**Product Requirements**

**Team: 202155-11-SWEN-261-2-BouncingBovines**

**Ryan Morrissey, Earl Sharkey, Matthew Graham, Christian Vaughan**

|  |  |  |  |
| --- | --- | --- | --- |
| ***Revision Number*** | ***Revision Date*** | ***Summary of Changes*** | ***Author(s)*** |
| 0.1 | 02/05/2016 | Initial requirements | Bouncing Bovines |
| 0.2 | 03/05/2016 | R1 Revisions | Ryan Morrissey |
| 0.3 | 03/30/2016 | R2 Planning Revisions | All members |
| 0.4 | 04/18/2016 | R2 Beta Revisions | Ryan Morrissey |
| 1.0 | 05/09/2016 | R2 Revisions | Ryan Morrissey |

# **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 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.

**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.

**Medical Staff** - represent the doctors, nurses and technicians who’s job is to help those admitted to their hospital. Since this software will control what they will do during their work hours, it is of the utmost importance to them that it helps rather than hinders.

**Patient** - represent the largest shareholder group. They are comprised of anyone who wants to go to a hospital withing the HN infrastructure. They will be completely unfamiliar with the interface as a general rule. It is very important that Patients can use the website effectivly.

# **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

# **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.

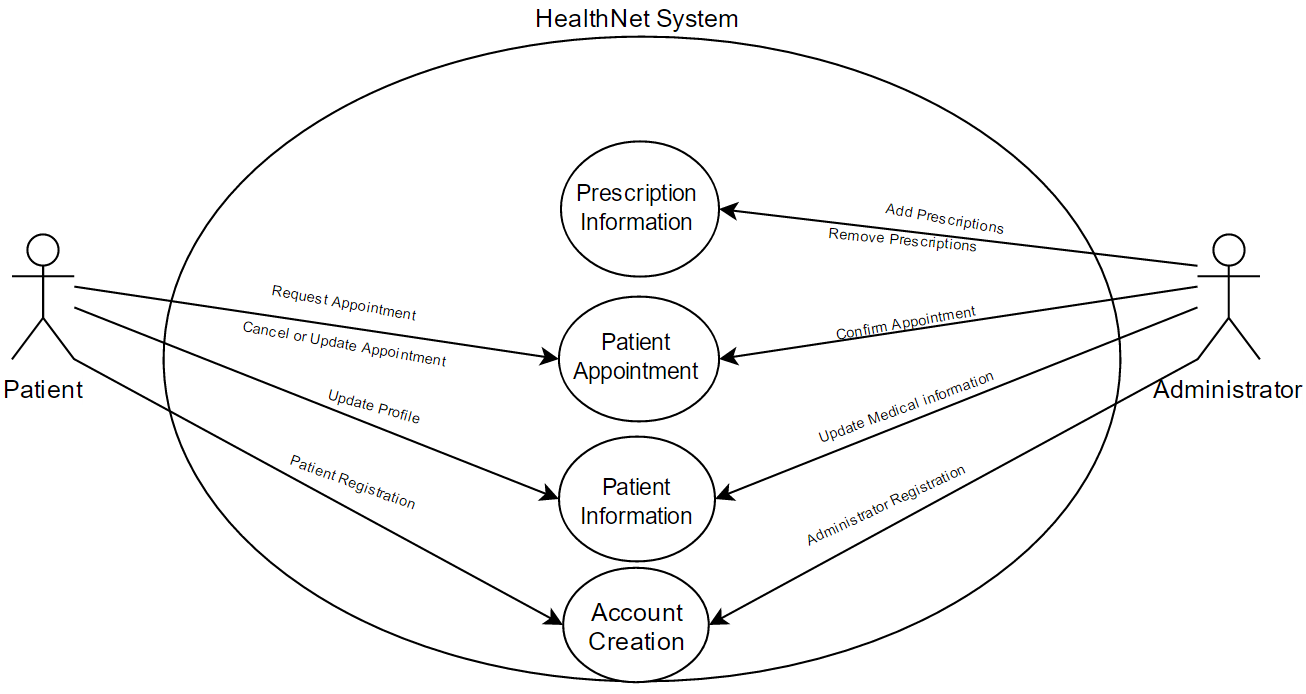
# **Feature requirements (user stories)**

The features will be finalized before we begin implementation. We need a little more time to get a good feel of the project. We would rather have a very strong base then come up with a million different features it *might* have.

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  (UC-01) | **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  (UC-04) | **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  (UC-02) | Patients can update their **profile** information. | **R1** |
| **4** | Update Patient Medical Information  (UC-03) | **Doctors and Nurses** can update patient medical information. | **R3** |
| **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  (UC-05) | **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  (UC-08) | Patients can cancel their existing **appointments**.  Doctors can cancel their existing appointments.  Nurses cannot cancel (only modify) existing appointments. | **R1** |
| **8** | Appointment Calendar (Part of UC-05) | Doctors and patients will easily be able to view all of their **appointment**s in a calendar view.  Nurses will be able to see all appointments for the day and week between Patients and Doctors. | **R3** |
| **9** | Add/Remove Prescriptions  (UC-06)  (UC-07) | 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. | **R1** |
| **10** | Viewing Patient Medical Information, Prescriptions and Tests and Results  (UC-09) | 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 (UC-13) | 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 (Automatic, not part of any single UC) | 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. | **R2** |
| **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. | **R3** |
| **14** | Viewing Activity Log  (UC-14) | 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. | **R2** |
| **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  (UC-15) | Patient can be transferred between hospitals.  Transfers can be carried out by either administrators or by doctors (ones who are at the receiving hospital). | **R3** |
| **17** | Upload Patient Medical Document  (UC-12) | 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. | **R3** |
| 18 | Send Private Message  (UC-16) | Doctors, nurses, patients and administrators can send private messages of limited length via the system. | **R1** |

**Use case context diagram**



**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 is set up and configured.** * **Actor has logged into a valid account** |
| **Scenario Flow:** | **Main (success) Flow:**   1. **Registrant selects option to register** 2. **System requests login information** 3. **Registrant provides login information** 4. **System verifies required information is provided**  * **If information is invalid System displays message. Return to Step 2**  1. **System requests personal, medical, and emergency contact information** 2. **Registrant provides required information.** 3. **System verifies required information is provided.**  * **If information is invalid System displays message. Return to Step 5**  1. **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:** | **Update Patient Profile Information** |
| **Overview:** | **Patients will be able to update their information such as their insurance provider, address, email, password, and dependents.** |
| **Actor(s):** | **Patient** |
| **Pre-condition(s):** | * **System is set up and configured.** * **Actor has logged into a valid account** |
| **Scenario Flow:** | **Main (success) Flow:**   1. **Actor selects option to update their profile information** 2. **Actor is prompted for password in order to update information**    1. **If password is incorrect, return to step 2** 3. **Actor can replace text already in fields [insurance provider information, address, email, password, phone number, and dependents]** 4. **Select Save option and Patient is redirected to the home page**    1. **If user attempts to redirect without saving, a prompt asks them if they are sure they want to leave before saving** 5. **Confirmation of save appears as notification** |
| **Alternate Flows:** | **Alternate Flow #1:**   1. **Actor tries to save information, but email is not in the form <label>@<email-service>.<service-extension>** 2. **No information is saved and a notification appears that says improper email format** 3. **Form is not saved until correct format is used**   **Alternative Flow #2**   1. **Actor tries to save information, but password uses invalid characters (,.;(){}[]#)** 2. **No information is saved and a notification appears that says invalid characters used in password** 3. **Form is not saved until correct format is used** |
| **Post Condition:** | **New user information is saved in the database.** |

|  |  |
| --- | --- |
| **Use Case Number:** | **UC-03** |
| **Use Case Name:** | **Update Patient Medical Information** |
| **Overview:** | **A doctor or nurse can edit a patent's medical information.** |
| **Actor(s):** | **Doctor, Nurse** |
| **Pre-condition(s):** | * **System is set up and configured.** * **Actor has logged into a valid account** * **Account that Actor is trying to edit exists** |
| **Scenario Flow:** | **Main (success) Flow:**   1. **Actor selects a patient account from available choices.** 2. **System provides access to chosen account’s medical info.** 3. **Actor makes changes to the files provided by system.** 4. **Actor requests to finish editing.** 5. **System prompts Actor to save changes.** 6. **Actor selects to save changes.** 7. **System saves changes that Actor has made and closes files.** 8. **System returns Actor to previous page.** |
| **Alternate Flows:** | **Alternate Flow #1: Actor decides not to save changes at step 6.**   1. **System discards changes made by Actor.** 2. **System closes files and returns Actor to previous page.** |
| **Post Condition:** | **The given patient’s health information has been updated and saved. Actor is returned to the previous page.** |

|  |  |
| --- | --- |
| **Use Case Number:** | **UC-04** |
| **Use Case Name:** | **Administrator Registration** |
| **Overview:** | **Administrator will provide name, job, salary, and other personal information for the person he/she is creating the account for.**  **The Admin will also provide the person with account credentials.** |
| **Actor(s):** | **Administrator** |
| **Pre-condition(s):** | **- System has been setup and configured.**  **- System is running and open for registrations.**  **- Admin has logged into admin account** |
| **Scenario Flow:** | **Main (success) Flow:**   1. **Actor selects option to register new employee** 2. **System requests employee information** 3. **Admin provided employee information.** 4. **System verifies required information is provided.**  * **If information is invalid System displays message. Return to Step 2**  1. **System requests employee emergency contact information** 2. **Registrant provides emergency contact information** 3. **System verifies required information is provided**    * **If information is invalid System displays message. Return to Step 8** 4. **System requests admin creates employee login information** 5. **Actor provides login information** 6. **System verifies required information is provided**    * **If information is invalid System displays message. Return to Step 9** 7. **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 login information is an existing user in the system. After step 10 the following steps would occur:**   1. **Admin selects option to create new login information** 2. **Admin enters new login information** 3. **System verifies required information is entered.**    * **If information is invalid return to step 2** 4. **System displays confirmation of registration** |

|  |  |
| --- | --- |
| **Use Case Number:** | **UC-05** |
| **Use Case Name:** | **Create or Update Patient Appointment** |
| **Overview:** | **User logged in as a patient shall put in a request to get a time and date for an appointment. The user may then select an appointment before its confirmed date and update the information of the appointment or cancel it completely.** |
| **Actor(s):** | **Patient, Nurse, Doctor** |
| **Pre-condition(s):** | **- System has been setup and configured.**  **- System is running and is accepting appointment requests.** |
| **Scenario Flow:** | **Main (success) Flow:**   1. **The actor selects option to make an appointment.** 2. **System requests information about the appointment.** 3. **The user provides information to the system about their appointment.** 4. **The system checks openings in the doctor’s and the patient’s scheduled appointments.** 5. **If either the doctor or patient is not free at that time, return to step 2 with an error message.** 6. **The system then confirms the appointment and updates the selected doctor’s and patient’s schedule with the new appointment.** |
| **Alternate Flows:** | **Alternate Flows #1: The user wishes to update information on an appointment request.**   1. **Actor selects “Modify or Cancel Appointment Request”.** 2. **System requests changes to the current appointment information.** 3. **User makes changes to the information in their request and provides this to the system.** 4. **They systems makes note of the changes and updates the appointment request.**   **Alternate Flow #2: The user wishes to cancel an appointment request.**   1. **Actor selects “Modify or Cancel Appointment Request”.** 2. **System requests changes to the current appointment request.** 3. **Actor selects “Cancel Appointment Request”.** 4. **System acknowledges the cancellation and deletes the request from the system.** |
| **Post Condition:** | **The patients requests/changes are saved to the systems database.** |

|  |  |
| --- | --- |
| **Use Case Number:** | **UC-06** |
| **Use Case Name:** | **Add Prescriptions** |
| **Overview:** | **Doctors can add a prescription to a Patient record.** |
| **Actor(s):** | **Doctor** |
| **Pre-condition(s):** | **- System has been setup and configured.**  **- System is running and is accepting prescription updates.** |
| **Scenario Flow:** | **Main (success) Flow:**   1. **Actor selects a write prescription option** 2. **Redirect to a form where the doctor must fill in patient name, prescription name, strength, quantity, refill count, pharmacy name, special instructions** 3. **Actor clicks save prescription**    1. **if invalid name is written, repeat step 3 and display an error message** 4. **Redirect to a confirmation screen where all information is formatted together** 5. **Actor confirms prescription.** |
| **Alternate Flows:** | **Alternate Flow #1:**   1. **Actor selects edit prescription on confirmation screen** 2. **Redirect to prescription form with all previous information inputted** 3. **Start again from step 4 in main flow**   **Alternate Flow #2**   1. **Actor selects cancel prescription** 2. **Confirmation of cancellation prompt appears on screen** 3. **Actor must select either no, do not cancel or yes, please cancel** 4. **Upon cancellation, redirect Actor to home screen**    1. **if the Actor decides not to cancel, redirect to step 5 of main flow** |
| **Post Condition:** | **Confirmation message appears on screen and Actor is redirected to their home screen.** |

|  |  |
| --- | --- |
| **Use Case Number:** | **UC-07** |
| **Use Case Name:** | **Remove Prescriptions** |
| **Overview:** | **Doctors can remove a prescription from a Patients record.** |
| **Actor(s):** | **Doctor** |
| **Pre-condition(s):** | **- System has been setup and configured.**  **- System is running and is accepting prescription updates.** |
| **Scenario Flow:** | **Main (success) Flow:**   1. **Actor selects view prescriptions option** 2. **Actor selects the remove option next to the prescription description** 3. **Prompts a confirmation of removal** 4. **Actor must select either confirm removal or cancel**    1. **cancel redirects to view prescriptions screen** |
| **Alternate Flows:** | **No Alternate Flows** |
| **Post Condition:** | **Prescription is removed from system and confirmation message appears on screen. Doctor is redirected to view prescription screen.** |

|  |  |
| --- | --- |
| **Use Case Number:** | **UC-08** |
| **Use Case Name:** | **Cancel Patient Appointment** |
| **Overview:** | **Patients can cancel their existing appointments. Doctors can cancel their existing appointments.** |
| **Actor(s):** | **Doctor, Patient** |
| **Pre-condition(s):** | * **System has been setup and configured.** * **System is running and is accepting appointment updates.** |
| **Scenario Flow:** | 1. **Actor clicks appointment schedule** 2. **From list of appointments, user clicks remove appointment option** 3. **Notifies user of removal via notification** |
| **Alternate Flows:** | **None** |
| **Post Condition:** | **Appointment is removed from the system and redirects to home page** |

|  |  |
| --- | --- |
| **Use Case Number:** | **UC-09** |
| **Use Case Name:** | **View Patient Medical Documents** |
| **Overview:** | **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 own medical information.** |
| **Actor(s):** | **Doctor, Nurse, Patient** |
| **Pre-condition(s):** | * **System has been setup and configured.** * **Actor is logged in.** |
| **Scenario Flow:** | 1. **Actor (Doctor) searches for selects the desired Patient’s profile** 2. **Selects view medical records option** 3. **Redirected to screen with Patients medical records in list form** |
| **Alternate Flows:** | **Alternate #1 (Nurse)**   1. **Searches for and selects desired Patient’s profile** 2. **Selects view medical records option** 3. **System verifies that Nurse works for the same hospital that the Patient belongs to**    1. **if verification fails, a notification will appear that says “you cannot view this medical information”**   **Alternate #1 (Patient)**   1. **Actor is currently on their own page.** 2. **Actor selects view medical records option.** |
| **Post Condition:** | **Redirectes to screen with Patient’s medical records in list form** |

|  |  |
| --- | --- |
| **Use Case Number:** | **UC-10** |
| **Use Case Name:** | **View Patient Prescriptions** |
| **Overview:** | **Doctors can view the prescriptions of anyone in the HealthNet database. Nurses can view the prescriptions of anyone in the HealthNet database that also belongs to the hospital the nurse works. Patients can view their own prescriptions.** |
| **Actor(s):** | **Doctors, Nurses, Patients** |
| **Pre-condition(s):** | * **System has been setup and configured.** * **User is logged into the database.** |
| **Scenario Flow:** | **Main (success) Flow:**   1. **Actor (Doctor) searches for and selects their Patient’s profile under the Patients screen,** 2. **Actor selects view prescriptions option,** |
| **Alternate Flows:** | **Alternate #1 (Nurse)**   1. **Actor searches for and selects their Patient’s profile under the Patients screen,** 2. **Actor selects view prescriptions option,**   **Alternate #2 (Patient)**   1. **Actor is currently on their own page.** 2. **Actor selects view prescriptions option,** |
| **Post Condition:** | **A screen displays the patients prescription information.** |

|  |  |
| --- | --- |
| **Use Case Number:** | **UC-11** |
| **Use Case Name:** | **Patient Transfer** |
| **Overview:** | **Patient can be transferred from one hospital to another.** |
| **Actor(s):** | **Administrator, Doctor** |
| **Pre-condition(s):** | * **System has been setup and configured.** * **User is logged into the database.** |
| **Scenario Flow:** | **Main (success) Flow:**   1. **Actor (Administrator) looks up a patient in the HealthNet database.** 2. **Actor selects the “transfer patient” option.** 3. **Actor selects the new hospital that the patient will be transferred to.** |
| **Alternate Flows:** | **Alternate (Doctor)**   1. **Actor looks up a patient in the HealthNet database.** 2. **Actor selects the “transfer patient” option.** 3. **System checks to make sure the patient is not currently a member of the doctors hospital.** |
| **Post Condition:** | **The patients information is updated to show that they now belong to another hospital.** |

|  |  |
| --- | --- |
| **Use Case Number:** | **UC-12** |
| **Use Case Name:** | **Upload Patient Medical Document** |
| **Overview:** | **Doctors are able to access the medical documents after tests have been run. The doctor may make comments and or changes as he sees fits the save the changes.** |
| **Actor(s):** | **Doctor** |
| **Pre-condition(s):** | * **System has been setup and configured.** * **Actor is logged into the database.** |
| **Scenario Flow:** | **Main Flow:**   1. **Actor has accesses correct patient profile.** 2. **Actor selects view medical documents** 3. **Actor enters document ID of document he wishes to access.**    1. **If ID does not exist repeat step 3 or cancel.** 4. **Actor makes comments or changes.** 5. **System prompts for save.** 6. **Actor saves and uploads changes to medical documents.** |
| **Alternate Flows:** | **None** |
| **Post Condition:** | **Medical document is added to patients profile.** |

|  |  |
| --- | --- |
| **Use Case Number:** | **UC-13** |
| **Use Case Name:** | **Release Test Results** |
| **Overview:** | **Doctors will release test results to the patient.** |
| **Actor(s):** | **Doctor** |
| **Pre-condition(s):** | * **System has been setup and configured.** * **Doctor is logged into the database.** |
| **Scenario Flow:** | **Main (success) Flow:**   1. **Actor selects medical document.** 2. **Actor selects release.** |
| **Alternate Flows:** | **None** |
| **Post Condition:** | **The Test results are added to the Patient’s medical documents library and are available for viewing.** |

|  |  |
| --- | --- |
| **Use Case Number:** | **UC-14** |
| **Use Case Name:** | **View Activity logs** |
| **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.** * **Actor is logged into the database.** |
| **Scenario Flow:** | **Main (success) Flow:**   1. **Actor navigates to activity log tab** 2. **Actor selects time frame from drop down menu** 3. **System displays chosen section of the log.** |
| **Alternate Flows:** | 1. **Actor selects to only see certain types of activities.** 2. **System only displaces activities of chosen types.** |
| **Post Condition:** | **Admin is given a page with the activity logs displayed** |

|  |  |
| --- | --- |
| **Use Case Number:** | **UC-15** |
| **Use Case Name:** | **Send Private Message** |
| **Overview:** | **Private messages of limited length can be sent between users in the system** |
| **Actor(s):** | **Doctor, Nurse, Patient, Admin** |
| **Pre-condition(s):** | * **System has been setup and configured.** * **System is set up to allow messages.** |
| **Scenario Flow:** | **Main (success) Flow:**   1. **Message tab is selected** 2. **Select create message option** 3. **Redirect to a form where they must enter recipient’s name, subject, and message** 4. **Click send message button** |
| **Alternate Flows:** | **Alt flow #1 (Patient sends message to Doctor)**   1. **On send, message is sent to a Nurse in the same department as the Doctor** 2. **Nurse can answer message and send or click forward to Doctor**    1. **forward to doctor will send message to Doctor’s messages**    2. **message is removed from Nurse’s messages** |
| **Post Condition:** | **Redirect to message tab and message should appear in the recipient’s messages tab** |

|  |  |
| --- | --- |
| **Use Case Number:** | **UC-16** |
| **Use Case Name:** | **View System Statistics** |
| **Overview:** | **Allows administrators to view statistics about hospital patients.** |
| **Actor(s):** | **Admin** |
| **Pre-condition(s):** | * **System has been setup and configured.** * **Actor is logged into the database.** |
| **Scenario Flow:** | **Main (success) Flow:**   1. **Statistics log is selected.** 2. **Select statistic type.**    1. **Patient Will show statistics about patients.**    2. **Hospital will show stats about the hospital in general. ie. Number of patients visiting and inpatient/outpatient.**    3. **Staff will show stats for all staff related to the hospital.**    4. **Return redirects to select another statistic type.**    5. **Reset, resets all stats on selected stat type.** 3. **Select exit to return to main page** |
| **Alternate Flows:** | **None** |
| **Post Condition:** | **Main admin panel is displayed** |

|  |  |
| --- | --- |
| **Use Case Number:** | **UC-17** |
| **Use Case Name:** | **Admission/Discharge from hospital** |
| **Overview:** | **Allows admin to discharge patients from the hospital** |
| **Actor(s):** | **Doctors, Nurses** |
| **Pre-condition(s):** | * **System has been setup and configured.** * **Actor is logged into the database.** |
| **Scenario Flow:** | **Main (success) Flow:**  **Flow #1 Admit a patient.**   1. **Actor chooses to admit a patient to the hospital for care.** 2. **Actor selects patient to admit.**    1. **Actor enters reason for admittance.** 3. **Actor selects Admit to finalize the admittance.**   **Flow #2 Discharge Patient**   1. **Actor (Doctor) selects a patient who is currently attending their hospital.** 2. **Actor selects to discharge a patient** 3. **A message is displayed which asks if they would like to discharge the patient.** 4. **After the Actor accepts, the patient’s information is updated to reflect that they are no longer staying at the hospital.** |
| **Alternate Flows:** | **Alternate Flow #1 Admit**   1. **Selected patient does not belong to hospital**    1. **patient must create account via HeathNET**   **Alternate Flow #2 Discharge**   1. **Patient not admitted to the hospital.** |
| **Post Condition:** | **The patients information now shows that they are currently staying at the hospital. If the alternative flow is competed, then it will update the patients information to show that they are no longer staying at the hospital.** |