Train Trax: Train Monitor for Positive Train Control Test Beds

Software Requirements Specification

Document

Version 1.8

12/08/2015

Stephen Jalbert

Rashad Madyun

Corey Sanders

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Description | Author |
| 1.0 | 10/02/2015 | Initial Version. Created temporary template for specification for system requirement delivery. | Stephen Jalbert  Rashad Madyun  Corey Sanders |
| 1.1 | 10/05/2015 | Updated document to new template. Improved document based on feedback with initial review. | Stephen Jalbert  Rashad Madyun  Corey Sanders |
| 1.2 | 10/19/2015 | Updating document to include specific requirements, definitions, and some analysis modeling of the system. | Stephen Jalbert  Rashad Madyun  Corey Sanders |
| 1.3 | 10/26/2015 | Updating document to include requirements for determining the required accuracy of angular velocity and acceleration measurements. Organized Requirements into Tables. Added formal use cases. Updated the software interfaces and communication interfaces sections. Added back references from version 1.0. Renamed the ‘Definitions, abbreviations, …’ section to the Glossary. | Rashad Madyun  Corey Sanders |
| 1.4 | 11/9/2015 | Reformatted Requirements, Relabeled requirements, and Updated terminology. Updated System Overview. Updated the System Interfaces section. | Stephen Jalbert  Corey Sanders |
| 1.5 | 11/16/2015 | Added Activity Diagrams to describe each Use Case.  Updated the labelling of the document.  Remove “Record Track Geometry” and “Save Track Geometry” Use Cases because they add little value to the customer. | Stephen Jalbert  Corey Sanders |
| 1.6 | 11/23/2015 | Removed use of Track Section for Track Block and expanded definition.  Updated Text for clarity, including better descriptions for how the system is expected to operate.  Added figure table. | Stephen Jalbert  Corey Sanders |
| 1.7 | 12/01/15 | Improved the language of the requirements to make them more testable.  Tried to make clearer parent-child relationships in the requirements.  Added new requirements and removed unnecessary requirements where appropriate.  Replaced ‘Train Controller’ terminology with ‘Command Station’ and ‘Switch Controller’ terminology. Expanded definition of Command Station.  Updated terms used in scope to clarify role of JMRI with the system.  Improved formatting of use case tables.  Split Motion Detection Unit requirements into hardware and software requirements | Corey Sanders  Stephen Jalbert |
| 1.8 | 12/08/15 | Updated glossary with the latest terms.  Added a new requirement to the GUI for showing the direction that the train will go when it crosses the next switch.  Updated figures to remove JMRI and show the latest understanding of the existing train system.  Updated the text describing the scope of the project. | Stephen Jalbert  Corey Sanders |

Table of Contents

Revision History 2

Table of Contents 4

Figures 5

**1** **Introduction** 6

**1.1** **Purpose** 6

**1.2** **Scope** 6

**1.3** **Glossary** 7

**1.4** **References** 9

**1.5** **Overview** 9

**2** **Overall description** 9

**2.1** **Product perspective** 13

**2.1.1** **System interfaces** 13

**2.1.2** **User interfaces** 13

**2.1.3** **Hardware interfaces** 14

**2.1.4** **Software interfaces** 14

**2.1.5** **Communications Interfaces** 15

**2.1.6** **Operations** 16

**2.1.7** **Site adaptation requirements** 16

**2.2** **Product functions** 16

2.2.1 Use Cases 17

2.2.2 Activity Diagrams 20

**2.3** **User characteristics** 22

**2.4** **Constraints** 22

**2.5** **Assumptions and dependencies** 22

**3** **Specific requirements** 23

3.1 System Component Requirements 23

3.1.1 Hardware Device: Motion Detection Unit 25

3.1.2 Firmware Program: Motion Detection Unit 25

3.1.3 Software Program: Train Navigation Service 27

3.1.4 Software Program: Train Navigation Database 29

3.1.5 Software Program: Train Monitor Terminal GUI 30

3.2 Performance Requirements 32

3.3 Other Requirements 32

**4** **Appendixes** 32

4.1 Analysis Models 32

4.1.1 Class Responsibility Collaborator Cards 32

Figures

[Figure 1 Positive Train Control Test Bed Without train trax 9](#_Toc436731293)

[Figure 2 Control Flow of Positive Train Control Test Bed Without Train Trax 10](#_Toc436731294)

[Figure 3 Control Flow Diagram of Train Trax Interaction with Positive Train Control Test Bed 11](#_Toc436731295)

[Figure 4 Train Trax Communication Diagram 14](#_Toc436731296)

[Figure 5 Monitor Train Position Use Case Activity Diagram 19](#_Toc436731297)

[Figure 6 Control Track Switch Use Case Activity Diagram 20](#_Toc436731298)

[Figure 7 Block Diagram of Main Components of System 22](#_Toc436731299)

[Figure 8 Control Flow of Train Trax System components integrated with Positive Train control Test Bed 23](#_Toc436731300)

[Figure 9 Placement of System Components Within the Positive Train Control Test Bed 23](#_Toc436731301)

# **Introduction**

In a real-word train environment, trains often carry very valuable assets: people, oil, merchandise, etc. It is important for railway system to be able to track the location of each train in order to prevent collisions and to monitor the state of trains in the event of attack.

## **Purpose**

The Computer Engineering department owns a Positive Train Control Test Bed, which is intended to mirror a typical train environment. The purpose of the train track is to be a teaching tool for instructing students on creating safety critical software. It is desired for the department’s Positive Train Control Test Bed to be able to track the location in for each train for this reason. Like subway trains, the department Positive Train Control Test Bed is completely indoors, so a Global Position System (GPS) is not possible.

The purpose of this document is to describe the needs and expectations for the Train Trax Train Monitor to assist the department with tracking trains as they move along the Positive Train Control Test Bed. It will cover requirements for both the desktop application and the embedded system software.

## **Scope**

Train Trax's primary purpose is to estimate the position of each train operating along the Positive Train Control Test Bed accurately enough to allow Train Operators schedule trains to run close enough to operation on the same section of track with minimal risk of collision. Additionally, Train Trax provides a means for Train Operators to easily control switches on the train track without the need to using any additional train control software. Train Trax is only a monitor for trains, not train control software. Furthermore, the development team is to assist the department with any modifications necessary to the Positive Train Control Test Bed to support proper operation of Train Trax, including the placement of markers on the track at pre-designated locations.

Train Trax consists of hardware that is equipped on either the train engine or rail cars to measure train movement. It also consists of software that will run on existing equipment within the department to graphically display train positions and to control movement.

A unit is attached to a rail car that is equipped with an Inertial Motion Unit (IMU) that measures the acceleration and angular velocity (rotational vectors) of the rail car as it is tugged by the train along the track. This unit, called a Motion Detection Unit, will send its collected measurements over WIFI to a train monitor terminal (i.e. computer) that will estimate the train’s position using numerical integration to solve for displacement kinematic equations. The resulting position is then displayed on the terminal as well as the layout of the track itself. RFID tags, whose position is already recorded in a database, will be used as the track markers and placed strategically throughout the track so that they can correct the position calculated from IMU measurements. Lastly, the monitor terminal displays representations of all of the switches on the track and allows the user to control them through a GUI that sends LOCONET messages to the track's switch controllers, which then control relays to change a switch’s state. Train control software, such as JMRI, is expected to be used to control/throttle the movement of the train via LOCONET messages to the Train Command Station.

## **Glossary**

**Digital Command Control (DCC)**

Digital Command Control protocol which is a electric signaling protocol used to control train engines on a train track through the rails.

**Java Model Railroad Interface (JMRI)**

Popular open-source software suite for controlling model trains.

**Inertial Motion Unit (IMU)**

A hardware device often composed of an accelerometer and a gyroscope used to perform dead-reckoning of the position of objects based on measurements of effects of forces acting on an object in space.

**LocoNet**

An Ethernet-link proprietary communication protocol created by DigiTrax for full train and track layout control of model train sets.

**Position**

A description of where a given object is located on the Position Train Control Test Bed. It uses a relative coordinate system based on the distance from a fixed point on the table.

**Positive Train Control Test Bed**

A model train system designed to scale to represent actual railway systems. Its purpose is to facilitate the testing, design, and training of train control systems without the risk of associated performing these activities on live trains, such as bodily injury and costs for scheduling and operating full scale trains.

.

**Rail Car**  
Simple wheeled container that is attached to the train to carry cargo.

**Railway System Owner**

The entity that owns Positive Test Control Test Bed.

**Radio Frequency Identification (RFID)**

Data exchange method that relies on the properties of induction to read information imprinted on a device when in close proximity.

**Track**  
The track is a pair of metal rails that the train runs on top of to move. It provides both power and control signals to the train. It is divided into different physical pieces called sections to simplify its assembly.

**Track Block**

A segment of the entire track of the test bed, which has been divided and identified into segments by the Train Technician and Train Operator, which is used to highlight areas of interest by these individuals and to divide the track into regions from which trains can go in different directions on the track. In practice, a block is Track Circuit Block. It is a single element where the Positive Train Control Test Bed Can Detect whether one or more trains is on it or not.

**Track Marker**  
Special hardware placed at different spots on the track to highlight places of interest on the track. Examples of train markers include RFID tags that are read by the train as it moves along the track, and track sections that signal when one or more trains are present.

**Track Switch**Devices on the track to control the direction of train engine movement by changing the sections of track that are connected together.

**Track Switch Controller**

A hardware device that is the bridge between hardware that physically controls switches of the test bed and software being used to remotely control the test bed. It is attached to the track that translates requests from operators to control track switches on the test bed into signals to switch relays that move the switches into different positions.

**Train**

A to-scale model of a commercial train engine. It is the primary vehicle used to move along the test bed

**Train Command Station**

A hardware device that is the bridge between hardware that physically controls trains of the test bed and software being used to remotely control the test bed. It is attached to the track that translates requests from operators to control the train into control signals that the train understands.

**Train Control Terminal**

The equipment, such as a laptop, used by the system to allow operators to control trains that belong to the test bed.

**Train Monitor Development Team**

A group of people who have been commissioned by the Railway System Owner to create a system for tracking the movement of trains along the railways system real time.

**Train Monitor Terminal**

The display equipment, such as a laptop, used by the system visually display to operators information about the test bed.

**Train Occupancy Detector**

A hardware device that is the bridge between hardware that physically detects when one or more trains are on a section of track and software being used to report train locations. It is attached to the track and uses changes in current draw that occur when one or more trains are being powered by a track block in order to detect train occupancy in that block. Lastly, it can issue messages for when a train is entering or exiting a track block.

**Train Operator**

A person or machine that controls one or more of the trains on the Positive Train Control Test Bed.

**Train Technician**

A train technician is a person who maintains the Positive Train Control Test Bed.

## **References**

1. IEEE Guide to Software Requirements Specifications (Std 830-1993). <https://standards.ieee.org/findstds/standard/830-1993.html>
2. University of Colorado at Colorado Springs for the CS330 Software Engineering Class Software Requirements Template. <http://www.uccs.edu/Documents/tboult/srs.doc>

## **Overview**

The remainder of this SRS will provide prospective for the requirements and the system followed by the specific requirements.

# **Overall description**

The following sections will provide high-level detail of the prospective of design, system functions, characteristics of use, and any assumptions and dependencies for the Train Position Monitor.

As shown in Figure 1, The Positive Train Control Test Bed is a model train system designed to scale to represent actual railway systems. Its purpose is to facilitate the testing, design, and training of train control systems without the risk of associated performing these activities on live trains, such as bodily injury and costs for scheduling and operating full-scale trains. A Positive Train Control Test Bed has the following components: train, rail cars, track, track markers, rail switches, train command station, train switch controller(s), and train monitor terminal. Figure 2 shows how the test bed components work together to control train movement along the track.



Figure 1 Positive Train Control Test Bed Without train trax

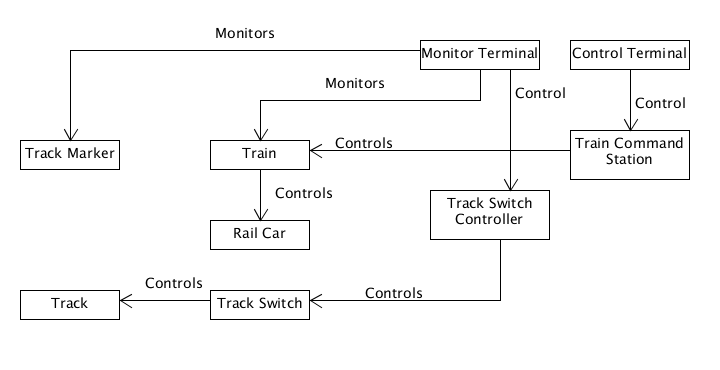


Figure 2 Control Flow of Positive Train Control Test Bed Without Train Trax

Figure 1 and Figure 2 show how the Positive Train Control Test Bed currently operates without Train Trax. In the existing Positive Train Control Test Bed, track markers are actually the track blocks themselves. There is hardware on the track, called a Train Occupancy Detector, to detect when one or more trains enter a block and when there are not any trains on a block based on the current draw on the track block. The Train Occupancy Detector sends messages for transitions for when the block is occupied (one or more trains on the block) and unoccupied (no trains on the block). When the track crosses into a track marker (i.e. track block), information about the marker that was crossed is relayed to the Monitor Terminal so that it can update the train’s last known position based on the known position of the track marker. Since the rail cars are attached to the train, the train controls where the rail cars move. The Train Command Station controls the speed of the train the direction that it moves along the track: either backward or forward. The Track Switch Controller controls track switches which in turn change the configuration of the track so that the path that the train moves along the track can be controlled.

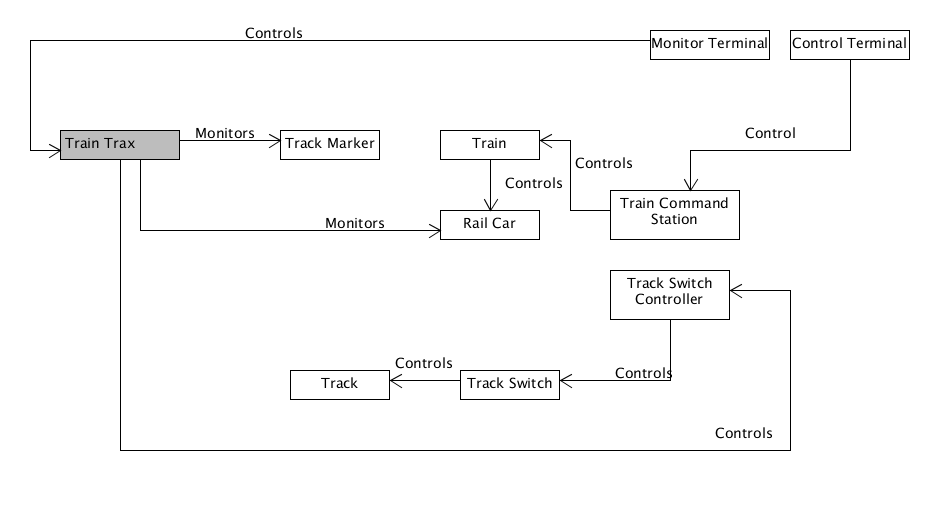


Figure 3 Control Flow Diagram of Train Trax Interaction with Positive Train Control Test Bed

Figure 3 shows how Train Trax will interact with the test bed. Train Trax will have a hardware device called the Motion Detection Unit placed on a rail car to detect changes in the motion of the rail car caused by it being pulled by the train as it moves along the track. Changes in motion include detection of the net force that is acting on the train (acceleration) as well as changes in rotation of the car (angular velocity). In practice, this means that acceleration is measured from an accelerometer that is part of an IMU, and angular velocity is measured from a gyroscope that is part of an IMU. Train Trax will also monitor the Track Markers that the rail car passes as it moves along the track to gain further information about where the train is. With Train Trax, track markers are actually RFID tags that are spaced along the track and whose position is stored in a database connected to the system. Train Trax will also control the Track Switch Controller(s) so that it can control Track Switches by sending LOCONET messages over the LAN where the Track Switch Controller(s) are connected. Lastly, Train Trax will be controlled by the Train Monitor Terminal so that it can get information about which switches the user wants to control, and so that it can display information about the current state of the test bed through a GUI that is operating from the terminal.

## **Product perspective**

### **System interfaces**

|  |  |  |
| --- | --- | --- |
| Interface | Description | Purpose |
| LOCONET | Interface used for communicating with the Train Command Station and Track Switch Controller of the Positive Train Control Test Bed. | Used to control track switches, and detect when the train leaves/enters a second of track. |
| JAVA Platform, Standard Edition | Environment that is running within the Train Monitor Terminal | Used to control the Train Monitor Terminal, including to get input from the Train Operator and display information to the Train Operator. Middleware for collaboration between internal modules of Train Trax. |

Table 1 SYSTEM INTERFACES

### **User interfaces**

The graphical user interface will consist of all windows that are required for the user to enter, store, and view information associated with the Train Trax system.

The UI will include the following types of views:

* Main Menu View
  + Initial view at startup. Presents the operations available to the system. This will show the layout of the track without any trains on it. It should resemble the general shape of the track and display switches and the division of train blocks on the track.
* Train Monitor View
  + This is the view for the Train Operator to observe trains as they move along the track. This is the primary window that will be used by train tracks. Train Operators can also change the state of switches by tapping on switches from this display.
* Train History View
  + This view allows the Train Operator to learn about where trains have travelled during a given span of time. It should be able to list a table of all of the reported movements of the train. As a bonus, it should graphically display the path travelled by each train.

### **Hardware interfaces**

|  |  |  |
| --- | --- | --- |
| Type | Description | Purpose |
| RS-232 | Popular protocol for serial communication across a DB-9 serial cable. | Used to Program Motion Detection Unit. Used to connect RFID Reader to Motion Detection Unit |
| Serial Peripheral Interface (SPI) | Three-wire serial protocol used to connect a CPU to peripherals. | Used to connect Wireless Ethernet Module to Motion Detection Unit |
| Inter-Integrated Circuit (I2C) | Two-wire serial protocol used to connect a CPU to peripherals | Used to connect IMU and Optical Sensor to Motion Detection Unit. |
| Universal Serial Bus (USB) | Popular four-wire serial protocol used for connection devices to PCs. | Used to connect RFID reader for testing.  Used by the Train Monitor Terminal to send LocoNet messages to test bed hardware, such as trains and switch controllers. |

Table 2 Hardware Interfaces

### **Software interfaces**

|  |  |  |
| --- | --- | --- |
| Type | Description | Purpose |
| Java Runtime API | Standard libraries provided by the Java Runtime Environment | Used to assist in Train Position Calculations By the Train Navigation Service  Used to render graphical displays for the Train Navigation GUI.  Provides interfaces for retrieving train measurements and other network data from Motion Detection Unit |
| Structured Query Language (SQL) | Standard language used to query and interact with databases. | Used for communication with the Train Navigation Database to access and store information. |
| Java Database Connectivity (JDBC) | Standard interface for connecting Java to a database. | Used to provide SQL access to the Train Navigation Database to the Train Navigation Service and the Train Navigation GUI. |

Table 3 Software Interfaces

### **Communications Interfaces**

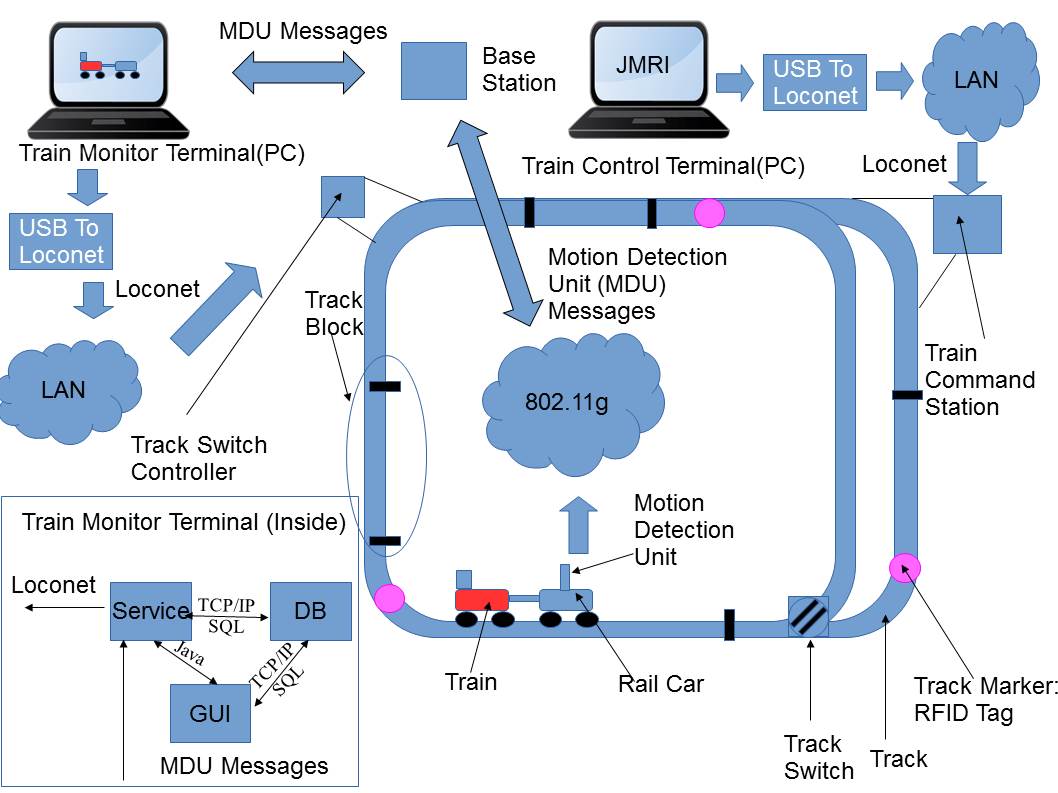


Figure 4 Train Trax Communication Diagram

|  |  |  |
| --- | --- | --- |
| Type | Description | Purpose |
| IEEE 802.11 abgn Wireless Ethernet | Popular 2.4 GHz Radio Physical and Datalink Protocols for Exchanging Information Between Machines | Establishes communication with the Train Command Station and Track Switch Controller.  Delivers real-time measurements of train movement from the Motion Detection Unit |
| Transmission Control Protocol (TCP) / Internet Protocol (IP) | Standard Transport / Network Protocols for transferring data across the Internet and within Local Networks. | Transport for measurements from the Motion Detection Unit.  Transport for communication with the Train Navigation Database. |
| LOCONET | Ethernet-like messaging protocol used to monitor and control model train systems. | Used for communication with the Train Command Station and Track Switch Controller. |

Table 4 Communication interfaces

As shown in Figure 4 and Table 4, most of Train Trax’s communications interfaces are for communication with different parts of itself. LOCONET messages are the primary vehicle for Train Trax to exchange information with parts of the test bed itself. These messages are transported via Wireless Ethernet. Communication between the Motion Detection Unit attached to a rail car and the Navigation Service running within the Train Monitor Terminal is to be custom message set created specifically for communicating with the Motion Detection Unit called MDU messages. MDU messages will be transported directly using Wireless Ethernet. If necessary, a base station may be used to relay messages from the Motion Detection Unit to the Train Monitor Terminal and vice versa.

### **Operations**

The Train Position Monitor will provide one mode of operation. This operation mode will encapsulate the providing of a GUI to the user that allows visualization of the position of trains in the system, and for the control of switches along the track.

### **Site adaptation requirements**

There are no site adaptation requirements for this system, as only one site is being developed for.

## **Product functions**

* Report the current position of each train on the rail system.
* Report the history of each train’s movements along the rail system.
* Predict the direction that a given train will go when it crosses the next switch in its path.
* Control switches on the rail system.
* Collect information to describe the shape and geometry of the track.
* Collect raw measurements used to estimate each train’s position.
* Alert when trains are too close together.
* Alert when there is a system failure.
* Alert when train reverses direction.
* Alert when train is stopped.

### Use Cases

|  |  |  |
| --- | --- | --- |
| **UC Name** | **Monitor Train** | |
| **Description** | The Monitor Train Use Case describes the process which will allow a train operator to monitor a moving train. | |
| **Actors** | Train Operator | |
| **Pre-Conditions** | A Train Operator is available.  Train is already moving.  Train Database already has information about all train markers on the test bed. | |
| **Post-Conditions** | Train is being monitored on track layout of the train monitor GUI. | |
| **Triggers** | Train operator chooses to monitor train movement. | |
| **Flow** | | |
|  | Actor | System |
|  | Launch the Train Monitor GUI from Train Monitor Terminal. |  |
|  |  | Displays layout of the track including all icons of switches. |
|  |  | Displays position icons and speed of each train on layout of track. |
|  | User Selects train icon |  |
|  |  | Symbol is displayed on icon to denote this is a train being monitored by operator. |
| **Exceptions** | Position estimate is calculated to be too inaccurate | |
| **Extension Points** | NONE | |

|  |  |  |
| --- | --- | --- |
| **UC Name** | **Monitor Train (Exception)** | |
| **Description** | Position estimate is calculated to be too inaccurate | |
| **Flow** | | |
|  | Actor | System |
|  | Launch the Train Monitor GUI from Train Monitor Terminal. |  |
|  |  | Displays layout of the track including all icons of switches. |
|  |  | Displays position icons and speed of each train on layout of track. |
|  |  | Highlights icon to indicate an error with the train |
|  |  | Text is displayed to indicate error details |
|  | User Selects train icon |  |
|  |  | Symbol is displayed on icon to denote this is a train being monitored by operator. |
| **Extension Points** | NONE | |

|  |  |  |  |
| --- | --- | --- | --- |
| **UC Name** | **Control Track Switch** | | |
| **Description** | The Control Track Switch Use Case describes the process which will allow a train operator to configure switches on the track for an upcoming train departure. | | |
| **Actors** | Train Operator | | |
| **Pre-Conditions** | Train Operator is available.  Train is already moving.  Train Database already has information about all train markers on the test bed. | | |
| **Post-Conditions** | Switch changes state | | |
| **Triggers** | Train Operator wishes to change the state of a switch. | | |
| **Flow** | | | |
|  | Actor | | System |
|  | Launch the Train Monitor GUI from Train Monitor Terminal. | |  |
|  |  | | Displays layout of the track including icons of all the switches. |
|  |  | | Display current state of switch on icon. |
|  | Press on the switch icon that corresponds to the switch wanted to change. | |  |
|  |  | | GUI acknowledgement of icon pressed. |
|  |  | | Processes switch change. |
|  |  | | Switch icon updated to new state. |
| **Exceptions** | Failure to send switch change to Track Switch Controller | | |
| **Extension Points** |  | | |
| **UC Name** | **Control Track Switch (Exception)** | | |
| **Description** | There is a failure to send switch change to the Track Switch Controller | | |
| **Flow** | | | |
|  | Actor | System | |
|  | Launch the Train Monitor GUI from Train Monitor Terminal. |  | |
|  |  | Displays layout of the track including icons of all the switches. | |
|  |  | Display current state of switch on icon. | |
|  | Press on the switch icon that corresponds to the switch wanted to change. |  | |
|  |  | GUI switch icon highlighted to indicate an error occurred with the switch. | |
|  | Press on highlighted icon | Processes switch change. | |
|  |  | Alert reports are displayed indicating there was a failure with the request to change the switch. | |
| **Extension Points** | NONE | | |

### Activity Diagrams

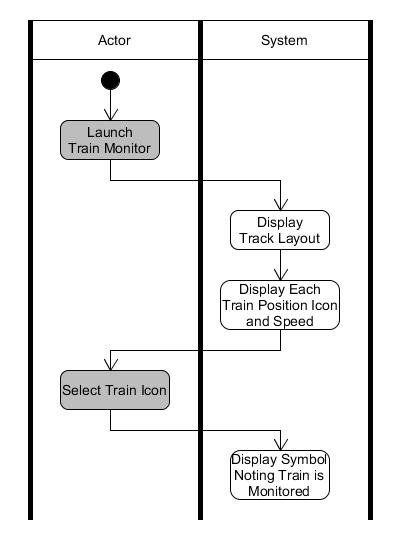


Figure 5 Monitor Train Position Use Case Activity Diagram

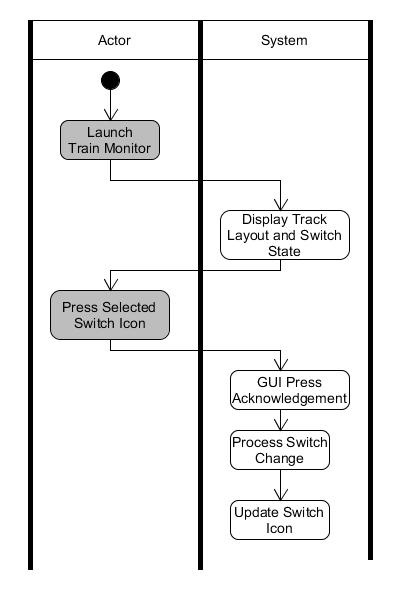


Figure 6 Control Track Switch Use Case Activity Diagram

## **User characteristics**

Stakeholders

* Train Operator
  + The operator’s primary responsibility is to ensure that trains reach their destination in a timely manner and to prevent collisions with other trains.
  + The train operator is the primary user for Train Trax.
* Train Technician
  + The technician ensures that the train track is in a state for trains to operation along and that trains are in a state to be able to move along the track and be controlled by train operators.
* Railway System Owner (UAH CPE Department)
  + The owner is responsible for providing all of the resources necessary for the Positive Train Control Test Bed to operate. This includes assigning and providing resources for Train Operators and Train Technicians.
* Train Monitor Development Team
  + The Train Monitor Development Team creates and maintains Train Trax to support the Railway System Owner’s Positive Train Control Test Bed.

## **Constraints**

* Train Trax must be able to work indoors.
* Train Trax must work with the existing Positive Test Control Bed, including the PC hardware acting as the Train Monitor Terminal to control and monitor the train.
* The Motion Detection Unit must use Inertial Measuring Units and RFID tags for position estimation.

## **Assumptions and dependencies**

* The system needs to be able to estimate the position of the train at least within inches so that we can measure if one train is close enough to another train to be able to be on the same section of track.
  + Based on customer estimate of the longest section of track being about 14 inches.
    - Feet or Meters would not precise enough.
    - Centimeters are more precise than inches.
* The track markers are placed frequently enough that the train maintains the minimum amount of accuracy necessary to prevent train collisions.
  + Already placed on Positive Train Control Test Bed.
  + At least one track marker is on each section of track.

# **Specific requirements**

## System Component Requirements

The train navigation system proposed by our team is composed of a User Interface (UI), a Motion Detection Unit, a Train Navigation Service, and a Train Position Database.  The UI is a graphical interface that is displayed from the Train Monitor Terminal to describe to Train Operators the current state of objects on the track including other trains, and to allow them to control switches on the track. The Motion Detection Unit is all of the hardware that is mounted onto one of the rail cars to capture information about when and how a train is moving, which includes Inertial Motion Units (IMUs), RFID tag readers, and the optical sensor. The Train Navigation Service is the brains of the system. It is a collection of all of the functions necessary to communicate with the Motion Detection Unit and the track as well as interpreting measurements from the Motion Detection Unit into location information for a given train. The Train Position Database is a datastore for all of the information that the train navigation system needs to permanently save all of the information needed by the system to operate, including settings and details about the geometry of the test bed.

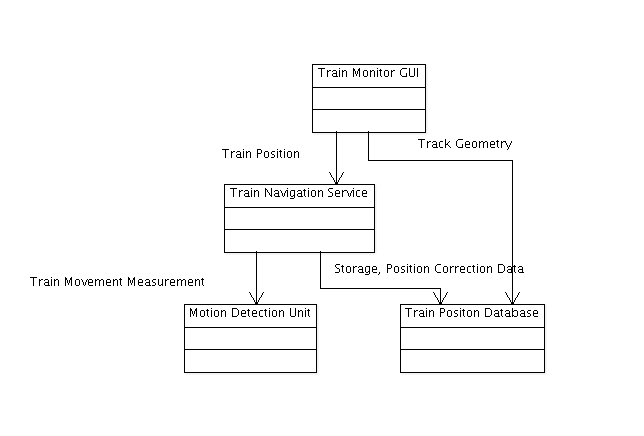


Figure 7 Block Diagram of Main Components of System

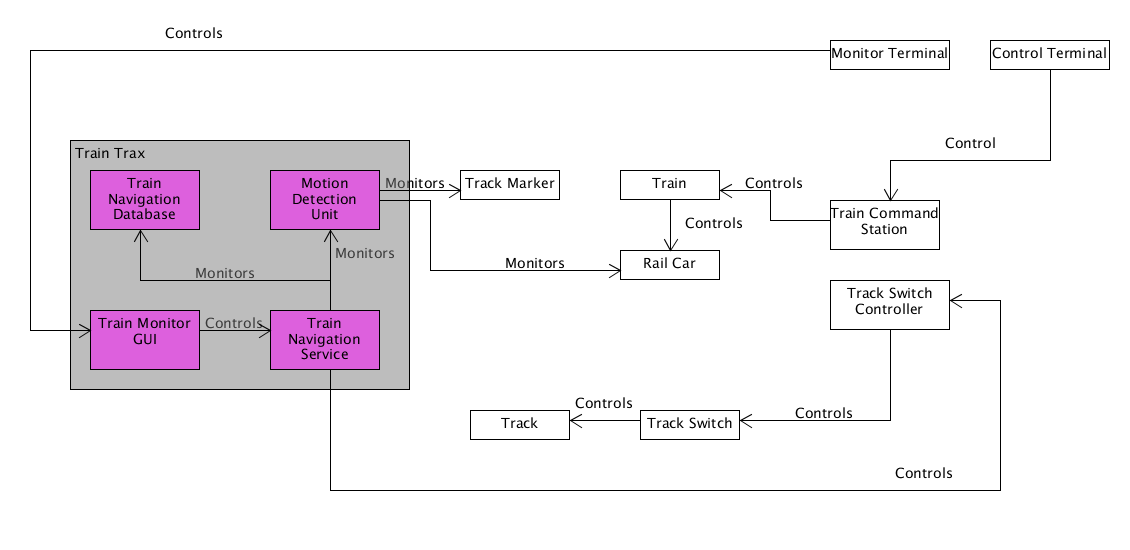


Figure 8 Control Flow of Train Trax System components integrated with Positive Train control Test Bed

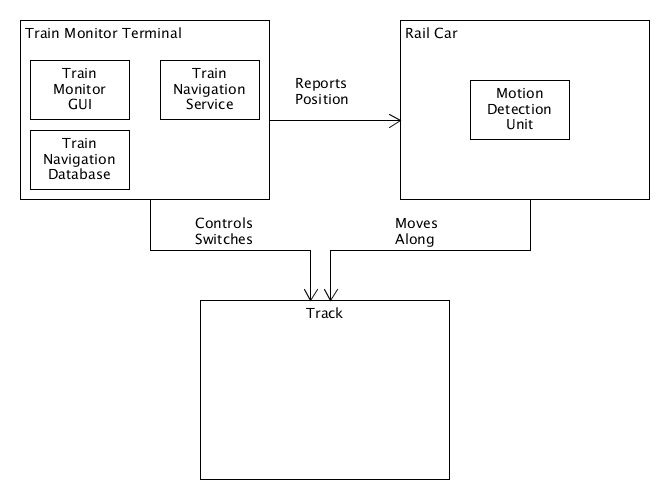


Figure 9 Placement of System Components Within the Positive Train Control Test Bed

### Hardware Device: Motion Detection Unit

| Requirement Number | Feature | Use Case | Priority |
| --- | --- | --- | --- |
| Requirement Text | | | |
| MDU-1000 | Estimate Train Position, Collect Raw measurements (Rollup) [Measurements] | Monitor Train |  |
| The Motion Detection Unit shall measure its own movement as it travels across the rail system. | | | |
| MDU-1010 | Estimate Train Position, Collect Raw measurements | Monitor Train |  |
| The Motion Detection Unit shall measure its acceleration across three independent axes (X, Y, Z) relative to the device. | | | |
| MDU-1020 | Estimate Train Position, Collect Raw measurements | Monitor Train |  |
| The Motion Detection Unit shall measure its angular velocity across three independent axes (X, Y, Z) relative to the device. | | | |
| MDU-1040 | Estimate Train Position, Collect Raw measurements | Monitor Train |  |
| When it moves across a RFID tag, The Motion Detection Unit shall identify the ID of the tag that was crossed. | | | |
| MDU-2000 | Estimate Train Position (Rollup) [Accuracy] | Monitor Train |  |
| So that a train operator can schedule trains closely together without risking collisions, the Motion Detection Unit shall measure its own movement across the rail system within a given a tolerance. | | | |
| MDU-2010 | Estimate Train Position, Collect Raw measurements | Monitor Train |  |
| The Motion Detection Unit shall measure its acceleration within +/- 0.1 meters per second squared. | | | |
| MDU-2020 | Estimate Train Position, Collect Raw measurements | Monitor Train |  |
| The Motion Detection Unit shall measure its angular velocity within +/- 0.1 radians per second. | | | |
| MDU-3000 | Estimate Train Position, Collect Raw measurements [Communication] | Monitor Train |  |
| The Motion Detection Unit shall communicate with remote machines. | | | |

### Firmware Program: Motion Detection Unit

| Requirement Number | Feature | Use Case | Priority |
| --- | --- | --- | --- |
| Requirement Text | | | |
| MDU-1030 | Estimate Train Position, Collect Raw measurements | Monitor Train |  |
| The Motion Detection Unit shall report acceleration measurements to the users of the device. | | | |
| MDU-1050 | Estimate Train Position, Collect Raw measurements | Monitor Train |  |
| The Motion Detection Unit shall report angular velocity measurements to the users of the device. | | | |
| MDU-1060 | Estimate Train Position, Collect Raw measurements | Monitor Train |  |
| The Motion Detection Unit shall report events where the device crosses an RFID tag to the users of the device. | | | |

### Software Program: Train Navigation Service

| Requirement Number | Feature | Use Case | Priority |
| --- | --- | --- | --- |
| Requirement Text | | | |
| TNE-1000 | Report Train Position (Rollup) | Monitor Train |  |
| The Train Navigation Service shall calculate the position of a given train along the test bed. | | | |
| TNE-1010 | Report Train Position | Monitor Train |  |
| The Train Navigation Service shall calculate the position of a given train relative to a fixed point on the test bed. | | | |
| TNE-1020 | Report Train Position | Monitor Train |  |
| The Train Navigation Service shall calculate the position of a given train within a radius that allows two trains to run on a 14-inch track block. | | | |
| TNE-1021 | Report Train Position | Monitor Train |  |
| The Train Navigation Service shall calculate the position of a given train within a 6-inch radius of its actual position. | | | |
| TNE-1022 | Report Train Position | Monitor Train |  |
| The Train Navigation Service should calculate the position of a given train with a 2-inch radius of its actual position. | | | |
| TNE-1030 | Report Train Position (Rollup) | Monitor Train |  |
| The Train Navigation Service shall use measurements from the Motion Detection Unit to calculate the position of a given train. | | | |
| TNE-1031 | Report Train Position | Monitor Train |  |
| The Train Navigation Service shall calculate the acceleration of the train from the Motion Detection Unit’s acceleration measurements. | | | |
| TNE-1032 | Report Train Position | Monitor Train |  |
| The Train Navigation Service shall calculate the velocity of the train from the Motion Detection Unit’s angular velocity measurements and acceleration measurements. | | | |
| TNE-1033 | Report Train Position | Monitor Train |  |
| The Train Navigation Service shall estimate the orientation of the train from the Motion Detection Unit’s angular velocity measurements. | | | |
| TNE-1040 | Report Train Position | Monitor Train |  |
| When the Motion Detection Unit of a train crosses a track marker, the Train Navigation Service shall update the position of that train. | | | |
| TNE-1050 | Report Train Position | Monitor Train |  |
| The Train Navigation Service should estimate of the accuracy of its train position calculations. | | | |
| TNE-2000 | Report Train Position (Rollup) | Monitor Train |  |
| The Train Navigation Service shall report the location of a given train for all modes of operation of the train. | | | |
| TNE-2010 | Report Train Position | Monitor Train |  |
| The Train Navigation Service shall report the location of a given train when it is moving forward. | | | |
| TNE-2020 | Report Train Position | Monitor Train |  |
| The Train Navigation Service shall report the location of a given train when it is at rest. | | | |
| TNE-2030 | Report Train Position | Monitor Train |  |
| The Train Navigation Service shall report the location of a given train when it is in reverse. | | | |
| TNE-4000 | Control Track | Control Track Switch |  |
| The Train Navigation Service shall issue commands to the rail system switch controller to change the state of switches on the rail system. | | | |
| TNE-5000 | Report Train Position | Monitor Train |  |
| The Train Navigation Service should detect any discrepancy between its estimates of train position and know track locations (i.e. significant differences between calculated position and RFID tag listed position or incorrect RFID tag order). | | | |
| TNE-6000 | Report Train Position | Monitor Train |  |
| The Train Navigation Service should detect when the train is at rest. | | | |
| TNE-7000 | Report Train Position | Monitor Train |  |
| The Train Navigation Service should detect when the train reverses direction. | | | |
| TNE-8000 | Report Train Position | Monitor Train |  |
| The Train Navigation Service should report the time of the last position correction. | | | |
| TNE-9000 | Report Train Position | Monitor Train |  |
| The Train Navigation Service shall estimate the speed of a given train on the test bed. | | | |

### Software Program: Train Navigation Database

| Requirement Number | Feature | Use Case | Priority |
| --- | --- | --- | --- |
| Requirement Text | | | |
| TND-1000 | Report Train Position History | Monitor Train |  |
| The Train Navigation Database shall save the history of train positions reported by the Navigation Service. | | | |
| TND-1010 | Report Train Position History | Monitor Train |  |
| The Train Navigation Database shall save estimates of the position of a given train on a track reported by the Navigation Service. | | | |
| TND-1020 | Report Train Position History | Monitor Train |  |
| The Train Navigation Database shall save estimates on the speed of a given train along the track reported by the Navigation Service. | | | |
| TND-2000 | Collect Raw Measurements (Rollup) | Monitor Train |  |
| The Train Navigation Database shall save the measurements collected by the Motion Detection Unit. | | | |
| TND-2010 | Collect Raw Measurements | Monitor Train |  |
| The Train Navigation Database shall save Motion Detection Unit acceleration measurements. | | | |
| TND-2020 | Collect Raw Measurements | Monitor Train |  |
| The Train Navigation Database shall save Motion Detection Unit angular velocity measurements. | | | |
| TND-2030 | Collect Raw Measurements (Rollup) | Monitor Train |  |
| The Train Navigation Database shall save Motion Detection Unit notifications about when it crosses a track marker (RFID Tag). | | | |
| TND-2031 | Collect Raw Measurements | Monitor Train |  |
| The Train Navigation Database shall save the time that the Motion Detection Unit crosses the train marker. | | | |
| TND-2032 | Collect Raw Measurements | Monitor Train |  |
| The Train Navigation Database shall save the unique identifier for the train marker (i.e. RFID Tag ID). | | | |
| TND-3000 | Collect Track Geometry (Rollup) | Monitor Train |  |
| The Train Navigation Database shall save the railway system track geometry (track size, track shape, and connectivity between track blocks) | | | |
| TND-3010 | Collect Track Geometry | Monitor Train |  |
| The Train Navigation Database shall save the position of track markers (i.e. RFID Tags) on the test bed. | | | |
| TND-3020 | Collect Track Geometry | Monitor Train |  |
| The Train Navigation Database shall save the track block that a given track marker (i.e. RFID Tag) belongs to. | | | |
| TND-3030 | Collect Track Geometry | Monitor Train, Control Track Switch |  |
| The Train Navigation Database shall save the position of track switches on the test bed. | | | |
| TND-3060 | Collect Track Geometry | Monitor Train |  |
| The Train Navigation Database should save the orientation of the track at a given track marker’s (i.e. RFID Tag’s) position. | | | |
| TND-4000 | Collect Track Geometry | Monitor Train |  |
| The Train Navigation Database shall save which track markers (i.e. RFID tags) that are adjacent to another track marker or switch. | | | |
| TND-4011 | Collect Track Geometry | Monitor Train |  |
| The Train Navigation Database shall save the position of objects on the test bed as the distance from a fixed point on the test bed. | | | |
| TND-5000 | Report Train Position | Monitor Train |  |
| The Train Navigation Database shall save the unique identifier associated with each train that belongs to the Positive Train Control Test Bed. | | | |

### Software Program: Train Monitor Terminal GUI

| Requirement Number | Feature | Use Case | Priority |
| --- | --- | --- | --- |
| Requirement Text | | | |
| GUI-1000 | Report Train Position | Monitor Train |  |
| The Train System GUI shall display to users the last reported position of a given train on the train track by the Train Navigation Service. | | | |
| GUI-2000 | Report Train Position, Control Track | Control Track Switch, Monitor Train |  |
| The Train System GUI shall display the Positive Train Control Test Bed track. | | | |
| GUI-2020 | Report Train Position, Control Track | Control Track Switch, Monitor Train |  |
| The Train System GUI should display the position of track markers from the Train Navigation Database. | | | |
| GUI-2030 | Report Train Position, Control Track | Control Track Switch, Monitor Train |  |
| The Train System GUI shall display track switches on the test bed from the Train Navigation Database. | | | |
| GUI-3000 | Report Train Position | Monitor Train |  |
| The Train System GUI shall display the speed of trains on track as last reported by the Train Navigation Service. | | | |
| GUI-4000 | Control Track | Control Track Switch |  |
| The Train System GUI shall allow users to control track switches. | | | |
| GUI-5000 | Report Train Position History (Rollup) | Monitor Train |  |
| The Train System GUI shall display the history of movement collected for a given train. | | | |
| GUI-5010 | Report Train Position History | Monitor Train |  |
| The Train System GUI shall display estimates of train positions from the Train Navigation Database. | | | |
| GUI-6000 | Control Track (Rollup) | Control Track Switch |  |
| The Train System GUI shall display to the user the current state of switches on the rail system. | | | |
| GUI-6010 | Control Track | Control Track Switch |  |
| The Train System GUI shall send requests to the Train Navigation Service to change the state of switches on the rail system. | | | |
| GUI-7000 | Alert When Train Stopped | Monitor Train |  |
| The Train System GUI should alert when the train stops. | | | |
| GUI-8000 | Alert When Train Reverses Direction | Monitor Train |  |
| The Train System GUI should alert when the train reverses direction. | | | |
| GUI-9000 | Train Position Prediction | Monitor Train |  |
| The Train System GUI should indicate the direction that a given train will go when it crosses the next switch in its path. | | | |

## Performance Requirements

N/A

## Other Requirements

N/A

# **Appendixes**

## Analysis Models

### Class Responsibility Collaborator Cards

|  |  |
| --- | --- |
| **Class:** Motion Detection Unit | |
| **Responsibilities** | **Collaborators** |
| Collecting information on how train is moving through measurements from sensors: Accelerometer for acceleration, gyroscope for angular velocity, RFID Tag Detector to refresh the absolute position and orientation of a train based on the knowledge of the location of RFID tags spaced along the track. |  |
| Reporting when the train crosses a train marker. |  |
|  |  |
|  |  |

|  |  |
| --- | --- |
| **Class:** Train Navigation Service | |
| **Responsibilities** | **Collaborators** |
| Interpreting measurements reported by the Motion Detection Unit to estimate the train’s position. | Motion Detection Unit  Navigation Database |
| Correcting the estimate of the train’s position when the train crosses a train marker. |  |
| Controls switches along the train track. |  |
|  |  |

|  |  |
| --- | --- |
| **Class:** Navigation Database | |
| **Responsibilities** | **Collaborators** |
| Provides geometry data about track, such as the location of switches, junctions, and train markers. |  |
| Provides additional data about the train markers necessary to correct train position estimates, including the orientation of the track where the marker is placed and the section of track that the marker belongs to. |  |
| Saves sensor measurements |  |
| Saves train position estimates |  |

|  |  |
| --- | --- |
| **Class:** Train Monitor Terminal GUI | |
| **Responsibilities** | **Collaborators** |
| Displays to user the train track based on the geometry data about the track. | Train Navigation Service  Navigation Database |
| Displays to the user the location of switches along the train track. |  |
| Displays to the user the position of trains along the train track. |  |
| Reports to the system, including the Train Navigation Service, when the user requests to toggle a switch along the track. |  |