|  |  |
| --- | --- |
|  | Fakultet turizma i ruralnog razvoja u Požegi  Vježba 1 |

1. **Organizacija timova od 3-5 studenata sa sljedećim ulogama:**
   * **Nadgledanje razvoja proizvoda**
   * **Implementacija proizvoda**
   * **Testiranje proizvoda**
   * **Pisanje dokumentacije**

**Napomena: svi članovi tima ravnopravno sudjeluju u razvoju proizvoda.**

1. **Definiranje softverskog projekta po slobodnom izboru (trajanje projekta je 3 mjeseca).**
   1. **Opisati softverski projekt**
   2. Softverski projekt koji sam odabrao je "Webshop za grafičku tiskaru". Projekat ima za cilj razviti online platformu za prodaju usluga i proizvoda grafičke tiskare. Kupci će moći kupiti različite proizvode i usluge putem interneta, a tiskara će moći upravljati svojim narudžbama, upravljati proizvodima i uslugama, te pratiti stanje zaliha.
      1. **odrediti vrstu softverskog proizvoda (objasniti zbog čega pripada toj grupi)**

Webshop za grafičku tiskaru pripada kategoriji web aplikacija. To je zato što korisnici pristupaju aplikaciji putem web preglednika.

* + 1. **odrediti viziju i doseg softverskog proizvoda**

Vizija projekta je stvoriti online platformu koja omogućuje kupcima da lako i brzo naruče usluge i proizvode grafičke tiskare. Platforma će pružiti kupcima mogućnost pregledavanja različitih proizvoda i usluga, te mogućnost personalizacije proizvoda, kao što su naljepnice, vizitke, plakati i sl. Webshop za grafičku tiskaru će omogućiti tiskari da upravlja narudžbama, proizvodima i uslugama, te da pratiti stanje zaliha.

* + 1. **definirati alate i tehnologije iv. definirati arhitekturu softverskog proizvoda (ugrubo)**

Za razvoj platforme koristit će se sljedeći alati i tehnologije:

* HTML, CSS i JavaScript za front-end razvoj
* React kao JavaScript biblioteka za front-end razvoj
* Node.js za server-side development
* Express.js kao Node.js framework
* MongoDB kao baza podataka

**iv. Arhitektura softverskog proizvoda**

Arhitektura softverskog proizvoda bit će zasnovana na MVC arhitekturi. Model će predstavljati bazu podataka, view će predstavljati korisnički interfejs, a kontroler će biti zadužen za logiku aplikacije i komunikaciju između view i model sloja.

* 1. **Napraviti popis aktivnosti i njihovo vremensko trajanje te navesti međuovisnosti pojedinih aktivnosti.**

Planiranje projekta - 1 tjedan

Definiranje zahtjeva - 2 tjedna

Dizajniranje korisničkog interfejsa - 3 tjedna

Razvoj front-end dijela - 4 tjedna

Razvoj back-end dijela - 4 tjedna

Integracija front-end i back-end dijela - 1 tjedan

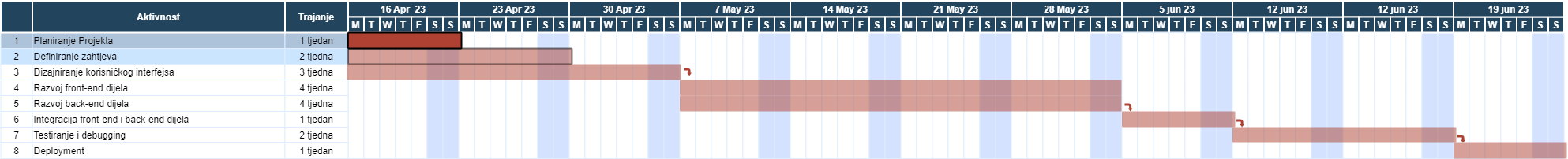
Testiranje i debugging - 2 tjedna

Deployment - 1 tjedan

Aktivnosti 1, 2 i 3 moraju biti završene prije nego što se može početi s aktivnostima 4 i 5. Aktivnosti 4 i 5.

* 1. **Napraviti vremenski dijagram aktivnosti.**

**Vremenski dijagram aktivnosti za "Webshop za grafičku tiskaru" projekt:**

****

Kao što se vidi iz dijagrama, aktivnosti 1, 2 i 3 će se provoditi istovremeno, dok će aktivnosti 4 i 5 biti provedene nakon što su aktivnosti 1, 2 i 3 završene. Nakon završetka aktivnosti 4 i 5, slijedi integracija front-end i back-end dijela u aktivnosti 6. Nakon toga, provodit će se aktivnost 7 - testiranje i debugging, prije nego što se nastavi s aktivnosti 8 - deployment.

* 1. **Napraviti dijagram alokacije ljudskih resursa.**

Primjer dijagrama alokacije ljudskih resursa za "Webshop za grafičku tiskaru" projekt:

java

Projekt manager (1)

|

Web developer (2)

|

Front-end developer (2)

|

Back-end developer (2)

|

UI/UX designer (1)

|

QA/Testing engineer (1)

U ovom primjeru, projekt manager nadgleda cijeli projekt i usklađuje aktivnosti među svim članovima tima. Web developer nadgleda razvoj cijelog projekta i usklađuje front-end i back-end developere. Front-end developeri rade na razvoju korisničkog sučelja, dok back-end developeri rade na izradi funkcionalnosti i baze podataka. UI/UX designer radi na dizajnu korisničkog sučelja, dok QA/testing engineer testira softverski proizvod kako bi se osiguralo da je stabilan i da radi ispravno. Ovo je samo primjer alokacije ljudskih resursa, a stvarna alokacija može varirati ovisno o veličini i složenosti projekta.

1. **Definirati model softverskog procesa koji će se koristiti i obrazložiti zbog čega.**

Za "Webshop za grafičku tiskaru" projekt, preporučuje se korištenje agilnog modela softverskog procesa. Ovaj model se fokusira na brzo iteriranje, često testiranje i usklađivanje sa zahtjevima klijenta. Agilni model se obično sastoji od slijeda kratkih iteracija, poznatih kao sprintovi, gdje se funkcionalnost dodaje softverskom proizvodu u svakoj iteraciji. Svaki sprint se završava demo prezentacijom softverskog proizvoda klijentu, što omogućava timu da dobije povratne informacije i da brzo reagira na eventualne probleme.

Agilni model se preporučuje za ovaj projekt iz nekoliko razloga. Prvo, projekt je prilično složen i zahtijeva veliku suradnju između različitih timova. Agilni model pruža fleksibilnost i omogućuje timovima da brzo reagiraju na promjene u zahtjevima. Drugo, model naglašava testiranje i provjeravanje funkcionalnosti softverskog proizvoda, što je ključno za webshop, posebno kada je riječ o sigurnosti i stabilnosti transakcija. Konačno, agilni model potiče tim na kontinuiranu komunikaciju sa klijentom i stvaranje prilagođenih rješenja koja će zadovoljiti njihove specifične potrebe.

Agilni model se oslanja na usko surađivanje između članova tima, visoko transparentnost, i brzo usvajanje novih zahtjeva. Kako bi se osigurala učinkovita primjena ovog modela, preporuča se korištenje alata za projekt menadžment i komunikaciju, poput Scrum-a, koji pomaže timovima da se organiziraju i planiraju sprints.

1. **Stvoriti projekt na github-u i dodati članove tima.**

https://github.com/Robertino2809/Programsko-in-enjerstvo---projekt