UML Class Diagram

User
hasIssue()

0...* ◇———— 1...1

Issue
Software
Hardware
openTicketSystem()
selectType()

Progress
time
new
in progress
on hold
finished
resubmitIssue()
waitForNotification

Software
Permission
Bug
Program Enhancement (in house)
Other
selectType()
provideExplanation()

Hardware
provideExplanation()

Notification
finished = true
ticketResolved()

Explanation
Visual Description
Textual Description
submitIssue()
checkProgress()

When the **user has an issue**, s/he **selects** whether it is a **software** issue or a **hardware** issue.
If it **is a software issue**, the user **selects** whether it **is a permission, bug, program enhancement, or other** issue.
Then the user **provides a visual and/or textual explanation** for the issue.
If it **is a hardware issue**, user goes straight to **providing a visual and/or textual explanation.**
When finished, **user submits the issue**
After issue submission, **user can check progress of the ticket (time the ticket has been marked new, in progress, on hold, or finished)**
If on hold or new for > 1 month, **user can resubmit ticket**
Else, **user waits for finished ticket notification**

Example	Grammatical Construct	UML model component
user	improper noun	class
has(issue)	having verb (improper noun)	aggregation (class)
select	doing verb	operation
is(software/hardware)	being verb (adjective, improper noun)	inheritance, (attributes/class)
select	doing verb	operation
is(bug, enhancement, permission, other)	being verb (adjectives)	inheritance (attributes)
provide(explanation)	doing verb (improper noun)	operation (class)
visual, textual	adjectives	attributes
submits(issue)	doing verb	operation
check(progress)	doing verb (improper noun)	operation (class)
time, new, in progress, on hold, finished	adjectives	attributes
resubmit(issue)	doing verb (improper noun)	operation (issue class)
wait for (notification)	doing verb (improper noun)	operation (class)
finished	adjective	attribute