

Appendix

1. Example UML Diagrams

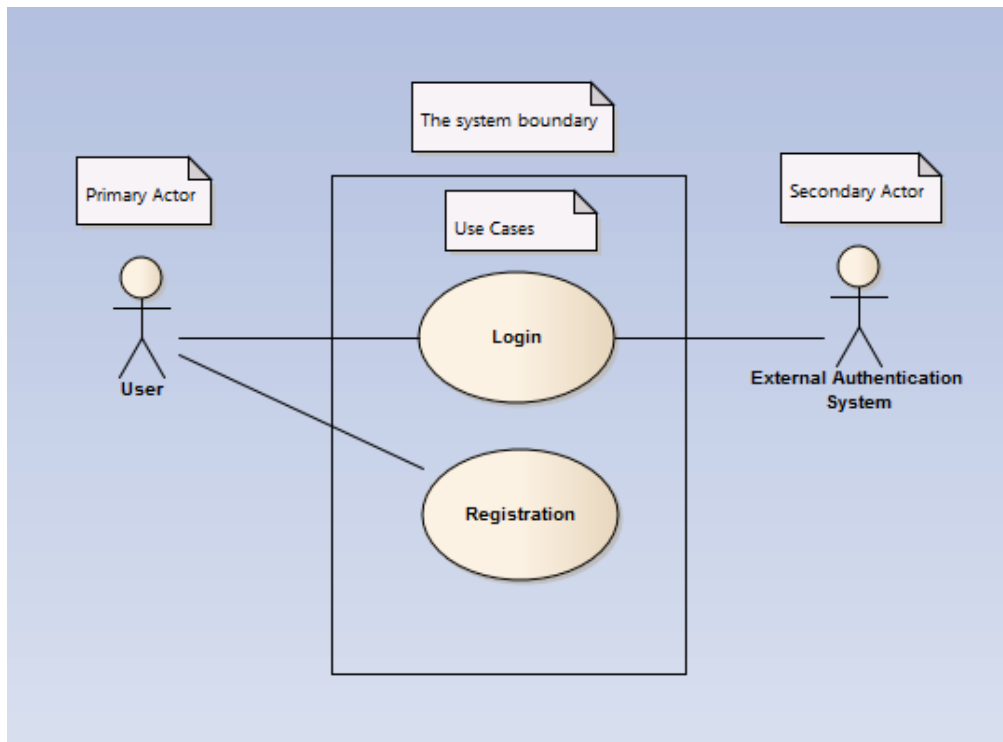


Figure 20: An example Use Case Diagram

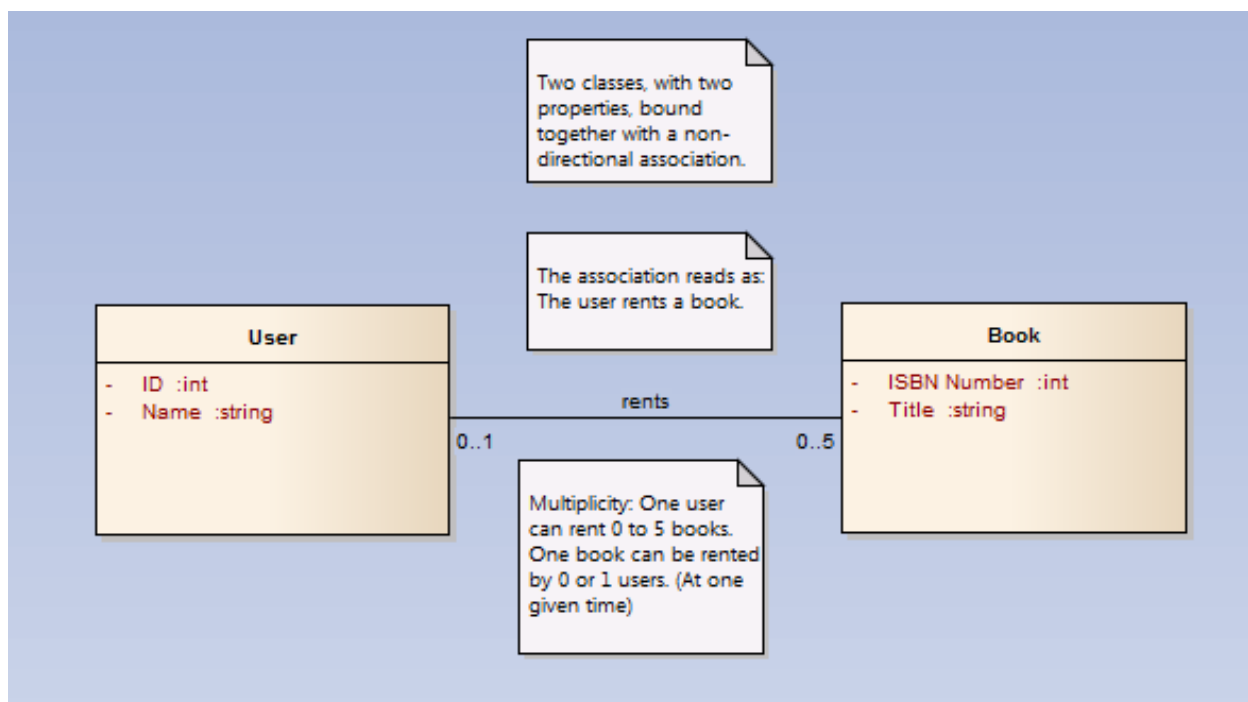


Figure 21: An example Class Diagram

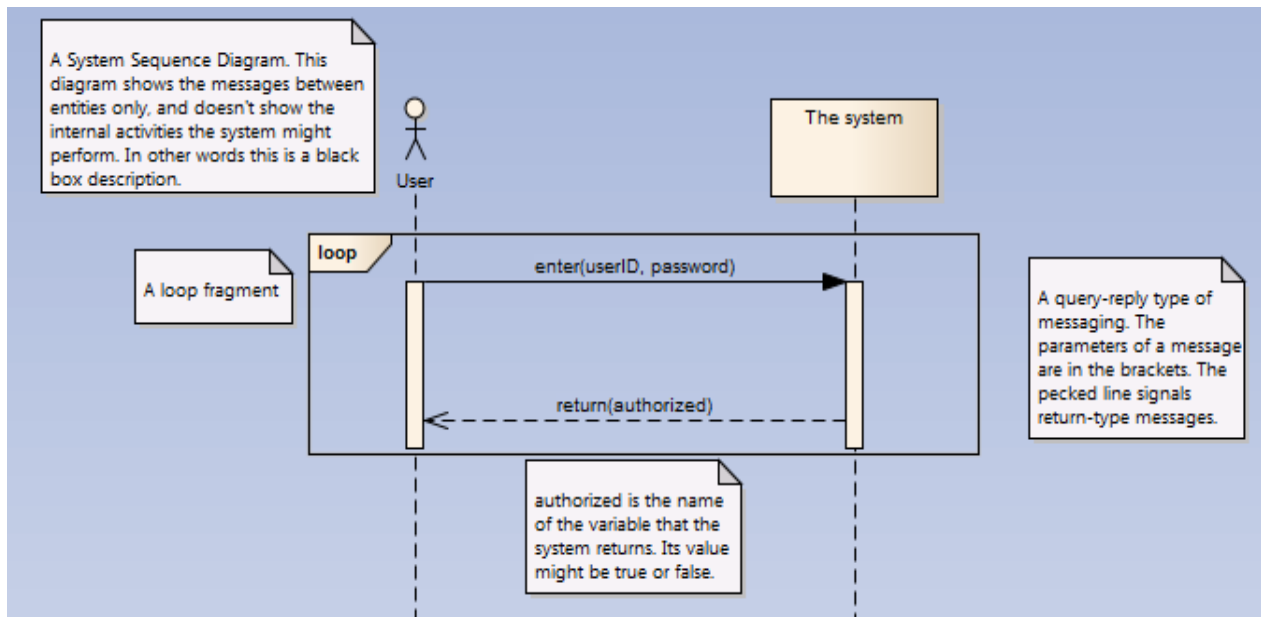


Figure 22: An example system sequence diagram

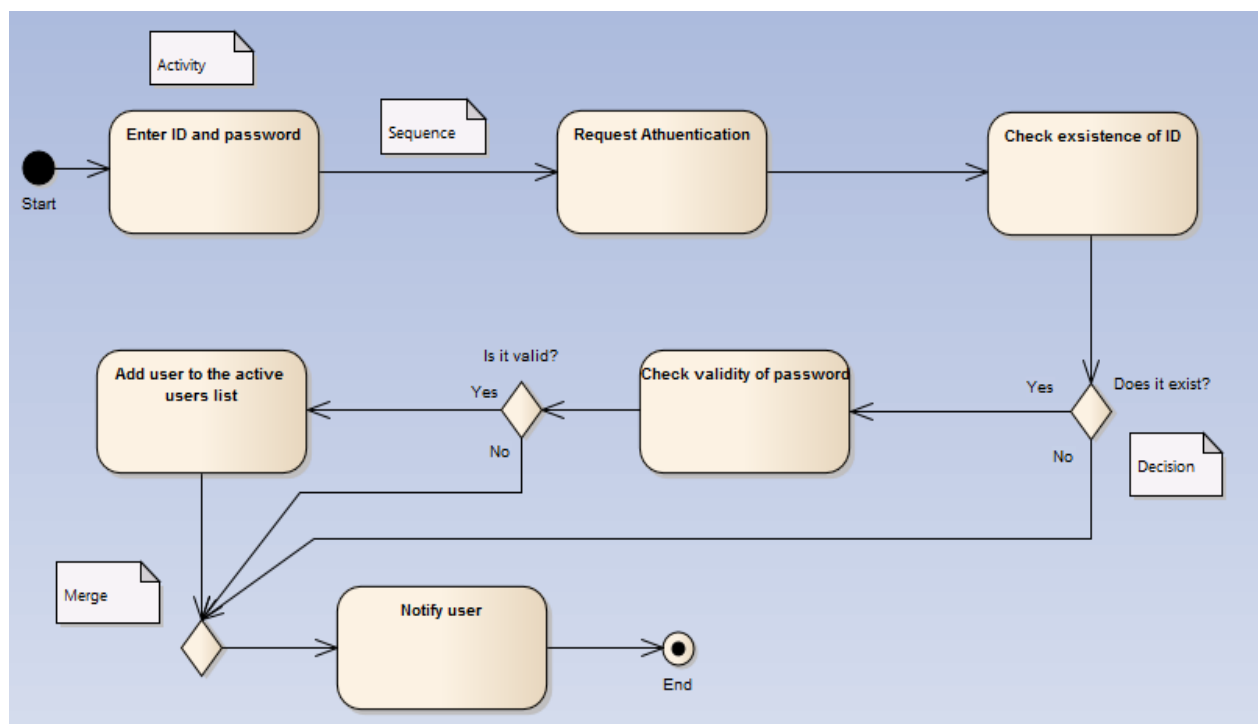


Figure 23: An example Activity Diagram

2. Domain Model Diagrams

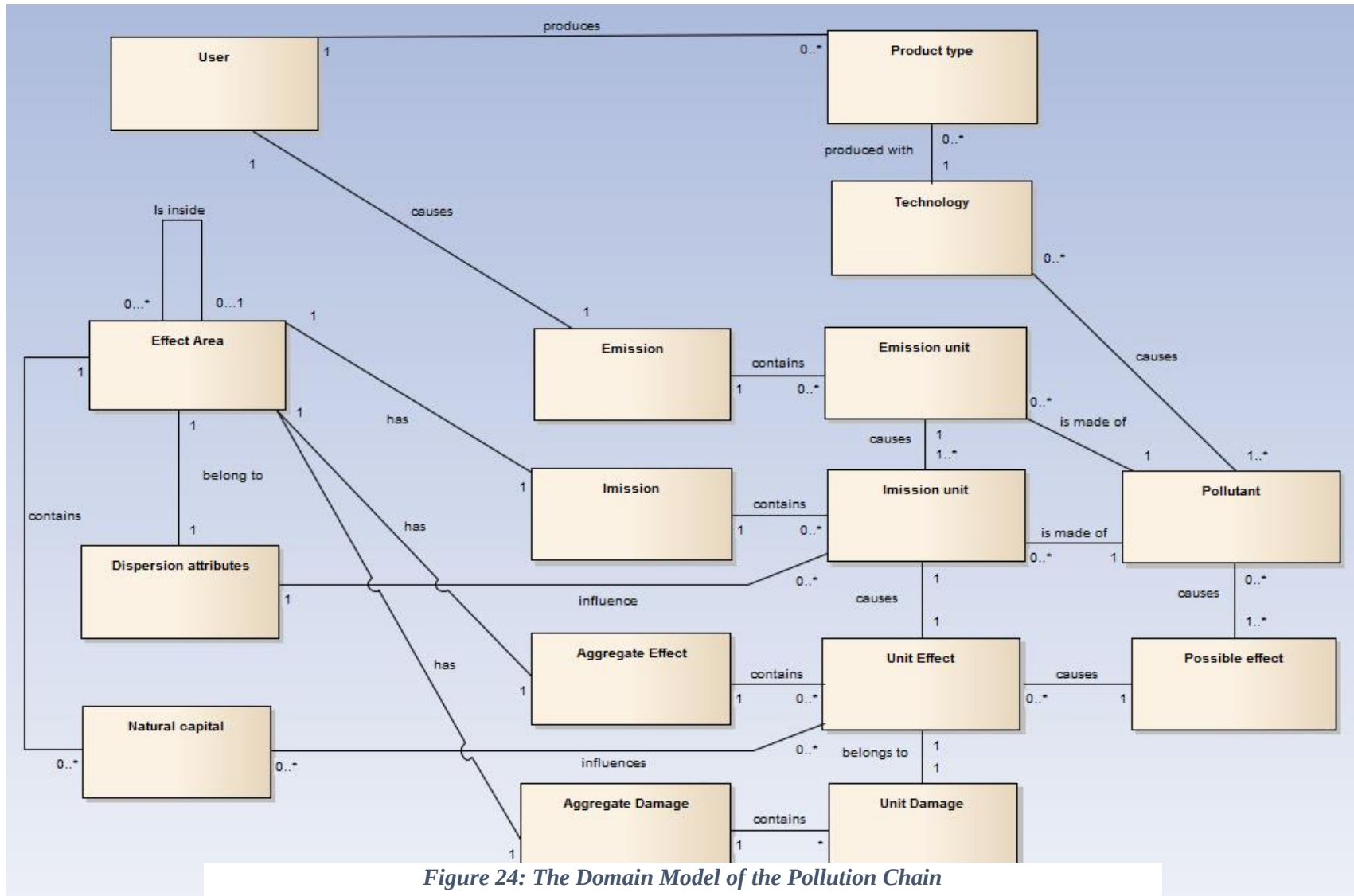


Figure 24: The Domain Model of the Pollution Chain

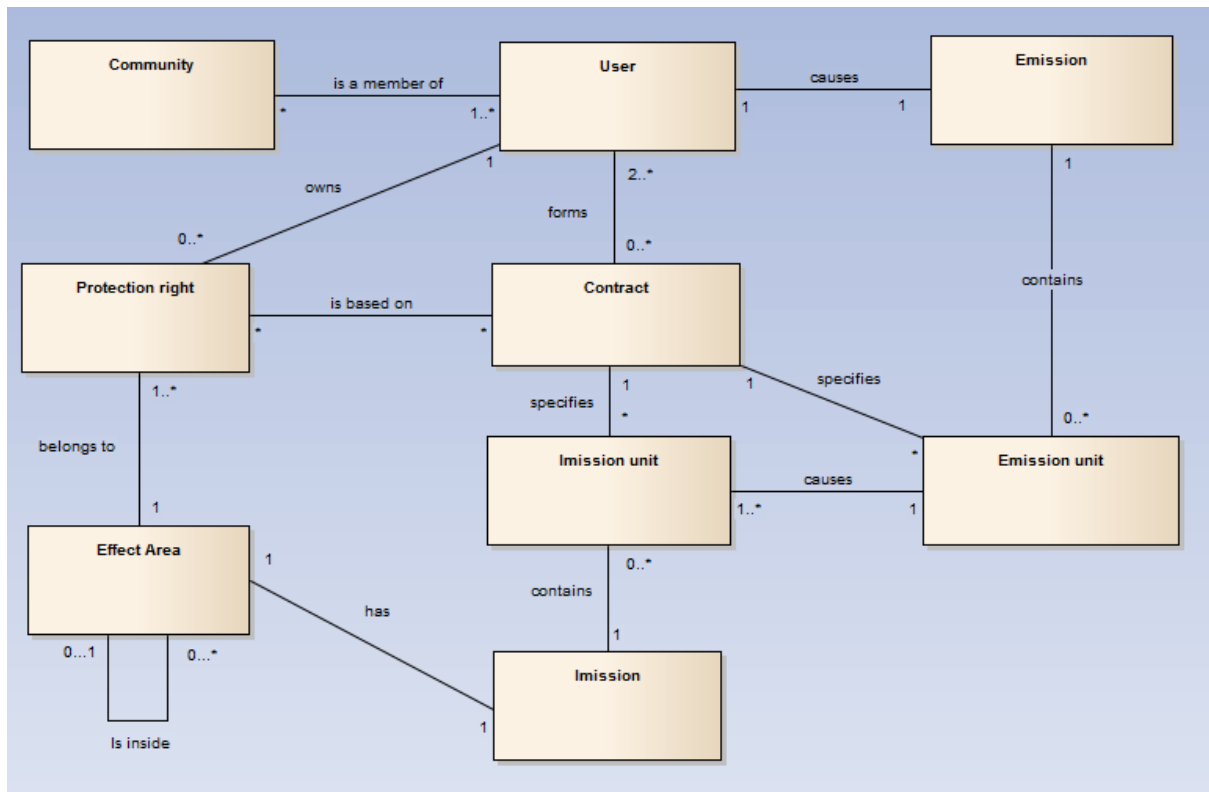


Figure 25: The Domain Model of Contracting

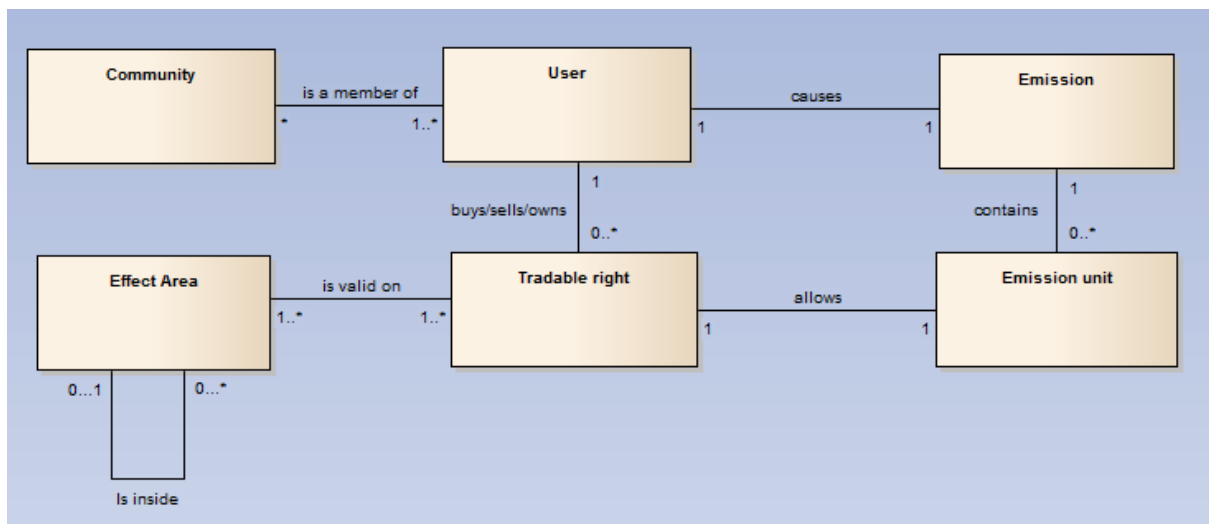


Figure 26: The Domain Model of Rights Trading

3. Use Case Diagrams

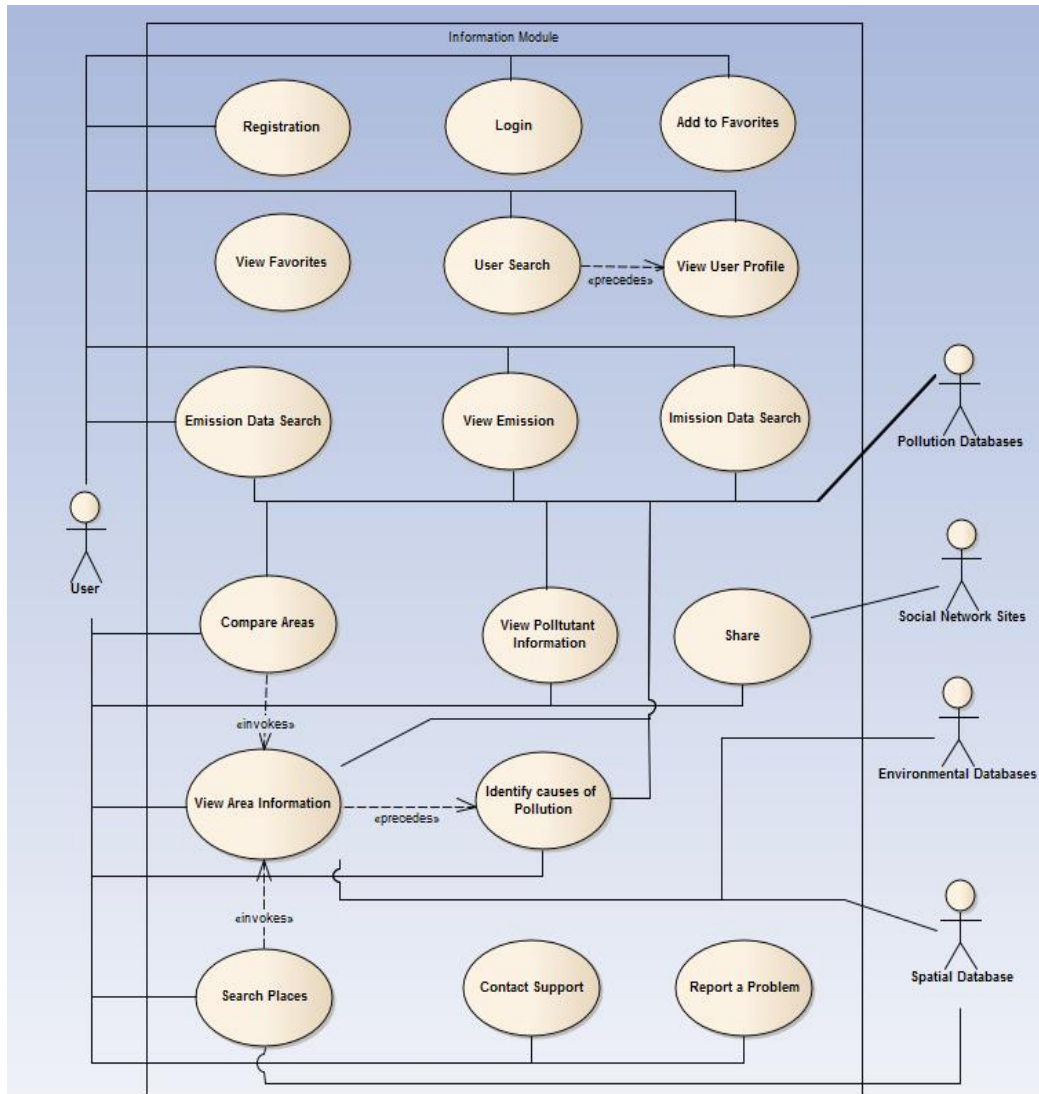


Figure 27: The Use Case Diagram of the Information Module

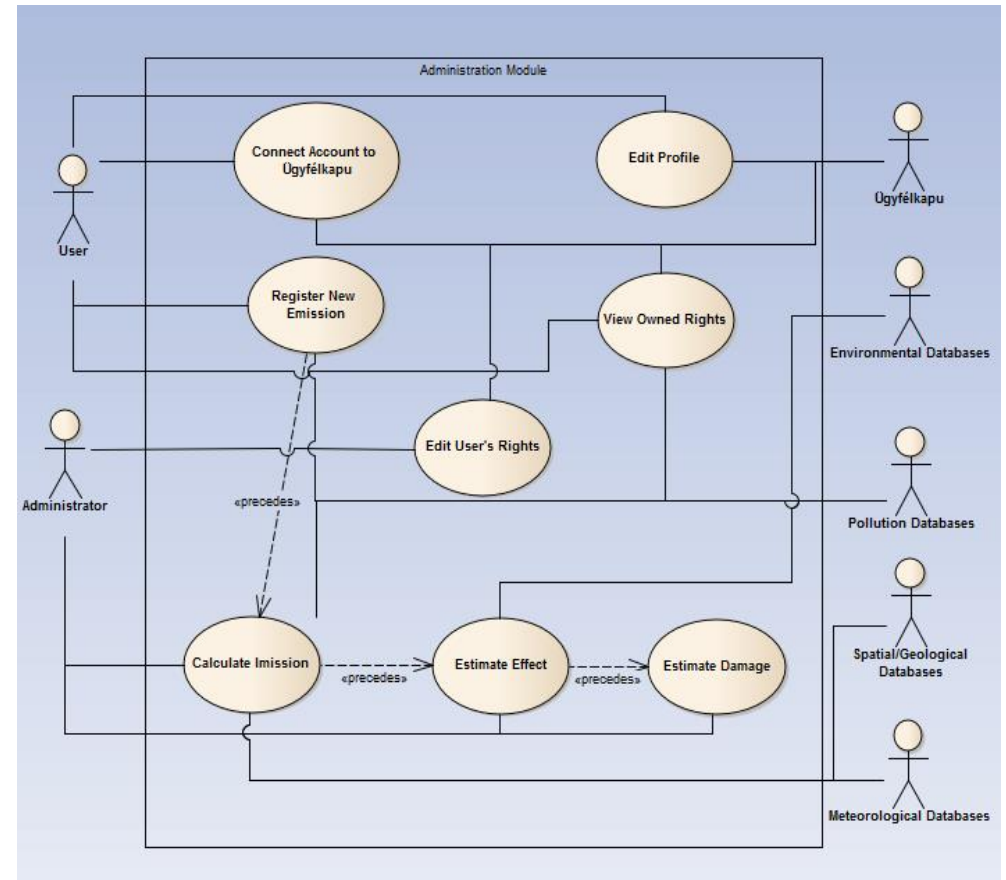


Figure 28: Use Case Diagram of the Administration Module

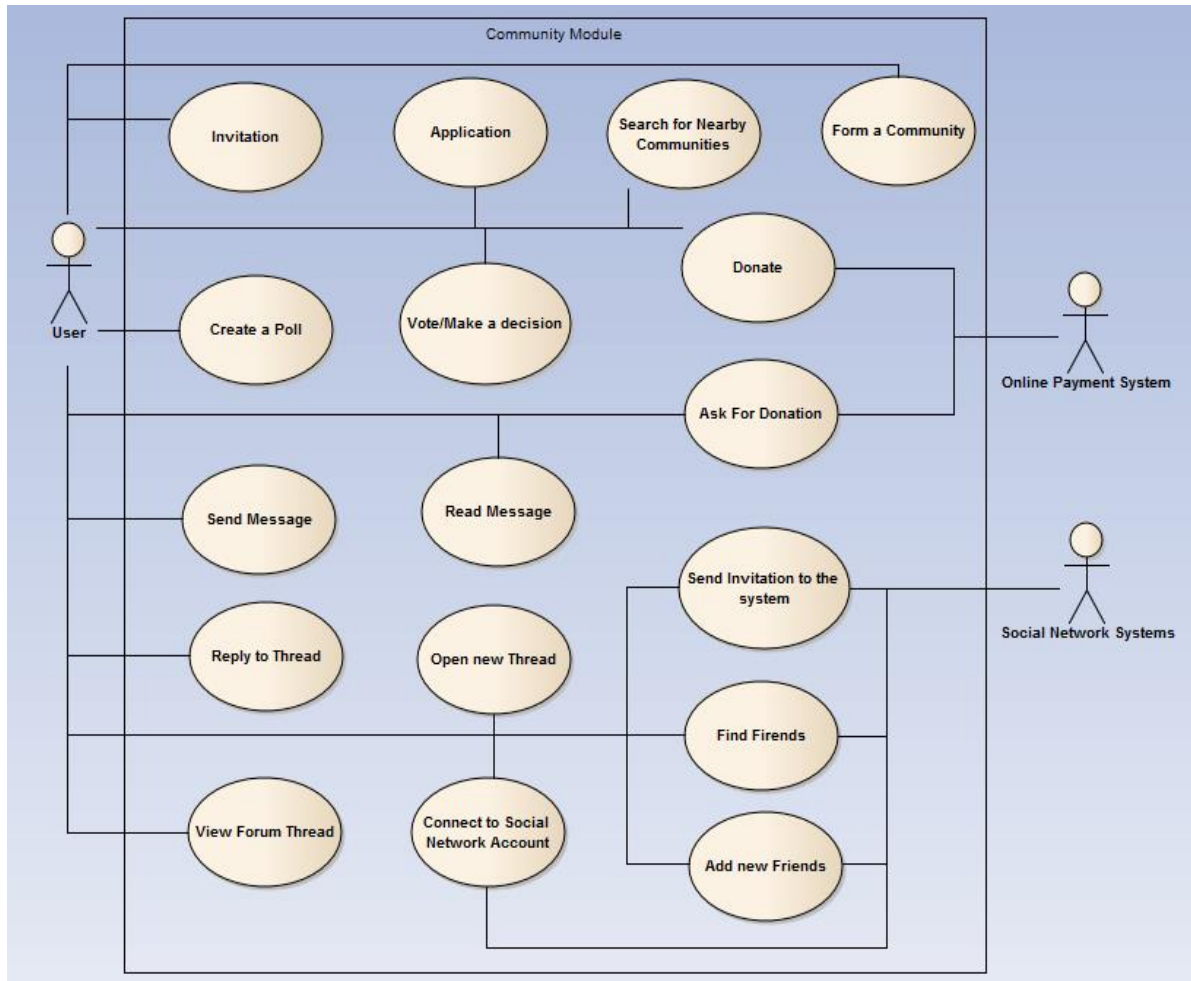


Figure 29: Use Case Diagram for the Community Module

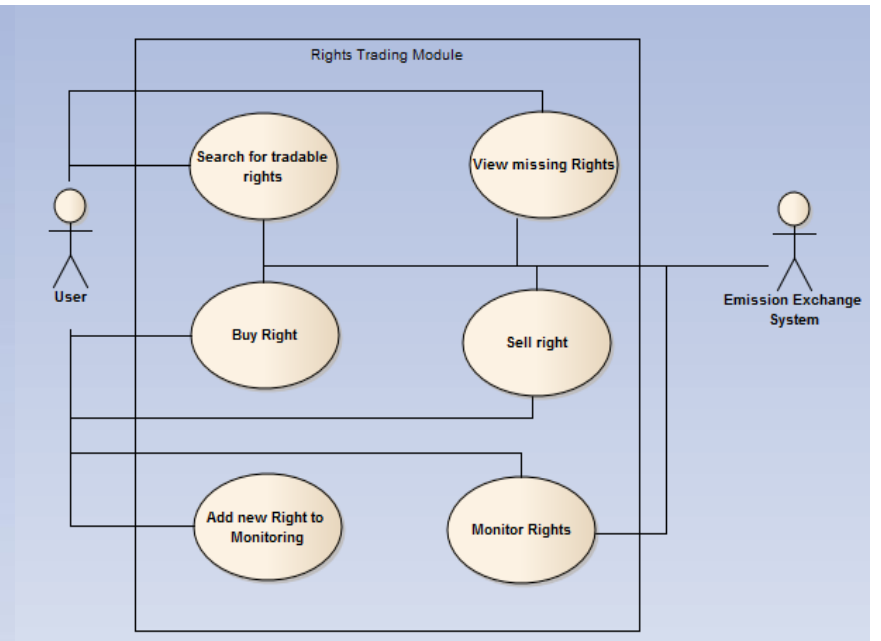


Figure 30: Use Case Diagram for the Rights Trading Module

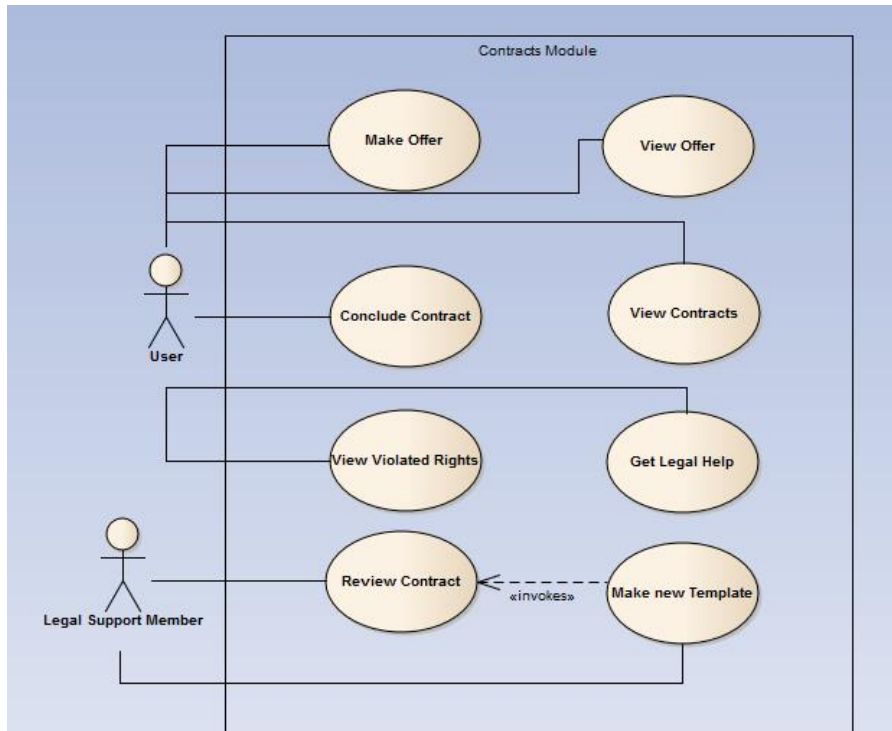


Figure 31: Use Case Diagram for the Contract Module

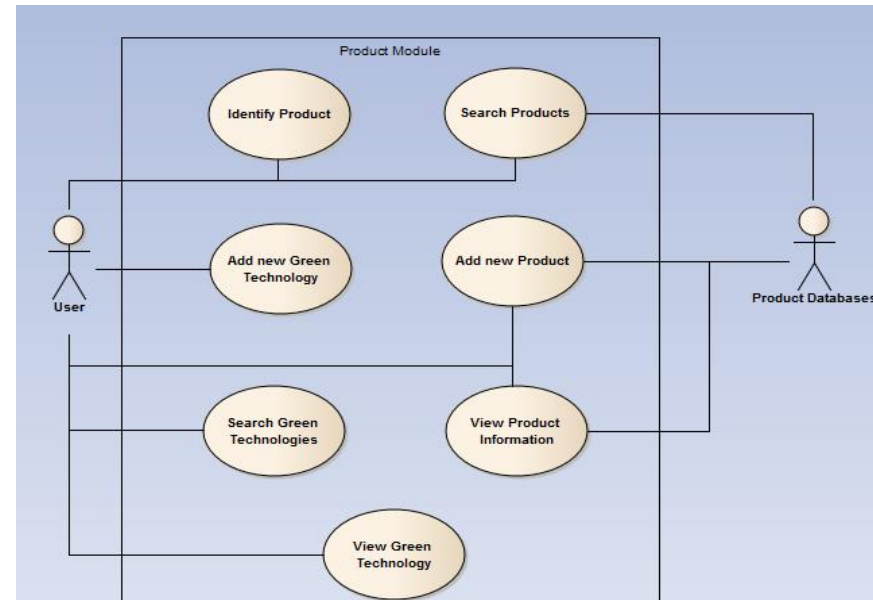


Figure 32: Use Case Diagram for the Product Information Module

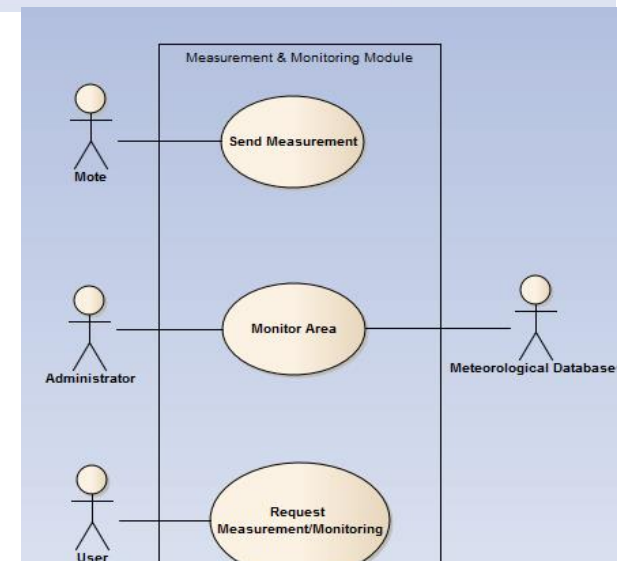


Figure 33: Use Case Diagram for the M&M Module

4. Activity and System Sequence Diagrams

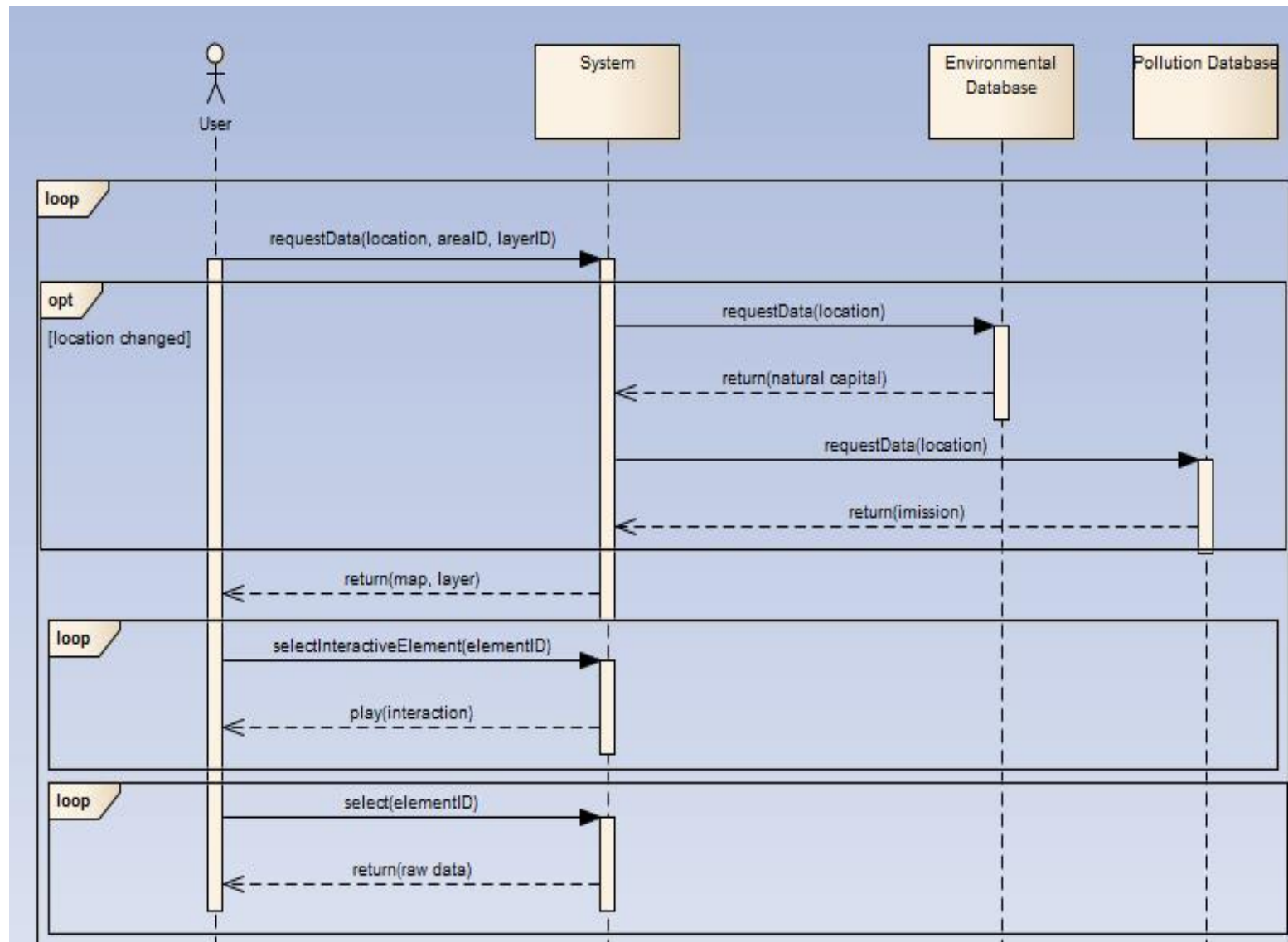


Figure 34: SSD of View Area Information

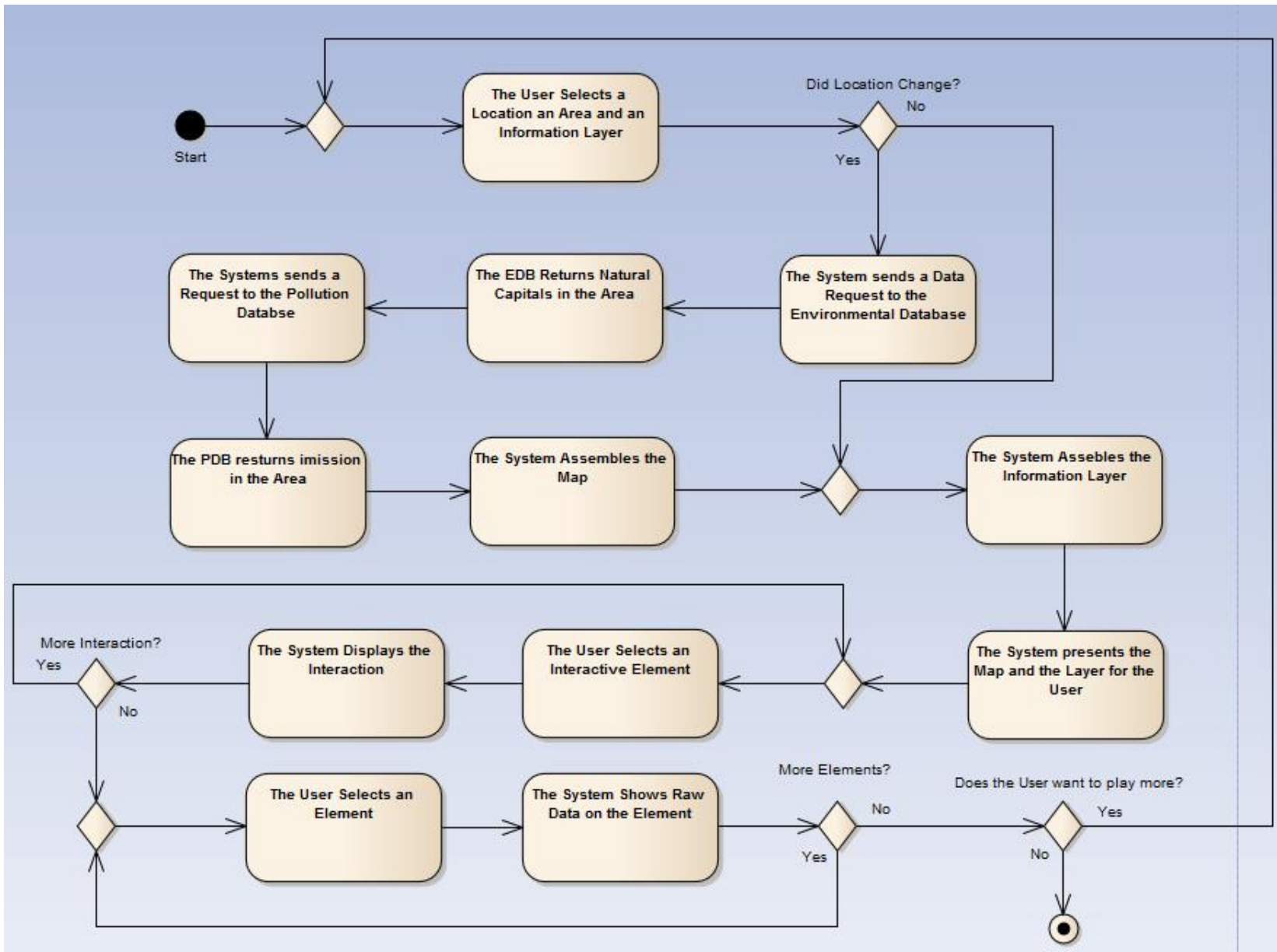


Figure 35: AD of View Area Information

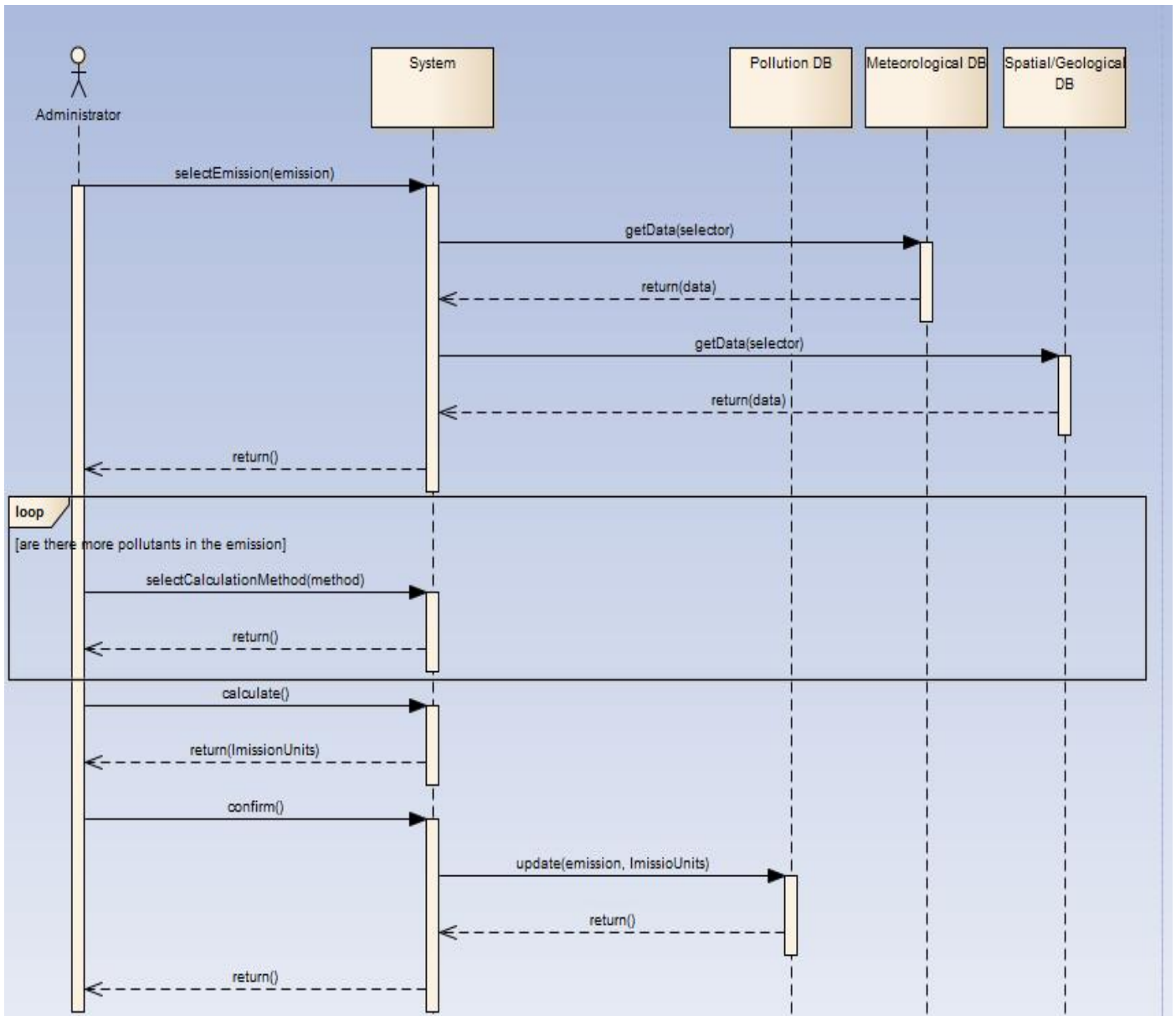


Figure 36: SSD for Calculate Emission

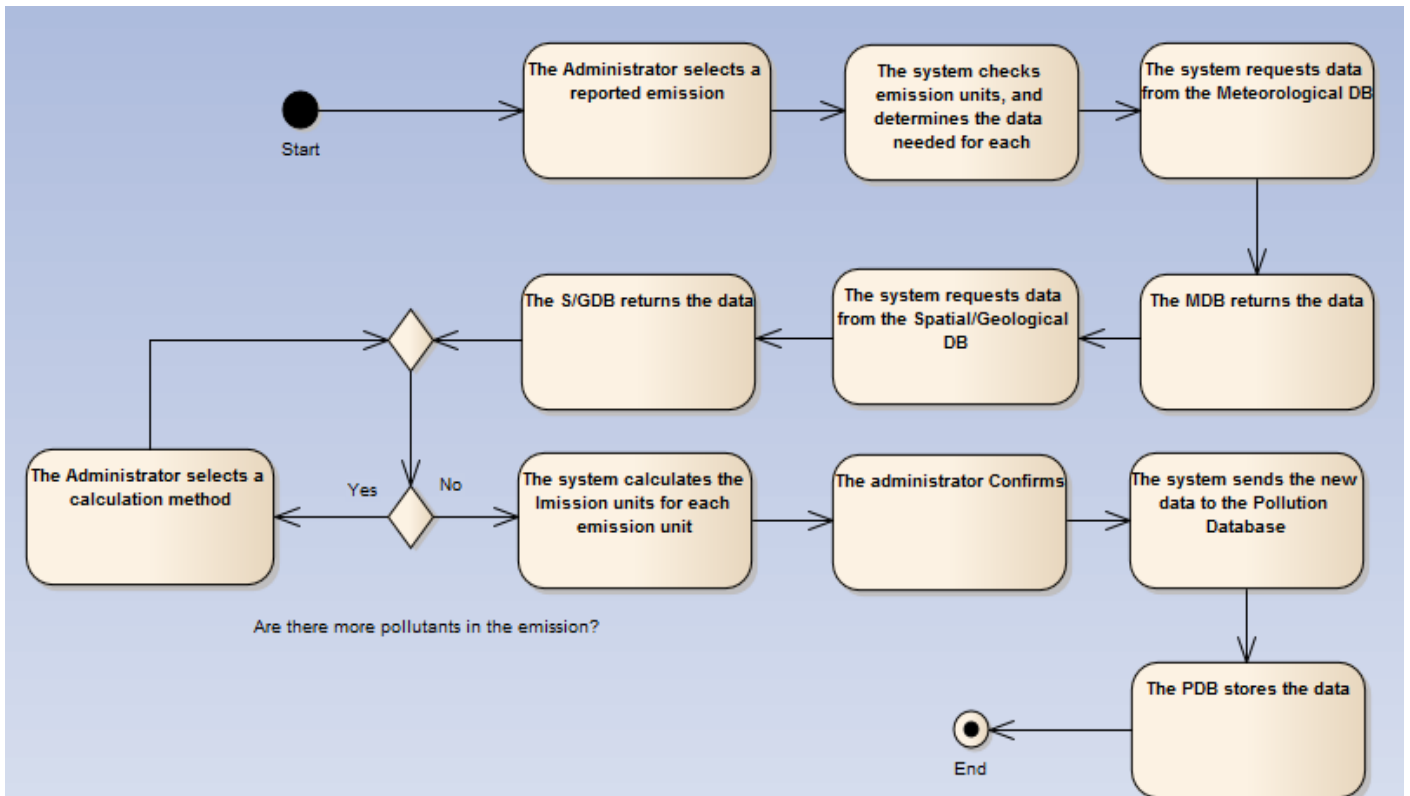


Figure 37: AD for Calculate Emission

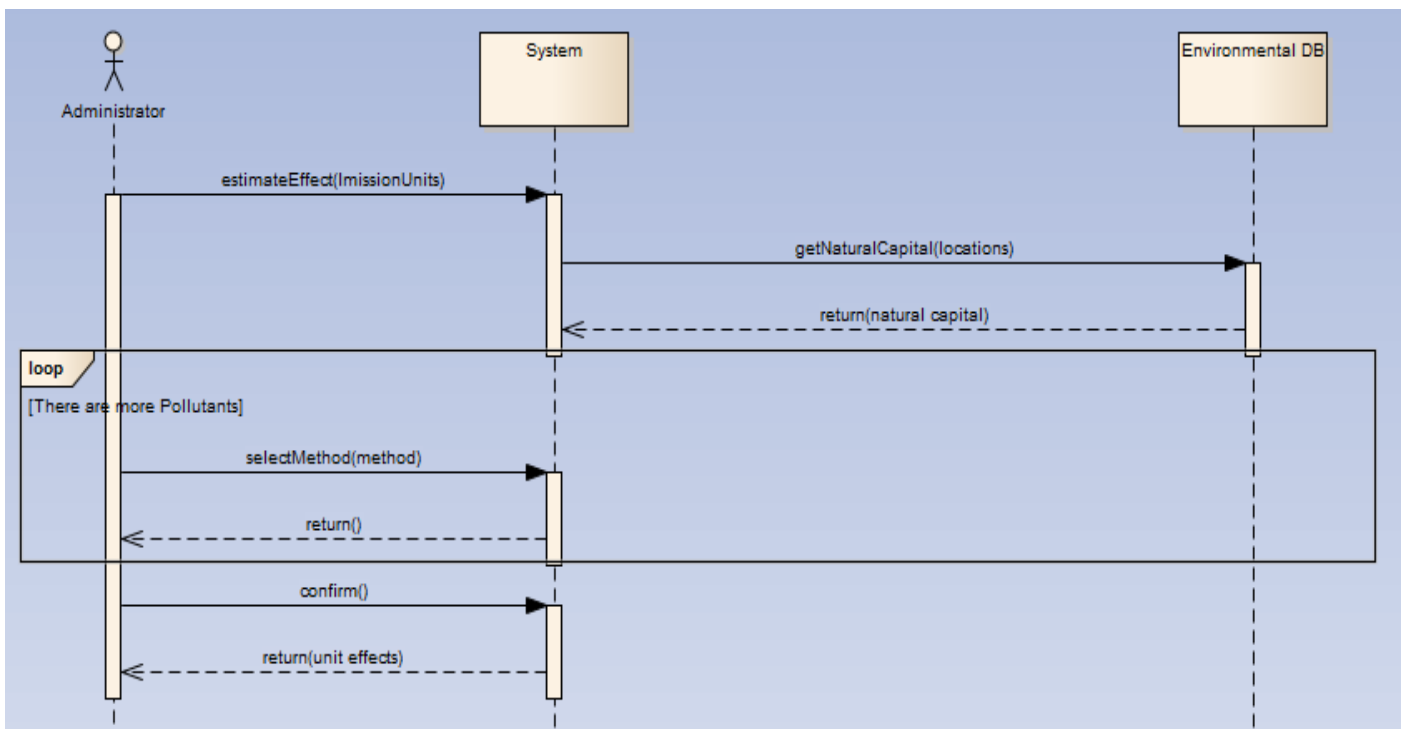


Figure 38: SSD for Estimate Effect

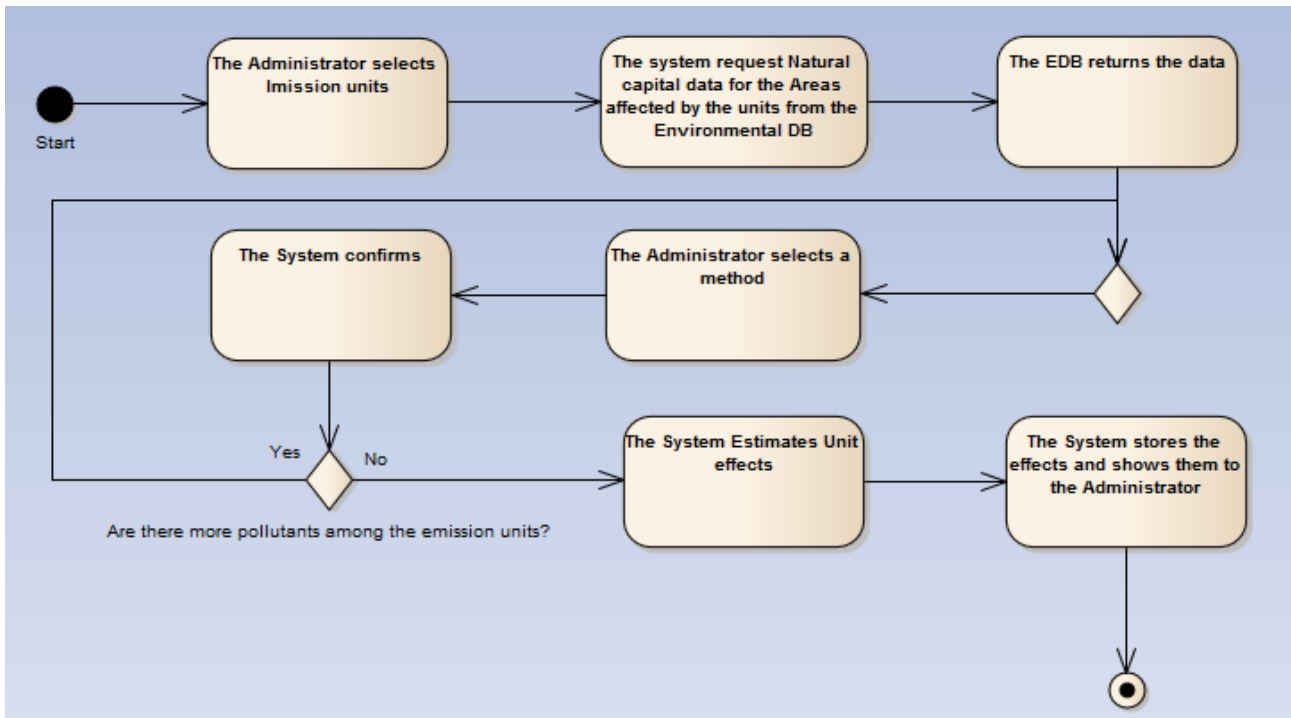


Figure 39: AD for Estimate Effect

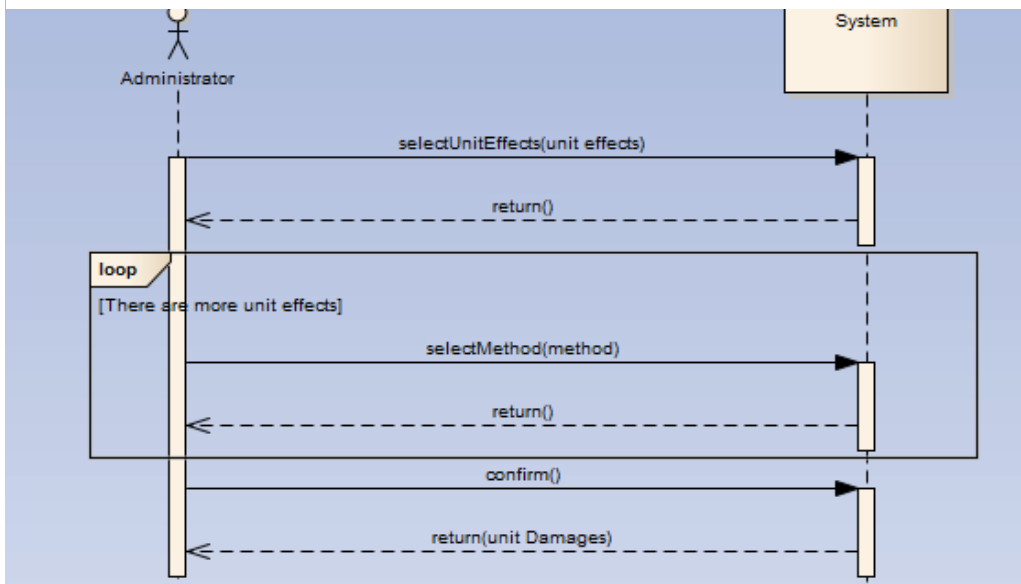


Figure 40: SSD for Estimate Damage

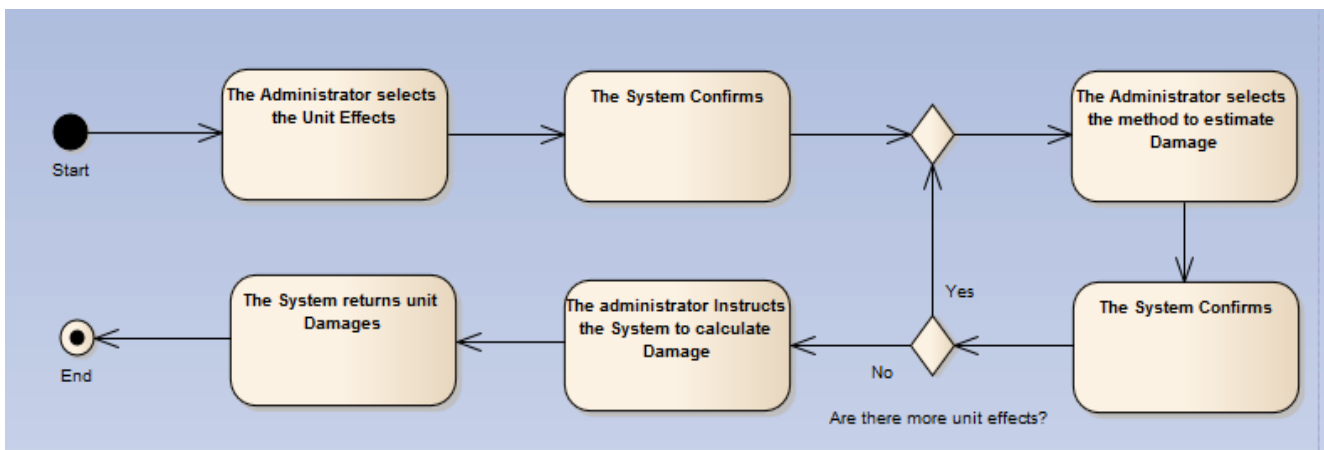


Figure 41: AD for Estimate Damage

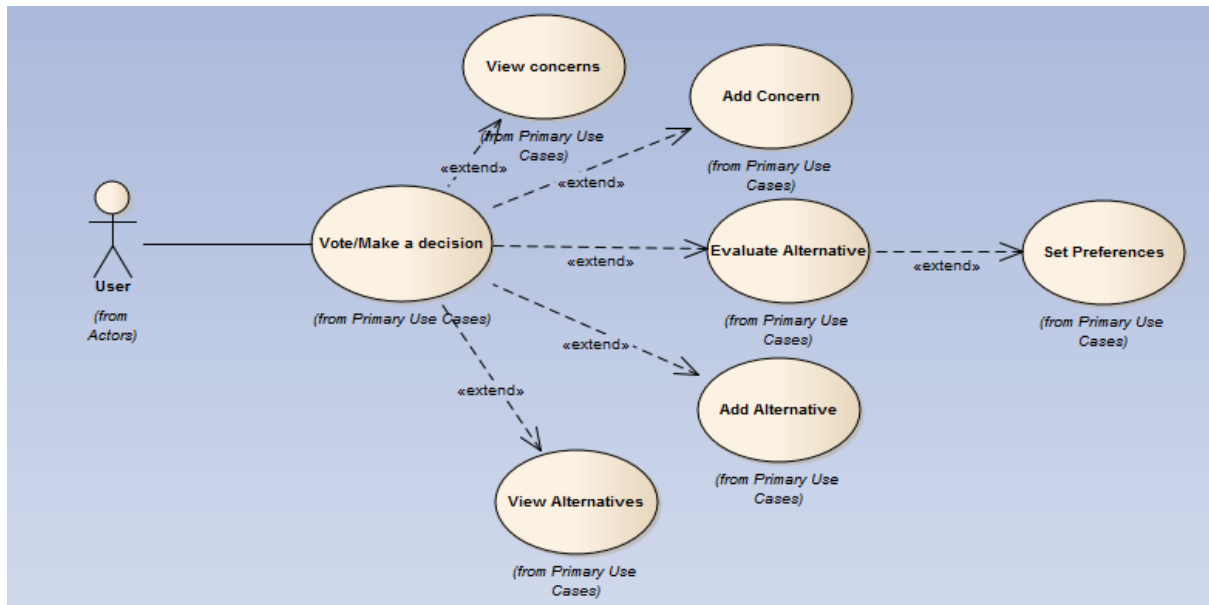


Figure 42: Use Case Diagram for Vote

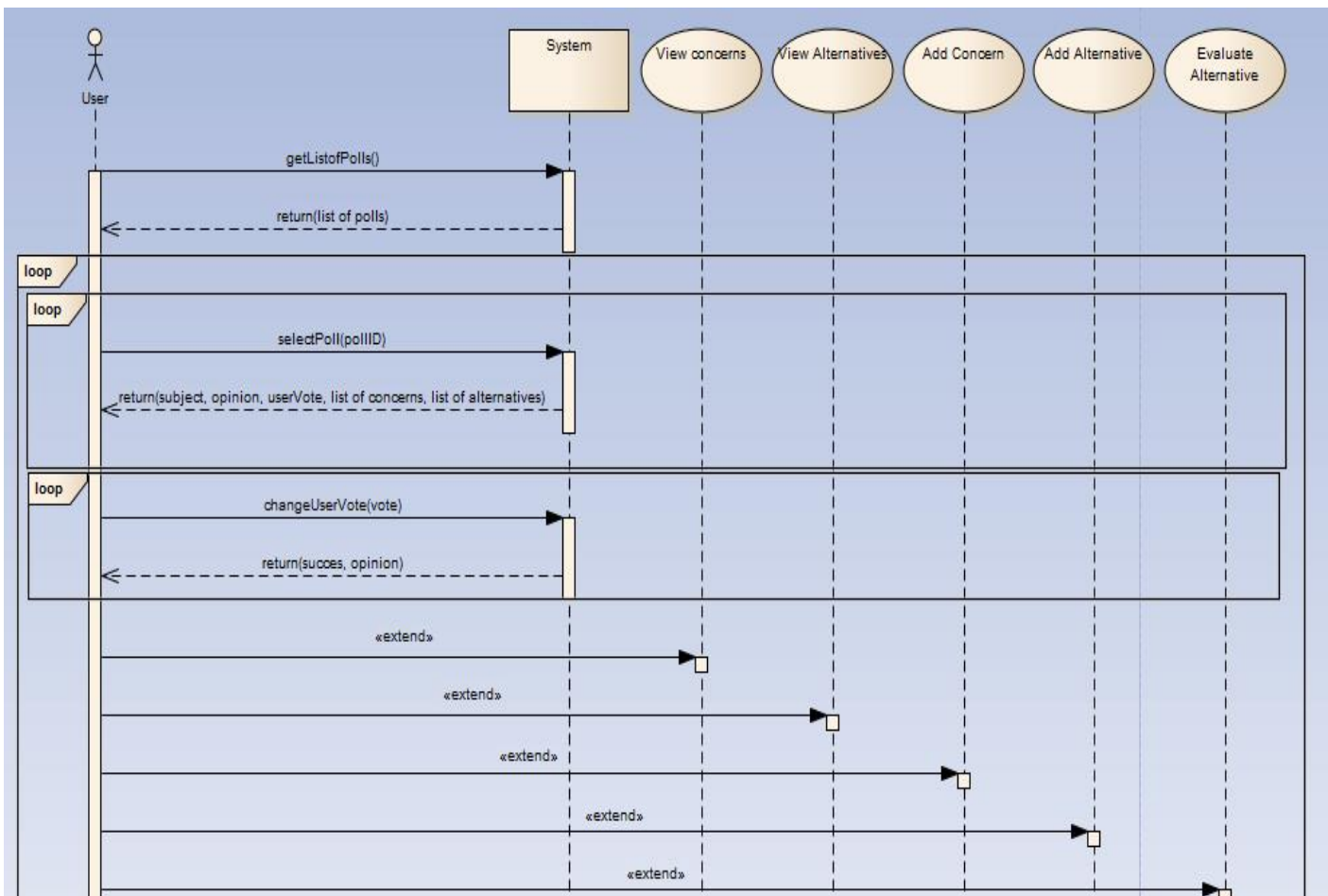


Figure 43: SSD for Vote

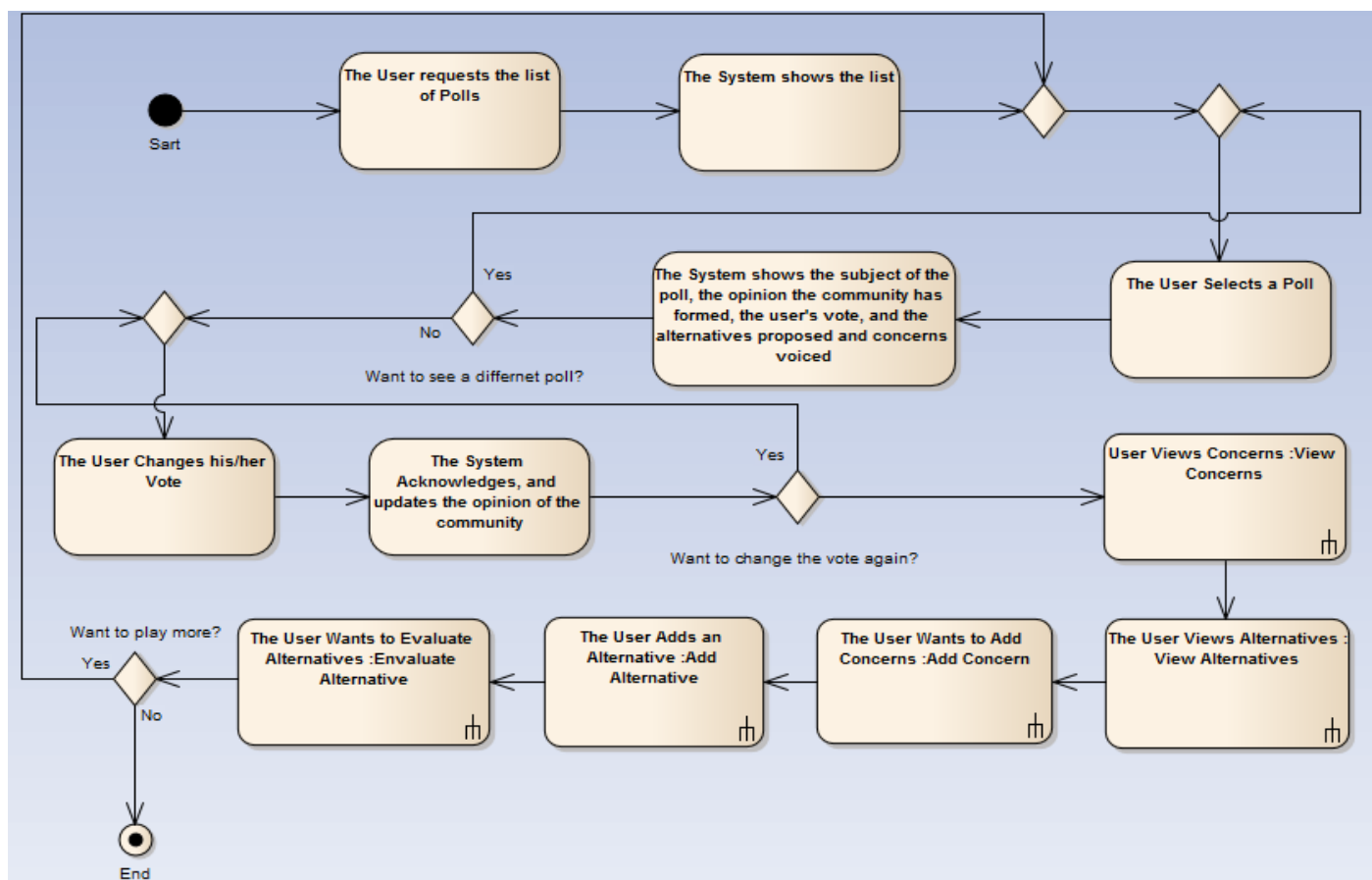


Figure 44: AD for Vote

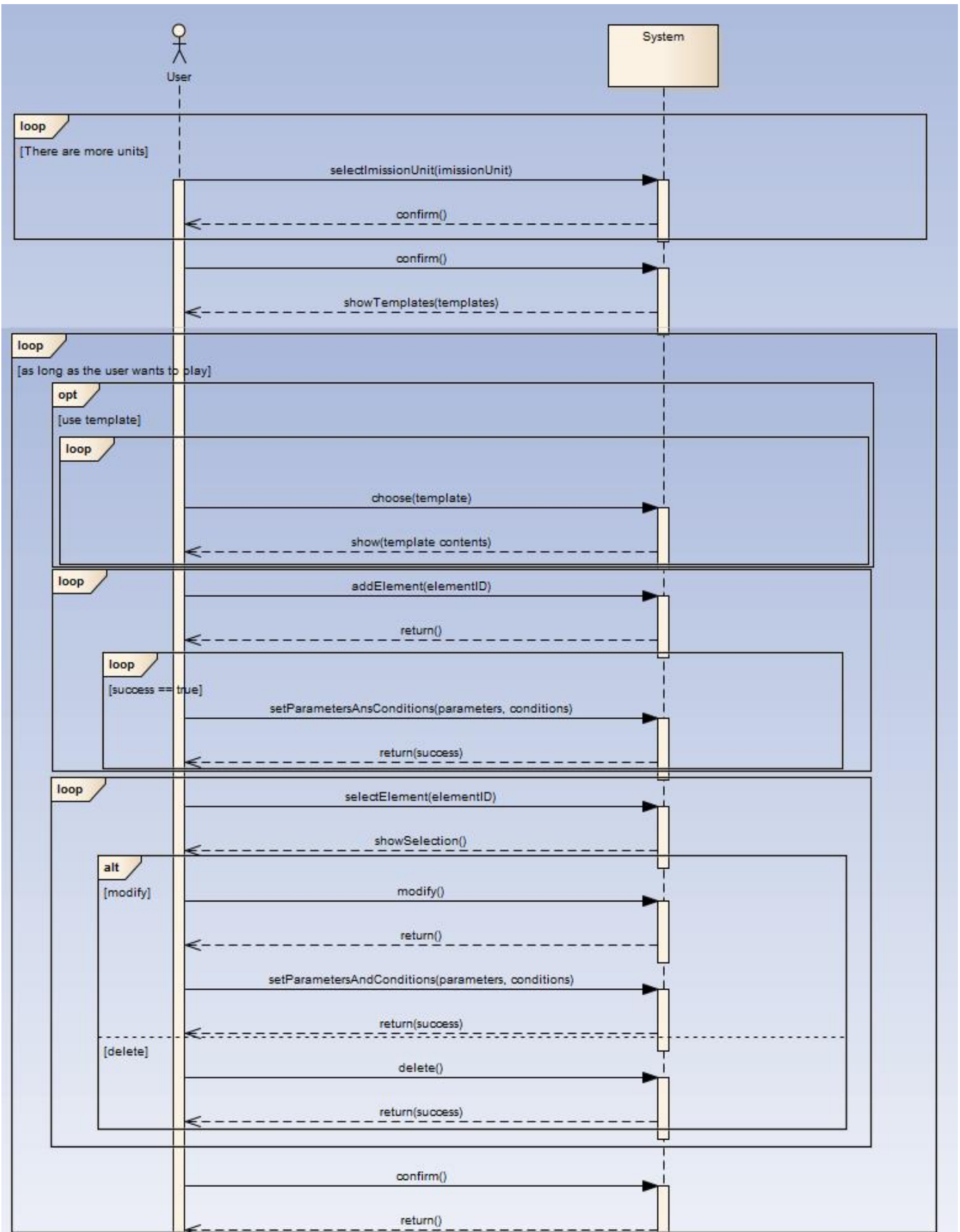


Figure 45: SSD of Make an Offer

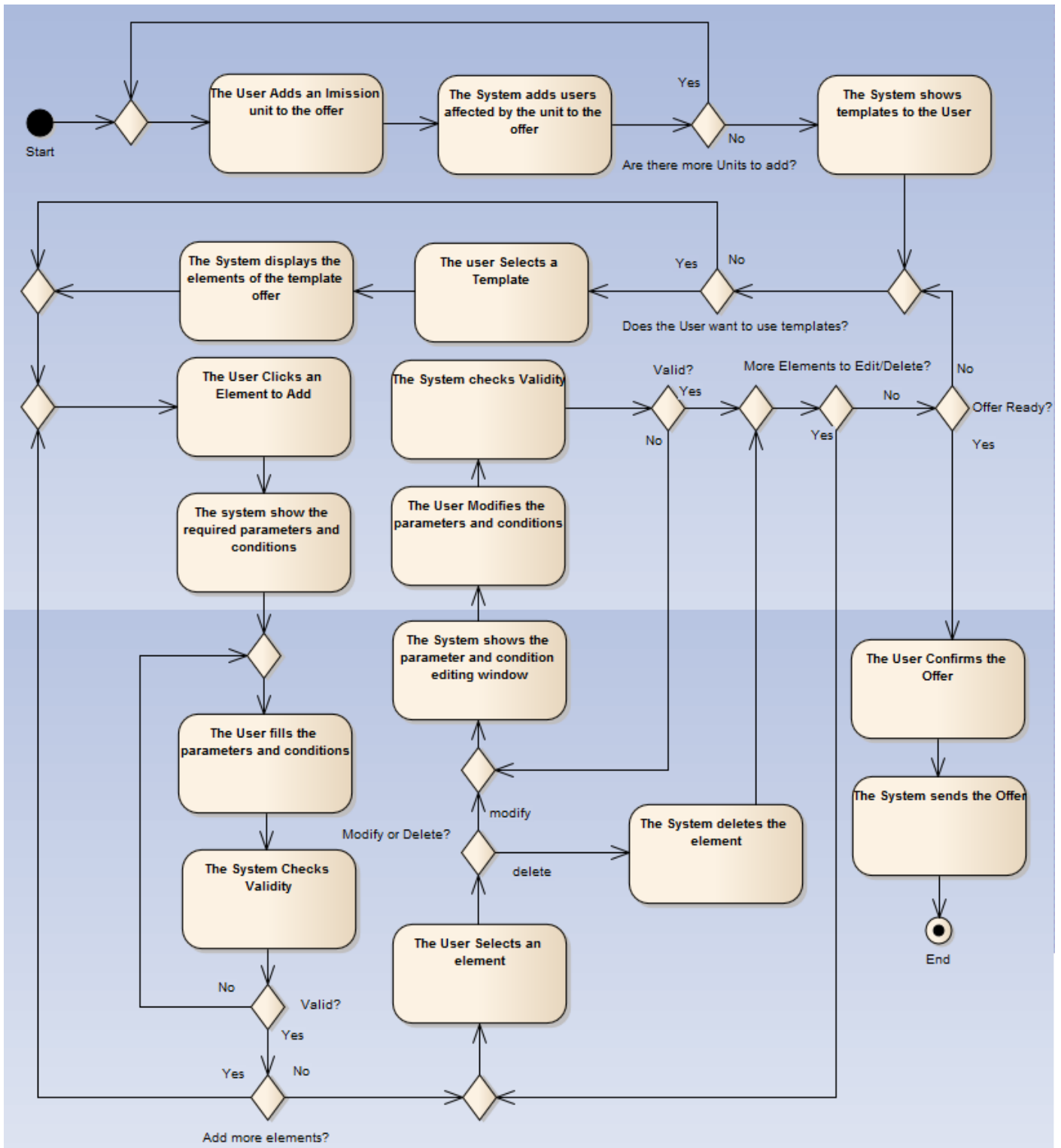


Figure 46: AD of Make an Offer

5. Vision

Introduction:

We envision a new application of information technology in environmental protection, which would not only provide information to the population, but it would also enable interested parties to bargain, strike deals by greatly reducing transaction costs.

Positioning

Business Opportunity

- There are many existing information portals, but none of them provides easily understandable information about pollution or the environment.
- There are numerous initiatives to make information accessible to people, or for other forms of e-Government.

Problem Statement

- The quantity or quality of the data is not the problem, but the method of presentation and the lack of context. (information is not equal to knowledge)
- There is little other use for IT in environmental protection yet.
- No system exists to help the management of environmental problems.
- Even the few existing solutions are disintegrated, hard to use, and largely unknown to common people.

Product Position Statement

- Users: Government agencies (at any level), Companies, Environmental organizations, individuals, or groups of individuals, other software engineers
- Users can get easily accessible, graphical information about the pollution and it's possible effects on them and their environment
- Users can get information about the rights that protect them or others from pollution
- Users can make contact with those that violate their rights
- Users can manage pollution emitted by them
- Users can make deals with each other or by rights to pollute
- Users can monitor and enforce the deal easily
- Users can access information about technologies that can help reduce their emission
- Users can access information about the environmental effects of a product
- Users can easily embed information or features from the system in their own applications

Stakeholders

Government

Companies

Green technology companies

Environmental organizations

Individuals/groups

Key High-Level Goals and Problems of the Stakeholders

Stakeholder	Concerns of the Stakeholder
Government(s)	• The solution needs to be efficient and worth it (positive social benefits) • The solution needs to be popular • The solution shouldn't have negative macroeconomic effects • The solution needs to provide alternative ways to manage pollutants
Regulators	• The system should provide an easy way to administer pollution • The system should aid estimating the effects and dispersion of pollution
Companies	• The cost of making deals should be minimal • The solution should offer ways to minimize risks (opportunism, asset specificity, holdup) • Possibility of using reduction of emission for PR
Green Technology Companies	• The system should provide features that aid discovery of their solutions • The price of advertising their products in the system should be reasonable
Environmental Organizations	• The solution needs to effectively reduce pollution • The solution needs to be transparent, easy to monitor
Individuals	• The solution needs to be easy to use, easy to access • Deal making should be easy, and aided by templates • Information should be understandable • Users want to decide how much information they want • The solution should offer ways to minimize risks (opportunism, asset specificity, holdup) • The solution needs to provide ways to reduce difficulties of bargaining (many parties, opportunism) • The solution needs to be transparent

Product Perspective

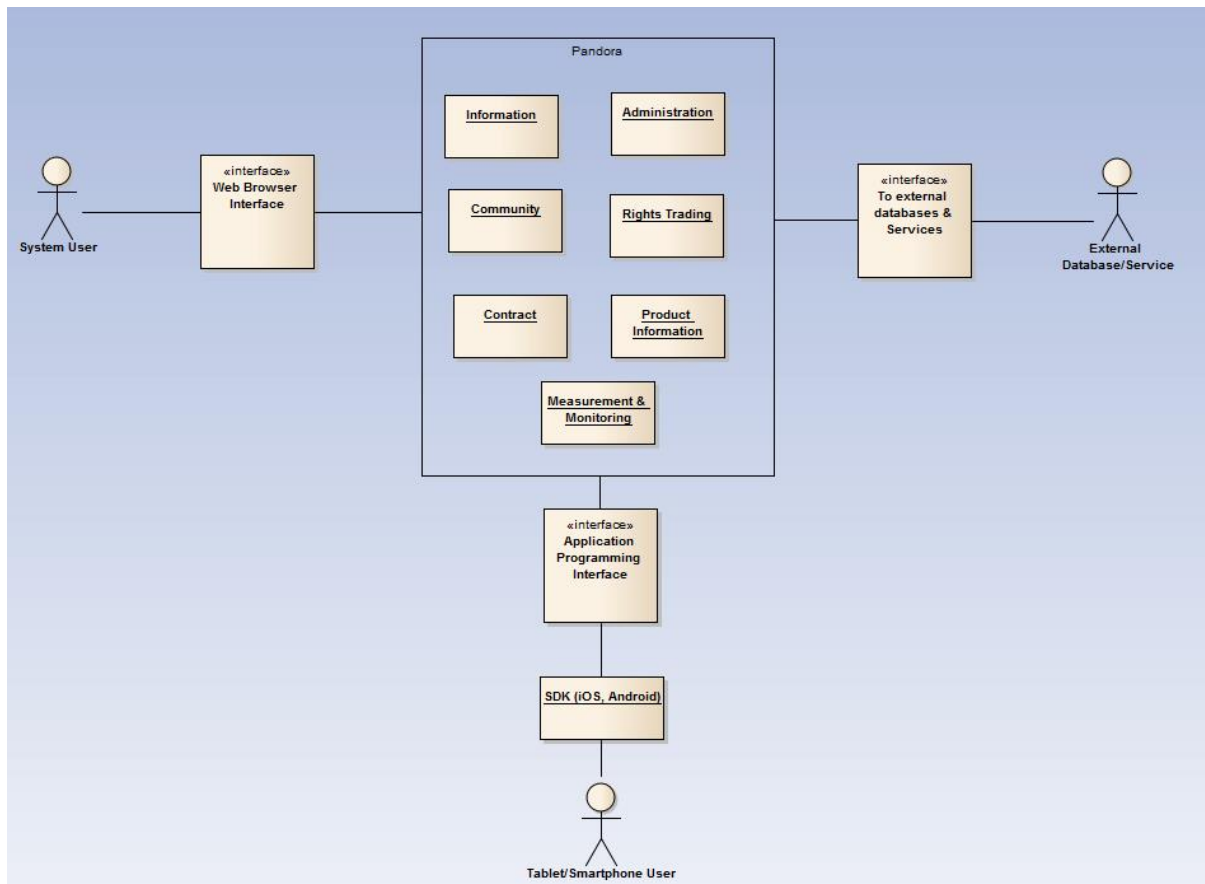


Figure 47: The Product Perspective

Summary of System Features

As discussed below, system features are a terse format to summarize functionality.

- Users can get easily accessible, graphical information about the pollution and it's possible effects on them and their environment
- Users can get information about the rights that protect them or others from pollution
- Users can make contact with those that violate their rights
- Users can manage pollution emitted by them
- Users can make deals with each other or by rights to pollute
- Users can monitor and enforce the deal easily
- Users can access information about technologies that can help reduce their emission
- Users can access information about the environmental effects of a product
- Users can easily embed information or features from the system in their own applications

6. Supplementary Requirement Specification

Functional:

- Many use cases have user authorization (signing in) as their basic requirement. (Exceptions: Use cases in the information module)
- Logging transactions and errors is essential for an easily maintainable system.
- Support for personalization of the user interface and system behavior is required. (Settings specified at each use case)
- At scenario points of some (yet to be defined) use cases arbitrary rules can be defined that execute at that point.
- Often accessed parts (specified by the user) of the system can be collected for quick use. (Favorites menu)

Usability:

- User interfaces for colorblind or visually disabled people. (Possibly easily resizable interface elements, voice over, etc.)
- Simple addition of language sets is required, since the software could be used in numerous different countries.

Reliability:

- If connection to the server fails use cases that do not require immediate response should continue and the messages should be sent at a later time.

Performance:

- Often used data (maps, images, etc.) should be stored locally on the user's computer, and not sent repeatedly over the internet.

Supportability:

- Every once in a while the system would be expanded to handle more pollutants, or changes could be made to the geographical areas. The system must be flexible to such database changes.
- Handling changing legal circumstances should be solved (pluggable rules).
- Frequently asked questions section should be created on the website.
- Support service helping users with problems should be maintained.

Components:

- Air, water and noise pollution modeling software
- Contract building software
- Barcode scanner or image recognition software

Interfaces:

- Public interface for requests from the internet (With authentication required)
- Private interface for notes
- Interfaces for external databases
- Interface for Ügyfélkapu
- Interface for online payment systems (PayPal, etc.)

Legal:

- All features must respect privacy laws. If a user's personal data is used in any way, the user must be prompted to allow or disallow.
- All transactions taking place must take tax laws into account. Also all transactions must be recorded.
- The features of the system must be regularly updated to eliminate conflicts with environmental, or contract law.

7. Glossary

User: Individuals or organizations that use one or more of the system's features.

Community: Group of users that share information or make decisions together.

Emission: A user's aggregate pollution or other use of nature.

Effect Area: A geographical area where the concentration of a pollutant can be considered constant in space.

Natural Capital: Natural capital is the extension of the economic notion of capital (manufactured means of production) to goods and services relating to the natural environment. Natural capital is thus the stock of natural ecosystems that yields a flow of valuable ecosystem goods or services into the future. [Wikipedia]

Dispersion attributes: The geological, meteorological, geographical, etc. attributes of an area that influence the dispersion of pollutants.

Imission: The aggregate concentration of all pollutants in an area.

Emission unit: The unit of a given pollutant that is defined by law (the unit used by tradable permits), or the magnitude of a pollutant/use of nature emitted by one user in one area.

Imission unit: The concentration caused by one emission unit, in one area.

Unit Effect: The effect of one imission unit.

Aggregate Effect: The aggregate of unit effects in one area.

Unit Damage: The money-value of one unit Effect.

Aggregate Damage: The aggregate of Unit Damages in one area.

Possible effect: An effect that one or more pollutants may cause.

Protection right: A right that is not directly tradable.

Tradable right: Emission or Use permit that is tradable.

Contract: Legal contract between two or more users that defines the rules and conditions of their use of nature.

8. Use Case Descriptions

Information

Table 74: Use Cases of the Information Module

Registration: Easy 1. Loop: 1. The user enters the user name, and e-mail address, and user type, and other data 2. The system validates data, and notifies the user 2. The user confirms 3. The system notifies the user of success, and registers a new user entry	Login: Easy 1. The user enters user name and password 2. The system validates data, and returns success
View Favorites: Easy 1. The user clicks on favorites 2. The system shows links to the user's favorite functions	Add to Favorites: Easy 1. The User selects a function and clicks "add to favorites" 2. The system adds the function to the list, and confirms
View User Profile: Average 1. The user selects another user. 2. The system shows the user's main public profile page 3. Loop: 1. The user clicks on an interactive element 2. The system displays the interaction 4. Loop: 1. The user clicks on a navigation item 2. The system shows the destination page of the navigation item	User Search: Average 1. User selects search terms and conditions 2. The system returns <i>public</i> users that match conditions 3. Loop: 1. The user changes ordering rules 2. The system changes the order of the results 4. Loop: 1. The user adds or modifies a filter value 2. The system selects results that match the filter and displays them 5. Loop: 1. The user modifies search terms and conditions 2. The system returns the updated list of matches

Emission Data Search: Complex 1. The user selects search terms and conditions 2. The system sends a request to the pollution databases 3. The database returns emissions 4. The system returns emissions 5. Loop: 1. The user changes ordering rules 2. The system changes the order of the results 6. Loop: 1. The user adds or modifies a filter value 2. The system selects results that match the filter and displays them 7. Loop: 1. The user modifies search terms and conditions 2. The system sends a request to the pollution databases 3. The database returns emissions 4. The system returns the updated list of matches 8. The user selects an emission 9. The system sends the request to the database 10. The database returns raw data 11. The system returns raw data	Emission Data Search: Average 1. The user selects search terms and conditions 2. The system returns emissions 3. Loop: 1. The user changes ordering rules 2. The system changes the order of the results 4. Loop: 1. The user adds or modifies a filter value 2. The system selects results that match the filter and displays them 5. Loop: 1. The user modifies search terms and conditions 2. The system returns the updated list of matches 6. The user selects an emission 7. The system returns raw data
Share: Average 1. The user selects a part of the site, and clicks share 2. The system shows the list of social network sites the user is connected to 3. The user selects one and confirms 4. The system sends the share request to the selected site 5. The site returns success 6. The system returns success	View Pollutant Information: Easy 1. The user selects a pollutant 2. The system returns the information available on it 3. Loop: 1. The user selects an interactive element 2. The system plays the interaction

Identify Causes of Pollution: Easy 1. The user selects an imission unit 2. The system returns the emission unit causing it and the user that the emission unit belongs to 3. Loop: 1. The user clicks on an element 2. The system shows in-depth information on it	Compare Areas: Average 1. Loop: 1. The user selects an area 2. The system displays the selection 3. The user clicks “compare” 4. The system checks whether the user reached the maximum number of areas, and returns success 5. The user views information on all areas (invokes View Area Information)
View Area Information: Complex 1. Loop: 2. The user moves above an area on the map interface 3. The system sends requests to the Pollution and the Environmental Database 4. The databases return raw data (imission, areas, and natural capital) 5. The system assembles map data, and presents it to the user 6. Loop: 1. The user selects an area 2. The system assembles the default information layer, and presents it to the user 3. Loop: 1. The user changes the information layer 2. The system assembles the information layer the user asks for, and shows it to the user 3. Loop: 1. The user clicks an interactive element 2. The system plays the interaction 4. Loop: 1. The user clicks on an element 2. The system displays in-depth data on the element	Search Places: Complex 1. The user enters search terms and conditions 2. The system sends the request to Spatial Database(s) 3. The Database(s) return the matches 4. The system returns the list of the places that match the terms and conditions 5. Loop: 1. The user changes ordering rules 2. The system changes the order of the results 6. Loop: 1. The user adds or modifies a filter value 2. The system selects results that match the filter and displays them 7. Loop: 1. The user modifies search terms and conditions 2. The system sends the request to Spatial Database(s) 3. The Database(s) return the matches 4. The system returns the updated list of matches

Contact Support: Average 1. The user selects "contact support" 2. The system shows the support request form to the user 3. Loop: 1. The user fills the form 2. The system checks validity 4. The user confirms 5. The system returns success, and notifies the support team	Report a Problem: Average 1. The user selects "report a problem" 2. The system shows the bug reporting form to the user 3. Loop: 1. The user fills the form 2. The system checks validity 4. The user confirms 5. The system returns success, and notifies the support team
View Emission: Average 1. Loop: 1. The user selects an emission 2. The system sends request to the Pollution Database 3. The database returns raw data on the emission 4. The system assembles the default information layer, and presents it to the user 5. Loop: 1. The user changes the information layer 2. The system assembles the information layer the user asks for, and shows it to the user 3. Loop: 1. The user clicks an interactive element 2. The system plays the interaction 4. Loop: 1. The user clicks on an element 2. The system displays in-depth data on the	

Administration

Table 75: Use Cases of the Administration Module

Connect Account to Ügyfélkapu: Easy 1. The user selects "connect account to Ügyfélkapu" 2. The system redirects the user to the Ügyfélkapu login page 3. The login page returns success 4. The system adds the Ügyfélkapu account to the user's account, and confirms success	Register New Emission: Average 1. The user selects a user and an area 2. The system returns whether the user can register emission in the given area 3. Loop: 1. The user selects a pollutant and a magnitude 2. The system checks consistence, and returns success 4. The user confirms 5. The system sends the emission data to the pollution database 6. The database returns success 7. The system notifies an administrator, and returns success
Edit Profile: Average 1. User clicks on profile 2. The system shows the user's private profile page 3. Loop: 1. The user selects an entry, and clicks edit 2. The system presents an editing window 3. The user enters new values, and confirms 4. The system checks validity, and notifies the user 4. The user confirms 5. The system saves the new profile page, and confirms	View Owned Rights: Average 1. The user selects view owned rights 2. The system return rights 3. Loop: 1. The user selects a right 2. The system send request to the Pollution Database 3. The database returns emissions near the rights' areas 4. The system returns contracts based on the right (if contract module is there), and emissions violating the right

Edit User's Rights: Complex 1. The administrator selects a user 2. The system returns the rights associated with the user 3. [The administrator selects a right] 1. [The administrator selects delete right] 1. The system removes the right from the user, and notifies the administrator, and the user 2. If the contract module is present, the system also returns the contracts invalidated by the removed right 2. [The administrator selects edit right] 1. The administrator modifies the properties 2. The system will update the right, and notify the administrator and the user 3. If the contract module is present, the system also returns the contracts invalidated by the edited right 4. [The administrator selects add new right] 1. The administrator adds properties, and confirms 2. The system registers the new right, notifies the administrator, and the user	Calculate Imission: Complex 1. The administrator selects an emission, and confirms 2. The system sends request for transmission and propagation data to Spatial/Geological and Meteorological Databases 3. The Databases return the data 4. The system calculates the imission and shows it to the administrator 5. The administrator confirms 6. The system registers the new imission, and notifies the administrator
Estimate Effect: Complex 1. The administrator selects the imissions, and confirms 2. The system sends request for natural capital data to Environmental Databases 3. The Databases return the data 4. The system calculates the effects and shows them to the administrator 5. The administrator confirms 6. The system registers the new effects, and notifies the administrator	Estimate Damage: Complex 1. The administrator selects the effects, and confirms 2. The system estimates the damage, and shows it to the administrator 3. The administrator confirms 4. The system registers new damage, and notifies the administrator

Community

Table 76: Use Cases of the Community Module

Invitation: Easy 1. The user selects another user 2. The system checks if the user can be added to the community, sends an invitation message, and returns success	Application: Easy 1. The user selects a community, and clicks “apply to the community” 2. The system checks if the user can be added to the community, and displays the application form 3. Loop: 1. The user fills the form 2. The system checks for validity 4. The user confirms 5. The system notifies the user of success, and the community of the application
Search for Nearby Communities: Average 1. The user selects search nearby communities 2. The system returns communities in the areas associated with the user 3. Loop: 1. The user changes ordering rules 2. The system changes the order of the results 4. Loop: 1. The user adds or modifies a filter value 2. The system selects results that match the filter and displays them 5. Loop: 1. The user modifies search terms and conditions 2. The system returns the updated list of matches	Form a Community: Average 1. The user enters the name and description of the community 2. The system acknowledges 3. Loop: 1. The user selects purpose of the community, and adds conditions on membership 2. The system checks eligibility, and notifies the user 4. Loop: 1. The user selects a circle to send invitation to 2. The system checks permissions, and returns success 5. The user confirms 6. The system registers new community, and sends invitations
Create a Poll: Average 1. User selects the poll template, and the circle to poll 2. The system checks if the user has the permission to create such a poll in the given circle, and returns the template	Vote/Make a Decision: Average 1. The user selects polls 2. The system returns all the polls the user can participate in 3. The user selects a poll

3. Loop: 1. The user enters missing information into the template 2. The system checks validity, and notifies the user 4. The user confirms 5. The system registers the poll, notifies other users, and returns success	4. The system checks if the user has already voted, and returns 5. The user fills all required fields, and confirms 6. The system registers the vote
Donate: Average 1. The user selects group, or donation request 2. The user selects amount, and confirms 3. The system checks if the user has a valid payment method 4. The system sends the request to the Online Payment System, and redirects the user 5. The OPS returns success 6. The system notifies the user	Ask for Donation: Easy 1. The user enters donation message, selects default amount, and the circle to send the donation (friends, members, people in the area, etc...) 2. The system checks, if the user or group has a bank account associated with, and sends the donation request
Send a Message: Complex 1. The user selects another user 2. The system confirms, and shows message editing UI 3. Loop: 1. Loop: 1. The user enters the message 2. The system confirms 2. Loop: 1. The user formats the message 2. The system changes the format 3. Loop: 1. The user selects attach file 2. The system shows file selection window 3. The user selects a file 4. The system attaches the file to the message 4. The user confirms 5. The system checks validity, and sends the message, and notifies the user of success	Connect to Social Network Account: Average 1. The user selects “connect to a social network” 2. The system shows the list of Social Networks the user isn’t already connected to 3. The user selects one, and confirms 4. The system redirects the user to the selected login page 5. The login page returns success 6. The system adds the Social Network account to the user’s account, and confirms success
Read a Message: Easy 1. The user enters “inbox” 2. The system presents the messages sent to the user 3. Loop: 1. The user selects a message	View Forum Thread: Average 1. The user selects a forum category 2. The system displays threads in the category 3. The user selects a thread 4. The system displays messages in the first page of the thread

2. The system shows the contents of the message	5. Loop: 1. The user selects a page 2. The system displays the messages in the selected page
Reply to Thread: Complex 1. The user clicks reply 2. The system check privileges, and shows message editing UI 3. Loop: 1. Loop: 1. The user enters the message 2. The system confirms 2. Loop: 1. The user formats the message 2. The system changes the format 3. Loop: 1. The user selects attach file 2. The system shows file selection window 3. The user selects a file 4. The system attaches the file to the message 4. The user confirms 5. The system adds the new message to the thread and returns	Open New Thread: Average 1. The user clicks add new forum thread 2. The system checks privileges and returns 3. The user enters the category name, thread name, then confirms 4. The user enters first reply (invokes reply to thread) 5. The system checks validity, and adds the new thread to the category, then returns
Send Invitation to the System: Average 1. The user selects “invite to the system” 2. The system asks the user to enter the invite-e’s e-mail address, or Social Network account name 3. The user enters the required data, and an optional message, and confirms 4. The system sends a request to the Social Network 5. The S.N. returns success 6. The system returns success	Find Friends: Easy 1. The user selects find friends 2. The system returns the list of the user’s friends in other S.N. sites who have accounts in the system
Add New Friend: Easy 1. The user selects another user, and clicks “add friend”	View Concerns: Average 1. Loop: 1. Loop:

2. The system notifies the other user, and returns	1. The user selects a concern 2. The system displays the concern 2. Loop: 1. The user changes his vote value 2. The system registers the new value, and confirms
Add New Concern: Average 1. The user presses “add concern” 2. The confirms 3. Loop: 1. The user enters the title and message 2. The system checks validity, and returns success 4. The user confirms 5. The system updates the vote page with the new concern, and returns	View Alternatives: Average 1. Loop: 1. Loop: 1. The user selects an offered alternative 2. The system displays the alternative 2. Loop: 1. The user changes his vote value 2. The system registers the new value, and confirms
Add New Alternative: Complex 1. The user selects “add alternative” 2. The system shows the interface that belongs to the poll type 3. [poll type == multiple choice] 1. Loop: 1. The user enters the text of the alternative 2. The system checks validity and confirms 2. The user confirms 3. The system adds the alternative, and returns success 4. [poll type == offer] 1. The user makes a new offer (extends Make Offer), and confirms 2. The system adds the new offer, and confirms 5. [poll type == simple yes or no] 1. The system returns error	Evaluate Alternatives: Average 1. The user selects an alternative 2. The system confirms 3. The user tells the system to evaluate the alternative 1. [user has not set preferences] 1. The user sets preferences (extends Set Preferences) 4. The system returns the goodness of the alternative, and proposed modifications
Set Preferences: Average 1. The user tells the system to set or modify his/her preferences 2. The system returns 3. Loop: 1. The user selects an item	

2. The system displays value, and possibilities 3. Loop: 1. The user sets a new value/chooses another possibility 2. The system checks validity and confirms 4. The user confirms 5. The system updates preferences, and confirms	
--	--

Rights Trading

Table 77: Use Cases of the Rights Trading Module

View Missing Rights: Average 1. The user selects view missing rights 2. The system send request to the EES 3. The EES returns permits owned by the user 4. The system matches permits with emissions, and notifies the users of emissions that do not have a matching permit 5. Loop: 1. The user clicks on a missing right 2. The system sends a request to the EES 3. The EES returns data on the right 4. The system displays data on the missing permit	Search for Tradable Rights: Average 1. The user selects search terms and conditions 2. The system rends request to the EES 3. The EES returns the rights that match the search 4. The system shows the rights 5. Loop: 1. The user changes ordering rules 2. The system changes the order of the results 6. Loop: 1. The user adds or modifies a filter value 2. The system selects results that match the filter and displays them 7. Loop: 1. The user modifies search terms and conditions 2. The system returns the updated list of matches
Buy Right: Easy 1. The user selects a right, and clicks buy 2. The system sends the request to the EES 3. The EES returns success 4. The system returns success	Sell Right: Easy 1. The user selects a right, and clicks sell 2. The system sends the request to the EES 3. The EES returns success 4. The system returns success

Monitor Rights: Average 1. The user selects "monitoring" 2. The system returns, and shows real-time data on the monitored rights 3. Loop: 1. The user changes filters 2. The system hides and shows rights 4. Loop: 1. The user changes ordering 2. The system rearranges rights	Add New Rights to the Monitoring: Easy 1. The user selects a right, and clicks monitor 2. The system adds the right to the list of monitored rights, and confirms
---	--

Contract

Table 78: Use Cases of the Contract Module

Make Offer: Complex 1. Loop: 1. The user selects an imission unit to add to the offer 2. The system checks eligibility and adds users in the area to the offer 2. The user confirms 3. The system shows contract templates to the user 4. The user chooses a template 5. The system shows its detailed contents 6. Loop: 1. The user adds/deletes/modifies an element 2. The system checks validity, and modifies the offer 3. Loop: 1. The user adds/modifies deletes a rule 2. The system checks validity, and modifies the offer 7. The user confirms 8. The system sends the offer, and notifies the user	View Offer: Average 1. The user selects view negotiations 2. The system shows the list of negotiations the user has 3. The user selects a negotiation 4. The system returns the list of offers 5. The user selects one 6. The system shows the contents of the offer 7. Loop: 1. The user selects an element 2. The system displays information on the element
View Contract: Average	Conclude Contract: Easy

1. The user selects view contracts 2. The system returns concluded contracts 3. The user selects one 4. The systems shows the contract 5. Loop: 1. The user clicks an element 2. The system displays data on the element	1. The user selects a negotiation, and confirms to accept current terms 2. The system checks whether the contract is valid, and notifies the users, and the legal support
View Violated Rights: Easy 1. The user clicks “violated rights” 2. The system returns emissions and imissions with no contract to cover them 3. Loop: 1. The user clicks on an emission 2. The system displays data on the emission (invokes View Emission)	Get Legal Help: Average 1. The user selects “get legal help” 2. The system shows the support request form to the user 3. Loop: 1. The user fills the form 2. The system checks validity 4. The user confirms 5. The system returns success, and notifies a legal support member
Review Contract: Complex 1. The legal support member opens the contract list 2. The system shows contracts that need reviewing 3. The legal support member opens a contract 4. The system shows the contract 5. The legal support member tells the system to finish review 6. The system shows the resolution form 7. Loop: 1. The legal support member fills the form 2. The system checks validity, and returns success 3. The legal support member makes an alternative (invokes make offer) 4. The system checks for validity and returns 8. The legal support member confirms 9. The system notifies the users on the contract of the decision	Make New Template: Complex 1. The legal support member adds new contract template 2. The system shows the template creation form 3. Loop: 1. The legal support member fills the form 2. The system checks for validity and returns 4. Loop: 1. The legal support member makes a new contract (invokes Make Offer) 2. The system generates text document from the contract, and shows it to the legal support member 5. The legal support member modifies the contract, and confirms 6. The system adds the new template, and returns success

Product Information

Table 79: Use Cases of the Product Information Module

Identify Product: Easy 1. The user presses “camera button” 2. The system checks if the device has a camera, and notifies the user 3. The user shows the product to the camera 4. The system identifies the product, and shows the product page to the user	Search Products: Average 1. The user selects search terms and conditions 2. The system returns the list of matching products 3. Loop: 1. The user changes ordering rules 2. The system changes the order of the results 4. Loop: 1. The user adds or modifies a filter value 2. The system selects results that match the filter and displays them 5. Loop: 1. The user modifies search terms and conditions 2. The system returns the updated list of matches
Add New Green Technology: Complex 1. The user presses “add green technology” 2. The system checks the users permissions, and returns 3. Loop: 1. The user chooses a product page template 2. The system displays the template 4. Loop: 1. The user adds or modifies elements on the product page 2. The system modifies the product page 5. Loop: 1. The user clicks “add a new page”, and chooses a template 2. The system adds a new page to the product information 6. Loop: 1. The user adds a new interactive element 2. The system checks if the user is eligible (only available for premium users), and returns 7. The user confirms	Add New Product: Complex 1. The user presses “add product” 2. The system checks the users permissions, and returns 3. Loop: 1. The user chooses a product page template 2. The system displays the template 4. Loop: 1. The user adds or modifies elements on the product page 2. The system modifies the product page 5. Loop: 1. The user clicks “add a new page”, and chooses a template 2. The system adds a new page to the product information 6. Loop: 1. The user adds a new interactive element 2. The system checks if the user is eligible (only available for premium users), and returns 7. The user confirms

8. The system registers new technology, and notifies the user	8. The system registers new product, and notifies the user
Search Green Technologies: Average 1. The user selects search terms and conditions 2. The system returns the list of matching technologies 3. Loop: 1. The user changes ordering rules 2. The system changes the order of the results 4. Loop: 1. The user adds or modifies a filter value 2. The system selects results that match the filter and displays them 5. Loop: 1. The user modifies search terms and conditions 2. The system returns the updated list of matches	View Product Information: Average 1. The user selects a product 2. The system shows the product page to the user 3. Loop: 1. The user selects an interactive element on the page 2. The system shows the interactive element to the user 4. Loop: 1. The user clicks on a navigation item 2. The system loads the destination page
View Green Technology: Average 1. The user selects a green technology 2. The system shows the technology page to the user 3. Loop: 1. The user selects an interactive element on the page 2. The system shows the interactive element to the user 4. Loop: 1. The user clicks on a navigation item 2. The system loads the destination page	

Measurement & Monitoring

Table 80: Use Cases of M&M Module

Request Monitoring: Average 1. The user selects "request monitoring" 2. The system shows the monitoring request form to the user 3. Loop:	Monitor Area: Complex 1. The administrator starts area check 2. Loop: 1. The system collects measurement data
--	--

1. The user fills the form 2. The system checks validity 4. The user confirms 5. The system returns success, and notifies an administrator	2. The system requests meteorological data from the time of the measurements 3. The Meteorological database returns the data 4. The system calculates the most probable distribution of emissions, and checks whether it matches the emission data 3. The system notifies the administrator if the results don't match the data
Send Measurement: Easy 1. The Mote sends id, location, time and measurement data to the system 2. The system checks validity, and acknowledges	

9. Diagrams for Vote Use Cases

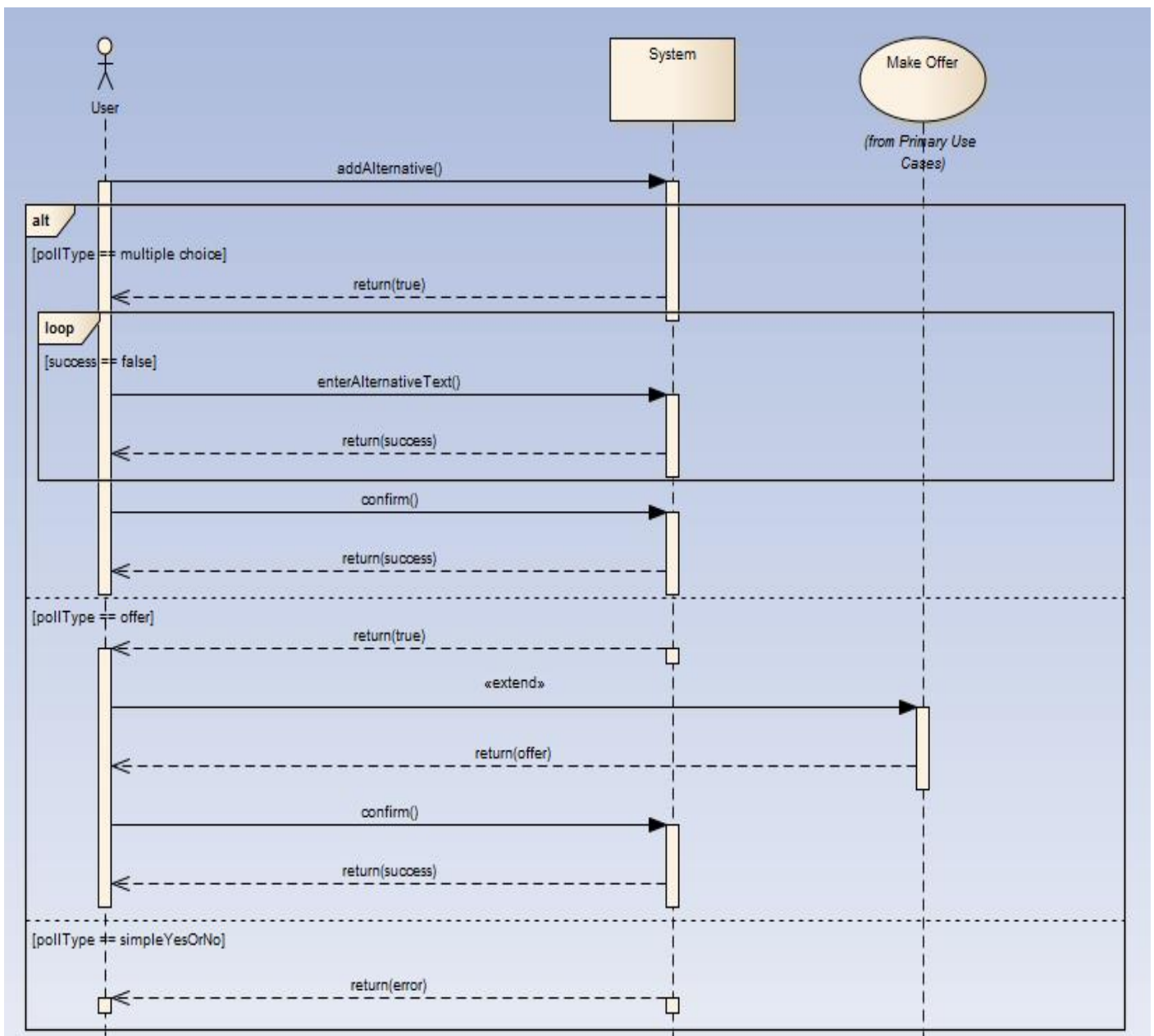


Figure 48: SSD for Add Alternative

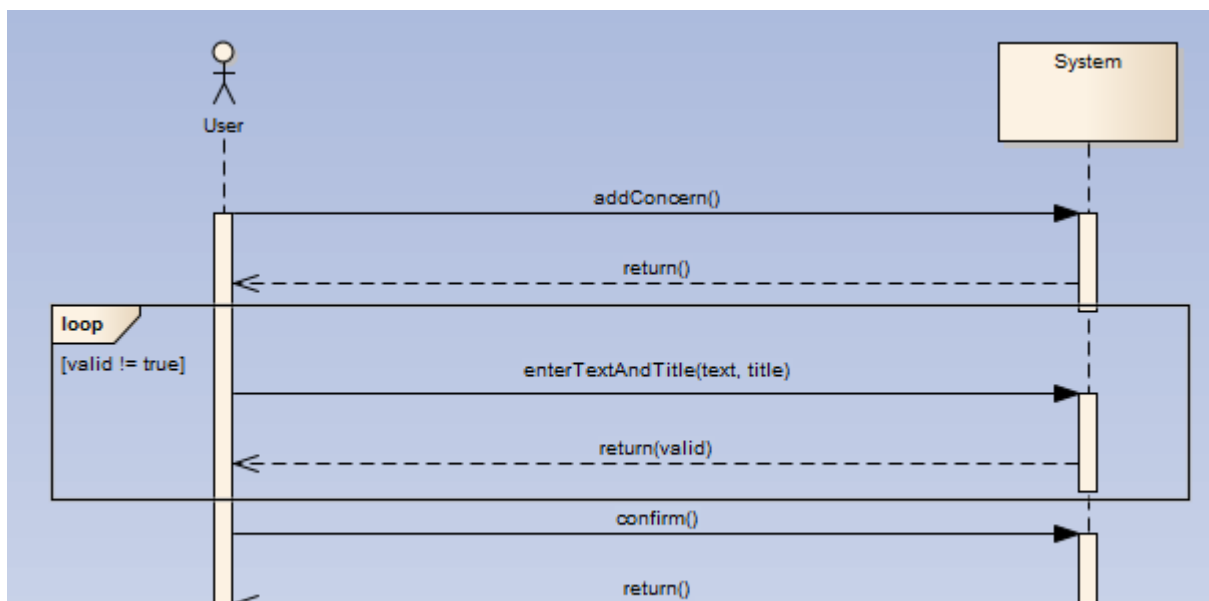
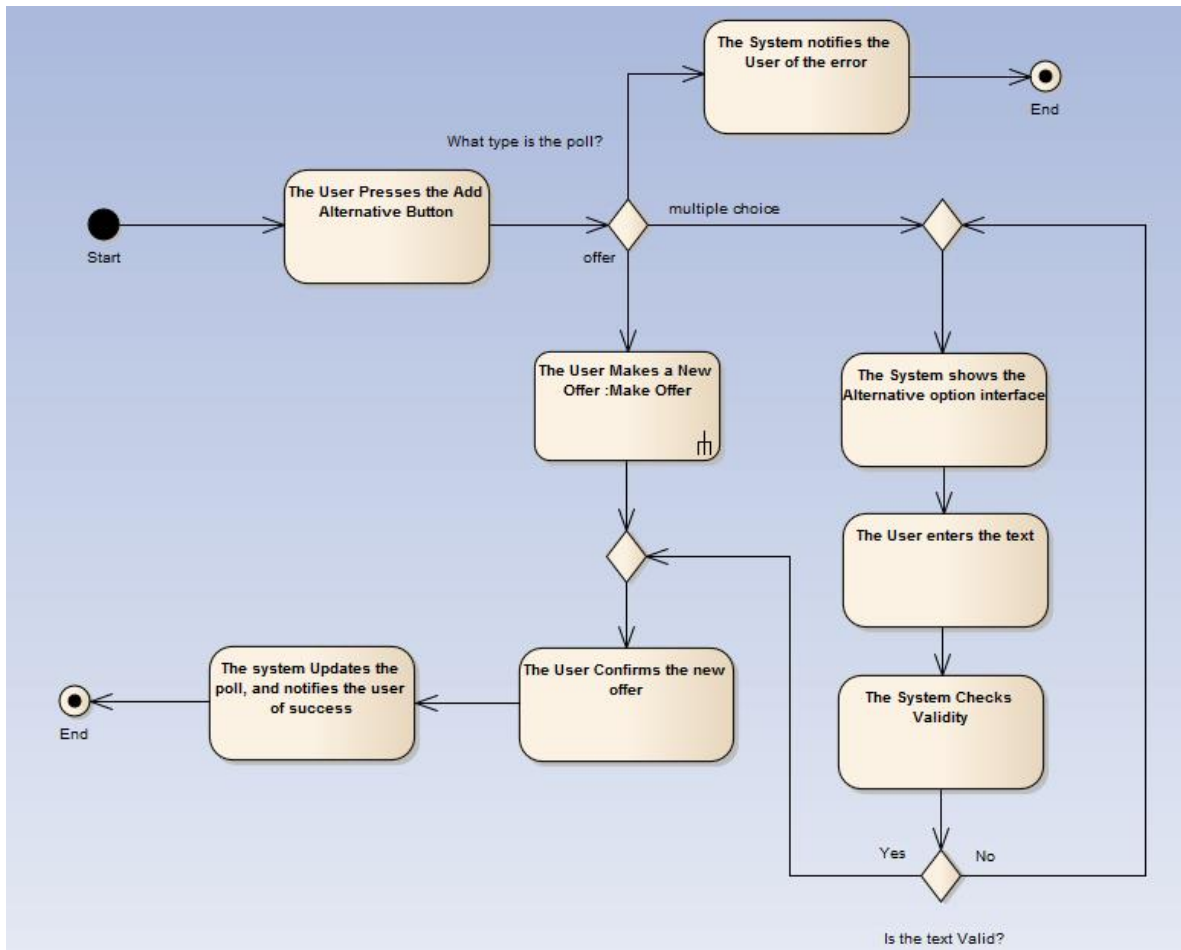


Figure 50: SSD for Add Concern

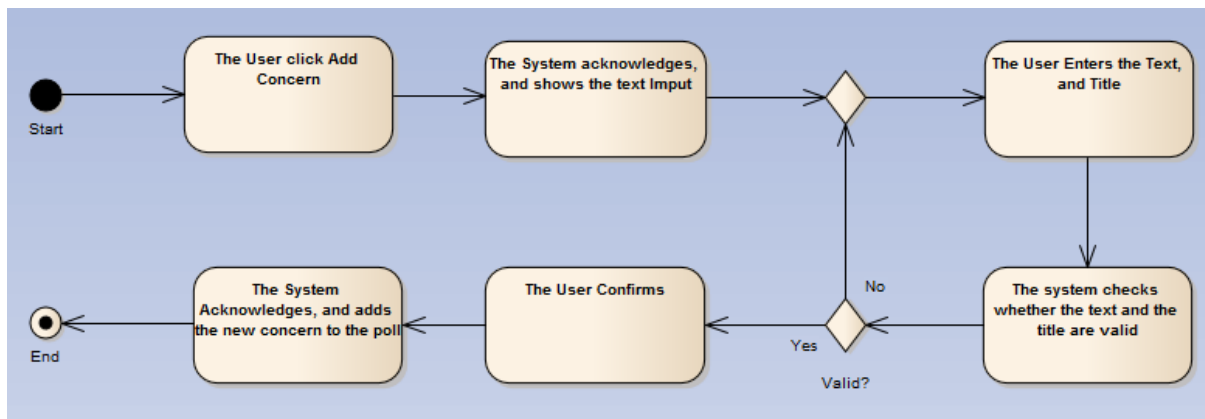


Figure 51: AD for Add Concern

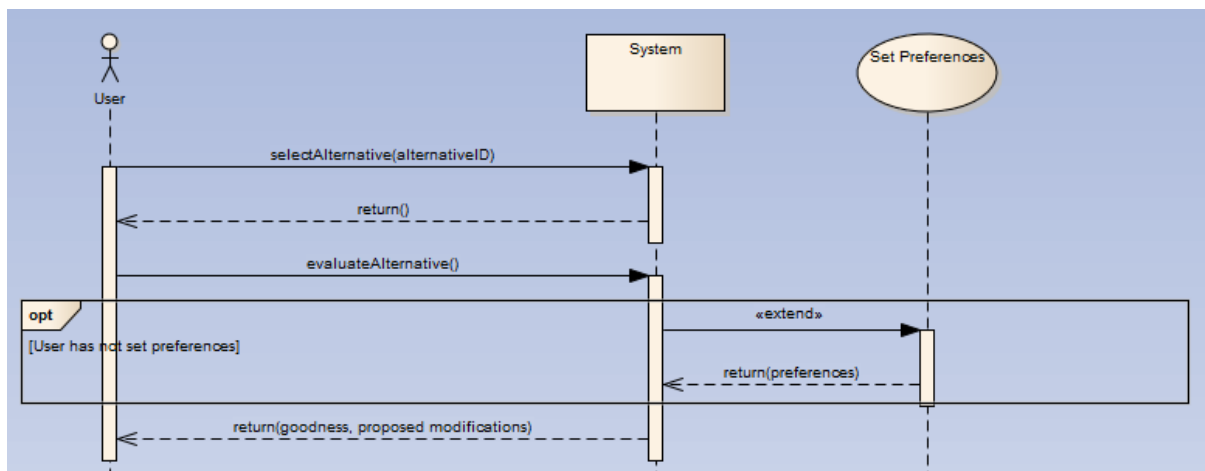


Figure 52: SSD for Evaluate Alternative

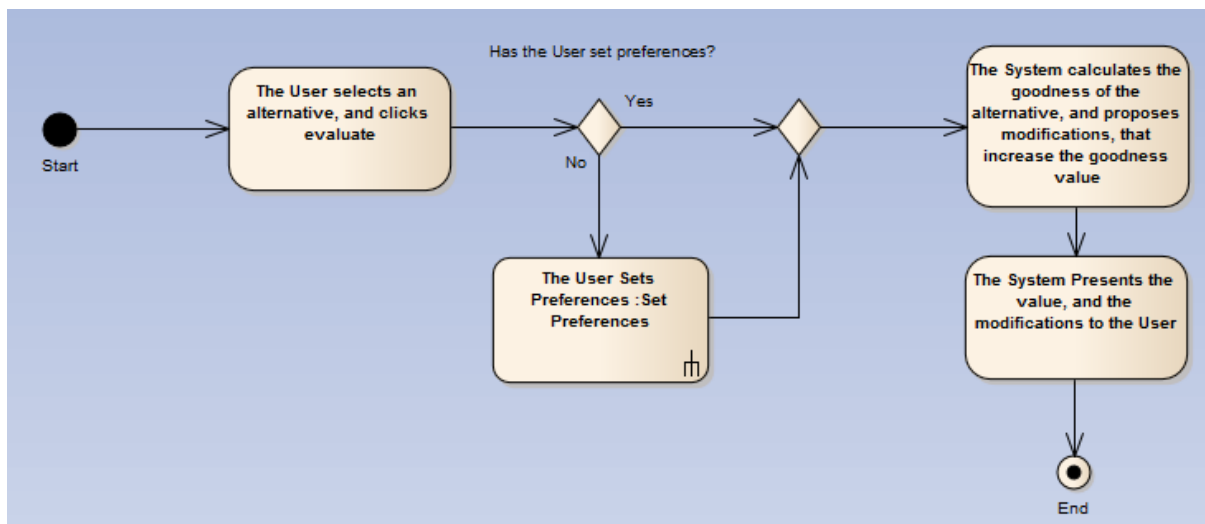


Figure 53: AD for Evaluate Alternative

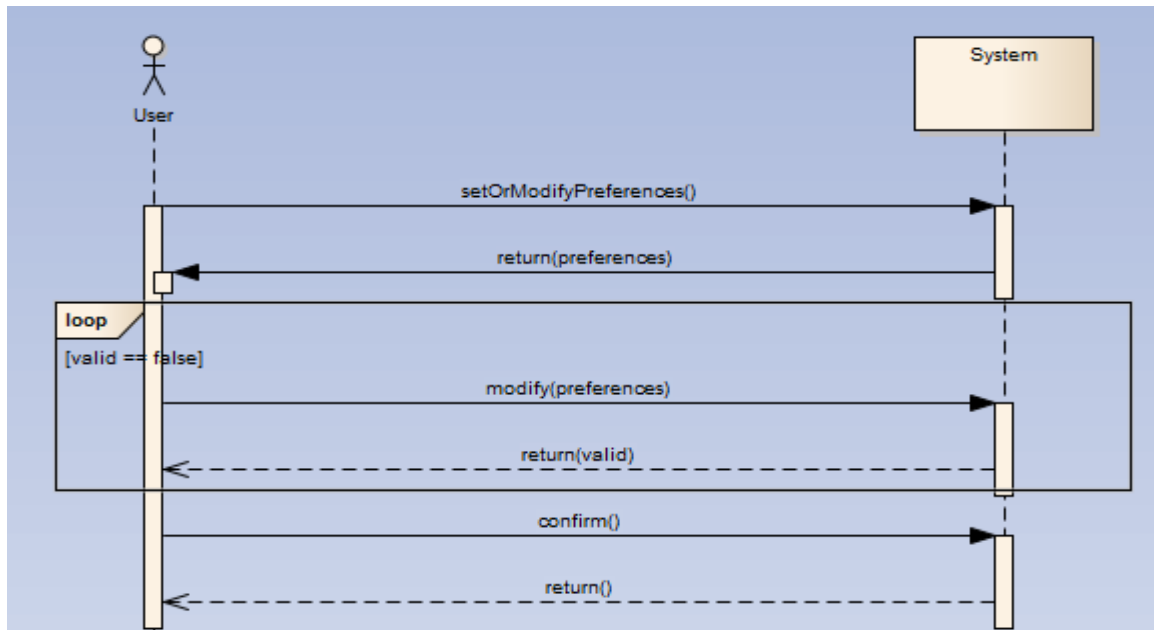


Figure 54: SSD for Set Preferences

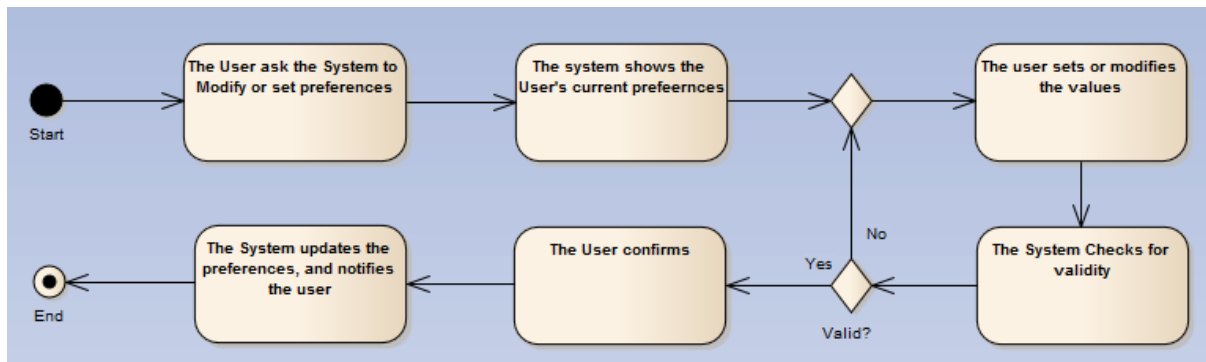


Figure 55: AD for Set Preferences

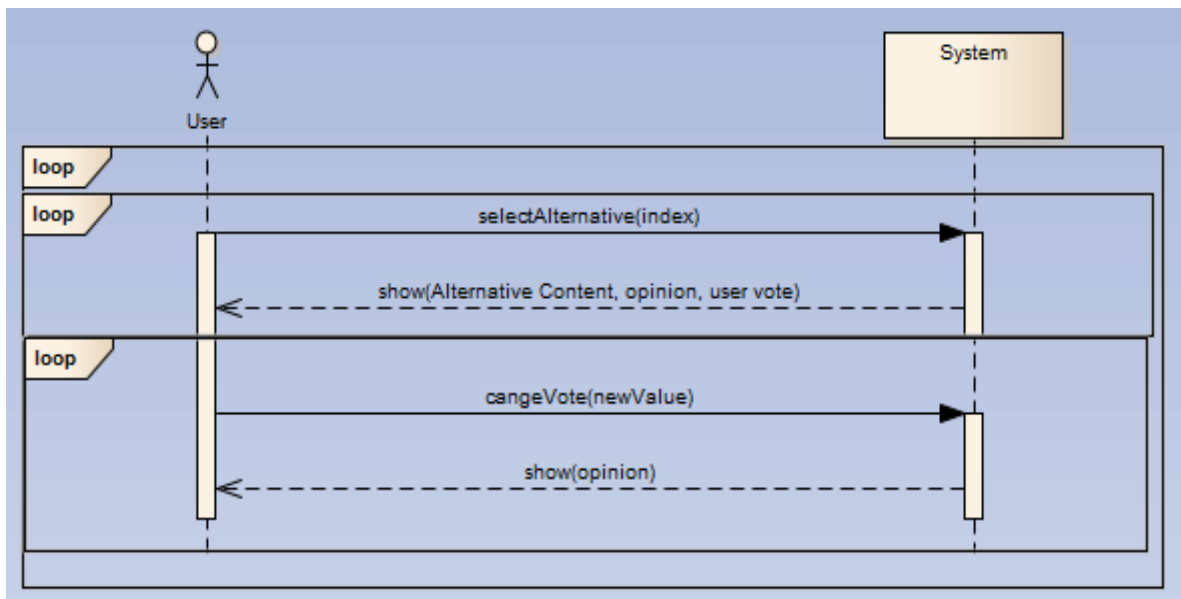


Figure 56: SSD for View Alternatives

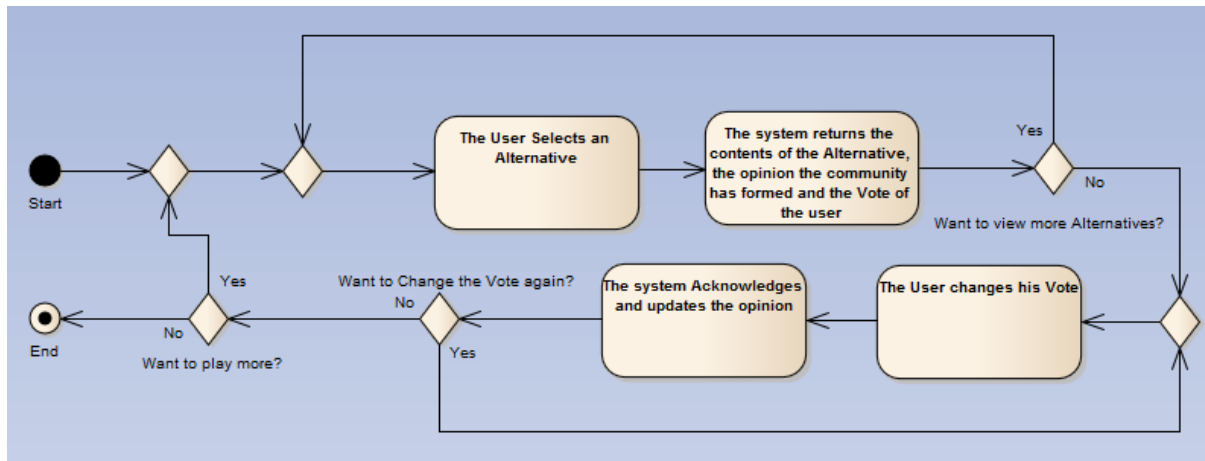


Figure 57: AD for View Alternatives

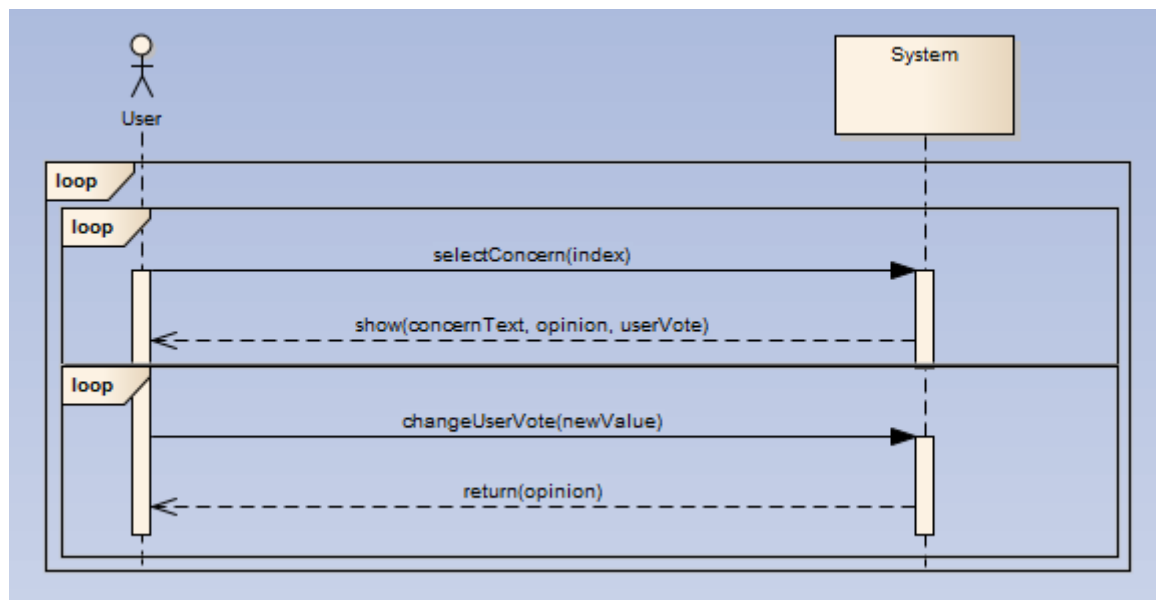


Figure 58: SSD for View Concerns

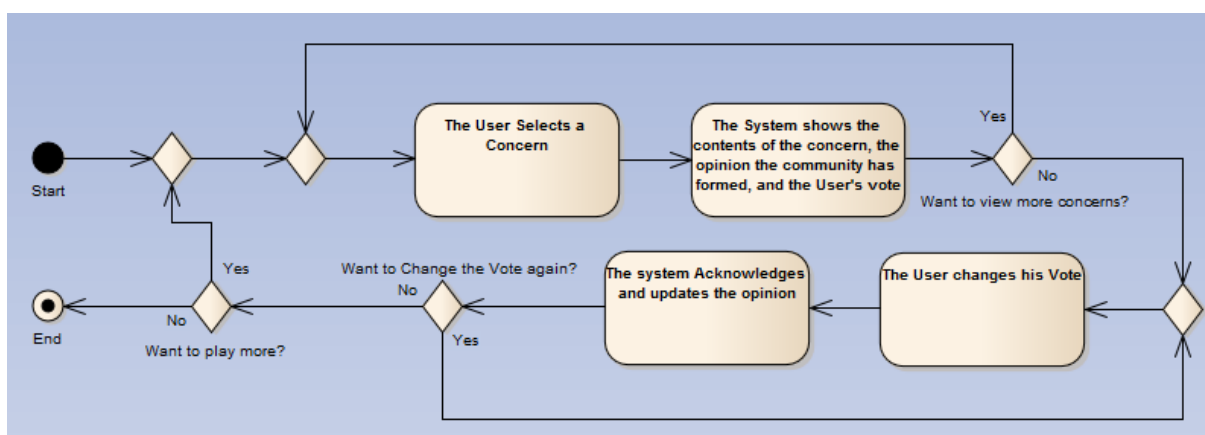


Figure 59: AD for View Concerns

10. Use Case Tables

Information

Actor Type	Description	Quantity	Weight	Subtotal
Simple	Defined API	2	1	2
Average	Interactive or Protocol-driven	3	2	6
Complex	Graphical User Interface	5	3	15
	Total Actor Points			23

Use Case Type	Description	Quantity	Weight	Subtotal
Simple	3 or fewer transactions	6	5	30
Average	4-7 transactions	8	10	80
Complex	7 or more transactions	3	15	45
	Total Use Case Points			155

$$\text{UUCP:} \quad \frac{\text{Total Actor Points} + \text{Total Use Case Points}}{2} = \frac{23 + 155}{2} = 178$$

Technical Factor	Description	Weight	Importance	Subtotal
T1	Must have a distributed solution	2	2	4
T2	Must respond to specific performance objectives	1	5	5
T3	Must meet end-user efficiency desires	1	5	5
T4	Complex internal processing	1	2	2
T5	Code must be reusable	1	4	4
T6	Must be easy to install	0.5	3	1.5
T7	Must be easy to use	0.5	5	2.5
T8	Must be portable	2	3	6
T9	Must be easy to change	1	5	5
T10	Must allow concurrent users	1	5	5
T11	Includes special security features	1	0	0
T12	Must provide direct access to 3rd parties	1	0	0
T13	Requires special user training facilities	1	0	0
	Total TFactor			40

$$\text{SzUC:} \quad \text{UUCP} * (0.6 + (0.01 * \text{Tfactor})) = 178$$

Experience Factor	Description	Weight	Importance	Subtotal
E1	Familiar with FPT software process	1	4	4
E2	Application experience	0.5	4	2
E3	Paradigm experience (OO)	1	4	4
E4	Lead Analyst capability	0.5	4	2
E5	Motivation	0	4	0
E6	Stable Requirements	2	1	2
E7	Part-Time workers	-1	0	0
E8	Difficulty of programming language	-1	1	-1
	Total EFactor			13

UCP: $1.4 - (0.03 * E_{factor}) * Sz_{UC}$ **179.78**
Total man-hours: $20 * UCP$ **3595.6**

Administration

Actor Type	Description	Quantity	Weight	Subtotal
Simple	Defined API	2	1	2
Average	Interactive or Protocol-driven	6	2	12
Complex	Graphical User Interface	6	3	18
	Total Actor Points			32

Use Case Type	Description	Quantity	Weight	Subtotal
Simple	3 or fewer transactions	1	5	5
Average	4-7 transactions	3	10	30
Complex	7 or more transactions	4	15	60
	Total Use Case Points			95

UUCP: $\frac{\text{Total Actor Points} + \text{Total Use Case Points}}{2}$ **127**

Technical Factor	Description	Weight	Importance	Subtotal
T1	Must have a distributed solution	2	2	4
T2	Must respond to specific performance objectives	1	4	4
T3	Must meet end-user efficiency desires	1	5	5
T4	Complex internal processing	1	5	5
T5	Code must be reusable	1	4	4

T6	Must be easy to install	0.5	3	1.5
T7	Must be easy to use	0.5	4	2
T8	Must be portable	2	4	8
T9	Must be easy to change	1	5	5
T10	Must allow concurrent users	1	4	4
T11	Includes special security features	1	5	5
T12	Must provide direct access to 3rd parties	1	0	0
T13	Requires special user training facilities	1	1	1
	Total TFactor			48.5

SzUC: $UUCP * (0.6 + (0.01 * Tfactor))$ **137.795**

Experience Factor	Description	Weight	Importance	Subtotal
E1	Familiar with FPT software process	1	4	4
E2	Application experience	0.5	4	2
E3	Paradigm experience (OO)	1	4	4
E4	Lead Analyst capability	0.5	4	2
E5	Motivation	0	4	0
E6	Stable Requirements	2	1	2
E7	Part-Time workers	-1	0	0
E8	Difficulty of programming language	-1	1	-1
	Total EFactor			13

UCP: $1.4 - (0.03 * Efactor) * SzUC$ **139.172**
20*UCP **2783.45**
Total man-hours: **9**

Community

Actor Type	Description	Quantity	Weight	Subtotal
Simple	Defined API	3	1	3
Average	Interactive or Protocol-driven	0	2	0
Complex	Graphical User Interface	5	3	15
	Total Actor Points			18

Use Case Type	Description	Quantity	Weight	Subtotal
Simple	3 or fewer transactions	6	5	30
Average	4-7 transactions	14	10	140

Complex	7 or more transaction	3	15	45
	Total Use Case Points			215

**Total Actor Points+
Total Use Case Points 233**

UUCP:

Technical Factor	Description	Weight	Importance	Subtotal
T1	Must have a distributed solution	2	2	4
T2	Must respond to specific performance objectives	1	5	5
T3	Must meet end-user efficiency desires	1	5	5
T4	Complex internal processing	1	2	2
T5	Code must be reusable	1	4	4
T6	Must be easy to install	0.5	3	1.5
T7	Must be easy to use	0.5	5	2.5
T8	Must be portable	2	3	6
T9	Must be easy to change	1	5	5
T10	Must allow concurrent users	1	5	5
T11	Includes special security features	1	5	5
T12	Must provide direct access to 3rd parties	1	0	0
T13	Requires special user training facilities	1	0	0
	Total TFactor			45

UUCP*(0.6+(0.01*Tfactor)) 244.65

SzUC:

Experience Factor	Description	Weight	Importance	Subtotal
E1	Familiar with FPT software process	1	4	4
E2	Application experience	0.5	4	2
E3	Paradigm experience (OO)	1	4	4
E4	Lead Analyst capability	0.5	4	2
E5	Motivation	0	4	0
E6	Stable Requirements	2	1	2
E7	Part-Time workers	-1	0	0
E8	Difficulty of programming language	-1	1	-1
	Total EFactor			13

**1.4-
(0.03*Efactor)*SzUC 247.096**

UCP:

5

Total man-hours:

20*UCP 4941.93

Rights Trading

Actor Type	Description	Quantity	Weight	Subtotal
Simple	Defined API	2	1	2
Average	Interactive or Protocol-driven	0	2	0
Complex	Graphical User Interface	5	3	15
	Total Actor Points			17

Use Case Type	Description	Quantity	Weight	Subtotal
Simple	3 or fewer transactions	3	5	15
Average	4-7 transactions	3	10	30
Complex	7 or more transaction	0	15	0
	Total Use Case Points			45

UUCP: $\frac{\text{Total Actor Points} + \text{Total Use Case Points}}{2}$

Technical Factor	Description	Weight	Importance	Subtotal
T1	Must have a distributed solution	2	2	4
T2	Must respond to specific performance objectives	1	3	3
T3	Must meet end-user efficiency desires	1	5	5
T4	Complex internal processing	1	0	0
T5	Code must be reusable	1	4	4
T6	Must be easy to install	0.5	3	1.5
T7	Must be easy to use	0.5	5	2.5
T8	Must be portable	2	3	6
T9	Must be easy to change	1	4	4
T10	Must allow concurrent users	1	5	5
T11	Includes special security features	1	5	5
T12	Must provide direct access to 3rd parties	1	0	0
T13	Requires special user training facilities	1	0	0
	Total TFactor			40

SzUC: $\text{UUCP} * (0.6 + (0.01 * \text{Tfactor}))$ 62

Experience Factor	Description	Weight	Importance	Subtotal
E1	Familiar with FPT software process	1	4	4
E2	Application experience	0.5	4	2

E3	Paradigm experience (OO)	1	4	4
E4	Lead Analyst capability	0.5	4	2
E5	Motivation	0	4	0
E6	Stable Requirements	2	3	6
E7	Part-Time workers	-1	0	0
E8	Difficulty of programming language	-1	1	-1
	Total EFactor			17

UCP: $1.4 - (0.03 * E_{factor}) * S_{zUC}$ **55.18**
Total man-hours: $20 * UCP$ **1103.6**

Contract

Actor Type	Description	Quantity	Weight	Subtotal
Simple	Defined API	1	1	1
Average	Interactive or Protocol-driven	1	2	2
Complex	Graphical User Interface	6	3	18
	Total Actor Points			21

Use Case Type	Description	Quantity	Weight	Subtotal
Simple	3 or fewer transactions	2	5	10
Average	4-7 transactions	3	10	30
Complex	7 or more transaction	3	15	45
	Total Use Case Points			85

UUCP: $\frac{\text{Total Actor Points} + \text{Total Use Case Points}}{106}$

Technical Factor	Description	Weight	Importance	Subtotal
T1	Must have a distributed solution	2	2	4
T2	Must respond to specific performance objectives	1	4	4
T3	Must meet end-user efficiency desires	1	5	5
T4	Complex internal processing	1	3	3
T5	Code must be reusable	1	4	4
T6	Must be easy to install	0.5	3	1.5
T7	Must be easy to use	0.5	5	2.5
T8	Must be portable	2	3	6

T9	Must be easy to change	1	5	5
T10	Must allow concurrent users	1	5	5
T11	Includes special security features	1	5	5
T12	Must provide direct access to 3rd parties	1	0	0
T13	Requires special user training facilities	1	0	0
	Total TFactor			45

SzUC: $UUCP * (0.6 + (0.01 * Tfactor))$ **111.3**

Experience Factor	Description	Weight	Importance	Subtotal
E1	Familiar with FPT software process	1	4	4
E2	Application experience	0.5	4	2
E3	Paradigm experience (OO)	1	4	4
E4	Lead Analyst capability	0.5	4	2
E5	Motivation	0	4	0
E6	Stable Requirements	2	1	2
E7	Part-Time workers	-1	0	0
E8	Difficulty of programming language	-1	1	-1
	Total EFactor			13

UCP: $1.4 - (0.03 * Efactor) * SzUC$ **112.413**
Total man-hours: $20 * UCP$ **2248.26**

Product Information

Actor Type	Description	Quantity	Weight	Subtotal
Simple	Defined API	1	1	1
Average	Interactive or Protocol-driven	3	2	6
Complex	Graphical User Interface	5	3	15
	Total Actor Points			22

Use Case Type	Description	Quantity	Weight	Subtotal
Simple	3 or fewer transactions	1	5	5
Average	4-7 transactions	4	10	40
Complex	7 or more transaction	2	15	30
	Total Use Case Points			75

UUCP: **Total Actor Points+
Total Use Case Points** 97

Technical Factor	Description	Weight	Importance	Subtotal
T1	Must have a distributed solution	2	2	4
T2	Must respond to specific performance objectives	1	5	5
T3	Must meet end-user efficiency desires	1	5	5
T4	Complex internal processing	1	5	5
T5	Code must be reusable	1	3	3
T6	Must be easy to install	0.5	4	2
T7	Must be easy to use	0.5	5	2.5
T8	Must be portable	2	4	8
T9	Must be easy to change	1	4	4
T10	Must allow concurrent users	1	5	5
T11	Includes special security features	1	0	0
T12	Must provide direct access to 3rd parties	1	0	0
T13	Requires special user training facilities	1	0	0
	Total TFactor			43.5

SzUC: **UUCP*(0.6+(0.01*Tfactor))** 100.395

Experience Factor	Description	Weight	Importance	Subtotal
E1	Familiar with FPT software process	1	4	4
E2	Application experience	0.5	4	2
E3	Paradigm experience (OO)	1	4	4
E4	Lead Analyst capability	0.5	4	2
E5	Motivation	0	4	0
E6	Stable Requirements	2	1	2
E7	Part-Time workers	-1	0	0
E8	Difficulty of programming language	-1	1	-1
	Total EFactor			13

UCP: **1.4-(0.03*Efactor)*SzUC** 101.398
20*UCP 95
Total man-hours: 2027.979

Measurement & Monitoring

Actor Type	Description	Quantity	Weight	Subtotal
Simple	Defined API	2	1	2
Average	Interactive or Protocol-driven	1	2	2
Complex	Graphical User Interface	6	3	18
	Total Actor Points			22

Use Case Type	Description	Quantity	Weight	Subtotal
Simple	3 or fewer transactions	1	5	5
Average	4-7 transactions	1	10	10
Complex	7 or more transaction	1	15	15
	Total Use Case Points			30

UUCP: **Total Actor Points+** **52**
Total Use Case Points

Technical Factor	Description	Weight	Importance	Subtotal
T1	Must have a distributed solution	2	2	4
T2	Must respond to specific performance objectives	1	3	3
T3	Must meet end-user efficiency desires	1	4	4
T4	Complex internal processing	1	4	4
T5	Code must be reusable	1	3	3
T6	Must be easy to install	0.5	3	1.5
T7	Must be easy to use	0.5	4	2
T8	Must be portable	2	3	6
T9	Must be easy to change	1	3	3
T10	Must allow concurrent users	1	3	3
T11	Includes special security features	1	5	5
T12	Must provide direct access to 3rd parties	1	0	0
T13	Requires special user training facilities	1	0	0
	Total TFactor			38.5

SzUC: **UUCP*(0.6+(0.01*Tfactor))** **51.22**

Experience Factor	Description	Weight	Importance	Subtotal
E1	Familiar with FPT software process	1	4	4
E2	Application experience	0.5	4	2
E3	Paradigm experience (OO)	1	4	4
E4	Lead Analyst capability	0.5	4	2

E5	Motivation	0	4	0
E6	Stable Requirements	2	3	6
E7	Part-Time workers	-1	0	0
E8	Difficulty of programming language	-1	1	-1
	Total EFactor			17

UCP:	1.4-	45.5858
	(0.03*Efactor)*SzUC	
Total man-hours:	20*UCP	911.716

11. Example User Interfaces

View Area Information

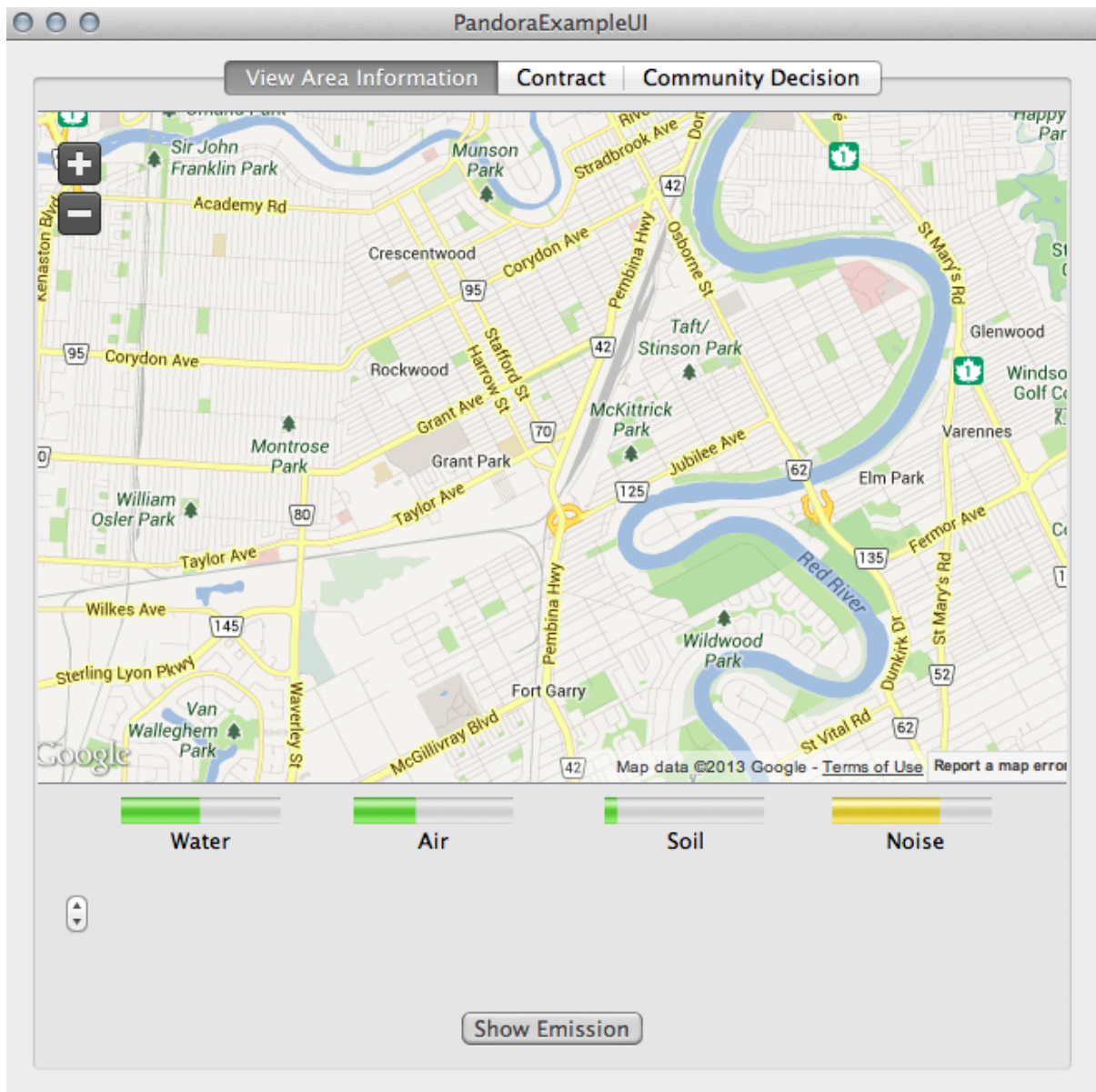


Figure 61: The second information layer

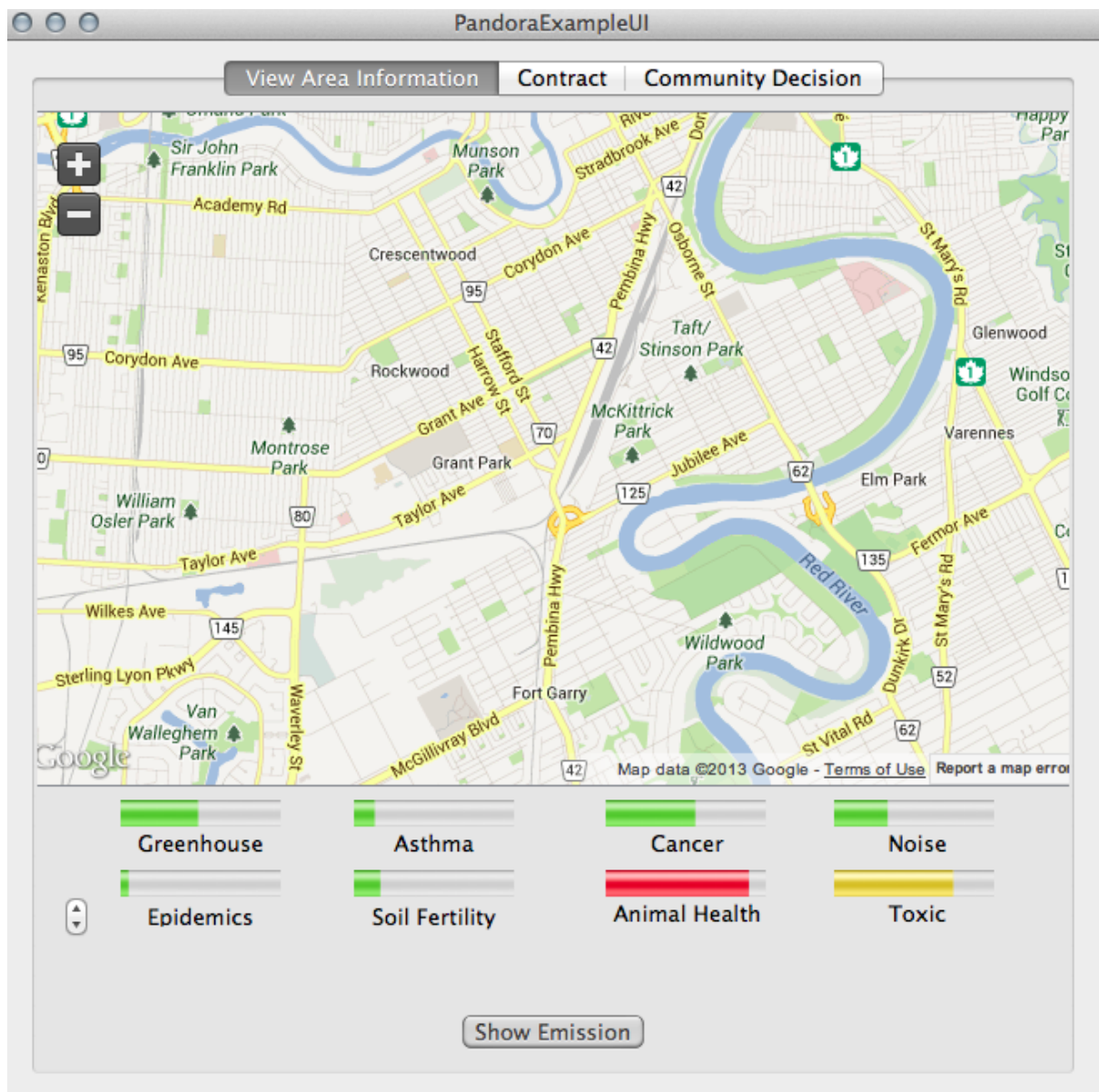


Figure 62: The third information layer

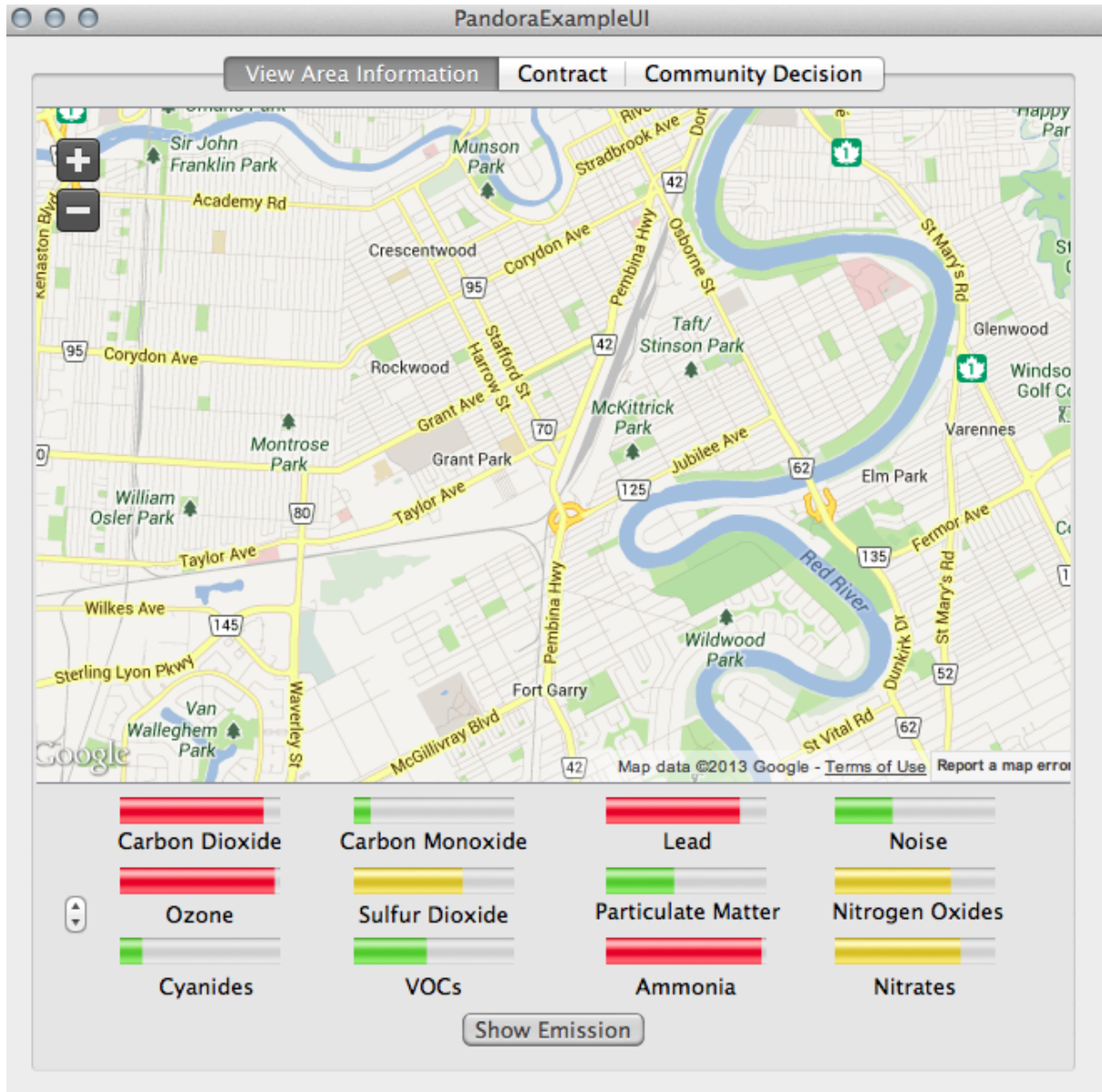


Figure 63: The fourth layer

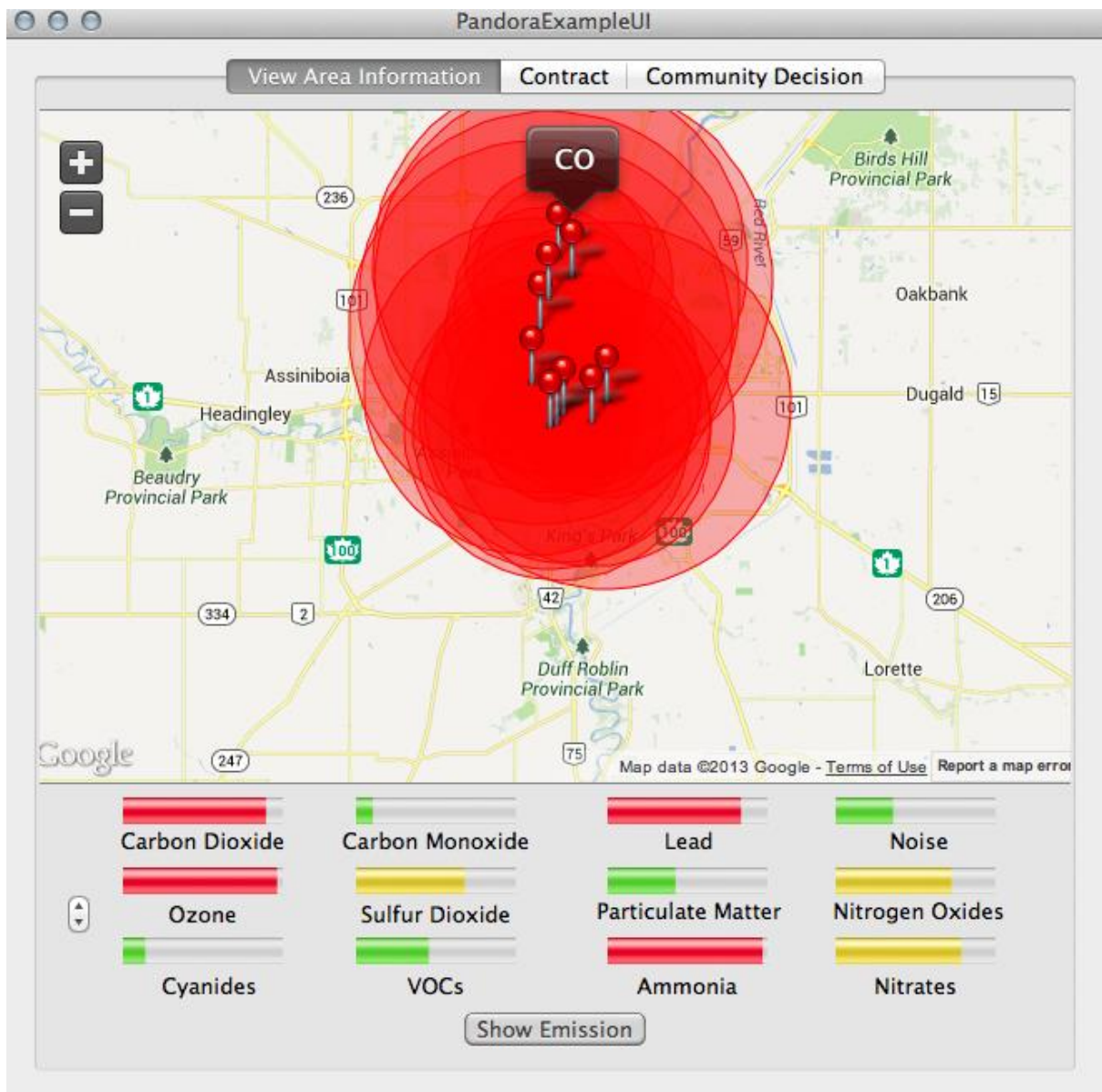


Figure 64: Identifying sources

Make an Offer

View Area Information

Contract

Community Decision

Offers

Single Payment
Periodic Payment
Reduce Pollutant Emission
Periodic System Checkups for safety
Get Insurance
Contributions to Area Infrastructure
Free Periodic Health Checkups
People Responsible Would Live in the Area

Demands

Single Payment
Periodic Payment
Contribute to Greener/Safer Technology
Allow Advertisements to be Placed in the Area
Upgrade Area Infrastructure

Make Offer

Templates

Polluting Factory
Non-Polluting Risky
Polluting No Law Protection
Risky No Law Protection

Pollution on the Table

Ammonia - risk - 500%
Noise - 3%
Nitrates - 0.1%
Ozone - chance - 1240%

Current Offer

----- Offers -----
Get Insurance
Periodic System Checkups for safety
----- Demands -----
Contribute to Greener/Safer Technology
Periodic Payment

Figure 65: Making an offer

View Area Information
Contract
Community Decision

Offers

Single Payment
Periodic Payment
Reduce Pollutant Emission
Periodic System Checkups for safety
Get Insurance
Contributions to Area Infrastructure
Free Periodic Health Checkups
People Responsible Would Live in the Area

Demands

Single Payment
Periodic Payment
Contribute to Greener/Safer Technology
Allow Advertisements to be Placed in the Area
Upgrade Area Infrastructure

Templates

Polluting Factory
Non-Polluting Risky
Polluting No Law Protection
Risky No Law Protection

Pollution on the Table

Ammonia – risk – 500%
Noise – 3%
Nitrates – 0.1%
Ozone – chance – 1240%

Current Offer

----- Offers -----
Periodic Payment
Reduce Pollutant Emission
Free Periodic Health Checkups
Contributions to Area Infrastructure
----- Demands -----
Upgrade Area Infrastructure
Contribute to Greener/Safer Technology
Allow Advertisements to be Placed in the Area

Make Offer

Figure 66: An offer template

Vote/Make a Decision

PandoraExampleUI

View Area Information Contract **Community Decision**

Polls	Status
Tesco	Under D...
GE Factory	Final Vote

Offer

----- Offers -----

Periodic Payment

Reduce Pollutant Emission

Free Periodic Health Checkups

Contributions to Area Infrastructure

----- Demands -----

Your Vote: ★★★★★

Community Consensus: 

Concerns
We are protected by law

I think the other template would be a better alternative

Alternatives
Other Template

Alternative Offer

Periodic Payment

Reduce Pollutant Emission

Free Periodic Health Checkups

Contributions to Area Infrastructure

----- Demands -----

Upgrade Area Infrastructure

Your Vote: ★★★★★

Community Consensus: 

Add new Concern

Your Vote: ★★★★★

Community Consensus: 

Add new Alternative

Figure 67: Vote/Make a Decision

12. Software Data

Information Module

Table 6: Actors of the Information Module

Type	Actors
API	Social Network Sites
PD	Pollution Databases Environmental Databases Spatial Databases
GUI	User

Table 7: Use Cases of the Information Module

Type	Use Cases
Easy	Registration Login View Favorites Add to Favorites View Pollutant Information Identify Causes of Pollution
Average	View User Profile User Search Imission Data Search Share Contact Support Report a Problem Compare Areas View Emission
Complex	Emission Data Search View Area Information Search Places

Table 8: Estimated Servers needed

Type of Server	Count	Min	Source
Web Server	80	5	Estimate (What the web's...)
Database Server	20	1	Estimate (What the web's...)
Storage Server	0	0	Estimate (What the web's...)
Other	0	0	Estimate (What the web's...)
Total	100	-	-

Table 9: Estimated Data Transfer Needed

Primary Transfer Activity	Transfer per user	Data per transfer (KBs)	Source
Page Downloads	4	4096	Estimate
Map image & data downloads	15	2048	Estimate
% of Monthly System Users Visiting this Module	100%	-	Estimate

Table 10: Additional Data

Missing Data	INSPIRE is not ready yet, but area data might not be available in a format we need it.
Third Party Software	-
Special Support	-

Administration Module

Table 11: Actors of the Administration Module

Type	Actors
API	Ügyfélkapu
PD	Pollution Databases Environmental Databases Spatial/Geological Databases Meteorological Databases
GUI	User Administrator

Table 12: Use Cases of the Administration Module

Type	Use Cases
Easy	Connect Account to Ügyfélkapu
Average	Register New Emission Edit Profile View Owned Rights
Complex	Edit Users' Rights Calculate Emission Estimate Effect Estimate Damage

Table 13: Servers Needed

Type of Server	Count	Min	Source
Web Server	5	1	Estimate (What the web's...)
Database Server	10	1	Estimate (What the web's...)
Storage Server	15	1	Estimate (What the web's...)
Other	5	1	Estimate (What the web's...)
Total	35	-	-

Table 14: Data Transfer needed

Primary Transfer Activity	Transfer per user	Data per transfer (KBs)	Source
Page Downloads	5	4096	Estimate
% of Monthly System Users Visiting this Module	10%	-	Estimate

Table 15: Additional Data

Missing Data	Data for effect estimation Rights need to be distributed
Third Party Software	Air & Noise pollution modeling Water pollution modeling
Special Support	Administrators

Community Module

Table 16: Actors of the Community Module

Type	Actors
API	Online Payment System Social Network Site
PD	-
GUI	User

Table 17: Use Cases of the Community Module

Type	Use Cases
Easy	Invitation Application Ask For Donation Read Message Find Friends Add New Friend
Average	Search For Nearby Communities Form a Community Create a Poll Vote/Make a Decision Donate Connect to S.N. Account View Forum Thread Open New Thread Send Invitation to the System View Concerns View Alternatives Add new Concern Evaluate Alternatives Set Preferences
Complex	Send a message Reply to Thread Add New Alternative

Table 18: Servers Needed

Type of Server	Count	Min	Source
Web Server	40	2	Estimate (What the web's...)
Database Server	10	1	Estimate (What the web's...)
Storage Server	0	0	Estimate (What the web's...)
Other	25	2	Estimate (What the web's...)
Total	75	-	-

Table 19: Data Transfer Needed

Primary Transfer Activity	Transfer per user	Data per transfer (KBs)	Source
Page Downloads	6	4096	Estimate

% of Monthly System Users Visiting this Module	100%	-	Estimate
---	-------------	----------	-----------------

Table 20: Additional Data

Missing Data	-
Third Party Software	-
Special Support	-

Rights Trading Module

Table 21: Actors of the Rights Trading Module

Type	Actors
API	Emission Exchange System
PD	-
GUI	User

Table 22: Use Cases of the Rights Trading Module

Type	Use Cases
Easy	Buy Rights Sell Right Add New Right to Monitor
Average	View Missing Right Search for Tradable Rights Monitor Rights
Complex	-

Table 23: Servers Needed

Type of Server	Count	Min	Source
Web Server	10	1	Estimate (What the web's...)
Database Server	5	1	Estimate (What the web's...)
Storage Server	0	0	Estimate (What the web's...)
Other	0	0	Estimate (What the web's...)
Total	15	-	-

Table 24: Data Transfer Requirements

Primary Transfer Activity	Transfer per user	Data per transfer (KBs)	Source
Page Downloads	2	4096	Estimate
Real-Time asset price data queries	10,000	10	Estimate

% of Monthly System Users Visiting this Module	10%	-	Estimate
--	-----	---	----------

Table 25: Additional Data

Missing Data	-
Third Party Software	-
Special Support	-

Contract Module

Table 26: Actors of the Contract Module

Type	Actors
API	-
PD	-
GUI	User Legal Support Member

Table 27: Use Cases of the Contract Module

Type	Use Cases
Easy	Conclude Contract View Violated Rights
Average	View Offer View Contract Get Legal Help
Complex	Make Offer Review Contract Make New Template

Table 28: Servers Needed

Type of Server	Count	Min	Source
Web Server	5	1	Estimate (What the web's...)
Database Server	1	1	Estimate (What the web's...)
Storage Server	2	1	Estimate (What the web's...)
Other	2	1	Estimate (What the web's...)
Total	10	-	-

Table 29: Data Transfer Needed

Primary Transfer Activity	Transfer per user	Data per transfer (KBs)	Source
Page Downloads	5	4096	Estimate
% of Monthly System Users Visiting this Module	10%	-	Estimate

Table 30: Additional Data

Missing Data	-
Third Party Software	Document Automation
Special Support	Legal Support Team

Product Information Module

Table 31: Actors of the Product Information Module

Type	Actors
API	-
PD	Product Databases
GUI	User

Table 32: Use Cases of the Product Information Module

Type	Use Cases
Easy	Identify Product
Average	Search Products Search Green Technologies View Product Information View Green Technology
Complex	Add New Product Add New Green Technology

Table 33: Servers Needed

Type of Server	Count	Min	Source
Web Server	30	2	Estimate (What the web's...)
Database Server	15	2	Estimate (What the web's...)
Storage Server	10	5	Estimate (What the web's...)
Other	20	1	Estimate (What the web's...)
Total	75	-	-

Table 34: Data Transfer Needed

Primary Transfer Activity	Transfer per user	Data per transfer (KBs)	Source
Page Downloads	3	4096	Estimate
Product Recognition Information	1	1024	Estimate
% of Monthly System Users Visiting this Module	50%	-	Estimate

Table 35: Additional Data

Missing Data	Environmental data about products is not always available, but we can make producers provide the data on their own products.
Third Party Software	Visual Object Recognition service
Special Support	-

Measurement & Monitoring Module

Table 36: Actors of the Module

Type	Actors
API	Mote
PD	Meteorological Databases
GUI	User Administrator

Table 37: Use Cases of the Module

Type	Use Cases
Easy	Send Measurement
Average	Request Monitoring
Complex	Monitor Area

Table 38: Servers Needed

Type of Server	Count	Min	Source
Web Server	1	1	Estimate (What the web's...)
Database Server	1	1	Estimate (What the web's...)
Storage Server	5	2	Estimate (What the web's...)
Other	5	1	Estimate (What the web's...)
Total	12	-	-

Table 39: Data Transfer Needed

Primary Transfer Activity	Transfer per user	Data per transfer (KBs)	Source
Mote/Measurement Unit data Send	1440	10	Estimate
% of Monthly System Users Visiting this Module	10%	-	Estimate

Table 40: Additional Data

Missing Data	-
Third Party Software	-
Special Support	Motes are needed

13. Cost Estimation Data

Table 42: General Data

Title	Value	Source
Number of work days in a month:	22	-
Number of work hours a day:	8	-
Number of months a year:	12	-
Number of days a month:	30.5	-
Number of hours a day:	24	-
Years:	10	Choice
Discount Rate:	5.5%	Choice
Uncertainty of Estimation:	50%	Choice
Development Estimation Uncertainty:	100%	Choice
HUF to \$:	225	-
Euro to \$:	1.32	-
Conversion between lbs and metric tons	0.00045359	-
Employer's cost as a percentage of salary:	128.5%	(Bérkalkulátor)
Hungarian wages to EU	1.5625	(EuroStat)
Minimum Monthly Users	20000	Web Statistics

Table 43: Assumptions of secondary efforts

Title	Value (percentage of the development effort)	Source
Reporting & Management:	40%	Estimate
Software Design:	25%	Estimate
Number of SDKs:	4	-

Table 44: Salary Data

Title	Value	Source
Average Monthly Salaries for Software Engineers:	\$2,000.00	(Fizetesek- Software)
Average Monthly Cost of Employer:	\$2,570.00	Calculation
Average Hourly Cost of Employer:	\$14.60	Calculation

Table 45: Estimates for Maintenance Effort

Title	Value (percentage of full development effort)	Source
Maintenance Effort (1st year):	15%	(Jones 2006)
Maintenance Effort (2nd year):	16%	(Jones 2006)
Maintenance Effort (3rd year):	17%	(Jones 2006)
Maintenance Effort (4th year):	18%	(Jones 2006)
Maintenance Effort (5th year):	20%	(Jones 2006)
Maintenance Effort (6th year):	21%	Estimate (Jones 2006)
Maintenance Effort (7th year):	22%	Estimate (Jones 2006)
Maintenance Effort (8th year):	23%	Estimate (Jones 2006)
Maintenance Effort (9th year):	24%	Estimate (Jones 2006)
Maintenance Effort (10th year):	25%	Estimate (Jones 2006)

Table 46: Estimates for the Total Development & Maintenance Costs

Module	Design & Development Man-hours	Dev. Costs	Reporting Costs	Maintenance Costs	Total
Information	8,989	\$131,260	\$52,504	\$193,874	\$377,638
Administration	6,959	\$101,612	\$40,645	\$150,084	\$292,341
Community	12,355	\$180,409	\$72,163	\$266,468	\$519,040
Rights Trading	2,759	\$40,288	\$16,115	\$59,506	\$115,909
Contract	5,621	\$82,074	\$32,830	\$121,226	\$236,130
Product Information	5,070	\$74,033	\$29,613	\$109,348	\$212,994
Measurement & Monitoring	2,279	\$33,283	\$13,313	\$49,160	\$95,756
Total	44,031	\$642,958	\$257,183	\$949,666	\$1,849,807

Table 47: Data about Server Cost Estimation

Title	Value	Source
Number of Servers in a rack	40	Product Data
TCO of a rack	\$65,000	(InfraStructureX)
Base users	1,000,000	Choice
Operating expenses (%of TCO)	50%	(Determining... 2005)

Table 48: The Estimated Number of Servers Needed

Number of Users	20,000	100,000	200,000	1,000,000	5,000,000	10,000,000
Information	6	10	20	100	500	1,000
Administration	4	5	7	35	175	350
Community	5	8	15	75	375	750
Rights Trading	2	2	3	15	75	150
Contract	4	4	4	10	50	100
Product Information	10	12	18	75	375	750
M& M	5	5	5	12	60	120
Total	36	45	72	322	1,610	3,220

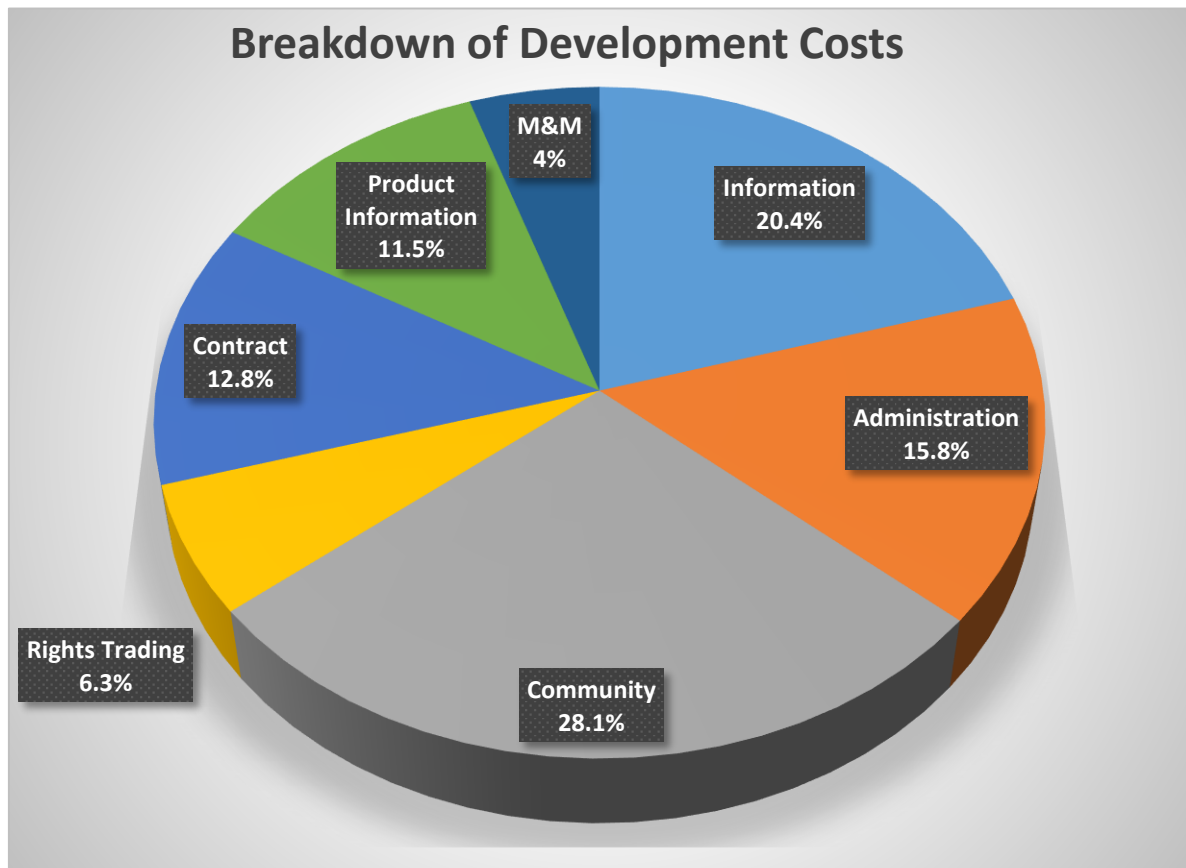


Table 49: Total 10-Year Server Costs

Number of Users	20,000	100,000	200,000	1,000,000	5,000,000	10,000,000
Information	\$12,824	\$21,374	\$42,748	\$213,740	\$1,068,699	\$2,137,398
Administration	\$8,550	\$9,618	\$14,962	\$74,809	\$374,045	\$748,089
Community	\$10,687	\$16,030	\$32,061	\$160,305	\$801,524	\$1,603,049
Rights Trading	\$4,275	\$4,275	\$6,412	\$32,061	\$160,305	\$320,610
Contract	\$8,550	\$8,550	\$8,550	\$21,374	\$106,870	\$213,740
Product Information	\$21,374	\$25,649	\$38,473	\$160,305	\$801,524	\$1,603,049
Measurement & Monitoring	\$10,687	\$10,687	\$10,687	\$25,649	\$128,244	\$256,488
Total	\$76,946	\$96,183	\$153,893	\$688,242	\$3,441,211	\$6,882,422

Table 50: Carbon Emission Data

Title	Value	Source
Cost of GHG emission (\$/metric ton)	\$30	Largest Price in (FactSheet)
2006 GHG Emission of the Internet (lbs/GB)	33.9	(Taylor 2008)
Power use decrease (2000 - 2006)	0.1	(Taylor 2008)
Assumed power use decrease (2006 – 2013)	0.2	Estimate (Taylor 2008)
Assumed power use decrease (2013 – 2022)	0.1	Estimate (Taylor 2008)

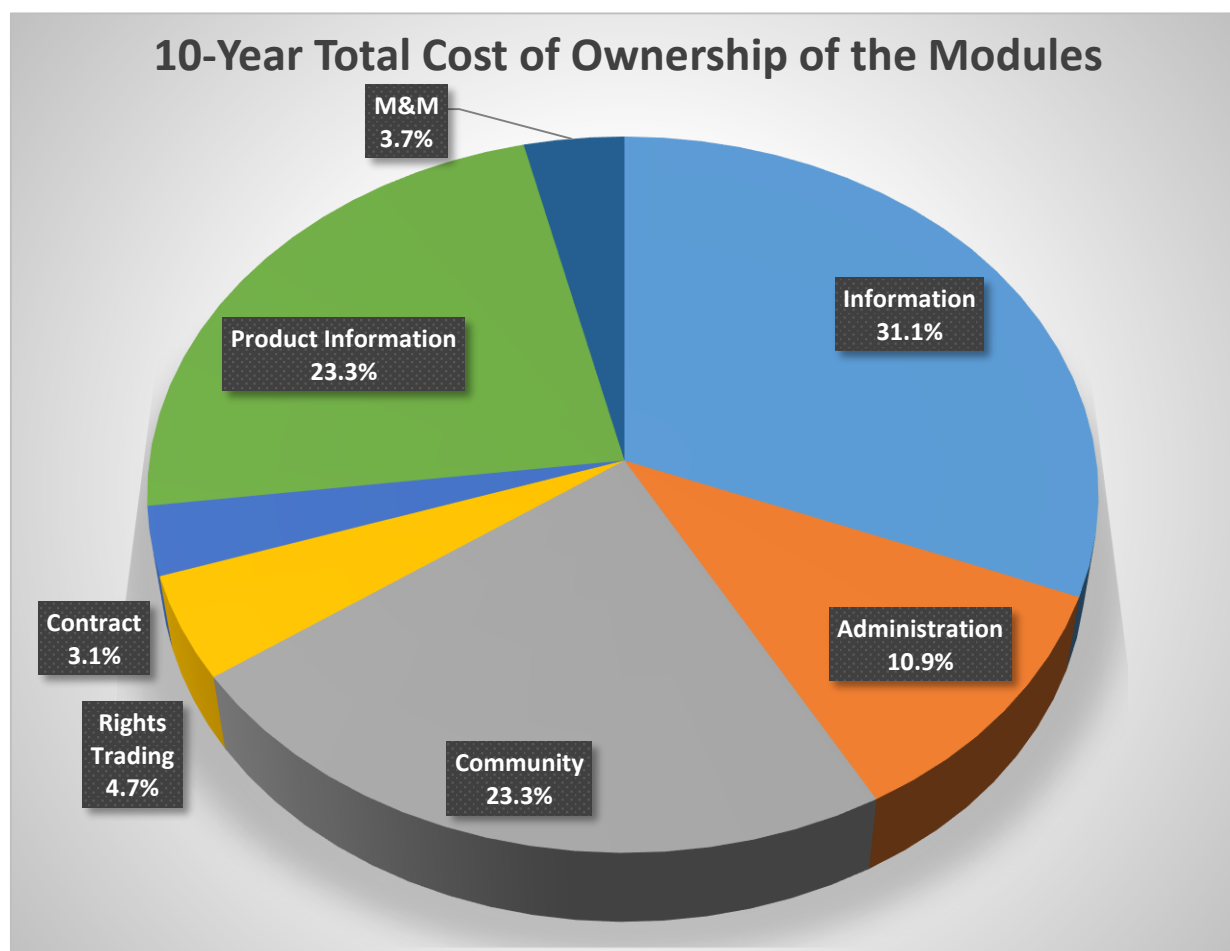


Table 51: Estimated Carbon Emission per Data Transfer for the Next Decade

Year	Value (lbs/GB)	Source
2013	6.78	Estimate (Taylor 2008)
2014	5.38	Estimate (Taylor 2008)
2015	4.27	Estimate (Taylor 2008)
2016	3.39	Estimate (Taylor 2008)
2017	2.69	Estimate (Taylor 2008)
2018	2.14	Estimate (Taylor 2008)
2019	1.70	Estimate (Taylor 2008)
2020	1.35	Estimate (Taylor 2008)
2021	1.07	Estimate (Taylor 2008)
2022	0.85	Estimate (Taylor 2008)

Table 52: Estimated Yearly Data Transfer of the Modules (in Gigabytes)

Number of Users	20,000	100,000	200,000	1,000,000	5,000,000	10,000,000
Information	16,957	84,787	169,574	847,872	4,239,360	8,478,720
Administration	737	3,686	7,373	36,864	184,320	368,640
Community	8,847	44,237	88,474	442,368	2,211,840	4,423,680
Rights Trading	3,895	19,475	38,949	194,746	973,728	1,947,456
Contract	737	3,686	7,373	36,864	184,320	368,640
Product Information	2,396	11,981	23,962	119,808	599,040	1,198,080
M&M	124	124	124	5,846	5,846	5,846
Total	33,695	167,977	335,829	1,684,368	8,398,454	16,791,062

Table 53: Total 10-Year Emission Costs

Number of Users	20,000	100,000	200,000	1,000,000	5,000,000	10,000,000
Information	\$5,650	\$28,252	\$56,504	\$282,521	\$1,412,606	\$2,825,213
Administration	\$246	\$1,228	\$2,457	\$12,284	\$61,418	\$122,835
Community	\$2,948	\$14,740	\$29,480	\$147,402	\$737,012	\$1,474,024
Rights Trading	\$1,298	\$6,489	\$12,978	\$64,892	\$324,458	\$648,916
Contract	\$246	\$1,228	\$2,457	\$12,284	\$61,418	\$122,835
Product Information	\$798	\$3,992	\$7,984	\$39,921	\$199,607	\$399,215
M & M	\$41	\$41	\$41	\$1,948	\$1,948	\$1,948
Total	\$11,228	\$55,972	\$111,902	\$561,252	\$2,798,467	\$5,594,987

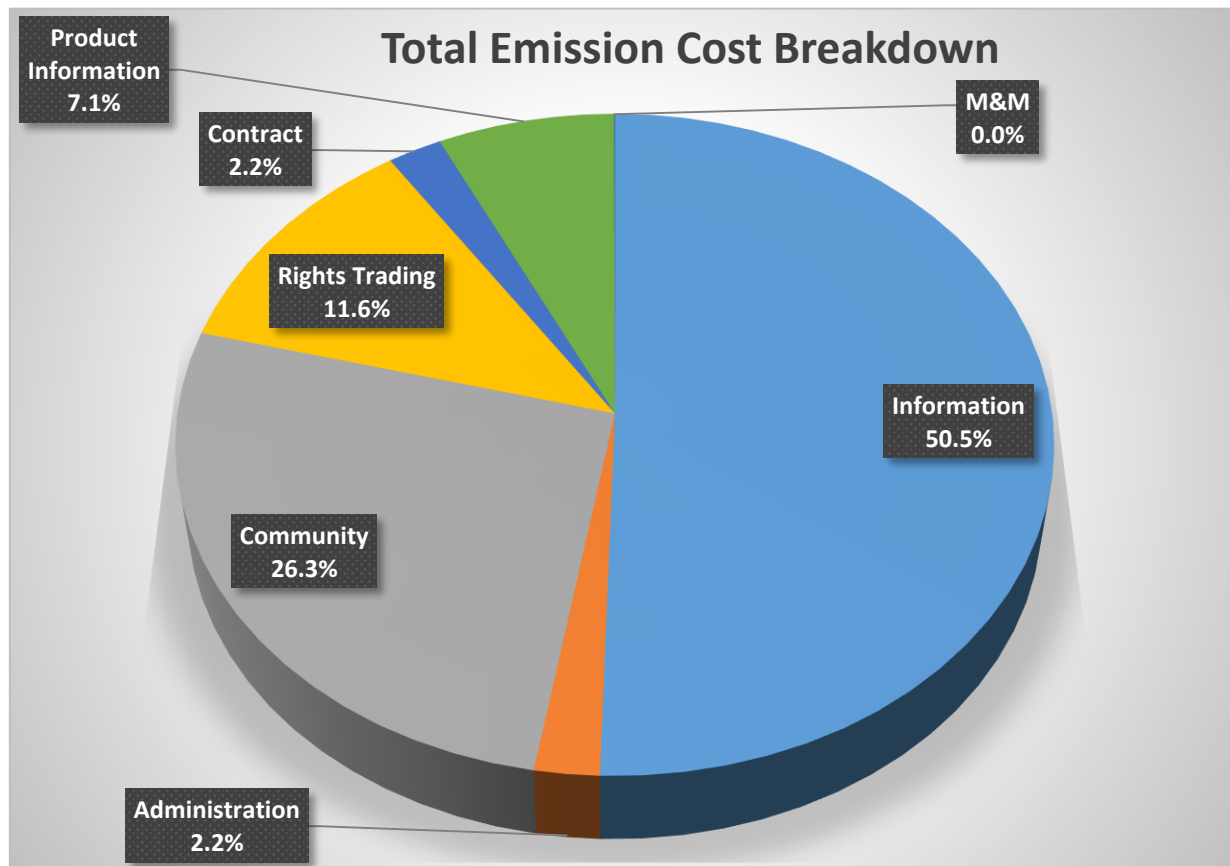


Table 54: Area and Rights Data Collection Data

Title	Value	Source
Estimated effort to collect pollution effect data (man-months)	15	Estimate
Estimated effort to collect training data (man-months)	4	Estimate
Estimated effort of creating ML algorithm (man-months)	6	Estimate
Monthly salary of an expert	\$1,111	(Fizetesek-Expert)
Monthly cost of employer for an expert	\$1,428	Calculation
Monthly salary of an engineer	\$2,222	(Fizetesek-Software)
Monthly cost of employer for an engineer	\$2,856	Calculation

Table 55: Estimated Effort for Automatic EIA Creation

Title	Value	Source
Estimated effort to make areas on an area of 100,000 km ² (man-months)	8	Estimate
Administrator monthly salary	\$667	(Fizetesek - Admin)
Monthly cost of employer	\$857	Calculation
Estimated Effort to collect rights data on 100,000 km ² (man-months)	12	Estimate

Table 57: Estimated Data for Legal Support Cost Calculation

Title	Value	Source
Percentage of support requests	1.0%	Estimate
Number of requests a lawyer can satisfy in a year	480	Estimate
Average Salary of a lawyer	\$18,667	(Fizetesek-Lawyer)
Cost of employer	\$35,980	Calculation
Minimum Lawyers needed for EU	10	Estimate

Table 59: Data Needed to Estimate Administration Costs

Title	Value	Source
Number of yearly transactions for an Admin:	25,344	Estimate
Monthly Salary of an Administrator:	\$667	(Fizetesek-Admin)
Cost of employer with an admin:	\$856	Calculation
Yearly cost of an administrator:	\$15,420	Calculation

Table 61: Estimates for Legislation Costs

Title	EU	Source	HUN	Source2
Yearly expenditures of the parliament:	\$1,301,515,152	(EU Budget)	\$170,673,778	(Hungarian Budget)
Sitting days in a year	134	(EU Parliament Schedule)	94	(Hungarian Parliament Schedule)
Sitting days in a year (% of days a year):	36.58%	Calculation	25.75%	Calculation
Average Number of Laws/Texts Passed:	550	(Text Search)	130	(Parliament Website)
Estimated Cost of a Law:	\$1,298,274	Calculation	\$507,165	Calculation

Table 62: Data Needed to Estimate Mote Equipment Costs

Title	Value	Source
Number of motes for 100,000 km ²	500	Estimate
Cost of a mote	\$95	(Mote Product Data)

Table 64: Ad Revenue Estimation Data

Title	Value	Source
Monthly Users (Reference)	1,000,000	Choice
Monthly Ad Revenue	\$141,000	(Real Revenue ...)
Monthly Business Users (% of all)	10%	Estimate
Monthly Green Tech Revenue	\$114,000	(Real Revenue ...)

Table 66: Estimates for Factors Affecting Deal-Making

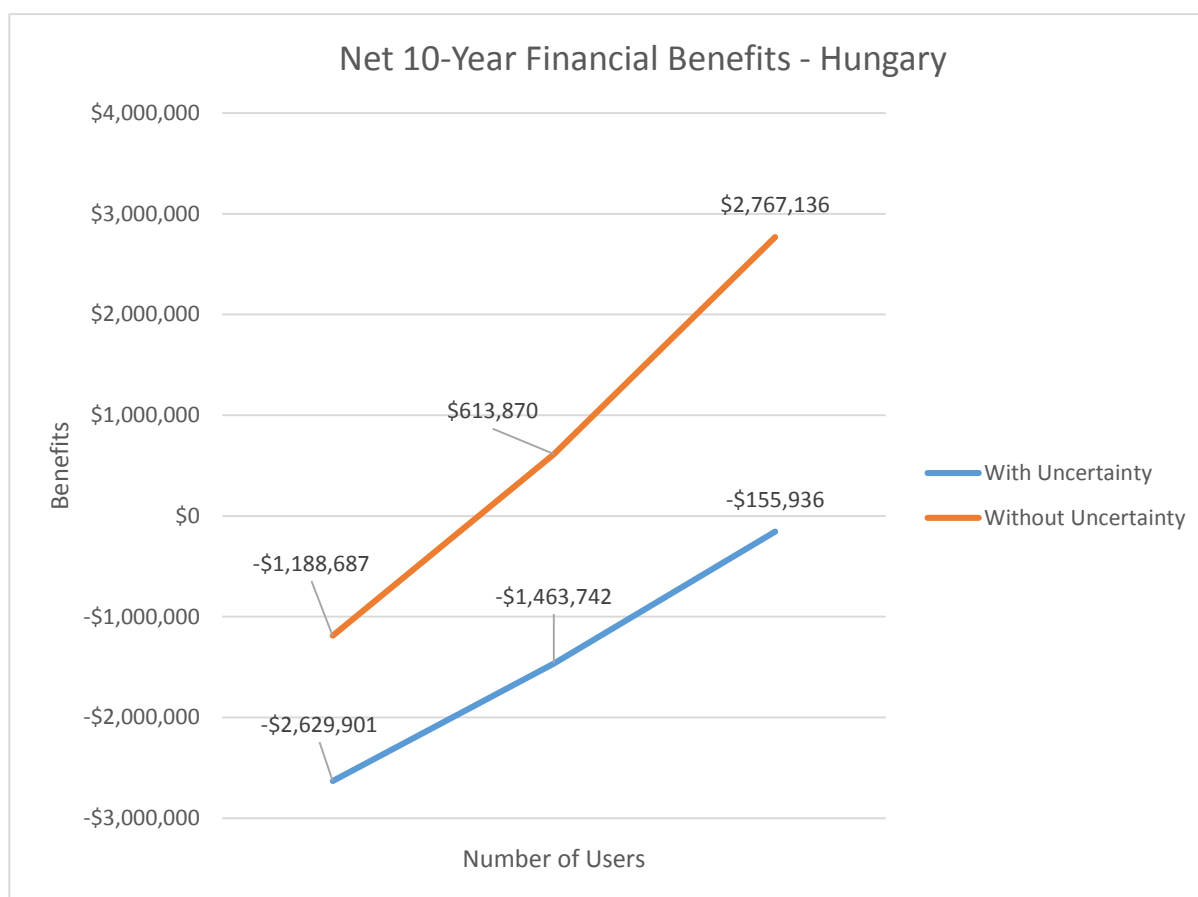
Title	Value	Source
Number of transactions per user:	2.00%	Estimate
Average value of subject matter:	\$222,222	Estimate
Cost of lawyer in percentage of value of S.M.:	3%	(Pricing Example)
Chance of litigation:	1.0%	Estimate
System Will Reduce Chance Of Litigation (% of original)	10%	Estimate
Number of Contracts a Lawyer can Make in a Month	10	Estimate
Cost of one man-hour:	\$6	(Fizetesek-Average)

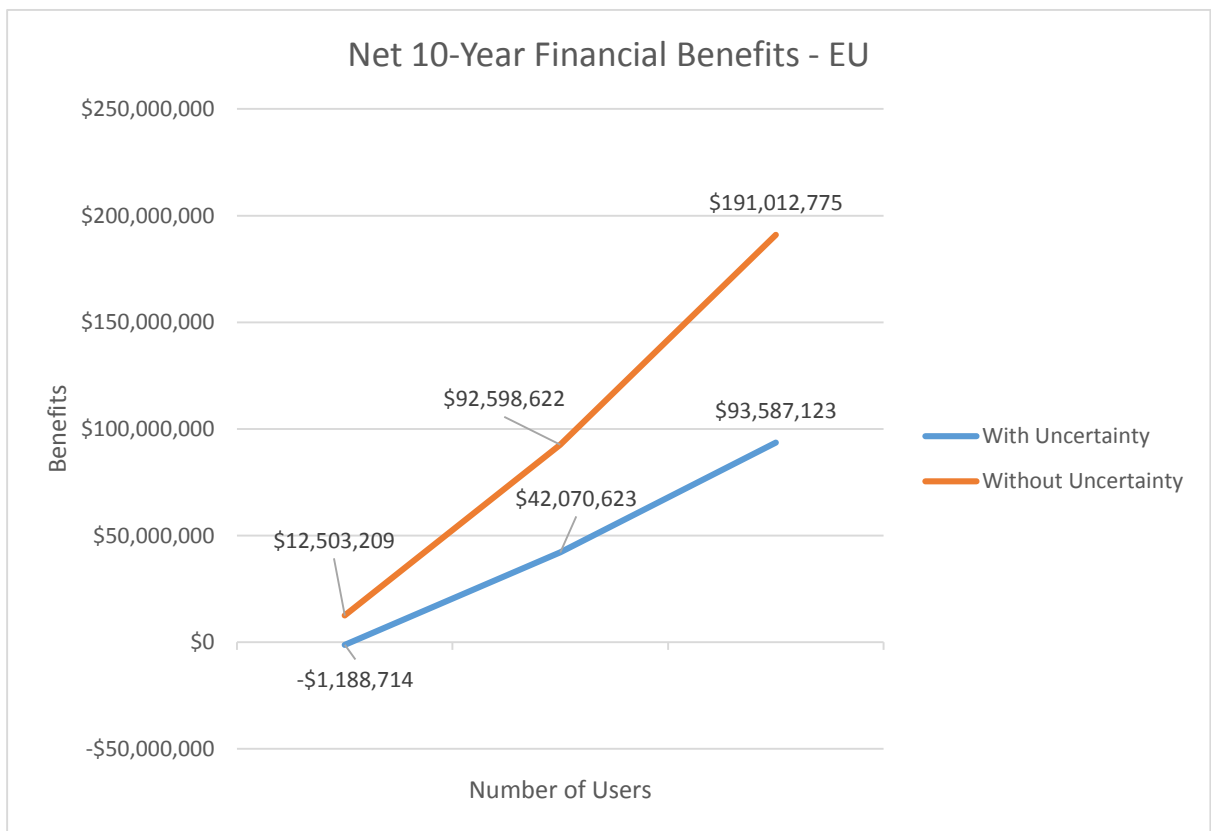
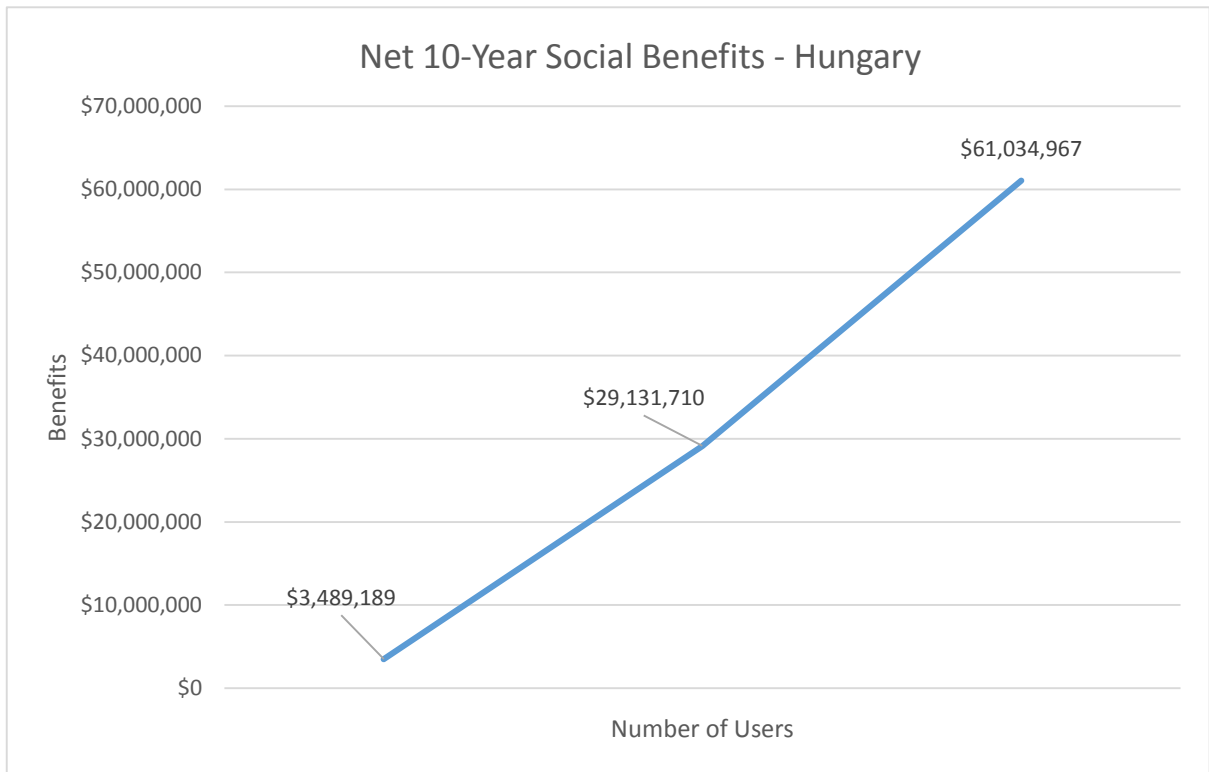
*Table 67: Breakdown of Deal-Making Cost Estimate (*the totals are adjusted for uncertainty)*

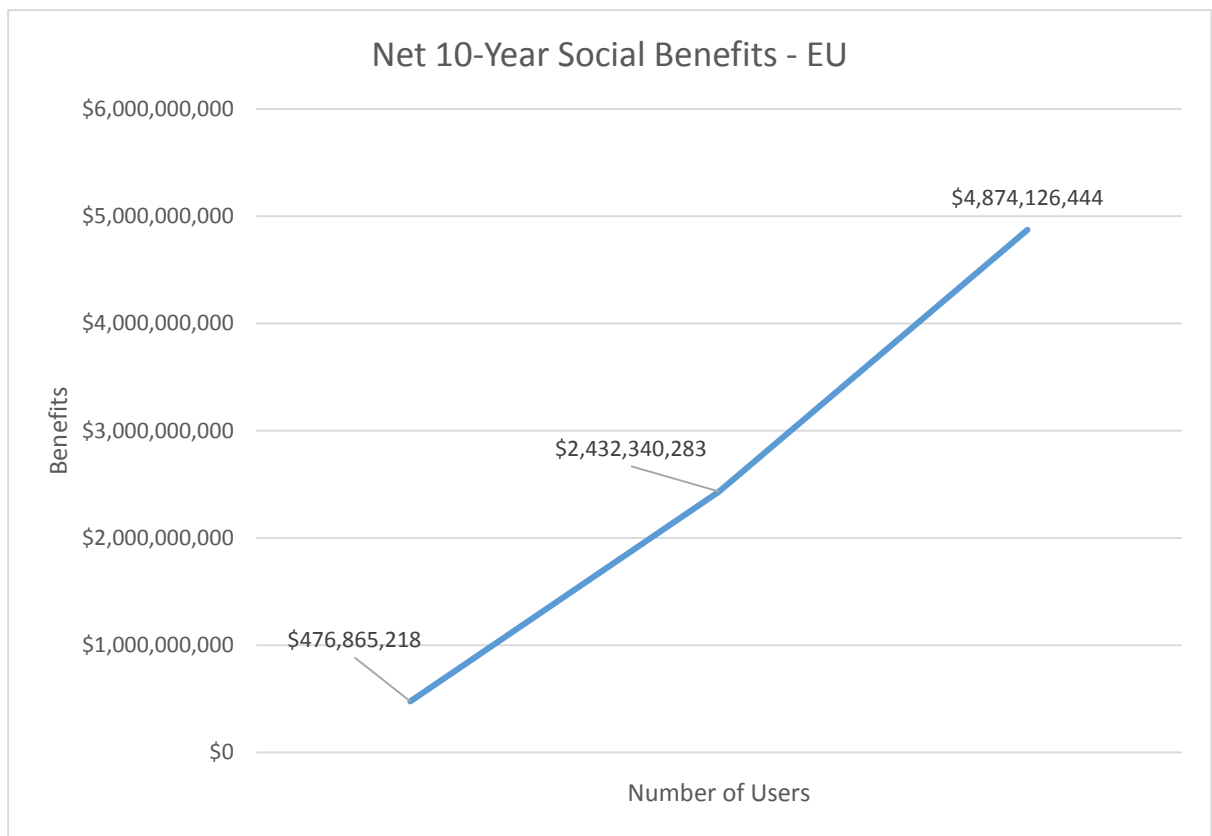
Search-Information	Before (man-hours)	After (man-hours)	Source
Find information about pollution around you	0.5	0.1	Estimate
Find parties to deal with	4	0.05	Estimate
Find alternatives	8	0.5	Estimate
EIA or other assessment	32	0.5	Estimate
Total S-I	44.5	1.15	-
Bargaining-Decision Making-Contracting	Before (man-hours)	After (man-hours)	Source
Effort spent Negotiating	20	6	Estimate
Effort of making a decision	20	6	Estimate
Lawyer costs of making a contract	17.6	0.01	Calculation
Litigation costs	11	1.1	Calculation
Total B-DM-C	69	13.11	-
Monitoring-Enforcement	Before (man-hours)	After (man-hours)	Source
Costs of reporting a problem	0.5	0.1	Estimate
Administrative costs of monitoring a deal	2	0.1	Estimate
Total M-E	2.5	0.2	-
Total	58.0*	28.92*	-

Table 69: Breakdown of Total Estimated Costs and Benefits (Thousands of \$)

Costs						
Number of Users	20,000	100,000	200,000	1,000,000	5,000,000	10,000,000
Development & Maintenance	\$1,849	\$1,849	\$1,849	\$2,890	\$2,890	\$2,890
Server Costs	\$76	\$96	\$153	\$688	\$3,441	\$6,882
Data Collection	\$55	\$55	\$55	\$1,859	\$1,859	\$1,859
Carbon Emission	\$11	\$55	\$111	\$561	\$2,798	\$5,594
Legal Support	\$271	\$271	\$271	\$4,237	\$10,593	\$21,187
Legislation	\$507	\$507	\$507	\$1,298	\$1,298	\$1,298
Administration	\$116	\$116	\$232	\$1,816	\$8,717	\$17,252
Motes	\$49	\$49	\$49	\$3,214	\$3,214	\$3,214
Total Costs	\$2,937	\$3,001	\$3,231	\$16,565	\$34,813	\$60,180
Benefits						
Ad Revenues	\$307	\$1,537	\$3,075	\$15,376	\$76,883	\$153,767
Deal-Making Benefits	\$6,119	\$30,595	\$61,190	\$478,05	\$2,390,269	\$4,780,539
Total Benefits	\$6,426	\$32,133	\$64,266	\$493,430	\$2,467,153	\$4,934,306
Net Benefits						
Total Social Benefits	\$3,489	\$29,131	\$61,034	\$476,865	\$2,432,340	\$4,874,126
Total Financial Benefits	-\$2,629	-\$1,463	-\$155	-\$1,188	\$42,070	\$93,587
TFB without Uncertainty	-\$1,188	\$613	\$2,767	\$12,503	\$92,598	\$191,012







14. The Unified Modeling Language (UML)

The Unified Modeling Language (UML) is a graphical language that helps us construct diagrams modeling complex things. ([UML... IBM](#)) The UML is not a part of the RUP, but an independent language, however the RUP requires the use of UML in the design process. The UML is not restricted to model software in any way, it can be used to model corporation structures, work activities, and so on. UML defines the syntax of the different diagrams, but not the semantics, which means that one diagram can mean several different things. In this chapter we will introduce the four most important diagrams of UML, and provide example diagrams in [Appendix 1](#). UML is used in a variety of software design application, including ([INSPIRE](#)), and is considered to be the standard language of software modeling. By providing UML diagrams for our software we can ensure that other software design teams can pick up our work where we left off.

Let's start with the Use Case Diagram. The Use Case Diagram shows us the “users” of a system and its “functions”. It has two major elements: Actors, and Use Cases. Actor is an “entity” that gets in contact with the system in a certain role. Actors have three types: Primary actors, who actively initiate interaction with the system, support actors, who don't initiate communication with the system, but the system uses them for certain use cases (external databases, etc.). The third type is background actor, who only have indirect requirements of the system, and are not showed on the use case model. In a use case diagram the functions of the system are connected with the actors who use those use cases.

The second is the class diagram. The class diagram is used to describe classes, and the relationships between classes. But first we have to define what a class is, because it is a basic element of object oriented analysis and design. A class is a group of objects that have the same type of properties, and functions. This is easier to understand through examples than definitions though: For example book is a class. All books have Authors, titles, ISBN numbers, one can read all books, etc. By object we mean a specific instance of a class. If book was our class, then Harry Potter and the Deathly Hallows is an object. The properties and functions of a class are shown on the diagram. So in a class diagram we represent classes, which are connected to each other with associations. Associations have a name, direction and multiplicity. Multiplicity gives the number of objects from the *other class*, that *one particular member of the first class* is associated with. e.g.: A man has an ID number. One particular man has exactly one ID number, and one ID number belongs to exactly one man. That association is 1 to 1. Other multiplicity types include ranges, which are denoted with 1...5, and any number of connections, which we denote with a *.

The third diagram is the sequence diagram, which is used to describe messages between objects sequentially in time. The main element of the sequence diagram is the lifeline, which is the vertical line connected to each object: The lifeline represents the time axis. The other elements are the messages, which go between lifelines horizontally. Messages often have arguments that can further enhance the meaning of the

message. (E.g. the message “error” only signals that there is something wrong, while the message “error (description)” also includes a description of that error that just occurred.) The sequence diagram has great difficulty in expressing parallel, looped or optional messages. Since UML 2.0 sequence diagrams have elements called “fragments” which are basically boxes around certain messages. Messages in a “loop” fragment are looped, and so on.

The activity diagram is used to describe not messages, but activities sequentially in time. It has no time axis, but uses arrows to signal which activity is next in execution. In contrast with the sequence diagram, the activity diagram can express conditional, looped, and parallel execution with ease.

15. Early Products of RUP

In chapter 3.1.2 we promised to talk more about the artifacts of the RUP. We will only describe some of the artifacts that RUP recommends to create in the inception phase. The most important one of these is the Use Case Model. The Use Case Model basically describes all the *functional requirements* of the software. It consists of three major parts:

- The Use Case Diagram
- The description of actors
- The descriptions of use cases

We will describe actors simply, by giving a one sentence description if needed. On the other hand, describing use cases is an extremely important part. Use cases are made up of scenarios. Each scenario describes a possible path of how the use case might proceed. A short use case description only describes the main success scenario, while a fully dressed use case description describes all possible scenarios, including general things about a use case, like pre-conditions, actors, success conditions, etc. In this paper we will refrain to the short descriptions. A use case can also be described by diagrams. The two alternatives are the System Sequence Diagram (SSD), and the Activity Diagram. The former is a Sequence diagram that focuses on a black box-type description of how the actor and the system communicate. The SSD always uses a query-reply structure of messages. A single SSD describes one scenario, so more might belong to a use case. (Since UML 2.0 one can use a SSD to describe all scenarios.) The Activity Diagram is a simple adaptation of the diagram mentioned in the previous chapter to the use case. One activity diagram can always easily describe one use case, therefore it is really helpful if we wish to examine the use case as a whole.

The next artifact is the Domain Model. The Domain Model is a simple class diagram that describes the real-world things our software might interact with. The main differences to an unrestrained class diagram are that a domain model never has directional associations, and the classes never have functions, only properties. It is important to note that the Domain Model describes real-world objects and not software objects. However, we must always consider how the software will handle that object: In practice one class might be split in two. Imagine our library example: We have books. Instances of the class book might mean “Harry Potter and the...” in general, or it might mean one actual copy of “Harry Potter and the...”. Which one does our library system need? The answer is: both. We need the book in general, because if we have a hundred copies of the book in our library, we don’t want to store all the information on it a hundred times, or similarly if we have none of them currently in our library, we don’t want the information to be lost. We also need the individual book object, because our hundred copies might be at different places in the library, and are likely to be rented by different visitors. Such examples are very common in software design.

In addition to these two diagram-based artifacts, we have three text-based ones as well. The simplest is the Glossary: Its function is to clarify and define the meaning of certain things among developers to avoid confusion. The Supplementary Requirement Specification (SRS) includes all requirements that are not in the Use Case Model. That usually means non-functional requirements. These requirements are collected according to the FURPS+ model, which breaks down the requirements into the following categories:

- Functionality
- Usability
- Reliability
- Performance
- Supportability
- + :Interfaces, Packaging, Legal, Implementation, Operation

The last document is the Vision. The Vision is a general description of the software. It includes the business opportunity, and the problem description. It also includes a description of the stakeholders, description of the architecture. It also summarizes the most important requirements, the major benefits of the software, and its most important features.

16. Macroeconomic Effects

In this section we will give a short qualitative analysis of the potential macroeconomic effects of the system, while also attempting to decide whether those effects are large enough to matter. From the CBA results we can tell, that most costs and benefits are relatively small, and are unlikely to have an effect on budget balance, or the real economy. Even the social benefits, which can be as large as a few billions of dollars, are very small relative to GDP, not to mention that they are benefits over a 10-year period, and not just one year.

There is, however, one effect that is worth mentioning. To arrive at that effect, let's consider the effect of stakeholders on the distribution of rights. It would be important for this system to work well that protection rights would be distributed, and companies (and individuals as well of course) would be required to make deals to cover their pollution. The interest of companies would be of course to have to pay or invest as little as possible: They would try to convince authorities to distribute rights in a way that would mostly protect them and not individuals.

At the stakeholder analysis we decided to consider influence and impact of the stakeholders on the software project only, not the entire process (legislation, etc.). However, it must be said, that companies would have a large influence over the distribution of rights (through lobbying), and also a very large – and definitely negative – impact. What does this have to do with macroeconomics? Well, it seems quite probable, that these vested interests would try to convince policymakers with macroeconomic arguments that would include the following three:

1. Making companies pay to individuals or invest into greener technologies will hurt businesses, destroying “confidence”, and hurting long-run growth prospects.
2. Making companies to invest more into green technologies will not increase investment, since the “money has to come from somewhere, and businesses will just cancel investments they would have made otherwise”.
3. Even if the demand for investments increased, it would only drive up interest rates, since interest rates are determined by the supply and demand of “funds”.

The first argument is quite interesting, since it supposes that for some reason money taken from businesses is wasted. This is not true. If we suppose that the law behind the system mostly protects individuals, and that the pollution belongs to the people on whose property it is emitted, then this system would redistribute wealth, and investment between social groups and industries, and the system would also make markets more efficient. The redistribution would mostly occur from more affluent agents to poorer ones, and from pollution-heavy industries to more environmentally friendly industries. It is very hard to see, how any of these effects would hurt short or long run growth prospects.

If the second argument was true, it would be really hard for the economy to grow. If increased demand in one sector would necessarily cancel out an equal amount of demand somewhere else, that no change to consumer or investor sentiment would ever affect the economy. To put it in other words, the amount of spending in an economy is the amount of money times the velocity of the money, and if businesses are made to invest more, that the velocity will increase. Of course it would be foolish to suppose, that if a business is made to invest \$100, then investment demand will increase by \$100. Some businesses will cancel out other investments, while some will not.

The falseness of the third argument is by far the most important, for it betrays a fundamental misunderstanding of how the interest rate works. The supply and demand for investment (or the loanable-funds market, defined by the IS curve) does not set the rate of interest. It only sets possible combinations of the interest rate and GDP. And it is the other market (the liquidity-preference market defined by the LM curve), that “chooses a point” on the IS curve. So if the demand for investment rises, that will make the rate of interest rise, but that higher interest rate will make the cost of holding money higher. That higher cost will make people to spend the money faster, thus the velocity of money will increase, and as a result, total spending will increase as well. Because of the higher in total spending, total incomes, and the amount of savings will rise, which will make the interest rate fall, and we reach a new equilibrium at a somewhat higher rate of interest, and a somewhat higher GDP ([DeLong, 2011](#)).

What determines the magnitudes of the rise in the rate of interest and GDP is how far the economy is from potential output. If the economy is roughly at potential output, then the effect will be mostly a rise in the interest rate. If the economy is below potential output, then the effect will be a rise in GDP. Europe is quite clearly in a Keynesian liquidity-trap, with interest rates at the zero lower bound and GDP well below potential. That means, that such a rise in investment demand will almost exclusively cause a rise in GDP, and not a rise in the interest rate. *But that is true only until the Union has fully recovered.* On the Web, we can find Nobel Prize-winning economist Paul Krugman making similar points. ([Krugman 2011](#)) ([Krugman 2013](#))

In other words, this system will likely have a similar effect to a government stimulus, except for the effect on government debt. This means, that our qualitative analysis says that our system will cause a positive macroeconomic effect. The question to ask now is this: Is there a possibility, that this effect will be serious? To answer that question, we will make a few assumptions to be able to give a by-and-large guess of the magnitude of this effect: We know the number of transactions, and the average value of the subject matter, and we estimate that 60% of all the transactions are new, we can estimate the amount of investment that the system makes companies to undertake. (Here, the average value of the subject matter is adjusted by the price level difference when considering the case of the EU.)

$$\text{Forced Investment} = \text{Yearly \# of Trans.} * \% \text{ of new ones} * \text{Avg Value} \quad (16)$$

We also estimate the ratio between forced investment and actual increase in investment demand to be 0.25. To the increased demand, we can apply a fiscal multiplier of 1.5 (this value is used by most economists today ([DeLong-Summers 2012](#)) ([Krugman 2009](#)) ([Blanchard 2013](#))) to get the effect on GDP.

$$GDP\ growth = Forced\ Invest.*\ Demand\ Increase\ \% * Fiscal\ Multiplier \quad (17)$$

We can see, that our guess shows, that this system would add about 0.25-0.3% to the annual GDP per year in the realistic case. It is very important to point out what this estimate *isn't*. It does *not* say that our system is likely to add 0.25% to the GDP annually. It only says, that we could make estimates that don't seem unreasonable at first glance that gave us this result. It only means that it is plausible, that our system would have a macroeconomic impact that is measureable as a percentage of annual GDP. It is also important, that a fiscal multiplier of 1.5 only applies in a liquidity trap. Once the EU recovers, the multiplier will be smaller than 1.

Table 70: Estimates for Macroeconomic Factors

Title	Value (Hungary)	Value (EU)	Source
Fiscal Multiplier	1.5	1.5	Assumption
Forced investment will increase demand by	25%	25%	Estimate
Percent of all transactions that are new	60%	60%	Estimate
Average value of subject matter	\$222,222	\$347,222	Estimate
GDP (millions of \$)	\$225,000	\$16,566,000	Actual Data

Table 71: Estimated 1-Year Effect on GDP

Number of New Transactions a Year	2,880	14,400	28,800	144,000	720,000	1,440,000
Effect in Year 1 (millions of \$)	\$120	\$600	\$1,200	\$9,375	\$46,875	\$93,750
In % of GDP	0.053%	0.267%	0.533%	0.057%	0.283%	0.566%

17. Additional References

- (Bérkalkulátor) [Bérkalkulátor](http://www.hrportal.hu/index.phtml?page=berkalkulator_2013): http://www.hrportal.hu/index.phtml?page=berkalkulator_2013
- (Blanchard, 2013) [*Growth Forecast Errors and Fiscal Multipliers*](#), Olivier Blanchard, Daniel Leigh, 2013
- (DeLong, 2011) [*A Socratic Dialogue: Liquidity Preference, Loanable Funds, And European Hedge Funds That Fear The Collapse Of U.S. Treasury Bond Prices - Grasping Reality With Both Hands*](#), <http://delong.typepad.com/sdj/2011/06/a-socratic-dialogue-liquidity-preference-loanable-funds-and-european-hedge-funds-that-fear-the-collapse-of-us-treasury-b.html>
- (EU Budget) <http://www.europarl.europa.eu/aboutparliament/en/00059f3ea3/The-budget-of-the-European-Parliament.html>
- (EU Parliament Schedule) <http://www.europarl.europa.eu/pdf/general/cal2013.pdf>
- (EuroStat) [*EuroStat*](#): http://epp.eurostat.ec.europa.eu/statistics_explained/index.php/Comparative_price_levels_of_consumer_goods_and_services
- (FactSheet) [*Carbon Pricing Fact Sheet*](#): <http://www.eesi.org/fact-sheet-carbon-pricing-around-world-17-oct-2012>
- (FizeteseK-Admin) [*FizeteseK.hu*](#): <http://www.fizeteseK.hu/fizeteseK/adminisztracio/adminisztrativ-alkalmazott>
- (FizeteseK-Average) [*FizeteseK.hu*](#): <http://www.fizeteseK.hu/elemzesek/a-magyar-atlagfizeteseK/50054>
- (FizeteseK-Expert) [*FizeteseK.hu*](#): <http://www.fizeteseK.hu/fizeteseK/oktatas-kepzes-tudomany-kutatas/kutatasi-es-tudomanyos-dolgozo>
- (FizeteseK-Lawyer) [*FizeteseK.hu*](#): <http://www.fizeteseK.hu/fizeteseK/jog/ugyved>
- (FizeteseK-Software) [*FizeteseK.hu*](#): <http://www.fizeteseK.hu/fizeteseK/informacios-technologiak/szoftvermernok>
- (Hungarian Budget) http://www.parlament.hu/irom39/07655/adatok/2013_tvjav_0615.pdf
- (Hungarian Parliament Schedule) <http://www.parlament.hu/naplo/index.htm>
- (InfraStructureX) <http://www.apc.com/tools/isx/tco/>
- (Krugman 2009) [*Multiplying Multipliers*](#): http://krugman.blogs.nytimes.com/2009/10/01/multiplying-multipliers/?_r=0
- (Krugman 2011) [*IS-LMentary*](#): http://krugman.blogs.nytimes.com/2011/10/09/is-lmentary/?_r=0
- (Krugman 2013) [*Invest, Divest and Prosper*](#): <http://www.nytimes.com/2013/06/28/opinion/krugman-invest-divest-and-prosper.html>
- (Mote Product Data) <http://www.pages.drexel.edu/~kws23/tutorials/motes/motes.html>
- (Parliament Website) <http://www.parlament.hu/fotitkar/tvalk/torvenyalkotas.htm>
- (Text Search) <http://www.europarl.europa.eu/plenary/en/texts-adopted.html#sidesForm>

(UML... IBM) [*UML diagrams*](#), IBM