
Text2SpeechEditor

Sprint No 1 Report

ΕΡΟΝΥΜΟΥΣ

ΙΩΑΝΝΗΣ ΧΟΥΛΙΑΡΑΣ , 2631

ΠΕΤΡΟΣ ΖΑΜΠΕΛΗΣ, 2970

ΔΗΜΗΤΡΙΟΣ ΖΩΓΑΝΑΣ, 2977

ΜΥΥ803 PROJECT - 2020

VERSIONS HISTORY

Date	Version	Description	Author
<i>1 may 2020</i>	<i>1.0</i>	<i>Last version of the requirements.</i>	<i>Team - Eponymous</i>

1. Introduction

This document defines the basic requirements for Text2SpeechEditor, an application for writing documents and transforming them to audio.

1.1. Purpose

The objective of this project is to develop a simple editor that allows preparing text documents and transforming them to audio. Such an application can be of much help for people who have speaking problem. Moreover, it could be a useful tool for preparing presentations that are automatically played without a physical speaker. The application consists of a graphical user interface, a back-end that enables the text editing process and the part that transforms text to speech, which will be based on external text to speech libraries.

1.2. Document Structure

The rest of this document is structured as follows. Section 2 describes out Scrum team and specifies the this Sprint's backlog. Section 3 specifies the main design concepts for this release of the project.

2. Scrum team and Sprint Backlog

<For the user stories included in this release specify below corresponding tests using a typical tabular form.>

2.1. Scrum team

Product Owner	<i>Apostolos Zarras</i>
Scrum Master	<i>Ioannis Chouliaras 2631</i>
Development Team	<i>Ioannis Chouliaras 2631, Petros Zampelis 2970, Dimitrios Zoganas 2977</i>

2.2. Sprint Backlog

- [US1]: As a user , I want to create a new empty document , by giving its title and author. The application should automatically record the creation date.*
- [US2]: As a user, I want to edit the contents of the document, via the application's user interface*

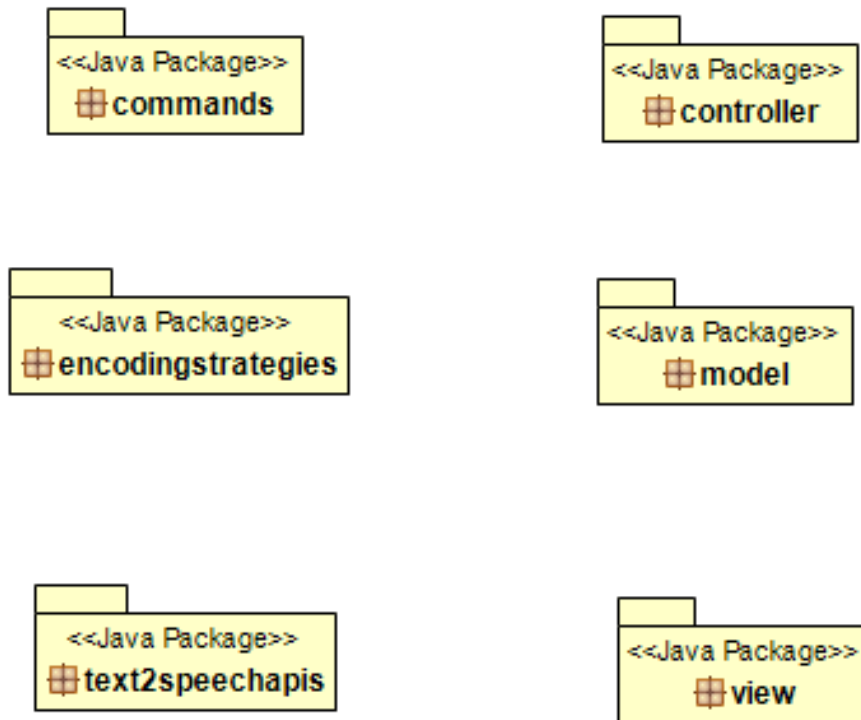
- *[US3]: As a user, I want to save the contents of the document to disk by providing a particular filename. The application should automatically record the save date.*
- *[US4]: As a user, I want to open the contents of an existing document from disk by providing a particular file path, or by browsing the file system folders.*
- *[US5]: As a user, I want to transform the contents of the document to speech.*
- *[US6]: As a user, I want to select a line and transform it to speech.*
- *[US7]: As a user I want to transform the contents of the document to speech in reverse, i.e. play the last word of the last line first and so on.*
- *[US8]: As a user I want to select a line and transform it to speech in reverse, i.e. and play the last word first and so on.*
- *[US9]: As a user I want to encode the contents of the document and then transform them to speech.*
- *[US10]: As a user I want to select a line, encode it and transform it to speech.*
- *[US11]: As a user I want to tune the text encoding technique. In particular the application should support at least the following encoding strategies: AtBash and Rot13*
- *[US12]: As a user I want to be able to tune the audio parameters, i.e., the volume, the speech rate and the pitch.*
- *[US13]: As a user I want to be able to store a sequence of actions/commands (e.g. open file, edit contents, play contents, play line, save file) that I have performed in main memory and re-execute them multiple times.*

-

3. Design

3.1. Architecture

UML package diagram:



Commands Package: contains all the classes for the actions/commands

Controller package: contains the class where the GUI(view) and back-end(model) communicates.

Model package: contains the back end of the application.

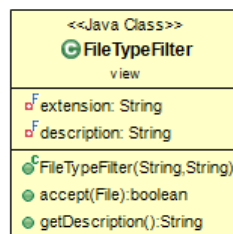
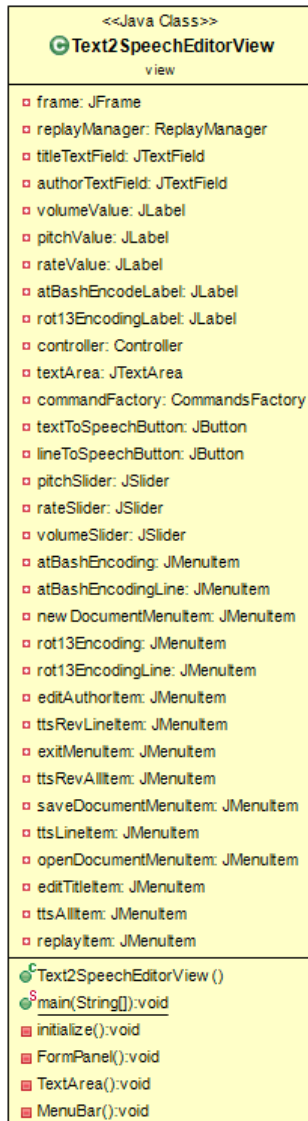
EncodingStrategies package: contains the interface and the classes for the two different encoding strategies.

Text2SpeechAPIS package: contains the interface and the classes for the transformation of the text to speech (freeTTS library)

View package: contains two classes. The GUI of the application(collaborating with Controller class) and the fileTypeFilter.

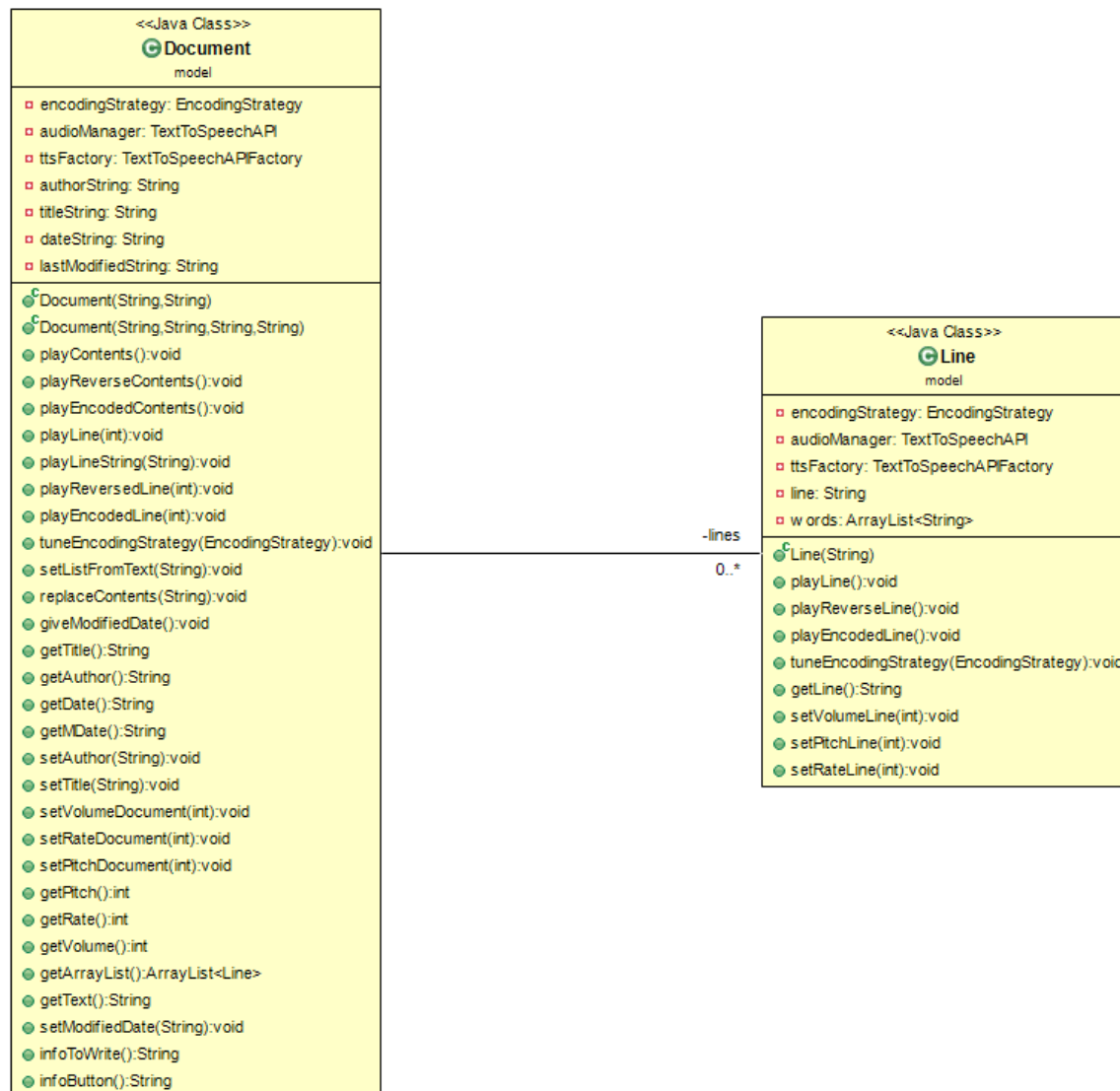
3.2. Design

UML class diagrams:



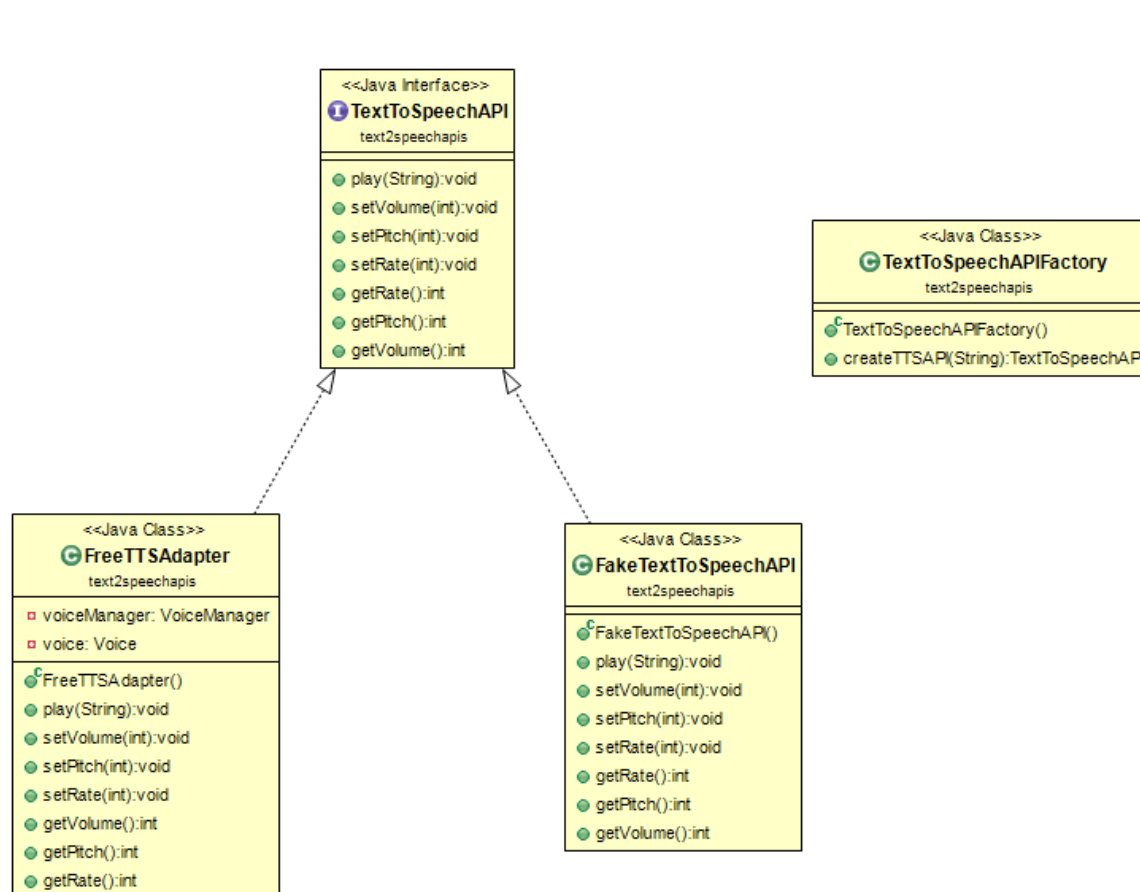
Class Name: FileTypeFilter	
Responsibilities: <ul style="list-style-type: none">Accepts the correct file.Extension .ttsOpening and saving file to disk	Collaborations: <ul style="list-style-type: none">openDocumentsaveDocument

Class Name: Text2SpeechEditorView	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> ■ <i>Whole GUI</i> ■ <i>Interaction with user</i> 	<ul style="list-style-type: none"> ■ <i>Controller package</i>



Class Name: Line	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> ■ <i>Keep the words of one line</i> ■ <i>Play the words, set pitch, volume, rate</i> ■ <i>Encode or play in reverse</i> 	<ul style="list-style-type: none"> ■ <i>EncodingStrategy</i> ■ <i>TextToSpeechAPI</i> ■ <i>TextToSpeechAPIFactory</i>

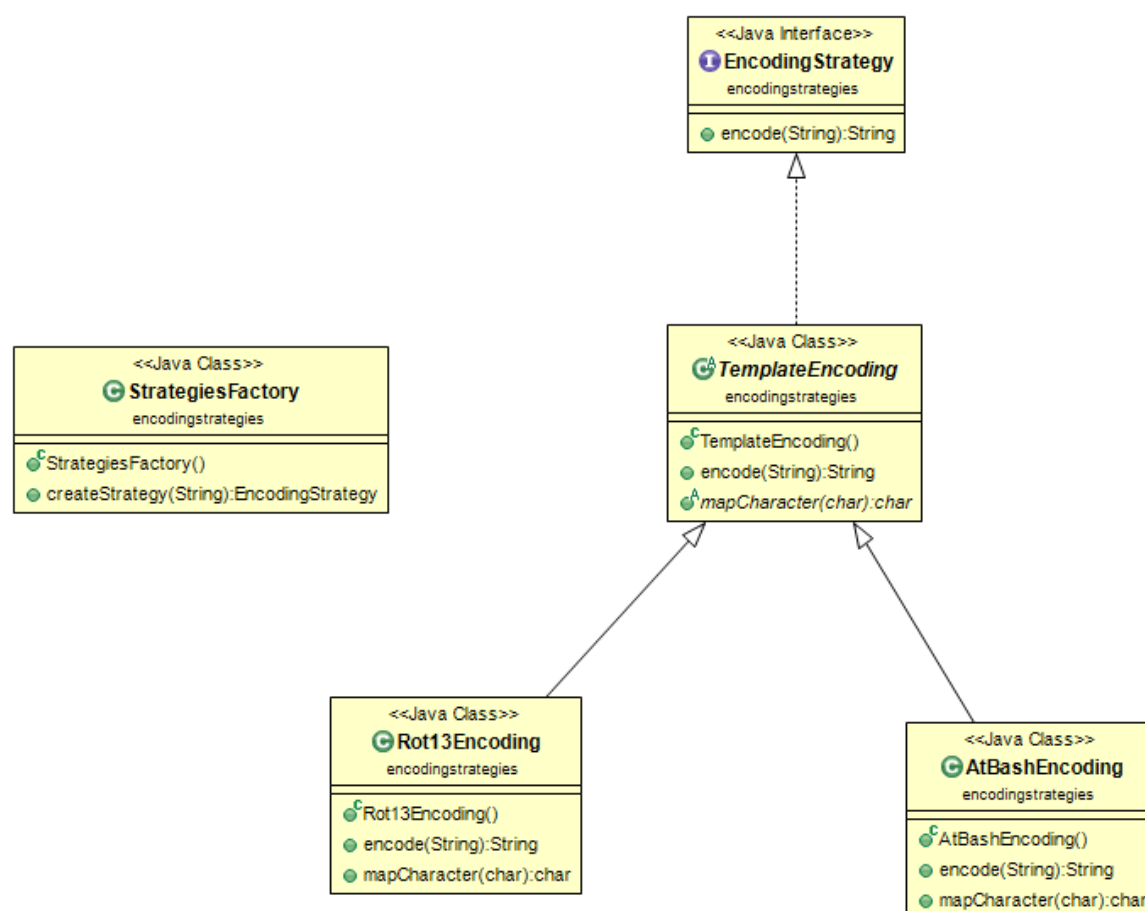
Class Name: Document	
Responsibilities: <ul style="list-style-type: none"> Keeps Document info Contains a list of Lines Has all the methods to change the document fields Play the whole text in reverse etc 	Collaborations: <ul style="list-style-type: none"> EncodingStrategy TextToSpeechAPI TextToSpeechAPIFactory Line



Class Name: TextToSpeechAPI	
Responsibilities: <ul style="list-style-type: none"> Interface with some methods that <i>FreeTTS</i> and <i>FakeTextToSpeechAPI</i> can implement 	Collaborations: <ul style="list-style-type: none"> <i>FreeTTSAdapter</i> <i>FakeTextToSpeechAPI</i>

Class Name: FreeTTSAdapter	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> To transform the text to speech with the library FreeTTS 	-

Class Name: FakeTextToSpeechAPI	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> Keeps the text we want to transform so we can test it 	-



Class Name: EncodingStrategy	
Responsibilities:	Collaborations:
<ul style="list-style-type: none"> Interface with one method. The abstract class <i>TemplateEncoding</i> will implement 	<ul style="list-style-type: none"> <i>TemplateEncoding</i>

Collaborations:

- *Rot13Encoding*
- *AtBashEncoding*

