

# Seminar 1

*Justas Mundeikis*

*2019-03-06*

## Contents

<b>1</b>	<b>Reikiamų programų instaliavimas</b>	<b>1</b>
<b>2</b>	<b>Paskyrų susikūrimas</b>	<b>1</b>
<b>3</b>	<b>Nuosavo tinklapio sukūrimas</b>	<b>1</b>
3.1	Seminaro darbo tikslas . . . . .	1
3.2	Žingsniai . . . . .	2
3.3	Naujos atšakos sukūrimas (paralelinis darbas) ir about.html sukūrimas . . . . .	11
3.4	Tinklapio paleidimas online: . . . . .	12
3.5	Tinklapio url į google sheets . . . . .	13

## 1 Reikiamų programų instaliavimas

Instaliuokite šias programas savo asmeniniame kompiuteryje..

- Instaliuokite R
- Instaliuokite RStudio
- Instaliuokite GitBash
- Instaliuokite Sublime
- Nepamirškite pagrindinių Git nustatymų

```
$ git config --global user.name "Vardas Pavardė"
$ git config --global user.email vardas.pavardė@evaf.vu.lt #arba kitas email, kurio registravotės @GitHub
$ git config --global core.pager cat
$ git config --global core.askPass ""
$ git config --global core.editor nano.exe
# here the absolute link to the execution file might be different on every computer!
$ git config --global core.editor /c/Users/USER/Download/sublime/sublime_text.exe
```

## 2 Paskyrų susikūrimas

Susikurkite savo paskyrą, naudodami bet kokį pseudonimą @Github, tačiau naudodami savo universitetinį email vardas.pavarde@stud.vu.lt. Tokiu atveju, jeigu nepavyktų aktyvuoti savo studentiško email, naudokite asmeninį email.

## 3 Nuosavo tinklapio sukūrimas

### 3.1 Seminaro darbo tikslas

Šio seminaro tikslas naudojantis *Git*, *Sublime* ir *Github* sukurti savo nuosavą internetinį puslapį. Pastaba: Tarkime, kad mano studento nr yra “S175”, Jūsų gali būti kitoks, tada naudojate savo studento nr vietoj mano “S175” Git komandose. Taip darome todėl, jog atliekant šias užduotis nuniversitete, nesimaišytų folderiai.

## 3.2 Žingsniai

Tinklapis bus kuriamas kiekvieno studento asmeniniame folderyje S1xxxxx, jame sukuriant subfolderį “webpage”.

```
# atspirties taškas VU PC /c/Users/studentas, todėl
cd Desktop
mkdir -p S175/webpage
cd S175/webpage
```

Siekiant, jog Git pradėtų sekti veiklą folderyje, būtina pirma inicijuoti Git

```
git init
```

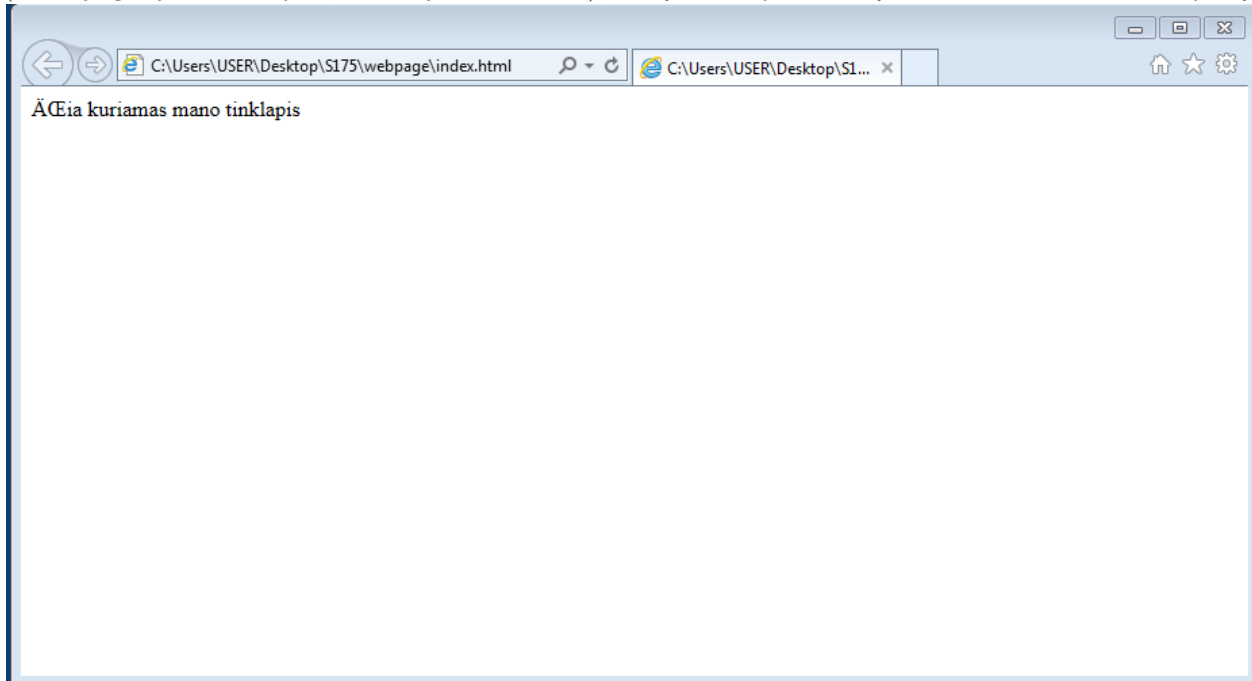
Kiekvieno tinklapio bazinis dokumentas yra “index.html”, kurį galime susikurti CLI pagalba

```
touch index.html
```

*Sublime* editorius pranašus tuo, jog pažįsta aibę skirtingų sintaksių. Koduojant šiuo atveju html, patartina nustatyti: View -> Syntax -> html Tada editoriuje rašome

Čia kuriamas mano tinklapis

išsaugome editoriaus pakeitimus su CTRL+S ir pabandome atidaryti sukurtą index.html failą (arba jeigu jau atidarytas, atnaujinant su F5), kurį atidaryti turėtų viena iš interneto naršyklių.

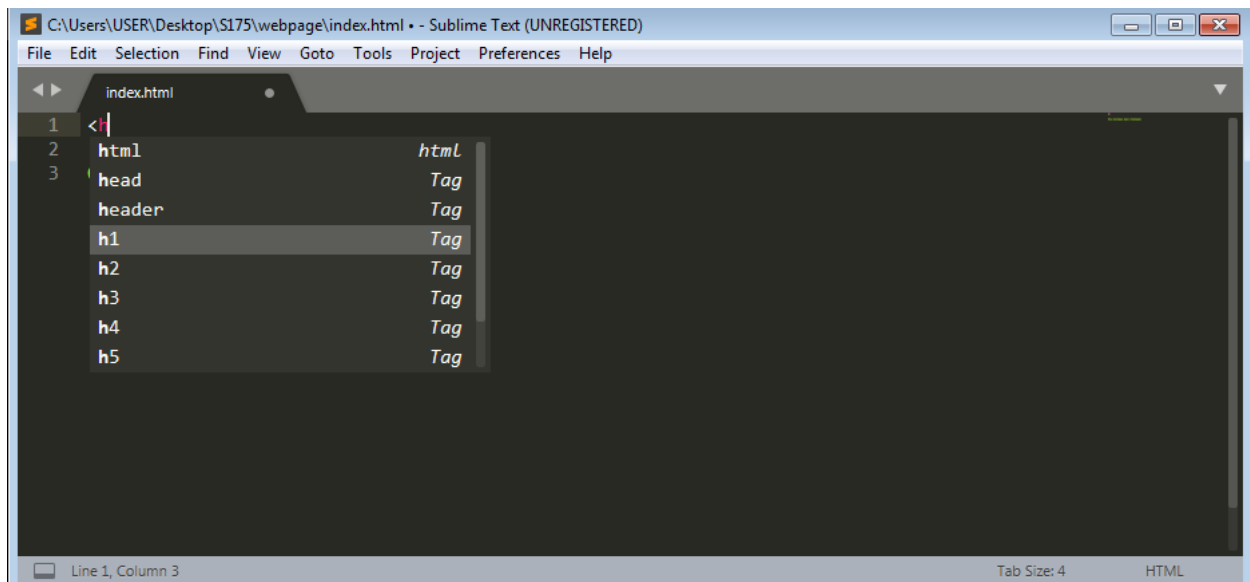


Tam kad dokumentas turėtų antraštę, 1 pirmą eilutę parašome: <h1> S175 tinklapis </h1>

```
<h1> S175 tinklapis </h1>
```

Čia kuriamas mano tinklapis

Tiesa, *Sublime* yra pakankamai protingas ir pradėjus rašyti <h išmeta pasirinkimą, pasirinkus su TAB / pele h1 pats supildo viską, bereikia rašyti tik savo tekstą.



Išsaugokite pakeitimus editoriuje ir atnaujinkite naršyklę.

Pasiekėme stadiją, kai norime neprarasti įdirbio, todėl padarome folderio “nuotrauką” su Git

```
git add .
git commit -m "Sukuriamas index.html failas"
```

Dabar prikurkime šiek tiek teksto mūsų tinklapiui pvz., “Šio tinklapio kūrimas yra 1.Seminaro darbas” ir išsaugome editoriaus pakeitimus

```
<h1>S175 tinklapis</h1>
```

Čia kuriamas mano tinklapis

Šio tinklapio kūrimas yra 1.Seminaro darbas

Su komanda `git status` patikriname, ar Git seka mūsų dokumentą. Git output turėtų būti daugmaž toks:

```
USER@PC MINGW64 ~/Desktop/s175/webpage (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   index.html
```

Taigi turime modifikuotą dokumentą, kurio pakeitimai dar nėra staged. Su komanda `git diff` galime patikrinti, o kokius skirtumus Git atpažįsta

Kadangi pakeitimai teisingi, galima `git commit -am "Sukurta index.html antraštė"`. Kadangi jau track-inam visus failus tai galima naudoti sutrumpintą komandą, kur `-am` reiškia `-all` ir `-m`

Na bet į internetinį puslapį toli gražu tai dar nepanašų... Todėl į editorių perrašome šį bloką (arba pradedame typinti

```
<title></title>
```

S175 Tinklapis

Čia kuriamas mano tinklapis

Šio tinklapio kūrimas yra 1.Seminaro darbas

““

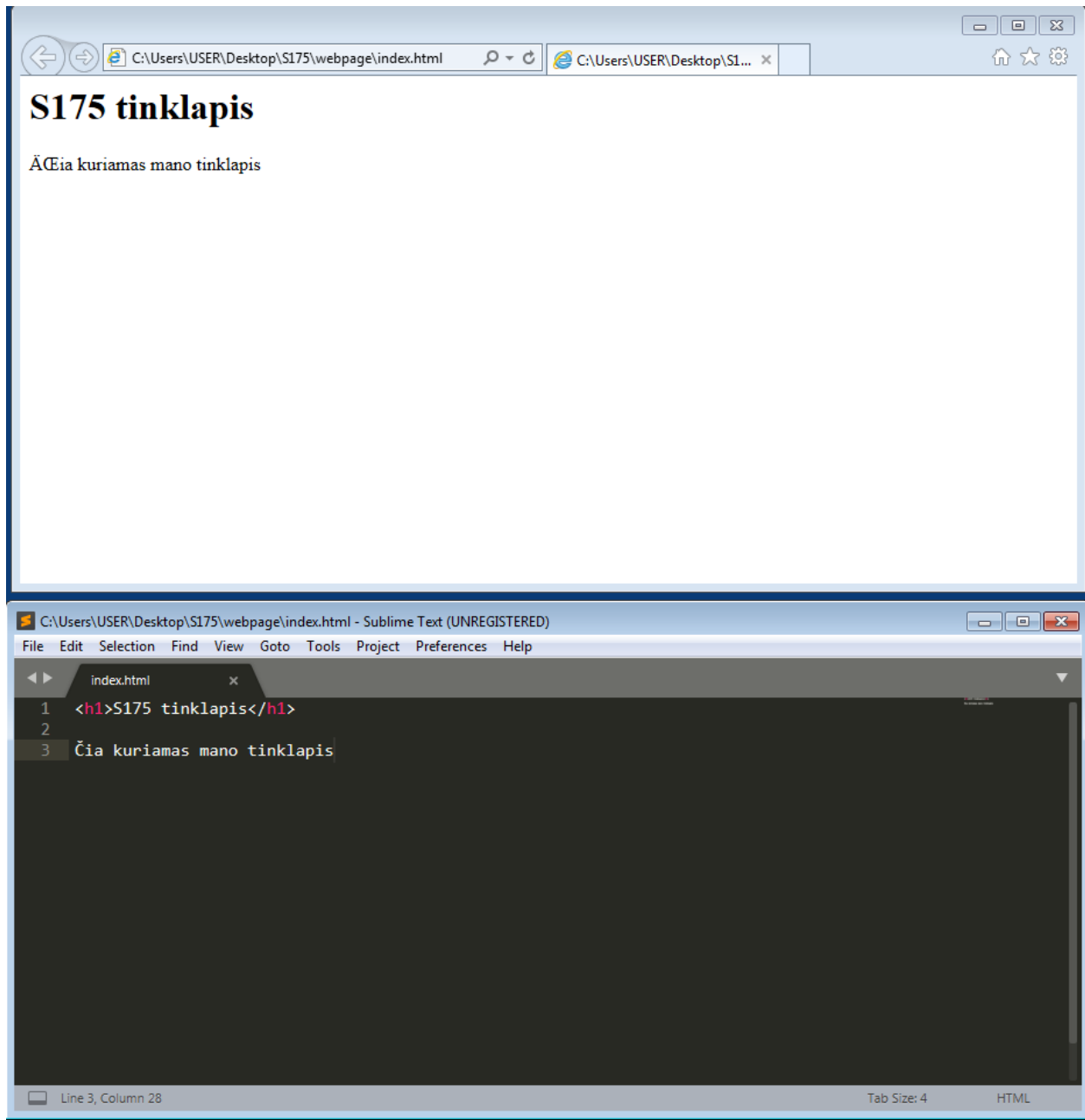


Figure 1: Caption for the picture.

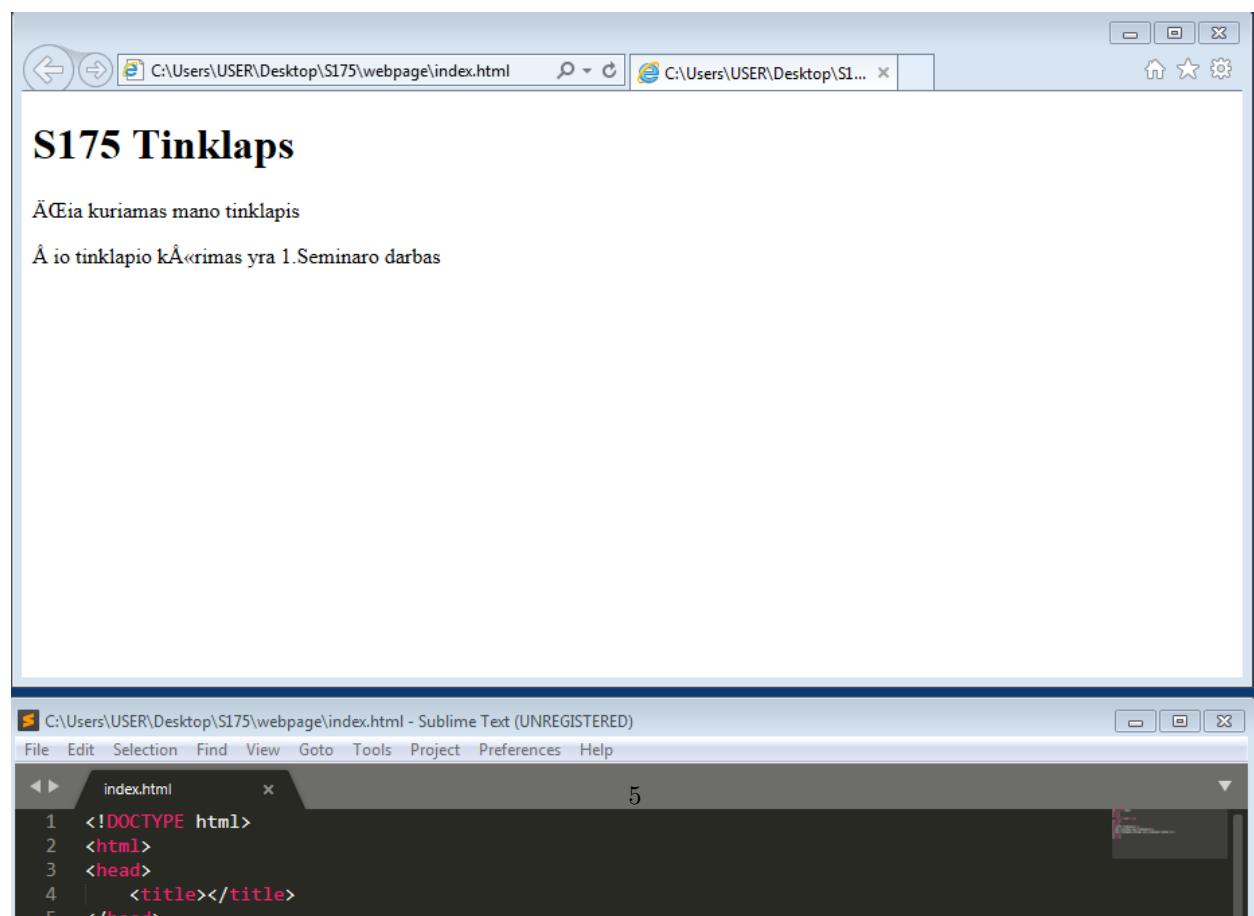
```
USER@PC MINGW64 ~/Desktop/s175/webpage (master)
$ git diff
diff --git a/index.html b/index.html
index 4d8137a..47c1b4b 100644
--- a/index.html
+++ b/index.html
@@ -1,3 +1,5 @@
<h1>S175 tinklapis</h1>

-Čia kuriamas mano tinklapis
\ No newline at end of file
+Čia kuriamas mano tinklapis
+
+Šio tinklapio kūrimas yra 1.Seminaro darbas
\ No newline at end of file

USER@PC MINGW64 ~/Desktop/s175/webpage (master)
$
```

Figure 2: Caption for the picture.

html kodas	kodo reikšmė
<!DOCTYPE html>	nusako koks tai dokumento tipas, šiuo atveju tai html dokumentas
<html>	atidaro html tag
<head>	atidaro head tag
<title>``</title>	atidaro ir uždaro title tag
</head>	uždaro head tag
<body>	atidaro body tag (t.y. puslapio turinys)
<h1></h1>	atidaro ir uždaro antraštės tipo tag
<p></p>	atidaro ir uždaro paragrafo tag
</body>	uždaro body tag
</html>	uždaro html tag



Jeigu norime, kad pats puslapis, turėtų naršyklei suprantamą pavadinimą, pvz., “Home” papildome kodą:

```
<!DOCTYPE html>
<html>
<head>
  <title>Home</title>
</head>
<body>
<h1>S175 Tinklapis</h1>
<p>Čia kuriamas mano tinklapis</p>
<p>Šio tinklapio kūrimas yra 1.Seminaro darbas </p>
</body>
</html>
```

Dar vienas nemalonus dalykas, naršyklė nesupranta lietuviškų simbolių, todėl įterpiame po pirmos eilutės: `<meta charset="utf-8"/>`, taigi priskiriame meta informaciją puslapiui ir pasakome, jog naršyklė skaitydama šį tekstą naudotų utf-8 kodavimą, kuris turi ir lietuviškus simbolius

```
<!DOCTYPE html>
<meta charset="utf-8"/>
<html>
<head>
  <title>Home</title>
</head>
<body>
<h1>S175 Tinklapis</h1>
<p>Čia kuriamas mano tinklapis</p>
<p>Šio tinklapio kūrimas yra 1.Seminaro darbas </p>
</body>
</html>
```

Dabartinis tinklapis jau daug maž panašus į tinklapį.

Išsaugome editoriaus pakeitimus ir commitinam pakeitimus `git commit -am "papildytas index.html"`.

Asmeninėje *Github* paskyroje sukuriame naują repo pavadinimu “webpage”. Praleidžiame aprašymą ir paliekame **be README**. Kodėl nesukuriame README.md? Nes lokali repo jau turi savo “istoriją” ir jeigu sukūrę nuotolinę repo inicijuosime ją su README.md, tada ir nuotolinė repo turės savo, bet jau kitokią istoriją. O tai reikš, jog siunčiant lokalią repo į nuotolinę, kils konfliktas dėl skirtingų Git istorijų.

```
# susikūrus repo webpage, Github parodys instrukciją, kaip susieti lokalią ir nuotolinę repo
# arba atitinkamai:
git remote add origin https://github.com/vartotojovardas/webpage.git

# susiejus su Github (pasitikrinti galima visada su "git remote -v")
git push -u origin master

# -u reiškia, jog vietinė repo išmoksta, kad Github susieta repo yra upstream repository, o tai leis ateityje tiesiog parsisiųsti
```

Ataujiname Github tinklapį ir pamatysim

Kadangi nėra blogai turėti README failą, tai Git Bash sukuriame jį.

```
touch README.md
```

Atsidarome failą su *Sublime* ir perrašome šį Markdown formatuotą tekstą:

```
# Bandomasis puslapis      Čia kuriamas bandomasis puslapis. Ši paskyra priklauso studentui
S....      Šis puslapis kurimas Vilniaus Universiteto Ekonomikos ir verslo administravimo
fakultete vykstančios "Duomenų analizės įvado" paskaitų metu. [Čia galima rasti daugiau
Markdown formatavimo subtilybių: Markdown cheatsheet ]
git add .
git commit -m "Sukurtas README.md failas"
git push origin master
```

Atnaujinus GitHub puslapį, matome:

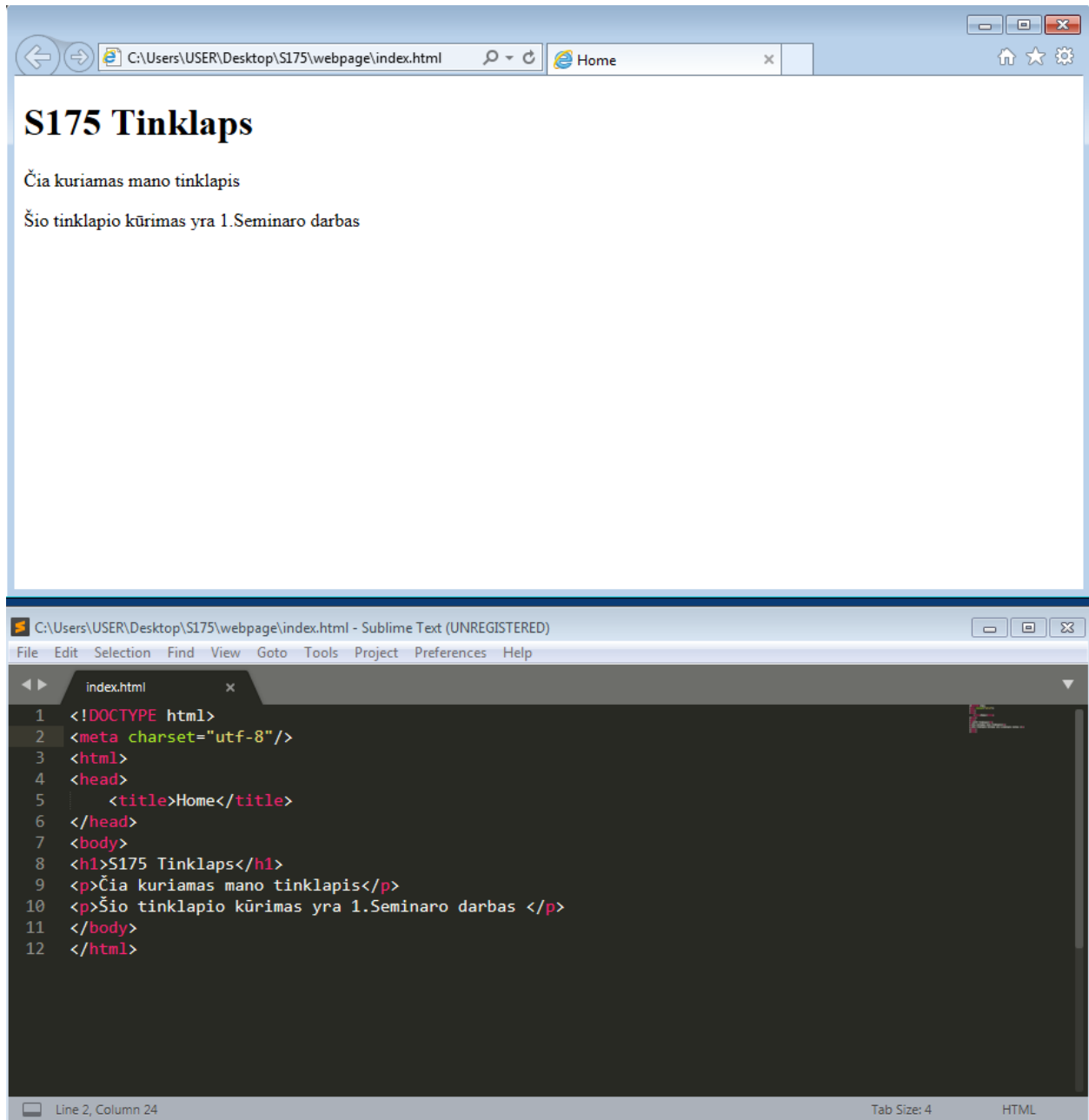


Figure 3: Caption for the picture.

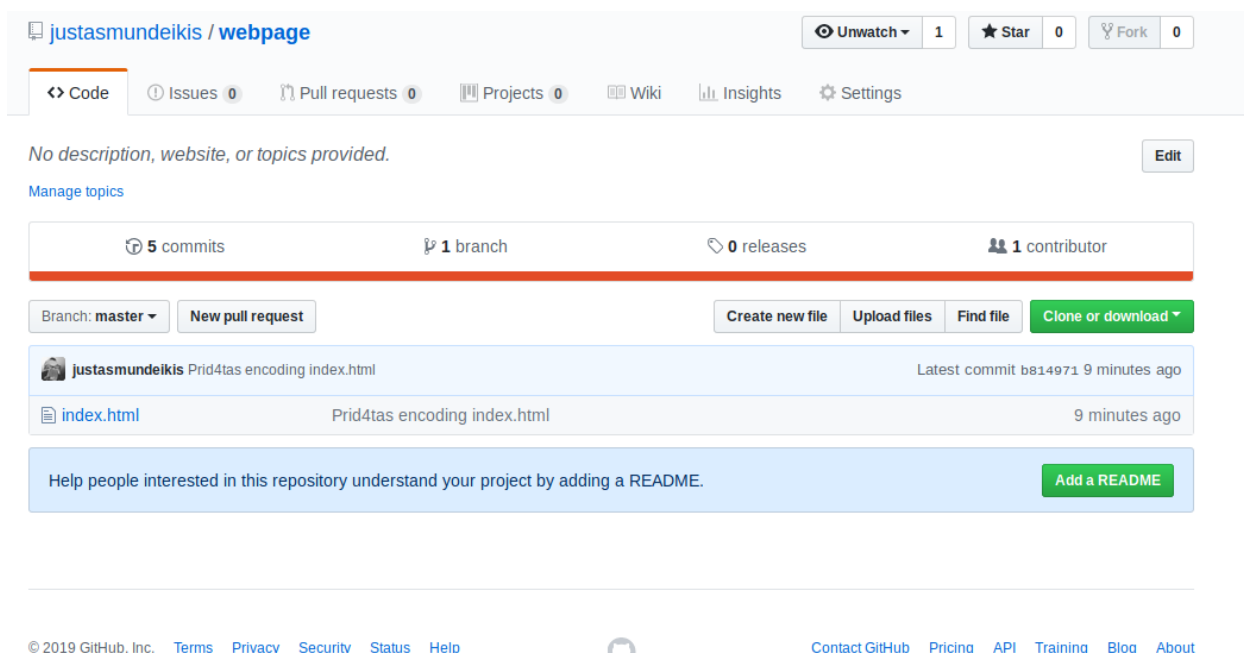
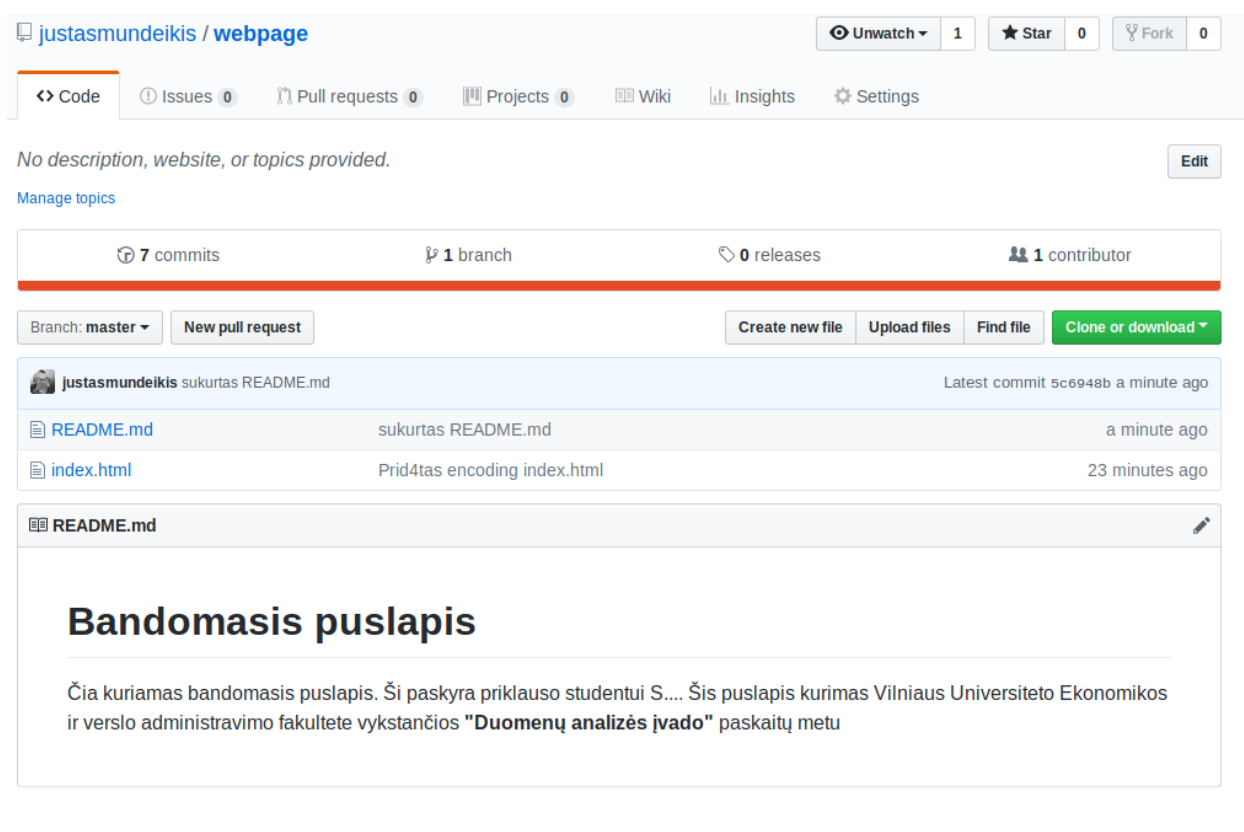


Figure 4: Caption for the picture.



Puslapis dabar atrodo gana nykiai, todėl pabandysime pridėti porą linksmų paveikslėlių. Siekiant, jog





Figure 5: Caption for the picture.

folderyje “webpage” būtų bent kiek tvarkos, paveikslukus dėsime atskirame subfolderyje “images”

```
mkdir images
```

GitBash stiprus, nes gali netgi atsiųsti failus iš interneto:

```
curl -o images/github.png -OL https://upload.wikimedia.org/wikipedia/commons/thumb/b/b3/GitHub.svg/567px-GitHub.svg.png
```

Folderyje images turėjo atsirasti šis paveikslukas

Dabar reikia dar šį paveiksluką įterpti į index.html failą. Tai galime padaryti html body įrašę tokią eilutę: ``. Kadangi paveikslukas yra images folderyje tai įrašome atitinkamai `src=""` (source). Na bet kad būtų linksmiau, pridėdame dar ir vieną paveiksluką iš interneto. Visas kodas dabar turėtų atrodyti taip. Alternatyviai galite įklijuoti kitus paveikslukus.

```
<!DOCTYPE html>
<meta charset="utf-8"/>
<html>
<head>
  <title>Home</title>
</head>
<body>
<h1>S175 Tinklapis</h1>
<p>Čia kuriamas mano tinklapis</p>
<p>Šio tinklapio kūrimas yra 1.Seminaro darbas </p>
<p></p>
<p> Code

Issues 0

Pull requests 0

Projects 0

Wiki

Insights

Settings

No description, website, or topics provided.

Edit

Manage topics

8 commits

1 branch

0 releases

1 contributor

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download

justasmundeikis prideti paveiksluikai

Latest commit 28c47db 2 minutes ago

|            |                       |                |
|------------|-----------------------|----------------|
| images     | prideti paveiksluikai | 2 minutes ago  |
| README.md  | sukurtas README.md    | 27 minutes ago |
| index.html | prideti paveiksluikai | 2 minutes ago  |

README.md

## Bandomasis puslapis

Čia kuriamas bandomasis puslapis. Ši paskyra priklauso studentui S.... Šis puslapis kurimas Vilniaus Universiteto Ekonomikos ir verslo administravimo fakultete vykstančios "Duomenų analizės įvado" paskaitų metu

Figure 6: Caption for the picture.

### 3.3 Naujos atšakos sukūrimas (paralelinis darbas) ir about.html sukūrimas

GitBash sukuriame naują atšaką ir einame į ją

```
git checkout -b about-puslapis
# tolygu
git branch about-puslapis
git checkout about-puslapis
```

Pasidarome index.html kopiją naudodamiesi CLI komanda cp

```
cp index.html about.html
```

Atsidarome about.html su Sublime ir pakeičiame turinį, supildant savo duomenis

```
<!DOCTYPE html>
<meta charset="utf-8"/>
<html>
<head>
  <title>About</title>
</head>
<body>
<p><strong>Vardas:</strong> Vardas Pavardė</p>
<p><strong>Stud. nr.:</strong> S175</p>
<p><strong>Kursas</strong>: Ekonomika</p>
<p><strong>Grupė:</strong> xx</p>
</body>
</html>
```

Kadangi norime, jog ir pagrindiniame puslapyje (Home) būtų nuorodą į puslapi apie mus (About), turime pakeisti index.html. Įterpiame `<a href="about.html">Daugiau informacijos apie mane</a>`, tagas `<a>` reiškia “anchor” arba “inkaras” ir yra naudojamas nuorodoms.

```
<!DOCTYPE html>
<meta charset="utf-8"/>
<html>
<head>
  <title>Home</title>
</head>
<body>
<h1>S175 Tinklapis</h1>
<p>Čia kuriamas mano tinklapis</p>
<p>Šio tinklapio kūrimas yra 1.Seminaro darbas </p>
<br>
<a href="about.html">Daugiau informacijos apie mane</a>
<p></p>
<p>.

#### Source

Your GitHub Pages site is currently being built from the master branch. [Learn more.](#)

master branch ▾

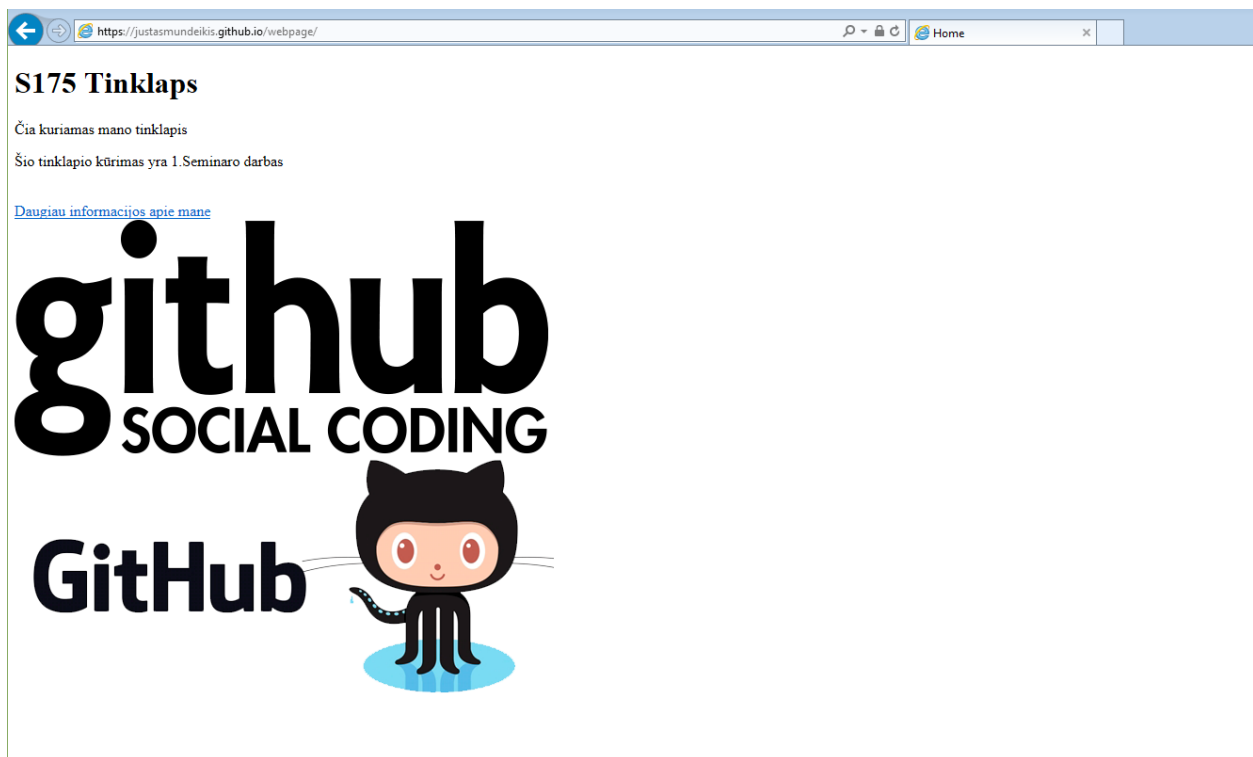
Save

#### Theme Chooser

Select a theme to publish your site with a Jekyll theme. [Learn more.](#)

Choose a theme

Atsidaro pilnai veikiantis tinklapis! Sveikinu!



Jeigu norite sukurti labiau sofistikuotą puslapį, galima naudotis html editoriaus pagalba, pvz., <https://html-online.com/editor/>

### 3.5 Tinklapių url į google sheets

Tam kad galėčiau įvertinti Jūsų darbą, šio tinklapio, ar kad ir koki susikūrėte kitą, url nukopijuokite į VMA pateiktą *Google Sheets*