

Project Ivanhoe

Design Document

Group 16

Rebecca Everett

Tobias Hassellöf

Henrik Törnvall

Johan Renner

Martin Waara-Grape

5. Design Details

5.1 Class Responsibility Collaborator (CRC) Cards

Class Profile

Responsibilities

Create Budgets
Create DatedEvents
Keep track of Budgets
Keep track of DatedEvents
Initiate analysis

Collaborators

Budget
DatedEvent
Budget
DatedEvent
AnalyseData

Class AnalyseData

Responsibilities

Create list of EconomicEvents
Create list of DatedEvents
Calculate difference between EconomicEvents and DatedEvents

Collaborators

Profile
Profile
Profile

Class Budget

Responsibilities

Keep track of EconomicEvent
Create EconomicEvents
Check budget deficit

Collaborators

EconomicEvent
EconomicEvent

Class EconomicEvent

Responsibilities

Keep track of data for EconomicEvents:
sum, category, name and Boolean for income/expense

Collaborators

Class DatedEvent

Responsibilities

Keep track of data for a date and all the data in an
EconomicEvent.

Collaborators

Class FileSystem

Responsibilities

Read data from file given a filename
Write data to a file

Collaborators

Profile
Profile

Class SystemController

Responsibilities

Allow user to create a Profile
Allow user to create a Budget
Allow user to add EconomicEvent to a Budget
Allow user to add DatedEvents to Profile
Allow user to delete EconomicEvents from a Budget
Allow user to edit EconomicEvents in a Budget
Allow user to edit DatedEvent in a Profile
Allow user to analyse data

Collaborators

Profile
Budget
Budget, EconomicEvent
Profile, DatedEvent
Budget
Budget, EconomicEvent
Profile, DatedEvent
profile, GUI

Allow user to save Profile to file
 Allow user to read Profile from file
 Allow user to change active Budget
 Allow user to change active Profile

FileSystem, Profile
 FileSystem, Profile
 GUI, Budget, Profile
 GUI, FileSystem, Profile

Class IvanhoeGUI

Responsibilities

Present data in the form of graphs and tables
 Always present an updated view of Profile

Collaborators

AnalyseData
 Profile

Class IvanhoeApplication

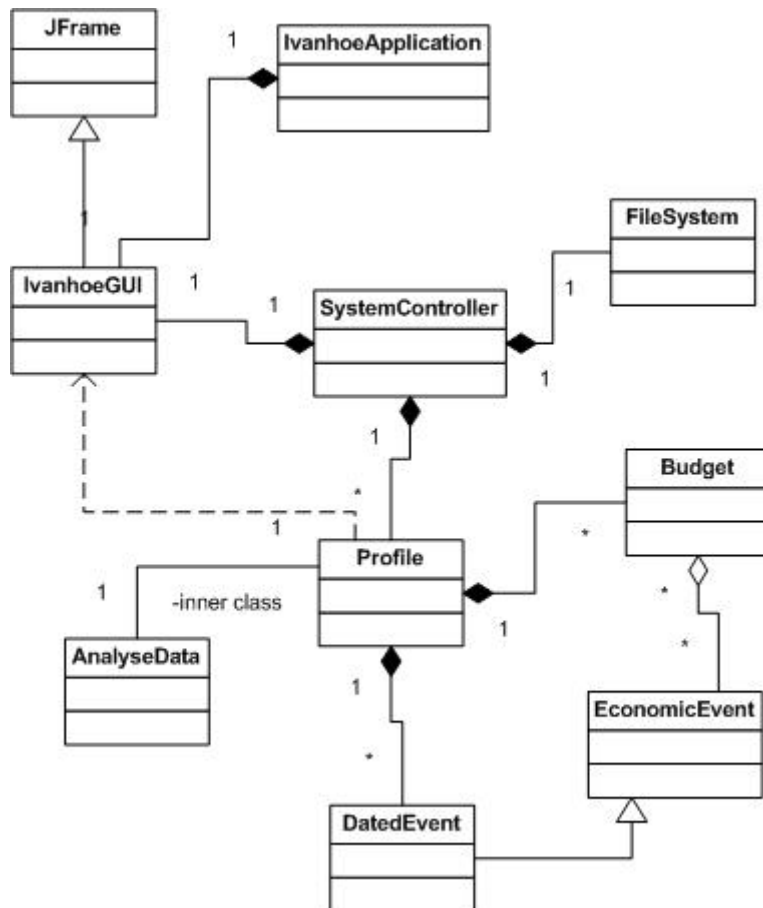
Responsibilities

Initiate system
 Exit system

Collaborators

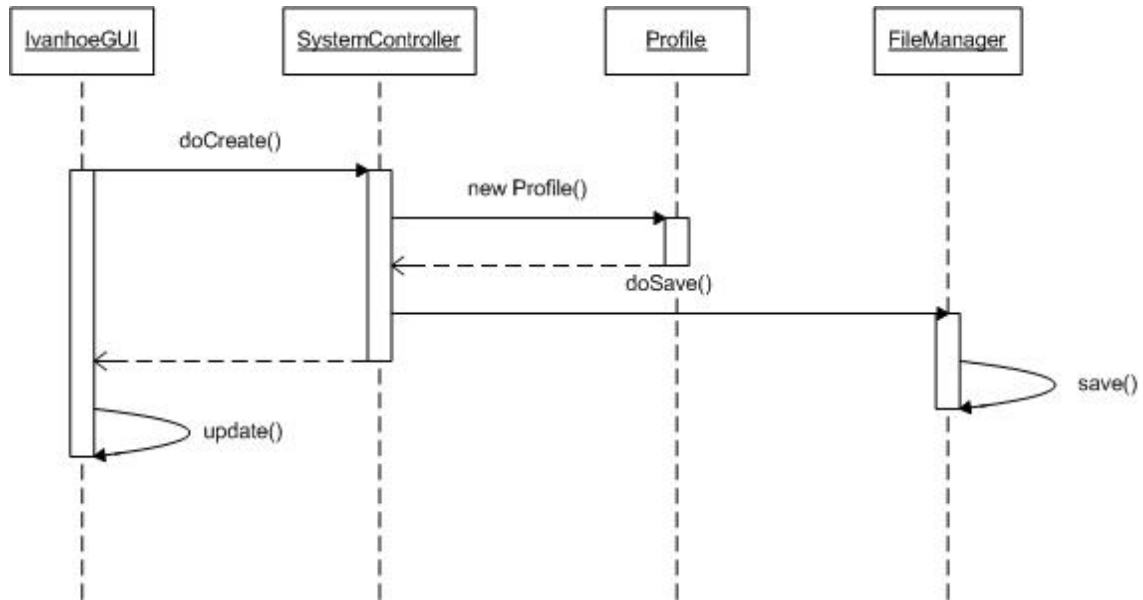
GUI

5.2 Class Diagram



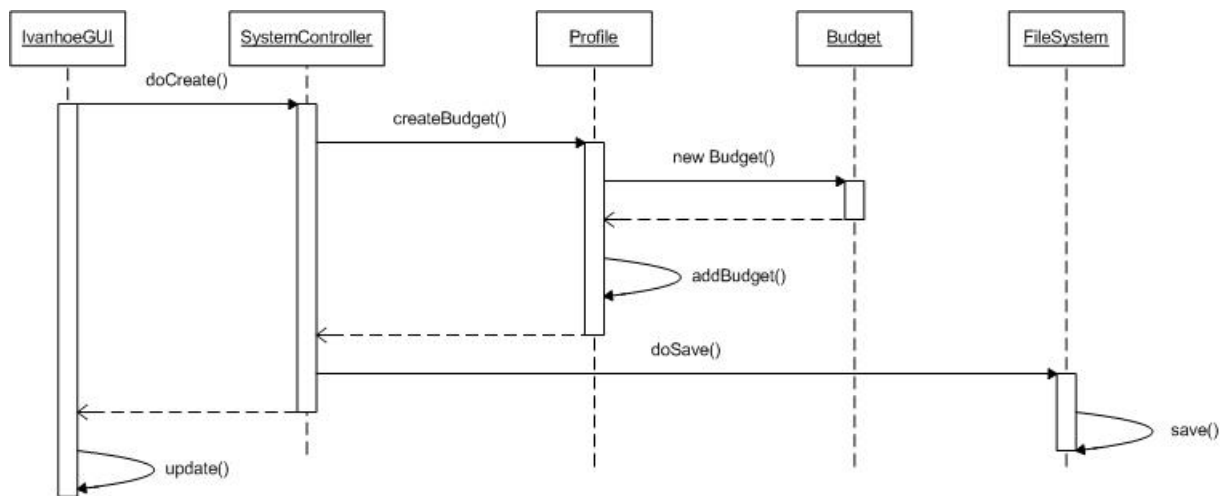
5.3 Interaction Diagrams

Diagram 1 – Create Profile



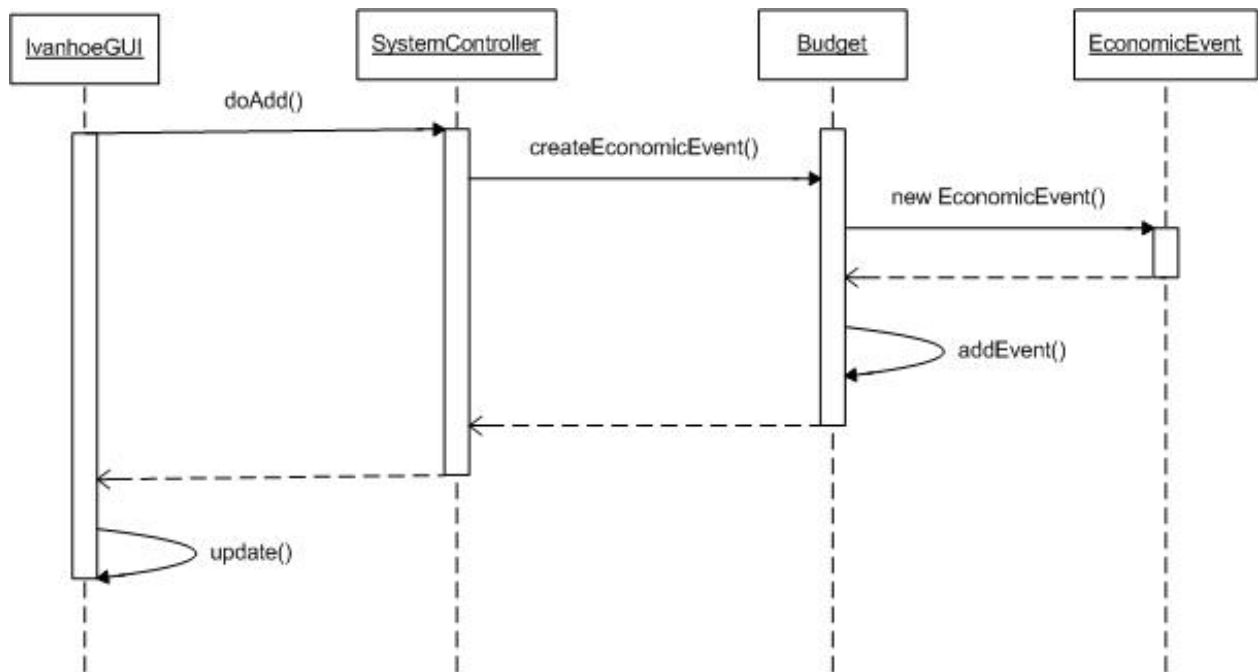
The user clicks a button on the GUI provided by IvanhoeGUI and the command is sent to SystemController. SystemController creates a new Profile and saves the profile by calling the save-method in FileManager. The FileManager saves the Profile by rewriting the XML-file. When the Profile has been saved, the IvanhoeGUI is updated and shows the Profile to the user.

Diagram 2 – Create Budget



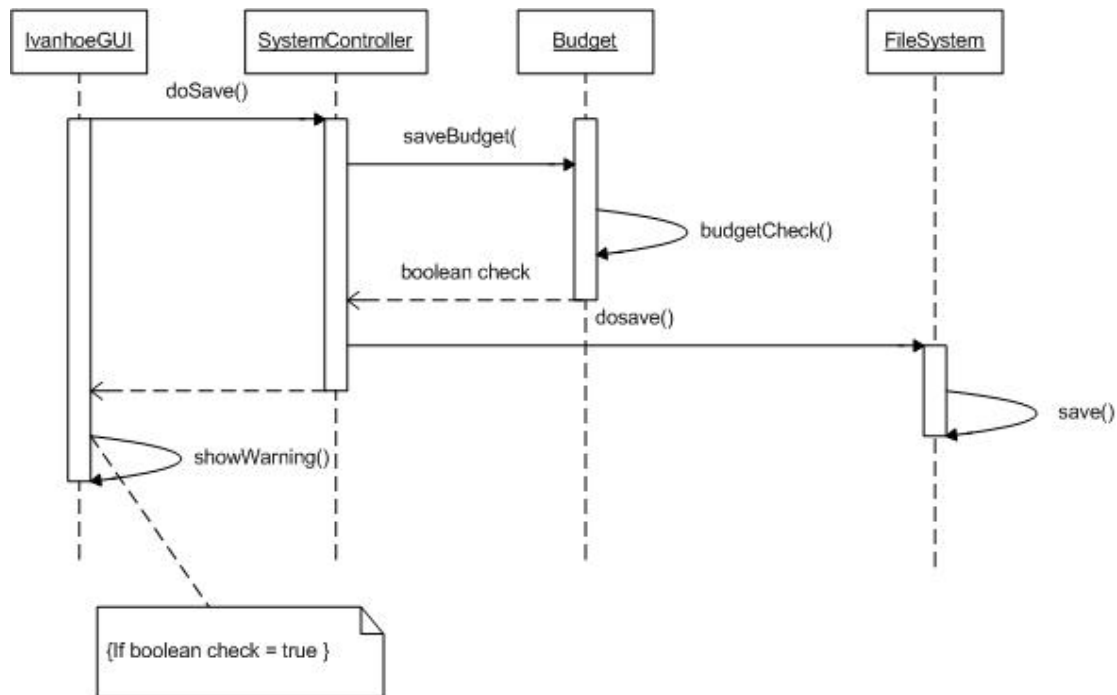
The user clicks a button to create a new profile and inputs the data needed. The command is sent to SystemController that calls the method `createBudget()` in Profile. The Profile creates a new Budget. When the Budget is created, it is added to the Profile. The SystemController saves the edited Profile by calling the `save`-method in FileManager. The FileManager saves the Profile by rewriting the XML-file. When the Profile has been saved, the IvanhoeGUI is updated and shows the Budget to the user.

Diagram 3 – Edit Active Budget



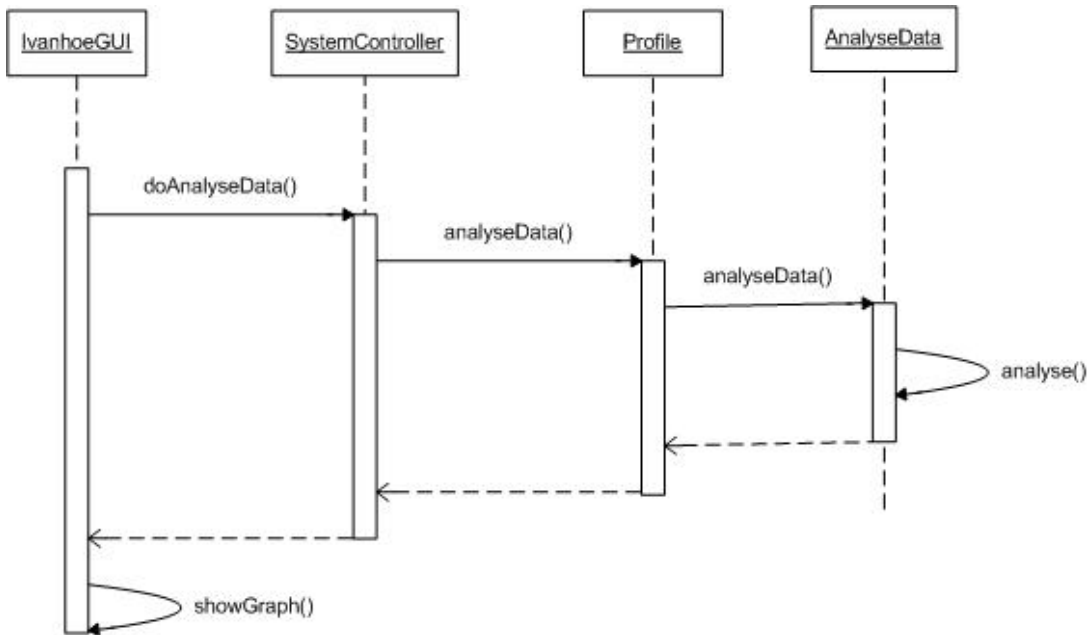
The user clicks a button to add an `EconomicEvent` to the active `Budget` after inputting the required data. The command is sent to `SystemController` that calls the method `createEconomicEvent()` in `Budget`. The `Budget` creates a new `EconomicEvent`. When the `EconomicEvent` is created, it is added to the `Budget`. When the `EconomicEvent` has been added to the `Budget`, the `IvanhoeGUI` is updated and shows the `Budget` to the user.

Diagram 4 – Save Active Budget



The user clicks a button to save an edited budget. The command is sent to SystemController that calls the method `saveBudget()` in Budget. The Budget checks if there is a budget deficit and returns a Boolean. The SystemController saves the edited Profile by calling the `save`-method in FileManager. The FileManager saves the Profile by rewriting the XML-file. If there is a budget deficit, the IvanhoeGUI shows a dialog warning the user. When the dialog is closed or if there wasn't a budget deficit the IvanhoeGUI shows the Budget to the user.

Diagram 5 – Create Graph



The user clicks a button to create a graph and inputs the data needed. The command is sent to SystemController that calls the method `analyseData()` in Profile. The Profile, in turn, calls the method `analyseData()` in AnalyseData. The AnalyseData analysis the data stored in the Profile according to the user's input and returns the result. The IvanhoeGUI displays the data as a graph.