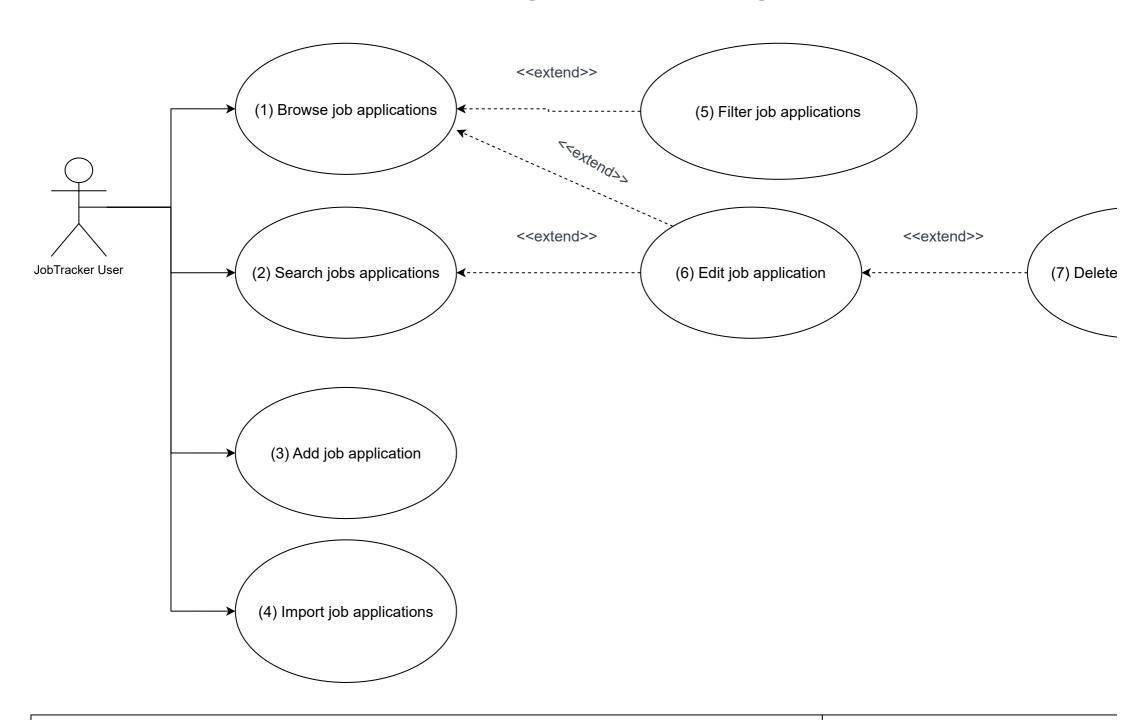
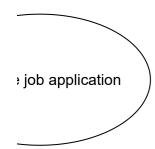
JobTracker Program Use Case Diagram



UC: Bı	UC: Sea		
<u>.</u> .	The user can look and peruse through all of their iob		1

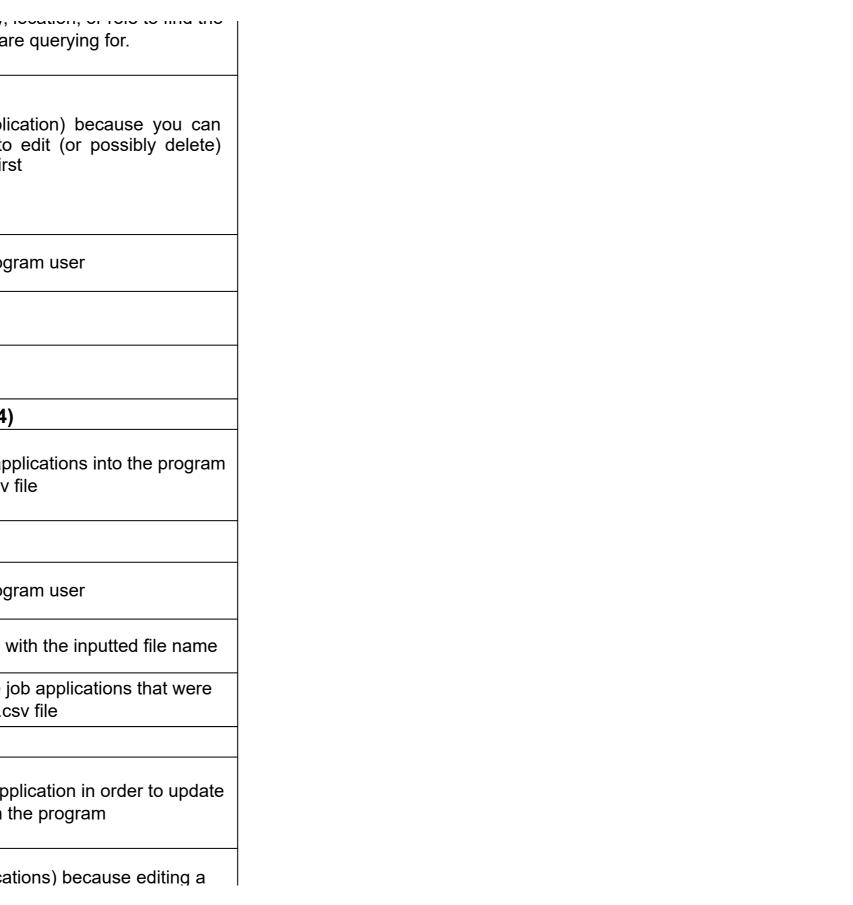


rch job applications (Ref # 2)

The user can search by company. location. or role to find the

applications within the tracker program	Overview		
- Extended by (5. Filter job applications) since the user can filter all of the job applications within the browse by a desired status - Extended by (6. Edit job application) because you can choose the job that you want to edit (or possibly delete) only after looking through them first	Related use cases (if any):		
"JobTracker" program user	Actors		
	Pre Conditions (Optional)		
	Post Conditions (Optional)		
UC: Add job application (Ref # 3)			
The user can add a new job application to the program in order to track it for the future	Overview		
	Related use cases (if any):		
"JobTracker" program user	Actors		
	Pre Conditions (Optional)		
Job tracker now contains the new job application to track with the inputted information	Post Conditions (Optional)		
UC: Filter job applications (Ref # 5)			
The user can filter out the job applications with a desired status out of all the applications tracked with the system	Overview		
	- Extended by (5. Filter job applications) since the user can filter all of the job applications within the browse by a desired status - Extended by (6. Edit job application) because you can choose the job that you want to edit (or possibly delete) only after looking through them first "JobTracker" program user Add job application (Ref # 3) The user can add a new job application to the program in order to track it for the future "JobTracker" program user "JobTracker" program user "Job tracker now contains the new job application to track with the inputted information Filter job applications (Ref # 5) The user can filter out the job applications with a desired status	applications within the tracker program - Extended by (5. Filter job applications) since the user can filter all of the job applications within the browse by a desired status - Extended by (6. Edit job application) because you can choose the job that you want to edit (or possibly delete) only after looking through them first "JobTracker" program user Actors Pre Conditions (Optional) Add job application (Ref # 3) The user can add a new job application to the program in order to track it for the future "JobTracker" program user Related use cases (if any): Overview Related use cases (if any): Pre Conditions (Optional) Actors Pre Conditions (Optional) Post Conditions (Optional) Filter job applications (Ref # 5) UC The user can filter out the job applications with a desired status	

our ocuron by company, recurer, or role to mia the job applications they are querying for. - Extended by (6. Edit job application) because you can choose the job that you want to edit (or possibly delete) only after looking through them first "JobTracker" program user ort job applications (Ref # 4) The user can import a list of job applications into the program from a .csv file "JobTracker" program user The directory contains a .csv file with the inputted file name Job tracker now contains all the job applications that were listed in the .csv file dit job application (Ref # 6) The user can edit a tracked job application in order to update its information with the program - Extends (2. Search job applications) because editing a



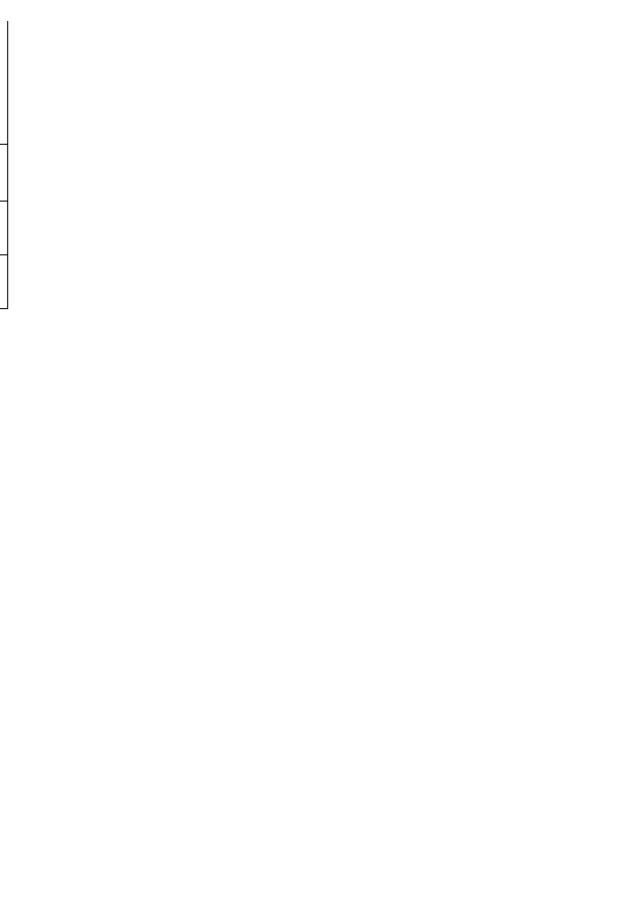
Related use cases (if any):	elated use cases (if any): - Extends (1. Browse job applications) because a user car only filter out their desired applications if they are first browsing them				
Actors	"JobTracker" program user	Actors			
Pre Conditions (Optional)		Pre Conditions (Optional)			
Post Conditions (Optional)		Post Conditions (Optional)	S		
UC: [UC: Delete job application (Ref # 7)				
Overview	The user can choose to delete a tracked job application from the system				
Related use cases (if any):	- Extends (6. Edit job application) because the user only has the option to delete a tracked job application after they choose to edit it				
Actors	"JobTracker" program user				
Pre Conditions (Optional)					
Post Conditions (Optional) Selected tracked job application is now deleted from the system					

job application (or deleting it) becomes an option after only after looking through them first
- Extended by (7. Delete job application) since the user only has the option to delete a tracked job application

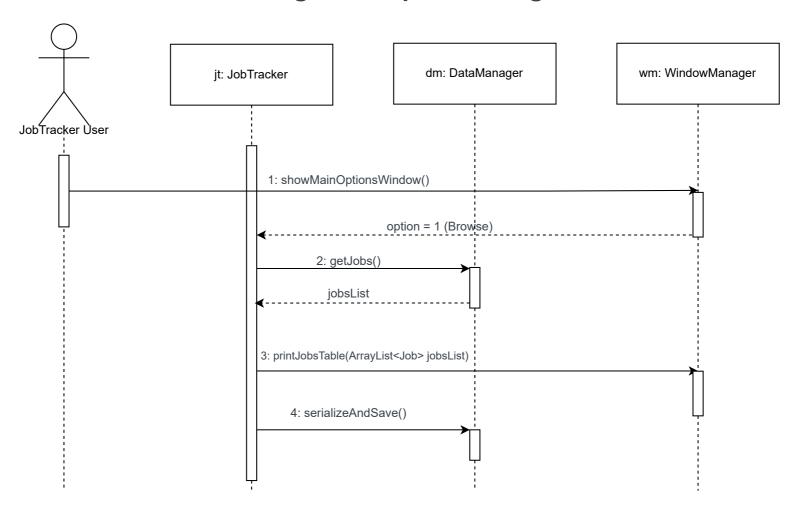
after they choose to edit it

"JobTracker" program user

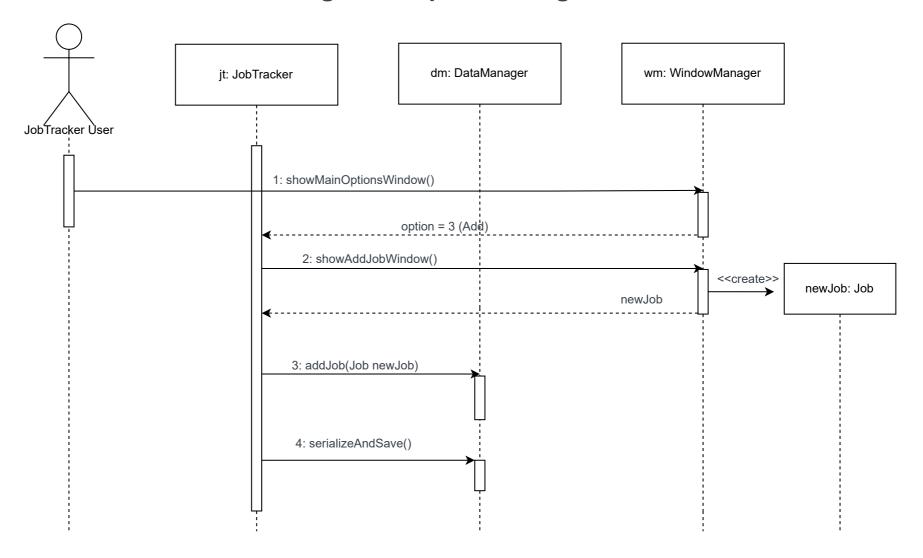
Selected tracked job application now has updated information



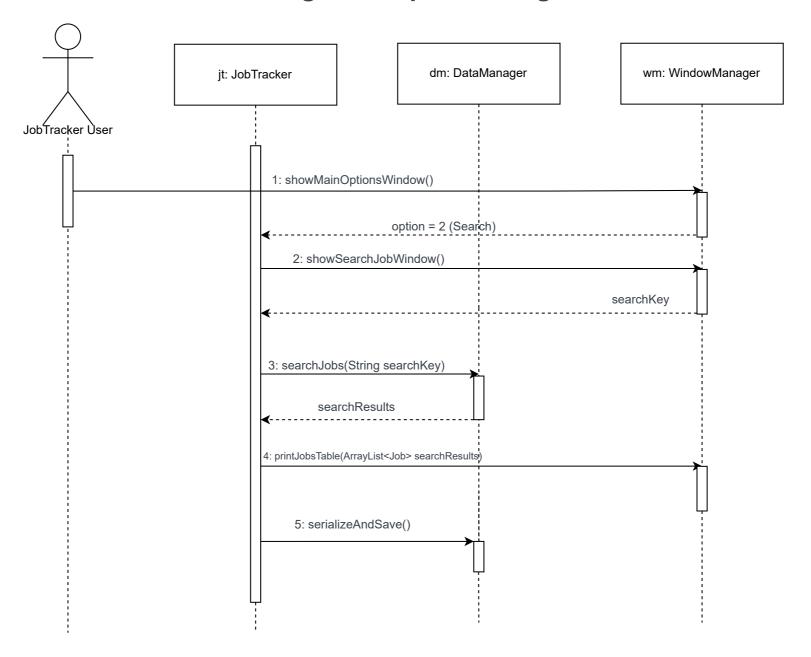
JobTracker Program Sequence Diagram: Browse



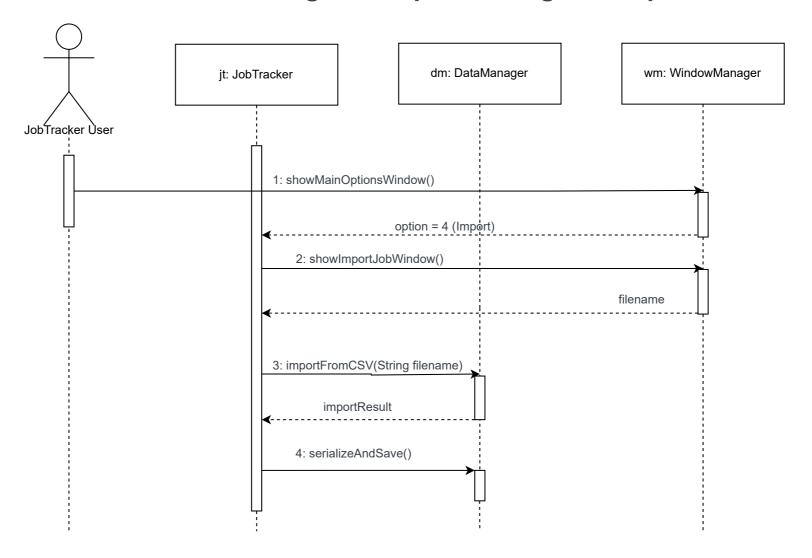
JobTracker Program Sequence Diagram: Add



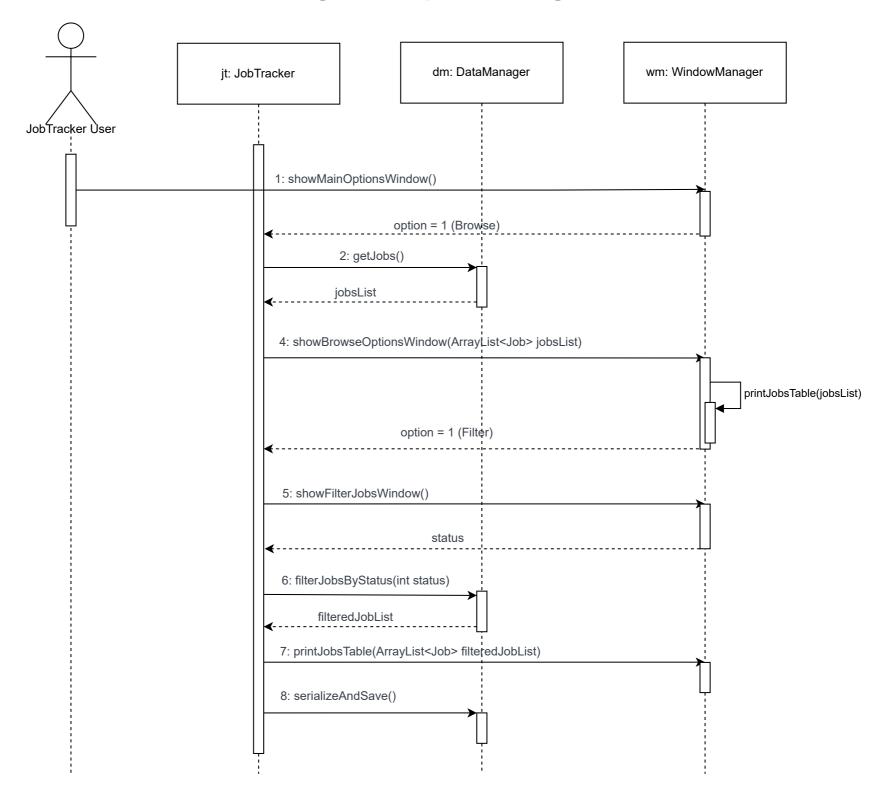
JobTracker Program Sequence Diagram: Search



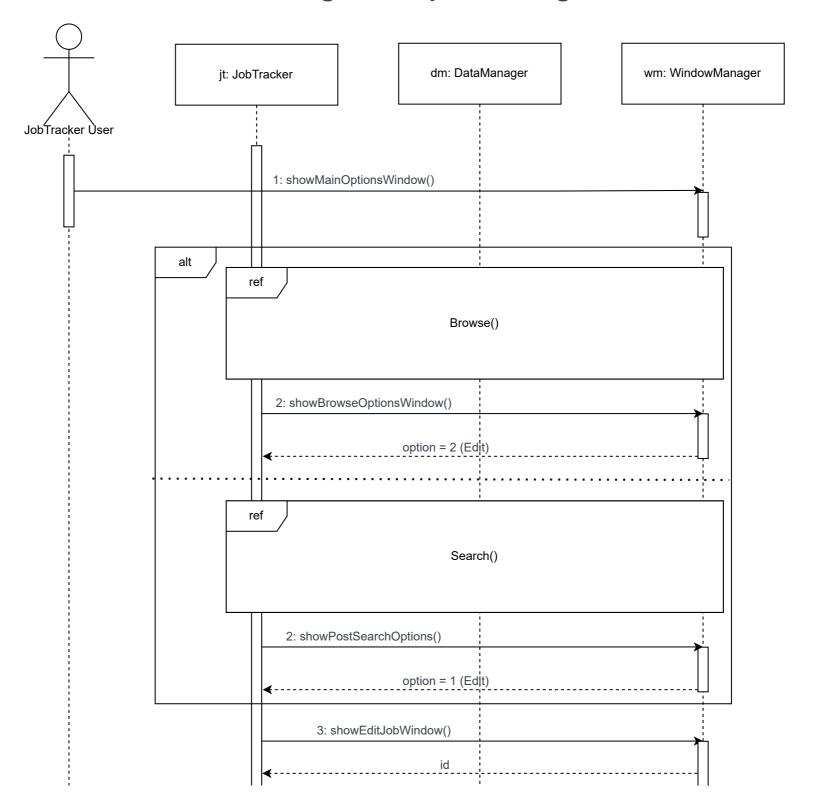
JobTracker Program Sequence Diagram: Import

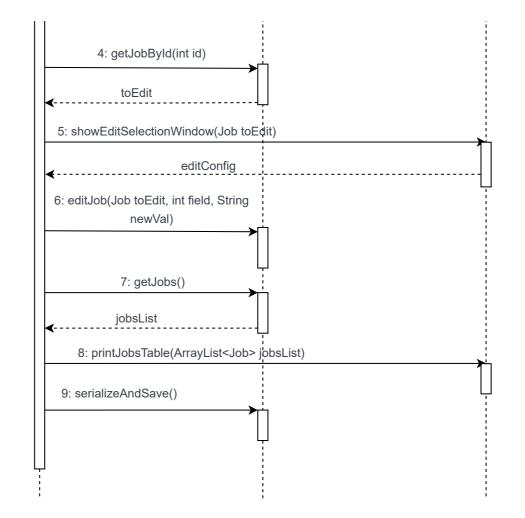


JobTracker Program Sequence Diagram: Filter

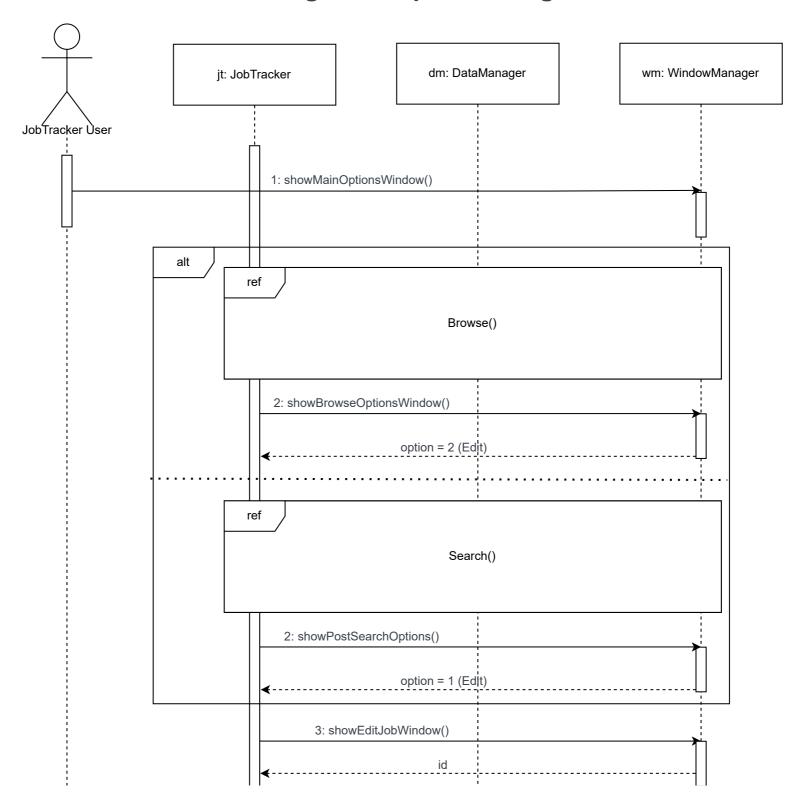


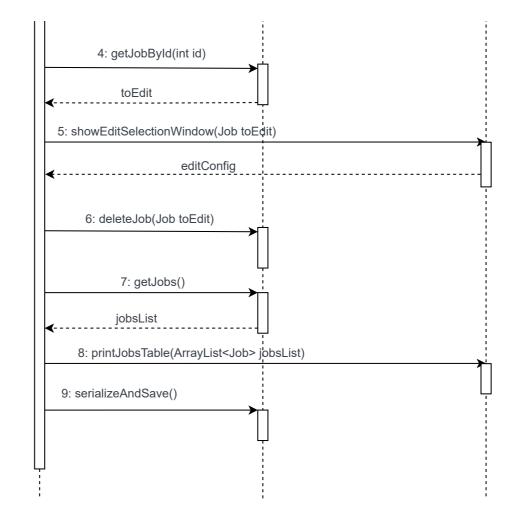
JobTracker Program Sequence Diagram: Edit





JobTracker Program Sequence Diagram: Delete





JobTracker Program Class Diagram

JobTracker

- instance: JobTracker

- JobTracker()
- + getInstance(): JobTracker
- + runApp(DataManager, WindowManager): void
- + validateOption(String, int): boolean

WindowManager

- instance: WindowManager
- scanner: Scanner
- WindowManager()
- + getInstance(): WindowManager
- + showMainOptionsWindow(): String
- + showBrowseOptionsWindow(ArrayList<Job>): String
- + showFilterJobsWindow(): String
- + printJobsTable(ArrayList<Job>): void
- + showEditJobWindow(): String
- + showEditSelectionWindow(Job): String[]
- + showSearchJobWindow(): String
- + showPostSearchOptions(): String
- + showAddJobWindow(): Job
- + showImportJobWindow(): String
- + clear(): void

DataManager

- instance: DataManager
- jobs: ArrayList<Job>
- DataManager()
- + getInstance(): DataManager
- + importFromCSV(String): boolean
- + addJob(Job): void
- + getJobs(): ArrayList<Job>
- + deleteJob(Job): void
- + editJob(Job, int, String): void
- + searchJobs(String): ArrayList<Job>
- + getJobByld(int): Job
- + filterJobByStatus(int): ArrayList<Job>
- + serializeAndSave(): void

Job

- lastIdUsed: int
- id: int
- company: String
- role: String
- salary: double location: String
- dateApplied: Date
- status: String
- linkToPosting: String
- + getLastId(): int
- + setLastId(int): void
- + getId(): int
- + getCompany(): String
- + setCompany(String): void
- + getRole(): String
- + setRole(String): void
- + getSalary(): double
- + setSalary(double): void
- + getLocation(): String

- + setLocation(String): void + getDateApplied(): Date + getDateAppliedFormatted(): String + setDateApplied(Date): void + getStatus(): String + setStatus(String): void + getLinkToPosting(): String + setLinkToPosting(String): void + toString(): void