

Teme:

- uporaba že zgrajenih razredov,
- elektronsko podpisovanje dokumentov, vizualen podpis (pdf)

Viri:

- Apache PDFBox, <https://pdfbox.apache.org/>, dost. marec 2022
- Examples, Apache PDFBox, signatures <https://svn.apache.org/viewvc/pdfbox/tags/3.0.0-RC1/examples/src/main/java/org/apache/pdfbox/examples/signature/>, dost. marec 2022
- Certifikati in postopki, konfiguracije : https://gitlab.vegova.si/SerhioN/comptest05/-/tree/master/certificate_infos, dost. marec 2022

Opomba:

Za izvedbo boste potrebovali digitalno potrdilo v javanski hrabi za potrdila. Vanj boste v skladu s predhodno vajo uvozili s strani Vegova.CA.2 podpisano OpenSSL potrdilo in to potrdilo uporabljali za podpisovanje dokumenta.

Naloga 1

Cilj naloge je spisati program za elektronsko podpisovanje dokumentov vrste PDF. V ta namen uporabimo 'signature' mehanizme pdfbox-a (glej vire spodaj). V elektronskem podpisu naj bo še časovni žig ene od javnih storitev časovnih podpisnikov. Predlagam tisto na : https://www.freetsa.org/index_en.php (URI za podpisovanje je na <https://freetsa.org/tsr>, kot je spisano na osnovni strani spodaj) , sicer pa je lep spisek tudi v:

Free TSA, Manouchehri, <https://gist.github.com/Manouchehri/fd754e402d98430243455713efada710>, dost. marec 2021.

Ponastavite dan primer v »CreateVisibleSignature2.java« . Zaradi odvisnosti predlagam, da dol-potegnete celotno mapo (s podmapami). Celoten postopek je opisan v metodi usage/0 na koncu te datoteke.

Mimogrede: vizualno mora biti vidno : podpisnik(podpisnik:), datum podpisa(datum:), namen podpisovanja(namen:) in pa identifikacijska slika (lastnoročni podpis, portret, ...). Primer vidnega podpisa dodajam tule:



Pozicija posameznih elementov v podpisu ni bistveno; torej je slika lahko tudi spodaj ali v ozadju. Če pa je to lastnoročen podpis, mora biti v celoti viden.

Vir slike ni znan. Avtorju se opravičujem. Nosilca licence blagovne znamke lahko razberete iz : <https://www.calimero.com/>, dost. marec 2021.

Dodajam še morebitne vire (navedbe so slabe in za verzijo 3.0.0-RC1):

- Apache PDFBox, <https://pdfbox.apache.org/2.0/examples.html>, dost. marec 2022
- Examples, Apache PDFBox, <https://svn.apache.org/viewvc/pdfbox/tags/3.0.0-RC1/examples/src/main/java/org/apache/pdfbox/examples/pdmodel/>, dost. marec 2022
- Podpisovanje, <https://svn.apache.org/viewvc/pdfbox/tags/3.0.0-RC1/examples/src/main/java/org/apache/pdfbox/examples/signature/>, dost. marec 2022
- Fonts, SIL Language Technology, <https://software.sil.org/fonts/>, dost. marec 2022
- Google Fonts, <https://fonts.google.com/>, dost. marec 2022
- Bouncy Castle crypto knjižnice, https://www.bouncycastle.org/latest_releases.html, dost. marec 2022

Zadnji test je bil izveden s fontbox-3.0.0-rc1, uporabljene pa so naslednje:

- commons-logging-1.2.jar
- fontbox-3.0.0-rc1.jar
- pdfbox-3.0.0-rc1.jar
- bcprov-jdk15on-170.jar
- bcutil-jdk15on-170.jar

In; originalen primer zaženete z : `java CreateVisibleSignature2 imeKeystora gesloZaKeystore pdfZaPodpis sklikaPodpisa`

Predlagam, da sliko podpisa, keystore, tsa strežnik fiksno vgradite v program za podpis, tako da boste podpisovanje zaganjali zgolj z imenom dokument, ki ga boste podpisovali in pa geslom keystora !

(Skrajni rok za dokončanje in oddajo vaje je 7 dni od datuma, objavljenega v glavi tega dokumenta. V nabiralnik se odda zgolj elektronsko podpisano poročilo, ki je rezultat te vaje.)