Product Requirements

Team: *StackOverflowGooglers*

Revision Number	*Revision Date*	*Summary of Changes*	*Author(s)*
0.1	*9/5/2015*	*Initial statement of requirements*	*Doug Gawne*
0.2	*10/4/2015*	*Updated Use Case Diagrams as well as completed all use cases present in the R1 release of project. Also updated some of the stakeholders in the project and added a new user story.*	*Doug Gawne*

1 Brief problem statement

We represent a funding group (HAccelerator) chartered to create applications for the benefit of health-care across the country. The project we currently want to make a reality will be called **HealthNet**. At its core, HealthNet is meant to enable their hospitals in the US to be able to manage both employees and patients. The successful implementation should make it easy for users to effortlessly sign-up as patients so that the hospital can, without difficulty, manage their procedures and patient related tasks to optimize day-to-day work-flow.

The HealthNet product is intended to improve hospitals by providing an easy mechanism for managing employees, gathering statistical data on the inner workings of the hospital, signing up patients, making appointments, and allowing ease of transfer of both patients and their information between hospitals.

We want a product whose emphasis is on ease of use, whose navigation is straightforward and where the status of any, and all, information shown is clearly displayed. Ultimately, a system where understanding and communication about hospital and patient matters is improved.

The goal of this software is to provide a medical company with an easy way to keep track of patients and doctors throughout the hospital. It is to be easy to use and have the inbuilt features the hospital needs to function day to day.

2 Stakeholders

HAccelerator Board of Directors – oversee the projects funding and expenses. Have vested interest in the proven success of the product but are not involved in the planning and execution.
HAccelerator Product Owner – will act as principle representative for HealthNet product needs. He/she champions the product with the Board of Directors, helps facilitate product decisions and has the ultimate say on when and what features should be released.
StackOverflowGooglers – responsible for the design and development of the product.
Beta Testing Team – represent the target user base for HealthNet. Will be available in later phases of the project to conduct acceptance testing and provide feedback on product release.
Doctors and Nurses - will be using software on a daily basis to keep track of hospital activities including appointments patient medical information and other key facets of goings on in the hospital.
Patients - will be using software to stay more connected with the hospital staff.
Hospital Network Admins - will be using the software to keep track of statistical data as it applies to the hospital.

3 Users profile

The target user must:

- Have basic experience using computers and browsing the internet. Has filled out online forms or surveys and may have purchased or sold a product.
- Have a computer with access to the internet
- Have an interest in improving their health by using an online way of interacting with their hospital
- Be willing to share information such as home address and contact information as well as more personal information such as medical history

4 System requirements

At a high-level this project will be source controlled in SVN, run on Django using python, sqlite and needs to be compatible with the latest browsers.

Although the application needs to be accessible through the internet, deployments and demonstrations for this phase of the project will take place within the RIT Software Engineering environment. To this end, you must understand and document the target platforms from the perspective of the client browser as well as that of the server. Make sure to capture versions or software dependencies, programming languages and hardware specifications that are available for your use and proceed only after you document and confirm these with the customer.

5 Feature requirements (user stories)

The following list of user stories is neither final nor comprehensive. You must consider it your responsibility to maintain its relevance, clarify any misunderstandings and keep it up-to-date. Any changes must be discussed with the Product Owner for approval.

No.	User Story Name	Description	Release
1	Patient Registration	Users sign up to become a Patient by providing their personal contact information, proof of insurance and unique login credentials. Additionally, a patient should provide the system with some basic medical profile information, a choice of preferred hospital and emergency contact information (linked to another patient if they are already in the system).	**R1**
2	Administrator Registration	Doctors, Nurses, and Administrators will be added to the system by other administrators. All information for creating these new accounts will be done through an administrator account.	**R1**

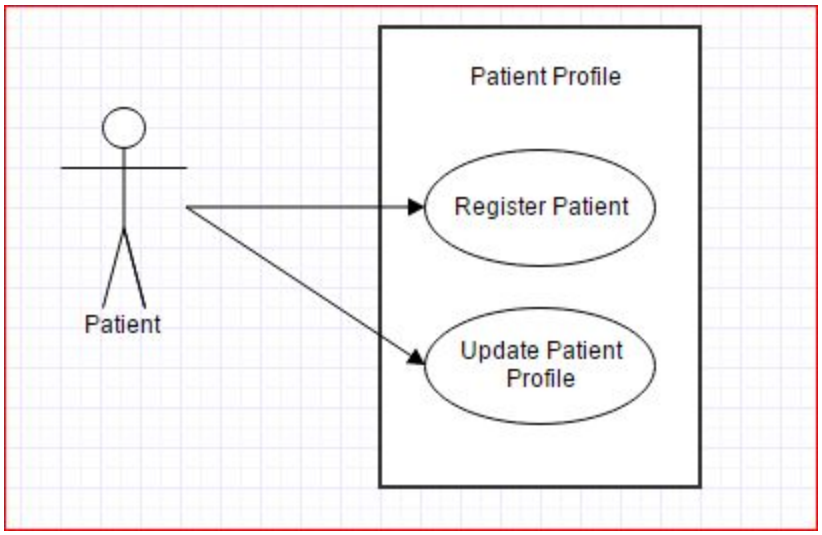
3	Update Patient Profile Information	Patients can update their profile information.	**R1**
4	Update Patient Medical Information	Doctors and Nurses can update patient medical information.	**R2**
5	Export Information	Patients will be able to export their information and their test results from the system with relevant privacy warnings.	**R2**
6	Create or Update Patient Appointment	Patients, doctors and nurses can create or update an appointment with a doctor and at one of the doctor's available locations. If the patient or doctor already has an appointment at the time selected, then the system will not allow for the appointment.	**R1**
7	Cancel Patient Appointment	Patients can cancel their existing appointments. Doctors can cancel their existing appointments. Nurses cannot cancel (only modify) existing appointments.	**R1**
8	Appointment Calendar	Doctors and patients will easily be able to view all of their appointments in a calendar view. Nurses will be able to see all appointments for the day and week between Patients and Doctors.	**R1**
9	Add/Remove Prescriptions	Doctors can add or remove a prescription to a patient record. Nurses can view the prescriptions of patients belonging to the same hospital. Patients can view their prescriptions from their account.	**R2**

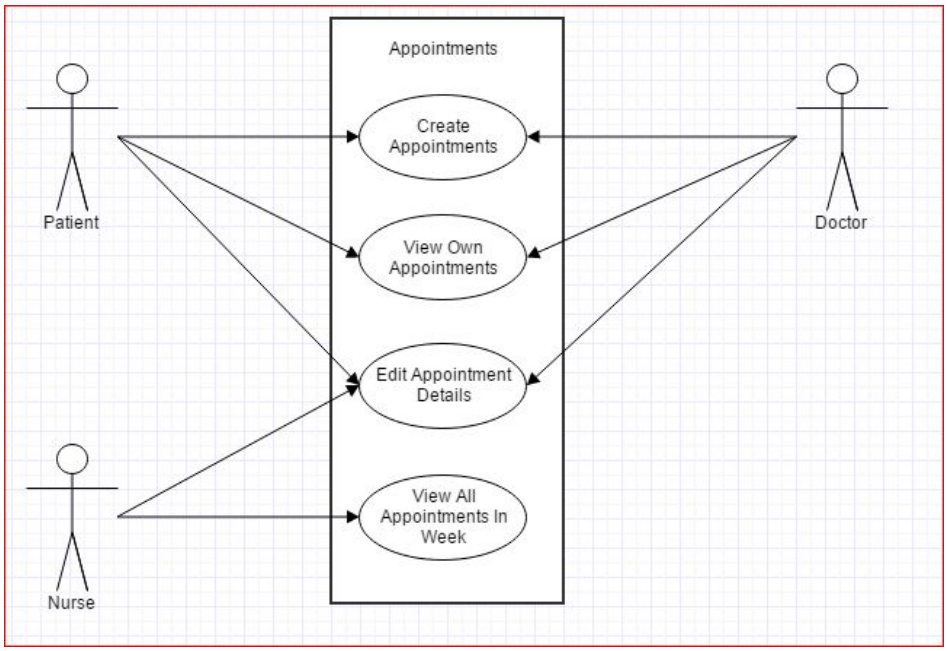
10	Viewing Patient Medical Information, Prescriptions and Tests and Results	Doctors can view all medical information for any patient in the system (regardless of Hospital). Nurses can only view patient medical information in the hospital they work for. Patients can view their tests (pending or completed) and view the corresponding results for those tests that have been released by the doctor. Prescriptions and other non-sensitive information is viewable by the patient without a need for doctor's release.	**R2**
11	Release Test Results	Doctors (within the patient's hospital) can, upon evaluating a patient's test results, release them for view by that patient. Comments may be added to the specific test result for view by the patient.	**R2**
12	Logging System Activity	For security, many actions in the system will be logged for review at a later date. Some examples of actions to be logged include but are not limited to updating of a Patient's information, viewing of a Patients information/records, and transfers of a Patient from one hospital to another.	**R1**
13	Admission and Discharge to/from Hospital	Doctors and Nurses can admit a patient to the hospital for an extended stay (reasons could be: emergency, observation, surgery, etc.). These are typically unexpected visits but can result from a decision made after a scheduled appointment. This event is recorded by the system. Doctors are the only ones to approve a patient's discharge from the Hospital. This event is recorded by the system.	**R2**

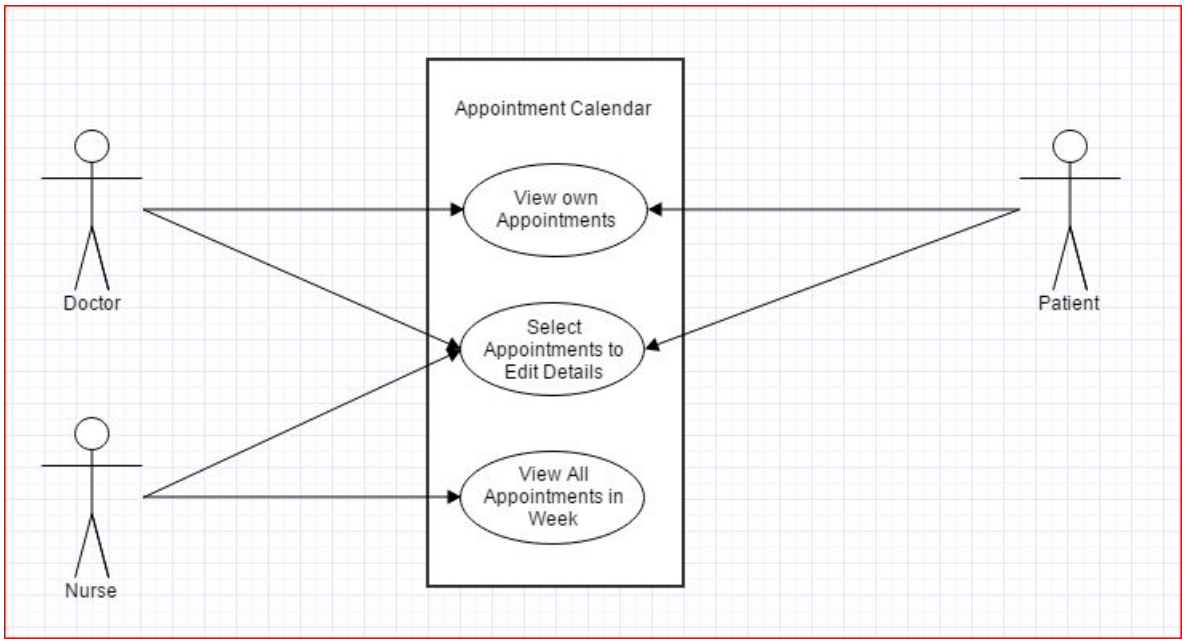
14	Viewing Activity Log	Administrators will be able to view the logs of all system activity for a given time-frame at their hospital. Some examples of this might be: - breakdown of the viewing activity of patient records or by system user - most common system activities (or by user) Other important and informative statistics yet to be determined.	**R1**
15	Viewing System Statistics	Administrators will be able to view compiled statistics for a given time-frame at their hospital. Some examples of this might be: - number of patients visiting the hospital - average number of visits per patient - average length of stay (from admission to discharge) - most common reasons for being admitted to the hospital - prescription statistics Other important and informative statistics yet to be determined.	**R2**
16	Patient Transfer	Patient can be transferred between hospitals. Transfers can be carried out by either administrators or by doctors (ones who are at the receiving hospital).	**R2**
17	Upload Patient Information	Doctors will be able to upload the results of a patient's tests if needed. Doctors will be able to upload images such as those used in X-Rays to update a patient's record. Uploads are considered as updates to a patient's medical information.	**R2**
18	Send Private Message	Doctors, nurses, patients and administrators can send private messages of limited length via the system.	**R2**

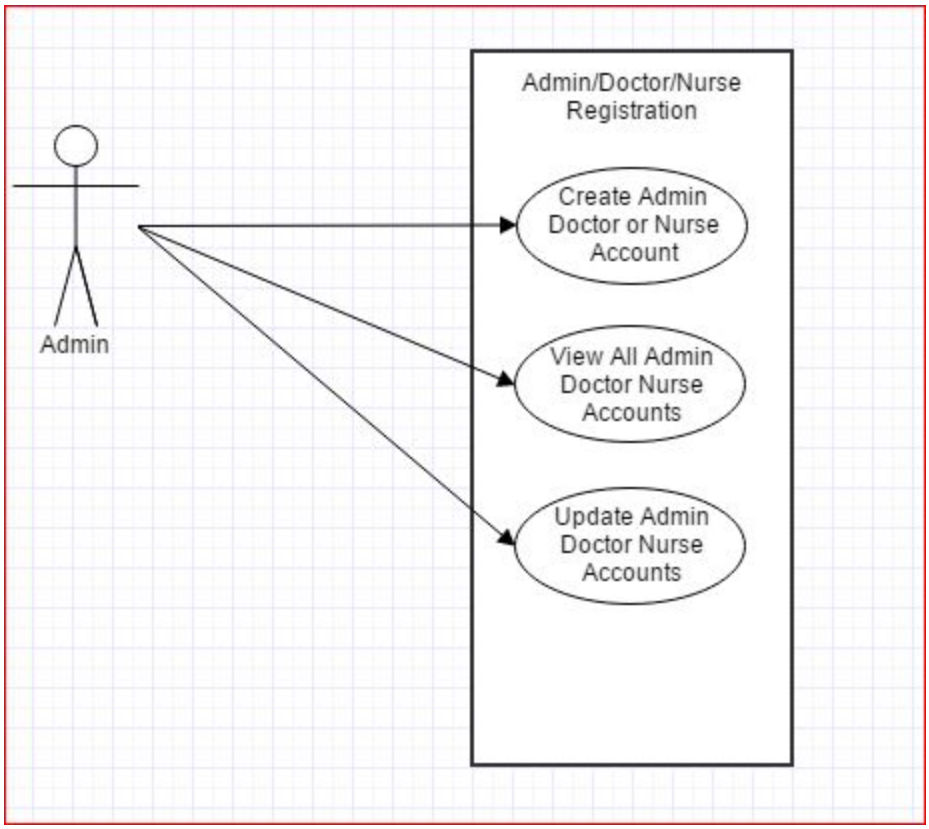
19	Administrator Message Board	Administrators will be able to post a message of the day that will be available for viewing by all doctors, nurses, and other administrators.	**R2**

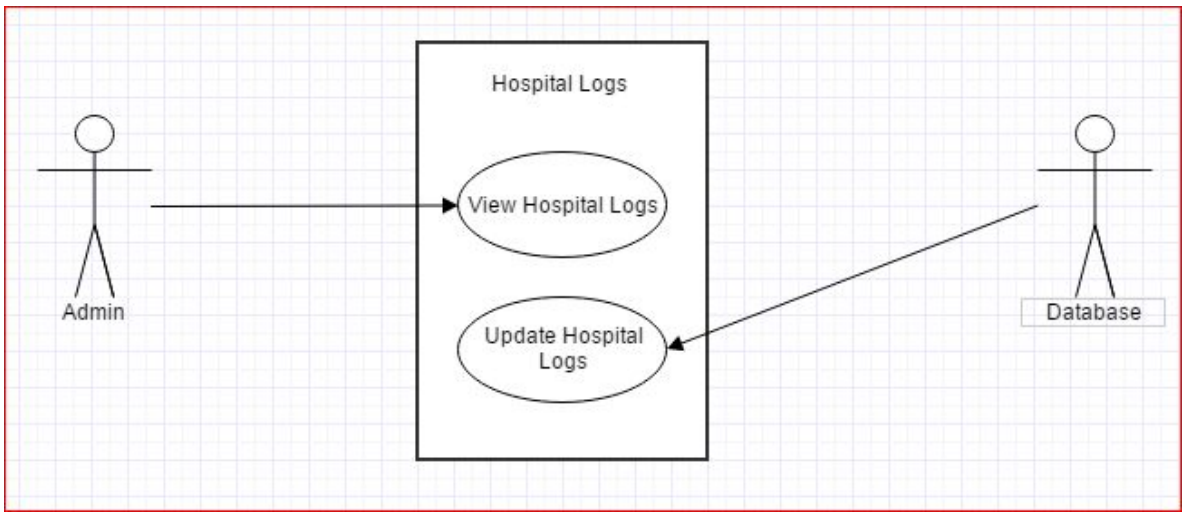
Use case diagram




Appointments
Create Appointments
View Own Appointments
Edit Appointment Details
View All Appointments In Week
Patient
Doctor
Nurse


Appointment Calendar
View own Appointments
Select Appointments to Edit Details
View All Appointments in Week
Doctor
Patient
Nurse


Admin/Doctor/Nurse Registration
Admin
Create Admin Doctor or Nurse Account
View All Admin Doctor Nurse Accounts
Update Admin Doctor Nurse Accounts


Hospital Logs
Admin
View Hospital Logs
Update Hospital Logs
Database

Use case description

Use Case Number:	***UC-01***
Use Case Name:	***Registration***
Overview:	***Registrant shall provide personal, medical, and emergency contact information to the System upon registering and becoming a Patient.***
Actor(s):	***Registrant***
Pre-condition(s):	***- System has been setup and configured.*** ***- System is running and open for registrations.*** ***- Registrant has accessed website via URL***
Scenario Flow:	***Main (success) Flow:*** ***1. Registrant selects option to register*** ***2. System requests personal information*** ***3. Registrant provided personal information.*** ***4. System verifies required information is provided.*** ● ***If information is invalid System displays message. Return to Step 2*** ***5. System requests basic medical information*** ***6. Registrant provides medical information***

	7. System verifies required information is provided. ● ***If information is invalid System displays message. Return to Step 5*** ***8. System requests emergency contact information*** ***9. Registrant provides emergency contact information*** ***10. System verifies required information is provided*** ○ ***If information is invalid System displays message. Return to Step 8*** ***11. System requests login information*** ***12. Registrant provides login information*** ***13. System verifies required information is provided*** ○ ***If information is invalid System displays message. Return to Step 11*** ***14. System displays confirmation of registration***
Alternate Flows:	***Alternate Flow #1: After Step 2 in success scenario System will display the option to Cancel the registration process. The following steps would occur:*** 1. ***Registrant selects option to cancel during registration*** 2. ***System requests confirmation to cancel***

	3. ***Registrant confirms intent*** 4. ***System returns to main screen*** ***Alternate Flow #2: The emergency contact information is an existing user in the system. After step 10 the following steps would occur:*** 1. ***Registrant selects option to select an emergency contact from the system*** 2. ***System displays a search bar for the Registrant to input the user's name*** 3. ***Registrant inputs the user's name and presses enter*** 4. ***System returns a list of users with matching names*** 5. ***Registrant chooses intended user*** 6. ***System sets that user as an emergency contact***
Post Condition:	***Registrant did not complete registration. System does not store Registrant's information.***

Use Case Number:	**UC-02**

Use Case Name:	**Administrator Registration**
Overview:	**Administrators with existing accounts will be able to add doctors, nurses, and new administrators to the system with administrative access.**
Actor(s):	**Prior administrator and a new administrator to be added**
Pre-condition(s):	***- System has been setup and configured*** ***- System has a prior administrative account created*** ***- New admin will be qualified to receive administrative privileges***
Scenario Flow:	**Main (success) Flow** 1. **Administrator logs into the system** 2. **Administrator selects option to add a new administrator** 3. **Administrator enters new account information** 4. **System verifies new administrator information** 5. **Status of doctor, nurse or administrator is requested** 6. **If nurse hospital of employment is requested** 7. **System requests login information** 8. **User enters login information**

	9. **System verifies registration**
Alternate Flows:	**Alternate Flow #1** 1. **User selects cancel** 2. **System requests verification of cancel** 3. **User confirms cancel** 4. **System returns to home screen**
Post Condition:	**Admin does not finish account creation and new account information is not stored**

Use Case Number:	**UC-03**
Use Case Name:	**Update Patient Profile Information**
Overview	**Patients will be able to update their personal information at any time once logged in to the system**

Actor(s):	**Patient**
Pre-Condition(s):	**-System is set up and configured** **-Patient has previously created an account** **-Patient is logged in to system**
Scenario Flow:	**Main (success) Flow** **1. From patient home page patient selects update information** **2. System requests personal, medical, and emergency information** **3. Once patient has changed information as desired patient selects to submit changes** **4. System verifies that information is valid** **5. System updates database of changes**
Alternate Flows:	**Alternate Flow #1** **1. Patient selects cancel** **2. System cancels the update of information and returns patient to home screen**
Post Condition:	**-Patient information is still in tact except for changed fields**

Use Case Number:	**UC-06**
Use Case Name:	**Create Patient Appointment**
Overview:	**Used for patients and doctors to create/update appointments in an available time slot.**
Actor(s):	**Doctor, patient**
Pre-condition(s):	**-System has been setup and configured** **-Both doctor and patient have an existing accounts** **-Time slot is free for both doctor and patient**
Scenario Flow:	**Main (success) Flow** 1. **User (can be doctor or patient) logs into the system** 2. **User selects option to create a new appointment** 3. **User selects a doctor or patient the appointment will be with** 4. **User selects time and date for appointment** 5. **System verifies that time slot selected is free for both doctor and patient**

	6. **Appointment is added to both the doctor and patient's account**
Alternate Flows:	**Alternate Flow #1** 1. **User selects cancel** 2. **System requests verification of cancel** 3. **User confirms cancel** 4. **System returns to home screen** **Alternate Flow #2** 1. **User selects a time slot that is already occupied** 2. **System shows a message saying the time slot is filled** 3. **System goes back to screen to select a new time and date**
Post Condition:	**User does not finish creating appointment and the appointment is not stored in either doctor or patient's account.**

Use Case Number:	**UC-07**

Use Case Name:	**Cancel/Update Patient Appointment**
Overview	**Patients and Doctors are able to cancel their appointments as desired**
Actor(s):	**Patient, Doctor**
Pre-Condition(s):	**-System has been setup and configured** **-User is logged into either a doctor or patient account** **-User has a previously created appointment**
Scenario Flow:	**Main (success) Flow** **1. User selects date of appointment from calendar on home page** **2. User selects the appointment on that date they wish to update/cancel** **3. System takes user to view appointment details** **4. From this page user can either update the appointment details or select cancel appointment** **5. Once appointment is changed or cancel is selected system verifies to make sure the new appointment details are valid** **6. System either cancels or updates the appointment in the database**
Alternate Flows:	**Alternative Flow #1** **1. User selects cancel while updating appointment details** **2. System returns user to homepage calendar**

Post Condition(s) :	**-Appointment is either removed or updated in database**

Use Case Number:	**UC-08**
Use Case Name:	**Appointment Calendar**
Overview	**From post-login homepage doctors and patients are able to see all appointments they have made on a calendar page. Nurses can see all doctors and patients and select one to see all appointments for the current week.**
Actor(s):	**Doctor, Nurse, Patient**
Pre-Condition(s):	**-System is set up and configured** **-User is logged into a valid nurse, doctor, or patient account**
Scenario	**Main (success) Flow**

Flow:	**1. User navigates to a post-login home page** **2. System displays all appropriate details in calendar form**
Alternate Flows:	**None**
Post Condition:	**None**

Use Case Number:	**UC-12**
Use Case Name:	**Logging Activity**
Overview:	**System logs all activity within the system for later viewing by an admin**
Actor(s):	**Any**
Pre-condition(s):	**-System has been set up and configured** **-User account(s) exist**

Scenario Flow:	**Main (success) Flow** 1. **Activity is performed within the system** 2. **At end of activity the logger is called to keep track of the activity** 3. **Logger saves activity for later viewing**
Alternate Flows:	**Alternate Flow #1** 1. **Activity is canceled** 2. **Logger is not called and does not save action**
Post Condition:	**Activity is saved within the logger.**

Use Case Number:	**UC-14**
Use Case Name:	**Viewing Activity Log**
Overview	**Administrators of the system are able to see all activity that has happened within the system**

Actor(s):	**Administrators**
Pre-Condition(s):	**-System is set up and configured** **-Administrator is logged into a valid account that has administrator privileges** **-Some activity has occurred within the system**
Scenario Flow:	**Main (sucess) Flow** **1. Administrator selects the logs link from the administrator home page** **2. Logs of system activity are displayed**
Alternate Flows:	**None**
Post Condition:	**None**