

Sadržaj

- [Uvod](#)
- [Embedded programiranje](#)
- [Internet of Things](#)
- [Organizacija rada](#)
 - [Sastanci - tehnički deo](#)
 - [Sastanci - netehnički deo](#)
 - [Struktura ljudi](#)
- [Struktura projekta](#)
- [Rad na platformi](#)
- [Alati](#)
- [Konkretni zadatak, opis i implementacija](#)
 - [Oblasti koje pokriva zadatak](#)
 - [Koncepti koji se koriste](#)
 - [Podešavanje okruženja](#)
 - [Postavka zadatka](#)
 - [Use Case](#)
 - [Dijagram entiteta](#)
 - [Poslovni Zahtevi](#)
 - [Automatizacija zahteva za održavanje](#)
 - [Sinhronizacija sa eksternim servisom](#)
 - [Konkretno rešenje zadatka](#)
- [Zaključak](#)
- [Literatura](#)

Uvod

Praksu sam obavio u kompaniji **Execom**, koju je u međuvremenu kupio **HIEC**. U pitanju je *outsourcing* kompanija koja ima niz timova, svaki od kojih radi na uglavnom jednom projektu za eksternog klijenta. Timovi se bave različitim tehnologijama, u zavisnosti od potreba klijenta.

Kao i svaka *outsourcing* kompanija, nema svoje interne projekte, već ih isključivo uslužno pravi za druge. Važno je napomenuti da je njihov nekadašnji interni projekat prerastao u tzv. *spin off* kompaniju **WolkAbout**, čije cloud tehnologije smo koristili na projektu koji sam izradio.

Tim u koji sam primljen bavi se sa **IoT** (*Internet of Things*) i **Embedded** tehnologijama i u tom momentu je radio sa dva različita klijenta na proizvodima koji su njima bili potrebni, o čijim detaljima ne mogu govoriti zbog NDA (Non-disclosure agreement) koji sam potpisao.

Srećom, projekat na kojem sam radio je smišljen od nule isključivo u edukativne, odnosno svrhe moje prakse, te o njemu mogu govoriti. Projekat je open-source i njegov kod je javno dostupan na mom ličnom GitHub profilu.

Budući da ove oblasti (a posebno embedded) zahtevaju visok nivo performansi, izabran je programski jezik **C++**. Njegova prednost u odnosu na **C** jeste prisustvo Objektno-orijentisane paradigme, koja nam je olakšala organizaciju koda.

Takođe korišćen je i **Qt Framework** za *crossplatform GUI* aplikaciju, koji je u potpunosti objektno-orijentisan.

Embedded programiranje

Embedded softver (od engl. *embed*, “usaditi”) je računarski softver, napisan da kontroliše mašine ili uređaje koji se obično ne smatraju računarima u punom smislu te reči, poznatim kao embedded sistemi. Tipično su specijalizovani za određenu hardversku platformu na kojoj rade i imaju vremenska i memorijska ograničenja.¹ Ovaj termin se ponekad koristi

naizmenično sa **firmware**-om.²

Hardver na kojem se nalazi embedded softver, zbog prostornih ograničenja ili energetske ograničenja, je generalno manji i ne preterano moćan, zbog čega je efikasnost softvera izuzetno važna.

Primeri gde se ovakav softver koristi: 1. elektroniku automobila 2. modemi 3. “mozak” robota 4. kućni aparati (posebno “pametni”) 5. igračke 6. sigurnosni sistemi 7. automatizovani sistemi proizvodnje.³

Ovaj softver može biti vrlo jednostavan, kao što su sistemi za kontrolu osvetljenja koji rade na 8-bitnom mikrokontroleru sa nekoliko kilobajta memorije. Međutim, embedded softver može postati vrlo sofisticiran u aplikacijama kao što su ruteri, optički mrežni elementi, avioni, rakete i sistemi za kontrolu procesa.⁴

Internet of Things

Internet of Things (IoT) odnosi se na mrežu fizičkih uređaja, vozila, aparata i drugih fizičkih objekata koji su opremljeni senzorima i/ili aktuatorima (motorima), softverom i mrežnom povezanošću koji im omogućavaju prikupljanje i deljenje podataka. Ovi uređaji — poznati i kao “**pametni objekti**” — mogu varirati od jednostavnih “pametnih kuća” uređaja poput pametnih termostata, do nosivih uređaja kao što su pametni satovi i očeća sa RFID tehnologijom, do složenih industrijskih mašina i transportnih sistema. Tehnolozi čak zamišljaju čitave “pametne gradove” zasnovane na IoT tehnologijama.⁵

Drugim rečima, IoT je sve ono što nekolicinu embedded uređaja, pomoću računarskih mreža (bežične i žičane veze) uvezuje u jedan širi, inteligentan sistem. Posebna vrednost je u tome što slab embedded računar, može osnažiti resursima pravog računara sa kojim komunicira (lokalni računar). To je ono što ovakvom sistemu omogućuje autonomno reagovanje na uslove i obavljanje zadataka. Ova lokalna mreža uređaja može da se poveže sa internetom, a odatle sa pametnim telefonima ili stonim računarima na bilo kom kraju sveta.⁶

To znači da IoT sistemi mogu raditi praktično bilo šta, poput: 1. praćenja ekoloških uslova na farmama 2. upravljanja saobraćajnim modelima pomoću pametnih automobila i drugih pametnih saobraćajnih uređaja 3. kontrole mašina i procesa u fabrikama 4. praćenja stanja pacijenta i pružanja adekvatne doze medikamenta 5. do praćenja inventara i pošiljki u skladištima.

Potencijalne primene IoT-a su ogromne i raznolike, a njegov uticaj se već oseća u svim granama industrije. Kako broj uređaja povezanih sa internetom nastavlja da raste, IoT će verovatno igrati sve važniju ulogu u oblikovanju našeg sveta i transformaciji načina na koji živimo, radimo i međusobno komuniciramo.⁷ IoT takođe podstiče gomilanje podataka koji se mogu koristiti u treniranju modela veštačke inteligencije.⁸

U najčešćim primenama u većim preduzećima, IoT uređaji se koriste za praćenje širokog spektra parametara kao što su temperatura, vlažnost, kvalitet vazduha, potrošnja energije i performanse mašina. Ovi podaci mogu se analizirati u realnom vremenu kako bi se identifikovali obrasci, trendovi i anomalije koji mogu pomoći preduzećima da optimizuju svoje operacije i poboljšaju svoju profitabilnost.⁹

Organizacija rada

Sastanci - tehnički deo

U kompaniji Execom korišćena je **Agile** metodologija, konkretnije **Scrum**. To je podrazumevalo kratke sastanke svako (radno) jutro tzv *daily*, kao i nešto duže sumacije do sada postignutog na nedeljnom nivou, tzv *weekly*, krajem sedmice.¹⁰

Sastanci - netehnički deo

Pored toga, za praktikante je postojala još jedna forma nedeljnih sastanaka sa jednim ili više HR (*human resources*) menadžera, sredinom radne sedmice, sa kojima se pratio i “ne-tehnički” deo prakse - utisci i zadovoljstvo praktikanta, uklapanje sa drugim članovima tima. Iako se on činio neformalnim, HR menadžerke su beležile teze o čemu se govorilo i povezivale su to sa utiscima mentora.

Struktura ljudi

U odnosu na mojih prvih mesec dana u Execomu, gde sam prve sedmice imao jednog mentora (koji je kasnije uključio još jednog mentora za svakodnevnu komunikaciju i razjašnjavanje/rešavanje problema), u drugih mesec dana sam imao sreću raditi sa čak dvojicom mentora i vođom tima (koji je rukovodio projektom, vodio mentore i pomagao kad se oni susretnu sa poteškoćama).

Budući da se drugi projekat sastojao iz više delova, te da je bio veoma obiman za mesec dana rada, veći broj mentora je bio od presudnog značaja za uspešno obavljanje svih zamišljenih radnih zaduženja na projektu.

Struktura projekta

Projekat je uključivao tri zasebne hardverske komponente, svaku sa pratećim softverom koji je bilo potrebno isprogramirati. Tu je bila i četvrta komponenta, odnosno *cloud platform* kompanije WolkAbout.

Šema strukture projekta

Slika 1: Šema strukture projekta

Rad na platformi

Alati

Konkretna zadatka, opis i implementacija

Oblasti koje pokriva zadatak

Koncepti koji se koriste

Podešavanje okruženja

Postavka zadatak

Use Case

Dijagram entiteta

Poslovni Zahtevi

Automatizacija zahteva za održavanje

Sinhronizacija sa eksternim servisom

Konkretno rešenje zadatka

Zaključak

Literatura

1. ["Abstraction and the C++ machine model", B. Stroustrup, 2005↔□](#)
2. ["Towards Automated Dynamic Analysis for Linux-based Embedded Firmware", D. Chen et al, 2016↔□](#)
3. ["Embedded Systems Methods and Technologies", R. Oshana, 2012↔□](#)
4. ["Embedded Software", E. Lee, 2001↔□](#)
5. ["What is the internet of things?", IBM↔□](#)
6. ["An Internet of Things \(IoT\) architecture for embedded appliances", T. Yashiro et al, 2013↔□](#)
7. ["What is the internet of things?", IBM↔□](#)
8. ["IoT, cloud, big data and AI in interdisciplinary domains", Y. Chen, 2020↔□](#)
9. ["What is the internet of things?", IBM↔□](#)
10. ["The 2020 Scrum Guide", J. Sutherland & K. Schwaber↔□](#)