UML Class Diagram
Group 8 - CS 3307
Stage #2

OurApplication

- title: string
- wholePage: WContainerWidget *
- header: WContainerWidget *
- content: WContainerWidget *
- footer: WContainerWidget *
- user: User
- dialog: WDialog *

- login (login_email: WLineEdit *, login_password: WLineEdit *, msg: WText *)
- signUp (first_name: WLineEdit *, last_name: WLineEdit *, email: WLineEdit *, password: WLineEdit *, passcheck: WLineEdit *, reg_msg: WText *)
- edit (firt_name: WLineEdit *, last_name: WLineEdit *, msg: WText *)
- logout ()
- delete_user ()
- change_password (old_pass: WLineEdit *, new_pass: WLineEdit *, new_check: WLineEdit *, msg: WText *)
- sign_dialog ()
- edit_dialog ()
- hide_dialog ()
- change_dialog ()
- addHeader ()
- addContent ()
- addFooter ()
- showLanding ()
- showHome ()
- showCreateUser ()
- showUserProfile ()

User

- email: string
- hashedPassword: string
- firstName: string
- lastName: string
- registerTime: int
- lastLogin: int
- scheduler: Scheduler
- bridges: vector<Bridge>

+ getEmail (): string
+ getHashedPassword (): string
+ getFirstName (): string
+ getLastName (): string
+ getRegisterTime (): int
+ getLastLogin (): int
+ getScheduler (): Scheduler &
+ setEmail (email: string)
+ setFirstName (firstName: string)
+ setLastName (lastName: string)
+ setRegisterTime (registerTime: int)
+ setLastLogin (lastLogin: int)
+ addToBridges (bridge: Bridge): bool
+ popBridge (): Bridge
+ peekBridges (): Bridge &
+ getBridge (id: int): Bridge &
+ removeBridge (id: int): bool
+ hashPassword (password: string): string
+ comparePassword (rewritePassword: string, originalPassword: string): bool
+ changePassword (originalPassword: string, newPassword: string, rewritePassword: string): int
+ checkUsernameFormat (username: string): int
+ checkPasswordFormat (password: string): int
+ checkNameFormat (name: string): int
+ empty(): bool

Bridge

- id: int
- name: string
- ipAddress: string
- lightBulbs: vector<LightBulb>
- groups: vector<Group>

+ getId (): int
+ getName (): string
+ getIpAddress (): string
+ setId (id: int)
+ setName (name: string)
+ setIpAddress (ipAddress: string)
+ addToLightBulbs (lightBulb: LightBulbs): bool
+ popLightBulbs (): LightBulb
+ peekLightBulbs (): LightBulb &
+ getLightBulb (id: int): LightBulb &
+ removeFromLightBulbs (id: int): bool
+ addToGroups (group: Group): bool
+ popGroup (): Group
+ peekGroups (): Group &
+ getGroup (id: int): Group &
+ removeFromGroups (id: int): bool
+ isFull (): bool

BridgeController

+ sendBridgeRequest (JSONRequest: string): bool
+ sendMassBridgeRequest (JSONRequest: string): bool
+ verifyBridge (toVerify: Bridge &): bool
+ createSingleJSON (lightBulb: LightBulb &, hue: int, saturation: int, on: bool, brightness: int): string
+ createSingleEffectJSON (lightBulb: LightBulb &, hue: int, saturation: int, on: bool, brightness: int, fadeTime: int, onTime: int, offTime: int, cycleSpeed: int): string
+ createGroupJSON (group: Group &, hue: int, saturation: int, on: bool, brightness: int): string
+ createGroupEffectJSON (group: Group &, hue: int, saturation: int, brightness: int, fadeTime: int, onTime: int, offTime: int, cycleSpeed: int): string

FileController

+ userExists (email: string): bool
+ getUser (user: User &, email: string, password: string): int
+ saveUser (user: User &)
+ createUser (user: User &)
+ deleteUser (email: string)
- emailLocation (email: string, dataStr: string): size_t
- getEmailStr (emailStr: string): string
- fileToString (): string
- stringToFile (dataStr: string)
- markFileAccessable (accessable: bool)
- isFileAccessable (): bool
- removeSemaphore (dataStr: string): string
- readFromFile (): string
- writeToFile (dataStr: string)

Group

- lightBulbs: vector<LightBulb *>

+ addToLightBulbs (lightBulb: LightBulb *): bool
+ popLightBulb (): LightBulb &
+ peekLightBulb (): LightBulb &
+ getLightBulb (id: int): LightBulb &
+ removeFromLightBulbs (id: int): bool

LightBulb

- id : int
- name : string
- hue: int
- saturation: int
- brightness: int
- defaultState: int[3]
- on: bool

+ getId (): int
+ getName (): string
+ getHue (): int
+ getSaturation (): int
+ getBrightness (): int
+ getDefaultState (): int[3]
+ getOn (): bool
+ setId (id: int)
+ setName (name: string)
+ setHue (hue: int)
+ setSaturation (saturation: int)
+ setBrightness (brightness: int)
+ setDefaultState (hue: int, saturation: int, brightness: int)
+ setOn (on: bool)
+ setState (hue: int, saturation: int, brightness: int): bool
+ setStateToDefault ()
+ transition (hue: int, saturation: int, brightness: int, fadeTime: int)
+ strobe (onTime: int, offTime: int, hue: int, saturation: int, brightness: int, fadeTime: int)
+ rainbow (cycleSpeed: int, startHue: int, brightness: int)
+ strobeRainbow (onTime: int, offTime: int, startHue: int, brightness: int, fadeTime: int)

Scheduler

- events: priority_queue<Event>
- isActive: bool
- currentTime: time_t

+ getIsActive (): bool
+ getCurrentTime (): time_t
+ setCurrentTime (currentTime: time_t)
+ setIsActive (isActive: bool)
+ enqueueEvent (event: Event): bool
+ dequeueEvent (): Event
+ peekEvents (): Event &
+ getEvent (eventId: int): Event &
+ removeFromEvents (eventId: int): bool
- poll ()
- executeEvent (event: Event &): bool

Event

- eventID: int
- enum Effect {NO_EFFECT, TRANSITION, STROBE, RAINBOW, STROBE_RAINBOW}
- eventEffect: Effect
- hue: int
- saturation: int
- brightness: int
- fadeTime: int
- onTime: int
- offTime: int
- cycleSpeed: int
- executeTime: time_t
- lightbulbs: vector<LightBulb *>
- groups: vector<Group *>

+ getEventID (): int
+ getEventEffect (): int
+ getHue (): int
+ getSaturation (): int
+ getBrightness (): int
+ getFadeTime (): int
+ getOnTime (): int
+ getOffTime (): int
+ getCycleSpeed (): int
+ getExecuteTime (): time_t
+ setEventID (eventID: int)
+ setEventEffect (eventEffect: int)
+ setHue (hue: int)
+ setSaturation (saturation: int)
+ setBrightness (brightness: int)
+ setFadeTime (fadeTime: int)
+ setOnTime (onTime: int)
+ setOffTime (offTime: int)
+ setCycleSpeed (cycleSpeed: int)
+ setExecuteTime (executeTime: time_t)
+ addToLightBulbs (lightBulb: LightBulb *): bool
+ popLightBulb (): LightBulb &
+ peekLightBulb (): LightBulb &
+ getLightBulb (id: int): LightBulb &
+ removeFromLightBulbs (id: int): bool
+ addToGroups (group: Group *): bool
+ popGroups (): Group &
+ peekGroups (): Group &
+ getGroup (id: int): Group &
+ removeFromGroups (id: int): bool