

Software Engineering Project



TEZPUR UNIVERSITY

Name: Hirok Jyoti Nath

Roll: CSB20094

Dept: CSE, 5th semester

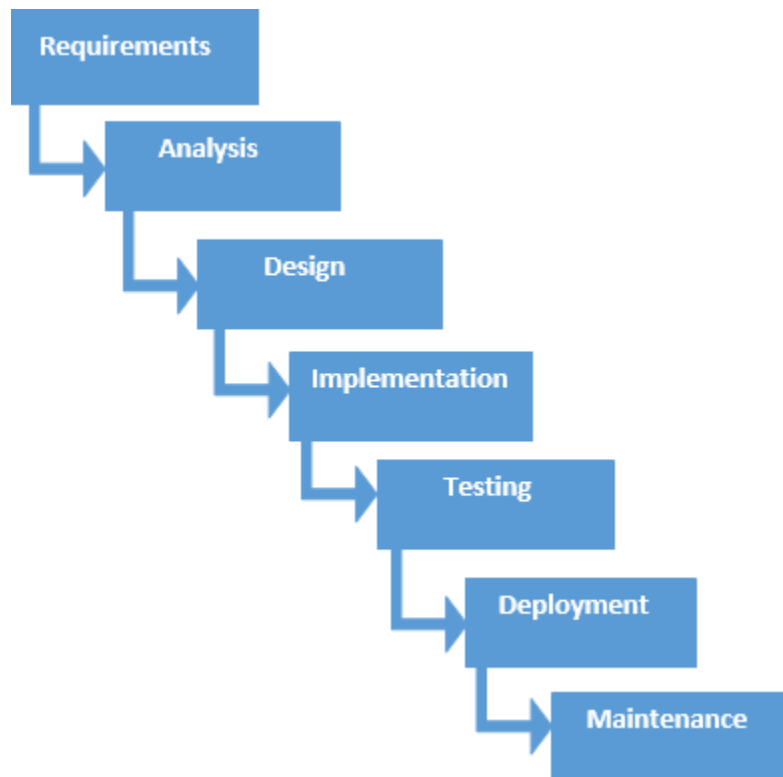
Purpose

Purpose is to prepare a database which stores information about DNA sequence and compute G+C%.

Waterfall Model

The Waterfall approach was the first SDLC Model to be used widely in Software Engineering to ensure success of the project. In "The Waterfall" approach, the whole process of software development is divided into separate phases. In this Waterfall model, typically, the outcome of one phase acts as the input for the next phase sequentially.

The following illustration is a representation of the different phases of the Waterfall Model.



Requirement analysis

All possible requirements of the system to be developed are captured in this phase and documented in a software requirement specification document.

Software Requirements Specification Document (SRS)

1. Introduction

a. Purpose :

The purpose of this document is to present a detailed description of the DNA analysis system. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli.

b. Scope of Project

This software system will be a DNA analysis system for personal usage. This system will be designed to maximize productivity by providing tools to assist in automating the process, which would otherwise have to be performed manually. By maximizing the work efficiency and production the system will meet the user needs while remaining easy to understand and use.

2. Overall Description

a. Product Perspective :

The product is being achieved by using modules from Tkinter, Python compiled code and a MySQL Database to store all the necessary details.

b. Product Function :

The product provides for the acceptance of a particular gene sequence file and checking for the validity of the nucleotide sequence present in the file. If found valid, it then calculates the length of the sequence, the G+C% and converts it into an AA sequence and stores these details into a database from where the information can be retrieved based on the queries.

c. Operating Environment :

The product is operable on windows and linux environments.

d. User Environment :

The product is restricted to desktop and laptop use.

3. Requirement Specifications

a. Functional Requirements

- i. Check input for valid gene file and save it in database
- ii. Generate AA-sequence from given Nucleotide Sequence
- iii. Find G+C percentage and gene length for Nucleotide sequence
- iv. Update details of gene according to information provided
- v. Search details of gene from provided gene info

b. Non-Functional Requirements

- i. Create database for gene details
- ii. Manage database to maintain the performance of system

4. External Interface Environment

a. User Interface

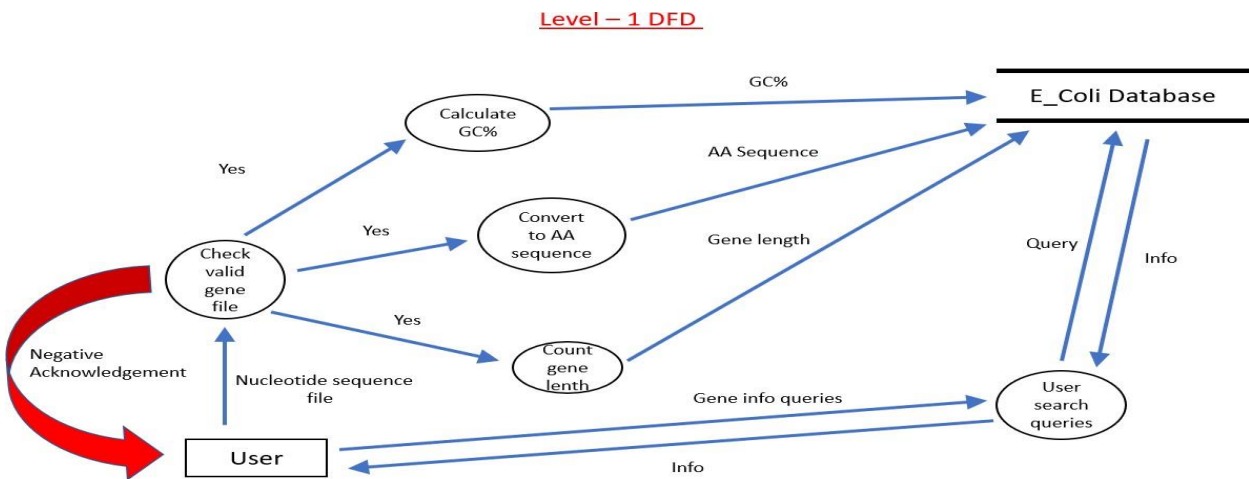
The user interface encounters a drag and drop implementation where the user has to drop the nucleotide file for its valid checking.

b. Software interface

The software has been developed using python, Tkinter and MySQL database and interlinked to communicate with each other.

System Design

Data Flow Diagram (DFD)



- User gives nucleotide sequence file
- The interface accepts this file
- The program check the validity of the file
- If file is not valid the user gets negative acknowledgement
- Else the program calculates GC% , AA sequence and Gene length
- GC% , AA sequence and Gene length are stored in Database
- After that user perform some query related to nucleotide sequence and the Database retrieves information related to the query and gives back to the user

Implementation

- Created interface through HTML and CSS
- Coded in JavaScript to check for validity of the file and convert nucleotide sequence into AA sequence, calculated GC% , Gene Length
- Connected MongoDB Database via Express.js connector module
- Push Nucleotide sequence and its corresponding AA sequence, calculated GC% , Gene Length to Database
- Queries can be retrieved from the MongoDB Database

Database :

SLno.	Info	NucleotideSeq	Size(aa)	G+C%	ProteinSeq	Remark
1	>gi 545778205 gb U00096.3 :190 Escherichia coli str. K-12 substr. MG1655, complete genome	ATGAAACGCATTAGCACCAC	21	51.515151515151516	MKRISTTTTTTITGNGAG	Valid seq
2	>gi 545778205 gb U00096.3 :337 Escherichia coli str. K-12 substr. MG1655, complete genome	ATGCGAGTGTGAAGTTCGC	820	53.06536743808363	MRVLKFGGTSVANAERFLRV	Valid seq
3	>gi 545778205 gb U00096.3 :280 Escherichia coli str. K-12 substr. MG1655, complete genome	ATGGTTAAAGTTTATGCCCC	310	56.27009646302251	MVKVYAPASSANMSVGFDT	Valid seq
4	>gi 545778205 gb U00096.3 :373 Escherichia coli str. K-12 substr. MG1655, complete genome	ATGAAACTCTACAATCTGAA	428	52.836052836052836	MKLYNLKDHNEQVSFAQAVT	Valid seq
5	>gi 545778205 gb U00096.3 :523 Escherichia coli str. K-12 substr. MG1655, complete genome	GTGAAAAAGATGCAATCTAT	98	53.87205387205387	VKKMQSIVLALSILVLVAPMA	Valid seq
6	>gi 545778205 gb U00096.3 :c64 Escherichia coli str. K-12 substr. MG1655, complete genome	ATGCTGATTCTTATTCACCT	258	49.67824967824968	MLILISPAKTLDYQSPLTTRY	Valid seq
7	>gi 545778205 gb U00096.3 :c79 Escherichia coli str. K-12 substr. MG1655, complete genome	ATGCCAGATTTTTCTCCTTC	476	53.319357092942	MPDFFSFINSVLWGSVMIIYL	Valid seq
	>gi 545778205 gb U00096.3 :823					

Unit Testing

We have 3 modules

- To create interface
- To check validity of file
- To connect Database

First we start from the interface testing and then we provide our interface with a file (E Coli Data) , and we check the validity of the file using source code, where we check for codon sequence, starting and ending codons and length of the nucleotide sequence, After that we convert the given nucleotide sequence to AA sequence, GC%.

The above are then pushed into the Database, where we can perform our query and get the desired result.

System Testing

We combine all the modules then perform system testing . If it's correct then we deliver our software to the user .

Maintenance

If a user wants changes in interface or some additional feature requirement, then we make necessary changes to our software.

Conclusion

From the above project we have gained knowledge regarding the development of a software and the various stages of development it passes through. We also learnt about how to connect the different modules of code and interlink it to develop the required software. Even though the software is small and is restricted to only personal use in a desktop environment, still we got to learn the different aspects of software development from scratch and its implementation in the real world.