Dent Light
СТОМАТОЛОГІЧНИЙ КАБІНЕТ

DENT LIGHT PROJECT

TECHNICAL DOCUMENTATION

[dent-light.com]

(2017-2018)

CONTENTS

BRIEF OVERVIEW 3

TECHNOLOGIES 3

PROJECT STRUCTURE 3

USER ROLES AND SYSTEM FUNCTIONALITY 3

USER INTERFACE 4

SECURITY 5

SYSTEM STATISTICS 5

BRIEF OVERVIEW

Dent Light is a client-server web-application developed specially for dentists and their patients, composed by combination of the ordinary website and web-registration system with authorization. The system is designed to register and save patients' personal data and to store their doctor visits' schedule. The system provides an easy interaction between doctors and patients, allowing both sides to follow the schedule and track all changes in real-time environment.
Web-site address: dent-light.com

TECHNOLOGIES

The Dent Light Project was designed in IntelliJIdea IDE and developed using JAVA 8 language and supported with additional frameworks and technologies. Specifications and technologies are listed in a table below:

Table 1. Specifications and technologies.

Programming language (Back-end)	Java 8 (1.8)
Programming language (Front-end)	JavaScript (+JQuery, AJAX)
Architecture	Client-server
Architectural style	REST
Java servlet container (web-server)	Apache Tomcat 8.5
Build framework	Apache Maven
Application framework	Spring 4.3 (MVC)
Security framework	Spring Security 4.3
RDBMS	MySQL 5.7
ORM	Hibernate 5.3
Mailing Service	Java Mail 1.4
CMS	None
Server operating system	Linux OS (Debian 9)
UI libraries/languages	Bootstrap 3, CSS3
Interface language	Ukrainian

PROJECT STRUCTURE

The project is developed to represent the Model-View-Controller (MVC) architecture, so has three-layer structure as follows:
Model (Data-access layer, DAL). Java objects are mapped and persisted to server MySQL database using Hibernate as ORM.
View (Representation, User interface layer, UIL). Composed of java server pages (JSP's), which are used to deliver dynamic HTML pages and to create UI. The native JavaScript, JQuery library and Bootstrap are used in JSP's to provide user's interaction with the system and corporate design.
Controller. Developed to handle URL requests from users and apply to service layer (@Service) to implement the business logic of the application and interact with DAL to store/retrieve DB data.

USER ROLES AND SYSTEM FUNCTIONALITY

The project has a basic CRUD (Create-Read-Update-Delete) functions extended with additional functionality (mailing services, security services).

All users are filtered and divided into 3 security groups ("roles"):

- ***Admin*** (top level registered user with extended privileges, full access to all functionality)
- ***User*** (registered user with basic privileges and limited CRUD rights)
- ***Anonymous*** (other) (any non-registered user with no CRUD privileges, read-only mode for non-secured content available)

Full functionality is available after login/registration based on definite user's role, granted by the system. Admin's role may be assigned manually by system developer in MySQL DB table. After registration and login, ordinary users can get the *User* role only (for security means) and may be upgraded to *Admin* role by changing role code in a relevant DB table.

Mailing service is performed using JavaMail in automatic mode to send e-mail notifications to user's e-mail address (after registration, creation of a new visit, visit confirmation etc.).

Table 2. Roles and functions

ANONYMOUS (any user)

- View non-secured pages of the web-site, which are available without registration/login.

USER (patient)

- View secured and non-secured pages of the web-site.
- Edit user's personal data
- View the schedule calendar with all events (only user's own visits in calendar can be viewed in details, all other visits are restricted)
- Create new visits to doctor (which are marked as "unconfirmed" after creation)
- Edit visits (after edition, the visit event is marked as "unconfirmed")
- Send mail after user's registration and new visit creation (performed automatically)

ADMIN (doctor/administrator)

- All functionality of the role "USER"
- Edit all user's personal data
- View the schedule calendar with all events (all visits in calendar are available to be viewed in details)
- Edit all user's visits
- Confirm user's visits (after this action, the visit event is marked as "confirmed")
- Search for users by user's name/mobile phone number
- Delete users' visits *(feature currently under development)*
- Delegate own visits to other users (if the visit is created by Admin and needs to be transferred to any other registered user) *(feature currently under development)*
- View and edit Black-list of users *(feature currently under development)*
- Send the multiple e-mails (mailouts) with company news/promotions *(feature currently under development)*

USER INTERFACE

User interface (UI) is developed without any content-management system (CMS) on JSP (converted to HTML). The visual effects, color schemes and client-side calculations are made with CSS3, Bootstrap 3 and JavaScript (incl. JQuery library).

The UI is fully adapted for mobile devices (tablets, smartphones) with resolutions up to FullHD (1920 x 1080), adopting the concept of "MOBILE FIRST" for wider user's audience.

Currently, only Ukrainian language is available for UI, but can be extended by multi-language support in next versions (using spring resource bundle).

SECURITY

Web-application is deployed on web-server using HTTPS protocol to encrypt sensitive personal data (incl. e-mails, passwords etc.) and securely send them between client and server. All ordinary http requests from clients are automatically redirected to secured https protocol by Tomcat (currently a Let’s Encrypt SSL certificate is installed). Additional security is implemented by using filters, provided by Spring Security Interceptor. All passwords are encrypted using bcrypt hashing function (current strength level = 12, can be adjusted when necessary).

SYSTEM STATISTICS

The system uses Google Analytics/Search Console integrated code to track and analyze the web-application’s usage statistics and activity.