

Jatkokehittäjien ohje TIM-liitetoiminnoille

Projekti: Titus

Dokumentin laatijat: Visa Naukkarinen, Enni Stylman

Laadittu: 25.4.2018

Muokattu: 29.5.2018

Lisätiedot: tim@jyu.fi

Ohjeessa kuvattavat toiminnot

Dokumentti kuvaa Titus-projektin kehittämien liitteisiin liittyvien toimintojen sijaintia ja rakennetta TIMissä. Liitetoiminnot sisältävät liitteiden lisäämisen TIM-dokumenttiin, liitteen leimauksen ja liitteiden yhdistämisen. Dokumentin tarkoitus on antaa kokonaiskuva, joka tukee koodin ja sen kommenttien ymmärtämistä.

Liitetoimintojen rakenne ja sijainti TIMissä

Liitetoimintoihin liittyvä koodi jakautuu TIMissä seuraavasti:

- Pääosa liitetoiminnoista sijaitsee `pdftools.py`-moduulissa.
- Liitteen leimaustoimintoon liittyvää koodia on myös TIM-dokumentin asetuksissa (preamble) sekä moduuleissa `parEditor.ts` ja `upload.py`.
- Liitteiden yhdistämiseen liittyvät reitit esitarkastuksineen ovat moduulissa `tim.py`.
- Liitteiden yhdistämiskäyttöliittymän koodia on tiedostoissa `mergePdfCtrl.ts`, `sidebarMenuCtrl.ts` ja `side_menu.html`.
- Leiman pohjana käytetään tiedostoa `stamp_model.tex`.
- Moduuleissa `containerLink.py` ja `svn3.py` on pienet lisäykset liitteiden `showPdf`-pluginia varten.

Lisäksi liitetoiminnot hyödyntävät TIMiin integroituja kolmansien osapuolien ohjelmia tai kirjastoja, mukaan lukien

- PdfLaTeX-komentoriviohjelmaa käytetään leimatiedostojen kääntämiseksi pdf:ksi, sekä
- Pdftk-komentoriviohjelmaa käytetään pdf-tiedostojen leimaamiseen ja yhdistämiseen.

Liitetyökalumoduuli `pdftools.py`

`pdftools`-moduuli kokoaa pdf-liitteiden leimaamiseen ja yhdistämiseen liittyviä toimintoja. Moduuli sijaitsee kansiossa `/timApp/tools/`.

Moduulin sisältö

Moduulin rajapintana käytettävät toiminnot kokoavat funktiot ovat seuraavat:

- `stamp_pdfs` lukee leimadataa (ks. luku 3.2) ja suorittaa leimauksen kutsumalla moduulin muita funktioita. Leimausprosessin vaiheet ovat - leimadatan oikeellisuuden tarkistus, - leimatekstien muodostaminen, - leiman luominen kullekin liitteelle, - liitteiden leimaaminen niillä sekä - väliaikaistiedostojen siivoaminen.
- `merge_pdfs` yhdistää pdf-tiedostoja hyödyntämällä moduulin muita funktioita. Kaksiosaiseen yhdistämisprosessiin kuuluvat
 - tarkistus siitä, ovatko yhdistettävät tiedostot olemassa ja pdf-muotoisia sekä
 - niiden yhdistäminen.

Leimaus- ja yhdistämisfunktioiden hyödyntämät pienemmät funktiot ovat seuraavat:

- `check_stamp_data_validity` antaa virheilmoituksen, jos leimadatassa havaitaan ongelmia.
- `check_pdf_validity` antaa virheilmoituksen, jos tiedosto ei ole pdf tai ei ole olemassa.
- `create_stamp` luo leimatiedoston pdf_{latex}-kutsulla.
- `call_popen` kutsuu komentoriviohjelmia kuten, `pdftk`.
- `stamp_pdf` leimaa pdf-tiedoston ja poistaa mahdolliset väliaikaistiedostot.
- `get_stamp_text` muodostaa leimatekstin leimadatasta halutulla muotoilulla.

Leimaus- tai yhdistystoimintoja käyttävät ylemmän tason moduulit voivat mahdollisesti hyödyntää myös seuraavia apufunktioita:

- `attachment_params_to_dict` muuntaa parametrilistan moduulin käyttämään dictionary-muotoon.
- `download_file_from_url` lataa tiedoston verkosta (oletuksena) liiteoperaatioiden käyttämään temp-kansioon.

Edellä mainittujen funktioiden lisäksi moduuli sisältää omia virheluokkia liittämisen ja yhdistämisen mahdollisille virhetilanteille, sekä muuttujien oletusarvoja, kuten leimojen oletusmuotoilu.

Leimadatan muoto

Moduulin keskeinen `stamp_pdfs`-funktio käyttää tietyn muotoista leimadataa. Leimadata `stamp_data` on muodoltaan lista dictionary-tietorakenteita, jotka ovat avain-arvo -parien kokoelmia. `stamp_data`-muuttujan käyttämät avaimet ovat

- `file` leimattavan liitteen polku merkkijonona,
- `date` liitteen päivämäärä merkkijonona,
- `attachment` liitenumero tai -kirjain kokonaislukuna tai merkkijonona,
- `issue` asia- tai listanumero kokonaislukuna tai merkkijonona, sekä
- `text` leimateksti vapaamuotoisena merkkijonona, joka on vaihtoehto muodoille `date`, `attachment` ja `issue`.

Avaimien määrittelyjärjestyksellä ei ole merkitystä niiden käytön kannalta. Leimatekstiä muodostettaessa avaimet `date`, `attachment` ja `issue` ottavat aina etusijan, jos dictionary sisältää kaikki kolme jossain järjestyksessä. `text`-avaimen arvoa käytetään vain, jos yksi tai useampi niistä puuttuu. Jos myös `text` puuttuu, leimausprosessi keskeytyy virheellisenä.

Kaikki muut muuttujat paitsi `file` hyväksyvät LaTeX-merkintöjä kuten `\newline`, LaTeX-taulukot ja tekstin värin vaihtamisen. `file` määrittää leimattavan liitteen sijainnin, joten se on pakollinen leimausprosessissa.

Leimatietueiden lisäksi funktiota kutsuttaessa voi asettaa `stamp_format`-parametrilla leimatekstin muotoilun, jolla voi sijoittaa avaimien `file`, `date`, `attachment` ja `issue` arvot merkkijonomuottiin (ks.

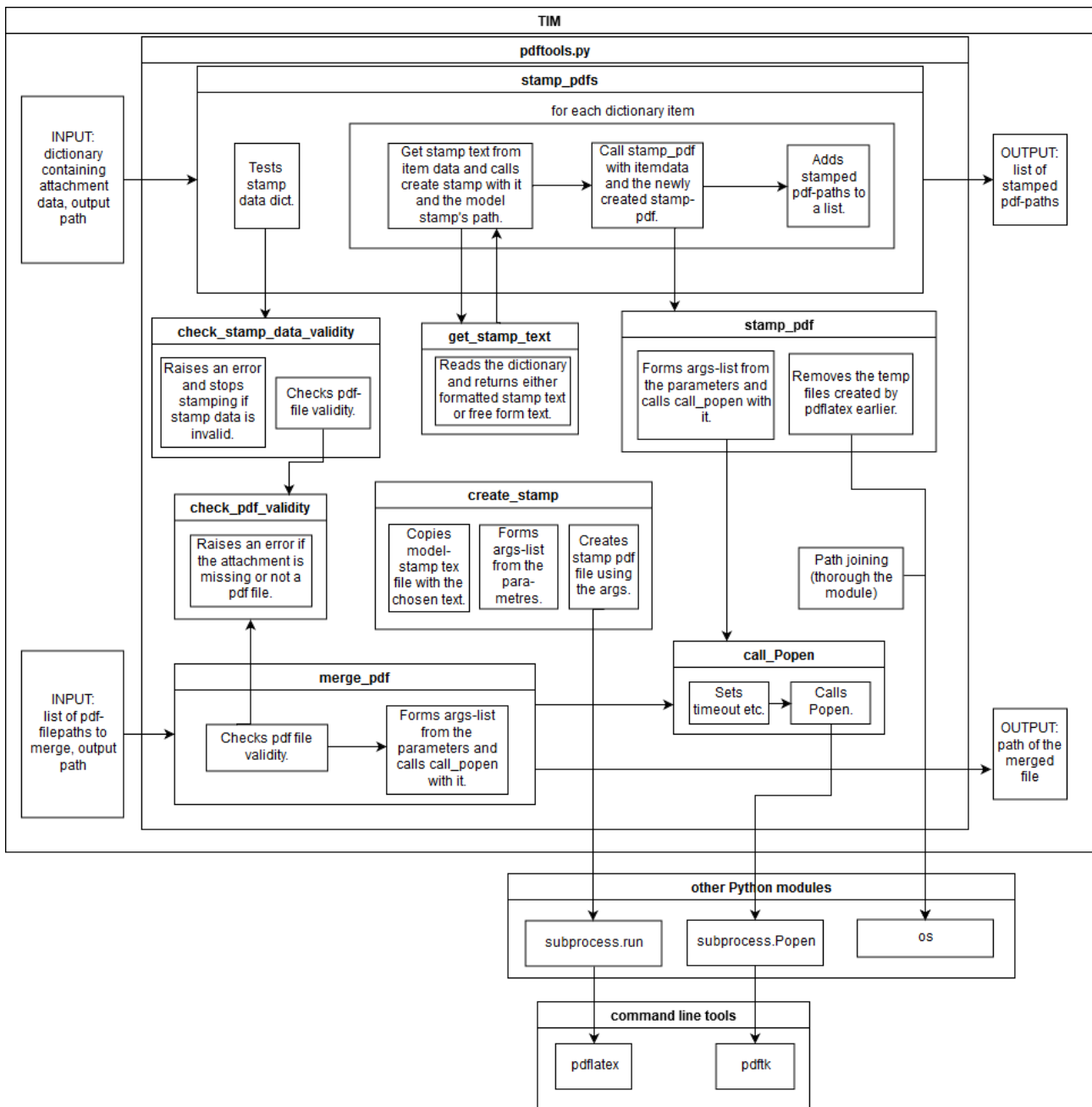
<https://tim.jyu.fi/view/kurssit/tie/proj/2018/titus/kayttoohjeet/liitetoiminnot#leiman-text-avaimen-arvoa-kaetetaessa-leimatekstiä-ei-muotoilla,-vaan-text-määrittää-suoraan-leiman-sisällön>.

Esimerkiksi, jos syötteenä annetaan leimadataparametri

ja leimaamiseen käytetään oletusmuotoilua `Kokous {date} \n\n LIITE {attachment} lista {issue}`, tällöin liitteiden leimateksteiksi tulee

Kaavio leimauksen ja yhdistämisen kutsupoluista

Oheinen yksinkertaistettu kaavio moduulin keskeisimmistä toiminnoista on laadittu draw.io-editorilla (ks. <https://www.draw.io/>). Alkuperäinen kaavio on saatavilla xml-tiedostona osoitteessa `/files/157509/pdftools.xml`.



TIM-reittien päämoduuli tim.py

tim.py on TIMin keskeisiä moduuleja. Se sisältää mm. osan reiteistä, joita TIM hyödyntää. Liitetoiminnoista liitteiden yhdistäminen tehdään siellä funktioissa merge_attachments ja get_attachments.

merge_attachments-funktiota kutsutaan mergeAttachments-reitin GET-metodilla, joka suorittaa seuraavat toimenpiteet:

- käyttäjän käyttöoikeuksien tarkastus,
- dokumentin kappaleiden lukeminen ja yhdistettävien liitteiden osoitteiden hakeminen showPdf-pluginien sisältä,
- liitteiden yhdistäminen pdfTools-moduulia käyttämällä, sekä
- yhdistetyn tiedoston palautus.

get_attachments-funktiota kutsutaan vastaavasti mergeAttachments-reitin POST-metodilla.

Sen tehtävänä on palauttaa merge_attachments-funktion yhdistämä dokumentti JSON-vastauksena, jonka TIM-sivun käyttöliittymä voi antaa käyttäjälle linkkinä.

Dokumenttien johdantosivut eli preamblet

Preamblet eli TIMin johdantodokumentit tai sivuasetukset sisältävät makrot, templatet, osan käyttöliittymästä sekä leimausmuuttujia (ks.

<https://tim.jyu.fi/view/kurssit/tie/proj/2018/titus/kayttoohjeet/liitetoiminnot#sivuaset> ja <https://tim.jyu.fi/view/tim/TIM-ohjeet#johdantodokumentti>).

Kappale-editori parEditor.ts

parEditor.ts sisältää kappale-editoriin liittyviä toimintoja ja sijaitsee kansiossa /timApp/static/scripts/

Liitetoimintoja koskevat lisäykset moduulin alussa hakevat TIM-dokumentin asetuksista (preamble) seuraavat leimaamiseen tarvittavat tiedot:

- knro kertoo kokousnumeron.
- dates sisältää kaikki vuoden kokouspäivämäärät. Siitä haetaan leimatekstiä varten knro:a vastaava päivämäärä.
- stampformat määrittää leimatekstin muotoilun.

Funktiossa onFileSelect editoritekstistä haetaan showPdf-moduulin sisältämä liitemakro, josta parsitaan käyttäjän syöttämä liitteen tunnuskirjain ja asialistan numero. Nämä yhdistetään leimadataksi attachmentParams, joka pakataan JSON-muotoon ja lähetetään normaalin upload-reitin mukana POST-metodilla. Liitettävä tiedosto kulkee saman metodin mukana.

Lisäksi moduuli käsittelee moduulin upload.py antaman vastauksen, jonka perusteella käyttäjälle palautetaan liitetiedoston linkki editoriin.

Tiedostojen palvelimelle lataus upload.py

upload.py sisältää palvelinpuolen toiminnot tiedostojen lataamiseksi palvelimelle. Liitetoimintoja koskevia muutoksia on tehty kahdessa sen funktiossa: upload ja upload_and_stamp_file.

Liitteiden leimaus hyödyntää upload-reittiä kuljettamalla liitettävän tiedoston lisäksi POST-metodin mukana leimaamiseen tarvittavan datan, jonka parEditor.ts lähettää. Jos kaikki tarvittava liitedata on mukana (mikä on mahdollista vain, jos editorissa on käytetty showPdf-makroa) tiedosto ohjataan leimauksen suorittavalle upload_and_stamp_file-funktiolle.

upload_and_stamp_file tekee muutoin samat asiat kuin normaalisti käytettävä upload_image_or_file-funktio, mutta se suorittaa alkuperäisen tiedoston lataamisen TIM-palvelimen kansioon ohella vielä tiedoston leimauksen ja palauttaa linkin leimattuun versioon. Lopputuloksena funktio palauttaa takaisin parEditor.ts:lle vastauksen, joka edelleen näytetään lopulta editoriin palautuvana leimatun version linkkinä.

Jos jokin leimaustoiminnon vaatimista ehdoista ei täyty (esim. tiedoston palvelimelle lataaminen suoritetaan ilman showPdf-makron mukana oloa), tiedoston lataaminen palvelimelle tapahtuu normaalisti ilman leimausta.

Mallileimatiedosto stamp_model.tex

Mallileimatiedosto toimii pohjana kaikille leimoille. Leimausprosessin aikana mallileimatiedoston sisältö kopioidaan väliaikaiseen tex-tiedostoon, jossa haluttu leimateksti lisätään sille osoitettuun kohtaan. Leimatekstin sijaintia merkitään tiedostossa tekstillä %TEXT_HERE, jonka päälle pdf`tools`-moduulissa kirjoitetaan kunkin leiman teksti.

Käytännössä mallileimatiedosto määrittelee yhden sivun mittaisen muutoin tyhjän dokumentin, jonka oikeassa yläkulmassa on leimatekstilaatikko. Laatikko ja sen sisältämä teksti tulevat oletuksena punaisella, sillä punainen väri näkyy paremmin poikkeustilanteissa, kuten liitetiedoston tekstin mennessä osittain päällekkäin leimatekstin kanssa.

Dokumentin taustaväriksi on asetettu läpinäkyvä, sillä oletuksena oleva valkoinen tausta peittäisi liitteiden sisällöt. Leimaus nimittäin toimii asettamalla leimatiedosto jokaisen leimattavan liitteen sivun päälle.

Jos leiman muotoilua halutaan muuttaa enemmän kuin vain tekstin sisällön osalta (esim. leimalaaticon värin, koon tai sijainnin vaihtaminen), muutokset tulee tehdä tähän tiedostoon. Vastaavasti jos leimatekstissä halutaan näyttää sisältöä, jota LaTeX ei oletuksena tue (esim. matemaattisia merkkejä), täytyy niitä vastaavat paketit määritellä mallitiedoston alussa.

Liitteiden yhdistämisen käyttöliittymä mergePdfCtrl.ts

Tiedosto vastaa liitteiden yhdistämistoiminnoissa käyttöliittymän painikkeen ja todellisen toiminnallisuuden yhdistämisestä. Kontrolleri kuuntelee käyttöliittymän painikkeita ja vastaa toimintojen toteuttamisesta sekä tulosten saattamisesta näkyville käyttöliittymään.

Käyttöliittymässä *Merge attachments* -painikkeen painaminen avaa liitteiden yhdistämisen dialogin käyttäjälle. Dialogin HTML-koodi on sijoitettu kontrollerin kanssa samaan tiedostoon. Mikäli kontrollerin koodimäärä alkaa kasvaa, voi HTML-koodin erottaa omaan tiedostoonsa (templates-kansioon TIMissä).