

# TutorBridge

Fortuna Eyob Fessehaye, fofe8356.

# 1. Sammanfattning

TutorBridge är en webbaserad tjänst där användare enkelt kan hitta en privatlärare eller student baserat på språk, ämne och personliga preferenser. Användaren fyller i ett formulär med mål, nivå, önskade tider och budget, och får sedan matchningshjälp med rätt tutor eller student. Tjänsten är byggd med HTML, CSS, JavaScript och jQuery, och fungerar smidigt på både dator och mobil. Plattformen innehåller även interaktiva element som sliders, valknappar och en stilren popup- fönster som bekräftar att ansökan skickats.

## 2. Framtagande

Detta projekt, **TutorBridge**, baserades på den frivilliga uppgiften 3.2.6 som handlade om formulärhantering. Under arbetets gång utökades projektet för att inkludera fler funktioner såsom localStorage, navigering mellan sidor och ett tydligt användarflöde. Nedan beskrivs utvecklingsprocessen i fem steg, från idé till färdig produkt, med exempel på hur koden utvecklades.

### 2.1 Idé och planering

Projektets mål var att skapa en webbtjänst där användare kan matchas med rätt privatlärare och vice versa. Både elever och lärare skulle kunna fylla i formulär för att uttrycka sina behov eller erbjudanden. Det första steget var att planera användarflödet. Följande struktur skissades:

- Startside med språkval eller möjlighet att navigera vidare.
- En sida för att välja vilket ämne man vill ha hjälp med.
- Ett formulär där användaren fyller i mål, nivå, tillgänglighet och budget.
- Ett separat formulär för lärare att ange kompetens, pris och tider.
- En bekräftelse i form av ett popup-fönster som informerar användaren om att vi kommer att kontakta dem via e-post.

### 2.2 Struktur och HTML-grund

När jag hade en klar idé om hur webbtjänsten skulle se ut började jag med att skapa fyra huvudsidor: index.html, tutors.html, becometutor.html och forum.html. Formulären byggdes med semantiska HTML-element såsom <fieldset>, <label>, <input>, <select>, <radio> och <range>. Fokus låg på att strukturera sidorna logiskt så att användaren förstår vart hen är i processen.

### 2.3 Interaktivitet med JavaScript och jQuery

Till en början användes JavaScript (DOM API) för att skapa interaktivitet, men för att förenkla kodhantering byttes delar ut mot jQuery. Några av de viktigaste funktionerna som implementerades var:

- Slider med färgad bakgrund för att välja budgetintervall:

```
function slideOne() {  
  if (parseInt(sliderTwo.value) - parseInt(sliderOne.value) <= minGap) {  
    sliderOne.value = parseInt(sliderTwo.value) - minGap;  
  }  
  displayValOne.textContent = sliderOne.value;  
  fillSliderTrack();  
}
```

```
function fillSliderTrack() {
  let percent1 = ((sliderOne.value - sliderOne.min) / (sliderMaxValue - sliderOne.min)) * 100;
  let percent2 = ((sliderTwo.value - sliderOne.min) / (sliderMaxValue - sliderOne.min)) * 100;
  sliderTrack.style.background = `linear-gradient(to right
    , #dadae5 ${percent1}%
    , #3264fe ${percent1}%
    , #007bff ${percent2}%
    , #dadae5 ${percent2}%` ;
}
```

- Toggle-funktion för att välja flera dagar och tider:

```
function initToggleButtons() {
  $('.select_buttons_days button, .select_buttons_times button, .select_buttons_lang button').on('click', function (e) {
    e.preventDefault();
    $(this).toggleClass('selected');
  });
}
```

- Navigering mellan sidor:

```
// Become a tutor button click
if (becomeTutorbutton) {
  becomeTutorbutton.addEventListener("click", () => {
    window.location.href = "becometutor.html";
  });
}
```

- Användning av localStorage för att lagra användarens språkval och ämne:

```
localStorage.setItem("selectedTutorLanguage", language);
```

## 2.4 Förbättringar och användarfeedback

När grundfunktionaliteten var klar fokuserade jag på förbättringar för att förbättra användarupplevelsen:

- Ett popup-fönster skapades som visas när formuläret skickats in. Det visar en personlig bekräftelse med användarens val:

```
<div id="thankYouPopup" class="popup">
  <div class="popup-content">
    <span class="close">&times;</span>
    <h2>Thank you!</h2>
    <p>
      We will contact you via email with students who are searching for
      <span id="selected-subject-text"></span> lessons in
      <span id="selected-lang-text"></span>. You'll be notified regularly as new matches become available.
    </p>
  </div>
</div>
```

- Formulärvalidering lades till för att kontrollera att användaren valt mål, nivå, minst en dag och minst en tid.
- Kodstädning: funktioner som slideOne() och slideTwo() flyttades ut från \$(document).ready() för att göra koden mer läsbar.

## 2.5 Testning och responsiv design

Projektet testades i olika skärmstorlekar och webbläsare. För att säkerställa att layouten fungerade på olika skärmar användes media queries i CSS:

```
@media (min-width: 481px) and (max-width: 768px){  
  
  header{  
    max-height: fit-content;  
  }  
  
  .intro_section {  
    display: flex;  
    flex-direction: column-reverse;  
    align-items: flex-start; /* Align everything to the left */  
    padding-top: 10px;  
    padding-left: 10px; /* Small left padding for spacing */  
  }  
}
```

Formulärens layout justerades för att undvika överlappningar, och popup-fönstret centrerades korrekt på mindre skärmar. Knappar, texttrutor och val blev också klickvänliga även på mobil.

## 3. Form

I TutorBridge-projektet spelar formulären en central roll för att samla in information från både studenter och handledare. Det finns totalt fyra HTML-sidor: index.html, tutors.html, forum.html och becometutor.html. Nedan beskriver jag hur varje sida fungerar och hur koden är strukturerad, samt inkluderar enkel pseudokod för att visa flödet.

### 3.1 Kodstruktur och Pseudokod

**Index.html:** Denna är startsidan och kompletteras av JavaScript-filen index.js. Den visar menu knappar: Teach, About och Search. Användaren kan välja att börja hitta en privatlärare, bli privatlärare själv, välja språk direkt från listan eller läsa mer om tjänsten.

#### Central Kodavsnitt:

```
<nav>
  <ul class="menu">
    <li>
      <a href="index.html" class="logo">
        
      </a>
    </li>
    <li class="menu_item">
      <a href="#become_a_tutor_section">Teach</a>
    </li>
    <li class="menu_item"><a href="#about_section">About</a></li>
    <li class="menu_item"><a href="#find_tutor">Search</a></li>
  </ul>
</nav>
<div>
  <div class="intro_section">
    <div class="intro_text">
      <h1>Learn Coding with your best language tutor.</h1>
      <button class="get-started">Get Started</button>
    </div>
```

```
document.addEventListener("DOMContentLoaded", () => {
  const getStartedbutton = document.querySelector(".get-started");
  const becomeTutorbutton = document.querySelector(".become_tutor_button");
  const findTutorBtn = document.querySelectorAll(".tutor-list-item a");

  // Get Started button click
  if (getStartedbutton) {
    getStartedbutton.addEventListener("click", () => {
      window.location.href = "tutors.html";
    });
  }

  // Become a tutor button click
  if (becomeTutorbutton) {
    becomeTutorbutton.addEventListener("click", () => {
      window.location.href = "becometutor.html";
    });
  }

  // Find Tutor link clicks
  findTutorBtn.forEach(link => {
    link.addEventListener("click", (e) => {
      e.preventDefault();
      const language = link.textContent.trim().split(" ")[0];
      localStorage.setItem("selectedTutorLanguage", language);
      window.location.href = "tutors.html";
    });
  });
});
```

#### Pseudokod

##### Start

When "Get Started" button in header is clicked  
Navigate to "tutors.html"

When "Teach" button in the navigation bar is clicked  
Scroll to the "Become a Tutor" section on the same page  
If "Become a Tutor" button in the section is clicked  
Navigate to "becometutor.html"

When "About" button is clicked  
Scroll to the "About" section on the same page

When "Search" button is clicked  
Scroll to the language list section  
If a language option is clicked  
Store the selected language in localStorage  
Navigate to "tutors.html"

**tutors.html** är sidan där användaren väljer vilket ämne de vill lära sig och vilket språk de vill ha undervisning på. Informationen som användaren väljer här sparas i localStorage så att den kan användas senare i formuläret på forum.html. Om användaren redan valt ett språk tidigare (från index.html), visas detta direkt och dropdown-menyn göms. Annars får användaren välja språk i dropdown-listan.

### Centralt Kodavsnitt:

```
if (selectedLang) {
  // Show selected language in title
  if (tutorLangDisplay) {
    tutorLangDisplay.textContent = selectedLang;
  }
  if (tutorChoiceDisplay) {
    tutorChoiceDisplay.style.display = "none";
  }
} else {
  // Show dropdown if no language is selected
  if (tutorLangDisplay) tutorLangDisplay.textContent = "";
  if (tutorChoiceDisplay) tutorChoiceDisplay.style.display = "";
}

// Get Started button behavior
if (getStartedBtn) {
  getStartedBtn.addEventListener("click", () => {
    const selectedSubject = subjectDropdown.value.trim();
    //save selected lang and subject
    if(selectedLang){
      localStorage.setItem("preferredLang", selectedLang);
      localStorage.removeItem("selectedTutorLanguage");
    } else{
      const selectedLanguage = tutorLangChoice.value.trim();
      localStorage.setItem("preferredLang", selectedLanguage);
    }
    localStorage.setItem("selectedSubject", selectedSubject);
    window.location.href = "forum.html";
  });
}
```

### Pseudokod

If selected language is in localStorage then

- Display the selected language in a heading

- Hide the language dropdown menu

else

- Show the dropdown so user can select a language

When "Get Started" button is clicked

- Get the selected subject from the dropdown

- If selected language already exists in localStorage then

  - Remove selectedTutorLanguage from localStorage

  - Save the new selected language as preferredLang

- else

  - Save the newly chosen language from dropdown as preferredLang

- Save the selected subject as selectedSubject

- Navigate to "forum.html"

**forum.html:** Det är en form för att hitta en tutor genom att fylla i mål, kunskapsnivå, önskade dagar och tider, språkpreferenser och budget.

### Centralt Kodavsnitt:

```

function handleFormSubmit() {
  $('#form').on('submit', function (e) {
    const goalSelected = $('input[name="goal"]:checked').length > 0;
    const levelSelected = $('input[name="level"]:checked').length > 0;
    const daysSelected = $('.select_buttons_days button.selected').length > 0;
    const timesSelected = $('.select_buttons_times button.selected').length > 0;

    if (!goalSelected || !levelSelected || !daysSelected || !timesSelected) {
      alert('Please fill all required selections (goal, level, days, and times).');
      e.preventDefault();
      return;
    }

    e.preventDefault(); // Prevent actual submission

    const lang = localStorage.getItem("preferredLang") || "your selected language";
    const subject = localStorage.getItem("selectedSubject") || "your chosen subject";

    $('#selected-lang-text').text(lang);
    $('#selected-subject-text').text(subject);
    $('#thankYouPopup').fadeIn();
  });
}

```

## Pseudokod

Activate toggle behavior for language, day and time buttons  
 Initialize two sliders to set a min and max budget

When user drags slider-1 or slider-2

- Update displayed values

- Update the background color between the two handles

When form is submitted

- Check if at least one radio button is selected in goal and level sections

- Check if at least one button is selected in days and times

- If any required field is missing, then

- Show alert and stop submission

- If form is valid then

- Prevent default submit behaviour

- Read saved language and subject from localStorage

- Show modal popup with personalized message

When modal close button is clicked

- Hide the modal popup

- Reset the form

- Clear selected buttons

- Clear saved values from localStorage

- Redirect back to index.html

**becometutor.html:** en registreringssida för privatlärare som vill bli en del av TutorBridge-plattformen. Den innehåller ett detaljerat formulär där läraren fyller i information som undervisning ämne, språkkunskaper, prisintervall via två sliders, önskade dagar och tider, och möjlighet att änge certificate och utbildningsbakgrund. Formuläret hanteras av becometutor.js som styr alla interaktiva funktioner: sliders, språkval, knappar för dagar/tider, kryssrutor och en bekräftelse-popup.



## Centralt Kodavsnitt:

```
// On Page Load
document.addEventListener("DOMContentLoaded", () => {

  // Initialize global variables after DOM is ready
  sliderOne = document.getElementById("slider-1");
  sliderTwo = document.getElementById("slider-2");
  displayValOne = document.getElementById("range1");
  displayValTwo = document.getElementById("range2");
  sliderTrack = document.querySelector(".slider-track");
  sliderMaxValue = sliderOne.max;

  initToggleButtons();
  initSliders();
  handleFormSubmit();
  handleModalClose();
  initCertificateToggle();
  initEducationToggle();
});
```

```
//Form Submission & Modal Logic
function handleFormSubmit() {
  const form = document.getElementById("form");
  form.addEventListener("submit", (e) => {
    e.preventDefault();

    const daysSelected = document.querySelectorAll(".select_buttons_days button.selected").length > 0;
    const timesSelected = document.querySelectorAll(".select_buttons_times button.selected").length > 0;

    if (!daysSelected || !timesSelected) {
      alert("Please fill all required selections (days and times).");
      return;
    }

    const selectedSubject = document.getElementById("select-option").value;
    const nativeLang = document.getElementById("language").value;

    const selectedExtraLangs = [];
    const selectedButtons = document.querySelectorAll(".select_buttons_lang button.selected");

    for (let i = 0; i < selectedButtons.length; i++) {
      selectedExtraLangs.push(selectedButtons[i].textContent);
    }

    let fullLanguageText = nativeLang;
    if (selectedExtraLangs.length > 0) {
      fullLanguageText += ", " + selectedExtraLangs.join(", ");
    }

    document.getElementById("selected-subject-text").textContent = selectedSubject;
    document.getElementById("selected-lang-text").textContent = fullLanguageText;

    document.getElementById("thankYouPopup").style.display = "block";
  });
}
```

## Pseudokod

On page load:

- Call `initToggleButtons()` to enable selection for language, day, and time buttons
  - When any language/time/day button is clicked
    - Toggle 'selected' class

- Call `initSliders()` to initialize min and max budget sliders
  - When either slider is moved:
    - Update displayed budget values
    - Update background color of slider track

- Call `handleFormSubmit()` to validate the form and display the modal on submission
  - When form is submitted:
    - Check if user has selected at least one day and one time
    - If user has not selected, then
      - Show alert and stop submission
    - else
      - Get selected subject from dropdown
      - Get native language from input field
      - Collect additional selected languages
      - Display summary in the modal popup

- Call `handleModalClose()` to reset form and redirect when popup closes
  - When modal close button is clicked
    - Hide modal
    - Reset form and clear selected buttons
    - Remove any saved localStorage values
    - Redirect back to index.html

- Call `initCertificateToggle()` to handle checkbox logic for certificate field
  - When checkbox1 (No certificate) is checked

- Disable certificate input field and subject dropdown
- else
  - Enable both fields

Call `initEducationToggle()` to handle checkbox logic for education fields

- When checkbox2 (No education) is checked

- Disable university, degree and specialization fields

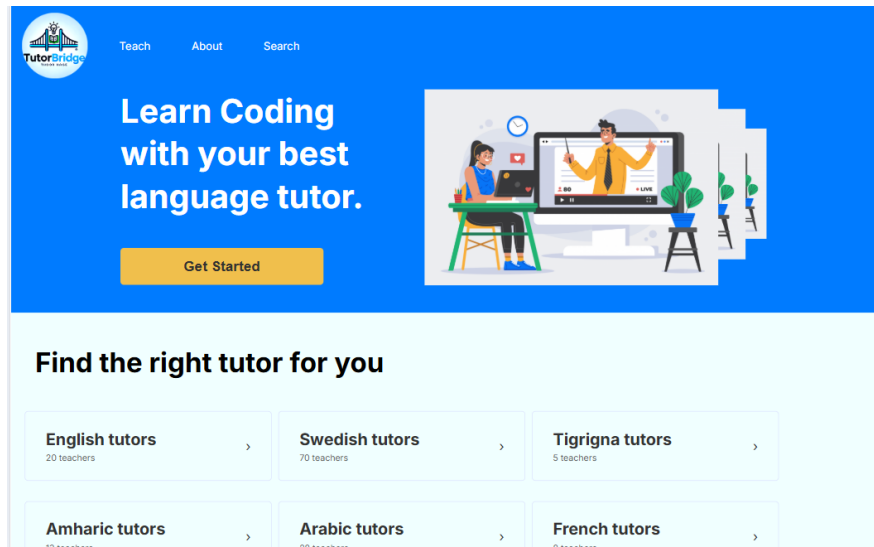
- Remove requirement (set required to false) for those fields

- else

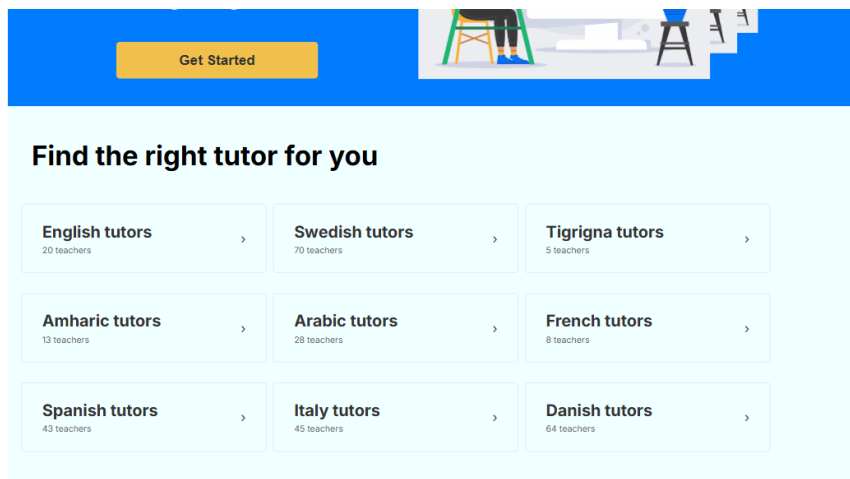
- Enable and require them

## 4. Funktion

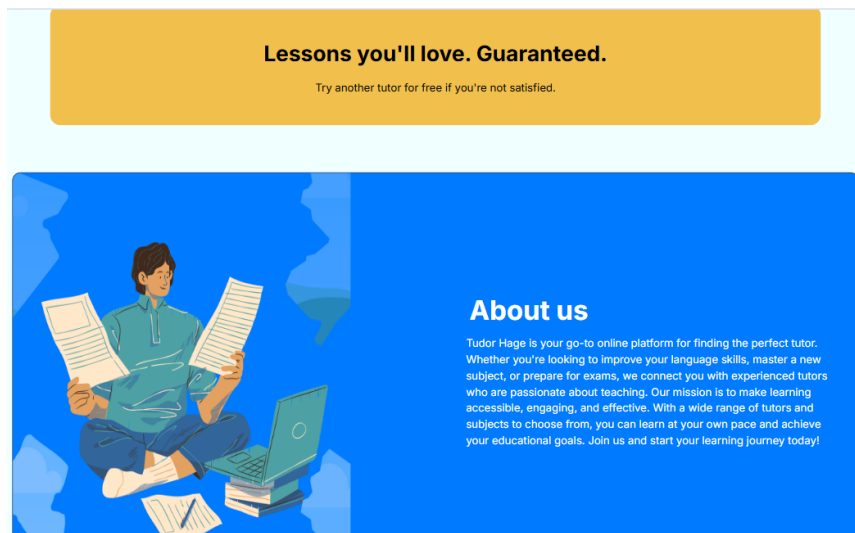
När användaren öppnar webbplatsen laddas startsidan automatiskt (index.html). Den består av tre menyknappar: **Search**, **About** och **Teach**, samt en knapp för att komma igång, **Get Started**.



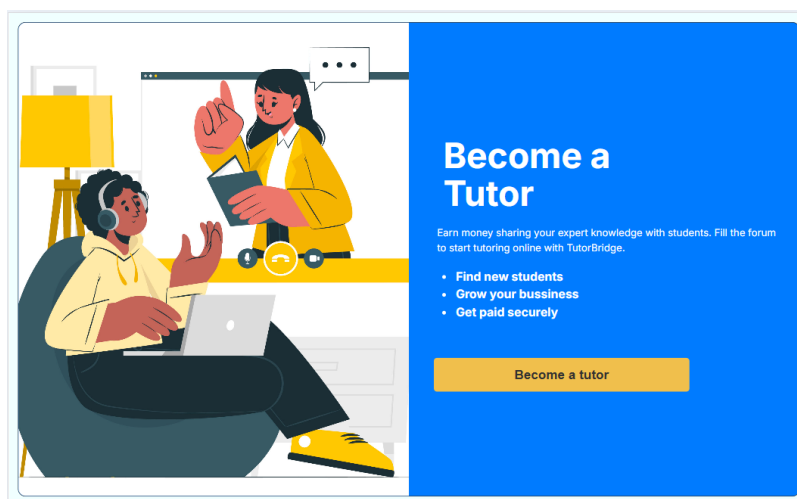
När användaren klickar på **Search** scollas sidan ner till språklistan där språk som "English", "Swedish" och "Arabic" visas som klickbara knappar. När användaren klickar på ett språk sparas valet i localStorage och programmet navigerar automatiskt vidare till **tutors.html**.



När användaren klickar på **About** scollar till en sektion med kort information om tjänsten.

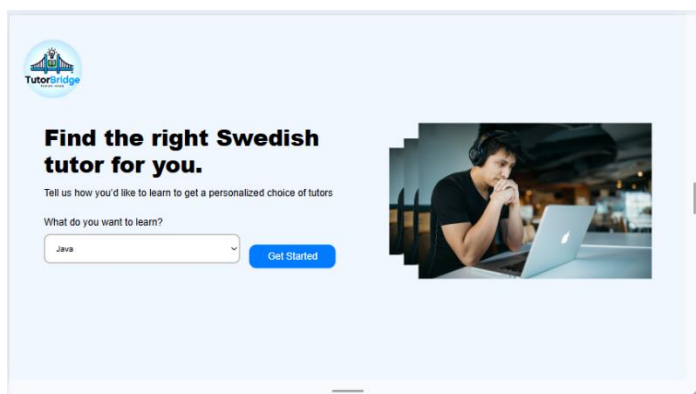


När användaren klickar på **Teach** scrollar till sektionen "Become a Tutor" på samma sida, där användaren sedan kan klicka på en ny knapp som leder till **becometutor.html**.

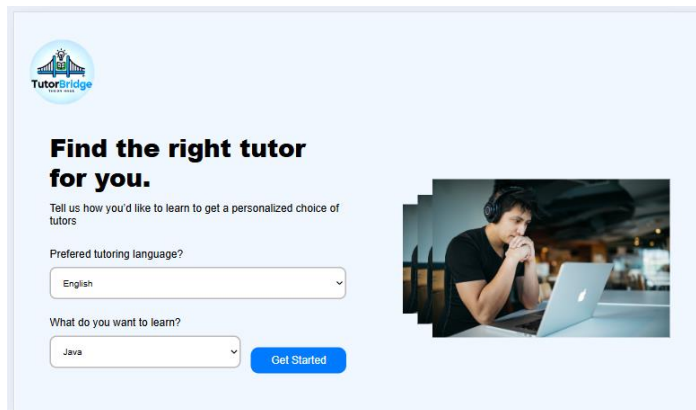


## 4.1 Ämnesval (tutors.html)

Om användaren redan har valt ett språk tidigare så visas det automatiskt som en rubrik högst upp.

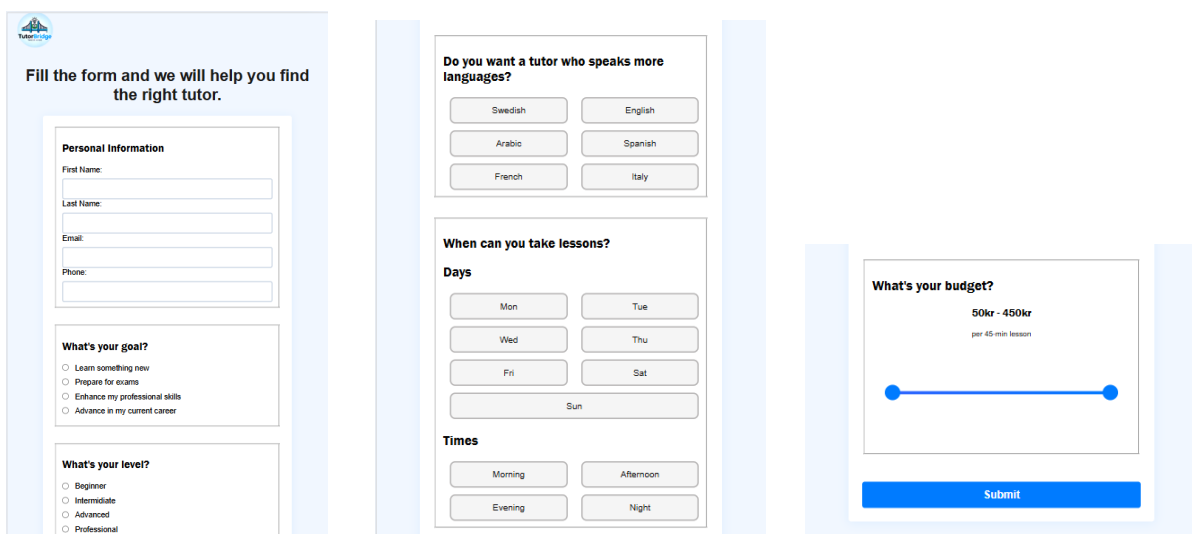


Har användaren inte valt något innan, så får den i stället välja språk själv från en dropdown. Efter det väljer användaren också vilket ämne man vill ha hjälp med. När användaren klickar på 'Get Started' sparas både språket och ämnet i localStorage, och man skickas vidare till forum.html – där själva formuläret finns för att hitta en privatlärare. Detta steg säkerställer att användaren specificerar sitt behov på ett tydligt sätt innan formuläret visas.



## 4.2 Formulär för att hitta tutor (forum.html)

På **forum.html** fyller användaren i ett formulär med personlig information, mål, nivå, dagar, tider och budget. Knapparna markeras med "selected" när de klickas, och sliders visar prisintervallet visuellt. Vid **Submit** kontrolleras att alla val är ifyllda – annars visas ett alert().



Om allt stämmer, visas en popup-fönster med språk och ämne. När popupsen stängs återställs formuläret och användaren skickas tillbaka till startsidan. Allt hanteras med jQuery och DOM.

What's your budget?

×

Thank you!

We will contact you via email once we find your best match tutor for Java in English.

Submit

## 4.2 Formulär för att bli tutor (becometutor.html)

På **becometutor.html** finns ett mer omfattande formulär för den som vill bli tutor. Användaren skriver in sitt modersmål och kan lägga till fler språk genom att klicka på knappar. Två kryssrutor låter användaren ange om de saknar certifikat eller utbildning. Om de kryssas i så inaktiveras fälten automatiskt och görs valfria.

tutorango

Start creating your public tutor profile and we will help you find students worldwide.

About

First Name:

Last Name:

Email:

Phone:

Subject you teach

Java

Language you speak

English

Level

Native

What other languages do you speak?

Swedish

English

Arabic

Spanish

French

Italy

Danish

Tigrina

Teaching certification

Do you have teaching certificates? If so, describe them to enhance your profile credibility and get more students.

☐

I don't have a teaching certificate

Subject

English

Fill verified certificate

Education

Tell students more about the higher education that you've completed or are working on

☐

I don't have a higher education degree

University

E.g. Stockholm University

Degree

E.g. Bachelor's degree in Computer Science

Degree type

Teaching degree

Specialization

E.g. Teaching in Java

Introduce yourself

Användaren får också möjlighet att introducera sig själv och välja tillgängliga dagar och tider med klickbara knappar, samt prisintervallet styrs med två sliders.

### Introduce yourself

Show potential students who you are! Share your teaching experience and passion for education and briefly mention your interests and hobbies

Hello, my name is.... and I'm from...

### When can you give lessons?

#### Days

MonTueWedThuFriSatSun

#### Times

MorningAfternoonEveningNight

### What's your price range?

50kr - 450kr  
per 45-min lesson

Create profile

När formuläret skickas in genom att klicka på "Create profile" kontrolleras att minst en dag och tid har valts. Därefter visas en popup-fönster med användarens val, och när popupen stängs återställs formuläret, localStorage rensas och användaren skickas tillbaka till startsidan. Formuläret är byggt för att vara flexibelt och passa olika typer av tutorprofiler.

Thank you!

We will contact you via email with students who are searching for Java lessons in English, Swedish, English. You'll be notified regularly as new matches become available.

### When can you give lessons?

#### Days

#### Times

MorningAfternoonEveningNight

### What's your price range?

50kr - 450kr  
per 45-min lesson

Create profile