CS374 Database Management

Elisha Parslow

Kionté Brown

Nico Rode

Project Deliverable #2

**EX01** -Description of Application

See project deliverable #1 for the overview of the application and a detailed description of what the injury tracking database will have. We decided to not separate injuries and near misses into different sections of the database. There will be an option to report the type of incident so the user can then decide if they are reporting an injury, illness, behavioral issue, near miss or death. This will make it easier for the user when using the search feature and simpler for the database to find all the asked information.

Another feature we are not going to use is having multiple users. Whitworth Outdoor Recreation would not want to have to distribute multiple accounts and passwords to the users of the database, rather have one login and password that everyone can use.

**EX02** -Project Management Schedule

|  |  |  |
| --- | --- | --- |
| **Date** | **Work** | **Person(s)** |
| 11/30/2016 | Database in SQL | Kionte, Elisha, Nico |
| 12/02/2016 | HTML and Java/ website | Elisha, Nico |
| 12/01/2016 | Project Status Update | Elisha, Kionté, Nico |
| 12/02/2016 | PHP | Kionté |
| 12/02/2016 | Written Queries | Nico |

**EX03** -Logical Diagram

Our data model will satisfy the needs of our application because the UML addresses every portion of a standard outdoor recreation incident report form. It also adds more information for the opportunity to have more recorded data. There will be many true/ false data types as well as some text types for more information on the incident.

Alternative designs included creating a separate diagram for each type of incident, meaning there would be several UMLs to satisfy the specifics of an injury, near miss, death, or behavior incident. We decided this would be too complex and counterintuitive for the simplicity of what the project is meant to do. Instead, we will just allow the search portion of the program to select if true for one of the types of incidents to pull up all of its documented information. We also decided against creating an E-R diagram because it is a little less organized than what we would like.

**UML can be found in Elisha’s drive at**: \\cs1\2018\eparslow18\CS374-1\GitHub DM\FINAL PROJECT\IncidentTracking\IncidentTracking

**EX04** -Queries Required

1. Pull all incident reports within the last year

1. Looks at Incident\_Information to find the date parameter, from there the query would return all tables and information for which the incident\_ID’s match, i.e. Type\_Of\_Illness, Type\_Of\_Injury, Injured\_Person\_Personal\_Info, etc.
2. This broad search is used primarily for tracking recent incident data and provides the user with all relevant data within their parameters.

2. Find all Incidents for which John Doe was the trip leader for

1. Looks at Incident\_Information to find the Trip\_Leaders element, matches with the name John Doe, and then returns all relevant incident reports for which John Doe was leading.
2. This query is useful for finding specific incidents of which you are interested in who was leading the trip. Might find that some leaders tend to have higher incident rates than others.

3. Which incidents resulted in death

1. Isolates incidents that return true for Death in the Typ\_Of\_Incident table and crosschecks with the Incident\_ID key.
2. Useful in determining severe incidents that resulted in death, hopefully none.

4. Which incidents resulted in a head injury for one of the participants

1. Searches through the Anatomical\_Loc\_of\_Injury table to find reports that have indicated an injury of the head. Uses the Incident\_ID key to make sure that the right incident is reported.
2. Might be used to see which anatomical locations are the most common for reported injuries. Finding reports that have head injuries might specifically be useful to check up on the patients at a later time.

5. What injuries have occurred on climbing trips?

1. Selects all the reports that return true for climbing in the Type\_Of\_Trip table. Uses the Incident\_ID foreign key to authenticate the report.
2. Useful in determining which injuries have occurred on previous climbing trips, perhaps to avoid the locations that produce the most injuries.

6. Edit the description of the incident after it has already been reported.

1. Searches the incident reports by name, id, or other fields, and then replaces the old incident description with a new one that the user enters.
2. Could be used to modify and edit poorly written descriptions, or add information that was previously missing.

7. Add new incident

1. Creates new tables and populates the information based on the data that the user inputted.
2. Foundation of the database, users will populate the system with standardized incident reports in order to have them organized and secure in one place for future lookups and record keeping.