

3. PLAN I SADRŽAJ MODULA

Uvod u kodiranje - Uvod u programiranje



Version 1.0

1. PLAN MODULA

Plan modula	
Naziv modula	Uvod u kodiranje
Trajanje	3 sata (ukupno)
Ciljevi učenja	Do kraja ovog modula, učesnici će biti u stanju da - ● Razumeti osnovne koncepte programiranja i kodiranja. ● Prepoznati različite programske jezike i njihovu upotrebu. ● Pisati jednostavne programe koristeći osnovne programske konstrukcije (npr. promenljive, petlje, uslovne izraze). ● Razviti veštine rešavanja problema kroz vežbe kodiranja. ● Upoznati se sa celokupnom strukturom programa. ● Razumeti važnost algoritama i logičkog razmišljanja u programiranju ● Naučiti kako da otklonite greške u jednostavnim kodovima ● Istražiti primene kodiranja u stvarnom svetu u različitim industrijama. ● Podstaci računarsko razmišljanje i kreativnost kroz aktivnosti programiranja..

Plan modula	
Plan časa	Osnove digitalnog dizajna i razvoja Modul obuke za gluvu omladinu Trajanje: ~60 minuta (3 teme × 20 minuta)
	1. Šta je kodiranje i programiranje? (20 minuta) Cilj: - Razumeti šta su kodiranje i programiranje i njihov značaj u današnjem digitalnom svetu. Glavne tačke koje treba obraditi : - Definicija kodiranja i programiranja - Razlika između programskih jezika i primera (npr. Pajton, Javaskript) - Primeri kodiranja aplikacija iz stvarnog sveta (aplikacije, veb stranice, igre) Aktivnosti: - Diskutujte i razmotrite gde se učenici susreću sa kodiranjem u svakodnevnom životu - Pogledajte kratak video koji objašnjava važnost programiranja u tehnologiji Nastavni metod: - Predavanje uz vizuelna pomagala (slajdovi, video snimci) - Interaktivna diskusija
	2. Osnovni koncepti i gradivni blokovi kodiranja (20 minuta) Cilj: Upoznati osnovne programske koncepte kao što su promenljive, tipovi podataka i osnovne kontrolne strukture.

Plan modula	
	Glavne tačke za obradu: • Promenljive i tipovi podataka (brojevi, tekst) • Osnovni operatori i izrazi • Jednostavan tok upravljanja: if-else naredbe i petlje • Koncept algoritama i koraci rešavanja problema Aktivnosti: • Praktična vežba: kreirajte jednostavan pseudokod ili dijagrame toka za svakodnevne zadatke (npr. pravljenje sendviča, spremanje za školu) Nastavni metod: • Demonstracija sa živim kodiranjem/primerima • Vođene vežbe, grupni rad 3Pisanje vašeg prvog programa (20 minuta) Cilj: • Omogućite učenicima da napišu i pokrenu svoj prvi jednostavan program. Glavne tačke koje treba obraditi : • Uvod u programsko okruženje prilagođeno početnicima (npr. Scratch, Code.org, Python IDLE) • Struktura jednostavnog programa (ispisna naredba, unos, osnovna logika) • Otklanjanje grešaka u uobičajenim slučajevima Aktivnosti: • Kodiranje korak po korak: Studenti pišu program koji ih pita za ime i pozdravlja ih • Vežbe recenziranja i otklanjanja grešaka

Plan modula	
	Nastavni metod : • Demonstracija kodiranja uživo • Vođena vežba uz podršku i povratne informacije • Saradnja sa vršnjacima
Potrebni materijali	1. Vizuelna pomagala i prezentacije: • Slajdovi ili PowerPoint prezentacija sa ključnim konceptima • Kratki video snimci koji objašnjavaju kodiranje i programiranje (opciono) 2. Računari i softver: • Računari ili laptopovi za svakog učenika ili parove • Programsko okruženje prilagođeno početnicima (npr. Scratch, Python IDLE, Code.org, Blockly) 3. Materijali i radni listovi: • Pseudokod ili šabloni dijagrama toka za aktivnosti • Radni listovi sa vežbama o osnovnim konceptima i rešavanju problema • Podsetnik za uobičajenu sintaksu i komande programiranja 4. Interaktivni alati: • Onlajn platforme za kodiranje ili veb stranice za vežbanje (npr. Code.org, Khan Academy, Trinket) • Vežbe za otklanjanje grešaka ili izazovi kodiranja u obliku igre 5. Dodatni materijali: • Tabla i markeri za objašnjenja i diskusije • Odštampani primeri jednostavnih programa i algoritama • Sveske ili papir za rad od početka ili razmišljanje

2. SADRŽAJ MODULA

Sadržaj	Opis	Sadržaj
Sadržaj lekcije	Digitalna pismenost	Digitalna pismenost znači posedovanje znanja i veština za korišćenje digitalnih alata kao što su softver za dizajn i programski jezici za kreiranje aplikacija, veb stranica i onlajn sadržaja. To uključuje razumevanje načina funkcionisanja digitalnih sistema i sposobnost kreiranja i dizajniranja jednostavnih digitalnih proizvoda. Ključni digitalni termini : • Programski jezik – Jezik koji se koristi za pisanje instrukcija koje računar može da razume. • Kod – Skup instrukcija napisanih u programskom jeziku. • Algoritam – Korak-po-korak skup instrukcija za rešavanje problema. • Promenljiva – Mesto skladištenja u programu koje sadrži podatke. • Tip podataka – Vrsta podataka (npr. celi brojevi, stringovi, brojevi sa pokretnim decimalom) sačuvanih u promenljivoj. • Funkcija / Metod – Blok koda koji obavlja određeni zadatak i može se ponovo koristiti. • Petlja – Niz instrukcija koji se ponavlja dok se ne ispuni uslov. • Uslov / Ako-Inače – Izjava koja izvršava različite radnje na osnovu toga da li je uslov tačan ili netačan. • Debugovanje – Proces pronalaženja i ispravljanja grešaka ili bagova u kodu. • Sintaksa – Pravila koja definišu strukturu koda u programskom jeziku. • Ulaz – Podaci koje program prima od korisnika ili drugog izvora.



Sadržaj	Opis	Sadržaj
		● Izlaz – Podaci koje program proizvodi za korisnika. ● IDE (Integrirano razvojno okruženje) – Softverska aplikacija koja pruža alate za pisanje, testiranje i debugovanje koda. ● Bytecode – Međukod koji se može interpretirati ili kompajlirati. ● Kompilacija – Proces prevođenja koda u mašinski jezik koji računar može da izvrši. ● Mašinski jezik – Binarni kod koji procesor računara direktno razume. ● Komentari – Beleške napisane u kodu koje objašnjavaju šta kod radi; računar ih ignoriše. ● Sintaksna greška – Greška usled netačnog koda koji krši pravila jezika. ● Logička greška – Greška gde se kod izvršava, ali proizvodi netačne rezultate..
Video rezime	Youtube Links	1. „Šta je kodiranje?“ – Kratki kurs računarstva https://www.youtube.com/watch?v=fikvFQoYZ3Q Lako razumljivo objašnjenje šta je programiranje i zašto je važno. 2. „Uvod u programiranje“ – freeCodeCamp https://www.youtube.com/watch?v=8oRjP8yj_w8 Pregled osnova programiranja prilagođen početnicima . 3. „Naučite programiranje za 1 sat“ – Tech with Tim https://www.youtube.com/watch?v=BoardJz_VRc Kratak pregled osnovnih programskih koncepata namenjenih početnicima.

Sadržaj	Opis	Sadržaj
		4. „Šta je programski jezik?“ – Računarske nauke o igrama https://www.youtube.com/watch?v=7sA8vfQWWvg Kratko objašnjenje programskih jezika i kako oni rade.
Aktivnosti		1. Grupna aktivnost: Cilj: • Podstaknite učenike da primenjuju osnovne programske koncepte kao što su sekvence, uslovi i petlje tako što ćete dizajnirati jednostavan, korak-po-korak algoritam za zadatak iz stvarnog sveta. Opis aktivnosti: 1. Podelite učenike u male grupe (3-4 učenika po grupi). 2. Svaka grupa bira ili joj se dodeljuje jednostavan svakodnevni zadatak (npr. pravljenje sendviča, pranje zuba, spremanje za školu). 3. Učenici rade zajedno kako bi kreirali detaljan korak-po-korak algoritam ili dijagram toka koji opisuje zadatak, uključujući tačke odlučivanja (uslove „ako-inače“) i ponavljajuće korake (petlje). 4. Svaka grupa predstavlja svoj algoritam razredu, objašnjavajući kako funkcioniše sa naglaskom na logičke korake i tačke odlučivanja. Potrebni materijali: • Veliki listovi papira ili bele table, markeri ili štampani šabloni za dijagrame toka. • Materijali sa osnovnim simbolima algoritama i smernicama za dijagrame toka (opciono).

Sadržaj	Opis	Sadržaj
		Ishod učenja : • Učenici će pokazati razumevanje programske logike i načina na koji se zadaci iz stvarnog sveta mogu razložiti na instrukcije slične kodu. • Razvije veštine rešavanja problema i saradnje.
		2. Igranje uloga : Cilj: • Da pomogne učenicima da razumeju kako da jasno objasne programske koncepte i da simuliraju komunikaciju u stvarnom svetu kodiranja. Takođe, to pojačava njihovo razumevanje programskih koraka i donošenja odluka. Opis aktivnosti: 1. Učenici se dele u parove; jedan učenik deluje kao „Instruktor kodiranja“, a drugi kao „početnik“ koji želi da nauči kako da programira jednostavan zadatak (npr. pravljenje sendviča, pranje zuba). 2. „Početnik“ traži od „Instruktor kodiranja“ da objasni kako da završi zadatak kroz korak-po-korak proces „kodiranja“. 3. „Instruktor kodiranja“ mora da opiše proces jednostavnim, jasnim uputstvima, uključujući tačke odlučivanja i ponavljanja, kao da uči početnika kako da razmišlja kao programer. 4. Zamenite uloge nakon završetka objašnjenja.

Sadržaj	Opis	Sadržaj
		5. Nakon što su obe uloge odigrane, učenici razgovaraju o važnosti jasnih uputstava i logičkog razmišljanja u programiranju. Potreban materijal: • Nema, samo prostor za igru uloga. • Opciono, dajte učenicima primere zadataka ili koraka koji će im pomoći da započnu . Ishodi učenja: • Učenici će vežbati pretvaranje zadataka iz stvarnog sveta u logična, korak-po-korak uputstva. • Razvije veštine jasnog objašnjavanja tehničkih koncepata i aktivnog slušanja.
		3. Scenariji iz stvarnog života : Ciljevi: • Da bi se učenicima pomoglo da razumeju kako da razlože dnevnu rutinu na jasne, logične korake slično kodiranju. Ova aktivnost podstiče rešavanje problema, sekvenciranje i veštine donošenja odluka. Opis aktivnosti: 1. Uparite učenike; jedan učenik igra „Programera“ (osobu koja dizajnira uputstva), a drugi je „Korisnik“ (osoba koja prati uputstva).

Sadržaj	Opis	Sadržaj
		2. Od „Programera“ se traži da opiše tipičnu jutarnju rutinu (npr. „Spremanje za školu“) kao da kreira jednostavan „program“ ili skup uputstava kojih se treba pridržavati. - o Trebalo bi da uključuju korake poput buđenja, pranja zuba, izbora odeće, doručka itd. - o Trebalo bi da uključuju tačke donošenja odluka (npr. „ako je hladno, obucite jaknu“) i ponavljanja (npr. „operite zube dva puta“). „Korisnik“ sluša, a zatim izvršava rutinu tačno onako kako je opisano . 2. Zamenite uloge kako bi oba učenika mogla da vežbaju davanje instrukcija i njihovo praćenje. 2. Nakon toga, razgovarajte o tome šta je bilo lako ili teško u razumevanju i praćenju instrukcija i kako se to odnosi na razbijanje problema na korake u programiranju. Potrbean materijal: • Nisu potrebni posebni materijali, samo prostor za igranje uloga. • Opciono: Obezbedite kontrolnu listu ili primer rutine kako biste pomogli učenicima da se pripreme . Ishodi učenja: • Učenici će iskusiti kako da pretvore svakodnevne rutine u niz instrukcija, slično programiranju .



Sadržaj	Opis	Sadržaj
		Poboljšaće svoje komunikacijske veštine i razumevanje uslovne logike i sekvenciranja .
Provera		1. Šta je program ? A) Vrsta hardvera koji se koristi u računarima . B) Igra koja se igra na računaru . C) Skup instrukcija koje računar prati da bi izvršio zadatak . ✓ D) Uređaj koji čuva podatke .
	Kviz	2 Šta od sledećih primera je promenljiva u programiranju ? A) Komanda za štampanje teksta na ekranu . B) Mesto skladištenja koje sadrži podatke, kao što su broj ili tekst . ✓ C) Poruka koja objašnjava šta kod radi . D) Vrsta programskog jezika .
		3. Šta radi petlja u programu ? A) Proverava da li je uslov tačan ili netačan . B) Ponavlja skup instrukcija više puta . ✓



Sadržaj	Opis	Sadržaj
		C) Čuva podatke za kasniju upotrebu . D) Šalje podatke na drugi računar .
		4. Zašto je debugovanje važno u kodiranju ? A) Da biste dodali nove funkcije programu . B) Da biste identifikovali i ispravili greške ili bagove u kodu . ☒ C) Da bi program brže radio . D) Da biste preveli kod na različite jezike .