Ballistic Missle Warning System

Warning Message

-warningMessage: string
-warningType: string
-langauge: string

+setMessage()
+setType()
+getVerification()
+setLanguage()

Verification

+verifyYes()
+verifyNo()
+sendMessage()
+cancelMessage()

message

Device

-deviceType

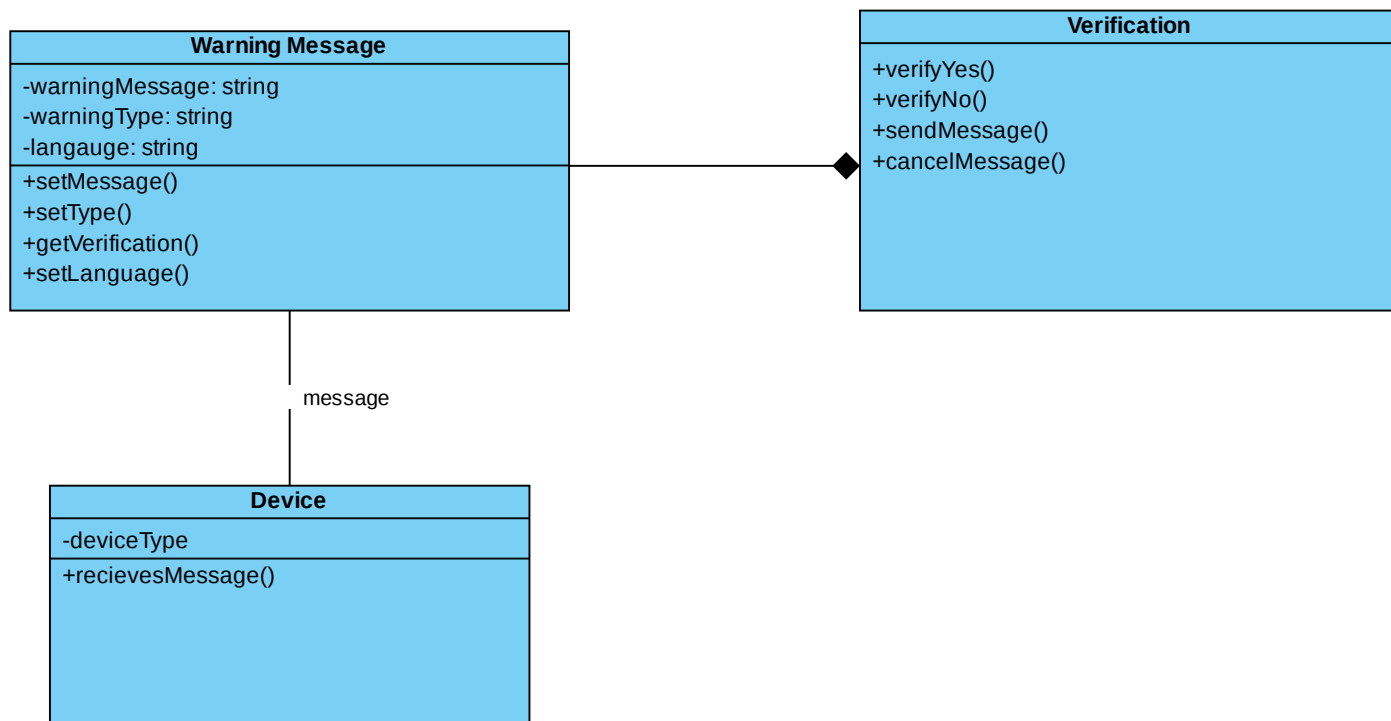+recievesMessage()

1) The incoming ballistic missile alert system:
    a) Shall display alert message for users across at least 5 different kinds of devices (TV, iphone, android, laptop, tablets)
    b) Shall have a verification process that involves the local government officials, so they can approve the message before it is sent out
    c) Shall send the message out in 2 of the most popular languages in the region.
    d) See attached PDF for class diagram
2) "as an investor, I want to be able to filter stocks by price-to-earnings so I can pick stocks that have the price-to-earnings that I want", "as an investor, I want to be able to filter stocks by price/ earnings-to-growth so I can pick stocks that have the price/ earnings-to-growth that I want", "as an investor, I want to be able to filter stocks by price-to-sales so I can pick stocks that have the price-to-sales that I want", "as an investor, I want to be able to filter stocks by price-to-book so I can pick stocks that have the price-to-book that I want", and "as an investor, I want to be able to filter stocks by debt-to-equity so I can pick stocks that have the debt-to-equity that I want"
    a) See attached PDF
3) Reading
    a) The first approach identifies candidate classes by considering all nouns and noun phrases in use case description as candidate classes and then selecting significant classes from candidates. The second approach identifies classes by checking if nouns in use case descriptions fall into classic categories such as tangible things, roles, concepts, events, and so on. Lastly, the goal of this approach is to identify classes from use cases goals rather than use case descriptions with the use case driven process. There are seven steps in this approach, determine the goal of each use case, identify entities using the goal of use cases, specify entities as classes and their features as attributes of the classes in the diagram, specify references between entities as association relationships between classes, identify collaboration between use cases and entities, specify collaboration as operation of classes, test use case-entity diagrams and class diagrams.
    b) I used the first approach of considering nouns and noun phrases and then I considered those as classes. For instance, I knew I needed to map out all of our metrics, so I started with the class "metric". I want the user to be able to choose between 5 key stock metric filters so from there I created a class for each metric that inherited from the main class. Each of those classes had their own specifications and operations.
    c) I used the same approach as above. I think this approach helps you in simple terms lay out what your service is trying to achieve. Rather than getting all jumbled in trying to explain the entire thing, start with nouns and phrases that come to mind and just link their relationships from there.

d) This paper showed me new ways to take use cases and transform them into class diagrams. For instance, I hadn't considered identifying classes from use cases goals rather than use case descriptions with the use case driven process, such as with the third approach for transforming use cases to class diagrams.

4) See attached PDF

Stock metric filters

-filterType: string
-metricValue; int

+setFilter()

P/E filter

-priceToEarningsMetric: int

+displayMetric()

P/B filter

-priceToBookMetric: int

+displayMetric()

D/E filter

-debtToEquityMetric; int

+displayMetric()

PEG filter

priceToEarningsToGrowth: int

+displayMetric()

P/S

priceToSalesMetric: int

+displayMetric()

Investor

Stock Screener Search

Company data

Company name

metric request

Price-to-book

Fetch data

Show user data

Investor

Stock Screener Search

Company data

Company name

metric request

Price-to-book

Fetch data

Show user data

Investor

Stock Screener Search

Company data

Company name

metric request

Price-to-book

Fetch data

Show user data

Investor

Stock Screener Search

Company data

Company name

metric request

Price-to-earnings

Fetch data

Show user data

Investor

Stock Screener Search

Company data

Company name

metric request

Debt-to-equity

Fetch data

Show user data