Bilkent University

Department of Computer Engineering

Object-Oriented Software Engineering Project

CS 319 Project: Tempo

Final Report

Project Group: 1.G

Member Names: Mert Saraç, A. A. M. Jubaeid Hasan Chowdhury, Burak Erkılıç, Kaan Kıranbay

Course Instructor: Eray Tüzün

Progress Report
March 17, 2018

Table of Contents 1

Analysis Report

CS 319 Project : Tempo

1. Introduction

At this point, we have finished the calendar page which is core part of our system. Calendar page uses HTML and JavaScript in the view part so HTML JavaScript wrapper is done to obtain better functionality. It took much time to implement. Also, we have buttons to add and cancel the events, log out button. However, we could not finish the application. It is far from being finished but we are going to work hard on it to finish it. There are some missing parts that we are working on such as authentication page, database, to-do list, profile page, and most importantly smart events.

2. Design Changes

In general, we keep the previous design we have mentioned before but some of the classes, methods and variables are changed in the implementation process. However, the system's architecture is changed in the implementation part since we have realized that we labeled our architecture as MVC but it was not quite like a MVC so we did changes according to this fact. After searching and reading about other architectures, we decided to choose Client/Server architecture since it is suitable for a system with a central database.

In addition, we have added two arrow buttons to display past and future events in the other months. In this way, Tempo reminds an agenda.

There will be more changes in the design because we have just focused on the calendar page and we just considered changes in calendar page but after implement other parts, the relationships among classes and functions will be also revised according to change in our architecture form Model View Controller to Client/Server.

3. Lessons Learnt

First lesson that we have learned is coming a compromise since it is core idea of group work. We have learned it when we are describing our system and specifying functional requirements of the system. It was hard because our system has too much possible features and every participant wanted to focus on different aspect of it. We have also experienced it when we are specifying non-functional requirements.

After having a design, we have distributed roles and parts of the system to group members. In distribution of the roles, we have learned more about the roles and their duties. Also, in distribution of parts, we distributed them according to the architecture that is Client/Server. It is distributed like two person to client and two person to server part.

While the implementation process, we have seen that architecture of the system has too much influence on the classes and the methods. Due to change in our system's architecture, we need to reconsider some functions' and classes' implementations and relationships among them.

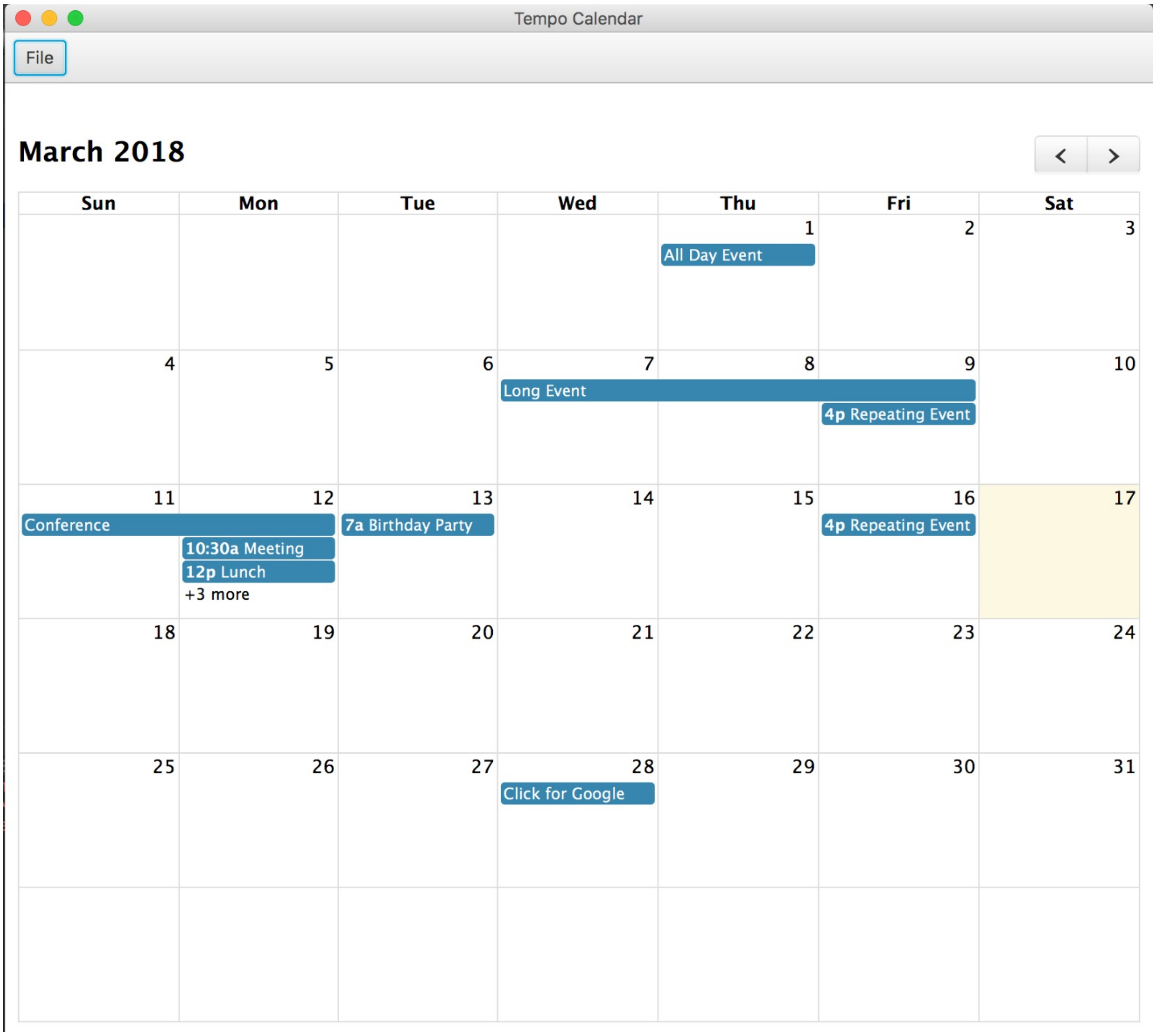
While the implementation process, we learned much about how to use JavaFX, HTML, JavaScript and GitHub more effectively. This helped us to produce much better understanding of what we are going to implement and how we are going to implement it. For JavaFX, HTML and JavaScript, we have watched numerous videos and reads books in the library about these languages.

4. User’s Guide

4.1. System Requirements & Installation

Our program does not require too much system requirements. In order to use Tempo, one needs to have Java in the computer. After that user needs to download a Java IDE such as Eclipse and NetBeans and JDK (especisally latest versions of it). On this IDE, the user needs to run the program which can be downloaded on our GitHub page (Available at https://github.com/jhchowdhury/Tempo). As an alternative way, user can use Tempo by using jar file which will be also available in our GitHub page. Running jar file is enough to use Tempo. In addition, Tempo needs an internet connection until user connects, so user’s computer needs to stay online when using connecting to the system. However, after connecting for once, the user can use the system’s offline features.

4.2. How to Use



March 2018

Sun	Mon	Tue	Wed	Thu	Fri	Sat
				1 All Day Event	2	3
4	5	6	7 Long Event	8	9 4p Repeating Event	10
11 Conference	12 10:30a Meeting 12p Lunch +3 more	13 7a Birthday Party	14	15	16 4p Repeating Event	17
18	19	20	21	22	23	24
25	26	27	28 Click for Google	29	30	31

Figure - 1

After the authentication page (that is not fully implemented), users can access to the calendar page that contains various events that are combination of temporary event , permanent event , smartEvent and personal event. In this page, user can create event by clicking and events can be dragged by mouse. This is the monthly view of calendar page. By using arrow buttons, user can see other months and see the past events.