3 Database

This part of the design has a clean interface with no further calls to other modules, implying that it could be constructed as a single unit. We have to let a third party provide some small part of the prototype since time is at a premium, for which the database would be the logical choice.

The database must comply with the following:

- It has to be stored as a text file, with each line corresponding to a record in the database.
- The fields in it should be separated by a single character (;).
- If the separation character appears in one of the fields it shall be escaped with \.
- All fields are texts and may have variable size
- The first field shall be the unique (primary) key
- The fields are:
  key;e-mail;url;current-status;last-modified-time;last-checked-time;timeout-time
- It must be possible to add more fields later and such fields must be added at the end of the record
- Locking of the database must be performed by the database itself
- The implementation language shall be C.

Datatype definitions	
char *DB;	Name of database-file
char *key;	Unique key for a record
int field;	Field-number in range 0..n
char *data;	Data for a field
char *record;	A record with only the unique key missing
Operations	
key getFirst(DB);	Get first record in database or empty string otherwise
key getNext(DB, key);	Get the next record after the record with given key (1)
data lookup(DB, key, field);	Get the information in a field of a record
update(DB, key, field, data);	Update one field in a specified record
key insert(DB, record);	Adds a new record to the database (2)
delete(DB, key);	Remove a record from the database if it exists
(1)	First record if invalid key is given and empty string if database is empty or at end of database
(2)	No checking for doubles shall be performed