

# 1

## Izrada projekata

U ovom poglavlju naučit ćeš:

- što je projekt
- osnovne pojmove vezane za izradu projekta
- faze izrade projekta
- raspravljati o kvaliteti dizajna rješenja
- kako projekt izraditi prema vodopadnom modelu.

## 1.1. Pojam projekta

U svakodnevnom je životu gotovo nezamislivo raspravljati o napretku u gospodarstvu ili u području znanosti bez spominjanja projekata. Projekti i upravljanje projektima kao znanstvena disciplina postoji tek od sredine 20. stoljeća, ali projektni pristup postoji tisućama godina. Prisjetimo se egiptskih piramida, antičkih olimpijskih igara, prekomorskih putovanja kojim su Europljani otkrili Ameriku... Ako malo bolje pogledamo, povijest je prepuna i manjih, svakodnevnih, osobnih iskustava koja su ostvarena nekim oblikom projektnog pristupa, primjerice organizacija proslave rođendana, uređenje svoje sobe ili radnog kutka, upis u željenu školu, stjecanje diplome itd.

Sve spomenuto možemo smatrati projektima jer imaju zajedničke značajke, a to su:

- jasan cilj, odnosno rezultat koji se želi postići
- poznato je vrijeme u kojem je potrebno izvršiti sve potrebne aktivnosti
- određeni su resursi koji će se koristiti (prirodni, ljudski i materijalni).

Uz ove zajedničke značajke svaki projekt ima i svoje posebnosti. Projekti se razlikuju s obzirom na područje ljudske djelatnosti, pa tako razlikujemo obrazovne, znanstvene, gospodarske... Rezultat projekta može biti određeni **produkt**, odnosno proizvod (na primjer vjetroelektrana ili novi model automobila). Rezultat projekta također može biti **nematerijalan**, npr. primjena novog ili poboljšanje postojećeg poslovnog procesa ili stečeno znanje kao rezultat istraživačkog projekta.

Bez obzira hoće li rezultat našeg projekta biti materijalan ili nematerijalan, razlikovat ćemo one koji kao rezultat imaju određenu analizu i prezentaciju samog projekta i one koji kao rezultat imaju i programsko rješenje.

Projekti s programskim rješenjem često su sastavni dio znanstvene discipline koja se naziva programsko inženjerstvo (engl. *software engineering*). Prikazat ćemo osnovne principe koji su primjenjivivi na sva područja ljudskog djelovanja koja koriste projektni pristup.

Uobičajeno je da projekt provode projektni timovi, no kada se radi o manjim projektima, kao što će biti naši, projekt može realizirati i samo jedna osoba ili tim od dvije do tri osobe.

**Što je tim?** Tim je skupina pojedinaca koji radeći zajedno postižu bolje rezultate od pojedinačnih postignuća članova tima. Tim potiče, koristi i razvija znanja, vještine i iskustva svojih članova i tako nadoknađuje individualne slabosti i značajno doprinosi kvaliteti rada.



## 1.2. Model projekta s programskim rješenjem

U sklopu programskog inženjerstva svaki će projekt za cilj imati novi **programski produkt**. Programski produkt je skup računalnih programa, pratećih datoteka i pripadne dokumentacije. Dokumentacija je sastavni i nezaobilazni dio svakog projekta. Dokumentaciju dijelimo na projektnu i korisničku dokumentaciju. Kao što i samo ime govori, projektna je dokumentacija ona koja se izrađuje i koristi za vrijeme realizacije projekta. Korisnička dokumentacija je dokumentacija dostupna krajnjem korisniku.

Projektnu dokumentaciju mogu činiti studije izvodivosti, elaborati zaštite okoliša, faze izrade projekta, troškovnici itd. Korisnička dokumentacija također se izrađuje za vrijeme projekta, ali namijenjena je korisniku (naručitelju). To su dokumenti poput korisničkog priručnika, vodiča za instalaciju, priručnika za administriranje i slično.

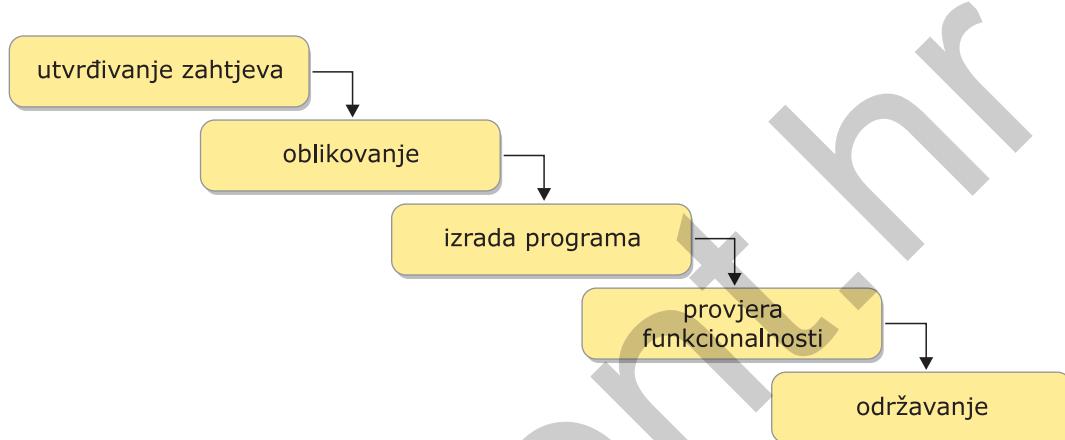
Upravljanje projektom je niz vrlo složenih aktivnosti, koje izravno ovise o području djelatnosti te o svim zahtjevima projekta. Tu se prvenstveno misli na razinu kvalitete produkta, veličini budžeta i vremenu raspoloživom za realizaciju projekta. Za izvođenje naših projekata pojednostavnit ćemo cijeli proces.

Naš model projekta sastojat će se od pet osnovnih aktivnosti:

- **utvrđivanje zahtjeva**
- **oblikovanje** (dizajn)
- **izrada proizvoda** (implementacija)
- **provjera funkcionalnosti** (verifikacija)
- **održavanje**.

## 1.3. Idealni vodopadni model projekta

U prvom smo razredu naučili korake kroz koje trebamo proći od postojanja problema do njegova konačnog rješenja. Kada govorimo o projektu, ti su koraci vrlo slični i možemo ih prikazati vodopadnim modelom (slika 1.1). U idealnoj ćemo situaciji po završetku jedne faze prelaziti u drugu.



Slika 1.1. Vodopadni model projekta

### 1.3.1. Utvrđivanje zahtjeva

Prije pokretanja projekta potrebno je izvršiti tzv. analizu isplativosti. Kako naši školski projekti nemaju finansijski značaj, u dalnjem ćemo razmatranju izostaviti ovu aktivnost.

Naručitelj (kupac) obično ima apstraktnu ideju o tome što želi kao krajnji rezultat, ali ne i što bi programski proizvod točno trebao raditi. U sklopu utvrđivanja zahtjeva projektni tim treba prepoznati nepotpune, nejasne ili proturječne zahtjeve te zajedno s naručiteljem jasno i jedinstveno definirati osnovne funkcionalnosti programa, odnosno **što bi program trebao raditi**.

Ovaj dio projekta uobičajeno se sastoji od mnoštva pregovora i dogovora između naručitelja i projektnog tima, a rezultat bi trebao biti jedan od osnovnih projektnih dokumenata – **specifikacija zahtjeva**.

#### Važno

Specifikacija zahtjeva u realnom se svijetu može smatrati pravnim dokumentom.

Specifikacija zahtjeva bi uz opis osnovnih funkcionalnosti programa trebala sadržavati i ostale zahtjeve na program, kao npr. brzina izvođenja programa, za koje će ulazne podatke dati valjane izlazne vrijednosti, mogućnost korištenja programa u različitim uvjetima itd.

Bez obzira što u ovom trenutku ne možemo u potpunosti sagledati cijelovitost nekog projekta, trebamo razmisiliti o značajkama važnim za kvalitetu proizvoda, poput prenosivosti, dostupno-

sti, pouzdanosti itd. Za detaljnu analizu svih ovih dodatnih zahtjeva potrebno je znanje koje nadilazi sadržaj predmeta Informatika te ih stoga u našim projektima nećemo detaljno razmatrati.

Tijekom definiranja zahtjeva projektni tim mora osmisliti i pripremiti načine provjere funkcionalnosti gotovog proizvoda – programa. Budući da se ova provjera treba raditi **isključivo na temelju zahtjeva** naručitelja, vrlo je važno da se testni podatci s pripadnim očekivanim rezultatima pripreme u ovoj fazi projekta tj. prije same izrade programa.

Važno je uočiti sve očekivane ulazne vrijednosti i pripadne izlazne vrijednosti tj. očekivana ponanja programa.

Popis svih testnih slučajeva kojima bi se provjerile sve funkcionalnosti programa definiranih u specifikaciji zahtjeva naziva se testna specifikacija.

### 1.3.2. Oblikovanje

Oblikovanje (dizajn) je projektna aktivnost tijekom koje će biti potrebno definirati kako će program raditi. Da bi se u projektu moglo krenuti s oblikovanjem, potrebno je imati jasno definiranu specifikaciju zahtjeva. Na temelju tog dokumenta projektni bi tim trebao analizirati i odlučiti koji bi programski alat bio najpogodniji za rješavanje problema, na kojim bi platformama program trebao raditi (Windows, Linux, IOS...), koje bi programske strukture bile najpogodnije, kakvu bazu podataka koristiti, kako će izgledati sučelje prema korisniku itd.

U našim školskim projektima najčešće nećemo imati mogućnost odlučivanja o programskom alatu, platformi i slično, s obzirom na to da smo ograničeni resursima u školi. Stoga ćemo u sklopu oblikovanja pokušati napraviti kratki opis rada programa i korištenja programskih struktura. U tu svrhu mogu se koristiti mentalne mape, dijagrami toka, opisi i sl.

Uobičajeno je tijekom oblikovanja predložiti nekoliko rješenja te nakon toga analizom odabrat optimalnije (za očekivati je da bi jedno rješenje imalo prednost pred drugim temeljem određenih parametara, ali isto bi tako i drugo rješenje moglo imati prednost pred prvim temeljem nekih drugih parametara). Proces odabira optimalnijeg rješenja zahtjeva veliko znanje i iskustvo.

**Primjer 1.1.** U obiteljskoj kući žive tri generacije: baka i djed, roditelji i djeca koja pohađaju srednju školu. Svaka generacija, naravno, ima svoje navike i želje vezane za slušanje glazbe. Srednjoškolci vole glazbu slušati nešto glasnije, baka i djed već slabije čuju i moraju poglašniti svoju glazbu, a roditelji bi u miru željeli čitati knjige. Dakle, na osnovi definiranih zahtjeva potrebno je napraviti prijedloge kako osigurati kvalitetan suživot ove obitelji.

Prvi prijedlog dizajna rješenja je postaviti zvučnu izolaciju zidova.

Drugi prijedlog dizajna rješenja je nadogradnja kata kuće.

Treći prijedlog dizajna rješenja je kupnja slušalica za svakog ukućana.

Četvrti prijedlog dizajna rješenja je prodaja kuće i kupnja dvaju stanova.

Analizom svih četiriju prijedloga dizajna, obitelj se odlučila za treći prijedlog jer je najjeftinije i najbrže rješenje.

### 1.3.3. Izrada programa

Izrada programa je aktivnost u kojem se piše programski kôd (kodira). Na temelju opisa nastalih u aktivnosti oblikovanje, projektni tim raspoređuje zadatke i stvara dijelove programa koji se zatim spajaju u smislenu cjelinu.

Za vrijeme procesa izrade program se testira, pronalaze se (engl. *debugging*) i otklanjaju greške. Proces testiranja usko je povezan s procesom izrade programa.

### 1.3.4. Provjera funkcionalnosti

Kad je program gotov, kreće se u završnu provjeru radi li program **onako kako je definirano u specifikaciji zahtjeva**. U osnovi se ovaj postupak svodi na ispitivanje testnih slučajeva opisanih u testnoj specifikaciji (iz aktivnosti specifikacija zahtjeva). Za svaki od testnih slučajeva potrebno je pripremiti početno stanje te nakon pokretanja programa unijeti ulazne podatke predviđene tim testnim slučajem. Odziv programa (ponašanje), uključujući i ispis (izlazne podatke), potrebno je usporediti s očekivanim, odnosno onima koji su navedeni u testnoj specifikaciji.

U školskim projektima provjeravat će se samo funkcionalnost programa. Ovakva se provjera uobičajeno zove funkcionalni test.



Slika 1.2. Provjera funkcionalnosti programa

### 1.3.5. Održavanje

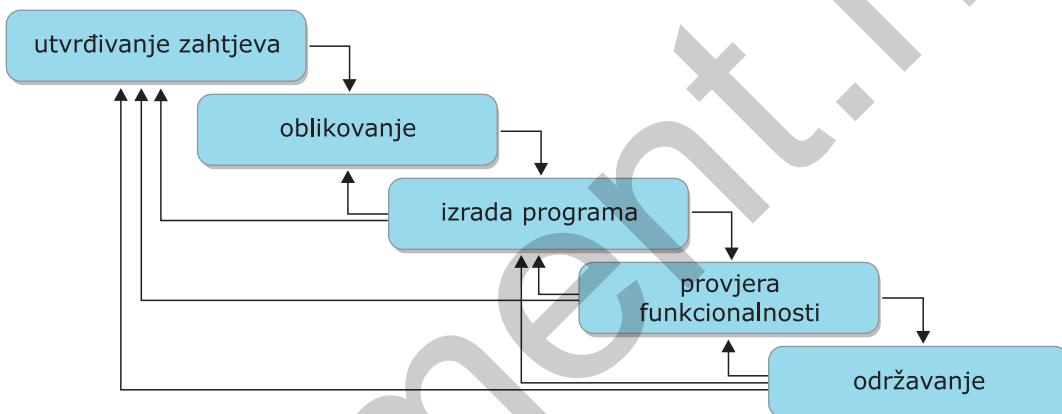
Održavanje je aktivnost koja kreće nakon što je program predan naručitelju. Ovo je najčešće zadnja aktivnost u projektu koja bi trebala osigurati podršku naručitelju tijekom određenog vremenskog razdoblja.

Ako tijekom tog razdoblja naručitelj odluči da želi promijeniti ili nadograditi određenu funkcionalnost, može se dogоворiti s projektnim timom oko nastavka projekta. U tom slučaju projektni se cilj mijenja (nadograđuje) i sam projekt kreće ponovno od utvrđivanja zahtjeva te se cijeli postupak ponavlja. Naručitelj, naravno, u ovom slučaju ima finansijsku obavezu prema projektnom timu.

Ako se tijekom razdoblja održavanja utvrde nepravilnosti, odnosno greške u radu programa, dužnost je projektnog tima otkloniti te nepravilnosti (ponovnim kodiranjem, testiranjem, a po potrebi i promjenama u dizajnu itd.). U ovom pak slučaju naručitelj nema finansijskih obaveza prema timu.

## 1.4. Vodopadni model projekta u svakodnevnosti

U stvarnom je životu malo vjerojatno da će projektni tijek pratiti idealni vodopadni model. Naime, već smo spomenuli da je moguće, zbog grešaka uočenih tijekom održavanja, ponoviti određene aktivnosti projekta. Također se često događa da se prilikom izrade programa uoče nedostatci u dizajnu zbog čega je potrebno "vratiti se" u aktivnost oblikovanja i ažurirati opise. Vrlo čest slučaj je i da se tijekom oblikovanja ili tijekom izrade programa uoče nepotpuni ili kontradiktorni zahtjevi, pa je potrebno ponoviti proces pregovaranja s naručiteljem i ažurirati zahtjeve. Iz svega navedenog zaključujemo da se u stvarnom svijetu, nakon što se uoče nedostaci ili greške, rad na projektu često "vraća" u neku od prethodnih faza projekta. Prikaz ovakvog, modificiranog modela dan je slikom 1.3.



Slika 1.3. Modificirani vodopadni model

To, naravno, ima za posljedicu reorganizaciju zaduženja unutar projektnog tima, a katkad i ponovno definiranje vremenskih rokova, što treba izbjegavati koliko je moguće.

## 1.5. Otklanjanje grešaka prilikom razvoja programa

Uobičajeni naziv za postupak kojim od ideje dolazimo do gotovog, upotrebljivog programa jest razvoj programa (engl. *program development*). Za razliku od nekih drugih proizvoda, prilikom postupka razvoja programa uobičajeno je da se javljaju greške koje se potom bez većih posljedica otklanjavaju. Već smo spomenuli da je funkcionalno testiranje sastavni dio programiranja, a provjera funkcionalnosti pri kraju projekta treba osigurati rad programa bez grešaka.

Nažalost, u stvarnosti se vrlo često događa da se određeni broj grešaka ipak ne otkloni tijekom projekta, već se one otkrivaju (i rješavaju) nakon što je program već predan naručitelju.

Jedan od glavnih ciljeva razvoja programa projektnim pristupom je **otkloniti čim više grešaka što prije**. To u praksi znači da svaka greška koja se otkrije u ranim fazama projekta ima znatno manje posljedice na kvalitetu gotovog programa od one koje se otkriju kad je program već gotov ili kad je već isporučen naručitelju. I u školi, uostalom, prilikom rješavanja testa možemo uočiti grešku koju smo napravili. Ako je greška uočena prije nego što smo test predali nastavniku/nastavnici, imamo još priliku ispraviti je, a ako smo grešku uočili nakon što smo test predali, ona će, nažalost, vjerojatno rezultirati slabijom ocjenom. I u projektnom je pristupu važno grešku detektirati i otkloniti prije nego program isporučimo naručitelju jer će to u suprotnom imati finansijske posljedice, ali i posljedice na ugled, vjerodostojnost i, vrlo vjerojatno, buduće poslove projektnog tima.

Uz sve spomenuto, vrlo je važno da se na kraju projekta provede postupak samovrednovanja, odnosno da se procijeni što je bilo dobro u projektu, što nije bilo dobro, što se treba mijenjati i slično. Ovako skupljena iskustva pomoći će nam kod izvođenja sljedećeg projekta.

## 1.6. Primjer aktivnosti prilikom izrade projekta

Projektnim pristupom izradimo najavu (promidžbenu kampanju) u školi za predstojeće događanje (proslava dana škole, školskog projekta, predstavljanje visokoškolske ustanove, dan sigurnijeg interneta ili sl.) Tijekom provođenja projekta važno je uočiti pet osnovnih aktivnosti:

### ■ **utvrđivanje zahtjeva**

Prije svega, potrebno je definirati tko je ciljana skupina kojoj se obraćamo (učenici, nastavnici, roditelji itd.) i što se želi postići: želimo li samo podijeliti osnovne informacije ili želimo upoznati ostale sa značajem događanja, zainteresirati ih iznošenjem nekih činjenica, pružiti mogućnost zabave uz učenje i sl. Uz to potrebno je i osmislići način kako ćemo provjeriti postiže li naš produkt zadani cilj i zapisati detaljan opis provjere u obliku dokumenta.

### ■ **oblikovanje (dizajn)**

Nakon što znamo što želimo postići, potrebno je osmislići barem tri načina kako postići ono što je definirano u prethodnoj fazi: na primjer izraditi plakate, web-stranice, zasebnu grupu na društvenoj mreži, snimiti kratki film ili napraviti animaciju, pripremiti izložbu i sl. Svaki od načina potrebno je analizirati s obzirom na to kako zadovoljava tražene zahtjeve te se temeljem analize odlučiti za jedno od rješenja.

### ■ **izrada proizvoda (implementacija)**

Na temelju rješenja u fazi oblikovanja izradimo produkt, te tijekom izrade ispravimo uočene greške.

### ■ **provjera funkcionalnosti (verifikacija)**

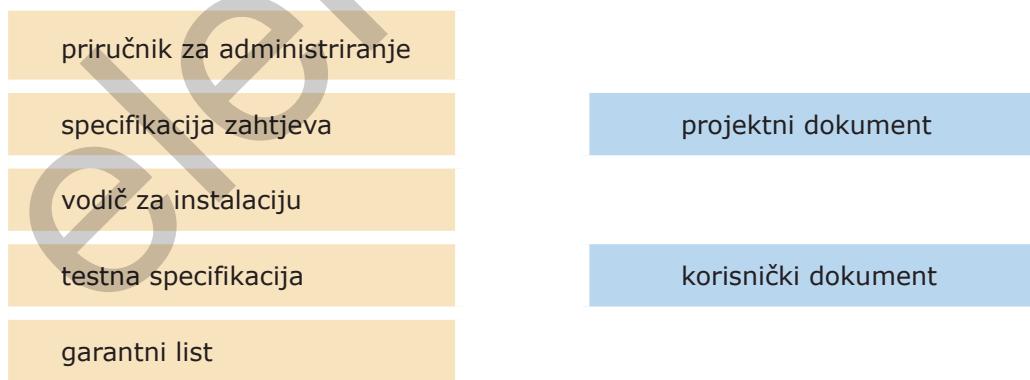
Na temelju opisa provjere iz aktivnosti utvrđivanja zahtjeva provjerimo odgovara li naš produkt traženim zahtjevima.

**■ održavanje**

Tijekom promidžbene kampanje uočit ćemo nedostatke ili neke mogućnosti poboljšanja. Ako je moguće, potrebno ih je odmah ispraviti, ali ih svakako treba zabilježiti i iskoristiti tijekom samovrednovanja (kako procjenjuješ svoj i rad drugih unutar projekta, kakav je bio timski rad tima, kakva je bila komunikacija unutar tima u projektu, koja znanja i kompetencije si stekao, kakva je bila organizacija projekta i sl.).

## 1.7. Zadatci za ponavljanje i vježbu

1. Nabroji faze izrade projekta.
2. Kod idealnog vodopadnog modela projekta redoslijed aktivnosti je:
  - a) specifikacija zahtjeva, dizajn, implementacija, verifikacija i validacija, održavanje
  - b) implementacija, specifikacija zahtjeva, dizajn, verifikacija i validacija, održavanje
  - c) dizajn, implementacija, specifikacija zahtjeva, verifikacija i validacija, održavanje
  - d) dizajn, specifikacija zahtjeva, implementacija, verifikacija i validacija, održavanje
3. Kako se zove provjera radi li proizvod točno onako kako je naručitelj tražio?
4. U kojoj ćeš aktivnosti projekta, kojemu je proizvod programsko rješenje, izabrati programski jezik koji ćeš koristiti prilikom izrade rješenja?
5. Od navedenih dokumenata označi koji su projektni dokumenti, a koji će biti dostavljeni naručitelju zajedno s proizvodom:



6. Na vijeću učenika učenici drugih razreda dogovorili su se da će u školi organizirati prezentiranje različitih zanimanja za koje je izrazito važno imati razvijeno računalno razmišljanje. Osmisli prve dvije faze izrade tog projekta. Ponudi barem dva rješenja dizajna.

element.hr