svi bitni podatci sa sastanka ovisno o odabirima atributa. Također, u tablici Meetings
        popunjavaju se preostali podatci vezani uz sastanak.
304     Private Sub Finish_meeting_button_Click(sender As Object, e As EventArgs) Handles
        Finish_meeting_button.Click
305         'Instanciranje objekata vezanih uz Word aplikaciju, te učitavanje predloška zapisnika
306         wordapp = CreateObject("Word.Application")
307         wordapp.Visible = True
308         worddoc = wordapp.Documents.Add(Globals.report_template_string)
309
310         'Ubacivanje preostalih podataka u bazi podataka za pojedini zapis sastanka
311         For i As Integer = 0 To meeting_records_table.RowCount - 1
312             With command
313                 .CommandText = "UPDATE Meeting_records SET record_topic = '" & meeting_records_table(1,
                    i).Value & "', record_subtopic = '" & meeting_records_table(2, i).Value & "',
                    record_current_status = '" & meeting_records_table(3, i).Value & "', record_next_step = '" &
                    meeting_records_table(4, i).Value & "' WHERE (record_ID = " & (initial_record_ID + i) & ")"
314                 .ExecuteNonQuery()
315             End With
316         Next
317
318         'Popunjavanje osnovnih podataka o sastanku na odgovarajućim knjižnim oznakama predloška zapisnika
319         command.CommandText = "SELECT meeting_group FROM Meetings WHERE meeting_ID = " &
            Globals.meeting_ID & ""
320         worddoc.Bookmarks("meeting_group").Range.Text = command.ExecuteScalar
321         worddoc.Bookmarks("meeting_date").Range.Text = Format("f", Date.Today())
322
323         command.CommandText = "SELECT location_ID FROM Meetings WHERE meeting_ID = " &
            Globals.meeting_ID & ""
324         location_ID = command.ExecuteScalar
325         command.CommandText = "SELECT location_name FROM Locations WHERE location_ID = " &
            location_ID & ""
326         worddoc.Bookmarks("meeting_location").Range.Text = command.ExecuteScalar
327
328         command.CommandText = "SELECT project_ID FROM Meetings WHERE meeting_ID = " &
            Globals.meeting_ID & ""
329         project_ID = command.ExecuteScalar
330         command.CommandText = "SELECT project_name FROM Projects WHERE project_ID = " &
            project_ID & ""
```

```
331         worddoc.Bookmarks("meeting_project").Range.Text = command.ExecuteScalar
332
333         'Slaganje teksta prisutnih zaposlenika, te ubacivanje teksta na odgovarajuću lokaciju u zapisniku
334         command.CommandText = "SELECT * FROM Meeting_attendants_joined WHERE meeting_ID = "
            & Globals.meeting_ID & ""
335         reader = command.ExecuteReader
336         Dim attendants_string As New System.Text.StringBuilder
337         While reader.Read
338             attendants_string.Append(reader("Name") & ", ")
339         End While
340         reader.Close()
341         worddoc.Bookmarks("meeting_attendants").Range.Text = attendants_string.ToString.Substring(0,
            attendants_string.ToString.Length - 2)
342
343         command.CommandText = "SELECT meeting_notetaker FROM Meetings WHERE meeting_ID = "
            & Globals.meeting_ID & ""
344         worddoc.Bookmarks("meeting_notetaker").Range.Text = command.ExecuteScalar
345
346         'Popunjavanje tablice u predlošku zapisnika s atributima pojedinog zapisa.
347         For i As Integer = 0 To meeting_records_table.RowCount - 1
348             worddoc.Tables(2).Cell(i + 3, 1).Range.Text = meeting_records_table(0, i).Value
349             worddoc.Tables(2).Cell(i + 3, 2).Range.Text = meeting_records_table(1, i).Value
350             worddoc.Tables(2).Cell(i + 3, 3).Range.Text = meeting_records_table(2, i).Value
351             worddoc.Tables(2).Cell(i + 3, 4).Range.Text = meeting_records_table(3, i).Value
352             worddoc.Tables(2).Cell(i + 3, 5).Range.Text = meeting_records_table(4, i).Value
353             worddoc.Tables(2).Cell(i + 3, 6).Range.Text = meeting_records_table(5, i).Value
354             worddoc.Tables(2).Cell(i + 3, 7).Range.Text = meeting_records_table(6, i).Value
355             worddoc.Tables(2).Cell(i + 3, 8).Range.Text = meeting_records_table(7, i).Value
356             If i = (meeting_records_table.RowCount - 1) Then
357                 Exit For
358             Else
359                 worddoc.Tables(2).Rows.Add()
360             End If
361         Next
362
363         'Promjena atributa sastanka u završeni sastanak.
364         command.CommandText = "UPDATE Meetings SET meeting_status = '" & "finished" & "' WHERE
            meeting_ID = " & Globals.meeting_ID & ""
365         command.ExecuteScalar()
366         My.Forms.meeting_form.Close()
367         My.Forms.Meeting_report_generator.Show()
368     End Sub
369 End Class
```

```vb
Public Class Meeting_notetaker_form
    Dim connection As New OleDb.OleDbConnection
    Dim command As New OleDb.OleDbCommand
    Dim reader As OleDb.OleDbDataReader

    Private Sub Meeting_notetaker_form_Load(sender As Object, e As EventArgs) Handles MyBase.Load

        'Povezivanje aplikacije i baze podataka.
        With connection
            .ConnectionString = Globals.db_connection_string
            .Open()
        End With

        'Popunjavanje kolekcije objekta za odabir zapisničara iz baze zaposlenika
        With command
            .CommandText = "SELECT * FROM Employees"
            .Connection = connection
        End With
        reader = command.ExecuteReader
        While reader.Read
            notetaker_cbox.Items.Add(reader("Name"))
        End While
        reader.Close()
    End Sub

    Private Sub continue_button_Click(sender As Object, e As EventArgs) Handles continue_button.Click

        'Upisivanje zapisničara u bazu podataka u odgovarajući slog s obzirom na prethodno odabrani indeks primarnog ključa sastanka meeting_ID
        With command
            .CommandText = "UPDATE Meetings SET meeting_notetaker = '" & notetaker_cbox.Text & "' WHERE meeting_ID = " & Globals.meeting_ID & ""
            .ExecuteScalar()
        End With
        My.Forms.Meeting_notetaker_form.Close()
        My.Forms.meeting_form.Show()
    End Sub
End Class
```

```
1  Public Class meeting_picker_form
2      Dim connection As New OleDb.OleDbConnection
3      Dim command As New OleDb.OleDbCommand
4      Dim reader As OleDb.OleDbDataReader
5
6      Private Sub meeting_picker_form_Load(sender As Object, e As EventArgs) Handles MyBase.Load
7
8          'Povezivanje aplikacije i baze podataka.
9          With connection
10             .ConnectionString = Globals.db_connection_string
11             .Open()
12         End With
13
14         'Popunjavanje kolekcije objekta za odabir željenog sastanka ovisno o odabranom datumu, te trenutnom
           statusu sastanka.
15         With command
16             .CommandText = "SELECT * FROM Meetings WHERE (meeting_date = #" &
               meeting_date_picker.Value.ToString("dd'/'MM'/'yyyy") & "# AND meeting_status = 'pending')"
17             .Connection = connection
18         End With
19         reader = command.ExecuteReader
20         While reader.Read
21             meeting_group_listbox.Items.Add(reader("meeting_group"))
22         End While
23         reader.Close()
24
25         'Popunjavanje kolekcije objekta za odabir zapisničara iz baze zaposlenika.
26         command.CommandText = "SELECT * FROM Employees"
27         reader = command.ExecuteReader
28         While reader.Read
29             notetaker_cbox.Items.Add(reader("Name"))
30         End While
31         reader.Close()
32     End Sub
33
34     'Ukoliko korisnik pritisne gumb "Start meeting", te ukoliko je odabrao željeni sastanakmi odgovarajućeg
       zapisničara, pokreće se proces vođenja sastanka.
35     Private Sub start_meeting_button_Click(sender As Object, e As EventArgs) Handles
       start_meeting_button.Click
36
37         'Ukoliko korisnik nije odabrao zapisničara sprječava se izvođenje upisa sastanka u bazu, te se
           obavještava korisnik da je potrebno odabrati zapisničara.
38         If notetaker_cbox.Text = "" Then
39             MsgBox("Please pick a notetaker")
40             Exit Sub
41         End If
42
43         'Određivanje indeksa primarnog ključa sastanka iz tablica Meetings na temelju odgovarajućeg datuma
           i grupe sastanaka, te upisivanje u globalnu varijablue meeting_ID
44         command.CommandText = "SELECT meeting_ID from Meetings WHERE meeting_group = '" &
           meeting_group_listbox.SelectedItem & "' AND meeting_date = #" &
           meeting_date_picker.Value.ToString("dd'/'MM'/'yyyy") & "#"
45         Globals.meeting_ID = command.ExecuteScalar
46         With command
47             .CommandText = "UPDATE Meetings SET meeting_notetaker = '" & notetaker_cbox.Text & "'
               WHERE meeting_ID = " & Globals.meeting_ID & ""
```

```
48              .ExecuteScalar()
49          End With
50          My.Forms.meeting_picker_form.Close()
51          My.Forms.meeting_form.Show()
52      End Sub
53
54      'Promjenom datuma na padajućem izborniku pokreće se proces ažuriranja rezerviranih sastanaka tokom odabranog dana.
55      Private Sub meeting_date_picker_ValueChanged(sender As Object, e As EventArgs) Handles meeting_date_picker.ValueChanged
56          With command
57              .CommandText = "SELECT * FROM Meetings WHERE (meeting_date = #" & meeting_date_picker.Value.ToString("dd'/'MM'/'yyyy") & "# AND meeting_status = 'pending')"
58              .Connection = connection
59          End With
60          reader = command.ExecuteReader
61          While reader.Read
62              meeting_group_listbox.Items.Add(reader("meeting_group"))
63          End While
64          reader.Close()
65      End Sub
66  End Class
```

```
1  Public Class new_meeting_form
2      Dim connection As New OleDb.OleDbConnection
3      Dim command As New OleDb.OleDbCommand
4      Dim reader As OleDb.OleDbDataReader
5
6      Dim project_id As Integer
7      Dim location_id As Integer
8      Public meeting_id As Integer
9      Dim attendant_name As String
10     Dim attendant_id As Integer
11     Public Sub new_meeting_form_Load(sender As Object, e As EventArgs) Handles MyBase.Load
12
13         'Povezivanje aplikacije i baze podataka.
14         With connection
15             .ConnectionString = Globals.db_connection_string
16             .Open()
17         End With
18
19         'Popunjavanje kolekcije objekta za odabir prisutnih zaposlenika iz baze podataka.
20         With command
21             .CommandText = "SELECT * FROM Employees"
22             .Connection = connection
23         End With
24         reader = command.ExecuteReader
25         While reader.Read
26             attendants_checked_listbox.Items.Add(reader("Name"))
27         End While
28         reader.Close()
29
30         'Popunjavanje kolekcije objekta za odabir željenog projekta iz baze podataka.
31         command.CommandText = "SELECT * FROM Projects"
32         reader = command.ExecuteReader
33         While reader.Read
34             project_cbox.Items.Add(reader("project_name"))
35         End While
36         reader.Close()
37
38         'Popunjavanje kolekvije objekta za odabir željene lokacije iz baze podataka.
39         command.CommandText = "SELECT * FROM Locations"
40         reader = command.ExecuteReader
41         While reader.Read
42             location_cbox.Items.Add(reader("location_name"))
43         End While
44         reader.Close()
45         connection.Close()
46     End Sub
47
48     'Aplikacija može nastaviti radom na dva moguća načina, gdje je prvi način da se za neki određeni dan
       rezervira sastanak s odabranim atributima, te se nakon rezervacije termina aplikacija vraća u glavni meni,
       dok se odabirom drugog načina pokreće skup naredbi za vođenje toka sastanka.
49
50     'Ukoliko korisnik pritisne gumb "Create meeting", u bazi podataka rezervira se termin sastanka s svim
       odabranim atributima, te vraća korisnika u glavni meni.
51     Public Sub create_meeting_button_Click(sender As Object, e As EventArgs) Handles
       create_meeting_button.Click
52
```

```
53          'Povezivanje aplikacije s bazom podataka.
54          With connection
55              .ConnectionString = Globals.db_connection_string
56              .Open()
57          End With
58
59          'Određivanje indeksa primarnog ključa odabranog projekta.
60          With command
61              .CommandText = "SELECT [project_id] FROM [Projects] WHERE [project_name] = '" &
                project_cbox.Text & "'"
62              .Connection = connection
63          End With
64          project_id = command.ExecuteScalar
65
66          'Određivanje indeksa primarnog ključja odabrane lokacije.
67          command.CommandText = "SELECT [location_id] FROM [Locations] WHERE [location_name] = '"
            & location_cbox.Text & "'"
68          location_id = command.ExecuteScalar
69
70          'Određivanje indeksa primarnog ključa rezerviranog sastanka.
71          command.CommandText = "SELECT COUNT(*) FROM Meetings"
72          meeting_id = command.ExecuteScalar + 1
73
74          'Upisivanje odabranih atributa sastanka u tablicu Meetings u bazi podataka.
75          With command
76              .CommandText = "INSERT INTO [Meetings]([meeting_date],[project_ID],[meeting_group],
                [location_ID],[meeting_ID], [meeting_status],[meeting_time]) VALUES (?,?,?,?,?,?,?)"
77              .Parameters.AddWithValue("@p1", Format("d", date_picker.Value.ToShortDateString))
78              .Parameters.AddWithValue("@p2", project_id)
79              .Parameters.AddWithValue("@p3", group_tbox.Text)
80              .Parameters.AddWithValue("@p4", location_id)
81              .Parameters.AddWithValue("@p5", meeting_id)
82              .Parameters.AddWithValue("@p6", "pending")
83              .Parameters.AddWithValue("@p7", Format("T", time_picker.Value))
84              .ExecuteNonQuery()
85          End With
86
87          'Petlja za odrešivanje indeksa primarnog ključa odabranih zaposlenika, te naknadno upisivanje
            zaposlenika u veznu tablicu Meeting_attendants
88          For i As Integer = 0 To (attendants_checked_listbox.CheckedItems.Count - 1)
89              attendant_name = attendants_checked_listbox.CheckedItems(i).ToString()
90              command.CommandText = "SELECT [employee_ID] FROM Employees WHERE [Name] = '" &
                attendant_name & "'"
91              attendant_id = command.ExecuteScalar
92              With command
93                  .CommandText = "INSERT INTO [Meeting_attendants] VALUES (" & meeting_id & "," &
                    attendant_id & ")"
94                  .ExecuteScalar()
95              End With
96          Next
97          Me.Hide()
98          My.Forms.Meeting_report_generator.Show()
99          connection.Close()
100     End Sub
101
102     'Ukoliko korisnik klikne na gumb "Start meeting", u bazi podataka rezervira se termin sastanka s
```

```
        odabranim atributima, te se otvara nova forma za odabir zapisničara na sastanku.
103     Private Sub start_meeting_button_Click(sender As Object, e As EventArgs) Handles
        start_meeting_button.Click
104
105         'Povezivanje aplikacije i baze podataka.
106         With connection
107             .ConnectionString = Globals.db_connection_string
108             .Open()
109         End With
110
111         'Određivanje indeksa primarnog ključa odabranog projekta.
112         With command
113             .CommandText = "SELECT [project_id] FROM [Projects] WHERE [project_name] = '" &
                project_cbox.Text & "'"
114             .Connection = connection
115         End With
116         project_id = command.ExecuteScalar
117
118         'Određivanje indeka primarnog ključa odabrane lokacije.
119         command.CommandText = "SELECT [location_id] FROM [Locations] WHERE [location_name] = '"
            & location_cbox.Text & "'"
120         location_id = command.ExecuteScalar
121
122         'Određivanje indeksa primarnog ključa rezerviranog sastanka
123         command.CommandText = "SELECT COUNT(*) FROM Meetings"
124         meeting_id = command.ExecuteScalar + 1
125
126         'Upisivanje odabranih atributa sastanka u tablicu Meetings u bazi podataka.
127         With command
128             .CommandText = "INSERT INTO [Meetings]([meeting_date],[project_ID],[meeting_group],
                [location_ID],[meeting_ID], [meeting_status]) VALUES (?,?,?,?,?,?)"
129             .Parameters.AddWithValue("@p1", date_picker.Value)
130             .Parameters.AddWithValue("@p2", project_id)
131             .Parameters.AddWithValue("@p3", group_tbox.Text)
132             .Parameters.AddWithValue("@p4", location_id)
133             .Parameters.AddWithValue("@p5", meeting_id)
134             .Parameters.AddWithValue("@p6", "pending")
135             .ExecuteNonQuery()
136         End With
137
138         'Petlja za odrešivanje indeksa primarnog ključa odabranih zaposlenika, te naknadno upisivanje
            zaposlenika u veznu tablicu Meeting_attendants
139         For i As Integer = 0 To (attendants_checked_listbox.CheckedItems.Count - 1)
140             attendant_name = attendants_checked_listbox.CheckedItems(i).ToString()
141             command.CommandText = "SELECT [employee_ID] FROM Employees WHERE [Name] = '" &
                attendant_name & "'"
142             attendant_id = command.ExecuteScalar
143             With command
144                 .CommandText = "INSERT INTO [Meeting_attendants] VALUES (" & meeting_id & "," &
                    attendant_id & ")"
145                 .ExecuteScalar()
146             End With
147         Next
148
149         'Ponovno određivanje indeksa primarnog ključa trenutno odabranog sastanka, te upisivanje dobivene
            vrijednosti u globalnu varijablu meeting_ID
```

```
150         command.CommandText = "SELECT COUNT(*) FROM Meetings"
151         Globals.meeting_ID = command.ExecuteScalar()
152         My.Forms.new_meeting_form.Close()
153         My.Forms.Meeting_notetaker_form.Show()
154         connection.Close()
155     End Sub
156 End Class
```