


# S.Leschev FullStack Developer Roadmap

## Awards

Ranking #Dev: Global TOP 300 ([Certificate](#))



**Sergey Leschev**  
sergeyleschev

Rank 386

IT-Entrepreneur / Tech Lead. Sailor Of The Year 2016, 2017. Winner ProGolf 2018, 2019. #sailing #golf #tennis

Solved Problems

1682

Solved

Easy

463 / 566

Beats 99.96%

Medium

890 / 1199

Beats 99.96%

Hard

329 / 488

Beats 99.93%



**Languages:** TypeScript, Shell, Database (T-SQL, PL/SQL, MySQL), Concurrency (Python3).

**Algorithms:** linked lists, binary search, hash table, queue/stack, dfs/bfs, sort, heap/hash, two pointers, sliding window, tree, greedy problems etc.

# Golden Award for the Year of the Tiger Challenge



**Algorithmic skills:** Dynamic programming, Greedy algorithms, Catepillar method, Binary search algorithm, Fibonacci numbers, Euclidean algorithm, Sieve of Eratosthenes, Prime and composite numbers, Maximum slice problem, Stack and Queues, Sorting, Time Complexity, Arrays, Prefix Sums, Leader, etc.

**Contest:** Algorithms, SQL, Data Structures, Bitwise operations (bit-ops), Frontend.

---

Tapping on a link will take you to relevant certificates.

## Front-End

---

- ☒ Basics
  - ☒ HTML
    - ☒ Learn the basics of HTML
    - ☒ Make a few pages as an exercise
  - ☒ CSS
    - ☒ Learn the basics of CSS
    - ☒ Style pages from previous step
    - ☒ Build a page with grid and flexbox
  - ☒ JS Basics
    - ☒ Get familiar with the syntax
    - ☒ Learn basic operations on DOM
    - ☒ Learn mechanisms typical for JS (Hoisting, Event Bubbling, Prototyping)
    - ☒ Make some AJAX (XHR) calls
    - ☒ Learn new features (ECMA Script 6+)
    - ☒ Additionally, get familiar with the jQuery library
- ☒ General Development Skills
  - ☒ Learn GIT, create a few repositories on GitHub, share your code with other people
  - ☒ Know HTTP(S) protocol, request methods (GET, POST, PUT, PATCH, DELETE, OPTIONS)
  - ☒ Get familiar with terminal, configure your shell (bash, zsh, fish)
  - ☒ Studied algorithms and data structures in the university
  - ☒ Studied design patterns in the university
- ☒ Learn React on [official website](#) or complete some [courses](#)
- ☒ Get familiar with tools that you will be using
  - ☒ Package Managers
    - ☒ [npm](#)
    - ☒ [yarn](#)
  - ☒ Task Runners

- ☒
  - ☒ npm scripts
  - ☒ gulp
  - ☒ Webpack
  - ☒ Rollup
  - ☒ Parcel
- ☒ Styling
  - ☒ CSS Preprocessor
    - ☒ Sass/CSS
    - ☒ PostCSS
    - ☒ Less
    - ☒ Stylus
  - ☒ CSS Frameworks
    - ☒ Bootstrap
    - ☒ Materialize, Material UI, Material Design Lite
    - ☒ Bulma
    - ☒ Semantic UI
  - ☒ CSS Architecture
    - ☒ BEM
    - ☒ CSS Modules
    - ☒ Atomic
    - ☒ OOCSS
    - ☒ SMACSS
    - ☒ SUITCSS
  - ☒ CSS in JS
    - ☒ Styled Components
    - ☒ Radium
    - ☒ Emotion
    - ☒ JSS
    - ☒ Aphrodite
- ☒ State Management
  - ☒ Component State/Context API
  - ☒ Redux
    - ☒ Async actions (Side Effects)

- ☒ [Redux Thunk](#)
  - ☒ [Redux Better Promise](#)
  - ☒ [Redux Saga](#)
  - ☒ [Redux Observable](#)
- ☒ **Helpers**
  - ☒ [Rematch](#)
  - ☒ [Reselect](#)
- ☒ **Data persistence**
  - ☒ [Redux Persist](#)
  - ☒ [Redux Phoenix](#)
- ☒ [Redux Form](#)
- [MobX](#)
- ☒ **Type Checkers**
  - ☒ [PropTypes](#)
  - ☒ [TypeScript](#)
  - ☒ [Flow](#)
- ☒ **Form Helpers**
  - ☒ [Redux Form](#)
  - ☒ [Formik](#)
  - ☒ [Formsy](#)
  - ☒ [Final Form](#)
- ☒ **Routing**
  - ☒ [React-Router](#)
  - ☒ [Router5](#)
  - ☒ [Redux-First Router](#)
  - ☒ [Reach Router](#)
- ☒ **API Clients**
  - ☒ **REST**
    - ☒ [Fetch](#)
    - ☒ [SuperAgent](#)
    - ☒ [axios](#)
  - ☒ **GraphQL**
    - ☒ [Apollo](#)

- ☒ [Relay](#)
- ☒ [urql](#)
- ☒ Utility Libraries
  - ☒ [Lodash](#)
  - ☒ [Moment](#)
  - ☒ [classnames](#)
  - ☒ [Numeral](#)
  - ☒ [RxJS](#)
  - ☒ [ImmutableJS](#)
  - ☒ [Ramda](#)
- ☒ Testing
  - ☒ Unit Testing
    - ☒ [Jest](#)
  - ☒ End to End Testing
    - ☒ [Selenium, Webdriver](#)
  - ☒ Integration Testing
    - ☒ [Karma](#)
- ☒ Internationalization
  - ☒ [React Intl](#)
  - ☒ [React i18next](#)
- ☒ Server Side Rendering
  - ☒ [Next.js](#) [sources](#)
- ☒ Static Site Generator
  - ☒ [Gatsby](#)
- ☒ Backend Framework Integration
  - ☒ [React on Rails](#)
- ☒ Mobile
  - ☒ [React Native](#)
- ☒ Desktop
  - ☒ [Electron](#)
  - ☒ [React Native Windows](#)

# Back-End

---

- ☒ Prerequisites
- ☒ [JavaScript](#)
- ☒ [NPM](#)
- ☒ [Node.js](#)
- ☒ [ECMAScript](#)
- ☒ General Development Skills
- ☒ Learn GIT, create a few repositories on GitHub, share your code with other people
- ☒ Know HTTP(S) protocol, request methods (GET, POST, PUT, PATCH, DELETE, OPTIONS)
- ☒ Studied algorithms and data structures in the university
- ☒ Studied design patterns in the university
- ☒ [Clean code](#)
- ☒ Web Frameworks
- ☒ [Express.js](#)
- ☒ [Adonis.js](#)
- ☒ [Meteor.js](#)
- ☒ [Nest.js](#)
- ☒ [Sails.js](#)
- ☒ [Koa.js](#)
- ☒ [Loopback.io](#)
- ☒ [egg.js](#)

- ☒ [midway](#)
- ☒ Databases
  - ☒ Relational
    - ☒ [SQL Server](#) [sources.sql](#)
    - ☒ [PostgreSQL](#) [sources.sql](#)
    - ☒ [MariaDB](#) [sources.sql](#)
    - ☒ [MySQL](#) [sources.sql](#)
  - ☒ Cloud Databases
    - ☒ [CosmosDB](#) [sources](#)
    - ☒ [DynamoDB](#) [sources](#)
  - ☒ Search Engines
    - ☒ [ElasticSearch](#)
    - ☒ [Solr](#)
    - ☒ [Sphinx](#)
  - ☒ NoSQL
    - ☒ [MongoDB](#) [sources](#)
    - ☒ [Redis](#) [sources](#)
    - ☒ [Apache Cassandra](#)
    - ☒ [LiteDB](#)
    - ☒ [RavenDB](#)
    - ☒ [CouchDB](#)
- ☒ Caching
  - ☒ [Node-Cache](#)
  - ☒ Distributed Cache
    - ☒ [Redis](#) [sources](#)
    - ☒ [Memcached](#) [sources](#)
  - ☒ [Memory Cache](#)
- ☒ Logging
  - ☒ Log Frameworks
    - ☒ [Node-Loggly](#)



- ☒ [Winston](#)
- ☒ [Node-Bunyan](#)
- ☒ [Morgan](#)
- ☒ Log Management System
  - ☒ [Sentry.io](#)
  - ☒ [Loggly.com](#)
- ☒ Template Engines
- ☒ [Mustache.js](#)
- ☒ [Handlebars](#)
- ☒ [EJS](#)
- ☒ [Pug](#)
- ☒ [Nunjunks.js](#)
- ☒ Real-Time Communication
- ☒ [Socket.IO](#)
- ☒ Typed Superset
- ☒ [TypeScript](#)
- ☒ [Flow](#)
- ☒ API Clients
  - ☒ REST
  - ☒ [Node-Rest-Client](#)
  - ☒ [Axios](#)
  - ☒ [GraphQL](#)
  - ☒ Authentication
- ☒ Good to Know
  - ☒ [Async.js](#)
  - ☒ [PM2](#)
  - ☒ [Commander.js](#)

- ☒ [Passport](#)
- ☒ [Nodemailer](#)
- ☒ [Marked](#)
- ☒ [ESLint](#)
- ☒ Testing
  - ☒ Unit, Behavior, Integration Testing
  - ☒ [Jest](#)
  - ☒ [Jasmine](#)
  - ☒ [Chai](#)
  - ☒ [Mocha](#)
  - ☒ [Enzyme](#)
  - ☒ [Sinon](#)
  - ☒ E2E Testing
  - ☒ [Selenium](#)
  - ☒ [Puppeteer](#)
- ☒ Task Scheduling
  - ☒ [Agenda](#)
  - ☒ [Cronicle](#)
  - ☒ [Node-Schedule](#)
- ☒ MicroServices
  - ☒ Message-Broker
  - ☒ [RabbitMQ](#)
  - ☒ [Apache Kafka](#)
  - ☒ [ActiveMQ](#)
  - ☒ [Azure Service Bus](#)
  - ☒ Message-Bus
  - ☒ [Distribus](#)
  - ☒ [BusMQ](#)
- ☒ [Design-Patterns](#)
- ☒ [JavaScript Algorithms and Data Structures](#)

# Computer Science

---

- ✓ Computer Science knowledge
  - ✓ Algorithms certificate
    - ✓ Sorting certificate
    - ✓ Graph Theory certificate
      - ✓ Trees certificate
    - ✓ Strings certificate
    - ✓ Greedy certificate
    - ✓ Dynamic Programming certificate
    - ✓ Bit Manipulation certificate
    - ✓ Recursion certificate
    - ✓ Game Theory certificate
    - ✓ NP Complete certificate
    - ✓ Big-O notation certificate
  - ✓ Abstract Data Types certificate
    - ✓ Stack certificate
    - ✓ Array certificate
    - ✓ List certificate
    - ✓ Map certificate
    - ✓ Multimap certificate
    - ✓ Set certificate
    - ✓ Multiset (Bag) certificate
    - ✓ Graph certificate
      - ✓ Tree certificate
    - ✓ Queue certificate
    - ✓ Priority Queue certificate
    - ✓ Double-ended priority queue certificate
    - ✓ Double-ended queue certificate
  - ✓ System design
    - ✓ Design large-scale systems / Amazon, Dropbox, Instagram, Facebook, Netflix, Pinterest, Twitter, Uber, Youtube

# Project Guidelines

---

A set of best practices in my projects.

- ✓ [Git](#)
- ✓ [Documentation](#)
- ✓ [Environments](#)
- ✓ [Dependencies](#)
- ✓ [Testing](#)
- ✓ [Structure and Naming](#)
- ✓ [Code Style](#)
- ✓ [Logging](#)
- ✓ [API](#)

## Licenses & certifications

---

- 🏆 LeetCode Global TOP 300 (TypeScript: [Certificate](#), Sources: [TypeScript](#)).
- 🏆 Golden Award for the Year of the Tiger Challenge (TypeScript: [Certificate](#), Sources: [Codility](#)).
- LinkedIn Skill Assessment (Front-End): [Front-end Development](#), [Angular](#), [React](#), [Javascript](#), [HTML](#), [CSS](#), [jQuery](#).
- LinkedIn Skill Assessment (Back-End): [Node.js](#), [Java](#), [Spring Framework](#), [Scala](#), [C#](#), [.NET Framework](#), [Unity](#), [Python \(Programming Language\)](#), [Django](#), [PHP](#), [C \(Programming Language\)](#).
- LinkedIn Skill Assessment (Databases): [MongoDB](#), [NoSQL](#), [Transact-SQL \(T-SQL\)](#), [MySQL](#).
- LinkedIn Skill Assessment (Infra/DevOps): [Bash](#), [Git](#), [Amazon Web Services \(AWS\)](#), [AWS Lambda](#), [Google Cloud Platform \(GCP\)](#), [Microsoft Azure](#), [Hadoop](#), [IT Operations](#).

# Contacts

---

I have a clear focus on time-to-market and don't prioritize technical debt.



#startups #management #cto #swift #typescript #database



Email: [sergey.leschev@gmail.com](mailto:sergey.leschev@gmail.com)



LinkedIn: <https://linkedin.com/in/sergeyleschev>



Twitter: <https://twitter.com/sergeyleschev>



Github: <https://github.com/sergeyleschev>



Website: <https://sergeyleschev.github.io>



PDF: [Download](#)

ALT: SIARHEI LIASHCHOU