1.0 Product Name & Description

Product name: CardoGlass™

Description: The CardioGlass system links a bio-sensor, Google Glass, and the user's smartphone together to give constant real-time feedback on the user's current vital health information, which will allow the user to take actions to reduce their heart rate, calm themselves or administer medication. The system also retains the user's data to be reported or further analysis by health care providers and researchers for the sake of providing a cost effective and innovative health care solution for the user.

2.0 Product Vision Statement

Target customer:	People with or wanting to prevent health issues
Need or opportunity for customer:	Patients want to have access to cardiac information without the help of a health care professional
Category of product:	Mobile health device
Key benefit:	Patients are presented with real-time heart and stress information so that they can react and mitigate the stress
Competing alternative:	Alere™ MobileLink, Healthbook, iWatch, EKG, Holter and Event Monitors
Primary differentiation:	CardioGlass gives a patient a real-time concept of their overall health telling them when they are doing well, and when they may need to take action. CardioGlass also offers suggestions on ways to mitigate health risks.

3.0 Key Project Stakeholders

Stakeholder role: Patient
Major project/product concerns: Security of their health information, readability of their health information, cost/return on investment, social aspects of wearing a medical device, accessibility

Stakeholder role: Patient's Family
Major project/product concerns: Security of their health information, readability of their health information, sharing a family member's health information, accessibility

Stakeholder role: Health Care Provider
Major project/product concerns: Better patient outcomes, accessibility, security of health information, accuracy of data

Stakeholder role: Medical Researcher
Major project/product concerns: Getting accurate data that complies with their internal review board standards

Stakeholder role: Developer
Major project/product concerns: Hardware and software updates on Google Glass, Health Care data transfer standards

Stakeholder role: RSI
Major project/product concerns: Legality of Google Glass, Social attitudes toward Google Glass, cost/return on investment

Stakeholder role: U.S. Department of Health & Human Services
Major project/product concerns: Legality of Google Glass, HIPPA compliance

4.0 Features Backlog

Features Backlog
As a Patient, I want CardioGlass to provide real time suggestions on how to lower my stress level, blood pressure, heart rate if applicable.
As a Patient, I want to be able to easily see how healthy I am compared to an average human like me.
As a Patient, I want to be able to see how healthy I am now compared to a time period in the past.
As a Patient, I want the font to be large and clear to read so that I don't misread the information.
As a Patient, I want to be able to monitor my vital signs and be alerted when any of them reach dangerous levels.
As a Patient, I want to be able to interact with the CardioGlass system using voice commands so that I will be less distracted during an activity.
As a Patient, I want the CardioGlass's bio-sensing unit to be light, compact, and discreet so that I can perform an activity in a natural way without attracting negative attention.
As a Patient, I want CardioGlass to produce easy-to-read statistical information and charts about my health periodically.
As a Patient, I want CardioGlass to aid me in making sure I am medicating myself correctly.
As a User, I want CardioGlass to store my longer medical record information and collected data in a central way so that I can retrieve the needed information.
As a User, I want to be access longer medical record information and collected data from different platforms and devices.
As a User, I want to be able to backup information and data gathered by CardioGlass using cloud computing technology.

As a User, I want to be able to easily recognize if I am signed in or out so I know my data is secure.
As a Health Care Provider, I want to be able to have access to the patient's medical history and I want to be alerted of any problems the patient experienced while not with me.
As a Health Care Provider, I want to be able to issue periodic updates to the patient's CardioGlass system based on actual medical records so that CardioGlass can come up with better and more accurate assessments and suggestions customized to meet the patient's needs.
As a Developer, I want to be able to use data collected by the bio-sensing unit to come up with innovative apps using the collected data in real time.
As a Developer, I want to be able to have access to data gathered from real CardioGlass users in order to enhance software user experience.
As a Billing Department worker, I want to be able to access the user's time logged on to the device, so that I know it is being utilized.
As a Researcher, I want to be able to access data that is deep and broad and is collected in a manner consistent with internal review boards so that I can further my research.
As the U.S. Department of Health & Human Services, I want medical devices maintain compliance with HIPPA regulations.

5.0 Risks/Opportunities

Risks/Opportunities
Risk: Many different standards for health care data exist, if we choose one that is not relevant to the technology doctors intend to interface with, this may slow adoption.
Risk: If a user's PHI (Protected Health Information) is exposed, the responsible party will pay a hefty fine.
Risk: There is a general concern with the aging of technology. As time goes on, some stubborn or lazy patients will continue to use older versions of hardware and software. This could lead to issues with the software working properly, potentially causing dangerous situations.
Risk: CardioGlass is not a stand-alone technology, and is dependent on other technologies, namely Google Glass, iOS, and Android. This might result in complex software development, maintenance, integration, and testing process. Problems related to such processes might surface as a result of Google Glass, iOS, or Android updating their hardware or software.
Risk: CardioGlass, Google Glass, and an iOS/Android smartphone have to be connected to each other at all times in order for the product to work. A smartphone might also have to be connected to Wi-Fi to take advantage of the full capability of the integrated system. If not, this could lead to functionality failure and render CardioGlass useless at times.

Risk: CardioGlass is a new technology and monitoring the physiological parameters of the user and sensors might have to be researched and developed, making sure that it gives accurate and reliable readings while maintaining the portable aspect of CardioGlass's bio-sensing unit. Since RSI has no experience in mobile technology, this could introduce a risk related to hardware design and development.
Risk: Having three technologies interacting with each other might be overwhelming in regards to interfaces, menus, settings, usability, and user experience. A great deal of users who are concerned about their health might be elderly, who might not be familiar with cutting edge technology such as the CardioGlass system. If the system is not user-friendly, this might throw a lot of such potential users off.
Risk: The CardioGlass system consists of three devices working together; the bio-sensing unit, Google Glass, and iOS/Android smartphone. This means that the user have to consider the cost of all three products in order to get the benefit from using the CardioGlass system. This might not appeal to a great deal of consumers who only need the functionality of CardioGlass, or those who do not own a smartphone and Google Glass already.
Risk: Google Glass cannot be used everywhere due to legal issues and privacy concerns in some public places as of yet, which in turn puts limitations on CardioGlass usage as it is dependent on Google Glass.
Risk: The success of CardioGlass is dependent on the success of Google Glass, which is a promising technology, but has no certain future yet.
Opportunity: Wider analysis of patient data may occur leading to improved patient outcomes.
Opportunity: If the health care field is ready for the CardioGlass it may embrace the technology and improve patient outcomes.
Opportunity: Google Mirror API uses a RESTful architecture - RESTful architectures are robust and relatively simple to interact with and there is a large body of support technology to get the technologies talking to each other.
Opportunity: Google wants people developing for the platform - This means that the documentation for Google Glass exists and is well maintained. Google will probably keep developers in mind as they continue to refine the product, so that there can be some assurance that apps developed for Google Glass will continue to function.
Opportunity: CardioGlass would allow patients to adjust their surroundings for minor problems, which would keep them out of the doctor's office longer. Health information technology has proven its ability to do this, and thus has also kept wait times for more severe patient cases shorter.
Opportunity: A bar code scanning feature that would ensure patients are medicating themselves correctly. This eliminates the chance of a patient taking the wrong medicine or overdosing.

Opportunity: Using the bio-sensing unit with Google Glass and related technologies enables the development of mobile applications that were not able to be realized before. Google Glass displays information for the user in way that is not distracting or obstructive, which enables the user to engage in activities while getting feedback from CardioGlass. Combined with the ability to use voice commands instead of having to interact with the device through a touch screen or by clicking buttons would contribute in making CardioGlass the next big thing in the mobile-health technology and user experience.
Opportunity: Third-party applications can be developed using the bio-sensing unit and CardioGlass. This will open a new door to business opportunities for RSI to become a leading company in mobile-health technology as more and more developers become dependent on the system. Along with their vast experience in stationary health technologies, it would give RSI an edge over the competition.

6.0 Team Contributions

Team member name	List specific contributions to this assignment
Nathan Petts	Set up Stormboard, contributed Stormboard ideas, helped write assignment document, helped review assignment document
Brandon Niedert	Contributed Stormboard ideas, contributed to assignment document, helped review assignment document, professor collaboration
Nezar Alsaati	Contributed Stormboard ideas, contributed to assignment document, helped review assignment document, assignment submission

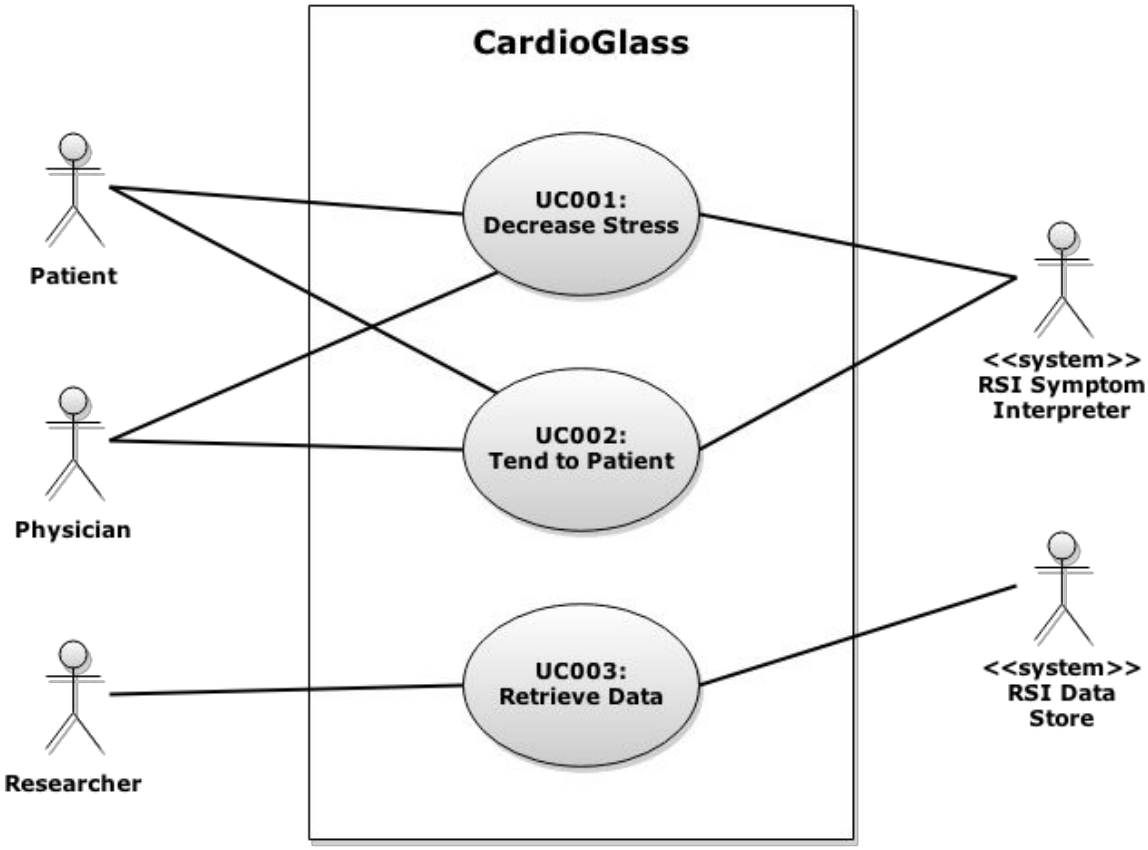
1. Brief Use Cases

Stakeholder: Patient	Use Case Identifier: UC001
Original User Story: As a patient, I want CardioGlass to provide real time suggestions on how to lower my stress level, blood pressure, heart rate if applicable.	
Use Case Name: Decrease Stress	
Actors: Patient, Physician, RSI Symptom Interpreter	

Stakeholder: Physician	Use Case Identifier: UC002
Original User Story: As a physician, I want to be alerted of any stress events the patient experienced while not with me.	
Use Case Name: Tend to Patient	
Actors: Physician, Patient, RSI Symptom Interpreter	

Stakeholder: Researcher	Use Case Identifier: UC003
Original User Story: As a Researcher, I want to be able to access data that is deep and broad and is collected in a manner consistent with internal review boards so that I can further my research.	
Use Case Name: Retrieve Data	
Actors: Researcher, RSI Data Store	

2. Use Case Diagram



3. Detailed Use Case

Use Case Identifier: UC001
Use Case Name: Decrease Stress
Actors: Patient, Physician, RSI Symptom Interpretor
Brief Description: CardioGlass detects and alerts the patient to an increased stress situation and suggests strategies to reduce stress.
Preconditions: ● Biosensor is working ● CardioGlass detects increased stress levels in the Patient ● The system can detect an internet connection ● The patient has been authenticated
Postconditions: ● Successful Completion: Patient's stress levels drop ● Failure Condition: CardioGlass directs the Patient to contact their Physician
Event Flow: 1. **Use case begins when** CardioGlass notices that the patient has an increased stressed level 2. CardioGlass attempts to detect what current action is making the patient stressed 3. If nothing is detected, CardioGlass prompts for input from the patient [Alternate Scenario A] 4. CardioGlass confirms current situation 5. CardioGlass sends the Patient's data to the RSI Symptom Interpreter 6. The RSI Symptom Interpreter responds with suggestions of actions for the Patient to take 7. CardioGlass displays suggestions for decreasing stress levels 8. Patient selects alternative 9. CardioGlass leads patient through the steps required to succeed at alternative 10. Patient responds by following instructions 11. Patient completes all instructions 12. CardioGlass checks the Patient's stress level and if the levels are high repeat steps 7-11[Alternate Scenario B] 13. **Use case ends when** the patient is no longer stressed
Alternate Flows: A. CardioGlass is unable to detect what is causing patient stress i) CardioGlass prompts user for input ii) Patient speaks about what he is currently doing iii) CardioGlass algorithm determines what is causing stress B. The Patient is not responding to the stress reduction suggestions i) CardioGlass contacts the Patient's Physician to alert them to the situation ii) CardioGlass suggests that the Patient contact Emergency Medical Services iii) the use case ends when the Patient's medical needs have been met

4. Team Member Contributions

Team Member Name	List specific contributions to this assignment
Nathan Petts	Brainstorming, meeting leader, final draft contribution and editing
Brandon Niedert	Brainstorming, Initial rough draft, final draft contribution, editing
Nezar Alsaati	Brainstorming, use case diagram, final draft contribution, edting

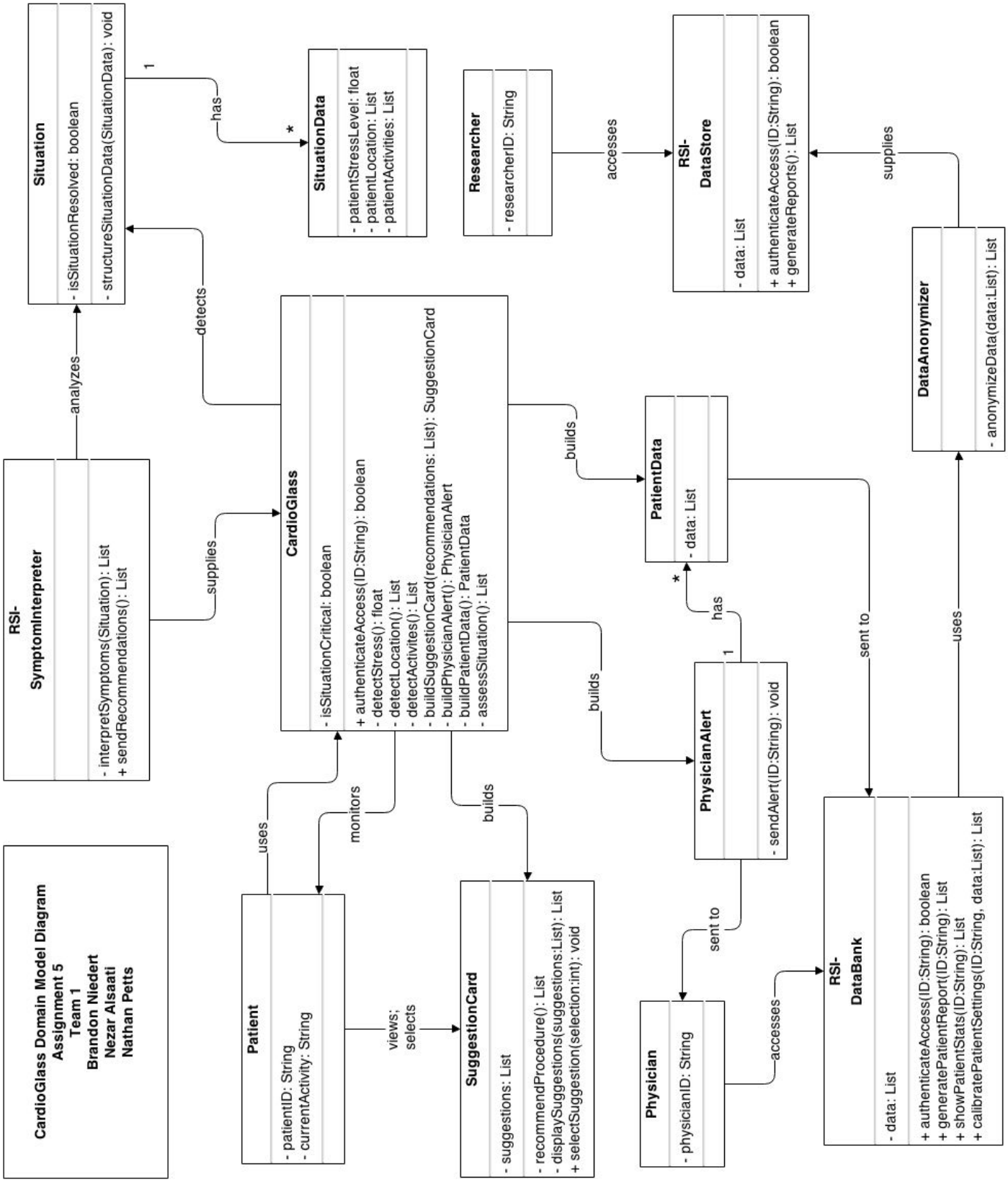
1. Abstraction Analysis Table

Conceptual Class Name	In DMD	Attributes	Responsibilities (Operations)	Collaborators
CardioGlass	x	currentCard, situation, currentLatLong, isSituationCritical	authenticate-Access, detectStress, detectLocation, detectActivites, buildSuggestion-Card, buildPhysician-Alert, buildPatientData, assessSituation	SymptomInterpreter, EmergencyServices Contact, HealthReference, SuggestionCard, Situation, Physician, Patient, HealthMilestone
Patient	x	fistName, lastName, patientID, currentLocation, currentActivity, stressLevel	runSystem, selectSuggestions, selectAlternatives	SuggestionCard, CardioGlass, HealthReference, HealthMilestone, EmergencyServices - Contact, PatientLocation, StatusCard, ActionStep, Physician, MedicalHistory
Physician	x	fistName, lastName, physicianID, hospital		PhysicianAlert, Physician'sHospital, CardioGlass, MedicalHistory, Patient, HIPPAPatient-InformationRules
Researcher	x	firstName, lastName, projectID, researcherID	viewReports, getData	ReserachData
RSISymptomInterpreter	x	symptoms	reviewSymptoms, interpretSymptoms, send-Recommendations	CardioGlass, Situation
RSIDataStore	x	data, specifiedData	scrubData, buildSpecified-DataSet, generateReports, authenticateAccess	ResearchData, Researcher, DataAnonymizer

SuggestionCard	x	suggestions	displayToUser, recommend-Procedure, displaySuggestions, selectSuggestion	CardioGlass, Patient, CardioGlass
ActionStep		stepText	displayToUser	Patient, suggestionCard
ResearchData		ageStatistics, howStressed-Statistics	downloadsAsCsv, generateReport	Researcher, DataStore
PhysicianAlert	x	patientData	sendAlert	Physician, CardioGlass, PatientData
MedicalHistory		patientName, medicalHistory,	DisplayToPhysician	Physician, HIPPA-PatientInformation-Rules, HealthCare-Technician, Patient
EmergencyServices-Contact		patientLocation, situation	alertEmergency-Services	CardioGlass, patient
PhysicianHospital		physicianName hospitalLatLong		Physician
PatientLocation		currentLatLong	sendToSituation	Situation, Patient
HealthcareTechnician		firstName, lastName, technicianID, hospital	activateCardioGlass	MedicalHistory, Patient, CardioGlass, HIPPAPatient-InformationRules, PhysicianHospital
StatusCard		patientPulse, patientStress-Level	displayToUser	CardioGlass, Patient
HealthMilestone		time	displayMilestone	CardioGlass, Patient
HIPPAPatient-InformationRules		healthInformation-Rules	checkPatient-Information	Medical History, Patient, Physician, Healthcare-Technician
HealthRefrence		data	assessHealth	PatientData CardioGlass, Patient

SituationData	x	patientStress-Level, patientLocation, patientActivities		Patient'sLocation, Situation
PatientData	x	data		DataStore, ResearchData, PhysicianAlert, RSIDataBank, CardioGlass
DataAnonymizer	x		anonymizeData	RSIDataStore RSIDataBank
RSIDataBank	x	data	authenticateAccess, accessPatientProfile generatePatient-report, showPatientStats, calibratePatient-Settings	DataAnonymizer, PatientData, Physician
Situation	x	isSituation-Resolved	structureSituation-Data	CardioGlass, RSISymptom-Interpreter, SituationData

2. Domain Model Diagram



3. Glossary

Conceptual Class Name	Definition
ActionStep	An instruction given by the Suggestion Card to relieve stress
CardioGlass	Wearable health monitoring system that monitors stress levels in patients with heart conditions
RSIDataStore	System maintained by RSI that collects patient health data, strips it of identifying information, and makes it available to researchers
EmergencyServices-Contact	When the patient is unable to contact EMS themselves, CardioGlass will do it for them
DataAnonymizer	Tool that strips data of patient identifying information
HealthcareTechnician	Any other healthcare professional assisting in the care of the patient
HealthMilestone	Messages received by patient informing them of achieving positive health goals
HealthRefrence	Reference to ideal vital sign information
HIPPAPatientInformation-Rules	Health information privacy rules that healthcare provides must abide by
MedicalHistory	Personal health information available to Physician
Patient	Person who uses CardioGlass to monitor their health
PatientLocations	Current location of the patient
PatientData	Data profile maintained by RSI that is associated with a particular patient
Physician	Doctor in charge of overseeing CardioGlass patient
PhysicianHospital	The main location that the Physician is associated with
PhysicianAlert	Alert sent to Physician when Patient experiences a stress episode
ResearchData	Structured patient data, stripped of identifying information presented to researchers
Researcher	Person who is interested in data collected by the CardioGlass system
RSIDataBank	RSI database of patients' information
RSISymptomInterpreter	System maintained by RSI to interpret symptom recorded by CardioGlass and send recommendations to CardioGlass that help the patient reduce those symptoms
Situation	All of the SituationData in a structured form that can be analyzed by the RSISymptomInterpreter

SituationData	Individual pieces of information about the patient including current location, environment, and stress level
StatusCard	periodically displayed, real time health information about the patient
SuggestionCard	Output displayed to the Patient of the suggestions provided by the RSISymptomInterpreter

4. Team Member Contributions

Team Member Name	List specific contributions to this assignment
Brandon	Added associations to diagram, contributed to Abstraction Analysis Table, contributed to Glossary, Alphabetize Glossary, editing
Nathan	Worked on the Domain Model Diagram - first and second draft, helped author the glossary, helped author the class table, editing and layout
Nezar	Contributed to Abstraction Analysis Table, Contributed to Domain Model Diagram, contributed to Glossary, editing

1. Design Scenario

Get CardioGlass Health Suggestions

1. Create abstraction **PDPackager** representing the component of **CardioGlassModel** that packages the patient current health information into a single object
2. Create abstraction **PackagedPD** representing the packaged data created by **PDPackager**
3. **PDPackager** creates instance of **PackagedPD** and packages current patient data within it
4. Create abstraction **CardioGlassRSISIProxy** representing the interface between CardioGlass and RSI Symptom Interpreter (RSISI)
5. **CardioGlassModel** sends **PackagedPD** to **CardioGlassRSISIProxy**
6. Create abstraction **RSISIBlackboardModeratorKS** acting as the controller knowledge source of the **RSIBlackboard** component in RSISI
7. **CardioGlassRSISIProxy** sends **PackagedPD** to **RSISIBlackboardModeratorKS**
8. Create abstraction **PDUnpacker** representing the component in **RSISIBlackboardModeratorKS** that handles the unpackaging of **PackagedPD**
9. Create abstraction **RSIBlackboard** representing the Blackboard architectural component in RSISI
10. Create abstraction **PatientData** representing the unpackaged data created by **PDUnpacker** from **PackagedPD**
11. **PDUnpacker** creates instance of **PatientData** and unpackages **PackagedPD** into **RSIBlackboard** for analysis and processing by the knowledge sources of the Blackboard
12. Create abstraction **SymptomAnalyzerKS** representing the knowledge source that analyzes **PatientData** along with other knowledge sources to find possible solution procedure
13. Create abstraction **PatientHistoryKS** representing the knowledge source that contribute to the analysis of **PatientData** by consulting information about the patient's medical history
14. **RSIBlackboard** notifies **SymptomAnalyzerKS** and **PatientHistoryKS** that **PatientData** is available for analysis and processing
15. **PatientHistoryKS** retrieves **PatientData** from **RSIBlackboard**
16. **PatientHistoryKS** updates **PatientData** on **RSIBlackboard** with information about the patient medical history if applicable
17. **SymptomAnalyzerKS** retrieves **PatientData** from **RSIBlackboard**
18. Create abstraction **Symptom** representing a piece of data that is found based on adverse health conditions the patient is experiencing
19. **SymptomAnalyzerKS** creates instance of **Symptom** for each unique symptom detected
20. **SymptomAnalyzerKS** compares each instance of **Symptom** to a bank of stressor data
21. Create abstraction **StressorList** representing a list of possible stressors that could be the cause of the problems the patient is experiencing
22. Create abstraction **Stressor** representing an event that is causing the patient to experience symptoms

23. **SymptomAnalyzerKS** creates instances of **StressorList** and **Stressor**, and each possible **Stressor** is added to the list as **Symptom** instances are paired to **Stressor** instances
24. Create abstraction **SolutionList** representing a list that contains solutions to the patient issues
25. **SymptomAnalyzerKS** creates instance of **SolutionList** and populates it with different stress relieving solutions based on **StressorList**
26. **SymptomAnalyzerKS** prioritizes items within **SolutionList** based on calculated relevancy
27. **SymptomAnalyzerKS** updates **RSIBlackboard** with **SolutionList**
28. **RSIBlackboard** notifies **RSISIBlackboardModeratorKS** that **SolutionList** is available for utilization
29. Create abstraction **SLPackager** representing the component in **RSISIBlackboardModeratorKS** that retrieves and handles the packaging of **SolutionList** from **RSIBlackboard**
30. Create abstraction **PackagedSL** representing the packaged data created by **SLPackager**
31. **SLPackager** creates instance of **PackagedSL** and packages **SolutionList** within it
32. Create abstraction **CardioGlassRSISIController** representing the controller architectural component in CardioGlass
33. Create abstraction **SLUnpacker** representing the component in **CardioGlassModel** that handles the unpackaging of **PackagedSL**
34. **RSISIBlackboardModeratorKS** sends **PackagedSL** to **CardioGlassRSISIProxy**
35. **CardioGlassRSISIProxy** sends **PackagedSL** to **CardioGlassRSISIController**
36. **CardioGlassRSISIController** sends **PackagedSL** to **SLUnpacker**
37. Create abstraction **CardioGlassView** representing the component handling the visual representation of information and data in CardioGlass
38. **SLUnpacker** unpacks **SolutionList** from **PackagedSL**
39. **SLUnpacker** notifies **CardioGlassView** that **SolutionList** is available for display
40. **CardioGlassView** retrieves **SolutionList** from **SLUnpacker** for display

Observers:
SymptomAnalyzerKS
PatientHistoryKS
RSISIBlackboardModeratorKS
CardioGlassView

Observables:
RSIBlackboard
SLUnpacker

2. CRC Analysis Tables

CardioGlassModel	
Responsibilities	Collaborators
Send PackagedPD	CardioGlassRSISIProxy

CardioGlassView	
Responsibilities	Collaborators
Implements Observer	SLUnpacker
Retrieve SolutionList from SLUnpacker	SolutionList, SLUnpacker
Displays SolutionList	SolutionList, Self

PatientHistoryKS	
Responsibilities	Collaborators
Implements Observer	RSIBlackboard
Retrieve PatientData	RSIBlackboard
PatientData	RSIBlackboard

SolutionList	
Responsibilities	Collaborators
Contain Possible Solutions	SymptomAnalyzerKS, StressorList

SLPackager	
Responsibilities	Collaborators
Create PackagedSL	PackagedSL
Retrieve SolutionList	RSIBlackboard

Symptom	
Responsibilities	Collaborators
Contain Symptom	SymptomAnalyzerKS
Compare with Stressor	SymptomAnalyzerKS, Stressor

PatientData	
Responsibilities	Collaborators
Contain a set of information about the patient	PDUnpacker, PatientHistoryKS, SymptomAnalyzerKS

CardioGlassRSISIController	
Responsibilities	Collaborators
Receive PackagedSL	CardioGlassRSISIProxy
Send PackagedSL	SLUnpacker

PackagedPD	
Responsibilities	Collaborators
Contain patient data	PDPackager

PDPackager	
Responsibilities	Collaborators
Create PackagedPD	PackagedPD
Package patient data within PackagedPD	PackagedPD

RSIBlackboard	
Responsibilities	Collaborators
Notify of PatientData Availability	SymptomAnalyzerKS, PatientHistoryKS, PatientData
Notify of SolutionList Availability	RSISIBlackboardModeratorKS , SolutionList
Implements Observable	SymptomAnalyzerKS, PatientHistoryKS, RSISIBlackboardModeratorKS

Stressor	
Responsibilities	Collaborators
Contain Stressor Data	SymptomAnalyzerKS
Compare with Symptom	SymptomAnalyzerKS, Symptom

SymptomAnalyzerKS	
Responsibilities	Collaborators
Implements Observer	RSIBlackboard
Analyze PatientData	RSIBlackboard
Create Symptom	Symptom
Compare Symptoms	Symptom
Create StresorList	StressorList
Create Stressor	Stressor
Populate StressorList	Stressor, StressorList
Create SolutionList	SolutionList
Populate SolutionList	SolutionList, StressorList
Prioritize SolutionList	SolutionList
Update SolutionList Results	RSIBlackboard, SolutionList

CardioGlassRSISIProxy	
Responsibilities	Collaborators
Receiver PackagedPD	CardioGlassModel
Send PackagedPD	RSISIBlackboardModeratorKS
Receive PackagedSL	RSISIBlackboardModeratorKS
Send PackagedSL	CardioGlassRSISIController

PackagedSL	
Responsibilities	Collaborators
Contain SolutionList	SLPackager, SolutionList

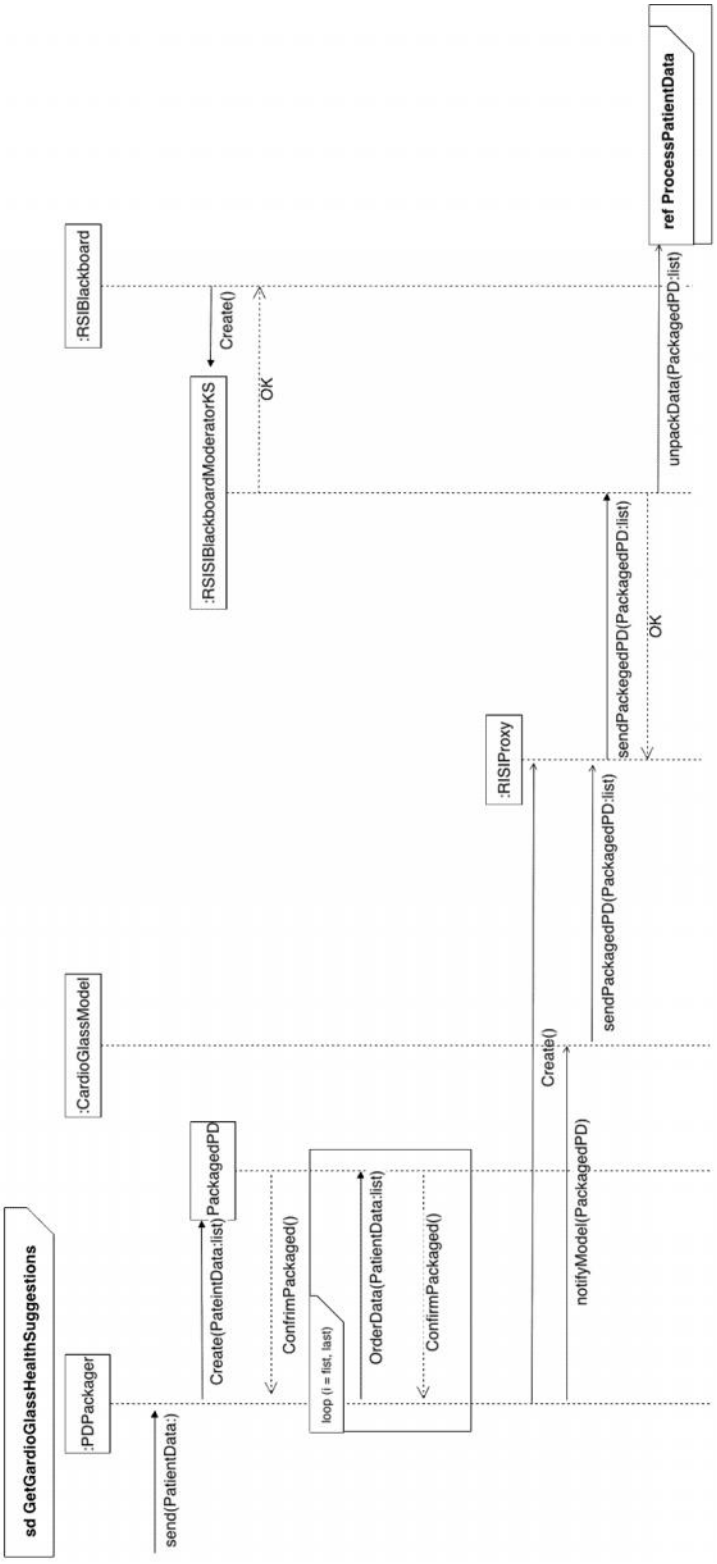
PDUnpacker	
Responsibilities	Collaborators
Unpack PackagedPD	PackagedPD
PatientData	PatientData
Post PatientData	RSIBlackboard

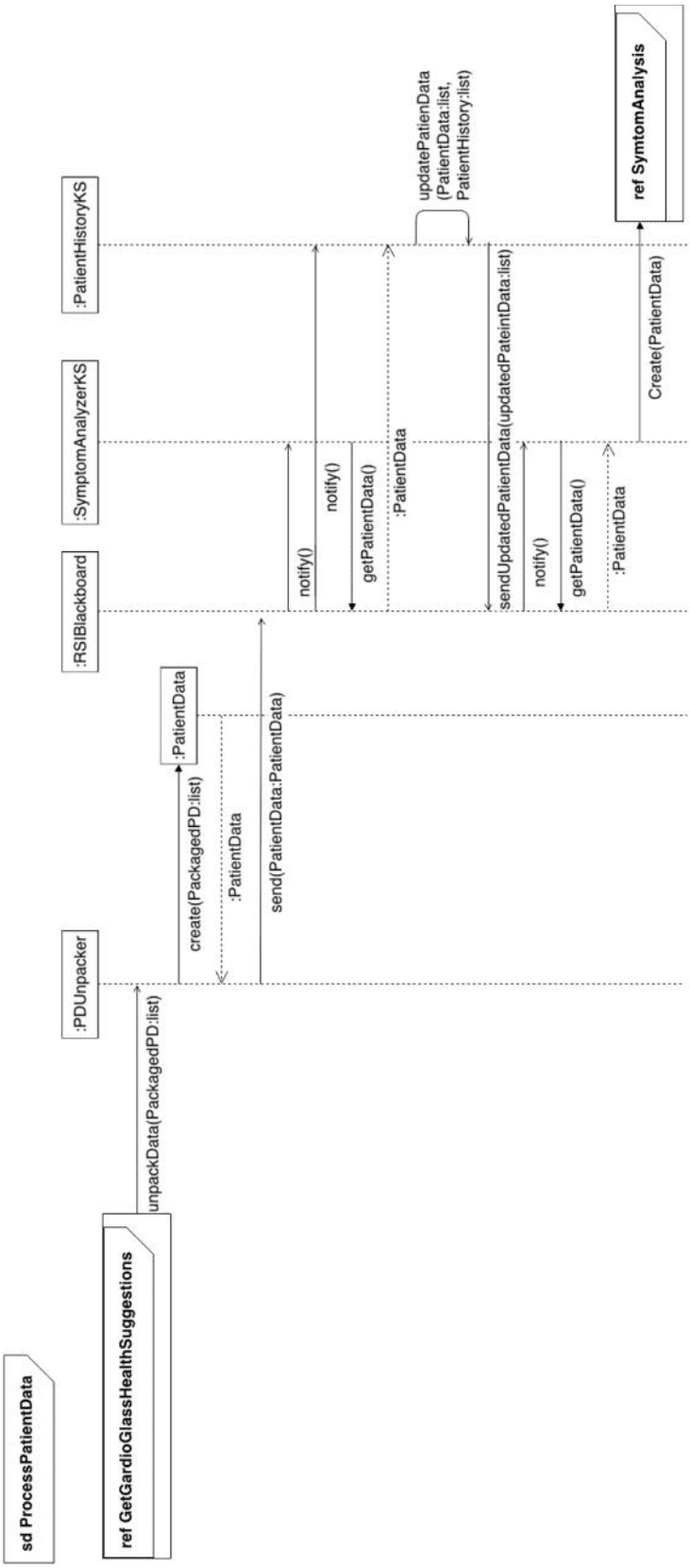
RSISIBlackboardModeratorKS	
Responsibilities	Collaborators
Receive PackagedPD	PackagedPD, CardioGlassRSISIProxy
Send PackagedSL	PackagedSL, CardioGlassRSISIProxy
Implements Observer	RSIBlackboard

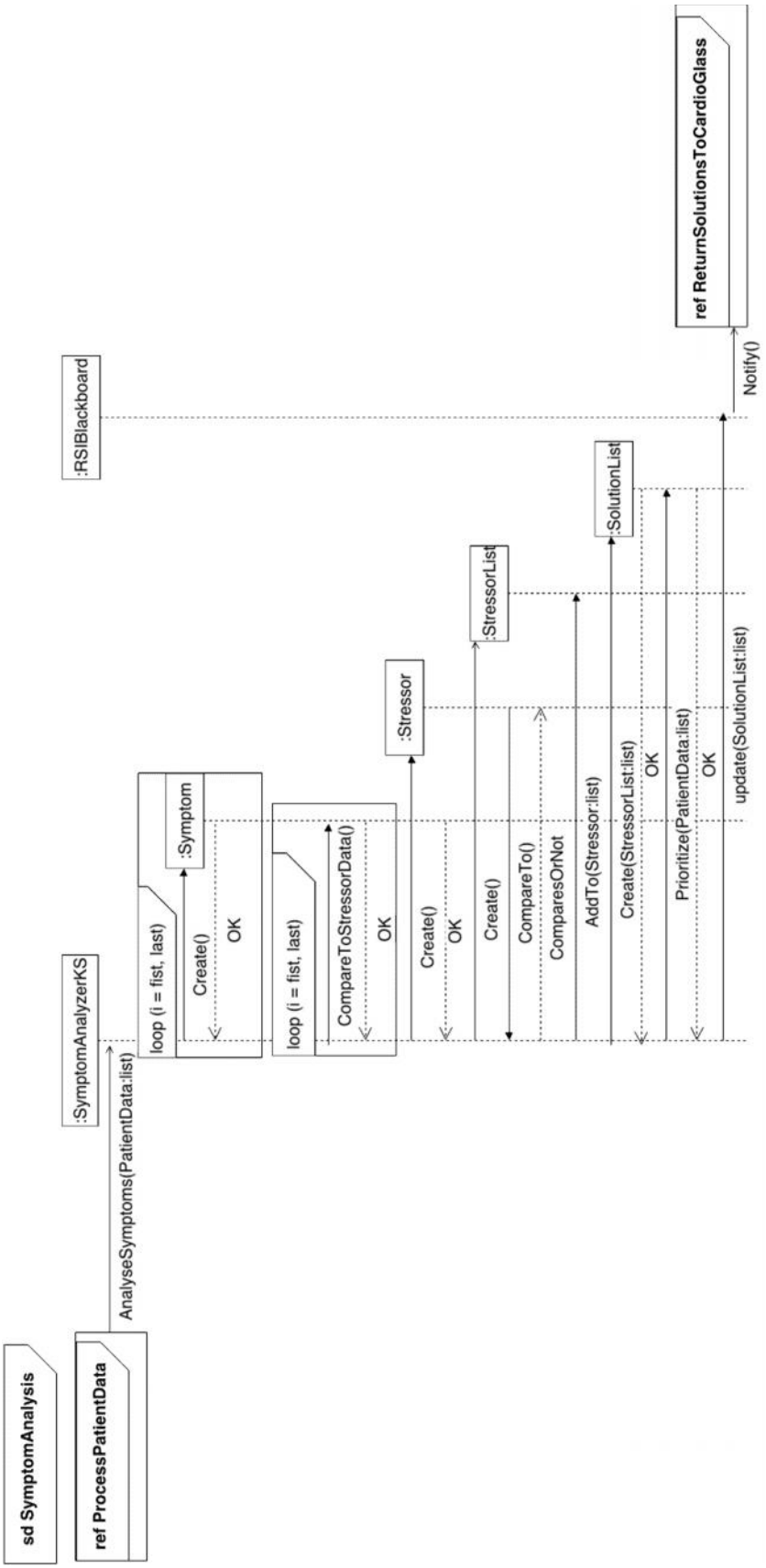
StressorList	
Responsibilities	Collaborators
Contain Stressors	Stressor, SymptomAnalyzerKS

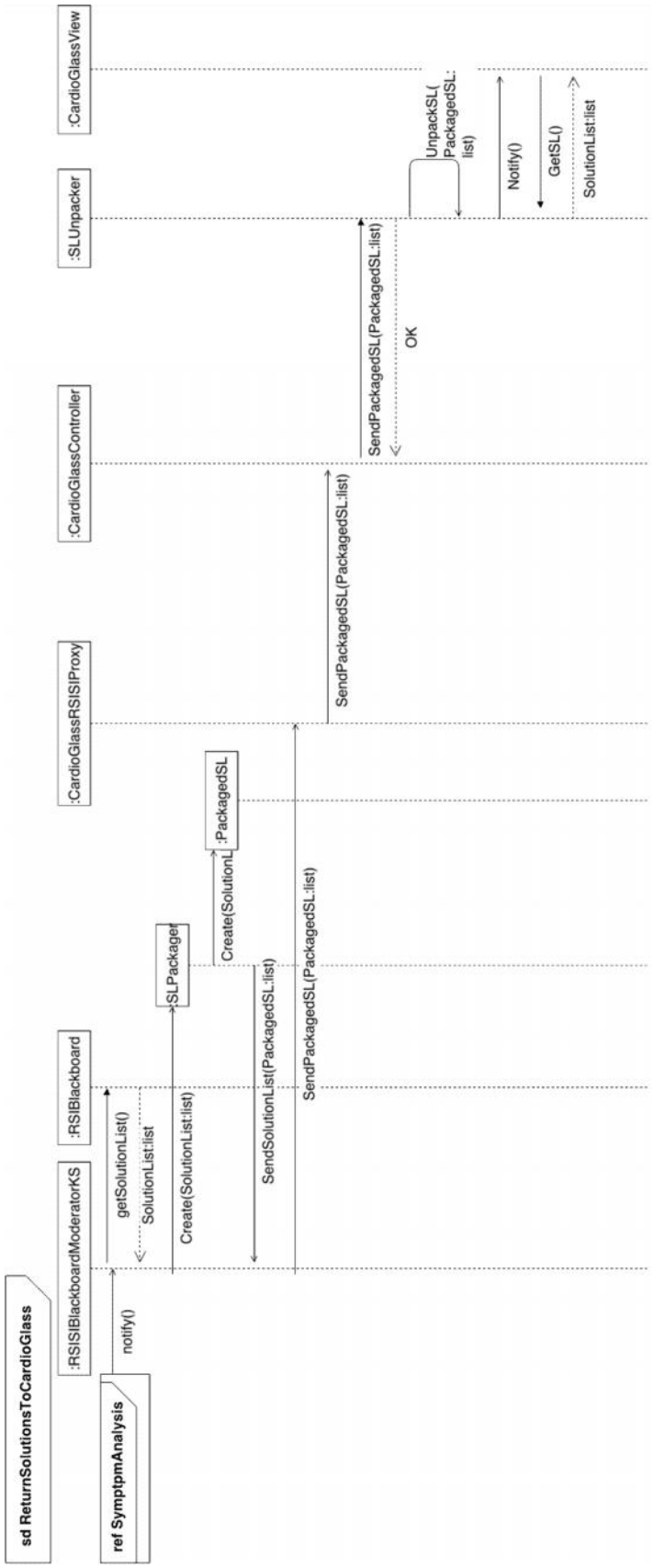
SLUnpacker	
Responsibilities	Collaborators
Unpack SolutionList	SolutionList, PackagedSL
Recieve PacakgedSL	PackagedSL, CardioGlassRSISIController
Implements Observable	CardioGlassView, SolutionList

3. Sequence Diagram

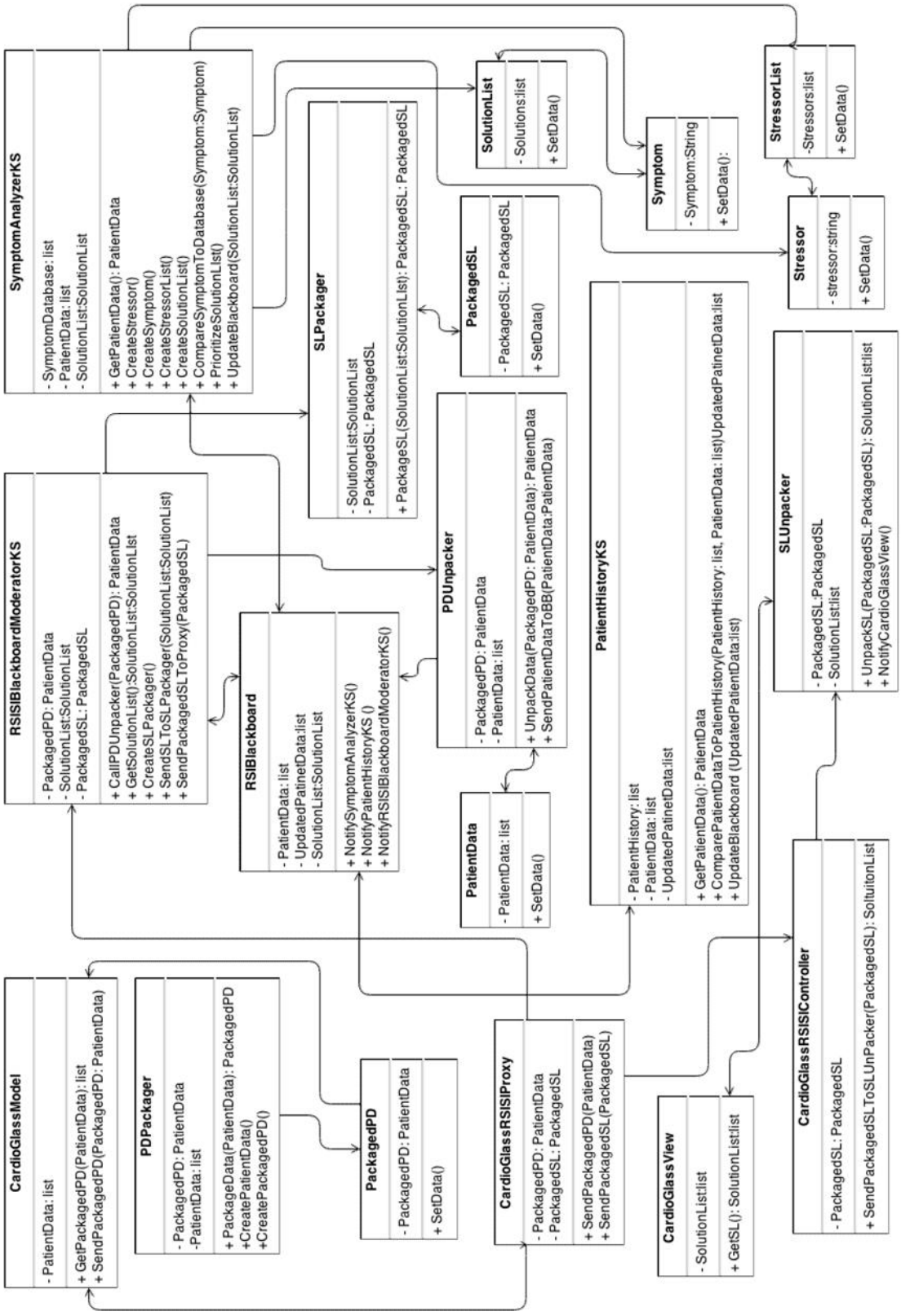



sd ProcessPatientData
ref GetGardioGlassHealthSuggestions
unpackData(PackagedPD:list)
:PDUnpacker
create(PackagedPD:list)
:PatientData
send(PatientData:PatientData)
:RSIBlackboard
notify()
notify()
getPatientData()
:PatientData
updatePatienData
(PatientData:list,
PatientHistory:list)
:SymptomAnalyzerKS
sendUpdatedPatientData(updatedPateinitData:list)
notify()
getPatientData()
:PatientData
:PatientHistoryKS
Create(PatientData)
ref SymtomAnalysis


sd SymptomAnalysis
ref ProcessPatientData
AnalyseSymptoms(PatientData:list)
:SymptomAnalyzerKS
loop (i = fist, last)
Create()
OK
:Symptom
loop (i = fist, last)
CompareToStressorData()
OK
:Stressor
Create()
OK
:StressorList
Create()
CompareTo()
ComparesOrNot
:SolutionList
AddTo(Stressor:list)
Create(StressorList:list)
OK
Prioritize(PatientData:list)
OK
update(SolutionList:list)
:RSIBlackboard
Notify()
ref ReturnSolutionsToCardioGlass


sd ReturnSolutionsToCardioGlass
ref SymptpmAnalysis
:RSISIBlackboardModeratorKS
notify()
:RSIBlackboard
getSolutionList()
SolutionList:list
:SLPackager
Create(SolutionList:list)
:PackagedSL
Create(SolutionL
:CardioGlassRSISIProxy
SendSolutionList(PackagedSL:list)
:CardioGlassController
SendPackagedSL(PackagedSL:list)
SendPackagedSL(PackagedSL:list)
OK
:SLUnpacker
UnpackSL(
PackagedSL:
list)
Notify()
GetSL()
SolutionList:list
:CardioGlassView

4. Design Class Diagram



5. Glossary

Abstraction Name	Definition
CardioGlassModel	The Model architectural component in CardioGlass; it holds all of the data for and about the Patient and their current situation
CardioGlassRSISIController	The controller architectural component in CardioGlass
CardioGlassRSISIProxy	The interface between CardioGlass and RSISI
CardioGlassView	The view for CardioGlass displayed to the Patient
PackagedPD	The instance that is created by PDPackager
PackagedSL	The instance of packaged data created by SLPackager
PatientData	Object that is created from the unpackaging of PackagedPD
PatientHistoryKS	The knowledge source of RSIBlackboard that contribute to the analysis of PatientData by offering information about the patient medical history
PDPackager	Class that packages the patient's current health information into one object
PDUnpacker	Class that handles the unpackaging of PackagedPD
RSIBlackboard	The Blackboard architectural component in RSISI
RSISIBlackboardModeratorKS	The controller knowledge source of RSIBlackboard
SLPackager	The component in RSISIBlackboardModeratorKS that retrieves and handles the packaging of SolutionList from RSIBlackboard
SLUnpacker	The component in CardioGlassModel that handles the unpackaging of PackagedSL
SolutionList	A list built by the SymptomAnalyzerKS that contains a list of Solutions to the patient's issues
Stressor	An event that is causing the patient to experience symptoms
StressorList	A list of possible stressors that could be the cause of the problems the patient is experiencing
Symptom	A piece of data that is found based on adverse health conditions the patient is experiencing
SymptomAnalyzerKS	The knowledge source of RSIBlackboard that analyzes patient data to find symptoms, possible stressors, and possible solutions to bring the patient's stress levels back down to healthy levels

6. Team Member Contributions

Team Member Name	List specific contributions to this assignment
Nathan	Brainstorming, Contributed to the first draft of the Design Scenario and Glossary, Drafted the System Sequence Diagram and the Design Class Diagrams, proofreading and editing
Nezar	Brainstorming, contributed to design scenario, glossary, and CRC cards. Significant design scenario expansion and formatting, proofreading and editing
Brandon	Brainstorming, Drafted Design Scenario, glossary, and CRC Cards, document formatting, proofreading and editing