

SENG2011 19T3



Interim Report

# Medics

Abanob Tawfik z5075490

Kevin Luxa z5074984

Lucas Pok z5122535

Michael Yoo z5165635

Rason Chia z5084566

# 1 Executive Summary

Project Vampire is an electronic records management system for the handling of medical blood inventories.

The project aims to replace paper-based records systems and provide typical benefits of digital records management – such as reducing costs, consistency and improving usability. However, concerns are often raised regarding the reliability of a digital system in managing a critical resource with human lives at stake. To provide assurance, our system uses formal verification techniques to ensure that it conforms to safety requirements.

Our system integrates with pre-existing business processes of blood inventory management. For example, there are built-in processes for the expiration of blood, distinction between blood types, requests and acceptance of donated blood, to list a few.

Our system comprises of a graphical interface that interacts with a verified datastore service. The implementation details will be detailed in their respective sections.

Our final project and report will describe a minimum viable product to demonstrate that safety and assurance requirements can be met with a digital records management system.

## 2 Requirements

### 2.1 Stakeholders

- **Vampire staff**
  - Handles the physical delivery of requested blood to hospitals
  - Physically disposes of expired blood by sending it to a blood disposal service
- **Hospital staff**
  - Can request blood to treat patients
  - Can query the level of different types of blood in Vampire's inventory
- **Batmobile admin** [Vampire employee]
  - Deposits blood into the system
- **Pathology staff**
  - Tests all blood within the system
- **Member of the public (potential donor)**
  - Finds out when, where they can donate blood, and if they can donate
- **Blood donor**
  - Donates blood to the Batmobile
  - Can see the outcome of their blood donation
- **Emergency donor**
  - Upon hitting a critically low supply of blood in the system, these people will donate blood via the Batmobile
  - Firefighters, police officers, volunteers, actively registered through the Vampire system
- **Vampire headquarters**
  - Business owners
  - Manage all critical and sensitive actions
- **Blood dump**
  - Will eliminate blood in a medically suitable fashion

## 2.2 Priority key

- **P1 (Priority 1, Must have)**
  - All these requirements form the minimum viable product, crucial to launch
- **P2 (Priority 2, Should have)**
  - All these requirements should be done in the time frame, but are not critical to launch, can be delayed for future release
- **P3 (Priority 3, Could have)**
  - All these requirements are features we would like, but in the time frame we may delay them to future releases to focus on priority 1 and priority 2
- **P4 (Priority 4, Would have)**
  - Features we do not expect to have but show the roadmap of where we see our product down the line

## 2.3 Requirements

### 1. Querying the system. [P1]

#### 1.1. Hospital staff can enquire about the levels of different types of blood in Vampire's inventory. [P1]

*Assumptions:*

- As in the real world, there are 8 blood types: A+, A−, B+, B−, O+, O−, AB+, and AB−.
- The disposal of expired blood occurs daily so queries will always return information about the amount of fresh blood.
- Whole blood can be divided into separate components (such as red blood cells, platelets, and plasma), and each component has a different lifespan. For simplicity, we assume that Vampire only needs to store whole blood.

##### 1.1.1. Hospital staff can see the total amount of each blood type in the inventory. [P1]

**1.1.1.1.** Requires an aggregation function that takes a collection of blood bags and produces a collection of results, where each result item consists of a blood type and an amount. [P1]

##### 1.1.2. Hospital staff can sort results by blood type to see which types are more abundant. [P2]

**1.1.2.1.** Requires a sorting function that takes a collection of result items (described in *requirement 1.1.1.1*) and sorts them by amount. [P2]

- 1.1.3. Hospital staff can use a blood type filter to see information relating to particular blood types. [P2]
  - 1.1.3.1. Requires a filtering function that takes a collection of blood bags and a set of blood types and produces a collection containing only those blood bags that contain one of the specified types. [P2]
- 1.1.4. Hospital staff can view the results of the query in an easy-to-read format (such as table). [P1]
- 1.2. A member of the public (potential donors) may enquire about how, when and where they may give blood. [P4]
  - 1.2.1. Potential donors can find out what the health requirements are for giving blood. [P4]
    - 1.2.1.1. Requires the Vampire site to have a page that lists the health requirements for giving blood. [P4]
  - 1.2.2. Potential donors are able to find out what locations the Batmobile will visit. [P4]
    - 1.2.2.1. Requires the Vampire site to have a page that displays the Batmobile's schedule. [P4]
- 1.3. A donor may enquire about the outcome of their blood donation. [P4]

*Assumptions:*

- When the hospital uses a donor's blood to treat a patient, the hospital will acknowledge the use of the blood.
- 1.3.1. Donors can check the results of their blood test to see if their blood was clean or unclean, and if unclean, which tests came back positive for disease. [P4]
  - 1.3.1.1. Requires the system to keep a record of all blood donations that have been made. [P4]
- 1.3.2. Donors can check if their blood was used to save another life. [P4]
  - 1.3.2.1. Requires the system to have a portal through which hospital staff can acknowledge the use of a donor's blood. [P4]

## 2. Hospital staff can request blood from Vampire. [P1]

*Assumptions:*

- Hospital staff can request multiple blood types at the same time. A full request (called a **batch of requests**) will be in the form of a mapping from blood type to bags/volume requested.
- Since Vampire staff dispose of expired blood daily, the inventory will contain only fresh blood at all times, so there is no need to filter out expired blood.
- For simplicity, blood delivery and transportation occur instantaneously.

- 2.1. The system must be able to check if the batch of requests can be fulfilled. **[P1]**
  - 2.1.1. Requires a function that takes a collection of blood bag objects and a collection of requests and determines if the request can be fulfilled. **[P1]**
    - 2.1.1.1. See *requirement 1.1.1.1* **[P1]**
- 2.2. The system must fulfill requests that can be fulfilled. **[P1]**
  - 2.2.1. Requires a function that fulfills a full batch of requests. **[P1]**
    - 2.2.1.1. Requires a function that fulfills a single request, which takes the blood inventory and split it into a collection that is used to fulfill the requests, and the remaining inventory. **[P1]**
  - 2.2.2. The system should additionally display a message to the hospital staff to confirm that the request has been fulfilled. **[P2]**
- 2.3. The system must send an alert to Vampire headquarters if a request cannot be fulfilled (see *requirement 4*). **[P1]**
  - 2.3.1. The system should also inform the hospital staff of the current situation and assure them their request will be handled but may be delayed. **[P2]**
- 2.4. The system should prioritise blood that is closer to expiration to minimise wastage. **[P2]**
  - 2.4.1. Requires a function that sorts a collection of blood bags by age. **[P2]**
- 2.5. The system should check if the inventory has low blood supply after fulfilling a request (see *requirement 4.2*). **[P1]**

### 3. Vampire staff can dispose of all expired blood within the inventory. **[P1]**

#### *Assumptions:*

- Expired blood is disposed of daily.
- All expired is sent to the blood dump so that the blood can be disposed of in a medically suitable fashion.

- 3.1. Vampire staff can make a request to the system to remove expired blood from the inventory. **[P1]**
  - 3.1.1. Requires a function to filter a collection of blood bags by age. **[P1]**
- 3.2. The system should check if the inventory has low blood supply after disposing of expired blood (see *requirement 4.2*). **[P1]**

#### 4. Vampire staff will ensure there is enough blood to satisfy all requests. [P1]

*Assumptions:*

- The default 'low supply' is assumed to be 50 bags or fewer.
- Every blood bag contains the same volume of blood, 3 Litres.
- Vampire staff can configure the threshold for 'low supply' by number of bags.
- Once Vampire Headquarters receives a low blood supply alert, Vampire headquarters will do something (e.g., acquire blood from emergency donors) to push the blood supply up to non-critical levels. Vampire Headquarters must acknowledge the request.
- There are enough emergency donors at all times to push the blood supply up to a non-critical level.

**4.1.** The system must check if the inventory is low on supply for any blood types after performing any operation that removes blood. [P1]

**4.1.1.** Requires a function that takes a collection of blood bags and determines if there are any blood types that are in low supply. Blood types that are in low supply are added to an 'alert' collection. [P1]

**4.1.1.1.** See *requirement 1.1.1.1* [P1]

**4.2.** The system must automatically alert Vampire headquarters if there is a low supply of any blood type. [P1]

#### 5. Depositing tested blood. [P1]

*Assumptions:*

- The Batmobile never deposits blood directly into Vampire's inventory - it always passes the blood to Pathology for testing, which then tests the blood and then deposits the blood into Vampire's inventory if the blood was clean.
- Donors can also donate at a hospital. The process is still the same - the blood is passed to pathology for testing.
- Pathology staff sends unclean blood to the blood dump so that it can be disposed of in a medically suitable fashion.
- Pathology staff enter details of unclean blood test into Vampire's system so that the donor can be notified.
- Pathology staff properly tests the blood for infectious diseases.
- All blood that is stored within the inventory is clean blood that contains no positive results for diseases.
- Information about the origin of the blood (donor name, donor id, blood type, donation date, donation location) is available, and was taken during the donation.
- For simplicity, it is assumed that the transportation of blood is instantaneous.
- It's assumed that in *requirement 5.1.1*, Pathology staff entering the details of the tested blood is equivalent to the blood being deposited into the inventory

- 5.1. Pathology staff can deposit clean, tested blood into the inventory. [P1]
  - 5.1.1. Pathology staff can enter the details of the tested clean blood into the system. [P1]
    - 5.1.1.1. The details must include the donor's name, id, and blood type, and the date and location where the donation occurred. [P1]
    - 5.1.1.2. Requires a function that takes a collection of clean blood and inserts it into the inventory. [P1]
    - 5.1.1.3. Requires the system to store blood test results. [P3]
- 5.2. Pathology staff may enter details of blood tests for unclean blood. [P3]
  - 5.2.1. The details include the donor's name, id, and blood type, the date and location of the donation, and tests that came up positive. [P3]
  - 5.2.2. Requires the system to store blood test results (see requirement 5.1.1.3). [P3]

## **6. All relevant users have an account on the system. [P4]**

- 6.1. Requires a database to store account details, e.g., ids, email addresses, (encrypted) passwords, notifications. [P4]
- 6.2. New users are able to create an account on the system. [P4]
  - 6.2.1. Requires a sign-up page. [P4]
  - 6.2.2. New donors can sign up with just their name and email address. [P4]
  - 6.2.3. Medical professionals (e.g., hospital staff and Pathology staff) need to provide additional identification such as proof of employment. [P4]
    - 6.2.3.1. May require the use of an external API service to prove identity and employment. [P4]
- 6.3. Users can log in to their account with their id and password. [P4]
  - 6.3.1. Requires a login page. [P4]
  - 6.3.2. Requires a server to authenticate users and a database (see *requirement 6.1*). [P4]
- 6.4. Every user has a profile and can update their personal details, such as their phone number and password. [P4]
  - 6.4.1. Requires a profile page and options to update and save details. [P4]
  - 6.4.2. Requires a database to store and update users' details. [P4]
- 6.5. When signed in, a user can perform the actions they are authorised to do. [P4]
  - 6.5.1. Requires a menu to enable users to select what action they want to perform. [P4]
  - 6.5.2. Requires pages for each action (e.g., querying the inventory, submitting blood test details) and a server to handle actions. [P4]



## 3 Use-Cases

The following use cases covers all Priority 1 requirements as well as some Priority 2 requirements.

### 3.1 Request blood

|                      |   |
|----------------------|---|
| <b>Use Case</b>      | Hospital staff request blood from Vampire   |
| <b>Actor</b>         | Hospital staff  |
| <b>Stakeholders</b>  | Hospital staff, Vampire staff, Vampire Headquarters   |
| <b>Overview</b>      | A hospital needs blood to treat patients. An employee at the hospital makes a request to the system specifying how much of each blood type they want. The system checks if there are sufficient supplies to carry out the request. If there is, the system delivers the requested amount. Otherwise, the system sends an alert to Vampire HQ. |
| <b>Category</b>      | Priority 1  |
| <b>Trigger</b>       | A hospital is low on blood and needs more to treat patients.  |
| <b>Precondition</b>  | The requested blood types are valid, blood in Vampire's inventory is not expired  |
| <b>Postcondition</b> | If the request can be fulfilled, then the blood used to fulfill the request is removed from the inventory, and the hospital receives the blood. Otherwise, Vampire headquarters is alerted of the blood shortage and no change is made to the inventory.  |

#### 3.1.1 Basic flow

| <b>Description</b> |  | <b>Requirement(s)</b> |
|--------------------|--|-----------------------|
| <b>1</b>           | A hospital staff queries the system to see the levels of different blood types in Vampire's inventory.             | 1.1.1                 |
| <b>2</b>           | The system determines the total amount of each blood type in the inventory by aggregating the blood based on type. | 1.1.1.1               |

|           |   |                             |
|-----------|---|-----------------------------|
| <b>3</b>  | The system returns the result of the query and it is displayed to the user.   | 1.1.4                       |
| <b>4</b>  | The user filters the query result by selected blood type.   | 1.1.3                       |
| <b>5</b>  | The system responds by showing the volume of available blood of only those types, after filtering out other blood types.  | 1.1.3.1                     |
| <b>6</b>  | The hospital staff then sends a request / multiple requests to the system for amounts of different blood types.   | 2                           |
| <b>7</b>  | The system checks if the request(s) can be fulfilled and determines that it can be fulfilled.   | 2.1                         |
| <b>8</b>  | For each blood type requested, the system fulfills the request by filtering the inventory to get blood of that type, and then removing the appropriate amount from the inventory. | 2.2<br>2.2.1<br>2.2.2.1     |
| <b>9</b>  | To minimise wastage, the system may choose to use the oldest blood to fulfill the request, by first sorting the blood by age.   | 2.4                         |
| <b>10</b> | The system displays a confirmation message to the user.   | 2.2.2                       |
| <b>11</b> | The hospital receives the blood and uses it to treat patients.  | Requirement 2<br>Assumption |
| <b>12</b> | The system checks if there is still a sufficient supply of each blood type and finds that there is.   | 2.5 / 4.1                   |

### 3.1.2 Alternative flow 1

| <b>Description</b> |  | <b>Requirement(s)</b> |
|--------------------|--|-----------------------|
| <b>1</b>           | Failure scenario - not enough blood in inventory to fulfill request.<br><br>A hospital staff queries the system to see the levels of different blood types in Vampire's inventory. | 1.1.1                 |

|          |  |         |
|----------|--|---------|
| <b>2</b> | The system determines the total amount of each blood type in the inventory by aggregating the blood based on type. | 1.1.1.1 |
| <b>3</b> | The system returns the result of the query and it is displayed to the user.  | 1.1.4   |
| <b>4</b> | The hospital staff then sends a request / multiple requests to the system for amounts of different blood types.    | 2       |
| <b>5</b> | The system checks if the request(s) can be fulfilled and determines that it can't be fulfilled.                    | 2.1     |
| <b>6</b> | The system sends an alert to Vampire headquarters so they can do something about the shortage.                     | 2.3     |
| <b>7</b> | The system displays a message to the user to inform them of the situation and that their request will be delayed.  | 2.3.1   |

### 3.1.3 Alternative flow 2

| <b>Description</b> |  | <b>Requirement(s)</b> |
|--------------------|--|-----------------------|
|                    | Partial failure scenario - low supply after fulfilling the blood request.  |                       |
| <b>1</b>           | A hospital staff queries the system to see the levels of different blood types in Vampire's inventory.             | 1.1.1                 |
| <b>2</b>           | The system determines the total amount of each blood type in the inventory by aggregating the blood based on type. | 1.1.1.1               |
| <b>3</b>           | The system returns the result of the query and it is displayed to the user.  | 1.1.4                 |
| <b>4</b>           | The hospital staff then sends a request / multiple requests to the system for amounts of different blood types.    | 2                     |
| <b>5</b>           | The system checks if the request(s) can be fulfilled and determines that it can be fulfilled.                      | 2.1                   |

|           |   |                             |
|-----------|---|-----------------------------|
| <b>5</b>  | For each blood type requested, the system fulfills the request by filtering the inventory to get blood of that type, and then removing the appropriate amount from the inventory. | 2.2<br>2.2.1<br>2.2.1.1     |
| <b>6</b>  | To minimise wastage, the system may choose to use the oldest blood to fulfil the request, by first sorting the blood by age.  | 2.4                         |
| <b>7</b>  | The system displays a confirmation message to the user.   | 2.2.2                       |
| <b>8</b>  | The hospital receives the blood and uses it to treat patients.  | Requirement 2<br>Assumption |
| <b>9</b>  | The system checks if there is still a sufficient supply of each blood type and finds that the inventory is low on a particular blood type.  | 2.5 / 4.1                   |
| <b>10</b> | The system sends an alert to Vampire headquarters so they can do something about the shortage.  | 4.2                         |

### 3.2 Dispose of expired blood

|                      |   |
|----------------------|---|
| <b>Use Case</b>      | Vampire staff dispose of expired blood to ensure Vampire's blood inventory does not have any expired blood.                                 |
| <b>Actor</b>         | Vampire staff   |
| <b>Stakeholder</b>   | Vampire staff, Blood dump, Vampire Headquarters   |
| <b>Overview</b>      | Blood donations will eventually expire, so Vampire needs to frequently search the blood inventory for expired blood and dispose of it.      |
| <b>Category</b>      | Priority 1  |
| <b>Trigger</b>       | It is the start of the day.   |
| <b>Precondition</b>  | None.   |
| <b>Postcondition</b> | All expired blood at the time of the request is removed from the store and sent to the blood dump. Additionally, if disposing expired blood |

|  |  |
|--|--|
|  | causes Vampire to have low supply for any blood type, an alert will be sent to Vampire headquarters. |
|--|--|

### 3.2.1 Main success scenario

| Description |   | Requirement(s)              |
|-------------|---|-----------------------------|
| <b>1</b>    | Main success scenario.  | 3<br>3.1                    |
| <b>2</b>    | At the beginning of the day, a Vampire staff makes a request to the system to remove all expired blood. | 3.1.1                       |
| <b>3</b>    | The system filters the blood by age to find all the expired blood and removes it from the inventory.    | Requirement 3<br>Assumption |
| <b>4</b>    | The expired blood is sent to the blood dump.  | 3.2 / 4.1                   |
|             | The system checks if there is still a sufficient supply of each blood type and finds that there is.     |                             |

### 3.2.2 Alternative flow

| Description |   | Requirement(s)              |
|-------------|---|-----------------------------|
| <b>1</b>    | Partial failure scenario - low supply after disposing of old blood.                                     | 3                           |
| <b>2</b>    | At the beginning of the day, a Vampire staff makes a request to the system to remove all expired blood. | 3.1<br>3.1.1                |
| <b>3</b>    | The system filters the blood by age to find all the expired blood and removes it from the inventory.    | Requirement 3<br>Assumption |
| <b>4</b>    | The expired blood is sent to the blood dump.  | 3.2 / 4.1                   |
| <b>5</b>    | The system checks if there is still a sufficient supply of each blood type and finds that there isn't.  |                             |
|             | The system sends an alert to Vampire headquarters so they can do something about the shortage.          | 4.2                         |

### 3.3 Deposit tested blood

|                      |  |
|----------------------|--|
| <b>Use Case</b>      | Pathology staff deposit clean blood into Vampire's inventory after running various blood tests on it.  |
| <b>Actor</b>         | Pathology staff  |
| <b>Stakeholder</b>   | Pathology staff, Vampire staff   |
| <b>Overview</b>      | After testing donor blood, Pathology staff will enter the details of the clean blood into Vampire's system and deposit the clean blood into Vampire's inventory. |
| <b>Category</b>      | Priority 1   |
| <b>Trigger</b>       | Pathology has finished testing a batch of donor blood.   |
| <b>Precondition</b>  | The blood to be inserted is clean  |
| <b>Postcondition</b> | The blood is added to Vampire's inventory  |

#### 3.3.1 Main success scenario

| <b>Description</b> |  | <b>Requirements</b>                    |
|--------------------|--|--|
| <b>1</b>           | Pathology staff enters details of the expired blood into the system, including the donor's name, id, and blood type, and the date and location of the donation | 5.1.1<br>5.1.1.1                       |
| <b>2</b>           | At the same time, the blood is deposited into Vampire's inventory  | Requirement 5<br>Assumption<br>5.1.1.2 |

### 3.4 Maintain blood inventory

|                      |  |
|----------------------|--|
| <b>Use Case</b>      | Vampire staff ensure that Vampire Headquarters is always notified of a low supply of any blood type.             |
| <b>Actor</b>         | Vampire system   |
| <b>Stakeholder</b>   | Vampire staff, Vampire Headquarters  |
| <b>Overview</b>      | The system must notify Vampire headquarters when blood levels drop below 50 bags of blood for any type of blood. |
| <b>Category</b>      | Priority 1   |
| <b>Trigger</b>       | An update is made to the inventory that removes blood.   |
| <b>Precondition</b>  | User is registered Vampire staff.  |
| <b>Postcondition</b> | If Vampire's inventory has a low supply of any blood supply Vampire Blood inventory replenished.                 |

#### 3.4.1 Success scenario 1

| <b>Description</b> |  | <b>Requirements</b> |
|--------------------|--|---------------------|
|                    | Success scenario 1 - There is enough blood supply for each blood type so no action is taken  |                     |
| <b>1</b>           | After an update to the inventory that removes blood (e.g., fulfilled blood request, disposal of expired blood), the Vampire system automatically checks the levels of all blood types. | 4.1                 |
| <b>2</b>           | Any blood type below the 'low-blood' threshold will be recorded on an alert list as system cycles through each blood type.   | 4.1.1               |
| <b>3</b>           | There are no blood types that are in low supply so nothing is done   | 4.2                 |

### 3.4.2 Success scenario 2

| <b>Description</b> | Success scenario 2 - Vampire HQ is notified about a low supply of some blood type(s) in Vampire's inventory and Vampire HQ restores the blood supply                                   | Requirements             |
|--------------------|--|--------------------------|
| <b>1</b>           | After an update to the inventory that removes blood (e.g., fulfilled blood request, disposal of expired blood), the Vampire system automatically checks the levels of all blood types. | 4.1                      |
| <b>2</b>           | Any blood type below the 'low-blood' threshold will be recorded on an alert list as system cycles through each blood type.   | 4.1.1                    |
| <b>3</b>           | The system pushes an alert to Vampire Headquarters for all blood types on the alert list.  | 4.2                      |
| <b>4</b>           | The system requires the acknowledge button to be clicked on by Vampire Headquarters for the alert to be dismissed.   | Requirement 4 Assumption |
| <b>5</b>           | Vampire Headquarters does something about the shortage and pushes the blood supply levels up to a non-critical level.  | Requirement 4 Assumption |



## **4 Work Breakdown Structure**

The following work breakdown structure splits the work into 5 work packages; project management, frontend backend, verification and functional correctness. The number in brackets next to each deliverable relates to the estimated overall percentage of work required to complete. Each deliverable that links to a requirement will have it listed under its percentage in the following format  
requirements: requirement x.x.x, requirement x.x.y and so on

The work breakdown structure diagram is attached to the next page of this document

## 5 Teamwork

- Abanob Tawfik, z5075490
  - Requirements, WBS
- Kevin Luxa, z5074984
  - Requirements, Use Cases
- Lucas Pok, z5122535
  - WBS
- Rason Chia, z5084566
  - Use Cases
- Michael Yoo, z5165635
  - Executive Summary