

MINNESOTA INCOME TAX CALCULATION PROJECT

OVERALL REPORT

VERSION <1.0>

Βλαχοθανάσης Νικόλαος, 2652

Γεωργίου Βασίλειος, 2658

Πασόη Σοφία, 2798

TABLE OF CONTENTS

Introduction	1
Refactored Design	2
Architecture	2
Detailed Design	3
CRC Cards	10

INTRODUCTION

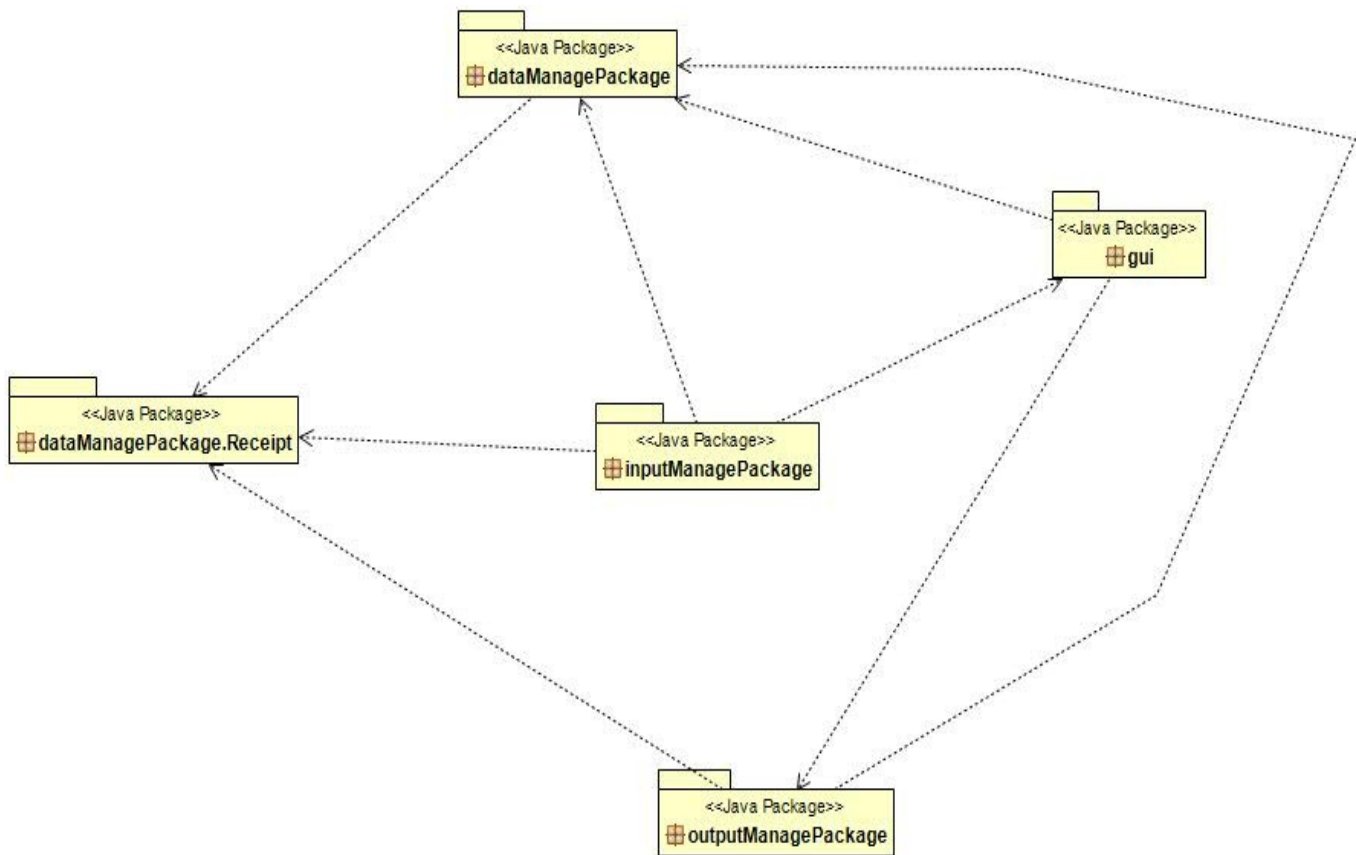
EVERYTHING THAT IS UNDERLINED IS A GUIDELINE ON HOW TO FILL THE REPORT - REMOVE ALL GUIDELINES FROM THE FINAL VERSION !!!

Η σχεδίαση και υλοποίηση ενός λογισμικού είναι απαραίτητο να γίνει με ορισμένα κριτήρια. Αυτό μας βοηθά να αποφύγουμε μια κακή σχεδίαση λογισμικού. Είναι πολύ σημαντικό να σχεδιάσουμε σωστά το λογισμικό, έτσι ώστε όταν έρθουν αλλαγές που αφορούν τις απαιτήσεις του συστήματος, αυτό θα πρέπει να είναι ευέλικτο και να μπορεί να τροποποιηθεί ανάλογα, με την ελάχιστη δυνατή προσπάθεια και τις μικρότερες δυνατές συνέπειες για τις υπόλοιπες λειτουργίες του.

Ο στόχος αυτού του έργου είναι να επανασχεδιάσει μια εφαρμογή Java παλαιού τύπου. Για να είμαστε σε θέση να πετύχουμε τον στόχο αυτό, πρέπει να έχουμε κατανοήσει σε βάθος τις αρχές μιας καλής σχεδίασης. Σκοπός της κατανόησης είναι να καταφέρουμε με τον επανασχεδιασμό του να προτείνουμε μια βελτιωμένη σχεδίαση, η οποία θα προσφέρει μελλοντική ευελιξία σε αλλαγές.

REFACTORED DESIGN

ARCHITECTURE

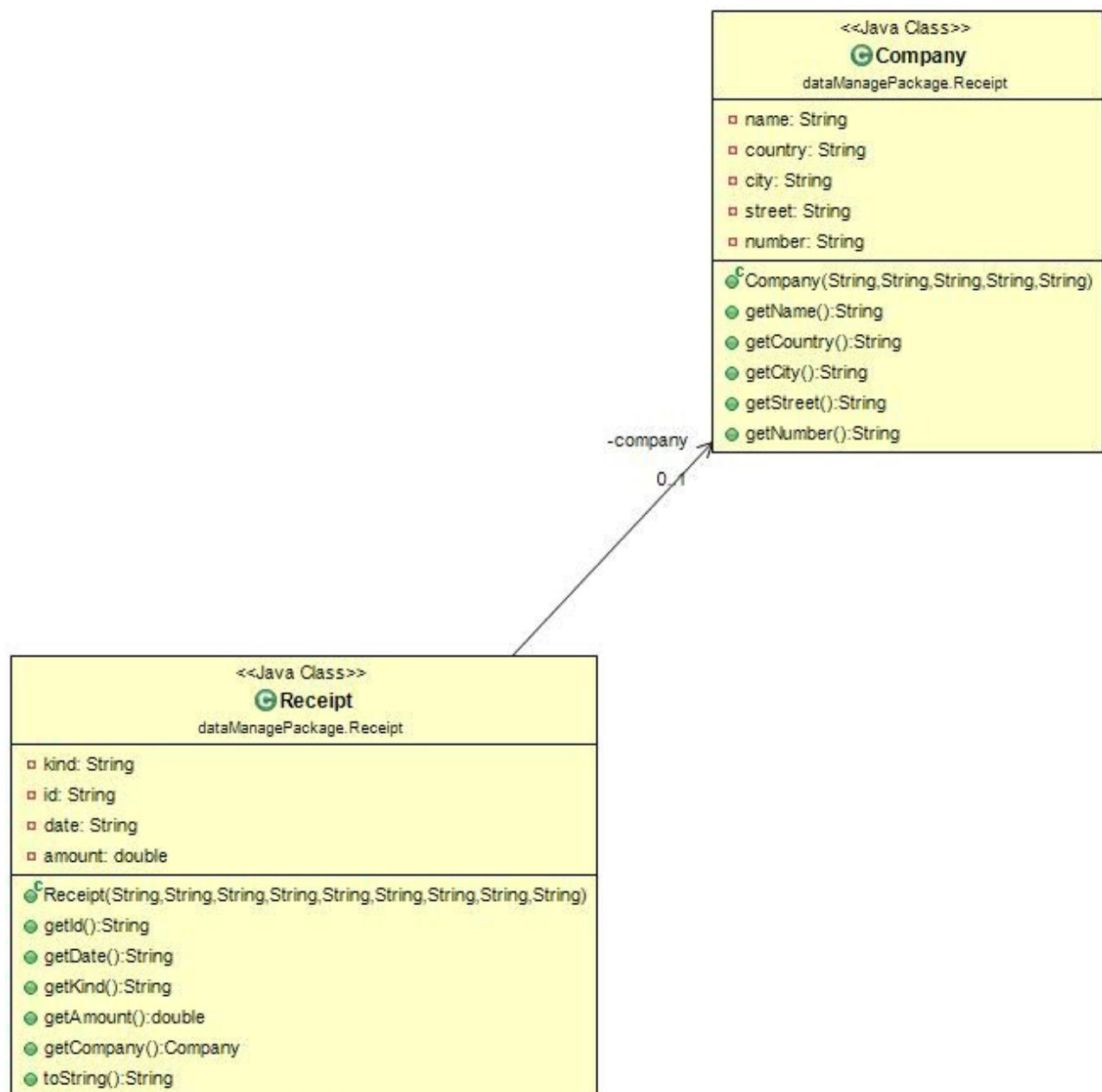


DETAILED DESIGN

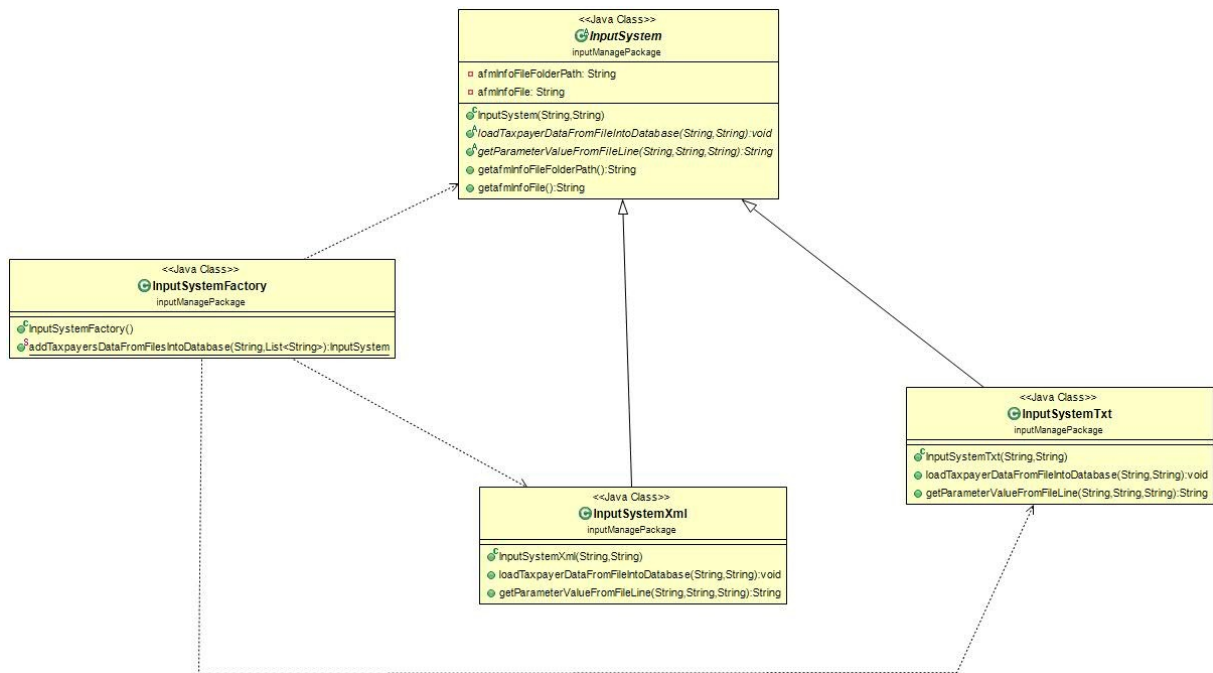
DataManagePackage:



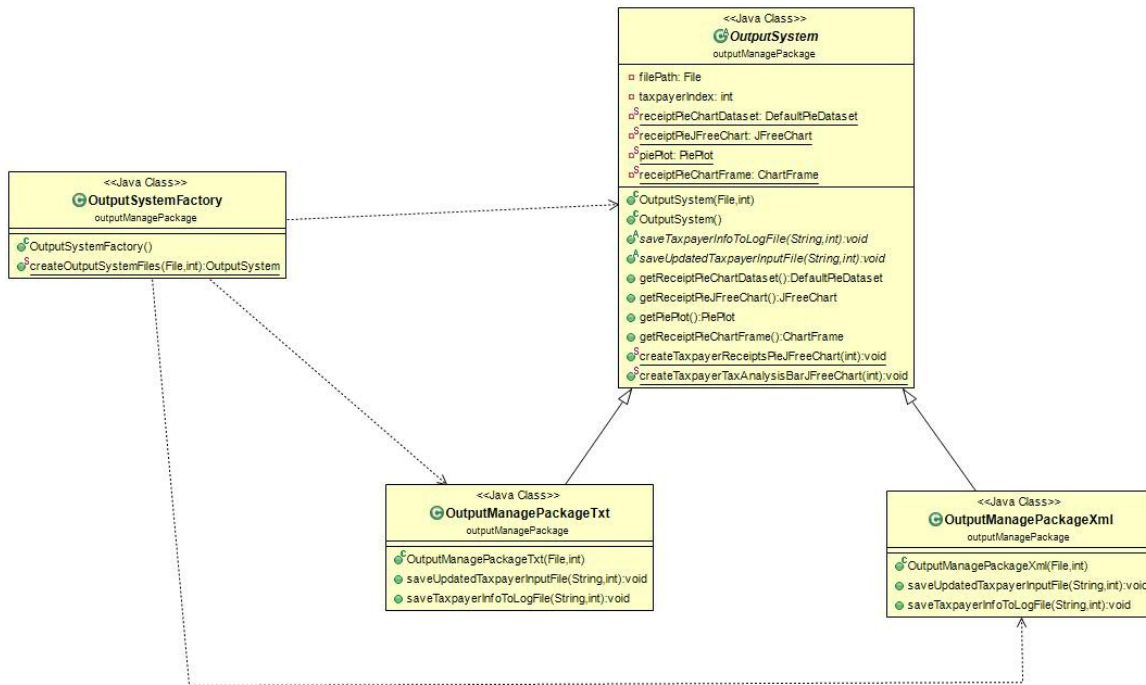
DataManagePackage.Receipt:



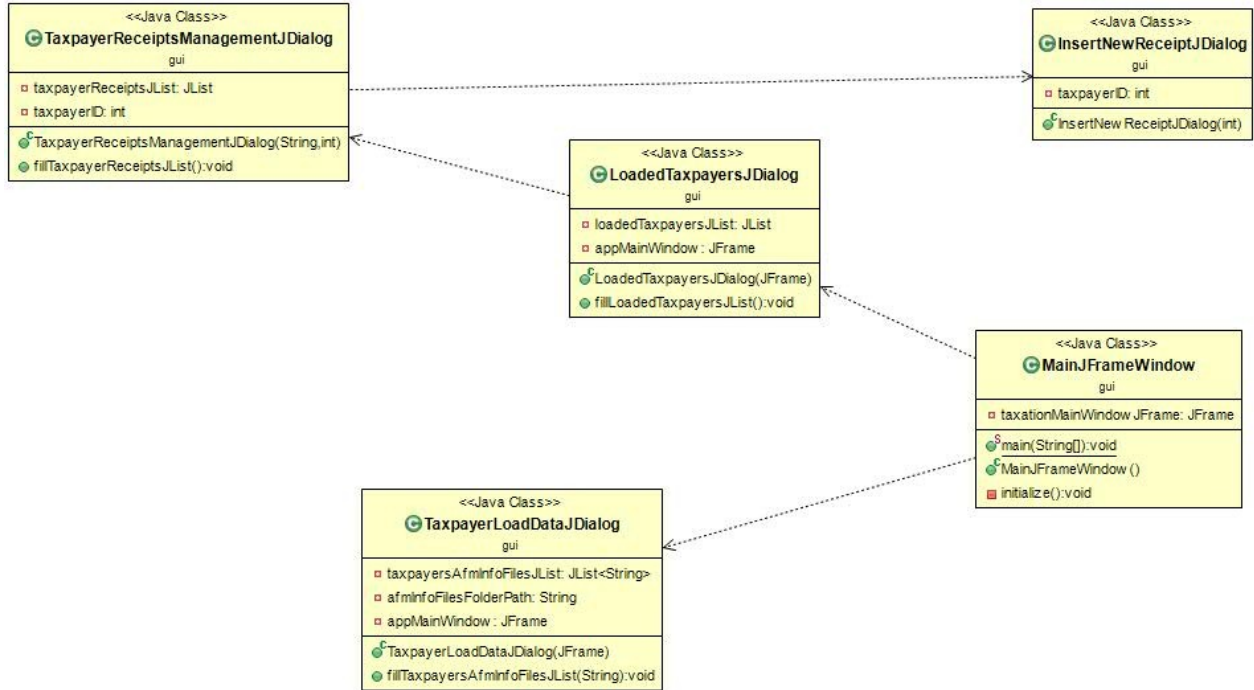
InputManagePackage:



OutputManagePackage:



gui:



Το πρώτο πρόβλημα που αντιμετωπίσαμε ήταν ότι η κλάση `Taxpayer` υλοποιούσε μεθόδους που είχαν παρόμοια λειτουργία. Για να αποφύγουμε τον διπλότυπο κώδικα:

- Συγχωνεύσαμε τις μεθόδους που υπολόγιζαν τον φόρο σύμφωνα με τα έσοδα της κάθε οικογενειακής κατάστασης σε μία μέθοδο.
- Συγχωνεύσαμε τις μεθόδους που υπολόγιζαν το άθροισμα της αξίας των αποδείξεων κάθε τύπου σε μία μέθοδο.

Καθώς υλοποιούσαμε τη συγχώνευση, παρατηρήσαμε ότι ο τρόπος που χρησιμοποιεί τις τιμές για τον υπολογισμό του φόρου επιβαρύνει την κλάση. Έτσι υλοποιήσαμε μια καινούρια κλάση, την `FamilyStatus`, η οποία φτιάχνει τους πίνακες με τις τιμές ανάλογα με την οικογενειακή κατάσταση. Τώρα η `Taxpayer` δεν υλοποιεί η ίδια του πίνακες, απλώς τους χρησιμοποιεί.

Επιλύοντας τα ζητήματα αυτά αποφύγαμε τον διπλότυπο κώδικα και ταυτοχρόνως πετύχαμε την μείωση των αρμοδιοτήτων της `Taxpayer` και την επεκτασιμότητα του κώδικα. Η επεκτασιμότητα επιτεύχθηκε διότι κάθε φορά που θα θελήσουμε να προσθέσουμε μία καινούρια οικογενειακή κατάσταση, θα κάνουμε αλλαγές μόνο στην κλάση `FamilyStatus`, προσθέτοντας ένα ακόμα `if` στον `constructor` της κλάσης.

Το δεύτερο πρόβλημα που αντιμετωπίσαμε ήταν το γεγονός ότι η κλάση `Receipt` είχε 5 παιδιά, τα οποία υλοποιούνταν ακριβώς με τον ίδιο τρόπο, καθώς και μια κλάση που έκανε το διαχωρισμό και καλούσε την μέθοδο της αντίστοιχης κλάσης-παιδί. Για να μειώσουμε την πολυπλοκότητα του σχεδιασμού:

- Διαγράψαμε τις 6 κλάσεις.
- Υλοποιήσαμε 1 μέθοδο μέσα στη `Receipt`, η οποία κάνει το διαχωρισμό του τύπου απόδειξης με `if` και δίνει ανάλογες τιμές στα πεδία.

Επιλύοντας το ζήτημα αυτό μειώσαμε την πολυπλοκότητα καθώς και τον διπλότυπο κώδικα.

Το τρίτο πρόβλημα που αντιμετωπίσαμε ήταν ότι η κλάση Database δεν ακολουθούσε το αντικειμενοστρεφές στυλ. Αυτό συνέβαινε επειδή οι μέθοδοι και τα πεδία ήταν στατικά. Για να επιλύσουμε το ζήτημα αυτό, επανασχεδιάσαμε την κλάση σύμφωνα με το Singleton Pattern, το οποίο δίνει νόημα σε στατικά πεδία και στατικές μεθόδους, καθώς δεν επηρεάζει την αντικειμενοστέφια. Έτσι, επαναφέραμε τις μεθόδους σε public και τα πεδία σε private. Δημιουργήσαμε ένα στατικό πεδίο τύπου Database και υλοποιήσαμε έναν στατικό constructor ο οποίος δημιουργεί το αντικείμενο μόνο στην περίπτωση που δεν υπάρχει(μονοδικό).

Το τέταρτο πρόβλημα που αντιμετωπίσαμε ήταν πως η InputSystem ήταν μια god κλάση. Για την επίλυση του ζητήματος, την υλοποιήσαμε ως abstract την οποία θα κληρονομούν 2 νέες κλάσεις 1 για κάθε τύπο και μία κλάση που θα κάνει τον διαχωρισμό ώστε να καλείται η μέθοδος της κατάλληλης κλάσης. Έτσι, πετύχαμε την επεκτασιμότητα του κώδικα, καθώς για κάθε νέο τύπο αρχείου απλώς προσθέτουμε μια νέα κλάση παιδί, και η InputSystem δεν είναι πλέον god κλάση.

Το πέμπτο πρόβλημα που αντιμετωπίσαμε ήταν στην κλάση OutputSystem, το οποίο το διαχειριστήκαμε όπως το τέταρτο πρόβλημα. Η μόνη διαφορά είναι πως εδώ η OutputSystem είναι υπεύθυνη και για τα γραφήματα.

IMPLEMENTATION

Class Name: Taxpayer	
Responsibilities	Collaborations
<ul style="list-style-type: none">▪ Υπολογίζει το φόρο με βάση τα έσοδα της οικογενειακής κατάστασης.▪ Μειώνει ή αυξάνει το φόρο με βάση τις αποδείξεις που κατατέθηκαν.▪ Υπολογίζει το συνολική αξία όλως των αποδείξεων, καθώς και για κάθε τύπο ξεχωριστά.	<ul style="list-style-type: none">▪ Database▪ FamilyStatus▪ Receipt▪ OutputManagePackageXml▪ OutputManagePackageTxt▪ InputSystemTxt▪ InputSystemXml▪ OutputSystem

Class Name: Database	
Responsibilities	Collaborations
<ul style="list-style-type: none">▪ Κρατάει μία λίστα με όλους του φορολογούμενος και τη διαχειρίζεται.	<ul style="list-style-type: none">▪ Taxpayer▪ OutputManagePackageXml▪ OutputManagePackageTxt

	<ul style="list-style-type: none"> ▪ InputSystemTxt ▪ InputSystemXml ▪ OutputSystemFactory ▪ InputSystemFactory ▪ OutputSystem ▪ InputSystem
--	--

Class Name: FamilyStatus	
Responsibilities	Collaborations
<ul style="list-style-type: none"> ▪ Υλοποιεί τους πίνακες που περιέχουν τα στοιχεία για τον υπολογισμό του φόρου με βάση το εισόδημα της οικογενειακής κατάστασης. 	<ul style="list-style-type: none"> ▪ <u>Taxpayer</u> ▪ <u>InputSystemXml</u> ▪ <u>InputSystemTxt</u>

Class Name: Receipt	
Responsibilities	Collaborations
<ul style="list-style-type: none"> ▪ Υλοποιεί τις αποδείξεις με βάση το τύπο τους. 	<ul style="list-style-type: none"> ▪ <u>Taxpayer</u> ▪ <u>InputSystemXml</u> ▪ <u>InputSystemTxt</u> ▪ <u>Company</u> ▪ OutputManagePackageXml ▪ OutputManagePackageTxt

Class Name: Company	
Responsibilities	Collaborations
<ul style="list-style-type: none"> ▪ Υλοποιεί την εταιρία από την οποία προέρχονται οι αποδείξεις. 	<ul style="list-style-type: none"> ▪ OutputManagePackageXml ▪ OutputManagePackageTxt ▪ Receipt

Class Name: InputSystem	
Responsibilities	Collaborations
<ul style="list-style-type: none"> ▪ Αφηρημένη κλάση η οποία περιέχει μεθόδους που τις υλοποιούν τα παιδιά. 	<ul style="list-style-type: none"> ▪ <u>InputSystemXml</u> ▪ <u>InputSystemTxt</u> ▪ <u>InputFactory</u> ▪ <u>Database</u>

Class Name: InputSystemTxt	
Responsibilities	Collaborations
<ul style="list-style-type: none"> ▪ Διαβάζει τα περιεχόμενα του αρχείου με format txt. 	<ul style="list-style-type: none"> ▪ <u>InputFactory</u> ▪ <u>InputSystem</u> ▪ <u>Taxpayer</u> ▪ <u>Database</u> ▪ <u>FamilyStatus</u> ▪ <u>Receipt</u>

Class Name: InputSystemXml	
Responsibilities	Collaborations
<ul style="list-style-type: none"> ▪ Διαβάζει τα περιεχόμενα του 	<ul style="list-style-type: none"> ▪ <u>InputFactory</u>

αρχείου με format xml.	<ul style="list-style-type: none"> ▪ <u>InputSystem</u> ▪ <u>Taxpayer</u> ▪ <u>Database</u> ▪ <u>FamilyStatus</u> ▪ <u>Receipt</u>
------------------------	---

Class Name: InputSystemFactory	
Responsibilities	Collaborations
<ul style="list-style-type: none"> ▪ Διαχωρίζει τα αρχεία σύμφωνα με το format και καλεί τη μέθοδο της κατάλληλης κλάσης. 	<ul style="list-style-type: none"> ▪ <u>InputSystem</u> ▪ <u>InputSystemTxt</u> ▪ <u>InputSystemXml</u> ▪ <u>Database</u>

Class Name: OutputSystem	
Responsibilities	Collaborations
<ul style="list-style-type: none"> ▪ Αφηρημένη κλάση η οποία περιέχει μεθόδους που τις υλοποιούν τα παιδιά. ▪ Υλοποιεί τα γραφήματα. 	<ul style="list-style-type: none"> ▪ OutputManagePackageXml ▪ OutputManagePackageTxt ▪ Taxpayer

	<ul style="list-style-type: none"> ▪ Database ▪ OutputSystemFactory
--	---

Class Name: OutputManagePackageTxt	
Responsibilities	Collaborations
<ul style="list-style-type: none"> ▪ Γράφει τα περιεχόμενα του αρχείου με format txt. 	<ul style="list-style-type: none"> ▪ <u>Database</u> ▪ <u>Receipt</u> ▪ <u>Company</u> ▪ <u>Taxpayer</u> ▪ <u>OutputSystem</u> ▪ <u>OutputSystemFactory</u>

Class Name: OutputManagePackageXml	
Responsibilities	Collaborations
<ul style="list-style-type: none"> ▪ Γράφει τα περιεχόμενα του αρχείου με format xml. 	<ul style="list-style-type: none"> ▪ <u>Database</u> ▪ <u>Receipt</u> ▪ <u>Company</u> ▪ <u>Taxpayer</u>

	<ul style="list-style-type: none"> ▪ <u>OutputSystem</u> ▪ <u>OutputSystemFactory</u>
--	---

Class Name: OutputSystemFactory	
Responsibilities	Collaborations
<ul style="list-style-type: none"> ▪ Διαχωρίζει τα αρχεία σύμφωνα με το format και καλεί τη μέθοδο της κατάλληλης κλάσης. 	<ul style="list-style-type: none"> ▪ OutputManagePackageXml ▪ OutputManagePackageTxt ▪ OutputSystem ▪ Database