CVEditor

Release Report

Lonely little chamomile
Chryssa Terizi, 2553, Id: 24

VERSIONS HISTORY

Date	Version	Description	Author
4/2/2017	<0.1>	1st release of CVEditor	C. Terizi
21/5/2017	<0.2>	2nd release of CVEditor	C.Terizi

1 Introduction

1.1 Purpose

The preparation of a Curriculum Vitae (a.k.a. resume) is a very important task for everyone who is looking for a job or an internship. The main goal of the project is to develop CVEditor, an application that makes this task easier, especially for young inexperienced people such as graduate and undergraduate students. At a glance, CVEditor shall allow a user to prepare a new CV based on well known templates (chronological, functional, combination), change the structure of an existing CV by switching between these templates, generate actual CV documents in well known formats (Latex, Microsoft Word), trace changes between different CV versions and so on.

1.2 Document Structure

The rest of this document is structured as follows. Section 2 specifies the acceptance tests that have been employed for this release of the project. Section 3 specifies the main design concepts for this release of the project.

2 Acceptance Tests

2.1 Tests for User Story **US2**

Test ID	*1*
Class	*Section.class*
Test Name	*DateTests*
Description	*Test valid or invalid date formats*
Test Data	*"12/05/2015 - 20/03/2016"*
Expected result	*True*
Test Data	*"12/10/2015 - 12/11/2015"*
Expected result	*True*
Test Data	*"12/10/2005 - 20/03/2017"*
Expected result	*True*
Test Data	*"12/05/2015 - 20/03/2013"*
Expected result	*False*

Test Data	*"12/10/201512/11/2015"*
Expected result	*False*
Test Data	*"eg.12/10/2005 - 20/03/2017"*
Expected result	*False*
Test Data	*""*
Expected result	*False*
Test Data	*"12/10/2005"*
Expected result	*False*
Test Data	*"-"*
Expected result	*False*

Test ID	*2*
Class	*GeneralInformationPanel.class, AdditionalInformationPanel.class*
Test Name	*SectionPanelTests*
Description	*Sections are correctly updated from section panel(GUI)*
Test Data	name = "Chryssa Terizi"; address = "Panousi"; telephone = ""; mobile = "6984368861"; email = "chryssa.terizi@gmail.com";
Expected result	*Correctly updated GeneralInformationSection*
Test Data	*text = "I have a dog";*
Expected result	*Correctly updated AdditionalInformationSection*

Test ID	*3*
Class	*CareerSummaryPanel.class,EducationAndTrainingPanel.class, FurtherCoursesPanel.class, SkillsAndExperiencePanel.class*
Test Name	*AddElementTest*
Description	*Correctly add element*
Test Data	*Create two summary elements*
Expected result	*Correctly two summary elements added*
Test Data	*Create two education and training elements*
Expected result	*Correctly two education and training elements added*
Test Data	*Create two further courses elements*
Expected result	*Correctly two further courses elements added*

Test Data	*Create four skills and experience elements*
Expected result	*Correctly four skills and experience elements added*

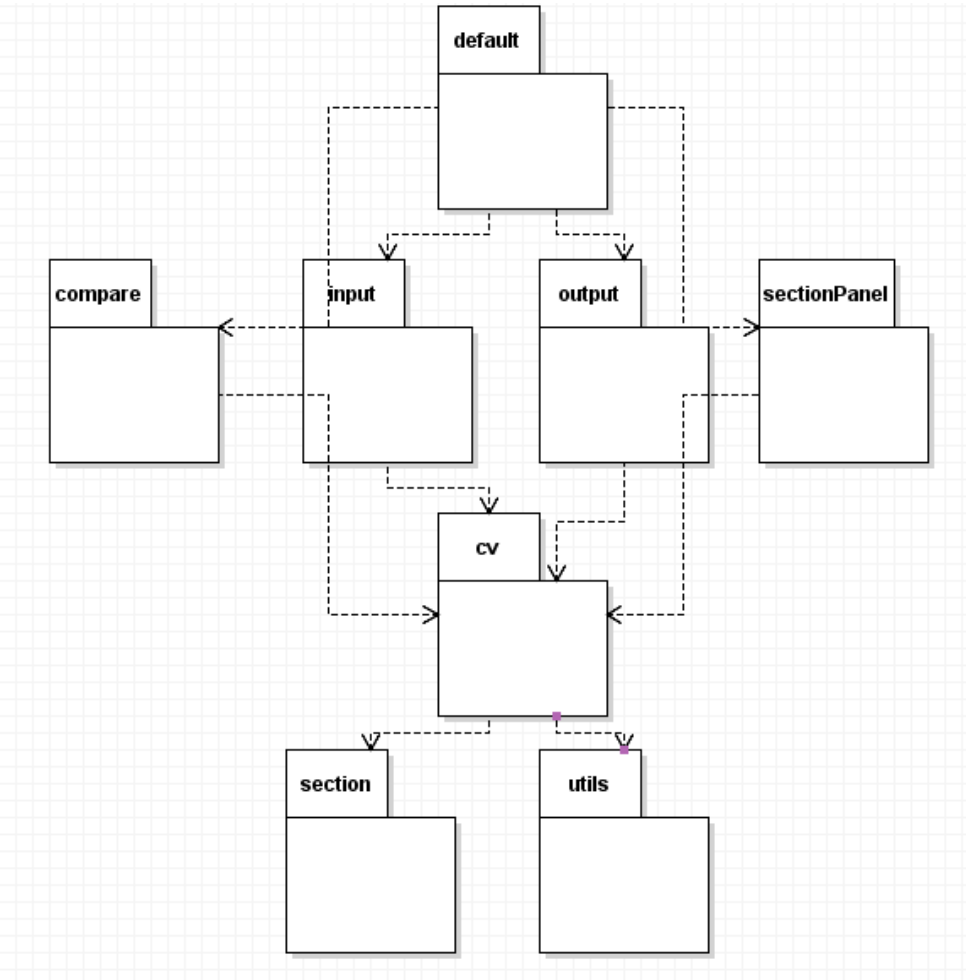
Test ID	*4*
Class	*ProfessionalExperiencePanel.class*
Test Name	*DeleteElementTest*
Description	*Correctly delete elements*
Test Data	*Create three achievements elements, delete second, first and first again*
Expected result	*Correctly three achievements elements added and deleted*

Test ID	*5*
Class	*TXTLoader.class, LATEXLoader.class*
Test Name	*LoadFileTests*
Description	*Correctly load file*
Test Data	*"loadFiles\\correct1.txt"*
Expected result	*True*
Test Data	*"loadFiles\\correct2.txt"*
Expected result	*True*
Test Data	*"loadFiles\\correct3.txt"*
Expected result	*True*
Test Data	*"loadFiles\\correct1.ltx"*
Expected result	*True*
Test Data	*"loadFiles\\correct2.ltx"*
Expected result	*True*
Test Data	*"loadFiles\\correct3.ltx"*
Expected result	*True*
Test Data	*"loadFiles\\missing_label.txt"*
Expected result	*False*
Test Data	*"loadFiles\\missing_section.txt"*
Expected result	*False*
Test Data	*"loadFiles\\wrong_label.txt"*

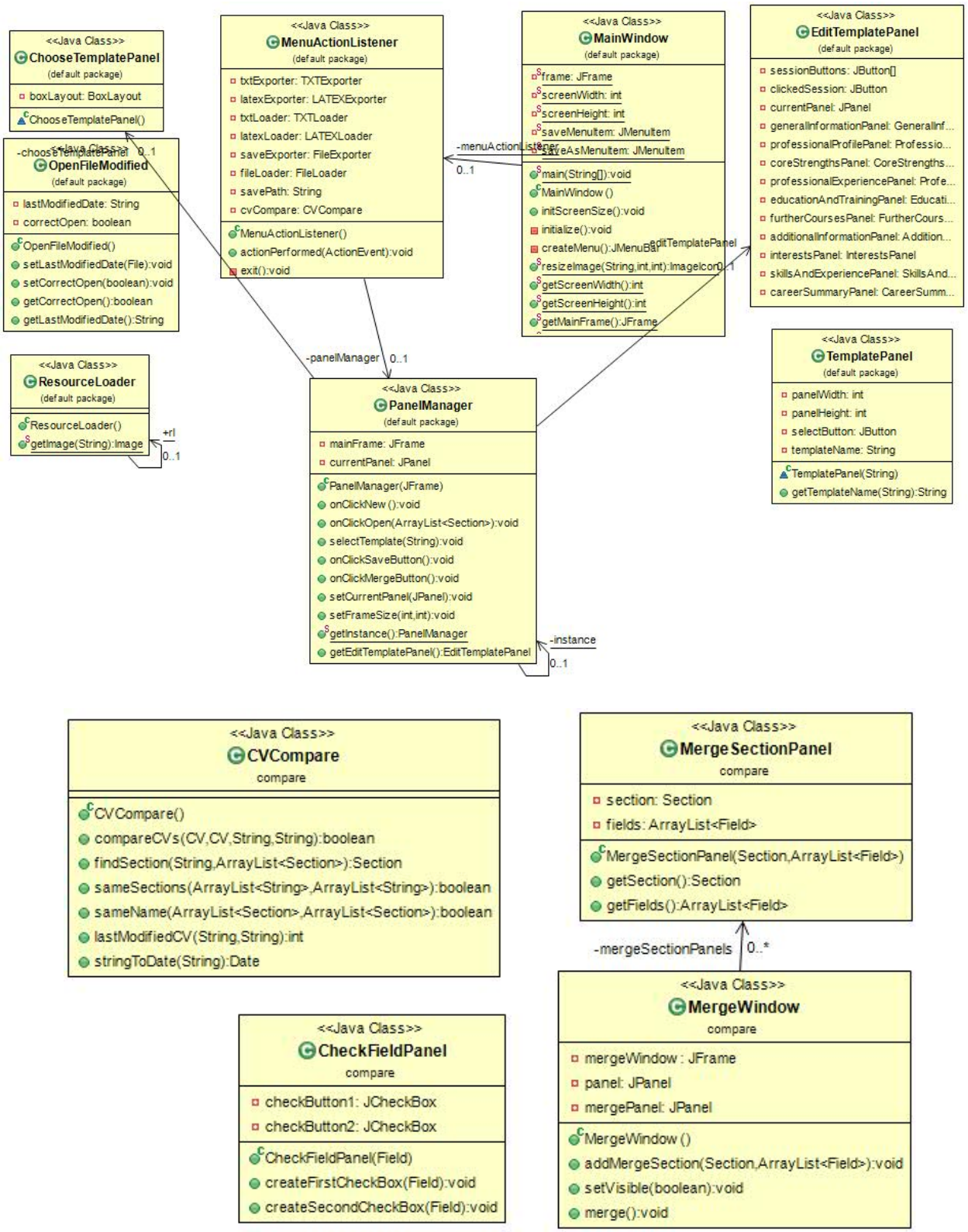
Expected result	*False*
Test Data	*"loadFiles\\wrong_end_block.ltx"*
Expected result	*False*
Test Data	*"loadFiles\\wrong_start_block.ltx"*
Expected result	*False*

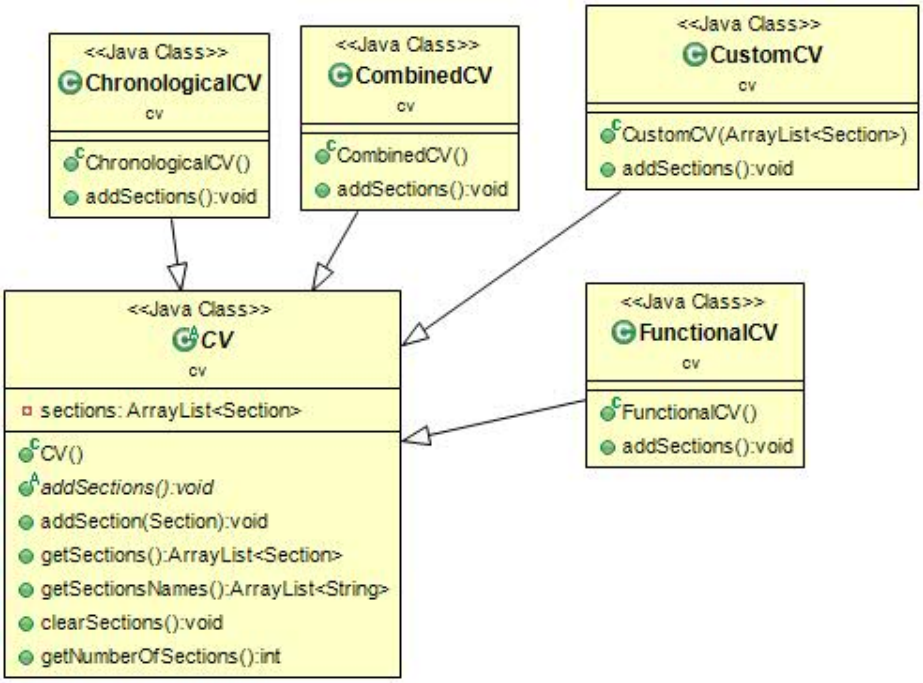
3 Design

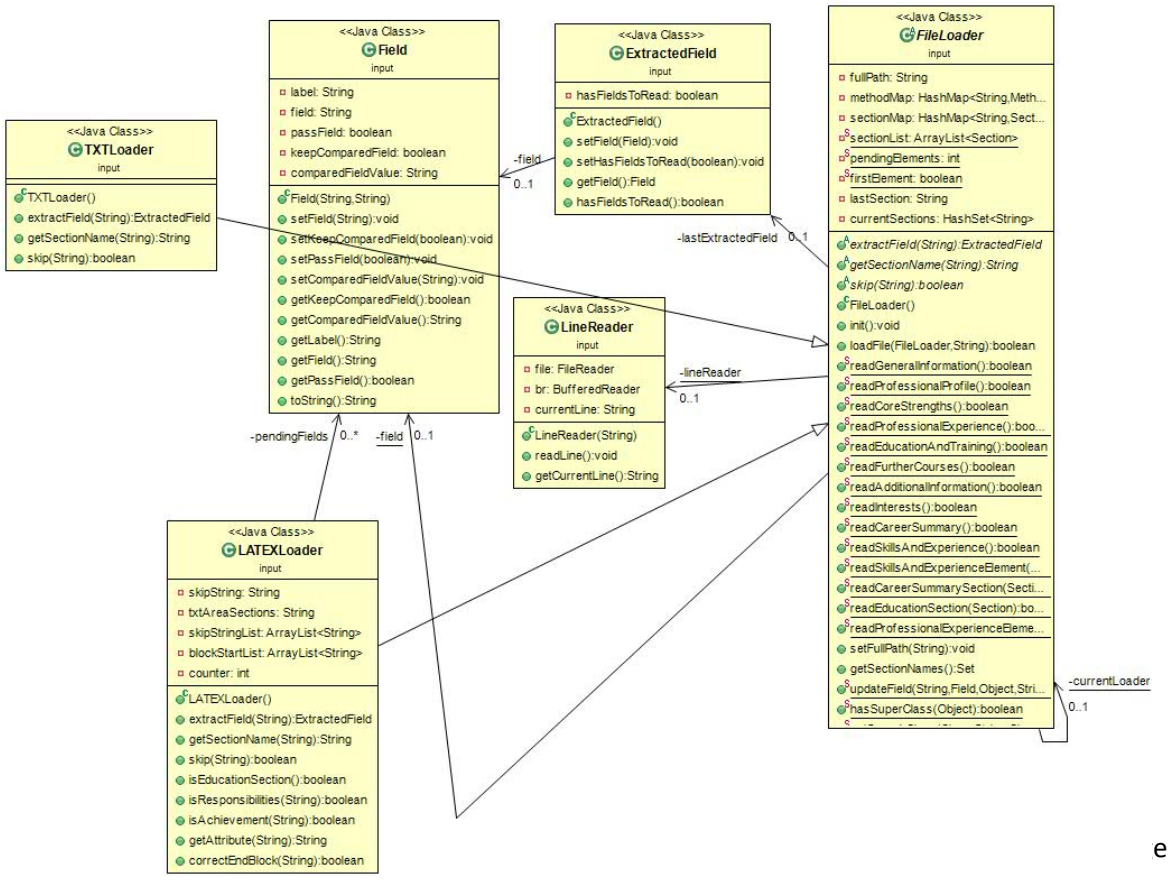
3.1 Architecture

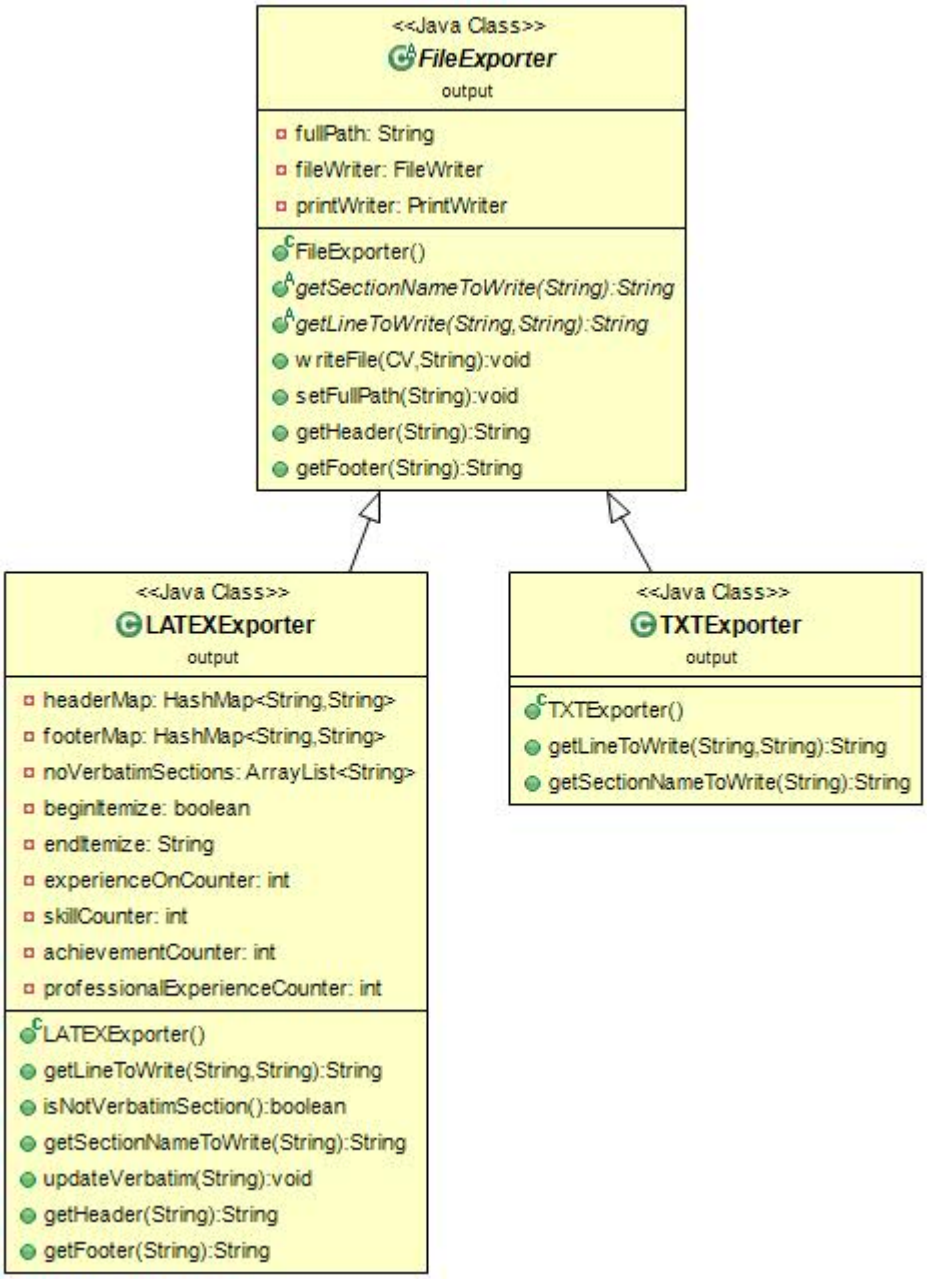


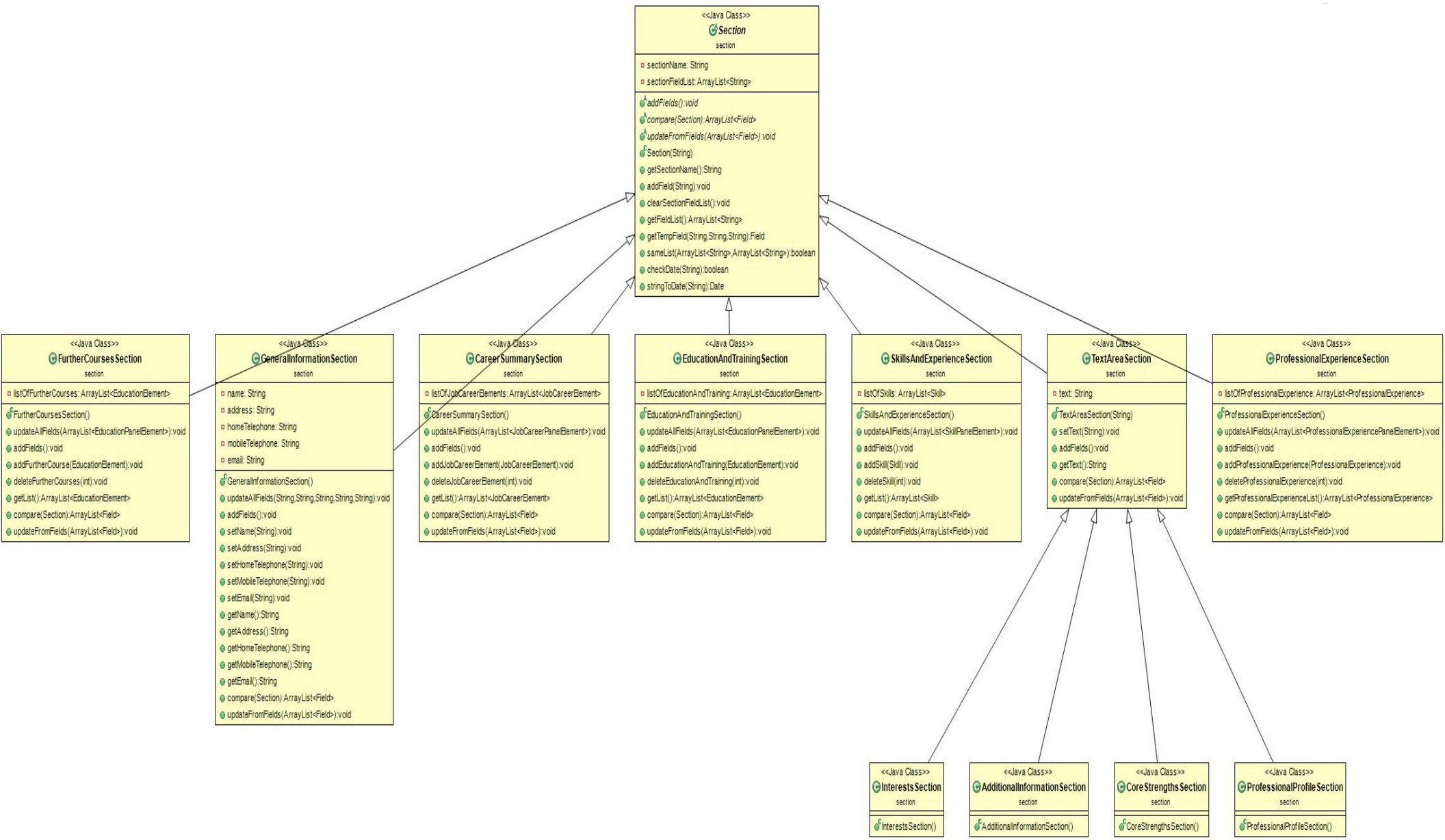
3.2 Design

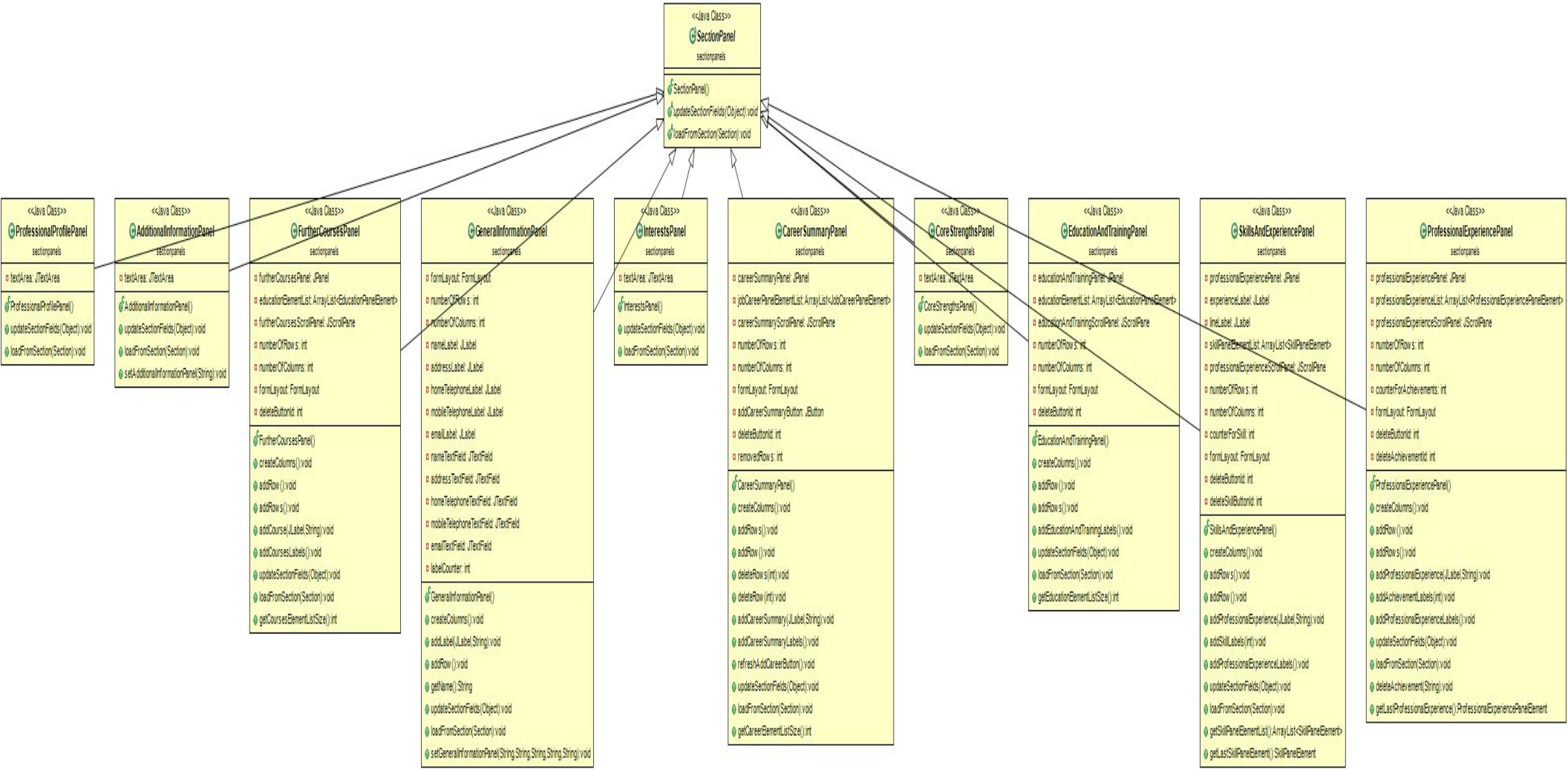

<<Java Class>>
ChooseTemplatePanel
(default package)
boxLayout: BoxLayout
ChooseTemplatePanel()
<<Java Class>>
OpenFileModified
(default package)
lastModifiedDate: String
correctOpen: boolean
OpenFileModified()
setLastModifiedDate(File):void
setCorrectOpen(boolean):void
getCorrectOpen():boolean
getLastModifiedDate():String
-chooseTemplatePanel
0..1
<<Java Class>>
MenuActionListener
(default package)
txtExporter: TXTExporter
latexExporter: LATEXExporter
txtLoader: TXTLoader
latexLoader: LATEXLoader
saveExporter: FileExporter
fileLoader: FileLoader
savePath: String
cvCompare: CVCompare
MenuActionListener()
actionPerformed(ActionEvent):void
exit():void
-menuActionListener
0..1
<<Java Class>>
MainWindow
(default package)
frame: JFrame
screenWidth: int
screenHeight: int
saveMenuItem: JMenuItem
saveAsMenuItem: JMenuItem
main(String[]):void
MainWindow()
initScreenSize():void
initialize():void
createMenu():JMenuBar
resizeImage(String,int,int):ImageIcon
getScreenWidth():int
getScreenHeight():int
getMainFrame():JFrame
editTemplatePanel
0..1
<<Java Class>>
EditTemplatePanel
(default package)
sessionButtons: JButton[]
clickedSession: JButton
currentPanel: JPanel
generalInformationPanel: GeneralInf...
professionalProfilePanel: Professio...
coreStrengthsPanel: CoreStrengths...
professionalExperiencePanel: Profe...
educationAndTrainingPanel: Educati...
furtherCoursesPanel: FurtherCours...
additionalInformationPanel: Addition...
interestsPanel: InterestsPanel
skillsAndExperiencePanel: SkillsAnd...
careerSummaryPanel: CareerSumm...
<<Java Class>>
ResourceLoader
(default package)
ResourceLoader()
getImage(String):Image
+rl
0..1
-panelManager
0..1
<<Java Class>>
PanelManager
(default package)
mainFrame: JFrame
currentPanel: JPanel
PanelManager(JFrame)
onClickNew():void
onClickOpen(ArrayList<Section>):void
selectTemplate(String):void
onClickSaveButton():void
onClickMergeButton():void
setCurrentPanel(JPanel):void
setFrameSize(int,int):void
getInstance():PanelManager
getEditTemplatePanel():EditTemplatePanel
-instance
0..1
<<Java Class>>
TemplatePanel
(default package)
panelWidth: int
panelHeight: int
selectButton: JButton
templateName: String
TemplatePanel(String)
getTemplateName(String):String
<<Java Class>>
CVCompare
compare
CVCompare()
compareCVs(CV,CV,String,String):boolean
findSection(String,ArrayList<Section>):Section
sameSections(ArrayList<String>,ArrayList<String>):boolean
sameName(ArrayList<Section>,ArrayList<Section>):boolean
lastModifiedCV(String,String):int
stringToDate(String):Date
<<Java Class>>
MergeSectionPanel
compare
section: Section
fields: ArrayList<Field>
MergeSectionPanel(Section,ArrayList<Field>)
getSection():Section
getFields():ArrayList<Field>
-mergeSectionPanels
0..*
<<Java Class>>
MergeWindow
compare
mergeWindow: JFrame
panel: JPanel
mergePanel: JPanel
MergeWindow()
addMergeSection(Section,ArrayList<Field>):void
setVisible(boolean):void
merge():void
<<Java Class>>
CheckFieldPanel
compare
checkButton1: JCheckBox
checkButton2: JCheckBox
CheckFieldPanel(Field)
createFirstCheckBox(Field):void
createSecondCheckBox(Field):void

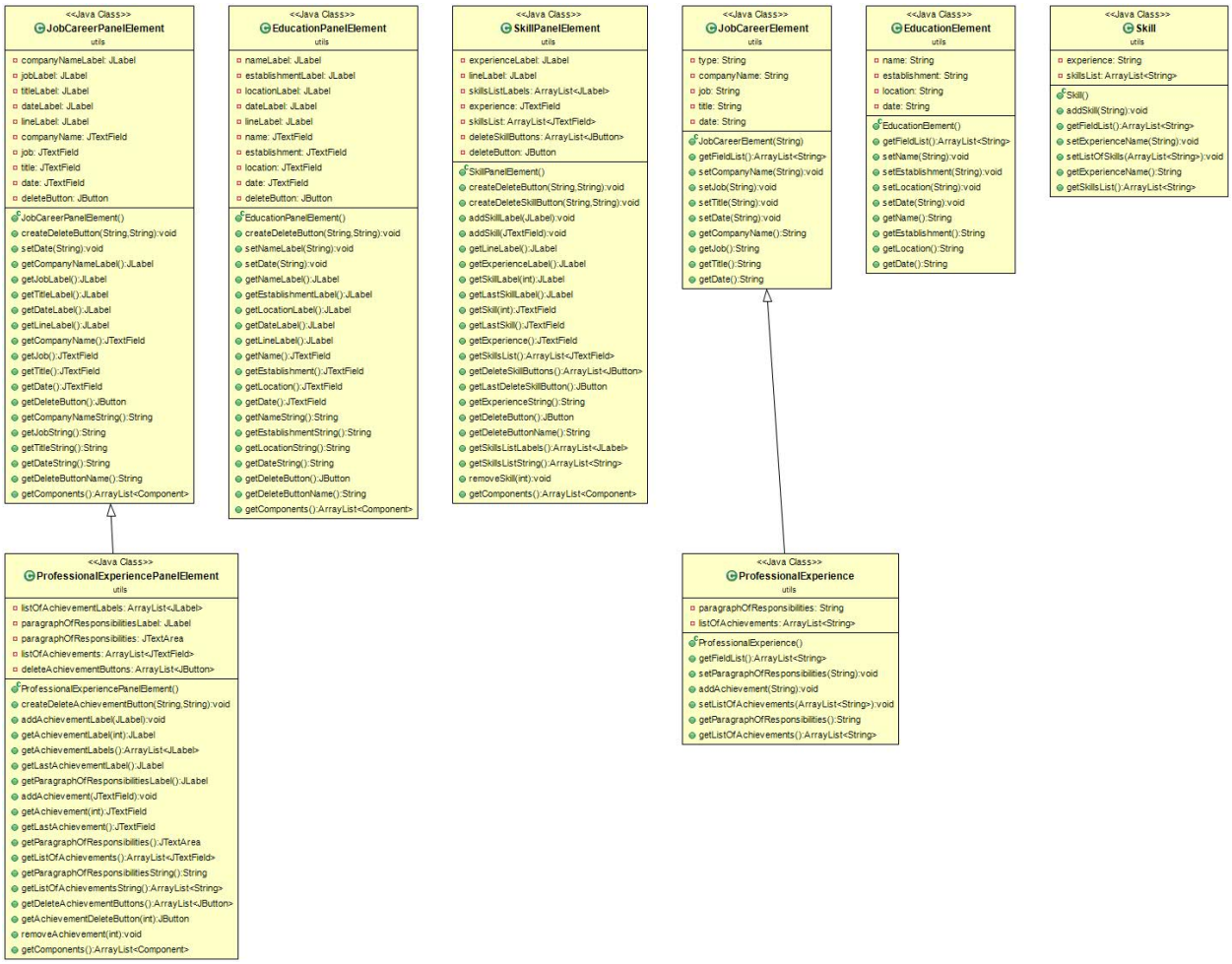

<<Java Class>>
ChronologicalCV
cv
ChronologicalCV()
addSections():void
<<Java Class>>
CombinedCV
cv
CombinedCV()
addSections():void
<<Java Class>>
CustomCV
cv
CustomCV(ArrayList<Section>)
addSections():void
<<Java Class>>
CV
cv
sections: ArrayList<Section>
CV()
addSections():void
addSection(Section):void
getSections():ArrayList<Section>
getSectionsNames():ArrayList<String>
clearSections():void
getNumberOfSections():int
<<Java Class>>
FunctionalCV
cv
FunctionalCV()
addSections():void


<<Java Class>>
TXTLoader
input
TXTLoader()
extractField(String):ExtractedField
getSectionName(String):String
skip(String):boolean
<<Java Class>>
Field
input
label: String
field: String
passField: boolean
keepComparedField: boolean
comparedFieldValue: String
Field(String,String)
setField(String):void
setKeepComparedField(boolean):void
setPassField(boolean):void
setComparedFieldValue(String):void
getKeepComparedField():boolean
getComparedFieldValue():String
getLabel():String
getField():String
getPassField():boolean
toString():String
-field
0..1
<<Java Class>>
ExtractedField
input
hasFieldsToRead: boolean
ExtractedField()
setField(Field):void
setHasFieldsToRead(boolean):void
getField():Field
hasFieldsToRead():boolean
-lastExtractedField
0..1
<<Java Class>>
FileLoader
input
fullPath: String
methodMap: HashMap<String,Meth...
sectionMap: HashMap<String,Sect...
sectionList: ArrayList<Section>
pendingElements: int
firstElement: boolean
lastSection: String
currentSections: HashSet<String>
extractField(String):ExtractedField
getSectionName(String):String
skip(String):boolean
FileLoader()
init():void
loadFile(FileLoader,String):boolean
readGeneralInformation():boolean
readProfessionalProfile():boolean
readCoreStrengths():boolean
readProfessionalExperience():boo...
readEducationAndTraining():boolean
readFurtherCourses():boolean
readAdditionalInformation():boolean
readInterests():boolean
readCareerSummary():boolean
readSkillsAndExperience():boolean
readSkillsAndExperienceElement(...
readCareerSummarySection(Secti...
readEducationSection(Section):bo...
readProfessionalExperienceEleme...
setFullPath(String):void
getSectionNames():Set
updateField(String,Field,Object,Stri...
hasSuperClass(Object):boolean
-currentLoader
0..1
<<Java Class>>
LineReader
input
file: FileReader
br: BufferedReader
currentLine: String
LineReader(String)
readLine():void
getCurrentLine():String
-lineReader
0..1
-pendingFields
0..*
-field
0..1
<<Java Class>>
LATEXLoader
input
skipString: String
txtAreaSections: String
skipStringList: ArrayList<String>
blockStartList: ArrayList<String>
counter: int
LATEXLoader()
extractField(String):ExtractedField
getSectionName(String):String
skip(String):boolean
isEducationSection():boolean
isResponsibilities(String):boolean
isAchievement(String):boolean
getAttribute(String):String
correctEndBlock(String):boolean


<<Java Class>>
FileExporter
output
fullPath: String
fileWriter: FileWriter
printWriter: PrintWriter
FileExporter()
getSectionNameToWrite(String):String
getLineToWrite(String,String):String
writeFile(CV,String):void
setFullPath(String):void
getHeader(String):String
getFooter(String):String
<<Java Class>>
LATEXExporter
output
headerMap: HashMap<String,String>
footerMap: HashMap<String,String>
noVerbatimSections: ArrayList<String>
beginItemize: boolean
endItemize: String
experienceOnCounter: int
skillCounter: int
achievementCounter: int
professionalExperienceCounter: int
LATEXExporter()
getLineToWrite(String,String):String
isNotVerbatimSection():boolean
getSectionNameToWrite(String):String
updateVerbatim(String):void
getHeader(String):String
getFooter(String):String
<<Java Class>>
TXTExporter
output
TXTExporter()
getLineToWrite(String,String):String
getSectionNameToWrite(String):String


<<Java Class>>
Section
section
sectionName: String
sectionFieldList: ArrayList<String>
addFields():void
compare(Section):ArrayList<Field>
updateFromFields(ArrayList<Field>):void
Section(String)
getSectionName():String
addField(String):void
clearSectionFieldList():void
getFieldList():ArrayList<String>
getTempField(String,String,String):Field
sameList(ArrayList<String>,ArrayList<String>):boolean
checkDate(String):boolean
stringToDate(String):Date
<<Java Class>>
FurtherCoursesSection
section
listOfFurtherCourses: ArrayList<EducationElement>
FurtherCoursesSection()
updateAllFields(ArrayList<EducationPanelElement>):void
addFields():void
addFurtherCourse(EducationElement):void
deleteFurtherCourses(int):void
getList():ArrayList<EducationElement>
compare(Section):ArrayList<Field>
updateFromFields(ArrayList<Field>):void
<<Java Class>>
GeneralInformationSection
section
name: String
address: String
homeTelephone: String
mobileTelephone: String
email: String
GeneralInformationSection()
updateAllFields(String,String,String,String,String):void
addFields():void
setName(String):void
setAddress(String):void
setHomeTelephone(String):void
setMobileTelephone(String):void
setEmail(String):void
getName():String
getAddress():String
getHomeTelephone():String
getMobileTelephone():String
getEmail():String
compare(Section):ArrayList<Field>
updateFromFields(ArrayList<Field>):void
<<Java Class>>
CareerSummarySection
section
listOfJobCareerElements: ArrayList<JobCareerElement>
CareerSummarySection()
updateAllFields(ArrayList<JobCareerPanelElement>):void
addFields():void
addJobCareerElement(JobCareerElement):void
deleteJobCareerElement(int):void
getList():ArrayList<JobCareerElement>
compare(Section):ArrayList<Field>
updateFromFields(ArrayList<Field>):void
<<Java Class>>
EducationAndTrainingSection
section
listOfEducationAndTraining: ArrayList<EducationElement>
EducationAndTrainingSection()
updateAllFields(ArrayList<EducationPanelElement>):void
addFields():void
addEducationAndTraining(EducationElement):void
deleteEducationAndTraining(int):void
getList():ArrayList<EducationElement>
compare(Section):ArrayList<Field>
updateFromFields(ArrayList<Field>):void
<<Java Class>>
SkillsAndExperienceSection
section
listOfSkills: ArrayList<Skill>
SkillsAndExperienceSection()
updateAllFields(ArrayList<SkillPanelElement>):void
addFields():void
addSkill(Skill):void
deleteSkill(int):void
getList():ArrayList<Skill>
compare(Section):ArrayList<Field>
updateFromFields(ArrayList<Field>):void
<<Java Class>>
TextAreaSection
section
text: String
TextAreaSection(String)
setText(String):void
addFields():void
getText():String
compare(Section):ArrayList<Field>
updateFromFields(ArrayList<Field>):void
<<Java Class>>
ProfessionalExperienceSection
section
listOfProfessionalExperience: ArrayList<ProfessionalExperience>
ProfessionalExperienceSection()
updateAllFields(ArrayList<ProfessionalExperiencePanelElement>):void
addFields():void
addProfessionalExperience(ProfessionalExperience):void
deleteProfessionalExperience(int):void
getProfessionalExperienceList():ArrayList<ProfessionalExperience>
compare(Section):ArrayList<Field>
updateFromFields(ArrayList<Field>):void
<<Java Class>>
InterestsSection
section
InterestsSection()
<<Java Class>>
AdditionalInformationSection
section
AdditionalInformationSection()
<<Java Class>>
CoreStrengthsSection
section
CoreStrengthsSection()
<<Java Class>>
ProfessionalProfileSection
section
ProfessionalProfileSection()


«Java Class»
SectionPanel
sectionpanels
SectionPanel()
updateSectionFields(Object):void
loadFromSection(Section):void
«Java Class»
ProfessionalProfilePanel
sectionpanels
textArea: JTextArea
ProfessionalProfilePanel()
updateSectionFields(Object):void
loadFromSection(Section):void
«Java Class»
AdditionalInformationPanel
sectionpanels
textArea: JTextArea
AdditionalInformationPanel()
updateSectionFields(Object):void
loadFromSection(Section):void
setAdditionalInformationPanel(String):void
«Java Class»
FurtherCoursesPanel
sectionpanels
furtherCoursesPanel: JPanel
educationElementList: ArrayList<EducationPanelElement>
furtherCoursesScrollPanel: JScrollPane
numberOfRows: int
numberOfColumns: int
formLayout: FormLayout
deleteButtonId: int
FurtherCoursesPanel()
createColumns():void
addRow():void
addRows():void
addCourse(JLabel,String):void
addCoursesLabels():void
updateSectionFields(Object):void
loadFromSection(Section):void
getCoursesElementListSize():int
«Java Class»
GeneralInformationPanel
sectionpanels
formLayout: FormLayout
numberOfRows: int
numberOfColumns: int
nameLabel: JLabel
addressLabel: JLabel
homeTelephoneLabel: JLabel
mobileTelephoneLabel: JLabel
emailLabel: JLabel
nameTextField: JTextField
addressTextField: JTextField
homeTelephoneTextField: JTextField
mobileTelephoneTextField: JTextField
emailTextField: JTextField
labelCounter: int
GeneralInformationPanel()
createColumns():void
addLabel(JLabel,String):void
addRow():void
getName():String
updateSectionFields(Object):void
loadFromSection(Section):void
setGeneralInformationPanel(String,String,String,String,String):void
«Java Class»
InterestsPanel
sectionpanels
textArea: JTextArea
InterestsPanel()
updateSectionFields(Object):void
loadFromSection(Section):void
«Java Class»
CareerSummaryPanel
sectionpanels
careerSummaryPanel: JPanel
jobCareerPanelElementList: ArrayList<JobCareerPanelElement>
careerSummaryScrollPanel: JScrollPane
numberOfRows: int
numberOfColumns: int
formLayout: FormLayout
addCareerSummaryButton: JButton
deleteButtonId: int
removedRow: int
CareerSummaryPanel()
createColumns():void
addRows():void
addRow():void
deleteRows(int):void
deleteRow(int):void
addCareerSummary(JLabel,String):void
addCareerSummaryLabels():void
refreshAddCareerButton():void
updateSectionFields(Object):void
loadFromSection(Section):void
getCareerElementListSize():int
«Java Class»
CoreStrengthsPanel
sectionpanels
textArea: JTextArea
CoreStrengthsPanel()
updateSectionFields(Object):void
loadFromSection(Section):void
«Java Class»
EducationAndTrainingPanel
sectionpanels
educationAndTrainingPanel: JPanel
educationElementList: ArrayList<EducationPanelElement>
educationAndTrainingScrollPanel: JScrollPane
numberOfRows: int
numberOfColumns: int
formLayout: FormLayout
deleteButtonId: int
EducationAndTrainingPanel()
createColumns():void
addRow():void
addRows():void
addEducationAndTrainingLabels():void
updateSectionFields(Object):void
loadFromSection(Section):void
getEducationElementListSize():int
«Java Class»
SkillsAndExperiencePanel
sectionpanels
professionalExperiencePanel: JPanel
experienceLabel: JLabel
lineLabel: JLabel
skillPanelElementList: ArrayList<SkillPanelElement>
professionalExperienceScrollPanel: JScrollPane
numberOfRows: int
numberOfColumns: int
counterForSkill: int
formLayout: FormLayout
deleteButtonId: int
deleteSkillButtonId: int
SkillsAndExperiencePanel()
createColumns():void
addRows():void
addRow():void
addProfessionalExperience(JLabel,String):void
addSkillLabels(int):void
addProfessionalExperienceLabels():void
updateSectionFields(Object):void
loadFromSection(Section):void
getSkillPanelElementList():ArrayList<SkillPanelElement>
getLastSkillPanelElement():SkillPanelElement
«Java Class»
ProfessionalExperiencePanel
sectionpanels
professionalExperiencePanel: JPanel
professionalExperienceList: ArrayList<ProfessionalExperiencePanelElement>
professionalExperienceScrollPanel: JScrollPane
numberOfRows: int
numberOfColumns: int
counterForAchievements: int
formLayout: FormLayout
deleteButtonId: int
deleteAchievementId: int
ProfessionalExperiencePanel()
createColumns():void
addRow():void
addRows():void
addProfessionalExperience(JLabel,String):void
addAchievementLabels(int):void
addProfessionalExperienceLabels():void
updateSectionFields(Object):void
loadFromSection(Section):void
deleteAchievement(String):void
getLastProfessionalExperience():ProfessionalExperiencePanelElement



Class Name: MainWindow	
Responsibilities: ▪ Initializes main window ▪ Creates menu bar	**Collaborations:** ▪ MenuActionListener

Class Name: MenuActionListener	
Responsibilities: ▪ Manages menu items' click events via listener	**Collaborations:** ▪ PanelManager

Class Name: PanelManager

Responsibilities:	**Collaborations:**
▪ Implements menu items' click events	▪ ChooseTemplatePanel ▪ EditTemplatePanel

Class Name: ChooseTemplatePanel	
Responsibilities:	**Collaborations:**
▪ Creates the interface where user chooses the template that he wants to create	▪ TemplatePanel

Class Name: TemplatePanel	
Responsibilities:	**Collaborations: None**
▪ Creates a panel that represents the template which the user will select via the **ChooseTemplatePanel**	

Class Name: EditTemplatePanel	
Responsibilities:	**Collaborations:**
▪ Implements the interface with which the user can edit the CV template that he chose	▪ GeneralInformationPanel ▪ ProfessionalProfilePanel ▪ CoreStrengthsPanel ▪ ProfessionalExperiencePanel ▪ EducationAndTrainingPanel ▪ FurtherCoursesPanel ▪ AdditionalInformationPanel ▪ InterestsPanel ▪ SkillsAndExperiencePanel ▪ CareerSumaryPanel ▪ ChronologicalCV ▪ FunctionalCV

Class Name: CV	
Responsibilities:	**Collaborations: None**

▪ Represents the sessions of a CV	

Class Name: FunctionalCV, **ChronologicalCV**	
Responsibilities: ▪ Fills the main CV template with it' s own sessions	**Collaborations:** ▪ CV

Class Name: GeneralInformationPanel, **ProfessionalProfilePanel**, **CoreStrengthsPanel**, **ProfessionalExperiencePanel**, **EducationAndTrainingPanel**, **FurtherCoursesPanel**, **AdditionalInformationPanel**, **InterestsPanel**, **SkillsAndExperiencePanel**, **CareerSumaryPanel**	
Responsibilities: ▪ Creates a panel based on the corresponding session. The user can edit the content of this panel	**Collaborations: None**

Class Name: CheckFieldPanel	
Responsibilities: ▪ Control check-boxes from panel	**Collaborations: None**

Class Name: CVCompare	
Responsibilities: ▪ Compare the sections from the selected files	**Collaborations: None**

Class Name: MergeSectionPanel	
Responsibilities: ▪ Create merge panel	**Collaborations:** ▪ Section

Class Name: MergeWindow	
Responsibilities: ▪ Create the merge window ▪	**Collaborations:** ▪ MergeSectionPanel

Class Name: ExtractedField	
Responsibilities: ▪ Read the fields from the file	**Collaborations:** ▪ Field

Class Name: Field	
Responsibilities: ▪ Take the label and textfield from the field	**Collaborations: None**

Class Name: FileLoader	
Responsibilities: ▪ Read the file ▪ For each section call the correct read method	**Collaborations:** ▪ Section ▪ LineReader ▪ FileLoader ▪ Field ▪ ExtractedField

Class Name: TXTLoader	
Responsibilities: ▪ Take the useful text from a txt file	**Collaborations:** ▪ FileLoader

Class Name: LATEXLoader	
Responsibilities: ▪ Read the userful text from a latex file	**Collaborations:** ▪ FileLoader ▪ Field

Class Name: LineReader	
Responsibilities: ▪ Read the file line by line	**Collaborations: None**

Class Name: FileExporter

Responsibilities:	Collaborations: None
▪ Write the file	

Class Name: TXTExporter	
Responsibilities:	**Collaborations:**
▪ Write the txt file	▪ FileExporter

Class Name: LATEXExporter	
Responsibilities:	**Collaborations:**
▪ Write the latex file	▪ FileExporter

Class Name: AdditionalInformationSection, CareerSummarySection, CoreStrengthsSection, EducationAndTrainingSection, FurtherCourses, GeneralInformation, InterestsSection, ProfessionalExperienceSection, ProfessionalProfile, Section, SkillsAndExperience, TextAreaSection	
Responsibilities:	**Collaborations:**
▪ The “back-end” from all sections. If something change in GUI change and the back data too	▪ Section

Class Name: Skill	
Responsibilities:	**Collaborations: None**
▪ Control the skills, edit them, add new skills	

Class Name: SkillPanelElement	
Responsibilities:	**Collaborations: None**
▪ Controls labels and textfields for Skills and Experience panel	

Class Name: ProfessionalExperiencePanelElement	
Responsibilities:	**Collaborations:**
▪ Controls labels and textfields for	▪ JobCareerPanelElement

Professional Experience panel	

Class Name: ProfessionalExperience	
Responsibilities: ▪ Controls responsibilities and list of achievements	**Collaborations:** ▪ JobCareerElement

Class Name: JobCareerPanelElement	
Responsibilities: ▪ Create labels and textfields for job and career panel	**Collaborations: None**

Class Name: JobCareerElement	
Responsibilities: ▪ Controls from “back end”labels and textfields for job and career panel	**Collaborations: None**

Class Name: EducationElement	
Responsibilities: ▪ Controls from “back end”labels and textfields for education panel	**Collaborations: None**

Class Name: EducationPanelElement	
Responsibilities: ▪ Controls labels and textfields for education panel	**Collaborations: None**

4 Implementation