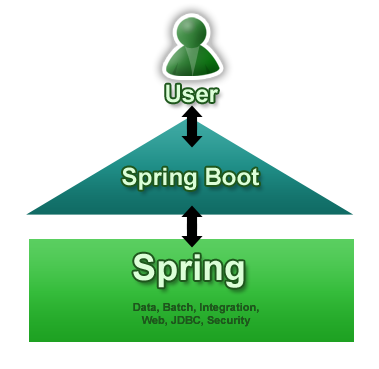
**Spring Boot**

**Spring Boot** je organziran na način da pojednostavni stvaranje, preko platforme **Spring**, aplikacije i servise na stručnoj razini s minimalnom razinom složenosti. Spring Boot također omogućuje da postojeći i novi korisnici vrlo lako koriste segmente Spring-a koji su im potrebni. Može se koristiti u svrhu stvaranja samostalnih (stand-alone) Java aplikacija. SB pruža i 'command line tool' preko kojeg se vrlo lako pokreću Spring skripte.



Spring Boot ne generira nikakav dodatni kod. Nije postoji niti potreba za XML postavkama za rad! Spring skrpite se pišu jezikom **Groovy**, sintaksa vrlo nalik na Javu, ali je pojednostavnjena s obzirom na Javu. Groovy je vrlo intuitivan pri pisanju, stoga je moguće logički deducirati što tuđi kod znači s vrlo malo razmišljanja. Primjer jednostavne skripte:

@Controller

class ThisWillActuallyRun {

@RequestMapping("/")

@ResponseBody

String home() {return "Hello World!"}

}

Preko Spring Boot CLI-a s komandom **‘spring run webapp.groovy’** se pokreće ova jednostavna skripta  
Vise informacija se nalazi na [spring.io](https://spring.io/blog/2013/08/06/spring-boot-simplifying-spring-for-everyone/)

**Spring Tool Suite**

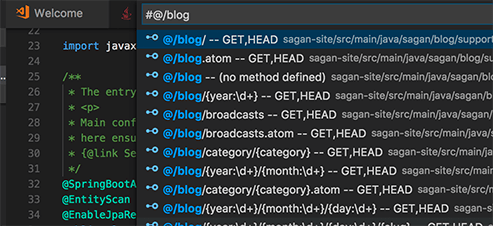
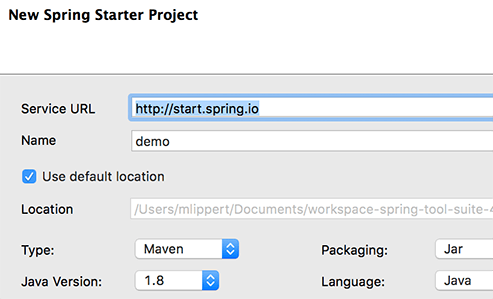
**Spring Tool Suite (STS)** pruža među najboljim okruženjenjima za stvaranje i razvoj aplikacija koje rade preko **Spring platforme**. **STS** pruza mnogo alata za rad s najnovijim izdanjima Jave i Springa te je najčešće na vrhu liste Eclipse-ovih alata. STS podržava razvojni tip **Pivotal tc Server**-a, dogovorena zamjena za Apache Tomcat, vrlo dobro optimiziran za rad u Spring-u.

**Pivotal Tc Server** pruža graficki (real-time) prikaz od performansi aplikacija, što omogućuje softver programerima lakše uočavanje problema u aplikaciji te ubrzava cijeli razvojni proces projekta.

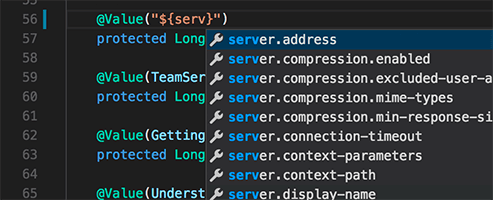
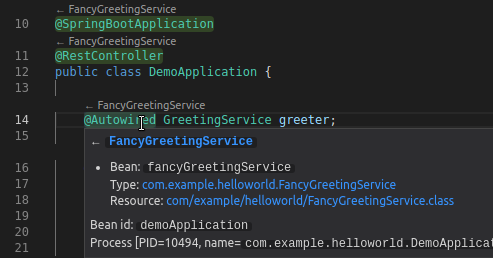
Aktualni STS je **Spring Tool Suite 4**:  
 **-** Najnovija generacija Spring alata za svoje radno okruženje  
 **-** STS4 je većinski napravljen ispočetka  
 **-** Pruža svjetsku razinu podrške za rad nad Spring aplikacijama  
 **-** STS4 se može koristiti u više razvojnih okruženja: **Eclipse, Visual Studio Code, Atom IDE**

**Spring aware**Stvoren za rad sa Spring platformomi Spring Boot-om**,** vrlo lako detektira bilo kakvu Spring aplikaciju i pruža mnoštvo alata i preporuka za pisanje koda.

**IDE diagnostic**Koristi se u svim inačicama STS-a kroz sva razvojna okruženja koja imaju njegovu podršku.  
Služi za dodavanje dodatnih sučelja i uključivanje različitih inačica alata iz STS-a koje bi programeru omogućile lakši razvoj aplikacije.



Intuitivno stvaranje **Spring** podržane aplikacije Olakšane pretrage svih izvornih kodova   
kroz par jednostavnih koraka. za 'importanje' u svoju aplikaciju.



IDE automatski prati što pišete u kodu te odmah  
traži sve moguće nadopune tako da se izbjegnu Uz to pruža dijagnostiku svih djelova vaše  
pogreške na razini tipkanja samog aplikacije čak dok je u pogonu, u svrhu koda. pronalaženja mogućih grešaka

**GitHub**

**GitHub** je distribuirani sustav za upravljenje izvornim kodom nastao 2005. godine iz nezadovoljstva razvijatelja Linux operacijskoga sustava karakteristikama postojećeg sustava za upravljanje izvornim kôdom ("BitKeeper").

**Svojstva GitHuba:**

* **Distribuiranost**, za razliku od ranijih sustava za upravljanje izvornim kôdom (CVS,SVN) GitHub je distribuiran, dakle nakon inicijalnog "kloniranja" spremišta/repozitorija nije nužno za svaki *commit* odnosno inačicu softvera spajati se na središnje spremište, moguće je pohraniti inačicu u lokalno spremište i sinkronizirati spremišta izvornog kôda onda kad je to prikladno
* **Usklađenost s postojećim protokolima**, GitHub podržava HTTP, FTP, rsync, git ili ssh internetske protokole
* **Efikasnost u radu s velikim projektima**, npr. Linux Kernel

Jednostavnost stvaranja repozitorija se jasno vidi na primjeru:

**moj-prvi-projekt** i stvoriti novi repozitorij u njemu:

$ mkdir moj-prvi-projekt // stvara se direktorij na računalu **moj-prvi-projekt**

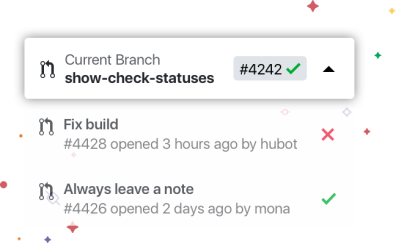
$ cd moj-prvi-projekt // lokaliziramo terminal na novostvoreni direktorij

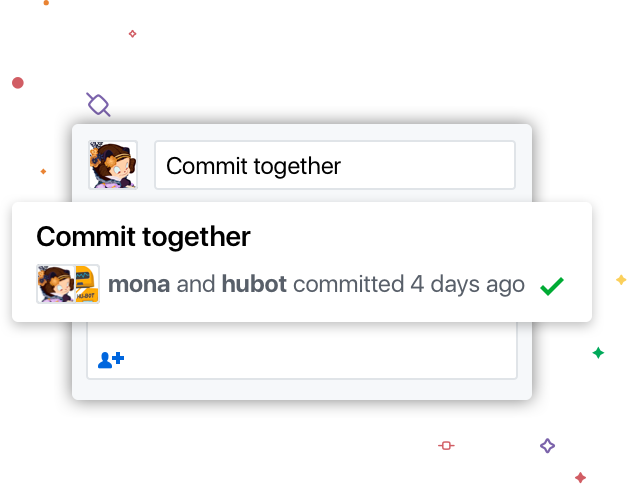
$ git init // git init stvara repozitorij s imenom izvorne mape

Initialized empty Git repository in /home/user/moj-prvi-projekt/.git/

$

Git naredbe imaju strog tip formatiranja: $ git <naredba> <opcija1> <opcija2> ...  
  
**GitHub Desktop**

Stvoren u svrhu brzog i jednostavnog pristupa  
GitHubu i korištenja većina njegovih funkcija   
bez potrebe korištenja Git naredbi preko CLI-a.



Vrlo lako se vidi tko je kada napravio 'commit'  
na GitHub.   
Također je vrlo fleksibilan za   
preuzimanje podataka iz jedne ili više grana.

Vrlo lako se primjete izmjene nad kodom  
sa obojanim **'diff'**-ovima.  
GitHub desktop kao i GitHub podržava   
razlikovanje vrsta jezika kojim je bilo koji  
**'push'**-an kod napisan.

**Google Cloud Platform**

**Google Cloud Platforma** je pružana od strane Googlea, sadrži skupinu servisa za programiranje na cloudu. Neki od pruženih servisa ubrajaju i podršku za spremanje podataka, analizu podataka, strojno učenje i slično. Za registraciju na GCP je potrebna **kreditna kartica** ili **bankovni račun**.  
  
**Sadržaj GCP**

**Compute**App Engine – Platforma za pokretanje Java, PHP, Python, C#, .NET, Ruby, Node.js i Go aplikacija  
Compute Engine – Infrastruktura za pokretanje MS Windows i Linux virtualnih mašina  
Cloud Functions – Funkicje kao servis za pokretanje 'event-driven' kod baziranog nad Node.js

**Pohrana i Baze Podataka**Cloud Spremište – Objektno spremište za nestrukturirane podatke.  
Cloud SQL – Baza podataka kao servis baziran nad MySQL i PostgreSQL.  
Cloud BigTable – Rukovođen NoSQL servis za baze podataka  
Cloud Spanner – Horizontalno skalabilan, snažno konzistentna, Relacijska Baza Podataka  
Persistent Disk – Block spremište za Compute Engine virtualne mašine

**Networking**VPC – Virtualni privatni cloud za upravljanje nad softver definiranim networkom cloud resursa  
Cloud Load Balancing – Softverski definirani servis za upravljanje stresom nad prometom podataka  
Cloud Armor – pruža sigurnost protiv napada 'hakera' npr. DDos napadi.

**Alati za Upravljanje**Stackdriver – promatranje, zabilježavanje te dijagnostika za aplikacije na GCP i AWS  
Cloud Deployment Manager – Alat za pokretanje GCP resursa koji su definirani na gotovim platformama  
Cloud Console – Web konzola za upravljanje GCP resursima  
Cloud Shell – Preko browsera (Firefox i slični) CLI za pristupanje GCP resursima.  
Cloud Console Mobile App – Android i iOS aplikacija za pristup GCP resursima.

**Identitet i Sigurnost**Cloud Identity – Single sign on (SSO) servis baziran na SAML 2.0 i OpenID.  
Cloud IAM – Identitiy & Access Managment (IAM) servis za definiranje postavki baziranih na   
role-based access control  
Cloud Identity-Aware Proxy – Servis za kontrolni pristup cloud aplikacijama koje rade na GCP bez korištenja VPN-a  
Cloud Data Loss Prevention API – Servis za automatizirano otkrivanje, klasificiranje, i raščlanjenje osjetljivih podataka  
Security Key Enforcement – Two-Step verification servis baziran na sigurnosnom ključu.

**IoT**Cloud IoT Core – Sigurna konekcija nad uređajem i upravljanje servisa za IoT  
Cloud IoT Edge – Dovodi AI (artifitial inteligence) u sloj za 'edge computing'.

**Google App Engine**

**Google App Engine** je web framework platforma za razvoj i pružanje web aplikacija u Google-upravljanim podatkovnim centrima. Aplikacije su prilagođene za rad preko mnoštvo servera.

App engine pruža automatsku skalabilnost podatkovnih resursa za aplikaciju. Što je potražnja za aplikacijom veća, to će se više resursa memorije pridodijeliti aplikaciji kako bi mogla brzo i neometano raditi na svakom klijentu bez kompromisa.

Google App Engine je besplatna platforma do određenih granica korištenja. Cijena za korištenje je diktirana potrebom za više memorije spremišta podataka, brzine konekcije, ili sati neometanog rada aplikacije.

Podržani pogramski jezici uključuju sve standardne razvojne programske jezike današnjeg svijeta. A to su uglavnom :

* Python
* Java
* JRuby
* Go
* PHP
* Node.js
* C#

Google App Engine sadrži mnoštvo Java i Python standarda i frameworkova,što uključuje  
Pyramid, Flask, web2py za Python i za Javine standarde je ključan servlet 2.5 tehnologija.

Preko integrirane baze podataka, Google Cloud Datastore, možda nepoznata mnoštvu programera, njoj se pristupa preko jednostavne Low-level API te ima podršku od JPA i JDO.

Spring platforma također može raditi preko Google App Engine.

**Pouzdanost i Podrška**App engine je dizajniran na način da može podnijeti više podatkovnih čvorova bez padova ili smetnja. Ova otpornost na gubljenje informacija je pokazana statistički da GAE ima 'down time' od 0%!  
Plaćena podrška je omogućena za motrenje aplikacija na dubljoj i sigurnijoj razini nego besplatna varijanta korištenja GAE.  
Besplatna podrška omogućuje sve standarne načine potražnje pomoći, Stack Overflow,GitHub, Server Fault itd.