Product Requirements

Team Suites

Revision Number	*Revision Date*	*Summary of Changes*	*Author(s)*
0.1	09/06/2016	Drafted the requirements document	Software Development Team
1.0	9/08/2016	Finalized the requirements document for initial submission	Software Development Team
1.1	10/3/2016	Edited the requirements document for submission with R1	Software Development Team
2.0	10/15/2016	Edited the requirements document for initial submission for R2	Software Development Team
2.1	11/05/2016	Edited the requirements document for the R2 Beta Submission. Also changed some formatting	Software Development Team
2.2	12/3/16	Edited the requirements document for the R2 Submission.	Software Development Team

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.

Stakeholders

- **HAccelerator Board of Directors:**
  - Oversee the project's 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 principal 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.

- **Software Engineering Team:**
  - Is responsible for the day-to-day operations and coordination of all aspects related to the software product's life-cycle. This include, among others: planning and delegation of team roles and responsibilities; elicitation and clarification of requirements; analysis and design; implementation, testing and release of all software components.

- **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.

Users profile

- **Who will be using the system:**
  - Doctors
  - Nurses
  - Administrators
  - Patients

- **The Patient 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

- **The Doctors 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 the efficiency and usability of patient, doctor, and nurse interactions
  - Be proficient at uploading information such as test results

- **The Nurses 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 the efficiency and usability of patient, doctor, and nurse interactions

System requirements

- At a high-level this project will be source controlled in SVN, run on Django 1.9.1 using Python 3.4.3, sqlite 3 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.

Feature requirements (user stories)

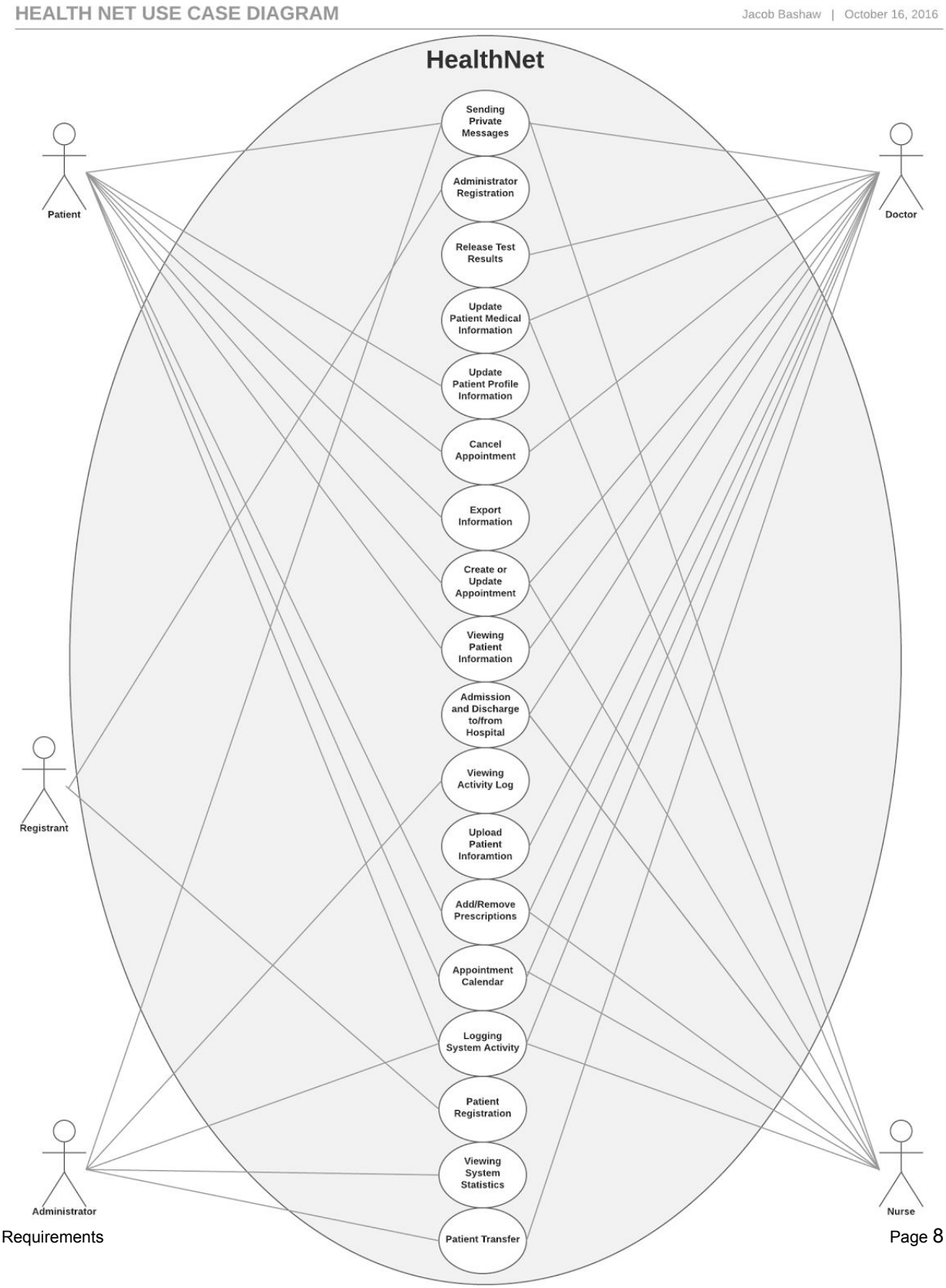
No.	User Story Name	Description	Release
1	Patient Registration	Users sign up to become a Patient by providing their personal contact information, proof of insurance and unique login credentials. Additionally, a patient should provide the system with some basic medical profile information, a choice of 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.	**R2 Beta**
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 Beta**

5	Export Information	Patients will be able to export their information and their test results from the system with relevant privacy warnings.	**R2 Beta**
6	Create or Update Patient Appointment	Patients, doctors and nurses can create or update an appointment with a doctor at the doctor's workplace. If the doctor already has an appointment at the time selected, then the system will not allow for the appointment.	**R1* Only patients** **R2 Beta**
7	Cancel Patient Appointment	Patients can cancel their existing appointments. Doctors can cancel their existing appointments. Nurses cannot cancel (only modify) existing appointments.	**R1* Only patients** **R2 Beta**
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 on month by month basis between Patients and Doctors .	**R1* Only patients** **R2 Beta**
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 Beta**
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.	**R2 Beta**

		Prescriptions and other non-sensitive information is viewable by the patient without a need for doctor's release.	
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 Beta**
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 Patient's 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	**R2 Beta**

		- 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.	
16	Patient Transfer	Patient can be transferred between hospitals. Transfers can be carried out 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 Beta**
18	Send Private Message	Doctors, nurses, patients and administrators can send private messages of limited length via the system.	**R2**

Use case context diagram


HEALTH NET USE CASE DIAGRAM
Jacob Bashaw | October 16, 2016
HealthNet
Sending Private Messages
Administrator Registration
Release Test Results
Update Patient Medical Information
Update Patient Profile Information
Cancel Appointment
Export Information
Create or Update Appointment
Viewing Patient Information
Admission and Discharge to/from Hospital
Viewing Activity Log
Upload Patient Inforamtion
Add/Remove Prescriptions
Appointment Calendar
Logging System Activity
Patient Registration
Viewing System Statistics
Patient Transfer
Patient
Doctor
Registrant
Administrator
Nurse

Use case description

Use Case Number:	***UC-01***
Use Case Name:	***Patient Registration***
Overview:	***Registrant shall provide personal, medical, and account information to the System to complete registration, upon registering they are redirected to their home page.***
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 redirects user to Registration page*** ***3. Registrant enters required information*** ***a. Required Information:*** ***i. Personal Information:*** ***1. First Name*** ***2. Last Name*** ***3. Address*** ***4. Phone Number*** ***5. Email address*** ***6. Insurance Provider*** ***7. Insurance ID*** ***ii. Medical Information:*** ***1. Height*** ***2. Weight*** ***3. Allergies*** ***4. Gender*** ***iii. Emergency Contact:*** ***1. First Name*** ***2. Last Name*** ***3. Address***

	4. Phone Number iv. Account Information: 1. Username 2. Password 3. Confirm Password 4. Registrant selects option to submit 5. System verifies that required information is provided (See 3A) a. If information is invalid or missing: i. Registrant is prompted to change invalid information (Step 3) 6. System displays confirmation of registration 7. Account is created and saved to the database 8. Registrant is redirected to Home page
Alternate Flows:	*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 process 2. System returns Registrant to login page
Post Condition:	*Failure: Registrant did not complete registration process. System did not store Registrant's information.* *Success: Registrant completed registration process. System stored account information and Registrant is now in the system. Registrant has been redirected to Home page*

Use Case Number:	*UC-02*
Use Case Name:	*Administrator Registration*
Overview:	*Administrator will be able to create new Doctors, Nurses, and Administrators*
Actor(s):	*Administrator*
Pre-condition(s):	*- System has been setup and configured.* *- System is running and open for registrations.* *- Administrator has accessed website via URL*

Scenario Flow:	***Main (success) Flow:*** ***1. Administrator selects option to register a new user (Doctor, Nurse, Administrator)*** ***2. System redirects Administrator to Staff Registration page*** ***3. Administrator enters required information*** ***a. Required Information:*** ***i. Personal Information:*** ***1. First Name*** ***2. Last Name*** ***3. User Type*** ***4. Workplace*** ***ii. Account Information:*** ***1. Username*** ***2. Password*** ***3. Confirm Password*** ***4. Administrator selects option to submit*** ***5. System verifies that required information is provided (See 3A)*** ***a. If information is invalid or missing:*** ***i. Administrator is prompted to change invalid information (Step 3)*** ***6. System displays confirmation of registration*** ***7. New account is created and saved to the database*** ***8. Administrator is redirected to Home page***
Alternate Flows:	***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. Administrator selects option to cancel during the registration process*** ***2. System returns Administrator to Home page***
Post Condition:	***Failure: Administrator did not complete registration process. System did not store the new user's information.***

	Success: Administrator completed registration process. System stored new user's account information and new user is now in the system. Administrator has been redirected to Home page

Use Case Number:	***UC-03***
Use Case Name:	***Update Patient Profile Information***
Overview:	***Patients are able to update their profile information.***
Actor(s):	***Patient***
Pre-condition(s):	***- System has been setup and configured.*** ***- System is running and open for modifications*** ***- Patient accessed website via login***
Scenario Flow:	***Main (success) Flow:*** ***1. Patient selects option to update profile information*** ***2. System redirects Patient to Update Profile page*** ***3. Patient edits their desired information*** ***a. Editable Information:*** ***i. Personal Information:*** ***1. First Name*** ***2. Last Name*** ***3. Address*** ***4. Phone Number*** ***5. Email address*** ***6. Insurance Provider*** ***7. Insurance ID*** ***ii. Emergency Contact:*** ***1. First Name*** ***2. Last Name*** ***3. Address*** ***4. Phone Number*** ***4. Patient selects option to submit*** ***5. System verifies that required information is still provided (See 3A)*** ***a. If information is invalid or missing:***

	i. *Patient is prompted to change invalid information (Step 3)* 6. *Profile information is changed and saved to the database* 7. *Patient is redirected to Information page*
Alternate Flows:	***Alternate Flows:*** ***Alternate Flow #1: After Step 2 in success scenario System will display the option to Cancel the update process. The following steps would occur:*** 1. *Patient selects option to cancel the update process* 2. *System returns Patient to Information page*
Post Condition:	***Failure: Patient did not update their profile. System did not store Patient's updated profile information.*** ***Success: Patient completed profile update. System saved the changed information, and Patient has been redirected to Information page***

Use Case Number:	***UC-04***
Use Case Name:	***Update Patient Medical Information***
Overview:	***Doctors and Nurses are able to update patient medical information in the case of any changes***
Actor(s):	***Medical Professional (Doctor, Nurse)***
Pre-condition(s):	***- System has been setup and configured.*** ***- Medical Professional has an account in the System*** ***- System is running and open for modifications*** ***- Medical Professional accessed website via login***
Scenario Flow:	***Main (success) Flow:*** 1. *Medical professional navigates to Information page* 2. *System displays Patients that the Medical Professional has access to* 3. *Medical Professional selects a Patient's name to update their medical information* 4. *System displays selected Patient's medical information* 5. *Medical Professional edits the desired information* a. *Editable Information:* i. *Height*

	ii. *Weight* iii. *Allergies* iv. *Gender* 6. *Medical professional selects option to submit* 7. *System verifies that required information is still provided (See 5A)* a. *If information is invalid:* i. *Medical Professional is prompted to change invalid information (Step 5)* 8. *System displays confirmation of medical information update* 9. *Changes are saved to the database* 10. *Medical Professional is redirected to Information page*
Alternate Flows:	***Alternate Flows:*** ***Alternate Flow #1: After Step 4 in success scenario System will display the option to Cancel the update process. The following steps would occur:*** 1. *Medical Professional selects option to cancel during the update process* 2. *System returns Medical Professional to Information page*
Post Condition:	***Failure: Medical Professional did not finish updating the patient's medical information. System did not save changes.*** ***Success: Medical information is changed and saved in the database. Medical Professional has been redirected to Information page***

Use Case Number:	***UC-05***
Use Case Name:	***Export Information***
Overview:	***Patients can export their information and test results from the system with privacy warnings.***
Actor(s):	***Patient***
Pre-condition(s):	***- System has been setup and configured*** ***- System is running*** ***- Patient accessed the website via login***
Scenario Flow:	***Main (success) Flow:*** 1. *Patient navigates to Information Page*

	2. *Patient selects option to export information* 3. *System displays the necessary privacy warnings* 4. *Patient selects the option to continue* 5. *System compiles a package of everything and downloads the package onto the computer the Patient has accessed the website from*
Alternate Flows:	***Alternate Flows:*** ***Alternate Flow #1: After Step 2 in success scenario System will display the option to Cancel the export process. The following steps would occur:*** 1. *Patient selects option to cancel the export process* 2. *System returns Patient to Information page*
Post Condition:	***Failure: The package failed to download. Nothing is changed*** ***Success: The Patient's information is downloaded onto their current computer***

Use Case Number:	***UC-06***
Use Case Name:	***Create or Update Patient Appointment***
Overview:	***Patients, Doctors, and Nurses can all create or update an appointment with a Doctor at one of the locations they work.***
Actor(s):	***User (Patient, Doctor, Nurse)***
Pre-condition(s):	***- System has been setup and configured*** ***- System is running*** ***- User accessed the website via login***
Scenario Flow:	***Main (success) Flow:*** ***Main Flow:*** 1. *User navigates to Appointments page* 2. *System displays a list of the User's accessible appointments as well as an option to create a new appointment* a. *View for Patient:* i. *A list of appointments that belong to the patient (Cancellable, editable)* ii. *Option to create a new appointment*

	b. *View for Doctor:* i. *List of all appointments at their workplace (Non cancellable, editable)* ii. *List of all appointments that belong to them (Cancellable, editable)* c. *View for Nurse:* i. *List of all appointments at their workplace (Non cancellable, editable)* 3. *User selects option to create a new appointment* 4. *System redirects user to Create Appointment page* 5. *User inputs the required information* a. *Required information for Patients:* i. *Doctor* ii. *Month* iii. *Day* iv. *Year* v. *Time* b. *Required Information for Doctors:* i. *Patient* ii. *Month* iii. *Day* iv. *Year* v. *Time* c. *Required information for Nurses:* i. *Patient* ii. *Doctor* iii. *Month* iv. *Day* v. *Year* vi. *Time* 6. *User selects option to create appointment* 7. *System verifies information is entered correctly*

	a. *If information is invalid:* i. *User is prompted to change invalid information (Step 5)* 8. *System checks if the Doctor has that date and time free* a. *If the Doctor is not free at the selected date and time:* i. *User is prompted to pick a new date and time (Step 5)* 9. *System displays that the appointment has been created* 10. *Appointment is created and saved to the database* 11. *User is redirected to Appointments page*
Alternate Flows:	***Alternate Flows:*** ***Alternate Flow #1: User selected an appointment on Step 3 instead of selecting to create one.*** 1. *System displays all information for the selected appointment* 2. *User updates the required information* a. *Required information for Patients:* i. *Doctor* ii. *Month* iii. *Day* iv. *Year* v. *Time* b. *Required Information for Doctors:* i. *Patient* ii. *Month* iii. *Day* iv. *Year* v. *Time* c. *Required information for Nurses:* i. *Patient* ii. *Doctor* iii. *Month*

	iv. Day v. Year vi. Time 3. User selects option to update appointment 4. System verifies information is entered correctly a. If information is invalid: i. User is prompted to change invalid information (Step 2) 5. System checks if the Doctor has that date and time free a. If the Doctor is not free at the selected date and time: i. User is prompted to pick a new date and time (Step 2) 6. System displays that the appointment has been updated 7. Changes to appointment are saved to the database 8. User is redirected to Appointments page *Alternate Flow #2: After Step 3 in success scenario System will display the option to Cancel the appointment creation process. The following steps would occur:* 1. User selects option to cancel the appointment creation process 2. System redirects user to Appointments page
Post Condition:	*Failure: The appointment was not finished being created or updated and changes were not saved.* *Success: The appointment was created/updated and the changes were saved to the database. User has been redirected to Appointments Page.*

Use Case Number:	*UC-07*
Use Case Name:	*Cancel Patient Appointment*
Overview:	*Patients and Doctors can cancel their existing appointments.*
Actor(s):	*User (Patient, Doctor)*
Pre-condition(s):	*- System has been setup and configured* *- System is running* *- User accessed the website via login*

Scenario Flow:	***Main (success) Flow:*** ***1. User navigates to Manage Appointments page*** ***2. User selects the option to cancel an appointment*** ***3. System removes the appointment from the database***
Alternate Flows:	***Alternate Flows:*** ***There are no alternate flows.***
Post Condition:	***Failure: User didn't select the option to cancel the appointment. The appointment remains active.*** ***Success: User cancelled their appointment, and it has been removed from the database***

Use Case Number:	***UC-08***
Use Case Name:	***Appointment Calendar***
Overview:	***Doctors and patients will be able to view all of their appointments in a calendar, while nurses will be able to see all appointments between patients and doctors.***
Actor(s):	***User (Doctor, Patient, Nurse)***
Pre-condition(s):	***- System has been setup and configured*** ***- System is running*** ***- User accessed the website via login***
Scenario Flow:	***Main (success) Flow:*** ***1. User selects the calendar option from the website navigation*** ***2. System redirects user to the Calendar page showing all appoints that pertain to them in a month calendar format***
Alternate Flows:	***Alternate Flows: No alternate flows***
Post Condition:	***Failure: User didn't select the option to view the calendar. The calendar is not displayed*** ***Success: The calendar has been displayed for the user***

Use Case Number:	***UC-09***
Use Case Name:	***Add/Remove/View Prescriptions***
Overview:	***Doctors can create and update prescriptions. Nurses can view the prescriptions of patients. Patients can view their prescriptions.***

Actor(s):	***User (Doctor, Nurse, Patient)***
Pre-condition(s):	***- System has been setup and configured*** ***- System is running*** ***- User accessed the website via login***
Scenario Flow:	***Main (success) Flow:*** ***Main flow for Patient:*** ***1. Patient navigates to Prescriptions page*** ***2. System displays all of the Patient's prescriptions that have been assigned to them by a Doctor*** ***Main flow for Nurse:*** ***1. Nurse navigates to Prescriptions page*** ***2. System displays all of the prescriptions created at the hospital the Nurse works at*** ***Main flow for Doctor:*** ***1. Doctor navigates to Prescriptions page*** ***2. System displays all prescriptions made by them*** ***3. Doctor selects option to create a new prescription*** ***4. System redirects Doctor to Create Prescription page*** ***5. Doctor enters required information*** ***a. Required Information:*** ***i. Name*** ***ii. Dosage*** ***iii. Patient*** ***6. Doctor selects option to submit*** ***7. System verifies that required information is provided (See 5A):*** ***a. If information is invalid or missing:*** ***i. Doctor is prompted to change invalid information (Step 5)*** ***8. System displays a confirmation message*** ***9. Prescription information is saved to the database*** ***10. Doctor is redirected to Prescriptions page***
Alternate Flows:	***Alternate Flows:***

	Alternate Flow #1 for Doctor: Doctor clicks on a prescription for a patient after Step 2 in Main Flow ***1. All information for the selected prescription is brought up*** ***2. System prompts for new prescription information*** ***a. Required Information:*** ***i. See Step #5A of main flow*** ***3. Doctor provides new information*** ***4. Doctor selects option to submit*** ***5. System verifies information was entered correctly*** ***a. If information is invalid:*** ***i. Doctor is prompted to re enter information (Step 3)*** ***6. System displays a confirmation message*** ***7. Doctor is redirected to Prescriptions page*** ***Alternate Flow #2 for Doctor: After Step 2 in Main flow, the Doctor selects the option to remove a prescription*** ***1. Doctor selects the option to remove prescription*** ***2. System removes the prescription from the database*** ***Alternate Flow #3 for Doctor: After Step 1 in Alternate Flow #1, and Step 3 in Main flow, System will display the option to Cancel the prescription creation process. The following steps would occur:*** ***1. Doctor selects option to cancel during prescription process*** ***2. System returns Doctor to Prescriptions page***
Post Condition:	***Failure: Prescription was not created, edited, or removed. Nothing was changed in the database*** ***Success: Prescription was created edited, or removed. Changes were saved in the database. User has been redirected to Prescriptions page.***

Use Case Number:	***UC-10***
Use Case Name:	***Viewing Patient Medical Information, Tests and Results***

Overview:	***Allows for the various user types (Doctor, Nurse, Patient), to view their respective information. Some user roles can view information that others can't.***
Actor(s):	***User (Doctor, Patient, Nurse)***
Pre-condition(s):	***- System has been setup and configured.*** ***- System is running.*** ***- Patients exist in the database***
Scenario Flow:	***Main (success) Flow:*** ***1. User navigates to Information page*** ***2. System pulls up pertinent information based on what type of user is logged in.***
Alternate Flows:	***Alternate Flows:*** ***1. Patient:*** ***a. On the Information page, the Patient can:*** ***i. View/Update their profile information (UC #3)*** ***ii. View their Medical information*** ***iii. View their test results (unreleased or released)*** ***iv. Export information (UC# 5)*** ***2. Nurse:*** ***a. On the Information page, the Nurse can:*** ***i. View Patient medical information in the hospital they work for*** ***ii. Edit Patient medical information in the hospital they work for (UC# 4)*** ***iii. Admit Patients into their Hospital (UC# 13)*** ***3. Doctor:*** ***a. On the Information page, the Doctor can:*** ***i. View all medical information for any patient in the system regardless of Hospital*** ***ii. Edit medical information for any patient in the system regardless of Hospital (UC# 4)*** ***iii. Release test results for Patients in their Hospital (UC #11)***

	iv. Add new tests for Patients in their Hospital (UC #11) *v. Admit a Patient into their Hospital (UC# 13)* *vi. Discharge a Patient from their Hospital (UC# 13)* *vii. Transfer a Patient into their Hospital (UC# 16)*
Post Condition:	***Nothing is changed, the system merely displays information***

Use Case Number:	***UC-11***
Use Case Name:	***Release Test Results***
Overview:	***Doctors (within the patient's hospital) can, upon evaluating a patient's test results, release them for view by that patient.***
Actor(s):	***Doctor***
Pre-condition(s):	***- System has been setup and configured*** ***- System is running*** ***- Doctor has accessed website via Login*** ***- Patient exists in database*** ***- Patient has undergone tests, appointments***
Scenario Flow:	***Main (success) Flow:*** ***1. Doctor navigates to Information page*** ***2. Doctor selects the Tests Results option for the intended patient*** ***3. System redirects Doctor to Tests page*** ***4. Doctor selects the option to release results for existing tests*** ***5. The test is changed from unreleased to released, and the results are now viewable by the associated Patient***
Alternate Flows:	***Alternate Flow #1: After Step 3 in the Main Flow, the Doctor selects the option to create a new test. The following steps would occur:*** ***1. Doctor enters required information*** ***a. Required Information:***

	i. *Test Name* ii. *Patient comments* iii. *File(s) associated with the test* 2. *Doctor selects option to submit* 3. *System verifies that required information is provided (See 1A):* a. *If information is invalid or missing:* i. *Doctor is prompted to change invalid information (Step 1)* 4. *System displays a confirmation message* 5. *Test information is created and saved to the database* 6. *Doctor is redirected to Tests page* ***Alternate Flow #2: After initializing Alternate Flow #1, the System will display the option to Cancel the test creation process. The following steps would occur:*** 1. *Doctor selects option to cancel during test creation process* 2. *System returns Doctor to Tests page*
Post Condition:	***Failure: No test results are released for a Patient to view*** ***Success: Results for the intended test are released, and the Doctor has been redirected to the Information page***

Use Case Number:	***UC-12***
Use Case Name:	***Logging System Activity***
Overview:	***For security, many actions in the system will be logged for review at a later date.***
Actor(s):	***User (Doctor, Nurse, Patient, Admin)***
Pre-condition(s):	***- System has been setup and configured.*** ***- System is running.*** ***- User has accessed website via Login***
Scenario Flow:	***Main (success) Flow:***

	1. *User performs any HealthNet action that updates or alters the database* 2. *System logs the action that was performed along with a timestamp*
Alternate Flows:	***Alternate Flows: None***
Post Condition:	***Failure: An action isn't performed and the action was never logged.*** ***Success: An action is performed and the action has been logged***

Use Case Number:	***UC-13***
Use Case Name:	***Admission and Discharge to/from Hospital***
Overview:	***Doctors and Nurses can admit a patient to the hospital for an extended stay. These are typically unexpected visits but can result from a decision made after a scheduled appointment. This event is recorded by the system.***
Actor(s):	***User (Doctor, Nurse)***
Pre-condition(s):	***- System has been setup and configured.*** ***- System is running.*** ***- User has accessed website via Login*** ***- A patient exists that needs to be admitted or discharged***
Scenario Flow:	***Main (success) Flow:*** ***Admittance Main Flow:*** 1. *Doctor navigates to User Information page* 2. *In the table titled "Non Admitted Patients", a Doctor can select the option to Admit a Patient if the Patient has no current Hospital associated with them* 3. *Changes are saved to the database* ***Discharge Main Flow:*** 1. *Doctor navigates to Information page* 2. *In the table titled "Patient Information", a Doctor can select the option to discharge a Patient in their Hospital* 3. *Changes are saved to the database*
Alternate Flows:	***Alternate Flows: None***

Post Condition:	***Failure: Patient is not released/admitted to/from hospital. The database is altered to accommodate for this change.*** ***Success: Patient is released/admitted to/from hospital. The database is altered to accommodate for this change.***

Use Case Number:	***UC-14***
Use Case Name:	***Viewing Activity Log***
Overview:	***Administrators will be able to view the logs of all system activity for a given time-frame at their hospital.***
Actor(s):	***Admin***
Pre-condition(s):	***- System has been setup and configured.*** ***- System is running.*** ***- Admin has accessed website via Login***
Scenario Flow:	***Main (success) Flow:*** ***1. Admin selects option to view the Activity log from Home page*** ***2. System loads Activity Log page*** ***a. Contents of Activity Page Log:*** ***i. Every action performed in HealthNet*** ***1. Registering*** ***2. Logging In*** ***3. Logging out*** ***4. Updating information (Profile and Medical)*** ***5. Creating/cancelling/updating appointments*** ***6. Exporting information*** ***7. Adding/removing prescriptions*** ***8. Releasing test results*** ***9. Admitting/Discharging a patient*** ***10. Transferring a patient*** ***11. Sending messages***

	12. Uploading patient information *ii. The user who performed the action* *iii. A timestamp when the action was performed*
Alternate Flows:	***Alternate Flows: None***
Post Condition:	***Failure: Nothing is done*** ***Success: The activity log is displayed for the Admin to view***

Use Case Number:	***UC-15***
Use Case Name:	***Viewing System Statistics***
Overview:	***Administrators will be able to view compiled statistics for a given time-frame at their hospital.***
Actor(s):	***Admin***
Pre-condition(s):	***- System has been setup and configured.*** ***- System is running.*** ***- Admin has accessed website via Login***
Scenario Flow:	***Main (success) Flow:*** ***1. Admin navigates to the Statistics page*** ***2. System loads Statistics page*** ***a. Contents of Statistics Page:*** ***i. Number of Doctors*** ***ii. Number of Nurses*** ***iii. Number of Patients*** ***iv. Number of Prescriptions*** ***v. Number of Appointments***
Alternate Flows:	***Alternate Flows: None***
Post Condition:	***Failure: Nothing is done*** ***Success: The Statistics page is displayed for the Admin to view***

Use Case Number:	***UC-16***

Use Case Name:	***Patient Transfer***
Overview:	***Patients can be transferred between hospitals.*** ***Transfers can be carried out by Doctors (ones who are at the receiving hospital).***
Actor(s):	***User (Doctor)***
Pre-condition(s):	***- System has been setup and configured.*** ***- System is running.*** ***- Doctor has accessed website via Login*** ***- Patient exists in database.*** ***- Patient isn't already in intended hospital***
Scenario Flow:	***Main (success) Flow:*** ***1. Doctor navigates to Information page*** ***2. In the table titled "Non Admitted Patients", a Doctor can select the option to transfer a Patient if the Patient is in any hospital other than theirs*** ***3. Changes are saved to the database***
Alternate Flows:	***Alternate Flows: None***
Post Condition:	***Failure: The Patient is not transferred from their current hospital. No changes are made*** ***Success: The Patient is transferred to a new hospital. Changes are saved in the database***

Use Case Number:	***UC-17***
Use Case Name:	***Upload Patient Information***
Overview:	***Allows the doctor to upload patient information after tests or procedures are completed.***
Actor(s):	***Doctor***
Pre-condition(s):	***- System has been setup and configured.*** ***- System is running.*** ***- Doctor has accessed website via Login*** ***- Patient exists in database.***
Scenario Flow:	***Main (success) Flow:***

	In Step #1B in Alternate Flow #1 for UC #11 the Doctor has the option to upload a file containing Patient Information. In this form submission process, the following steps would occur: ***1. Doctor, selects option to upload a file for a test*** ***2. System pulls up a file browser for the user to search for their intended file*** ***3. Doctor selects their desired file*** ***4. Once the rest of the form is filled out properly (See UC #11), the Doctor selects the option to submit*** ***5. The selected file is then uploaded to the database, and change is made to the Patient's information***
Alternate Flows:	***Alternate Flow #1: After Step 1 in success scenario System will display the option to Cancel the upload process. The following steps would occur:*** ***1. Doctor selects option to cancel during the upload process*** ***2. System returns Doctor to Tests page***
Post Condition:	***Failure: Patient information has not been updated. Changes are discarded and System saves nothing*** ***Success: Patient information has been uploaded. Changes have been made and are saved to the database.***

Use Case Number:	***UC-18***
Use Case Name:	***Send Private Messages***
Overview:	***Every type of user (Doctor, Nurse, Patient, Admin) can send private messages to each other.***
Actor(s):	***User (Patient, Doctor, Nurse, Admin)***
Pre-condition(s):	***- System has been setup and configured.*** ***- System is running.*** ***- User has accessed website via Login***
Scenario Flow:	***Main (success) Flow:*** ***1. User navigates to Messages page***

	2. *User selects option to create new message* 3. *System redirects User to Message Creation page* 4. *User enters the required information* a. *Required Information:* i. *Subject* ii. *Message* iii. *Recipient* 5. *User selects option to submit* 6. *System verifies that required information is provided (See 4A)* a. *If information is invalid or missing:* i. *User is prompted to change invalid information (Step 4)* 7. *Message is created and sent to the intended user* 8. *User is redirected to Messages page*
Alternate Flows:	*Alternate Flow #1: After Step 3 in success scenario, the System will display the option to Cancel the message creation process. The following steps would occur:* 1. *User selects option to cancel the message creation process* 2. *System redirects user to Messages page* *Alternate Flow #2: After Step 1 in success scenario, the User can select the option to view a message sent to them. The following steps would occur:* 1. *User selects the option to "View" a received message* 2. *System redirects user to Message page* a. *On this page the user can:* b. *Go back to the main Messaging page* i. *Delete the message* ii. *Reply to the message* *Alternate Flow #3: After Step 1 in success scenario, the User can select the option to view a message that they have already sent. The following steps would occur:*

	1. ***User selects the option to "View" a sent message*** 2. ***System redirects user to Message page*** a. ***On this page the user can:*** iii. ***Go back to the main Messaging page*** iv. ***Delete the message*** ***Alternate Flow #4: After Step 1 in success scenario, the User can select the option to reply to a message that they have received:*** 1. ***User selects the option to "View" a received message*** 2. ***System redirects user to Message page*** 3. ***User selects the option to reply to the message*** 4. ***System redirects user to Reply page*** 5. ***User enters the required information*** a. ***Required Information:*** i. ***Subject*** ii. ***Message*** 6. ***User selects option to submit*** 7. ***System verifies that required information is provided (See 5A)*** a. ***If information is invalid or missing:*** i. ***User is prompted to change invalid information (Step 5)*** 8. ***Reply is created and sent to the intended user*** 9. ***User is redirected to Messages page*** ***Alternate Flow #5: After Step 1 in success scenario, the User can select the option to delete a message sent to them, or a message that they have sent. The following steps would occur:*** 1. ***User selects the option to "View" a received/sent message*** 2. ***System redirects user to Message page*** 3. ***User selects the option to delete the message*** 4. ***System removes the message from the database*** 5. ***User is redirected to the Messages page***

Post Condition:	***Failure: A private message is not sent sent/viewed/replied to/deleted*** ***Success: A private message is sent/viewed/replied to/deleted***