SE 302
Principles of Software Engineering

Software Design Document

for

Family Tree Builder

Team 8 - Project Members

Ozan ŞAHİN (20190602036)

Ege ALTIOK (20190602002)

Barış ÖZDİL (20170602023)

Efe KAYLAK (20180602019)

Course Lecturer and Client

Asst. Prof. Kaya OĞUZ

07 December 2021

Table of Contents

I. PREFACE

I.I. Document Version

- Document Version: 1.0

I.II. Change History

- **v1.0**
  - Initial version

II. INTRODUCTION

II.I. Purpose of the Document

- This Software Design Document briefly explains the structure of the components of the Family Tree Project.
- This document will provide a design which will satisfy functional and non-functional requirements that are stated in the Software Requirements Document.

II.II. Scope of the Document

This document shall contain the general definition and features of the project, behaviour of the system, user's interaction with the system.

III. ATTRIBUTES OF DESIGN ENTITIES

III.I. TreeOperations

Variables

- **tree_content** stores the tree content of the tree
- **active_tree** is a GUI component that resembles a tree-like view

Methods

- **create_tree()** gets input from the user and creates a new tree with the input name.
- **add_member()** gets input from user and creates new member with that information.
- **check_relation()** requests user to select two members then checks what relation they have.
- **import_tree()** imports raw tree data and constructs the tree.
- **export_tree()** exports raw tree data to a user selected destination.
- **merge_trees()** requests user to select a new tree to merge with his/her currently selected family tree.
- **save_as_image()** creates an image file of the tree at the user selected destination.
- **update_info_tab()** updates the selected member tab with the information of the member object that user has clicked on.
- **add_filter()** gets input from the user and adds that input as a new filter in the filter list.
- **remove_filter()** removes the selected filter from the filter list.

III.II. Member

Variables

- **name** is the name of the family member
- **surname** is the surname of the family member
- **age** is the age of the family member
- **birthday** is the Birthday object of the family member
- **member_layer_level** is the member's vertical position relative to its root member node (0 -> grandparent, 1 -> parent, 2 -> child)

Methods

- **check_level()** checks and returns the value of member_layer_level attribute.

III.III. Birthday

Variables

- **day** is the day of the birthday
- **month** is the month of the birthday
- **year** is the year of the birthday

Methods

- **date_to_string()** converts the attributes of the Birthday object to string format.

IV. INTERFACE DESCRIPTION

- **Tree Operations Field** allows user to use the operations in the TreeOperation class.

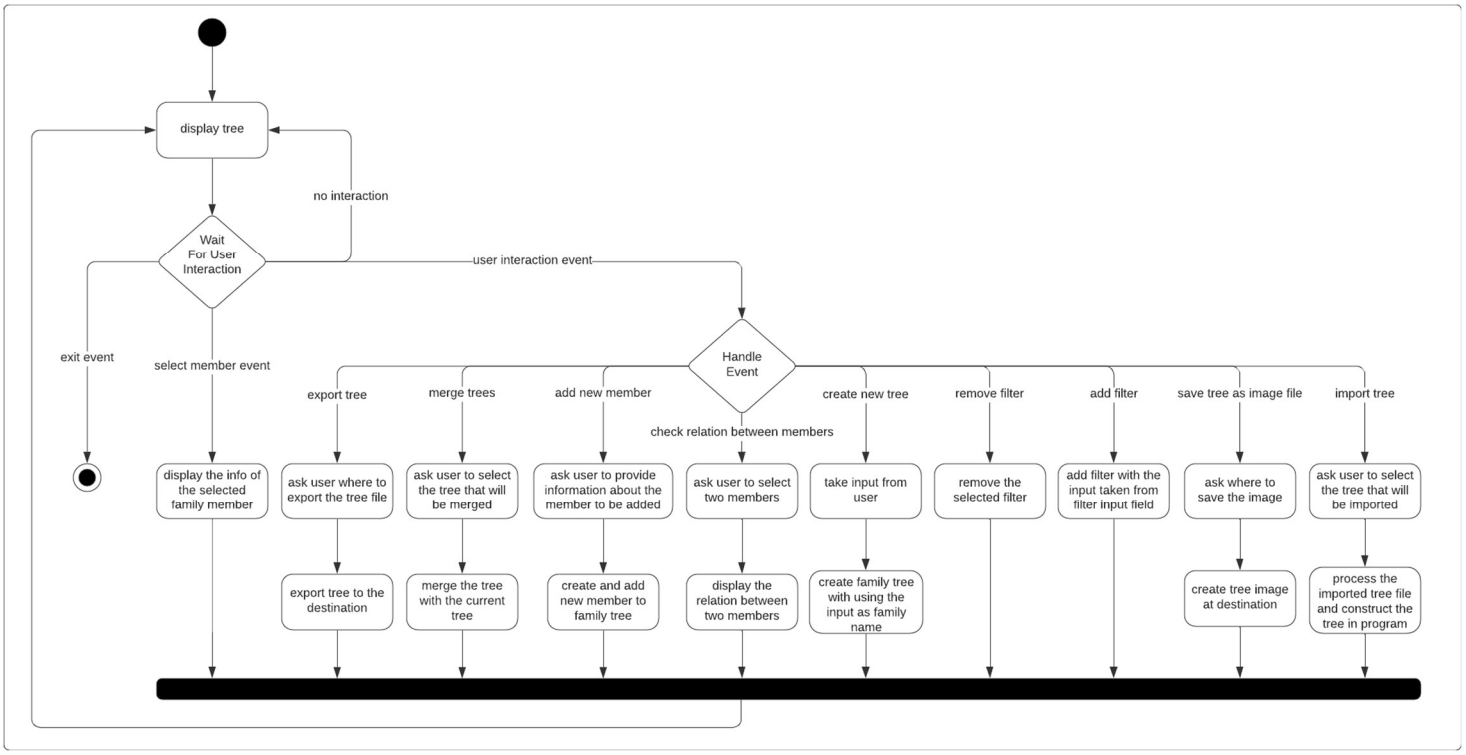
- **Filters Field** allows users to find the people that they are looking for by adding and removing filters.

- **Family Tree Contents** is the field where the family tree contents are displayed. Users can select a person from the family tree.

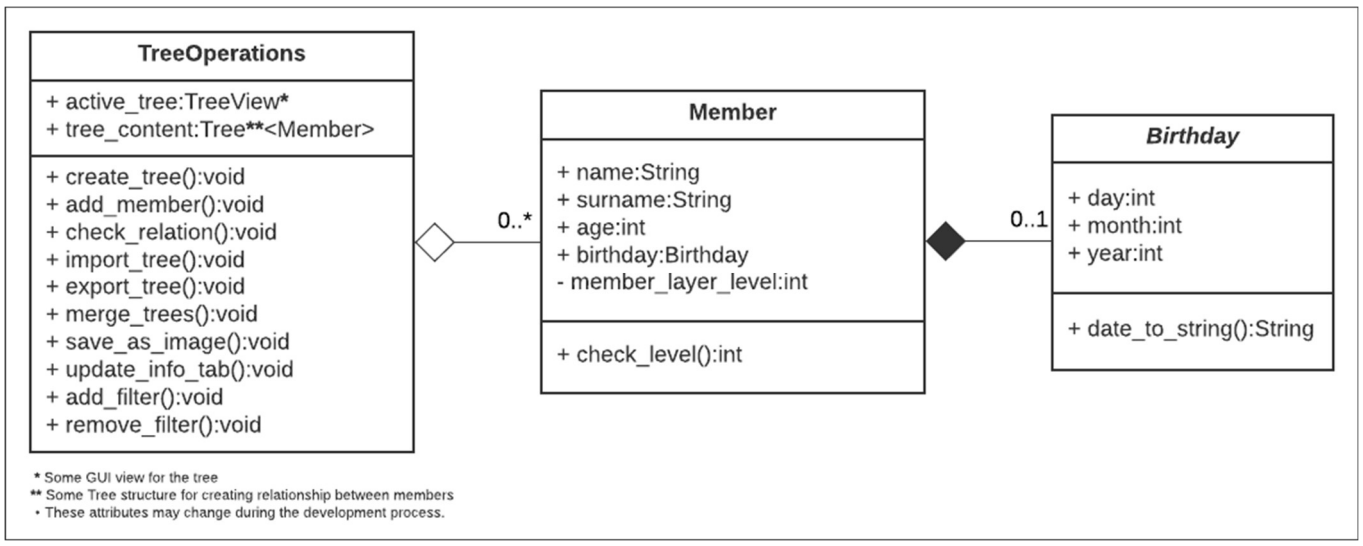
- **Selected Person Info** allows users to see the selected person's information in this field.

V. DECOMPOSITION DESCRIPTION

V.I. Activity Diagram



V.II. Class Diagram



V.III. Use Case Diagram