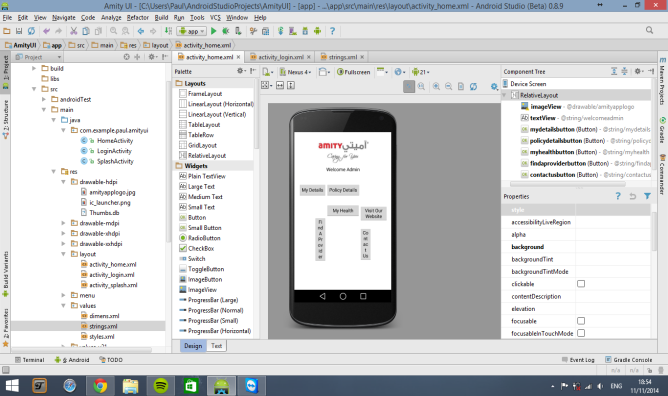
Upoznavanje sa Android Studio okruženjem

 Android Studio je razvojno okruženje zasnovano na IntelliJ IDEA softveru. Namenjen prvenstveno za razvoj Android mobilnih aplikacija, Android Studio je optimalno okruženje koje pruža sve mogućnosti trenutno dostupne u ovoj grani programiranja. Android aplikacije se mogu pisati i u drugim razvojnim okruženjima (na primer Eclipse IDE ili Visual Studio), ali sam razvoj je znatno otežan, jer druga okruženja nisu namenjena isključivo za Android i zahtevaju dodatna podešavanja u vidu plugin-a specifičnih za Android projekte. Budući da je Android Studio zasnovan na IntelliJ editoru koda i razvojnim alatima istog, sasvim je opravdano reći da je okruženje koje omogućava kompletno iskustvo u razvoju Android projekata. Korisnički interfejs je kreiran tako da su najkorišćenije funkcionalnosti uvek dostupne na vidljivim mestima, a takođe i pruža mogućnost reorganizacije paleta alata onako kako korisniku odgovara. Pored ove dve karakteristike, Android Studio pruža i više nego dovoljno opcija za neometani razvoj Android aplikacija, uključujući podešavanja raličitih perspektiva za prikaz strukture projekta, integraciju sa nekoliko raličitih kontrola koda (version control ili source control) i mnoge druge.

Slika 2

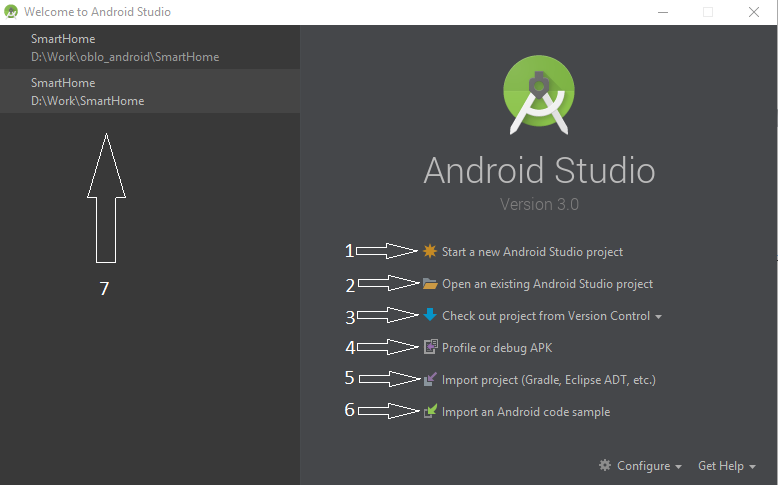
Slika 1

Oktobra 2017. godine je Android Studio 3.0 postao dostupan za korišćenje. Uz njega se otvorio novi spektar mogućnosti kao dodatak već postojećim. Ispravljeni su neki nedostaci prethodne verzije i dodati su novi plugin-i kao i integracija sa novim alatima za brži i bolji razvoj Android aplikacija. Poboljšanja u odnosu na prethodne verzije uključuju jednostavniju konfiguraciju okruženja, kao i brže kompajliranje projekata, zatim podršku za novu verziju Jave (Java 8) kao i Android Oreo (Android 8, API 26 i 27), ubrzanu sinhronizaciju Gradle build sistema, podršku Kotlin programskog jezika i mnogo drugih.

U nastavku je opisana procedura kreiranja Android projekta, kao i opis najosnovnijih alata korišćenih pri razvoju Android aplikacija.

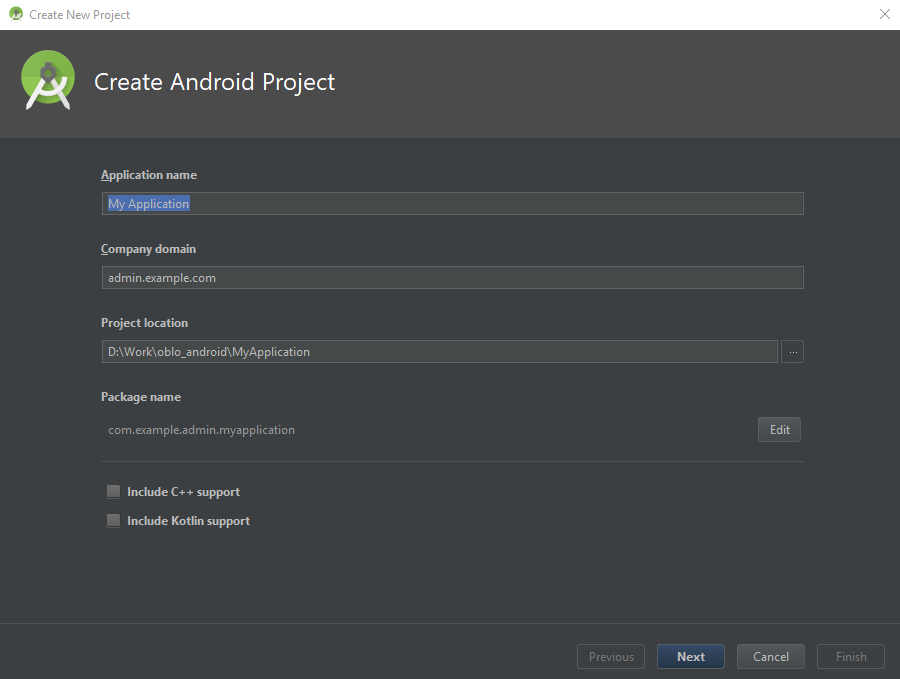
Kreiranje novog projekta i upoznavanje sa radom u Android Studio okruženju

Inicijalnim pokretanjem Android Studio razvojnog okruženja, otvara se početna forma koja nudi nekoliko opcija (slika 3):



Slika 3

1. Pokretanje novog Android projekta,
2. Otvaranje postojećeg Android projekta,
3. Dobavljanje projekta pomoću neke od postojećih kontrola koda,
4. Analiziranje ili debagovanje postojećeg APK fajla,
5. Ubacivanje postojećeg projekta u Android Studio okruženje,
6. Ubacivanje postojećeg Android koda u Android Studio,
7. Lista prethodno otvaranih projekata u Android Studio orkuženju. (Ova lista je prazna pri prvom pokretanju nakon instalacije okruženja)

Odabirom prve opcije za kreiranje novog projekta, otvara se nova forma (slika 4):

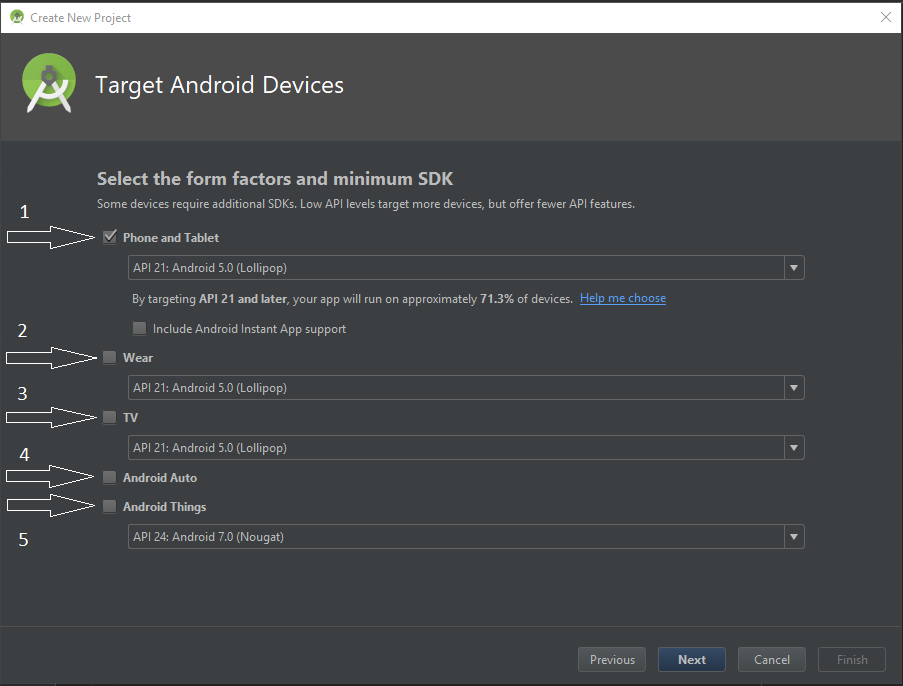
Slika 4

U polju *Application name* treba navesti ime nove aplikacije i ujedno i ime samog projekta. Polje *Company domain* predstavlja ime paketa u kome će prvobitno biti smeštene klase izgenerisane pri uspešnom prvom kreiranju Android projekta, a koji takođe služi kao jedinstveni identifikator aplikacije ukoliko bi se ona negde delila (na primer Google Play Store). Polje *Project location* predstavlja apsolutnu putanju do Android projekta u fajl sistemu. Sekcija *Package name* prikazuje strukturu i ime paketa na osnovu prethodno popunjena tri polja.

**Napomena: Menjanjem *Application name* ili *Company domain* polja, menja se i *Package name* prikaz.**

Dalje je moguće još omogućiti podršku za C++ ili Kotlin programske jezike. Android Studio ih ne smatra kao podrazumevane opcije, pa su zato po default podešavanjima isključene.

Klikom na *Next,* otvara se drugi deo forme za inicijalizaciju novog Android projekta (slika 5):

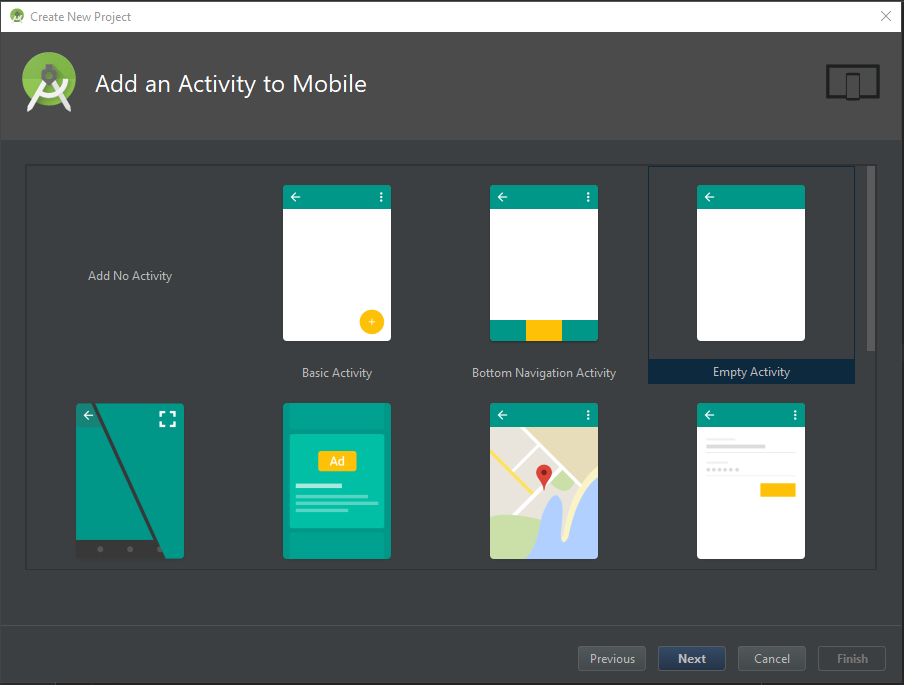


Slika 5

1. Polje *Phone and Tablet* služi za izbor minimalne API verzije koja će biti korišćenja prilikom implementacije projekta. Nakon izbora nekog API-a iz ponuđene liste, Android Studio ispisuje statistički podatak koji govori na koliko procenata mobilnih uređaja (na osnovu njihove podržane API verzije) će aplikacija moći da se pokrene. Što je viša verzija API-a, to je manji procenat uređaja koji će moći da pokrenu aplikaciju,
2. Polje *Wear* označava da će projekat biti implementiran i za druge uređaje poput pametnih satova (Smart watch),
3. Polje *TV* označava da će projekat biti implementiran i podržan i na TV uređajima,
4. Polje *Android* *Auto* označava da će aplikacija biti podržana i u vozilima sa odgovarajućom podrškom za Android aplikacije (na primer GPS navigacija),
5. Polje *Android Things* označava razvoj aplikacije koja će biti namenjena za druge hardverske platforme sa podrškom za Android aplikacije (na primer Raspberry Pi 3).

Navedene stavke mogu biti obeležene proizvoljno, ali minimalno jedna mora biti izabrana.

Klikom na *Next* otvara se pretposlednja forma za konfiguraciju novog projekta (slika 6):

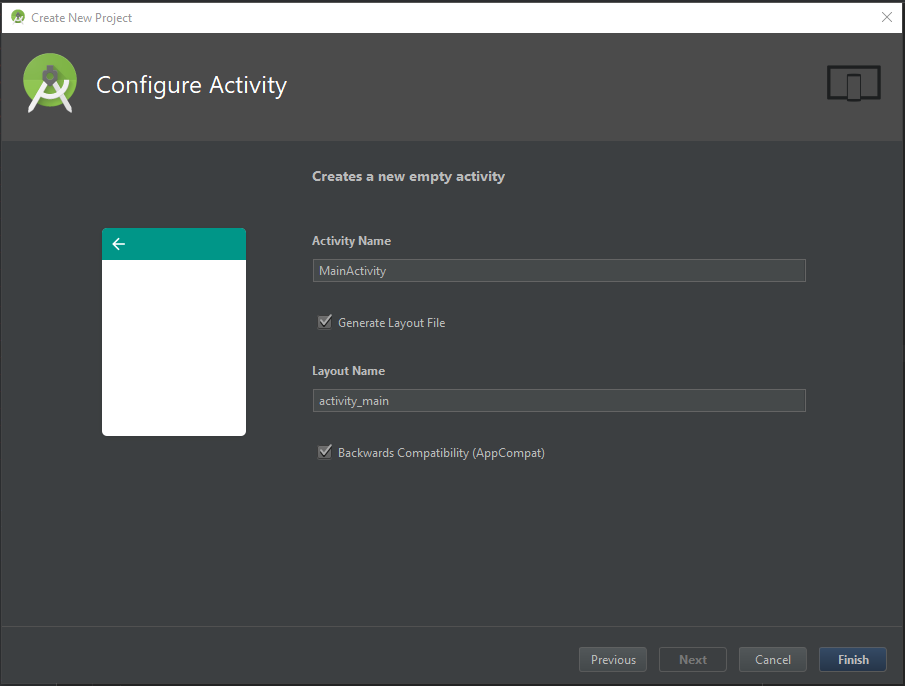


Slika 6

Ova forma služi za izbor početnog ekrana nove aplikacije. Prilikom odabira neke od opcija, kreiraće se odgovarajući fajlovi (.java i .xml) kojima će biti predstavljen prvi ekran aplikacije.

Moguće je odabrati i opciju *Add No Activity*  kojom će Android Studio biti obavešten da nije potrebno automatski generisati nikakve fajlove, već samo podrazumevanu strukturu projekta (detaljniji opis u nastavku). Odabirom ove opcije, Android Studio omogućava opciju završetka konfiguracije projekta već na ovoj formi.

Klikom na *Next,* otvara se još jedna i poslednja forma za početnu konfiguraciju Android projekta (slika 7):



Slika 7

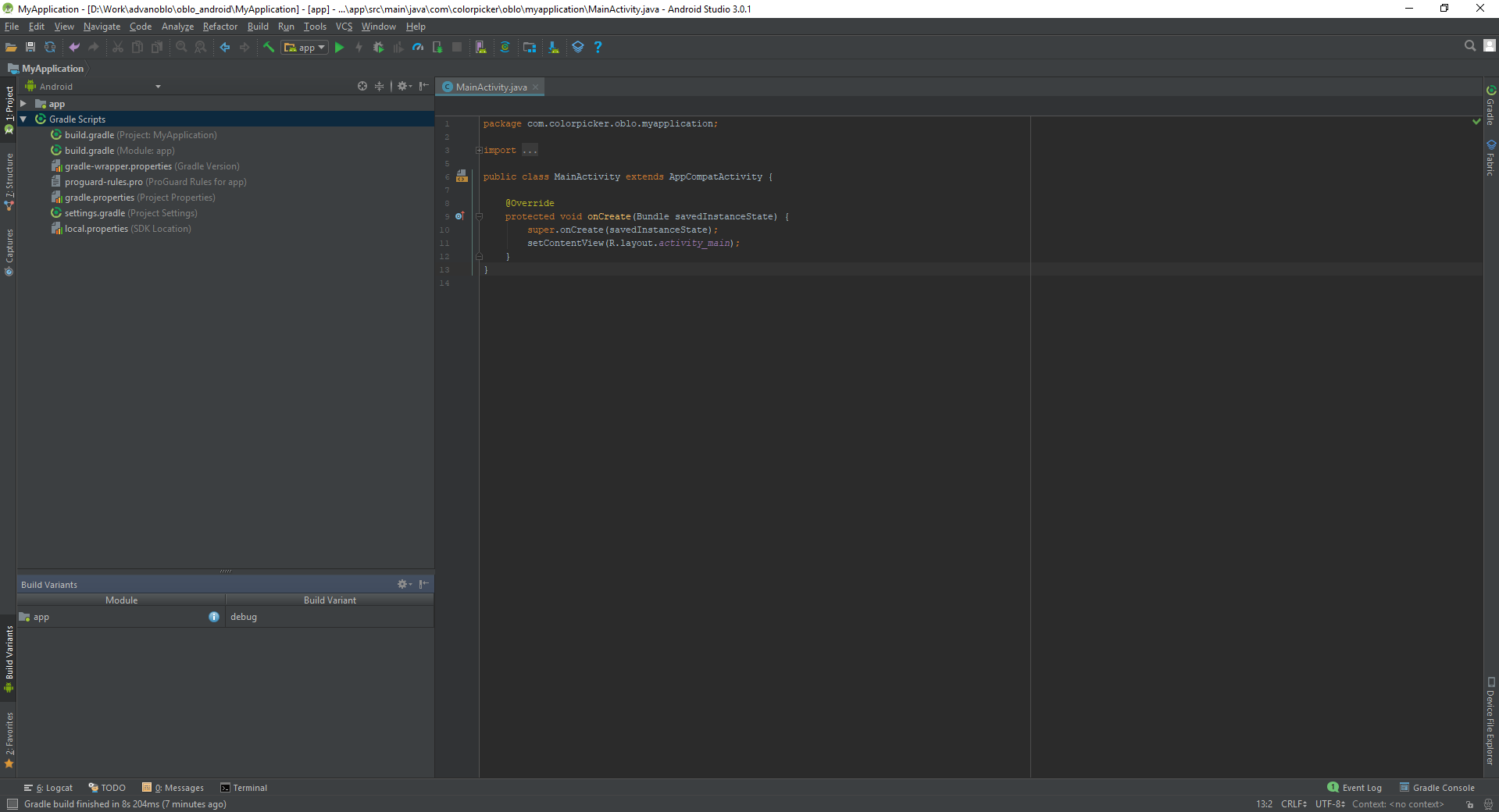
Ukoliko je prethodno izabrana neka od opcija da Android Studio automatski izgeneriše neki pošetni ekran i odgovarajuće fajlove za njega, na ovoj formi će biti ponuđene razne opcije u zavisnosti od izabranog ekrana. U polje *Activity name* se upisuje naziv početnog ekrana (Activity-a). Standard za izbor imena se drži uobičajene Camel Case notacije (na primer MainActivity). Android Studio nudi opciju automatskog generisanja odgovarajućeg *layout* (.xml) fajla za izabrani ekran. Ukoliko je generisanje .xml fajla izabrano, moguć je i odabir naziva tog fajla, iako Android Studio nudi automatski naziv na osnovu imena Activity-a.

Poslednja opcija je izbor da li će se kao osnovna klasa, koju će Activity nasleđivati, koristiti obična (starija) Activity ili novija АppCompatActivity klasa.

Nakon završetka konfiguracije projekta, klik na *Finish* pokreće generisanje potrebih fajlova, podešavanje optimalnog okruženja i generalnu organizaciju strukture projekta na osnovu odabranih opcija prilikom konfiguracije projekta.

Osnovni alati za korišćenje Android Studio okruženja

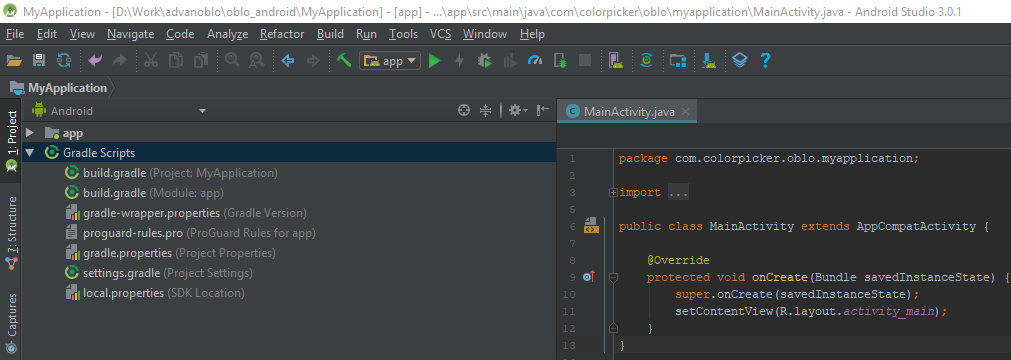
Nakon uspešnog kreiranja novog projekta, Android Studio prikazuje pošetno okruženje (slika 8), uređeno na osnovu konfiguracije i nekih podrazumevanih podešavanja. Sve što je konfigurisano tokom kreiranja projekta, kao i ugrađene i podrazumevane opcije, može se dodatno uređivati ili menjati zahvaljujući alatima i opcijama okruženja.



Slika 8

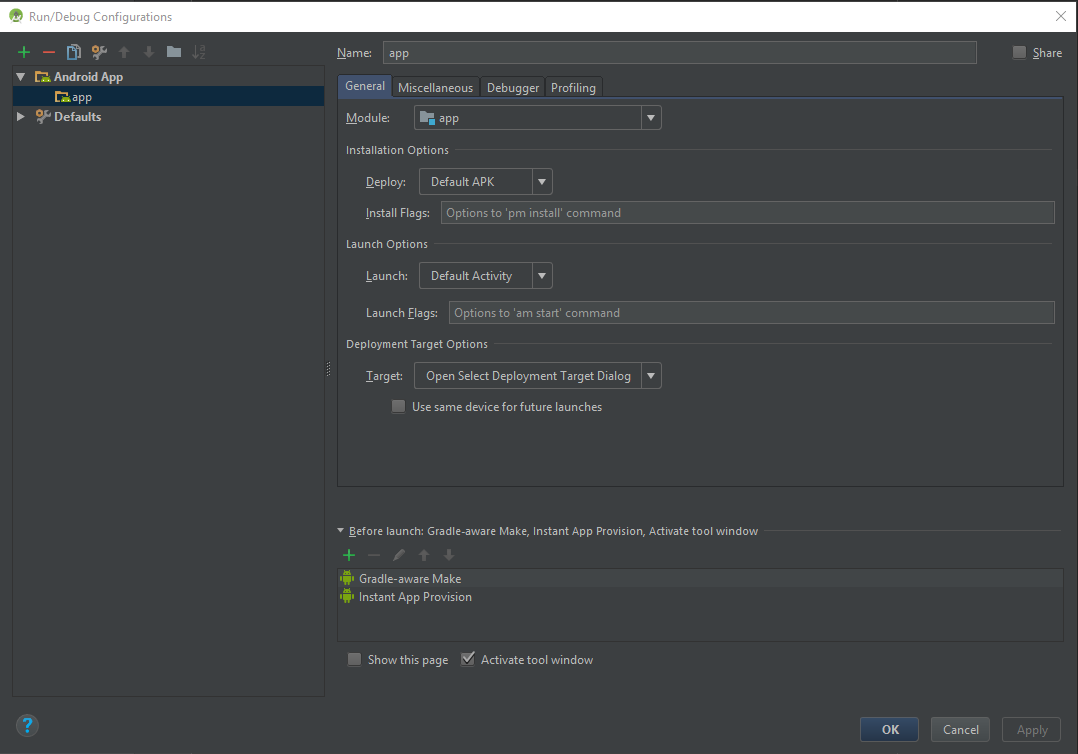
**Napomena: Na slici je prikazano Android Studio okruženje sa predefinisanom temom (Darcula theme). Boje pozadine editora, logova i drugih delova okruženja se mogu podešavati u ponuđenim opcijama ( na primer File => Settings => Appearance => Theme).**

Kao i u mnogim drugim okruženjima za razvoj aplikacija, Android Studio nudi osnovne opcije za konfiguraciju projekta, okruženja, koda, kompajliranja, verzionisanja, ažuriranja okruženja itd. Ispod navedenih opcija se nalaze najčešće korišćeni alati pri razvoju aplikacija. (slika 9)

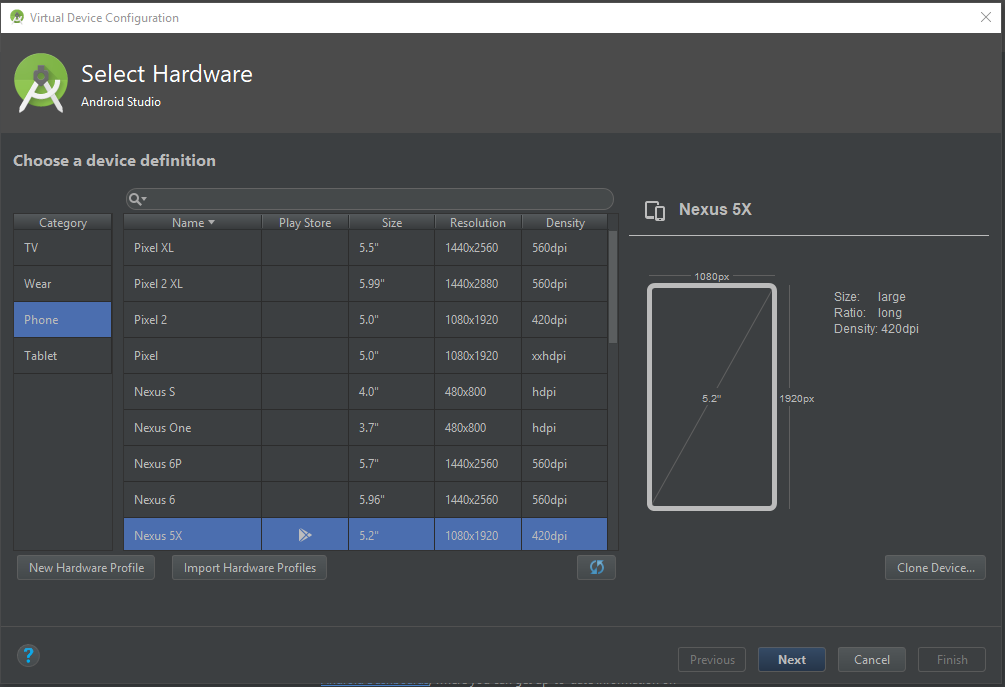


Slika 9

Ponuđene opcije obuhvataju:

1. Otvaranje novog fajla ili projekta,
2. Čuvanje svih prethodnih izmena,
3. Sinhronizaciju projekta – pregled svih fajlova koji imaju neku vezu sa projektom i njihovo ponovno učitavanje u slučaju da su menjani,
4. Undo, Redo, Cut, Copy, Paste, Find i Replace opcije za manipulaciju kodom i projektnim fajlovima,
5. Back i Forward opcije za vraćanje ili odlaženje na prethodno fokusirane delove projekta,
6. Ponovno kompajliranje projekta (Make project),
7. Konfiguraciju za kompajliranje projekta. Moguće je odabrati neku od predefinisanih konfiguracija za kompajliranje projekta, ali je ponuđena i opcija *Edit Configurations* koja otvara novu formu za kreiranje neke nove proizvoljne konfiguracije (slika 10) o čemu će biti reči kasnije,

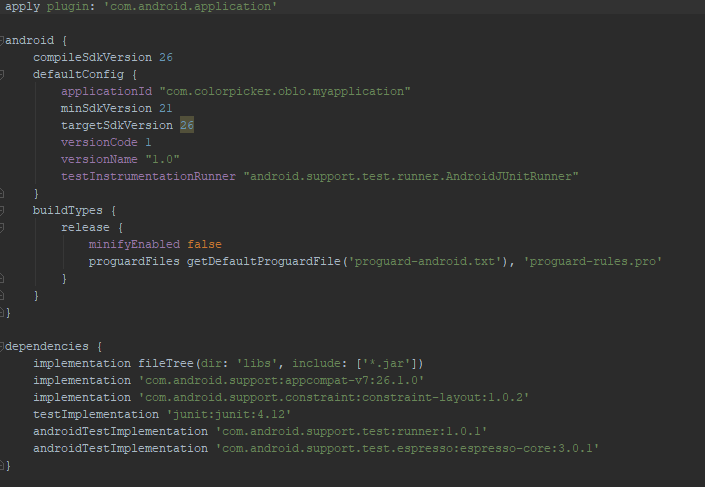
Slika 10

1. Opcije za kompajliranje i pokretanje aplikacije (samo u slučaju da je kompajliranje prošlo bez grešaka), debug aplikacije, pokretanje debug alata za već pokrenutu aplikaciju itd.,
2. AVD (Android Virtual Device) Manager – podešavanje virtuelnog uređaja (Emulator) na kome će moći da se pokrene aplikacija i prikažu ili testiraju njene funkcionalnosti (slika 11). O podešavanju i pokretanju emulatora će biti reči kasnije:

Slika 11

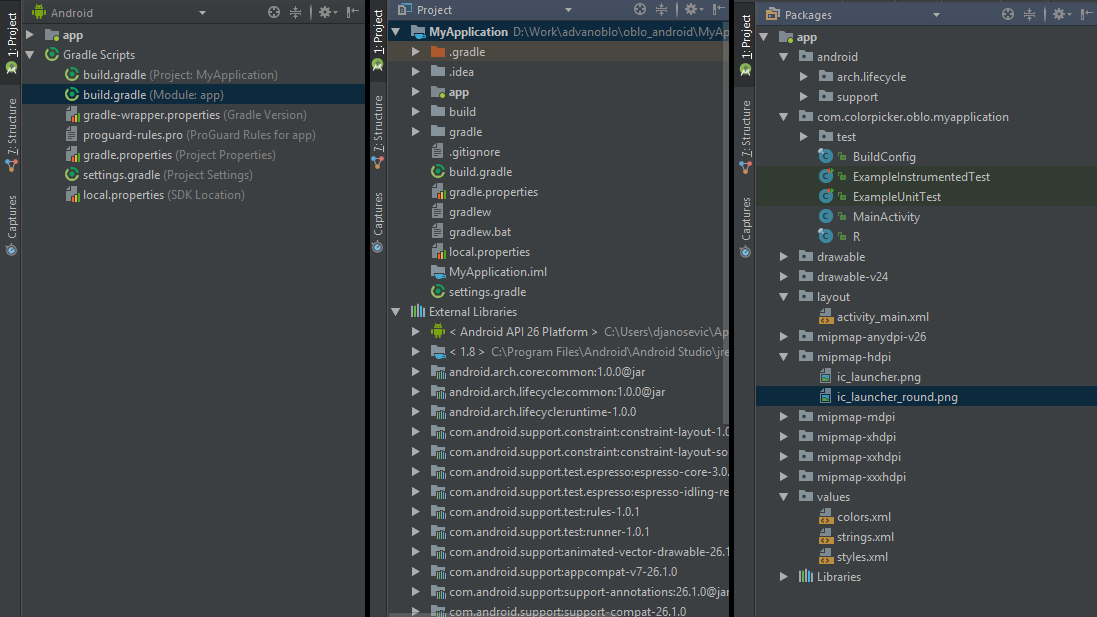
1. Sinhronizaciju projekta sa Gradle fajlovima. Ukratko, Gradle fajlovi su konfiguracioni fajlovi samog Android projekta i oni nisu vezani za okruženje. Gradle fajlovi se sami generišu pri pokretanju novog projekta, ali se njihov sadržaj može menjati tokom razvijanja aplikacije (na primer minimalna verzija API-a, pomenuta prilikom podešavanja projekta). Menjanjem Gradle fajlova, menja se i ponašanje Android aplikacije u različitim situacijama. Gradle fajlovi sadrže osnovne informacije o podešavanjima aplikacije kao što su: ID aplikacije (potreban kao jedinstveni identifikator aplikacije), minimalna verzija SDK-a, tipovi build-ova aplikacije (opširnije opisano kasnije) kao i neke eksterne ili ne-standardne biblioteke (dependencies) koje aplikacija koristi u svom radu (slika 12),

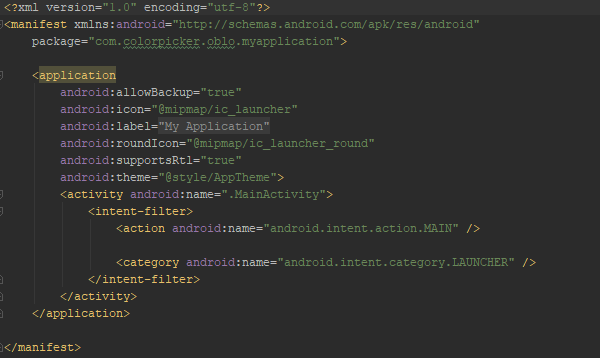
Slika 12

1.  Opcije za podešavanje strukture projekta,
2. SDK (Software Development Kit) Manager – podešavanje SDK-a aplikacije. Ova podešavanja uključuju izbor SDK platformi (verzije API-a), izbor SDK alata za lakšu kontrolu građenja i implementacije projekta i izbor lokacija sa kojih se mogu skidati dodatni SDK alati,
3. Pomoć za korišćenje Android Studio okruženja koja vodi na zvanični sajt Android community-a (<https://developer.android.com>).

Ispod navedene palete sa alatima, nalazi se apsolutna putanja do trenutno fokusiranog ekrana (fajla). Dalje, ostatak forme Android Studio okruženja može biti proizvoljno organizovan kao i prethodno navedeni delovi. Prvi sektor sa leve strane se, uglavom, koristi kao prikaz strukture fajlova u projektu. Mogu se izabrati i različite perspektive i, u zavisnosti od odabira, struktura projekta će biti prikazana na različite načine (slika 13):

Slika 13

Na slici 12 su prikazane tri različite perspektive strukture projekta (Android, Project i Packages). Ne postoji optimalna perspektiva, koriste se sve u zavisnosti od trenutnih potreba. Za početak je ipak najbolja preporuka koristiti Android perspektivu, jer se na njoj najlakše raspoznaju logičke podceline projekta. Prvi element je korenski *app* paket (ime zavisi od imena koje se odabere prilikom konfiguracije projekta). U ovom paketu se nalaze tri osnovna podpaketa (ovaj broj se menja u zavisnosti od smera razvoja projekta): *manifests*, *java* i *res*.

 U paketu *manifests* se nalazi jedan (ili više) AndroidManifest.xml fajlova. AndroidManifest.xml fajl sadrži osnovne informacije o aplikaciji koje su potrebne Android operativnom sistemu, kao i definicije svih ekrana, servisa i drugih komponenti koje aplikacija koristi prilikom svog funkcionisanja (slika 14).

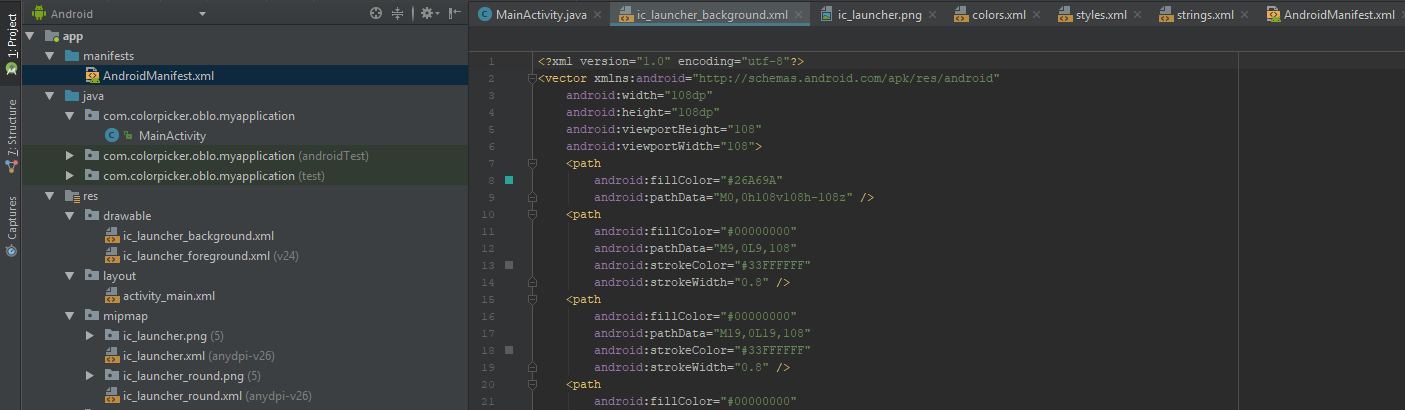
Slika 14

U paket *java* se smeštaju svi .java fajlovi koji se koriste u aplikaciji. Moguće je i poželjno razdvajati ih u logičke celine (na primer podela po ekranima aplikacije), kako bi struktura projekta bila jasnija i čitljivija.

Paket *res* predstavlja sve resursne fajlove aplikacije među kojima su .xml layout fajlovi (ununtar *layout* paketa), .xml drawable fajlovi (*drawable* paket), slike i/ili ikonice korišćene u aplikaciji (*mipmap* paket) i .xml fajlovi za stilove, stringove i boje korišćene u aplikaciji.

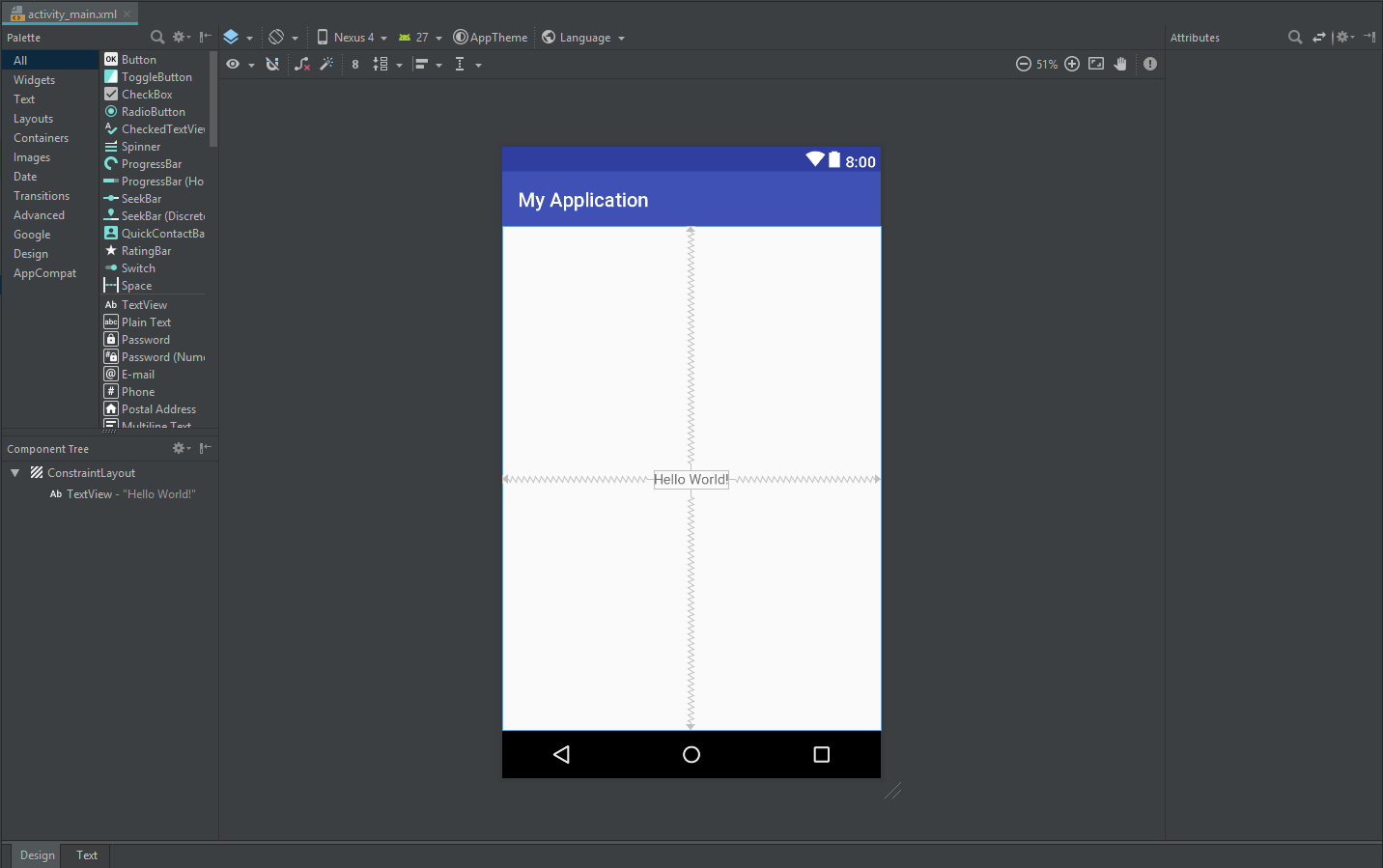
Ispod *app* paketa nalazi se još jedan *Gradle Scipts* odeljak sa svim konfiguracionim fajlovima, među kojima su i ranije pomenuti Gradle fajlovi. Uz svaki od njih, Android Studio nudi i lokaciju tog fajla, kao kratki opis njihove funkcije.

U centralnom delu Android Studio okruženja nalazi se i najveća forma koja predstavlja glavni editor koda Android aplikacije (slika 15). Svaki fajlova koje aplikacija sadrži može biti izabran i predstavljen na ovoj formi. Fajlovi se predstavljaju u različitim tabovima, što poboljšava preglednost editora, i tačno naznačava odabrani fajl. Boje kojima su predstavljene promenljive, rezervisane reči i logovi mogu se menjati medju File => Settings opcijama.

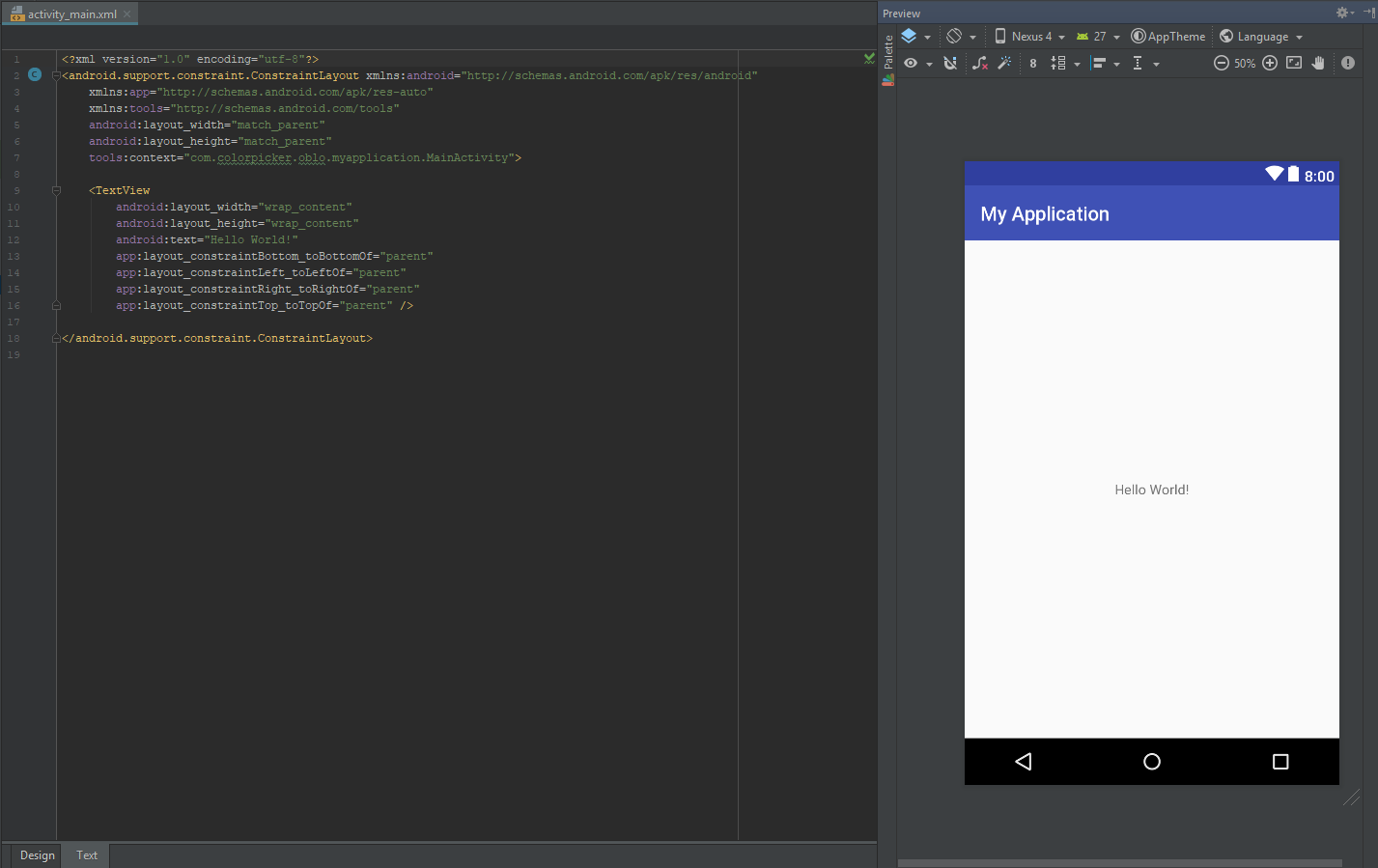


Slika 15

Prilikom odabira nekog od .xml layout fajlova, Android Studio omogućava dva tipa prikaza glavne forme (*Design* i *Text*). Jedan tip obuhvata paletu komponenata koje se mogu dodavati na formu prilikom čega se automatski generiše deo xml koda (*Design*), dok drugi tip (*Text*) predstavlja prikaz forme grafičkog interfejsa u vidu xml koda (slike 16 i 17). Dodavanjem xml tagova, Android Studio generiše (render) odgovarajuće komponente koje se prikazuju na *Design* perspektivi.



Slika 16



Slika 17

Na donjem delu ekrana nalaze se neke od opcija za analiziranje i praćenje funkcionisanja aplikacije tokom njenog rada. Najčešće korišćena perspektiva je *Logcat.* Ova perspektiva služi za prikaz svih logova napisanih u aplikaciji. Ukoliko aplikacija nema logova, moguće je izabrati i različite filtere za prikaz logova drugih aplikacija koje ih nude, a da su u isto vreme pokrenute na posmatranom uređaju (nebitno da li je fizički ili virtuelni). Ispis i praćenje logova su ključni elementi za lakše razumevanje života aplikacije, njenog ponašanja nakon različitih akcija korisnika, kao i mogućih grešaka koje mogu izazvati njeno prevremeno zaustavljanje. Osim *Logcat*  perspektive, postoje još *Messages* (ispis poruka prilikom kompajliranja aplikacije), *Terminal*  i *Debug* (postavljanje tačaka zaustavljanja aplikacije – break point kako bi se lakše očitavala trenutna stanja promenljivih i dolazilo do rešenja mogućih problema).