

Exercise Sheet 8

Exercise 1 (Google App Engine – GAE)

Statement	true	false
The GAE implements Infrastructure as a Service.		
The GAE implements Platform as a Service.		
The GAE implements Software as a Service.		
The GAE supports all programming languages.		
Applications inside the GAE can use different infrastructure and storage services.		
Free re-implementations of the GAE exist.		
The GAE supports authentication/authorization via Google accounts.		
Objects, stored in the Datastore are erased automatically after 24 h.		
Datastore is a persistent storage service, implemented as a key/value database.		
Datastore provides a query language, which is similar to the SQL (Structured Query Language).		
Memcache is a persistent storage service.		
Memcache uses tape storage to store the data.		
Memcache is a high-performance storage service for temporary data.		
Memcache uses physical main memory to store the data.		
Objects in the Datastore and Memcache can be of any size.		
Applications in the GAE can send and receive emails with any sender address.		
Applications in the GAE can send and receive XMPP messages.		
Applications in the GAE have read and write access to the file system of the physical servers.		
Only the ports 80, 443, 4443, 8080-8089, 8188, 8444 and 8990 can be used by Applications in the GAE to communicate with other web services.		
Applications in the GAE can communicate with other web services via all ports.		
The GAE provides support for Python 3.		
Customers of the GAE can upload and run multiple versions of each one of their applications.		
Each customer of the GAE can have an infinite number of applications.		
Each customer of the GAE can have up to 10 applications.		
Application names must be unique inside the GAE namespace.		

Exercise 2 (Sudoku in the Google App Engine)

Develop a web application for the Google App Engine, which allows its users to play Sudoku. The application can be implemented in any programming language, the App Engine supports. The application should meet the following requirements:

- A Sudoku with 9x9 fields and some given numbers is presented in the browser. Every time, the web application is queried, it generates (calculates) a new, solvable Sudoku. If this is too difficult for you to implement, for the sake of simplicity, you can just implement („hard-code“) 5 Sudoku fields and every time, the web application is queried (reloaded), one of the 5 already available Sudokus is randomly chosen.
- Users can insert numbers into empty fields.
- Users can erase numbers, which have been inserted prior by themselves.
- Every time, a number is inserted into an empty field, the web application checks, whether the field is still in consistent state.
- Every time, a number is erased, the web application checks, whether the field is still in consistent state.
- If the Sudoku is solved correctly, the web application recognizes this and reacts to this situation in an appropriate way.

A helpful collection of the mathematical fundamentals of Sudoku is [1].

[1] http://en.wikipedia.org/wiki/Mathematics_of_Sudoku

Your tasks:

1. Implement the Sudoku web application for the Google App Engine.
2. Upload the web application into the App Engine.
3. Present your implemented application live during the exercise session.