## Introduction

The purpose of biotechnology is to use micro-organisms like moulds and bacteria for the production of chemical substances. These chemical substances are used as energy (biofuels), in products (bioplastics), in food and in medicine. Not only can these micro-organisms produce, they can also be used to break down chemical substances, for example in the purification of water.1

Using molecular biology, these micro-organisms can be altered to add, change or remove functionality. This is done by mutating the DNA of the organism. Synthetic biology is the science in which biology and engineering are combined to create new biological functionality. 2 The engineering part is made easier by using biobricks3. These “bricks” are simple genetic circuits which provide a basic functionality (for example the imitation of an OR or AND gate) and share a common interface. By combining these biobricks, a new biological systems can be engineered.

In its abstract form, this system can be visualized as a logical circuit, but in the organism the circuit corresponds with a group of molecules (proteins, genes, RNA) that react with each other. How (and if) they will react depends on many different elements, like the concentration and reaction speed of a protein. This reaction can be simulated using a (heavily) simplified model. 1

A logical circuit of biobricks can be defined using the System Biology Markup Language (SBML). SBML is a free and open interchange format for computer models of biological processes. 4 It is a widely supported format that continues to evolve and expand.

In this report, we will describe the design of a visual modeling environment for synthetic biology, in which biotechnologists can design, simulate and validate a logical circuit built using biobricks. We will clarify our proposed system using a list of requirements (2.2-2.4), a few analysis models (2.5), a business object model (2.6), a few dynamic models (2.7) and a preliminary drawing of the interface (2.8). In chapter 3 we will give a planning of the following of the project.

1. Programming Life – Contextproject assignment (Dick de Ridder, Marcel Reinders)
2. Programming Life – Contextproject kickoff (Dick de Ridder, Marcel Reinders)
3. http://biobricks.org/
4. http://sbml.org/Main\_Page

## Overview

In this document, we will provide an analysis and design proposal for a visual modeling environment for synthetic biology, in which biotechnologists can design, simulate and validate a logical circuit built using biobricks.

We have made up a list of requirements which answer the following questions. What does the application have to do? What doesn't belong in the basic functionality? What kind of programming language will we use to develop this program and when is it due? First we will go into detail on functional requirements (2.2), so what does the program do? Secondly, we will discuss non-functional requirements (2.3). These answer questions like: What programming language will we use, but also how we will increase the usability of the application. Finally we will have a look at constraints (2.4).

Following the requirements we will specify a few use case models (2.5). These models describe specific scenarios of the program, what steps the user has to take to reach a goal and how the system should react to these steps. The main scenarios we will specify are loading/saving, modeling and simulating. Dynamic models such as sequence diagrams and activity diagrams will visualize the steps and interaction of the user and system in chapter 2.7.

Our business object model (2.5) will clarify the key concepts and their roles of our application. It will give a simplified overview of how proteins, biobricks and the System Biology Markup Language (SBML) relate to each other.

Last but not least, we will show a preliminary drawing of our interface (2.8) and explain why we have chosen for this interface and how it will work. Chapter 3 contains a planning for the rest of the project. This schedule is built up around the deadlines for the other documents and has a preliminary planning for the implementation fase.