

Labo 22 – Verslag My internet startpage

Opleidingsonderdeel: **Web Development 1**

Bachelor in de: **Toegepaste informatica**

Afstudeerrichting: **Applicatieontwikkeling**

Specialisatie: **Cybersecurity & infrastructure**

Leerling: **Jari Opsomer**

Docent: **Pim Debaere**

Academiejaar: **2025-2026**

2 Opdracht: My internet startpage

HTML-code:

```
<!DOCTYPE html>
<html lang="nl">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>My Internet Start Page</title>

  <!-- Bootstrap 5 CSS voor layout en componenten -->
  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet">

  <!-- Eigen stijlbestand -->
  <link rel="stylesheet" href="styles/styles.css">
</head>

<body>
<!-- Alles zit in een Bootstrap container voor een nette centrering -->
<div class="container mt-4">

  <!-- Bovenste sectie: zoekbalk en knop -->
  <section id="zoek-sectie">
    <h1 class="mb-3">My Internet Start Page</h1>

    <label for="command-input" class="form-label fw-
semibold">Zoekopdracht</label>
    <input
      type="text"
      id="command-input"
      class="form-control mb-1"
      placeholder="Enter commando"
      autocomplete="off"
    >
    <!-- Kleine hint tekst onder het invoerveld -->
    <small class="text-muted d-block mb-3">
      Start your commando with "[character] [zoekopdracht]"
    </small>

    <button id="go-button" class="btn btn-success w-100 mb-4">GO!</button>
  </section>

  <!-- Onderste sectie: history van zoekopdrachten -->
  <section id="history-sectie">
    <div class="d-flex justify-content-between align-items-center mb-3">
      <h2 class="mb-0">History</h2>
      <!-- Sorteerknop, tekst wordt via JS aangepast -->
      <button id="sort-btn" class="btn btn-outline-secondary btn-sm">Sort: A
→ Z</button>
    </div>

    <!-- Bootstrap row: cards worden hier dynamisch ingevoegd door JavaScript -
->
    <div id="cards-container" class="row g-3"></div>
  </section>
</div>

<!-- Popup container: wordt leeg gelaten, JS vult dit in bij een fout -->
<div id="popup-overlay" class="popup-overlay d-none"></div>
<div id="popup-box" class="popup-box d-none">
  <div class="popup-header">
```

```

        <h5 id="popup-titel" class="mb-0">Fout</h5>
        <button id="popup-sluit" class="popup-sluit-btn">&times;</button>
    </div>
    <p id="popup-bericht" class="mt-2 mb-0"></p>
</div>

<!-- Bootstrap JS -->
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js">
</script>

<!-- Eigen JavaScript -->
<script src="scripts/code.js"></script>
</body>

</html>

```

CSS-code:

```

/* =====
My Internet Start Page – eigen stijlen
Bootstrap zorgt voor de basisstijlen.
Hier passen we alleen specifieke dingen aan.
===== */

body {
    background-color: #f8f9fa;
    font-family: 'Segoe UI', sans-serif;
}

/* ---- Zoekbalk & knop ---- */

#command-input {
    font-size: 1rem;
    border: 1.5px solid #ced4da;
    border-radius: 6px;
    transition: border-color 0.2s;
}

/* Blauw randje als het veld geselecteerd is */
#command-input:focus {
    border-color: #0d6efd;
    box-shadow: 0 0 0 3px rgba(13, 110, 253, 0.15);
}

#go-button {
    font-size: 1.1rem;
    font-weight: 600;
    letter-spacing: 0.05em;
    padding: 10px;
    border-radius: 6px;
    transition: background-color 0.2s;
}

/* ---- History cards ---- */

/* Elke card neemt op kleine schermen de volledige breedte in,
op grotere schermen krijg je 3 cards naast elkaar (Bootstrap col). */
.zoek-card {
    border: none;
    border-radius: 8px;
    color: #fff;
    padding: 16px;
    box-shadow: 0 2px 6px rgba(0, 0, 0, 0.15);
}

/* Titel in de card (naam van de zoekmachine) */

```

```

.zoek-card .card-naam {
  font-size: 1.1rem;
  font-weight: 700;
  margin-bottom: 4px;
}

/* De zoekopdracht zelf */
.zoek-card .card-query {
  font-size: 0.95rem;
  margin-bottom: 14px;
  opacity: 0.9;
}

/* De "Go!"-link in de card */
.zoek-card .card-link {
  display: inline-block;
  background-color: rgba(255, 255, 255, 0.25);
  color: #fff;
  text-decoration: none;
  padding: 5px 14px;
  border-radius: 5px;
  font-weight: 600;
  font-size: 0.85rem;
  transition: background-color 0.2s;
}

.zoek-card .card-link:hover {
  background-color: rgba(255, 255, 255, 0.45);
  color: #fff;
}

/* ---- Foutmelding popup ---- */

/* Donkere achtergrond over de hele pagina */
.popup-overlay {
  position: fixed;
  inset: 0; /* top/right/bottom/left = 0 */
  background: rgba(0, 0, 0, 0.45);
  z-index: 1000;
}

/* De witte popup zelf, gecentreerd op het scherm */
.popup-box {
  position: fixed;
  top: 50%;
  left: 50%;
  transform: translate(-50%, -50%);
  background: #fff;
  border-radius: 10px;
  padding: 24px 28px;
  z-index: 1001;
  min-width: 280px;
  max-width: 400px;
  box-shadow: 0 8px 24px rgba(0, 0, 0, 0.2);
}

.popup-header {
  display: flex;
  justify-content: space-between;
  align-items: center;
  border-bottom: 1px solid #eee;
  padding-bottom: 10px;
  margin-bottom: 10px;
}

/* Rode sluitknop rechtsboven in de popup */
.popup-sluit-btn {
  background: #dc3545;
}

```

```

border: none;
color: #fff;
font-size: 1.1rem;
font-weight: bold;
width: 28px;
height: 28px;
border-radius: 50%;
cursor: pointer;
line-height: 1;
display: flex;
align-items: center;
justify-content: center;
}

.popup-sluit-btn:hover {
background: #bb2d3b;
}

```

JS-code:

```

/* =====
My Internet Start Page - JavaScript

Aanpak: ik gebruik gewone functies (geen class).
Alle data bewaar ik in localStorage zodat de
history bewaard blijft na een refresh.
===== */

// =====
// CONSTANTEN
// =====

// Sleutels voor localStorage - zo vermijd ik typefouten later
const STORAGE_KEY_HISTORY = "startpage_history";
const STORAGE_KEY_SORTERING = "startpage_sortering";

// Alle ondersteunde zoekmachines met hun prefix, kleur en URL-logica
// Ik gebruik encodeURIComponent zodat speciale tekens correct gecodeerd worden
const ZOEKMACHINES = [
  {
    prefix: "g",
    naam: "Google",
    kleur: "#4285f4",
    maakUrl: (q) => `https://www.google.com/search?q=${encodeURIComponent(q)}`
  },
  {
    prefix: "y",
    naam: "Youtube",
    kleur: "#FF0000",
    maakUrl: (q) =>
`https://www.youtube.com/results?search_query=${encodeURIComponent(q)}`
  },
  {
    prefix: "x",
    naam: "X (Twitter)",
    kleur: "#1d9bf0",
    maakUrl: (q) => `https://x.com/search?q=${encodeURIComponent(q)}`
  },
  {
    prefix: "i",
    naam: "Instagram",
    kleur: "#e1306c",
    maakUrl: (q) => `https://www.instagram.com/explore/tags/${q.replace(/\s+/g,
"")}`
  },
  {

```

```

        prefix: "d",
        naam: "DuckDuckGo",
        kleur: "#de5833",
        maakUrl: (q) => `https://duckduckgo.com/?q=${encodeURIComponent(q)}`
    },
    {
        prefix: "t",
        naam: "TikTok",
        kleur: "#010101",
        maakUrl: (q) => `https://www.tiktok.com/search?q=${encodeURIComponent(q)}`
    }
];

// Sorteervolgorde: true = A→Z, false = Z→A
// We laden de vorige keuze uit localStorage, of starten standaard op A→Z
let sorteerOplopend = laadSortering();

// =====
// LOCALSTORAGE HULPFUNCTIES
// =====

// Bewaar een waarde (als JSON) in localStorage
function bewaarInStorage(sleutel, waarde) {
    localStorage.setItem(sleutel, JSON.stringify(waarde));
}

// Haal een waarde op uit localStorage en zet het terug naar een JS-object
// Geeft null terug als de sleutel niet bestaat
function laadUitStorage(sleutel) {
    const data = localStorage.getItem(sleutel);
    return data ? JSON.parse(data) : null;
}

// Laad de hele history-array uit localStorage
// Als er nog niets bewaard is, geef een lege array terug
function laadHistory() {
    return laadUitStorage(STORAGE_KEY_HISTORY) ?? [];
}

// Voeg één item toe aan de bestaande history en sla alles opnieuw op
function slaHistoryItemOp(item) {
    const history = laadHistory();
    history.push(item);
    bewaarInStorage(STORAGE_KEY_HISTORY, history);
}

// Laad de opgeslagen sorteervolgorde
// Als nog nooit opgeslagen, gebruik de standaard (A→Z = true)
function laadSortering() {
    const opgeslagen = laadUitStorage(STORAGE_KEY_SORTERING);
    return opgeslagen !== null ? opgeslagen : true;
}

// Bewaar de huidige sorteervolgorde
function bewaarSortering() {
    bewaarInStorage(STORAGE_KEY_SORTERING, sorteerOplopend);
}

// =====
// POPUP HULPFUNCTIES
// =====

// Toon een foutmelding in de popup
function toonFout(bericht) {
    document.getElementById("popup-bericht").textContent = bericht;
    document.getElementById("popup-overlay").classList.remove("d-none");
}

```

```

    document.getElementById("popup-box").classList.remove("d-none");
}

// Sluit en verberg de popup
function sluitPopup() {
    document.getElementById("popup-overlay").classList.add("d-none");
    document.getElementById("popup-box").classList.add("d-none");
}

// =====
// COMMANDO VERWERKEN
// =====

// Haal de zoekmachine op op basis van de prefix (bv. "g" → Google-object)
// Geeft null terug als de prefix onbekend is
function zoekMachineOpPrefix(prefix) {
    return ZOEKMACHINES.find(zm => zm.prefix === prefix) ?? null;
}

// Controleer het ingevoerde commando en geef de zoekmachine + zoekopdracht terug
// Gooit een fout (en toont popup) als het commando ongeldig is
function verwerkCommando(invoer) {
    // Commando moet starten met een slash
    if (!invoer.startsWith("/")) {
        toonFout('Ongeldig commando: start je commando met "[letter]
[zoekopdracht]");
        return null;
    }

    // De prefix is de letter direct na de slash (positie 1)
    const prefix = invoer[1]?.toLowerCase();

    // Alles na "/x " is de zoekopdracht (we slaan de eerste 3 tekens over: /x +
    spatie)
    const zoekopdracht = invoer.slice(3).trim();

    // Controleer of de prefix bekend is
    const zoekmachine = zoekMachineOpPrefix(prefix);
    if (!zoekmachine) {
        toonFout(`Onbekende prefix "${prefix}". Gebruik: /g, /y, /x, /i, /d of
/t`);
        return null;
    }

    // Controleer dat er ook een zoekopdracht meegegeven werd
    if (zoekopdracht.length === 0) {
        toonFout("Vul ook een zoekopdracht in na de prefix.");
        return null;
    }

    // Stel de volledige URL samen via de functie van de zoekmachine
    const url = zoekmachine.maakUrl(zoekopdracht);

    // Geef alles terug als één object
    return {
        naam: zoekmachine.naam,
        kleur: zoekmachine.kleur,
        query: zoekopdracht,
        url: url
    };
}

// =====
// CARDS WEERGEVEN
// =====

```

```

// Voeg één card toe aan de container in de HTML
// Elke card krijgt de Bootstrap col-klassen zodat het grid werkt
function voegCardToe(item) {
    const container = document.getElementById("cards-container");

    // Bootstrap kolom: 12 breed op xs, 6 op sm, 4 op md (= 3 per rij op desktop)
    const kolom = document.createElement("div");
    kolom.className = "col-12 col-sm-6 col-md-4";

    // De gekleurde card zelf
    const card = document.createElement("div");
    card.className = "zoek-card";
    card.style.backgroundColor = item.kleur;

    // Naam van de zoekmachine als titel
    const naam = document.createElement("div");
    naam.className = "card-naam";
    naam.textContent = item.naam;

    // De zoekopdracht als tekst
    const query = document.createElement("div");
    query.className = "card-query";
    query.textContent = item.query;

    // Link die de URL opent in een nieuw tabblad
    const link = document.createElement("a");
    link.className = "card-link";
    link.href = item.url;
    link.target = "_blank"; // Nieuw tabblad, niet nieuw venster
    link.rel = "noopener"; // Veiligheidsregel bij target="_blank"
    link.textContent = "Go!";

    // Alles samenvoegen
    card.appendChild(naam);
    card.appendChild(query);
    card.appendChild(link);
    kolom.appendChild(card);
    container.appendChild(kolom);
}

// Haal alle history op, sorteer ze, en toon ze opnieuw
// We leegmaken we eerst de container zodat we niet dubbel tonen
function herlaadCards() {
    const container = document.getElementById("cards-container");
    container.innerHTML = ""; // Leegmaken

    const history = laadHistory();

    // Sorteren op naam (A-Z), bij gelijkheid op zoekopdracht
    history.sort((a, b) => {
        const naamVergelijking = a.naam.localeCompare(b.naam);
        if (naamVergelijking !== 0) return naamVergelijking;
        return a.query.localeCompare(b.query);
    });

    // Omdraaien als Z→A gewenst is
    if (!sorteerOplopend) {
        history.reverse();
    }

    // Elke card toevoegen aan de pagina
    history.forEach(item => voegCardToe(item));
}

// =====
// SORTERING
// =====

```

```

// Wissel de sorteervolgorde om en pas de knoptekst aan
function wisselSortering() {
    sorteerOplopend = !sorteerOplopend;
    bewaarSortering();
    updateSorteerKnop();
    herlaadCards();
}

// Pas de tekst van de sorteerknop aan zodat het overeenkomt met de huidige
volgorde
function updateSorteerKnop() {
    const knop = document.getElementById("sort-btn");
    knop.textContent = sorteerOplopend ? "Sort: A → Z" : "Sort: Z → A";
}

// =====
// GO-KNOP LOGICA
// =====

// Alles wat er gebeurt als de gebruiker op GO! drukt (of Enter indrukt)
function verwerkGo() {
    const invoerVeld = document.getElementById("command-input");
    const invoer = invoerVeld.value.trim();

    // Commando verwerken - geeft null terug bij fouten
    const resultaat = verwerkCommando(invoer);
    if (!resultaat) return; // Fout al getoond via popup, stoppen

    // Open de URL in een nieuw tabblad
    window.open(resultaat.url, "_blank");

    // Bewaar in localStorage en herlaad de cards
    slaHistoryItemOp(resultaat);
    herlaadCards();

    // Invoerveld leegmaken
    invoerVeld.value = "";
}

// =====
// INITIALISATIE
// =====

// Wacht tot de volledige HTML geladen is voor we eventlisteners toevoegen
document.addEventListener("DOMContentLoaded", () => {

    // GO!-knop: klik-event
    document.getElementById("go-button").addEventListener("click", verwerkGo);

    // Invoerveld: Enter-toets werkt ook als GO!
    document.getElementById("command-input").addEventListener("keydown", (event) =>
    {
        if (event.key === "Enter") {
            verwerkGo();
        }
    });

    // Sorteerknop: wissel volgorde
    document.getElementById("sort-btn").addEventListener("click", wisselSortering);

    // Popup sluiten via de X-knop
    document.getElementById("popup-sluit").addEventListener("click", sluitPopup);

    // Popup sluiten door op de donkere achtergrond te klikken
    document.getElementById("popup-overlay").addEventListener("click", sluitPopup);
}

```

```
// Zet de juiste tekst op de sorteerknop bij opstarten
updateSorteerKnop();

// Laad de bestaande history uit localStorage en toon ze
herlaadCards();
});
```

Resultaat:

The screenshot shows a web application interface. At the top, the title "My Internet Start Page" is displayed. Below the title is a search section labeled "Zoekopdracht" with a text input field containing "/d zoekopdracht via duckduckgo". Below the input field is a green button labeled "GO!". Underneath is a "History" section with a "Sort: A - Z" button. The history section contains five cards, each with a platform name, a search query, and a "Go!" button:

- Google** (blue card): hogeschool vives
- Instagram** (pink card): vakantie fotos
- TikTok** (black card): kattenfilmpjes
- X (Twitter)** (light blue card): oude twitter post
- Youtube** (red card): toffe YT video