

Dizajn REST Web API-ja

Denis Kranjčec, Srce
denis.kranjcec@srce.hr



Sveučilište u Zagrebu
Sveučilišni računski centar



srce
otvoreni pristup

Dizajn REST Web API-ja

- Što je REST Web API?
- Resursi i njihove reprezentacije
- HTTP i REST
- Dizajn API-ja baziranih na hipermediji (strategije, hipermedija formati, semantika, procedura)
- Primjer(i) RESTful Web API-ja



Što je REST? (1)

- „*Representational State Transfer (REST) is a software architecture style consisting of guidelines and best practices for creating scalable web services. REST is a coordinated set of constraints applied to the design of components in a distributed hypermedia system that can lead to a more performant and maintainable architecture.*” - https://en.wikipedia.org/wiki/Representational_state_transfer
- „*The World Wide Web represents the largest implementation of a system conforming to the REST architectural style.*”
- „*The term representational state transfer was introduced and defined in 2000 by Roy Fielding in his doctoral dissertation.*”
<http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm>
http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm



Što je REST? (2)

- Constraints

1. Client–server
2. Stateless
3. Cacheable
4. Layered system
5. Uniform interface - „REST is defined by four interface constraints: identification of resources; manipulation of resources through representations; self-descriptive messages; and, hypermedia as the engine of application state.”
6. Code on demand (optional)



Što je REST Web API?

- Poštuje REST ograničenja
- Baziran je na HTTP-u
- Izložen je kroz točno jedan početni URL (npr. <https://www.isvu.hr/api/>)
- Koristi standardne HTTP metode (npr. GET, POST, PUT, DELETE)
- Koristi „*Internet media type*” za reprezentaciju podataka (JSON, HTML, HAL, ...) - <http://www.iana.org/assignments/media-types/media-types.xhtml>
- Koristi *hypertext* za opisivanje stanja i povezanih resursa

Resurs

- Resurs može biti bilo što što je dovoljno važno da ga se treba referencirati.
- To može biti neka datoteka, redak u tablici u bazi, slika, ...
- Resurs ne mora biti nešto pohranjeno na računalu, nego to može biti bilo što kao npr. čovjek, svemir, srce, ideja, knjiga, ...
- Ključno je da svaki resurs mora imati svoj jedinstveni URL

Reprezentacija resursa

- Reprezentacija resursa je bilo koji elektronički dokument koji sadrži bilo koju informaciju o resursu.
- Kada klijent zatraži (GET) neki resurs, server bi mu trebao vratiti dokument koji opisuje resurs na neki razumljiv način – to je reprezentacija resursa.
- Resurs (uglavnom) ne može biti razmijenjen kroz internet, to može biti samo njegova reprezentacija.
- Jedan resurs može imati više reprezentacija – JSON i XML, JPEG i PNG, ...
- Klijent vidi isključivo URL i reprezentaciju resursa.

HTTP metode - značenje

- GET – dohvaća reprezentaciju resursa
- POST – kreira resurs niže u hijerarhiji temeljem poslano reprezentacije
- PUT – zamjenjuje u potpunosti stanje postojećeg resursa s poslanom reprezentacijom
- DELETE – briše resurs
- HEAD – vraća isključivo HTTP *Header*, ali ne i reprezentaciju resursa
- OPTIONS – vraća HTTP metode koje resurs trenutno podržava (temeljeno i na stanju resursa)
- PATCH – zamjenjuje dio stanja resursa s poslanom reprezentacijom (dodatak na HTTP standard)

Hipermedija

- Hipermedija (*hypermedia*) povezuje resurse i opisuje koje su akcije nad tim resursima moguće
 - Server tako daje do znanja klijentu koji su sljedeći koraci mogući, a klijent odlučuje što će se od toga dogoditi („koji link će pratiti”)
- To je **ključna razlika** u odnosu na uobičajene API-je
- Pomoću hipermedije klijent zna kreirati HTTP zahtjev (URL, HTTP metoda, HTTP zaglavlje i reprezentaciju koju može poslati).
- Postoji veliki broj hipermedijskih standarda
 - HTML
 - URI Templates – <http://example.org/search/{keyword}>
 - Link Header - <https://tools.ietf.org/html/rfc5988>
 - Link Relations - <http://www.iana.org/assignments/link-relations/link-relations.xhtml>
 - ...

Hipermedija – primjeri (1)

- Collection+JSON – baziran na *collection pattern-u* i JSON-u, dizajniran za rad i pretraživanje kolekcija podataka. (<http://amundsen.com/media-types/collection/>)
- Atom Publishing Protocol (AtomPub) – protokol za objavu i rad s web sadržajima (tipično „novinskim člancima”) temeljen na *collection pattern-u* (<https://tools.ietf.org/html/rfc5023>)
- HTML – temeljni jezik za kreiranje dokumenata i aplikacija za najširu moguću upotrebu
 - Namijenjen ljudima
- Microformats (ver. 1 i 2) – dizajnirani za ljude i strojeve, to su jednostavni i otvoreni podatkovni formati temeljeni na prihvaćenim standardima (<http://microformats.org/>). Primjeri su hCard/h-card, hCalendar/h-event -
`Pero Perić`

Hipermedija – primjeri (2)

- Hypertext Application Language (HAL) – općeniti jednostavni format koji omogućava jednostavno povezivanje između resursa (http://stateless.co/hal_specification.html):
 - JSON (*application/hal+json*) i XML (*application/hal+xml*) varijanta
 - Definirani su samo resursi i linkovi
- Uniform Basis for Exchanging Representations (UBER) – minimalni dizajn s podrškom za hipermedijom (<https://github.com/mamund/media-types/blob/master/uber-hypermedia.asciidoc>):
 - JSON (*application/vnd.amundsen-uber+json*) i XML (*application/vnd.amundsen-uber+xml*) varijanta
- JSON API - <http://jsonapi.org/>
- Hypermedia-Driven Web APIs (Hydra) - <http://www.markus-lanthaler.com/hydra/>
- Siren - <https://github.com/kevinswiber/siren>
- ...

Profili (*profiles*)

- Kako saznati dodatne detalje o semantici reprezentacije resursa koji nisu definirani *media type-om* („gdje je dokumentacija?”)?
- 'profile' Link Relation Type - <https://tools.ietf.org/html/rfc6906>
- Na tom linku programer/aplikacija može saznati sve detalje o reprezentaciji – ograničenja, konvencije i proširenja).
- Application-Level Profile Semantics (ALPS - <http://alps.io/>) – format za jednostavan opis semantike na razini aplikacije (slično kao microformats).
 - Može opisati aplikacijski semantiku uz HTML, HAL, Collection+JSON itd.
 - Opisuje akcije i podatke servisa, ali ne i format poruka, protokol ili alate.
- JSON for Linking Data (<http://json-ld.org/>) – opisuje *context* kroz poseban JSON dokument u kojem se nalaze dodatni linkovi s dodatnim opisom semantike.
- Moguće je i uključiti dodatne informacije u neke formate – npr. HTML, HAL

Postupak dizajna REST Web API-ja (1)

1. Popisati sve informacije koje klijent API-ja želi dobiti iz API-ja ili ga predati u API.
2. Definirati resurse i njihove reprezentacije te veze među njima (*site map*). Obično je to dijagram gdje su elementi reprezentacije resursa, a veze među njima linkovi. Linkovi i njihovi nazivi („*ref*”) su tu ključni.
3. Pronaći standarde koji odgovaraju u konkretnom slučaju
 - Link Relations - <http://www.iana.org/assignments/link-relations/link-relations.xhtml>
 - Dodatne opise - <http://schema.org/> i <http://apis.io/>
4. Odabrati media type
 - Poželjno neki standardni - Media Types - <http://www.iana.org/assignments/media-types/media-types.xhtml>

Postupak dizajna REST Web API-ja (2)

5. Opisati *profile*
 - I tu je poželjno bazirati se na standardima
6. Implementirati API baziran na dijagramu iz točke 2.
 - Koristiti *media type* iz točke 4.
 - Koristiti *profile* iz točke 5.
7. Objaviti početni URL – u idealnom slučaju to je dovoljno da klijenti počnu koristiti API, ali u praksi još vjerojatno nedostaje dokumentacija, upute i primjeri.

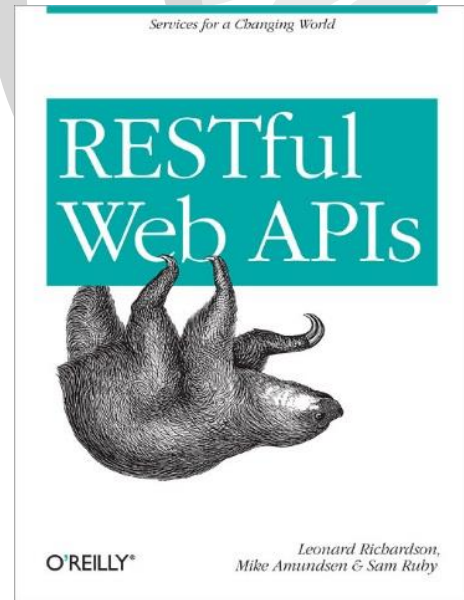
Primjeri RESTful WEB API-ja

- Amazon Web Services
- PayPal - <https://developer.paypal.com/docs/api/>
- GitHub - <https://developer.github.com/v3/>



RESTful Web APIs

- RESTful Web APIs - By Leonard Richardson and Mike Amundsen
 - <http://restfulwebapis.com/>
- {API}Search - <http://apis.io/>
- Mike Amundsen (@mamund)
 - <http://amundsen.com/>
- Roy T. Fielding (@fielding)
 - <http://roy.gbiv.com/untangled/>



Dizajn REST Web API-ja

Denis Kranjčec, Srce
denis.kranjcec@srce.hr



www.srce.unizg.hr

Ovo djelo je dano na korištenje pod licencom Creative Commons *Imenovanje-Nekomercijalno-Bez prerada* 4.0 međunarodna.

creativecommons.org/licenses/by-nc-nd/4.0/deed.hr



Srce politikom otvorenog pristupa široj javnosti osigurava dostupnost i korištenje svih rezultata rada Srca, a prvenstveno obrazovnih i stručnih informacija i sadržaja nastalih djelovanjem i radom Srca.

www.srce.unizg.hr/otvoreni-pristup

