

<Hotel Reservation System>

Object Design

<1.0>

<02.01.2017>

< Çağla Çınar

Berk Ergun

Serdar Ünlüsoy

Öner Ercan >

Prepared for

SE301 Software Engineering



IŞIK UNIVERSITY
COMPUTER
SCIENCE AND
ENGINEERING

Table of Contents

1. Introduction	3
1.1. Object Design Trade-offs	3
1.2. Interface Documentation Guidelines	3
1.3. Definitions, Acronyms and Abbreviations	3
1.4. References	4
2. Packages	4
3. Class Interfaces	4

OBJECT DESIGN DOCUMENT

1. Introduction

Hotel Reservation System provides people a platform which they can use to find their desired hotels. The system also helps hotel owners to register their hotels so that they can reach the customers easily.

1.1. Object Design Trade-offs

To be able to make this system efficient, reliable and user friendly, we have to make some trade-offs which costs us development time, readability and monetary for clients.

Because there are a lot of functionality provided for users, hotel users, admins and registrants, development time increases drastically. We need to invest more time for these functions to be working properly. Also, increasing code means increasing mess. We need to keep our codes clean and readable so that everyone in the team can look and understand easily. To be able to achieve a more readable coding environment, we need to make use of comments, name our variables, functions and classes more clearly. This process also consumes time which means development progress gets longer. This tedious process also increases the costs for clients since there is more work put into it.

1.2. Interface Documentation Guidelines

Classes

- admin_page
- index
- login
- logout
- mysqli_connect
- register
- search

1.3. Definitions, Acronyms and Abbreviations

- HRS: Hotel Reservation System
- HTML: HyperText Markup Interface
- JS: Javascript
- PHP: Hypertext Preprocessor
- DB: Database
- Server: A server is a computer program or a device that provides functionality for other programs called clients.
- Client: A client is a software that uses the services provided by servers.

- User: A person that is using the system.

1.4. References

1 - Bruegge B. & Dutoit A.H.. (2010). Object-Oriented Software Engineering Using UML, Patterns, and Java, Prentice Hall, 3rd ed.

2 - Instructor's Course Slides

2. Packages

We didn't write any java code so we have no packages.

3. Class Interface

We didn't write any java code so we have no class interfaces.