Садржај

Предговор 1

Поглавље 1. Рачунарске мреже, интернет и веб 3

Рачунарске мреже 4
Поделе рачунарских мрежа 6
Повезивање са интернетом 8
Протоколи 10
IP шема адресирања 14
Веб и систем доменских имена – *DNS* 18
Клијентске и серверске технологије 21
Статички и динамички веб 23
Пример коришћења клијент-сервер веб-апликације 25
Безбедност 28

Поглавље 2. Основно о креирању веб-сајтова 31

Креирање веб-сајтова 32
Дизајн веб-страница 32
Организација веб-сајта 34
Ауторска права 35
Готова веб-дизајн решења 36
Креирање сајта употребом готовог решења – *Google Sites* 37

Поглавље 3. Креирање веб-страница употребом описног језика *HTML* . . 45

Описни језик *HTML* 46
Креирање статичких веб-страница 48
Основни *HTML* тагови 49
Глава *HTML* документа и таг *HEAD* 50
Наслови 52
Форматирање текста 53
Листа 54
Табела 55
Слике 57
Линкови 58

Уметнути садржај 58
Елемент *FORM* 59
Пример статичких веб-страница 62

Поглавље 4. Дизајн веб-страница употребом каскадних стилова *CSS* . . . 67

Каскадни стилови – *CSS* 68
HTML документи и каскадна листа стилова 70
CSS и елемент *DIV* 71
Елементи за распоред садржаја на веб-страници 72
Пример веб-сајта креираног употребом *HTML* и *CSS* докумената . . . 73
Измена дизајна 84
Пројектни рад 86

Поглавље 5. Скрипт језик *JavaScript* за клијентско програмирање 87

Скрипт језик *JavaScript* 88
Кратак преглед скрипт језика *JavaScript* 89
Скрипт језик и елемент *FORM* 95
Скрипт језик, модел *DOM* и метода *getElementById* 98
Пример 1 101
Пример 2 102
Пример 3 105
Пример 4 108
Пример 5 109
Пример 6 111

Поглавље 6. Креирање игрица у виду аплета употребом алата *Greenfoot* 115

Кратак преглед програмског језика *Java* 116
Аплети 123
Креирање веб-странице са аплетом употребом алата *Greenfoot* 124
Пример 1 128
Пример 2 132
Пример 3 138

Поглавље 7. Серверско програмирање – *ASP.NET* и *Visual Studio* 145

Програми који се извршавају на серверу 146
Кратак преглед програмског језика *C#* 147

Креирање *ASP.NET* веб-сајта 150
Пример 1 154
Пример 2 156
Пример 3 158
Пример 4 160
Пример 5 163
Пример 6 166
Пример 7 169
Пример 8 172
Пример 9 175
Пример 10 181

Додатак 1. Креирање *Prezi* налога и онлајн презентације 185

Додатак 2. Креирање *Google* налога 188

Додатак 3. Табла са цедуљицама *Lino* 190

Додатак 4. Инсталација алата *Greenfoot* 193

Додатак 5. Инсталација алата *VS Express 2013 for Web* 196

Додатак 6. Онлајн учионица *w3schools* 199

Упутство за наставнике 201
Упознавање са програмирањем на вебу кроз делове књиге 201
Упутство за реализацију наставе 203
Организација диференциране наставе 205
Рад на пројектима 207
Међупредметне компетенције и примена стечених знања 208
Савесна употреба информационо-комуникационих технологија . . . 209
Инсталација софтвера и додатна упутства 210

Литература 211

Упутство за наставнике

- Упознавање са програмирањем на вебу кроз делове књиге
- Упутство за реализацију наставе
- Организација диференциране наставе
- Рад на пројектима
- Међупредметне компетенције и примена стечених знања
- Савесна употреба информационо-комуникационих технологија
- Инсталација софтвера и додатна упутства

Упознавање са програмирањем на вебу кроз делове књиге

Ученици су се већ добро упознали са програмирањем кроз предмете које су имали током прва три разреда у Рачунарској гимназији. Планом и програмом овог предмета је предвиђено да се ученици упознају са напредним техникама програмирања, и то са **програмирањем за веб**.

Пре него што се приступи програмирању за веб, потребно је да се ученици упознају са клијент-сервер организацијом садржаја на вебу, као и описним језиком *HTML* и каскадним стиловима *CSS* који су у основи свих садржаја на вебу.

1. **Рачунарске мреже, интернет и веб** – Први део књиге садржи основну причу о повезивању рачунара у мреже, интернету и вебу. Са неким од појмова су се ученици већ упознали током претходних разреда. Сада је потребно да обнове и систематизују знање. Како је најважнији задатак предмета Напредне технике програмирања да се ученици упознају са креирањем програмског кода који се извршава на клијенту, и креирању програмског кода који се извршава на серверу, посебно је важно да разумеју начин функционисања клијент-сервер технологија.

2. **Основно о креирању веб-сајтова** – Иако је главна тема предмета програмирање, потребно је да се ученици, на основном нивоу, упознају и са другим темама које су везане за креирање садржаја на вебу, као што су дизајн веб-страница, организација веб-сајтова, као и ауторска права која треба да се поштују приликом употребе информација из различитих извора за креирање веб-сајтова. Као илустрација, у овом поглављу је приказано креирање веб-сајта употребом готовог веб-дизајн решења. Пример је толико једноставан да свако може да га без потешкоћа савлада, а служи да ученици добро разумеју шта све чини веб-сајт, пре него што самостално крену да их креирају од почетка.
3. **Креирање веб-страница употребом описног језика *HTML*** – Свака веб-страница је *HTML* документ, и пре него што се веб-страници дода функционалност у виду програмског кода, потребно је да ученици савладају основне елементе описног језика *HTML*, разумеју структуру и организацију *HTML* документа, и буду у стању да самостално креирају статичке веб-странице. Како се ученици са овим нису сретали раније, и најједноставнији примери су детаљно објашњени.
4. **Дизајн веб-страница употребом каскадних стилова *CSS*** – За дизајн веб-страница се користе каскадне листе стилова, и након што ученици савладају основе описног језика *HTML*, а пре него што пређу на програмирање, потребно је да се упознају и са каскадним стиловима.
5. **Скрипт језик *JavaScript* за клијентско програмирање** – Прво веб програмирање са којим се ученици упознају је писање програмског кода који је део *HTML* документа. Програмски код писан скрипт језиком *JavaScript* се заједно са *HTML* документом учитава у веб-браузер и ту на клијенту се и извршава. У примерима су објашњене само специфичности скрипт језика *JavaScript* и како се пише код унутар *HTML* документа, док ученицима нема потребе објашњавати рад са променљивима, операторе, наредбу доделе, наредбе гранања и циклуса и сл, јер су све те теме савладали током претходних разреда.
6. **Креирање игрица у виду аплета употребом алата *Greenfoot*** – Програмски код који се извршава на клијенту може да буде креиран и у виду аплета, програмског кода који је издвојен у засебном фајлу, а позива се из *HTML* документа. За ученике који нису у потпуности савладали градиво из програмерских предмета у прва два разреда, окружење *Greenfoot* може јако лепо да послужи као алат у којем би они напредовали у свом програмерском знању. Чињеница да креирају игрице представља велику мотивацију за ученике. Окружење је довољно слично окружењу у којем су програмирали у претходним разредима да се ученици не би осећали као да се хватају у коштац са нечим потпуно непознатим, али је и довољно другачије да би дало довољно простора за напредовање.

7. **Серверско програмирање – *ASP.NET* и *Visual Studio*** – Највећи део књиге је посвећен страницама активним на серверу, серверским контролама и серверском програмирању. Изложени су примери разних контрола за које се програмирају реакције на догађаје, које се извршавају на серверу.

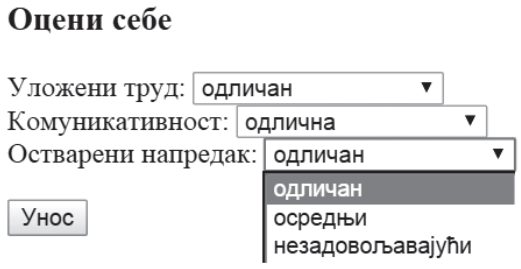
Упутство за реализацију наставе

За сваки пример који је приказан у књизи се предлажу следећа четири корака у раду са ученицима.

1. **Практична реализација примера приказаног у књизи**

   Сви примери у књизи су комплетно решени тако да ученици, пратећи дата упутства, могу самостално да их реализују на рачунару.

   На пример, један од решених примера у књизи је креирање веб-странице са *JavaScript* кодом на којој се налази формулар за евалуацију школе и самоевалуацију личног ангажовања и напредовања. Ученици могу да унесу комплетно приказано решење у рачунар и детаљно се упознају са свим приказаним контролама и програмским кодом.



Део странице са формуларом за самоевалуацију

2. **Преправка и проширење креираног примера**

   Следећи корак за ученике би било проширење креиране веб-странице. Ученици могу да додају још оцена у падајуће листе, на пример, листе могу да садрже оцене од 1 до 10. Након тога би требало и да преправе функцију у којој се рачуна просечна оцена. Могу затим да додају још неколико падајућих листи за оцењивање још неких важних ствари, као што су: активност на часу, спремност да се помогне другу, спремност да се прича о проблемима, сналажљивост у непредвиђеним ситуацијама и сл. Могу да поставе још неколико поља за унос текста тако да посетиоци веб-странице могу, поред оцењивања, да унесу и своје коментаре, тј. образложења датих оцена.

3. **Самостална израда примера који личи на урађени пример**

   Када су се детаљно упознали са новим градивом кроз решен пример и кроз самосталан рад на проширењу истог, ученицима треба да се постави јако сличан задатак да га потпуно сами ураде од почетка и тиме утврде градиво.

   На пример, може да се ученицима постави задатак да креирају веб-страницу на којој би се давале оцене и коментари везани за актуелне филмове или музику.

4. **Самостална израда сложенијег примера који има елементе примера приказаног у књизи**

   Од свих ученика треба захтевати прва два корака, а велики број њих ће успешно савладати сва три. Четврти предложени корак ће бити тежиште рада за ученике којима јако добро иде програмирање, и који ће брже од осталих проћи кроз обавезне прве кораке. Примери које они ураде током прва три корака треба да им буду инспирација за неки пројекат који би сами осмислили и реализовали. Иако би имао неке елементе показаног примера, пројекат би требало да има и неке нове елементе које би ученици сами истражили, а затим и имплементирали у своје решење.

Књига је писана са уважавањем програмерског предзнања које ученици имају. Пре него што се приступи примерима, у књизи су изложени и кратки прегледи приказаних програмских језика. Ови прикази могу да послуже ученицима као подсећање и пружају добру основу за успешно савладавање изложених примера.

Иако су сви примери комплетно решени, детаљније су објашњени појмови са којима се ученици срећу по први пут, а није се улазило у детаљна објашњења свега што им је већ познато из свих предмета који су се бавили програмирањем током прва три разреда.

Сви ученици који су током прва три разреда савладали програмирање са прелазном оценом, бар знају да читају са разумевањем програмерски код. Током реализације примера из књиге на рачунару, ученици треба самостално да разумеју све делове решења. Некима ће то ићи лакше, некима теже, тако да је по потреби потребно ученике упутити на документацију доступну на вебу.

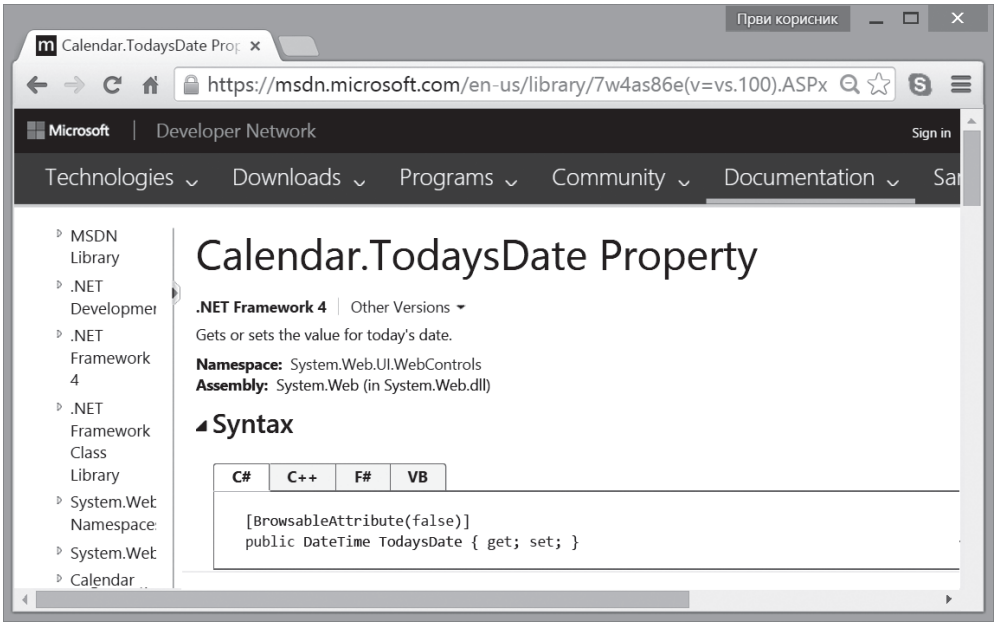
На пример, нема потребе ученицима детаљно објашњавати следећу наредбу:

```
Calendar1.TodaysDate = new DateTime(2005,5,5);
```

Иако се нису раније срели са контролом *Calendar*, ученици у тренутку када се сретну са овим примером, већ више од три године програмирају у програмском језику *C#* и добро им је познато објектно-оријентисано програмирање, приступање пољима објекта путем својства и креирање објеката позивом конструктора. Дакле, требало би да им је јасно шта се овом командом постиже.

Са елементарним познавањем енглеског језика, ученицима значење многих метода и својстава може да буде сасвим јасно из самог назива. На пример, назив својства *TodaysDate*, када се преведе је „ДанашњиДатум”, па је јасно коју вредност ово својсто има.

Уколико постоји потешкоћа да се самим превођењем назива разуме неко својство или метода, потребно је упутити ученике да сами истраже то својсто или методу у документацији која је доступна на вебу. Довољно је укуцати у претраживач неколико кључних речи, као што је назив својства, и појавиће се детаљно објашњење.



Својство TodaysDate објашњено у документацији

Организација диференциране наставе

Иако су ученици у претходна три разреда имали велики број часова програмирања и било је предвиђено да савладају све основне теме из ове области, нису сви ученици савладали програм подједнако успешно и претходне разреде су завршили са различитим знањем које је оцењено различитим оценама.

Књига је организована тако да пружа могућност организације диференциране наставе према којој се предвиђа да се ученицима дају примери и пројектни задаци према њиховим интересовањима, знању и способностима. Изложени су

различити алати и приступи у креирању садржаја на вебу, тако да код појединих ученика може акценат да се стави на једноставније алате, а код осталих на сложеније.

Ученици који су у потпуности савладали програм претходних разреда требало би да што више самостално истражују и раде пројекте према сопственом интересовању.

Показани примери могу да послуже да ученици, који нису у потпуности савладали градиво током прва три разреда, унапреде своје знање које су стицали током претходних разреда.

На пример, план и програм предмета Програмирање и програмски језици у трећем разреду предвиђа да се ученици упознају са развојем класа и виде примере наслеђивања. Приликом развоја класа су учили како да напишу дефиницију неке класе са свим приватним атрибутима, конструкторима и методама. Класе које су креирали личе, на пример, на класу *Hrana* из примера игрице у којој дете хвата здраву, а избегава нездраву храну, решеног у овој књизи:

```
public class Hrana extends Actor
{
    //privatni atribut
    int vy;

    //konstruktor
    public Hrana() {...}

    //javne metode
    public void pada() {...}
    public boolean pala(){...}
}
```

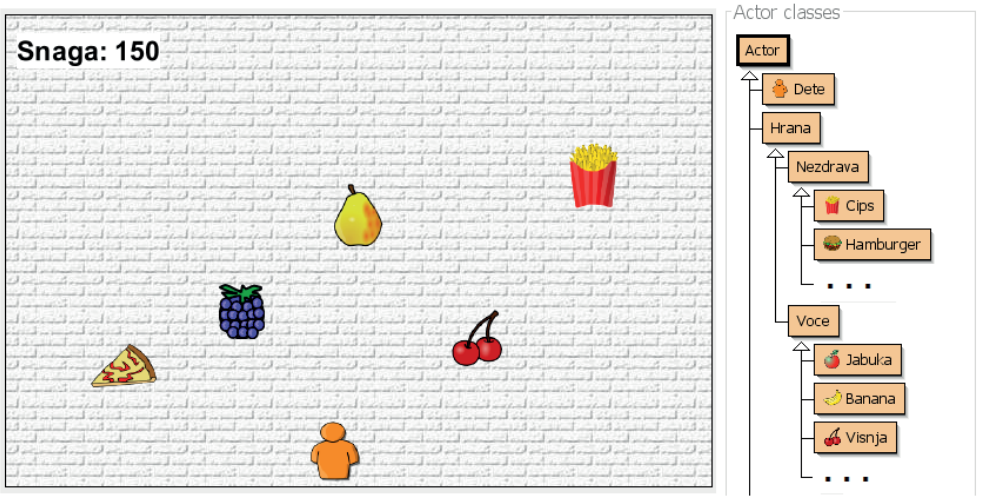
Како је наслеђивање јако важан концепт у објектно-оријентисаном програмирању, ученици су видели и примере класа које наслеђују другу класу, кроз примере сличне примеру класа *Nezdrava* и *Voce* које наслеђују класу *Hrana*.

```
public class Nezdrava extends Hrana
{
    int losFaktor;
    public int getLosFaktor(){...}
}
public class Voce extends Hrana
{
    int dobarFaktor;
    public int getDobarFaktor(){...}
}
```

Током ове године могу да обнове то градиво и да га на тај начин утврде, али у мало другачијем окружењу. Дакле, примери би за њих имали и доста познатог,

али би били и изазов у смислу сналажења у новом окружењу и креирања садржаја за веб. Окружење *Greenfoot* нуди леп графички приказ наслеђивања класа и ученици могу да направе напредак у свом знању уколико у трећем разреду нису до краја савладали програм предмета Програмирање и програмски језици.

Као што је већ раније написано, пре него што приступе самосталном раду на пројектима, ученици могу да стичу знање кроз надоградњу примера изложених у књизи. Поменута игрица може да се надогради тако што би се увели: животи за играча, различити нивои, различити критеријуми по којима се храна класификује, други играч и сл.



Друштвено одговоран пример са наслеђивањем класа у алату Greenfoot

Рад на пројектима

Ученицима се теорија најбоље објашњава кроз конкретне примере, а најбоље се учи кроз истраживање и практичан рад на конкретном задатку, тј. пројекту. У складу са упутством за начин остваривања програма прописаним планом и програмом предмета, у овој књизи су коришћене савремене методе обрађивања градива кроз конкретне примере и пројектну наставу. Након што савладају неколико основних примера изложених у књизи, од ученика се очекује да наставе сами да истражују алате са циљем да научено примене приликом израде пројектних задатака. Основни примери који су приказани могу да послуже као инспирација ученицима за пројекте које би самостално радили.

С обзиром да су ученици у претходна три разреда научили основне елементе програмирања, у оквиру овог предмета треба да продубљују знања и вештине. Једна од јако важних вештина у данашње време је прилагођавање брзим

променама у свету технологија. Када се ученици буду запослили, од њих ће се вероватно захтевати да програмирају у неком окружењу које данас не постоји и које ће тек бити развијено. Важно је оспособити ученике да сами истражују литературу, укључујући и литературу која је доступна на интернету, и да проналазе упутства, повезују знање и примењују научено у конкретним пројектима.

Међупредметне компетенције и примена стечених знања

На почетку сваког дела књиге је наведен списак школских предмета који су у вези са приказаним примерима у том делу. Ове међупредметне компетенције су само пример могућег повезивања предмета Напредне технике програмирања са другим школским предметима.

Узмимо на пример, да је у књизи приказано креирање веб-апликације која решава задатак из математике. На почетку одговарајућег дела књиге, наведена је Математика као једна међупредметна компетенција. Тај приказани пример може да послужи као инспирација да се креира пројекат који би повезао градиво овог предмета са неким другим сличним предметом, на пример, да се креира веб-апликација којом се решава задатак из физике или хемије. Слично је и са другим школским предметима. Уколико је, на пример, приказано како може да се повеже овај предмет са историјом, на сличан начин га је могуће повезати и са географијом, биологијом, филозофијом, астрономијом...

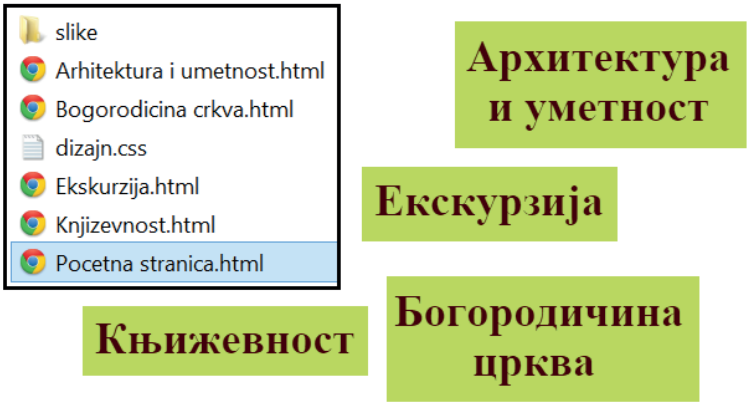
Пожељно је охрабривати ученике, да према својим интересовањима, раде пројектне задатке који су повезани са другим школским предметима.

Поред међупредметних компетенција и примене стечених знања, у неким деловима књиге је приказано како овај предмет може да се повеже и са другим сегментима школског живота. Једноставан сајт којим је у књизи приказано како се сајтови креирају употребом готовог веб-дизајн решења *Google Sites*, се бави проблемом насиља у школи. Израда таквог пројекта може да послужи да се успостави комуникација са родитељима о тој важној теми, тако што ће ученици позвати родитеље да погледају сајт и оставе своје коментаре, на пример, у виду цедуљица.

Могуће је на нивоу одељења креирати и један једноставан сајт где би се објављивали сви креирани пројекти током школске године. Тако би родитељи били у прилици да виде резултате рада у школи и прате напредак своје деце.

Осим што сви примери показују везу програмерских са другим школским предметима, део књиге се бави и примером у којем је на једном месту показана веза више школских предмета. Креиран је веб-сајт о Студеници. Тема средњовековне

Србије се у школи обрађује кроз неколико школских предмета: Српски језик и књижевност, Историја и Уметност. Осим у оквирима ових предмета, ученици могу да се са Студеницом упознају и током школског излета или екскурзије. Добро је за ученике да повежу знање из више предмета, као и стечено лично искуство током путовања, а све кроз креирање веб-сајта.



Једна тема кроз различите школске предмете

Савесна употреба информационо-комуникационих технологија

Ученици, иако дугогодишњи корисници рачунара, често се налазе у ситуацијама у којима несавесно користе информационо-комуникационе технологије. Један од највећих пропуста је то што никада не читају уговор о коришћењу и правила приватности, а на њих пристају не размишљајући о могућим последицама. Познавање текста уговора о коришћењу и правила приватности не треба да обесхрабри ученике у употреби технологије, али може да допринесе савесној и безбедној употреби истих.

Узмимо на пример друштвену мрежу *facebook*. Део њиховог уговора о коришћењу подразумева сагласност да се прикупљају информације о уређајима са којих се приступа њиховим услугама. Између осталог, свако ко користи друштвену мрежу *facebook*, је пристао да *facebook* прикупља и користи његове податке, укључујући, на пример, податак о томе коју мобилну мрежу користи, као и спискове фајлова и софтвера које има и користи на рачунару или паметном телефону са којег приступа друштвеној мрежи.

Device information.
We collect information from or about the computers, phones, or other devices where you install or access our Services, depending on the permissions you've granted. We may associate the information we collect from your different devices, which helps us provide consistent Services across your devices. Here are some examples of the device information we collect:

- Attributes such as the operating system, hardware version, device settings, file and software names and types, battery and signal strength, and device identifiers.
- Device locations, including specific geographic locations, such as through GPS, Bluetooth, or WiFi signals.
- Connection information such as the name of your mobile operator or ISP, browser type, language and time zone, mobile phone number and IP address.

Један део уговора о коришћењу друштвене мреже facebook (преузето 2015. године)

У оквиру предмета Рачунарство и друштво, ученици су се бавили темама савесне употребе информационо-комуникационих технологија. Потребно је још да, поред свести о томе шта је савесна употреба технологије, стекну и навику да кроз све што раде то и практикују. Дакле, не само да знају шта су ауторска права, већ да у свим пројектним задацима поштују ауторска права и наведу извор употребљених информација и материјала.



Навођење аутора употребљених материјала

Инсталација софтвера и додатна упутства

Књига нуди кратка и прегледна упутства за инсталацију потребног софтвера, креирање онлајн налога и употребу неких додатних алата. Потребно је упутити ученике на ове додатке да би могли без потешкоћа код куће да инсталирају софтвер и користе предложене алате. Сви софтверски алати који су наведени у књизи су бесплатни за коришћење.