

**EGE UNIVERSITY**

**FACULTY OF ENGINEERING**

**COMPUTER ENGINEERING DEPARTMENT**

**SOFTWARE ENGINEERING**

**2022–2023 FALL SEMESTER**

**CITY GUIDE APPLICATION**

**ARCHITECTURAL MODEL**

**DELIVERY DATE**

09/12/2022

**PREPARED BY GROUP 32**

05190000072, Tuğcan TOPALOĞLU

05180000070, Doğukan ARGÜÇ

05190000902, Lale Elif YEŞİL

05190000764, Osman CANBOĞA

İçindekiler

[1) Introduction 3](#_Toc120969344)

[2) Defining the Architecture 3](#_Toc120969345)

[3) Block Diagram 3](#_Toc120969346)

[4) Explaining 3](#_Toc120969347)

[5) Small Legend 3](#_Toc120969348)

[6) Conclusion 3](#_Toc120969349)

# Introduction

We will choose and design the architectural model of our project during this section. We will choose the architectural style that best fits the needs of our program and then discuss why it is appropriate. We will describe the relationships, parts, and subsystems we employ. We'll evaluate it in comparison to other architectural designs. We'll describe the entities we employ.

# Defining the Architecture

We chose mvc model. It’s components are model, view and controller. Subsystems are xxx. Their relationships xxx.

# Block Diagram

# Explaining

Layering the subsystems and making them independent from each other is necessary for the correct and stable operation of our system. The "mvc" pattern is our best option, as we need to work more planned and concurrently than usual in our limited time. We didn’t choose the Layered architecture because due to the numerous levels of interpretation in our design, performance issues may arise. We couldn't choose the Repository architecture either as there is no need for storage all in one place. A simple error can affect the entire system, and we don’t want that during tourist travel or any time that she/he wants to eat, drink, buying public transport card, searching for accommodation.

# Small Legend

# Conclusion

The architectural model that best fits our program was selected. We went into great length on why we settled on the MVC approach and why we rejected all other options. With the MVC approach, we created a block diagram and declared our entities.