Requirements Analysis Document

STINT Project

Software Engineering

Winter 2014

University of Western Australia

Crawley, WA 6009

Preface:

This document addresses the requirements of the STINT system. The intended audience for this document are the designers and the clients of the project.

Target Audience:

Client, Developers

STINT Members (Team B):

Dean Cook
Ashwin D'Cruz
Cameron Johnson
Kieran Hannigan
Marcus Pham
Alex Tonkin

Meeting Schedule

Past Meetings

3pm 30/07/14	Initial meet-up of group
3pm 04/08/14	Initial preparation and set up
1pm 06/08/14	Client Meeting with Ted Polglaze
3pm 06/08/14	Team meeting (IO tasks/UI tasks)
3pm 11/08/14	Team meeting
3pm 18/08/14	Team meeting
3pm 20/08/14	Team meeting (RAD + organisation)

Future Meetings

Future team meetings to occur weekly at 3pm Mondays.
Time may change to 2pm after week 5.
Meetings to be held in level 2 Computer Science UNIX labs
Extra meetings to be scheduled as required

Sign-Off

This section is to note that Mr Ted Polglaze has read and agreed to the contents of this document.

Signed: _____

1.0 General Goals

The primary goal of the STINT project is to create a program to analyse various forms of in-game sensor data from hockey sports players, and to use this data to determine the start and end points of periods in which the players were active in a hockey match, called stints.

This is to replace the current process, in which stints are determined manually from graphs of the data.

The STINT project will combine various inputs, via a graphical user interface (GUI), with data from the pre-existing program, Catapult Sprint, to accurately determine these stints and remove any irrelevant data.

2.0 Current System

Currently, Catapult Sprint is used to graphically visualise the player's sensor information. An analyst must then decide, based on these graphs, which times the player was active on the field. The analyst then produces a small data file containing the stint start and finish times (there may often be multiple stints) to match the sensor data file. When the graphs are viewed again, these stint boundaries appear, allowing for more accurate analysis of that player's data in later processes (determining average speed, or game time, for example). This is because is primarily designed for scenarios where an analysis is present at the game to note stints in real time. The functionality of the proposed system will allow for post-hoc analysis.

The goal is to remove these intermediate steps performed by the analyst.

3.0 Proposed System

3.1 Overview

The proposed system is to have a user-friendly graphical interface which takes as input a number of pre-defined game periods, and a raw CSV file containing in-game data from the Catapult Sprint program. After analysing accelerometer and GPS information, it then produces a .vid file containing the stint boundary times for the later analysis back in the Catapult Sprint program. The proposed system is to effectively filter out times when a player is not on the pitch (on the bench) and times outside of game time (i.e. warm-ups, half time breaks and rest periods).

3.2 Functional Requirements

The main functional requirement of this system is to develop an analyser that:

1. reads in input match periods, and a CSV file produced by the Catapult Sprint program and chosen by the user which contains GPS sensor and accelerometer data
    a. Optionally takes as input a folder containing several of these CSV files for batch analysis.
2. analyses this data to determine appropriate stint times, and
3. returns a .vid file containing these times that is usable in Catapult Sprint for each input file.

Another functional requirement we could include (if we chose to expand the scope) would be to perform some of the later analysis in our program too. This would include determining maximum and average player speeds, total distance covered etc.

3.2.1 Value Estimate Ratio (Requirement Rankings)

Requirement	**Client Value**	**Estimated Time**	**Ratio**	**Rank**
Analyser Includes File IO, GUI, processing algorithms	$60	40 hrs	1.5	1
Grounds Manager Includes grounds library and grounds editor	$15	15 hrs	1	2
Visualisation Visual display of the analysis, and the cut-off points	$15	25 hrs	0.6	3
Reporting Display of averages, distances and general info on GUI	$10	20 hrs	0.5	4

3.3 Non-functional Requirements

The non-functional requirements include:

- An extension which allows for the creation and saving of hockey grounds based upon GPS coordinates (called the Grounds Manager). This extension will improve the usability of the core system
- A GUI to prompt the user for the correct input file and the correct hockey ground relating to the data
- A display to inform the user of the progress of the analysis

Other non-functional requirements for the extended scope system include:

- Another display for the progress of further analysis.

3.3.1 User Interface and Human Factors

The main users of this program are to be analysts for hockey teams. The users will need to be familiar with Catapult Sprint as well as simple navigation of file systems using a GUI.
Training for use of the program will not be required due to the simplicity of the usage of the program however the program will include a help section to guide users if in need.

The user is assumed to be using STINT to analyse hockey pitches only.

STINT will check for reasonable ground sizes & game period durations (error checking) during the runtime of the program, should invalid GPS or game data be input to the system.

3.3.2 Documentation

The system will have a full Javadoc detailing all of the methods and classes inside of the executable.
Additionally there is to be a help function inside the GUI that will guide users to use the software.

3.3.3 Hardware Consideration

STINT is to be used on a laptop or PC.
The processing time of the program is dependant on the CPU of the user's computer.
Memory is not an issue due to the small size of the program.

3.3.4 Performance Characteristics

The main performance aim of STINT is to analyse the data in less than 10 minutes per game, as per request from the client.

3.3.5 Error Handling and Extreme Conditions

Input errors are to be caught before processing. This includes checking the validity of GPS coordinate data and also that the CSV has the appropriate columns and amount of data.
Users will be notified of errors via a message on the GUI and prompted to check the inputs.

When the analyst inputs a folder containing several CSV input files, the system will also check that the folder structure is not beyond a reasonable limit and will confirm with the analyst before proceeding on large jobs.

Failed files will also prompt an error message.

The application exit button will be disabled when unsafe to close (in the event that data could be lost).

3.3.6 System Interfacing

For use by Catapult Sprint, all output files are to be in the same format as the output given when done manually.
Filenames are to follow the same name as the input filename, with only a change in the extension.

Input files are not to be modified in any form.

3.3.7 Quality Issues

The program is set to trap errors and not output bad .vid files.
There is to be a high percentage of accuracy for predicting start and finish times for each stint. Each stint is to be calculated within a predetermined time of manual processing.

3.3.8 System Modifications

It is possible that the STINT program could include visualisation and reporting functionality for the data once the stint times have been determined.
Additionally, the program could be useful for other sports, and so we must consider the ease of extending the program to include different size or shape pitches and different calculations on the player data.
A final feature to be added is an editor that can view the processing done by the program and further allowing users to adjust the times using the GUI.

3.3.9 Physical Environment

N/A

3.3.10 Security Issues

N/A

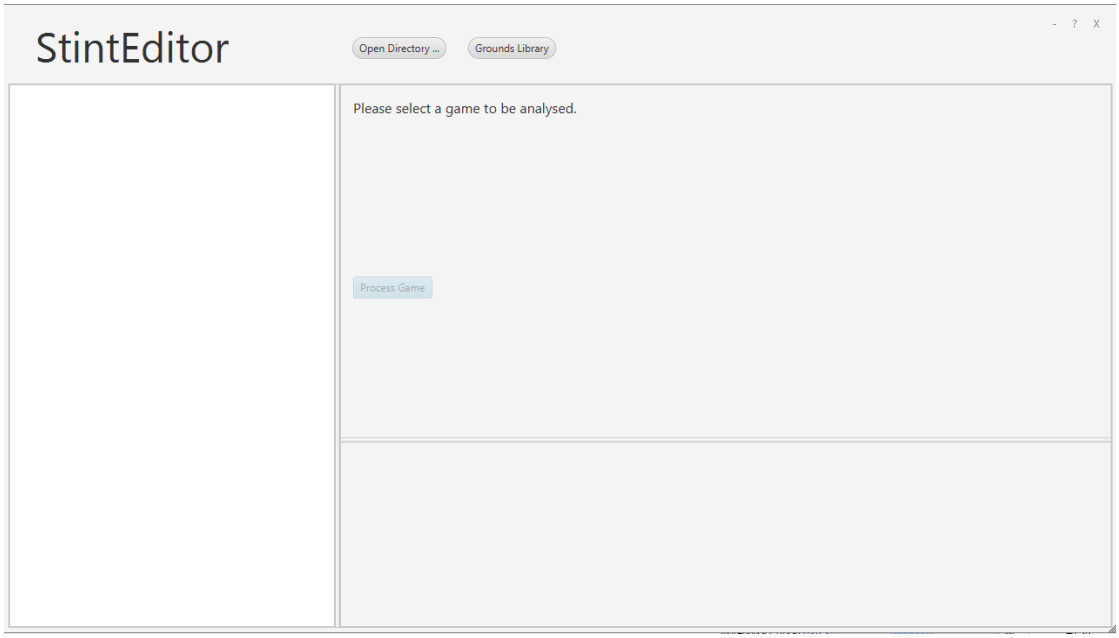
3.3.11 Resource Issues

N/A

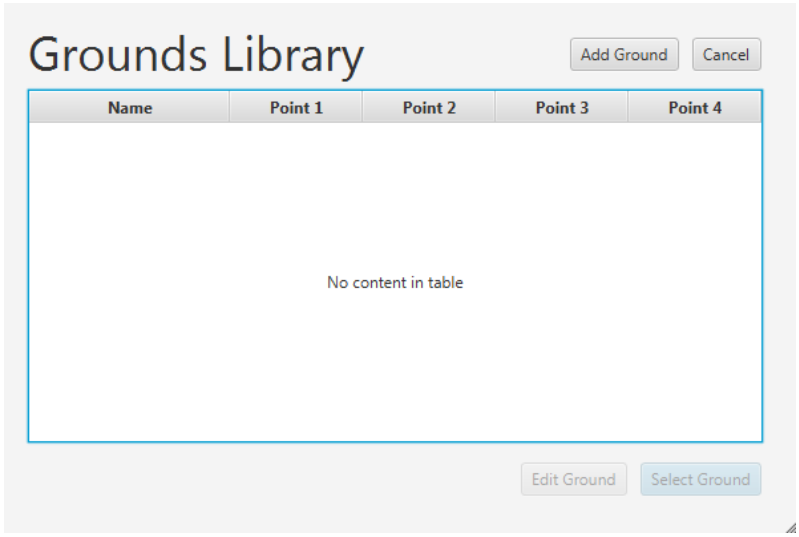
3.4 Constraints

STINT is to be developed in Java using JavaFX. JRE8 must be installed on target system. STINT is to be used alongside Catapult Sprint to shorten manual processing time for the user. The program is to be used in Windows Operating System, in particular Windows 7/8.

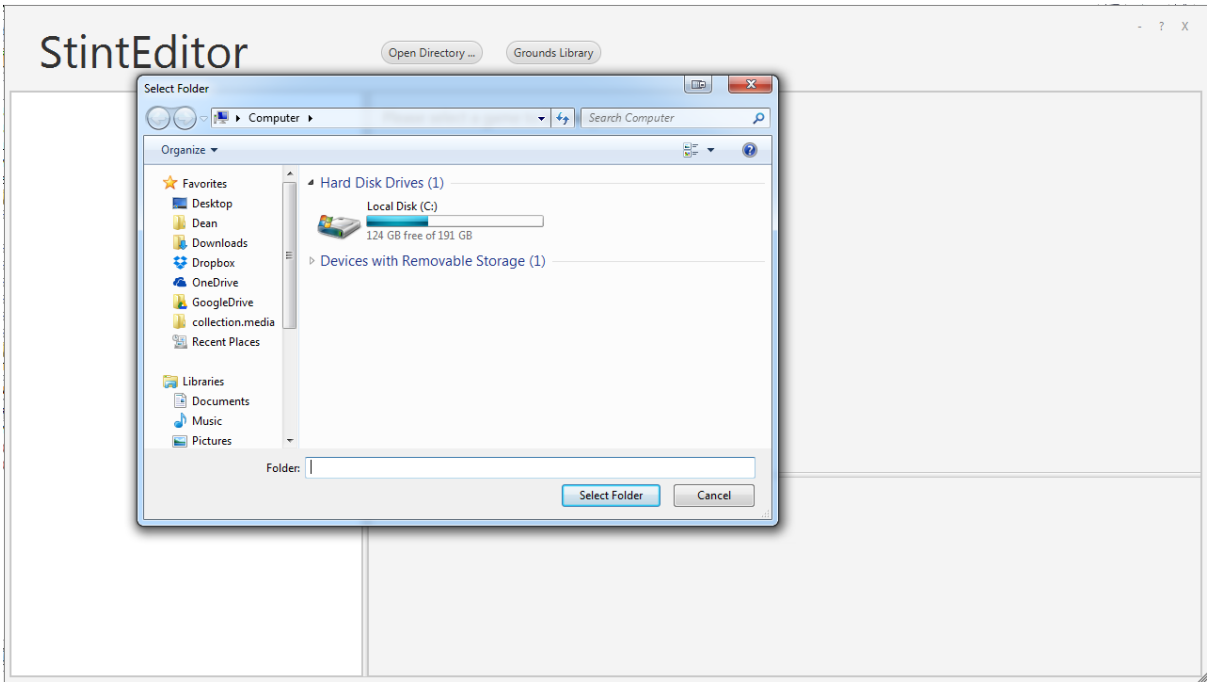
3.5 System Model



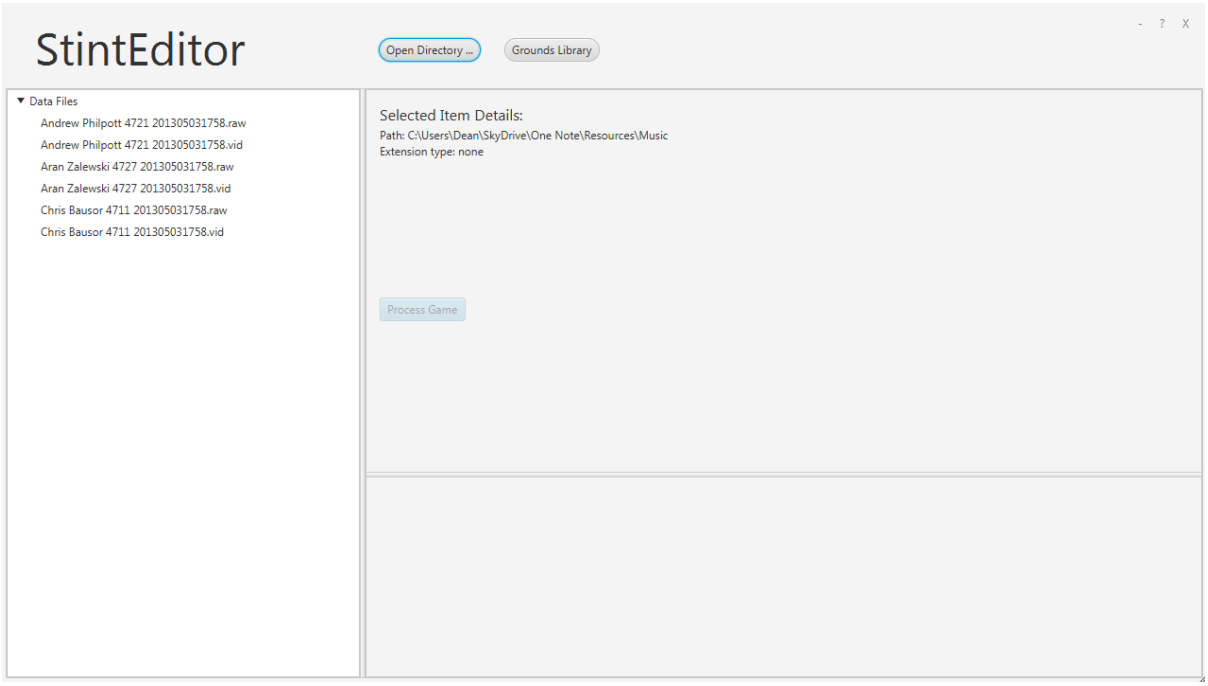
The home screen, prompting the user to select a CSV file (or folder containing several) and giving them access to the grounds library, where they can select an existing hockey field or create a new one



The grounds library editor, showing the name and 4-defining GPS coordinates of each ground stored within the library (none yet in library)

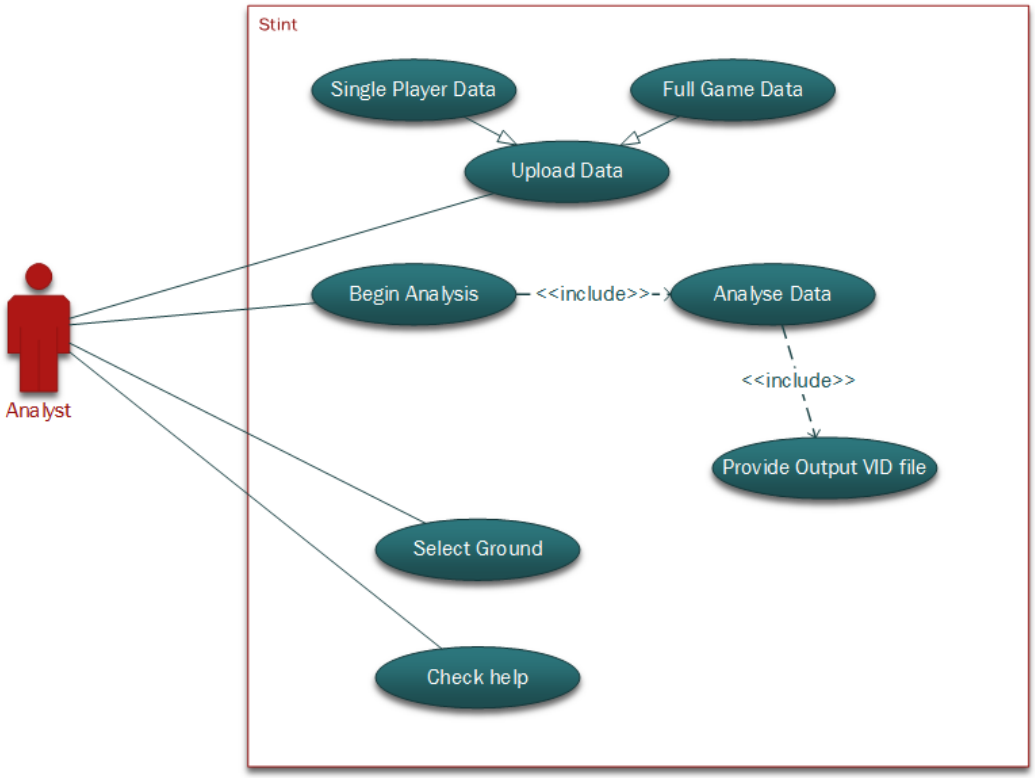


The standard windows interface for selecting folders or files.

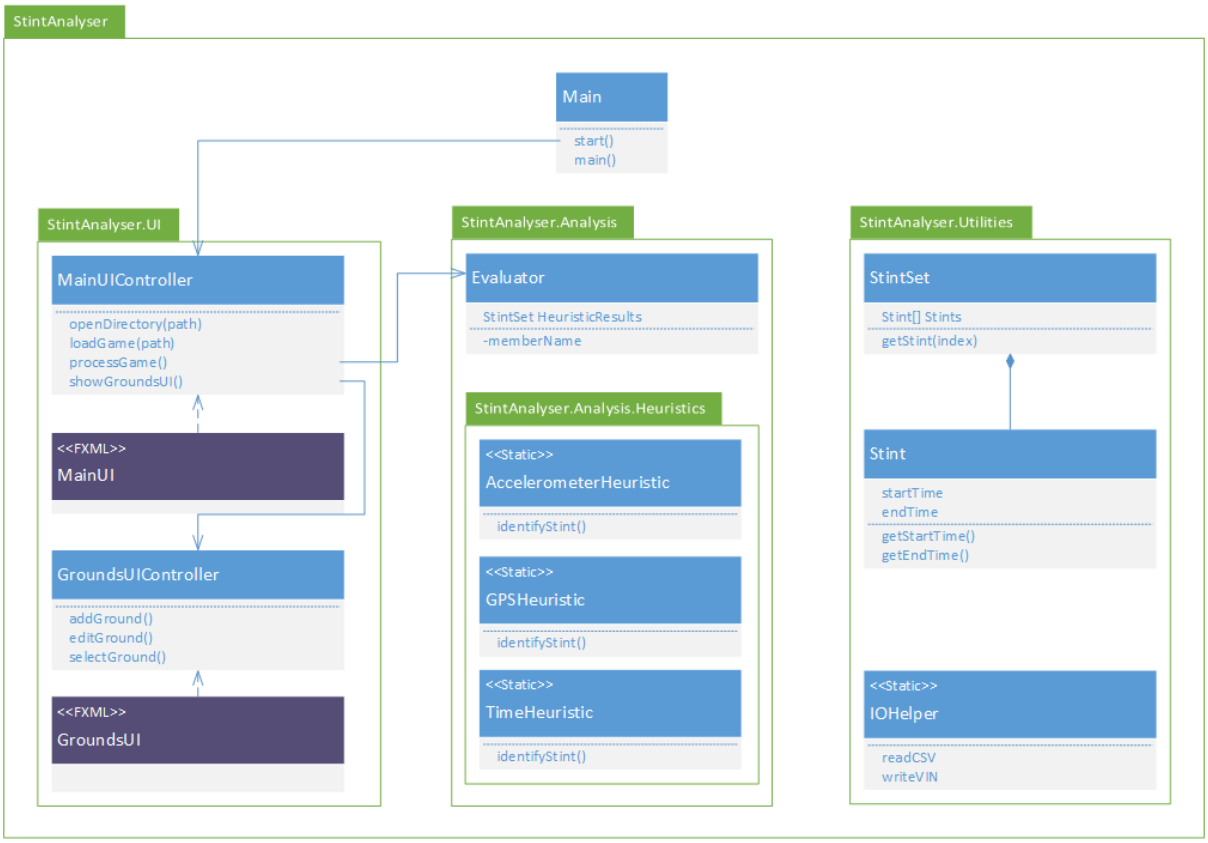


The home screen again with files imported for selection. From here, the analyst selects the "Process Game" button, and awaits the results.

3.5.2 Use Cases

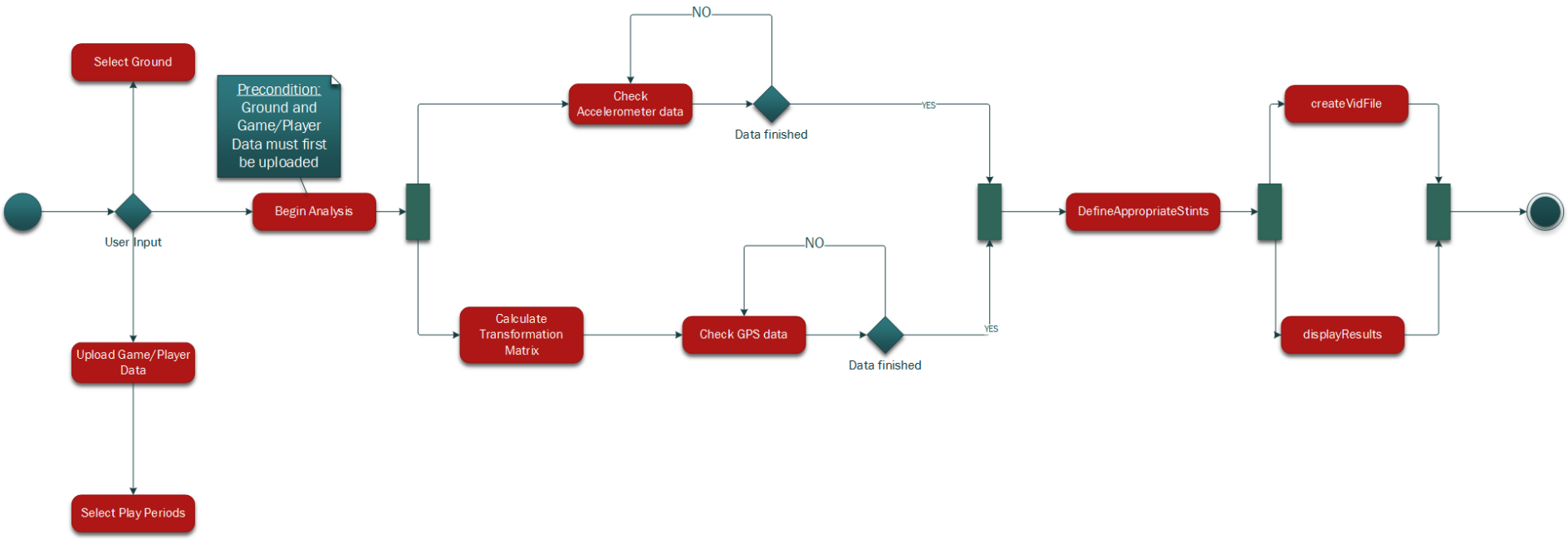


3.5.3 Object Models

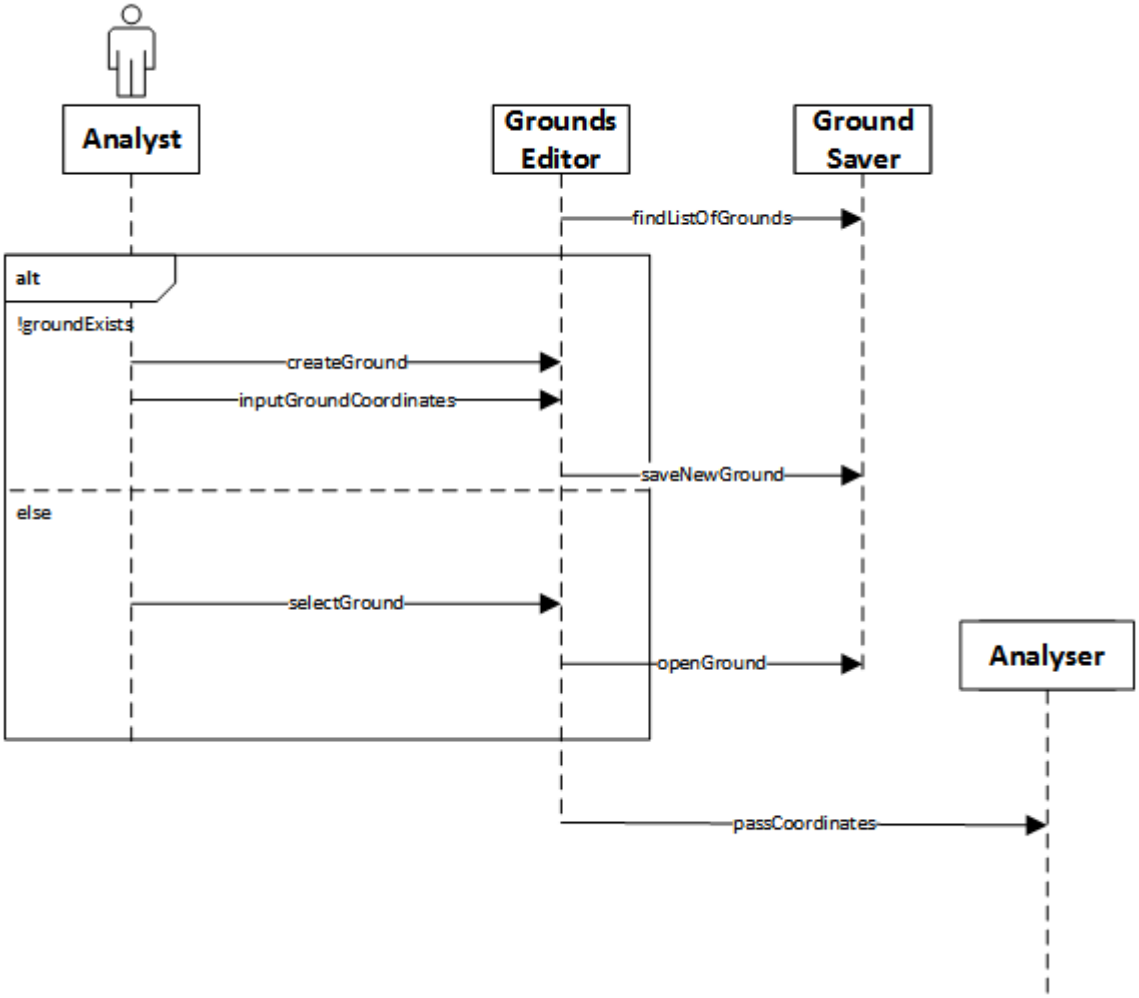


3.5.4 Dynamic Models

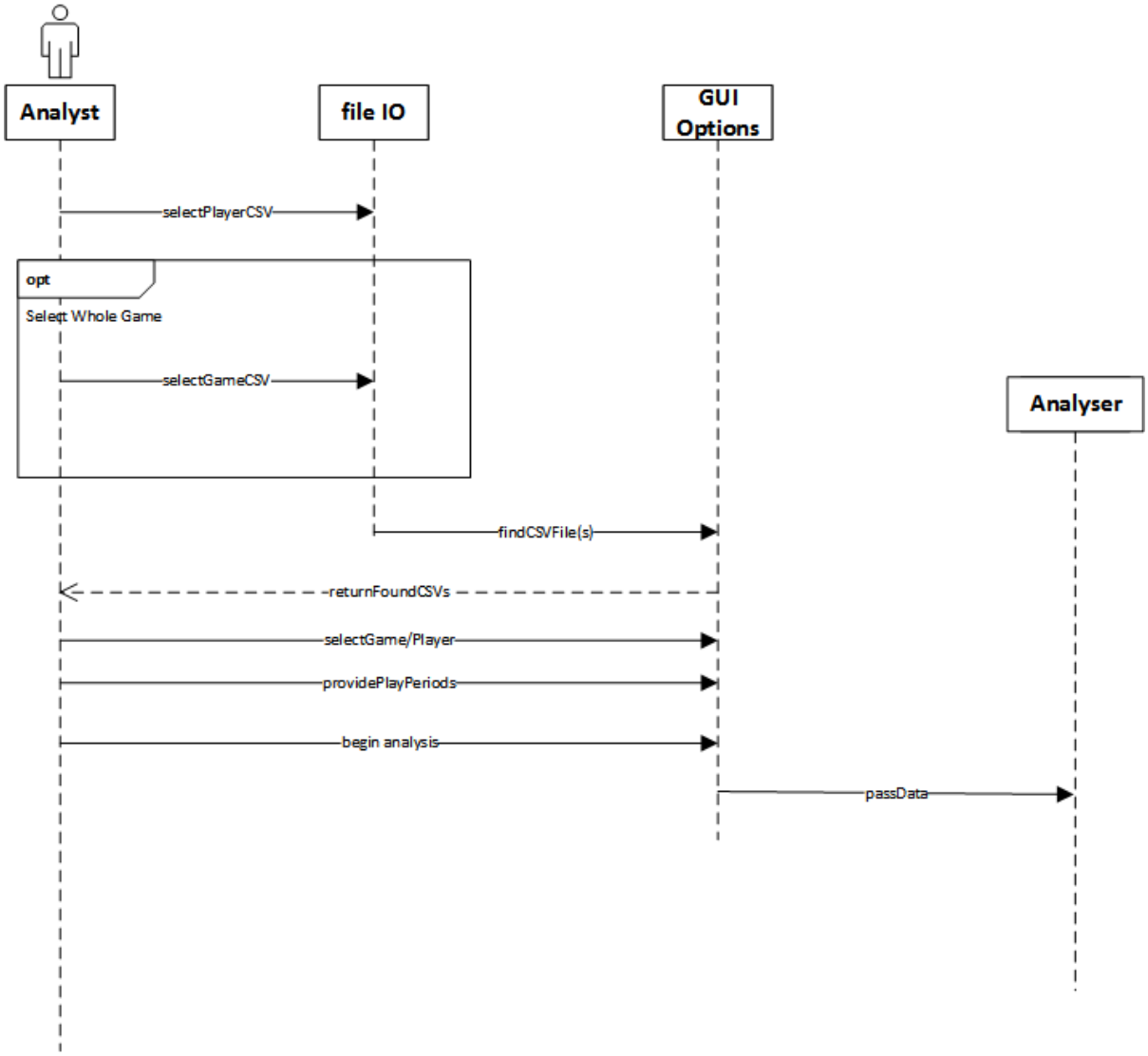
1. Activity Diagram



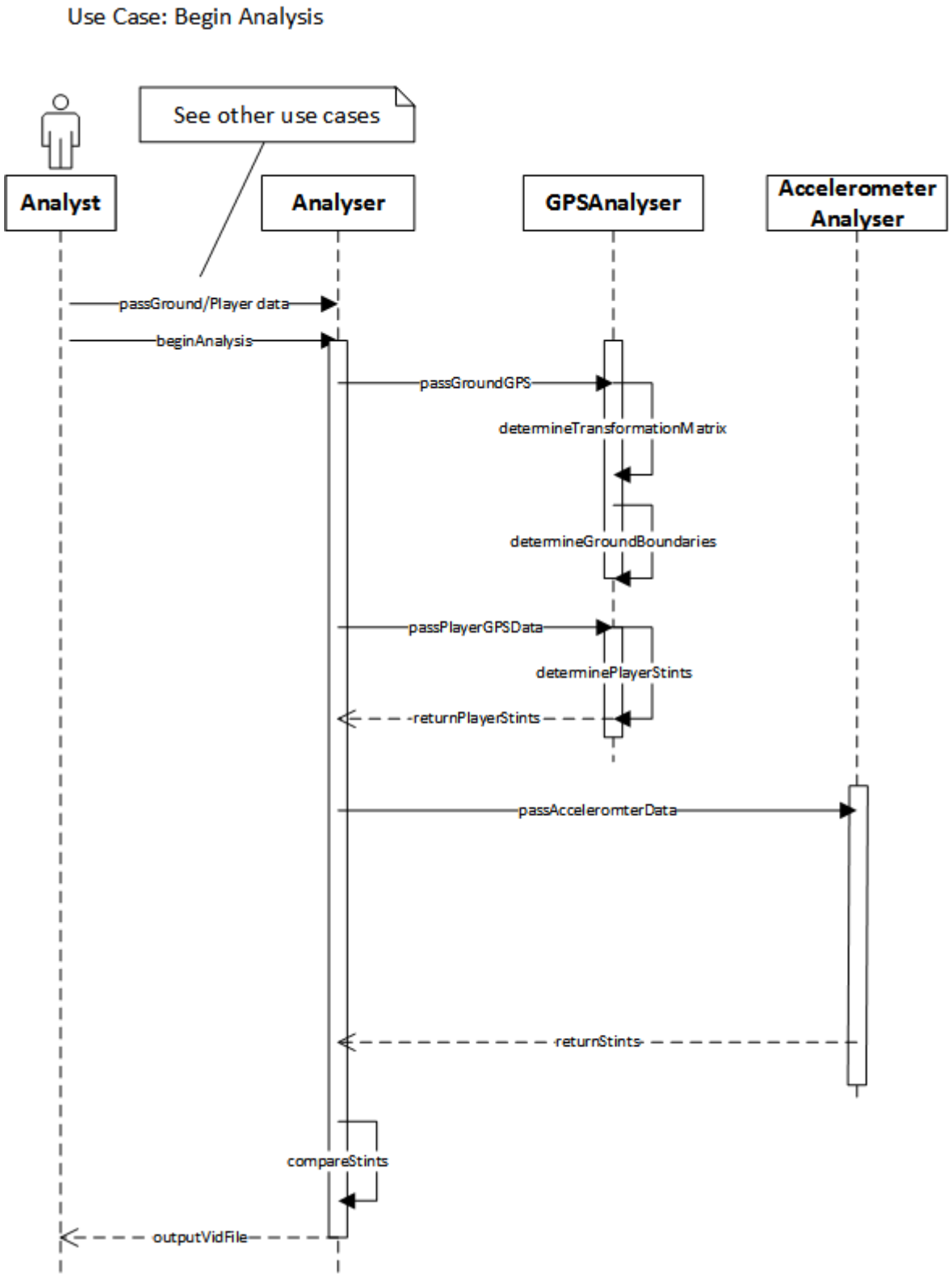
2. Ground Selection Sequence



3. Loading Data Sequence



4. Begin Analysis Sequence



4.0 Glossary

STINT: The name of the current system in development that analyses stints.

RAD: Requirements Analysis Document. This document details and clarifies all the requirements for the project.

IO: Inputs and outputs.

UI: User interface, the front-end pages that users will access.

GUI: Graphical User Interface, the interface users will access.

Catapult Sprint: A program that is currently in place for analysis of sports using various sensor data.

.csv file: Comma Separated Values file. A text file that contains data which is separated by commas to represent different types of data.

.vid file: The file type used by Catapult Sprint to allow users to specify time periods to analyse and to remove invalid times.

GPS: Global Positioning System. A system that uses satellites to accurately provide the location and time information.

Accelerometer: A device that measures acceleration of an object.

Grounds Library: A library that contains all the currently stored GPS coordinates of the sports grounds.

Grounds Editor: An interface that allows users to add, delete and modify current grounds in the grounds library.

Javadoc: A documentation generator to view the information in the Java source code.

PC: Personal Computer, a general purpose computer to be operated by an end-user.

Windows: Microsoft Windows, a graphical interface operating system.

CPU: Central Processing Units, a piece of hardware within the computer that carries out instructions of a program.

Memory: The physical devices on a computer that are able to store programs and run them.

JavaFX: A software platform to create detailed Graphical User interfaces on Java.

JRE 8: Java Runtime Environment 8, a downloadable platform required to run java applications.