

სილაბუსი

ვებ დეველოპმენტის საფუძვლები

- გაიგებთ როგორ მუშაობს ვებსაიტები და რა წვლილი შეაქვს HTML-ს, CSS-სა და JavaScript-ს.
- გაიგებთ როგორ მუშაობს ინტერნეტი.
- დაიწყებთ კოდინგი HTML-ით, CSS-ით, და JavaScript-ით.
- ისწავლით და გაიგებთ git-ის, GitHub-ისა და version control-ის შესახებ.
- გამოიყენებთ Unix command line-ს რათა გახდეთ power user და დაწეროთ bash ბრძანებებს.
- შეიძენთ troubleshooting და debugging უნარების, რათა თქვენს პროექტებში გამოიყენოთ.

HTML5

- ისწავლით HTML-ის სინტაქსის ანატომიას, რათა ააწყოთ თქვენი ვებსაიტები.
- გაიგებთ HTML boilerplate-ისა და HTML doctypes-ის შესახებ.
- ისწავლით როგორ უნდა მოაწყოთ ტექსტი HTML-ში.
- როგორ უნდა ააწყოთ არათანმიმდევრული და თანმიმდევრული ლისტები სიები HTML.
- როგორ უნდა ჩასვათ სურათები HTML-ის გამოყენებით.
- როგორ უნდა შექმნათ hyperlinks, anchor tags-ის გამოყენებით.
- გაიგებთ როგორ უნდა გამოიყენოთ HTML tables კონტენტისთვის და როგორ უნდა გამოიყენოთ tables layout-თვის.
- ისწავლით HTML-ის საუკეთესო პრაქტიკებს.
- გაიგებთ HTML ფორმების შესახებ და შექმნით მარტივ contact me ფორმას.
- HTML Divs და როგორ უნდა განაცალკევოთ კონტენტი CSS-ით ეფექტურად გასტილვისთვის.

CSS 3

- გაიგებთ რა არის **cascading style sheets** და როგორ შეგიძლიათ მისი გამოყენება თქვენი ვებსაიტის გასტილვისთვის.
- როგორ უნდა გამოიყენოთ **CSS selectors** და **properties**.
- ისწავლით როგორ უნდა გამოიყენოთ **inline**, **internal** და **external CSS**.
- გაიგებთ **CSS**-ით კოდინგის საუკეთესო პრაქტიკებს.
- ისწავლით **CSS**-ის **sizing** მეთოდებს.
- ისწავლით **CSS** -ის ანატომიასა და სტრუქტურაზე. **CSS**-ის სპეციფიკურობა და სტილის იერარქიის დანერგვა.
- **Class vs. Ids** და როგორ უნდა დაუკავშიროთ ერთმანეთს.
- **CSS display** და მისი გამოყენება ლაიაუტის აწყობისას.
- როგორ უნდა გამოიყენოთ **CSS** სტატიკური, რელაციური და აბსოლუტური პოზიციონირების სისტემები.
- ფონტების გასტილვა **CSS**-ით და **web safe fonts**-ის გამოყენება.
- ელემენტების ცენტრალიზება **CSS**-ის გამოყენებით.
- ვებსაიტის დიზაინის საფუძვლები და ტიპოგრაფია.
- როგორ უნდა გამოიყენოთ **CSS float** და **clear**.
- როგორ უნდა დააკომბინიროთ **CSS selectors** და **selector priority**-ის გააზრება.

BOOTSTRAP4

- რისფონსიული იუზერ ინტერფეისის შექმნა.
- როგორ უნდა გამოიყენოთ **Balsamiq** რათა შექმნათ მოკაპები და ვაიარფრეიმები.
- ვებსაიტის **UI** დიზაინის საწყისები .
- როგორ უნდა დააინსტალიროთ **Bootstrap framework**.
- ისწავლით **Bootstrap grid layout** სისტემას.
- როგორ უნდა გამოიყენოთ **bootstrap** კონტეინერები, რათა შექმნათ თქვენი ვებსაიტის ლეიაუტი ადვილად.
- ისწავლით სხვადასხვა **Bootstrap** კომპონენტების გამოყენებას, როგორიცაა **buttons**.

- Font Awesome-ის გამოყენებით სიმბოლოების დამატება.
- ისწავლით Bootstrap carousels-ის გამოყენებას.
- დაამატებთ Bootstrap cards თქვენს ვებსაიტს.
- გამოიყენებთ Bootstrap navigation bars.

JAVASCRIPT ES6

- კოდის სანყისები.
- კოდის დაწყება alerts-ებითა და prompts-ებით.
- გაიგებთ ცვლადებსა და მინაცემთა ტიპებზე ჯავასკრიპტში.
- ცვლადების დასახელება JS-ში.
- Strings-თან და რიცხვებთან მუშაობა.
- რენდომიზაცია და ლოგიკური ოპერაციები.
- Loops, collections და Conditionals.
- ფუნქციები და ინოვაციური სქემები.
- ECMAScripts-ზე დისკუსია.
- შუალედური დონის JavaScript.
- ისწავლეთ JS Expressions-ის გამოყენებას, ოპრეტორები, Statements და დეკლარაციები.
- ობიექტზე ორიენტირებული პროგრამირება.
- JS Objects და პროტოტიპები.
- `This`, Scope და Closures.
- ობიექტები და პროტოტიპები.
- Refactoring და Debugging.

DOCUMENT OBJECT MODEL (DOM)

- ისწავლეთ HTML-ზე ანყობილი ვებსაიტების ხისებრ სტრუქტურაზე.
- დოკუმენტის განხილვა ობიექტის ნოტაციის გამოყენებით.
- პრობლემების ჩაშლა და კოდინგის საუკეთესო პრაქტიკები.
- HTML ელემენტებზე მანიპულაცია და მათი შეცვლა თქვენი DOM-ის ცოდნის გამოყენებით.

JQUERY

- jQuery framework-ის ინსტალაცია და გამოყენება.
- jQuery-ის ფუნქციონალის სწავლა.
- ფუნქციების შესავალი jQuery-ში.
- text, styles და attributes-ით მანიპულაცია.
- jQuery-ით შექმნილი ანიმაციები.
- თქვენი jQuery კოდნის გამოყენებით, ვებსაიტი ინტერაქციულად გადაქცევა.
- ინიცირებულ მოვლენებზე იუზერისთვის პასუხის დაბრუნება jQuery-ით.

THE UNIX COMMAND LINE

- როგორ უნდა გამოიყენოთ ძირითადი bash ბრძანებები Unix/Linux ტერმინალში.
- ფაილებითა და საქალაქდებოთ მანიპულირება GUI-ს (მომხმარებლის გრაფიკული ინტერფეისის) გარეშე.
- როგორ შეგიძლიათ გადმოწერა და დაინსტალირება თქვენს კომპიუტერში command line-ის გამოყენებით.

GIT GITHUB AND VERSION CONTROL

- Git-ის გამოყენება ვერსიების კონტროლისთვის და Git forking, branching, cloning კოლაბორაციისთვის.
- GitHub-ის გამოყენება remote repository-თვის.
- Checkout და ცვლილებების უკან დაბრუნება git-ის გამოყენებით.
- GitHub-ის და git-ის გამოყენება Xcode-ით.

NODE.JS

- back-end development-ის კომპონენტები, MVC framework-თან მუშაობა

- მინაცემთა ტიპები, ობიექტები, ობიექტზე ორიენტირებული პროგრამირება და კლასები **back- end development**-ის კონტექსტში.
- **Server-Side JavaScript.**
- **Node**-ის გამოყენება **command line**-ით.
- **NPM.**
- **JavaScript**-ის დაბილდვის პროცესი.
- **Event Loop** და **Emitters.**
- ფაღურ სისტემასთან ინტერაქცია.
- მოდულები.
- **Native Node drivers.**

EXPRESS.JS

- გაიგებთ როგორ უნდა დააინსტალიროთ და გამოიყენოთ **express**-ი **Node**-ის აპლიკაციებში.
- **Node**-სა და **Express**-ზე აწყობილი სერვერების შექმნა.
- **RESTful Routing**-ი **Express**-ით.
- **Middleware**-ის გამორეზება **Node**-ს აპლიკაციებში.

APPLICATION PROGRAM INTERFACES (APIS)

- გაიგებთ რა არის **API** და როგორ მუშაობს ის.
- **HTTP** სილრმისეულად.
- **API**-ის გამოძახება.
- **API** დოკუმენტაციის წაკითხვა.
- ძირეული **API** აუთენტიფიკაცია.
- **Server**-ებს შორის კომუნიკაცია.
- **JSON vs XML**, ინფორმაციის გადაგზავნა კაბელის მეშვეობით.

EJS

- რას აეთებს EJS და როგორ უნდა გამოიყენოთ ის Node-სა და Express-თან.
- სქემები EJS-ით
- კოდის გაშვება EJS სქემებში
- ინფორმაციის მიმოცვლა სერვერიდან სქემამდე და პირიქით
- layouts/partials-ის შექმნა EJS-ით

DATABASE FUNDAMENTALS

- მონაცემთა შორის კავშირი
- მონაცემთა Model-ების შექმნა
- რელაციური მონაცემთა ბაზები
ალტერნატიული მონაცემთა ბაზები
- Entity Relationship Modelling (ERM) and Object Relational Mapping (ORM)

SQL DATABASES

- მონაცემთა ბაზების სქემებთან მუშაობა
- შექმნა-წაკითხვა-განახლება-წაშლა (CRUD) ოპერაციები
- Joins(შეერთებები)
- Querying SQL databases

NOSQL DATABASES WITH MONGODB AND MONGOOSE

- სერიალიზაცია.
- NoSQL მონაცემთა მოდელირება.
- Document Databases (MongoDB).
- შექმნა-წაკითხვა-განახლება-წაშლა (CRUD) .
- NoSQL საუკეთესო პრაქტიკები.
- Mongo Shell და command line-ის გამოყენება.
- MongoDB-ის დაინსტაირება.
- რილეიშენების მაპინგი MongoDB-ით.

- **object-data** მოდელინგ ბიბლიოთეკის (**Mongoose**)-ის გამოყენება.
რათა მონაცემებთან ადვილად იმუშაოთ.

DEPLOYMENT

- ისწავლით **hosting**-ს და **deployment**-ს.
- სტატიკური ვებსაიტების დაჰოსტვა **GitHub** გვერდებით.
- **server-based** აპლიკაციების დადგენა **Heroku**-თი.
- მონაცემთა ბაზების დადგენა **Mongo Atlas**-ით.

BUILDING RESTful APIs

- **REST**-ის გააზრება და პრინციპები **API** დიზაინის უკან.
- **MongoDB GUI Robo 3T**-თან მუშაობა.
- **GET, POST, PUT, PATCH** და **DELETE** -ის იმპლემენტაცია და **API**-ის **0**-დან აწყობა.
- გაიგებთ და გამოიყენებთ **chained route handlers Express**-დან.

AUTHENTICATION and SECURITY

- გაიგებთ იდენტიფიკაციისა და მომხმარებლის ინფორმაციის დაცულად შენახვის საჭიროებას.
- ისწავლით **Encryption**-ზე და გამოიყენებთ მას რათა შეინახოთ მონაცემები დაცულად.
- ისწავლით **Hashing** და **Salting** და გამოიყენებთ **bcrypt**-ით.
- სესიებისა და ქუქიების გამოყენება, რათა გაგრძელდეს მომხმარებლის **log in** სესიები.
- ლოკალური იდენტიფიკაციის აწყობა ნახაზიდან.
- **Passport**-ის დანერგვა რათა ადვილად მოხდეს მომხმარებლების იდენტიფიკაცია.
- გაიგებთ და გამოიყენებთ **environment variables**, რათა შეინარჩუნოთ **personal keys** უსაფრთხოდ.

- გაიგებთ და გამოიყენებთ **OAuth 2.0** რათა მომხმარებლები შევიდნენ **Google**-სა და **Favebook**-ის გამოყენებით.

REACT.JS

- ისწავლით **front-end** დეველოპმენტს **React**-ით.
- გაიგებთ როდის და როგორ უნდა გამოიყენოთ **React** კომპონენტები.
- ისწავლით **Props**-ის გადაცემას და მათთან მუშაობას.
- ისწავლით **JSX**-ის წერას და გაიგებთ **JSX** სინტაქსს.
- ისწავლით **React DOM**-ის შესახებ.
- ისწავლით **State Management**-ს **React**-ში.
- ისწავლით **React Hooks**-ის შესახებ.
- ისწავლით პირობით რენდერინგს **React**-ში.
- გაიგებთ განსხვავებას კლასზე დაფუძნებულ და ფუნქციონალურ კომპონენტებს შორის.

JAVASCRIPT ES6

- **Arrow** ფუნქციები.
- **Map/Filter/Reduce**.
- **Find/FindOne/FindIndex**.
- **Import/Export and Modules**.
- **Object** და **Array Destructuring**.
- **Spread Operator**.