Tuks RFID project proposal

# Project name: Keep calm and save the rhinos

# Project vision

“In the years of the primal course  
in the dawn of terrestrial birth  
Man mastered the mammoth and horse  
and man was lord of the earth.”

We live in a new age. Realtime information is available at our fingertips day and night. We grind mindlessly forward into a technological supernova blankly staring at the screens that control us, tell us where to go, what to do and when. Yet the age old questions are the same when we stop to think. Why am I here? What is my purpose? I know my purpose and you may choose yours.

This project has many visions and as many aspects of the same core system. Primarily, this is an asset tracking system and in this day and age, anything or anyone may be considered an asset. The primary objective is to save the rhino.

The vision is this:

1. Use the system around the university to get a feel for it. Since the university itself will own the software, the ethical issues and potential for abuse may be identified at this point.
2. Link the system on a game farm with existing hardware (previously purchased by Heinrich Strauss), dart a few Waterbuck and test the animal tracking capabilities before moving onto endangered species such as rhino or high value animals like sable.
3. During this time, the system can be used to track Tuks supply chains or internal assets (ideas?) and be made commercially available. The funds generated can be used towards the development of the project and the funding required to track, dart and tag endangered species. Options exist for the upliftment of the surrounding communities.
4. With a wide scale network of RFID nodes developing (can be used on vehicles as a mobile nodes) If any node within the system detects an illegal asset (rhino horn without a rhino?) within the network, alarms will be generated.
5. This technology can be incorporated into many things at manufacture level. Imagination.

# Project owner:

This technology has taken 10 years of personal experience to develop. I feel that it is too big for me and too dangerous in the wrong hands. While I would like to act as a guide, I would like the project owner (IP included) to be the University of Pretoria and ethical decisions should be made via a voting system through the student council.

# Project scope:

The core functional system is complete; however, there are numerous paths for evolution. The system is very advanced and may take a while for students to familiarise themselves before they begin to work with confidence. There are enough aspects that multiple groups can easily work simultaneously on the system many new requirements may be initiated.

### Firmware (C skills required)

1. Sensor firmware.   
   The sensor tag requires anti-collision code to partially randomise the transmission time in order to avoid potential packet loss.   
   The maximum number of tags in an area must be tested and optimised.  
   Additional sensor types may be added (currently temperature, optical and humidity).  
   The transceiver tag requires receive side code.
2. Repeater code.   
   The repeaters currently use a small delay (1ms) to avoid re-transmission and data bouncing. While functional, this is inefficient and the repeater protocol needs work.
3. Receiver unit (GPS/GPRS/RFID).  
   Here there is a lot of room for work, only a fraction of the available processing power is currently used so much intelligence may be added as the product evolves toward specialised use. Additions include:  
   Real-time clock implementation (this will allow unit time without the GPS).  
   History storage and optimisation – the existing history storage methodology is crude and work on the memory and the protocol will allow for smoother upload of history.  
   Predefined rules – The receiver may become more data aware and allow for the download of rules for receiver-level processing to optimise information flow.  
   Live firmware download (needs boot loader and oodles of work).

### Middleware (C#, SQL)

The middleware may be expanded for many specialised uses. I have systems in both Delphi and C#, the Delphi system (building management) is currently taking a back seat.

1. Any hardware based protocol changes will need associated middleware changes
2. The database transaction thread can use some optimisation and its threading methodology should change.
3. An integration portal needs to be built to allow for integration into existing building management, camera, biometric, passive RFID etc. systems.
4. Internal statistics and configuration interface needs development.
5. An attendance portal linked to student schedule should be built to sms/email users who are not in their required class areas (starting with computer science).

## Web interface (HTML, PHP, SQL, javascript, css, smarty templates)

My artistic ability when it comes to user interface skills leaves much to be desired. This is truly the realm of the imagination. This user interface may evolve in any direction, vehicle tracking, cold chain, mining... I will highlight the wildlife and security aspect.

1. Graphs.   
   Bar graphs, line graphs, pie graphs.  
   More graphs.  
   Comparative graphs, trending graphs.
2. Visual movement reports allowing the movement patterns of wildlife over time to be displayed on the existing Google maps functionality.
3. Integration and setup of student schedules. There are also a few architectural questions to be decided here.
4. Brush up on security – there are numerous hacker holes in the system (some intentional), these need to be examined and tidied. All the holes must be closed. Security and access to information is paramount where the survival of a tracked species is concerned.
5. Expansion of information towards specific purpose as needed.

# Architectural requirements:

### Physical network:

The middleware requires windows hosting with a fixed external IP address (this is programmed into the receivers).

The Web front end requires apache and Smarty (latest). I use the Xampp package.

An interface to a My SQL database is required.

The above easily runs on a single server. As the system expands, the aspects should be divided into different clusters.

The receivers require cellular coverage (GPS configurable) and electricity (options exist, prototype uses USB).

### Security:

This is a crucial matter. Data must be encrypted. Strict rules for the access of sensitive data must be established. Security in this matter is a true test of character – hacking is an inside job.

### User access

Users will login using email address and password through a web interface, built for Google chrome.

# Skill requirements

Learning is easy, talent is hard to find. The scope of the project is large enough to incorporate every IT skill across 5+ languages and electrical engineering for future hardware changes. All skills are needed. If you have an IT talent, a place in the project can be found.

# Project deliverables

1. Live rhinos.
2. Fast, clean, commented source code. All code should be peer reviewed before being applied to the live system.
3. I have built existing simulators and testing tools, these can be expanded and upgraded.
4. Something to be proud of. (Grammar Nazi)

# Intellectual ownership

I would like you to own the system. It is your work, it is your time and you are paying to do this. There are rules. The security of information cannot be breached. If the work that I have done will help you in the future, use it. And use it for the greater good. We are the minds that decide the systems that control us. Make sure they are good choices.

# Client commitments

I am 100% committed. Working system (hardware and source code) is available. My purpose is to leave this world a better place than I found it. I am giving my best. I hope for the same of you.